

How to secure general acceptance for a matching? - Inverse Popular Matchings

Erika Bérczi-Kovács^{a,b} Eszter Szabó^b

Submitted: Jan 9, 2025; Accepted: Aug 21, 2026; Published: Oct 9, 2026

© The authors. Released under the CC BY-ND license (International 4.0).

Abstract

Inverse optimization problems focus on minimizing the adjustments made to the input data to achieve a desired outcome. Popular matchings are relaxations of stable matchings that prioritize the overall welfare of society over individual preferences. In this work, we aim to determine a minimum cost set of nodes in order to make a given matching popular by updating preferences at the chosen set. We show that the problem is NP-hard, even if the input matching is perfect or the graph is bipartite. If the desired matching is maximum size in the graph, we present a polynomial time algorithm to determine this minimum in bipartite graphs, and provide a dual characterization.

Mathematics Subject Classifications: 49N45, 90C27, 68Q17, 05C85

1 Introduction

Inverse optimization refers to a methodology that operates in opposition to conventional mathematical optimization techniques. It focuses on minimizing changes to the input data to obtain a desired outcome (see [11], [19]). Inverse optimization approaches are often applied in practical scenarios. For instance, they are applicable to examining market outcomes, such as reconstructing the electricity grid layout based on locational marginal prices [22] or determining competitive retail locations [23]. The application of inverse optimization methods extends to vehicle routing problems [25], forecasting the total power output of an electric vehicle fleet [28], estimating schedule preferences [26], and studying household activity patterns [27].

In this paper, we investigate the question of how to make a given matching “acceptable” for most of the participants if preferences are given. Matching problems with preferences are frequently encountered in daily life, such as medical match [29], school choice

^aHUN-REN-ELTE Egerváry Research Group on Combinatorial Optimization.(erika.bercz-kovacs@ttk.elte.hu)

^bDepartment of Operations Research, ELTE, Eötvös Loránd University, Budapest, Hungary.(szaboeszti@student.elte.hu)

[30] or organ donation [31], and several algorithms have been developed [9]. We can define the fairness of a matching in different ways. A classic approach is stable marriage introduced by Gale and Shapley [6]. However, stability can sometimes be overly restrictive and prioritize individual needs over collective well-being. In this paper, we consider inverse optimization related to a relaxed version of stability representing the common good, called popularity (see [5] for a survey). No matching can beat a popular matching in a one-on-one election. In other words, no fewer people prefer a popular matching over any other matching. The concept of popularity was introduced by Gärdenfors [7]. There are situations in the stable roommates problem where a stable matching does not exist, but a popular one can be provided [13]. It is known that every stable matching is a minimum size popular matching [1]. In a bipartite graph, a maximum-size popular matching can be computed in $\mathcal{O}(mn_0)$ time, where m is the number of edges and n_0 is the size of the smaller partition [14]. However, in a general graph, determining whether a popular matching exists is an NP-complete problem [15].

We aim to modify the preferences “as little as possible” to make a given matching popular. There are several definitions for the phrase “as little as possible”. In the situations that interest us, the desired result may already be predetermined, and the objective is to ensure its acceptance by most of the participants. To achieve this, some individuals can be influenced by bribes, and we aim to determine the minimum total cost of the needed bribes. More precisely, the Inverse Popular Matching problem¹ involves a graph containing strict preference lists on each node and a predetermined matching. Our goal is to minimize the total cost of the nodes that need to modify their preferences in order to make the input matching popular with the updated preference lists. As for the stable variant, it was proven in [17] that the Inverse Stable Roommates problem is equivalent to the minimum vertex cover problem and can be solved in polynomial time for bipartite graphs. However, there are instances where it is enough to change one preference list to make a given matching popular, while we need to modify the preferences at every third node of the graph to make the same matching stable.

Section 2 outlines more details about the Inverse Popular Matching problem. In Section 2.1, we show that the Inverse Popular Matching problem is NP-hard. In reality, we establish a stronger proposition, stating that the problem is NP-hard, even if the input matching is perfect and the cost function is $c \equiv 1$. We demonstrate in Section 3 that the problem is also NP-hard in bipartite graphs with uniform cost function, moreover, even when the maximum degree is bounded by 7. Finally, Section 4 focuses on the case when the graph is bipartite and the input matching is of maximum size within the graph. In this scenario, we present a polynomial time algorithm that can determine the minimum cost of the nodes requiring preference modifications in Section 4.2. Moreover, we provide a minmax characterization of these optimal modifications in Section 4.3.

Previous work

Several studies have examined inverse optimization related to preference changes in stable matching problems. One of them is to improve the man-optimal stable matching by

¹It can be found under a different name in the literature, specifically the Popular Roommates problem, as referenced in [8].

changing at most k men's preference lists "as little as possible" [18]. A man-optimal stable matching is improved if the sum of the partner ranks is increased and every man obtains a partner who is at least as favourable as in the original man-optimal matching. Inoshita et al. presented polynomial time algorithms for both the optimization and decision variants of this problem. Boehmer et al. examined preference modifications in bipartite graphs to obtain a desired outcome, such as making a fixed matching stable or ensuring a fixed matching is the only stable one or finding a stable matching with fixed edges [17]. They investigated different versions of this problem by employing various manipulative actions, identifying polynomial time solvable cases and NP-hard instances. Bérczi et al. examined different modifications in the stable roommates problem, such as deleting nodes and extending preference lists to ensure that the remaining graph admits a stable matching [16]. Moreover, they investigated the problem of modifying the preferences as little as possible to make a fixed matching stable using the l_1 -norm. They showed that the problem is NP-hard in roommates instances and polynomial-time solvable in bipartite graphs. We are not aware of any similar result to the ones above with preference changes in popular matching problems.

2 Inverse Popular Matching Problem

Let $G = (V, E)$ be an undirected graph and let us denote the neighbours of u by $N(u)$ for any vertex $u \in V$. Let $c : V \rightarrow \mathbb{N}$ be a cost function. For a set of nodes $N \subseteq V$, we define the cost of N as the sum of the cost of the elements within the subset, i.e., $c(N)$ is equal to $\sum_{v \in N} c(v)$. Let $<_u$ be a linear order on $N(u) \cup \{\emptyset\}$, called the **preference list** of the node for any $u \in V$. We assume that $\emptyset$ is the least preferred element for any $u \in V$. A pair $(G, <)$ is called an **instance of the Popular Matching problem**.

For a matching M , we denote the pair of u by $M(u)$. If u is an unmatched node in M , then $M(u) = \emptyset$. A node u prefers matching M over M' if $M'(u) <_u M(u)$.

Definition 1. Let $(G, <)$ be an instance of the Popular Matching problem. A matching M is called **more popular** than M' if there are more nodes that prefer M over M' than those preferring M' over M . A matching M is called **popular** if no other matching is more popular than M .

Let us note that a popular matching is not necessarily stable (see Fig. 1).

Our aim is to modify preference lists with minimum cost in order to make a predetermined matching popular.

Definition 2. Let $(G, <)$ be an instance of the Popular Matching problem with a fixed matching M . A set of nodes N is called **eliminative** if it is possible to change the preference lists of the nodes within N , such that M becomes popular.

Problem 3 (Inverse Popular Matching problem). Graph $G = (V, E)$, matching $M \subseteq E$, preference list $<_u$ for each $u \in V$ and a positive cost function $c : V \rightarrow \mathbb{N}$. Find a minimum cost eliminative set, i.e., an eliminative set N minimizing

$$\sum_{v \in N} c(v).$$

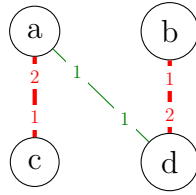


Figure 1: An example of a graph with a popular matching that is not stable. The red and green matchings are both popular; however, only the green matching is stable. On the edges, the number close to a vertex indicates the preference that vertex assigns to the other vertex connected by the edge. Lower numbers indicate a more preferred position.

Note that if $v <_u M(u)$ and $v' <_u M(u)$, the relation between v and v' does not influence the solution to Problem 3. Thus, we have the followings.

Observation 4. *A minimal eliminative set with respect to containment does not contain an unmatched node or a node where the matching edge is originally the most preferred.*

Observation 5. *Changing a preference list of a matched node can be assumed to position the matching edge $vM(v)$ as the most preferred edge of v .*

With respect to the matching M , the following edge categories can be defined.

Definition 6. An edge $uv \in E$ is called

$$\begin{cases} \text{a **negative edge** if } v <_u M(u) \text{ and } u <_v M(v). \\ \text{a **zero edge** if exactly one of the following holds } M(u) <_u v \text{ or } M(v) <_v u. \\ \text{a **blocking edge** if } M(u) <_u v \text{ and } M(v) <_v u. \end{cases}$$

A path in a graph is a sequence of nodes $\langle v_1, v_2 \dots v_k \rangle$, where $v_i \neq v_j$ and $v_i v_{i+1} \in E$ for any $i, j \in \{1, 2 \dots k\}$, $i \neq j$. For a matching M , an **alternating path** is a path whose edges belong alternately to the matching and not to the matching.

We will use the following characterization of popularity proven by Huang and Kavitha [1].

Theorem 7 (Huang and Kavitha). *Let $(G, <)$ be an instance of the Popular Matching problem, and M be a matching in the graph. Let G_M be the graph obtained from G by deleting all negative edges. The matching M is popular if and only if G_M does not contain any of the following structures:*

1. *an alternating cycle with a blocking edge,*
2. *an alternating path connecting two disjoint blocking edges,*
3. *an alternating path connecting a blocking edge with an unmatched node.*

Short proof. If a graph contains one of the above structures, we obtain a more popular matching by swapping edges along that structure. Thus, the matching M is not popular. For the other direction, let us suppose that there is a matching M' in the graph that is

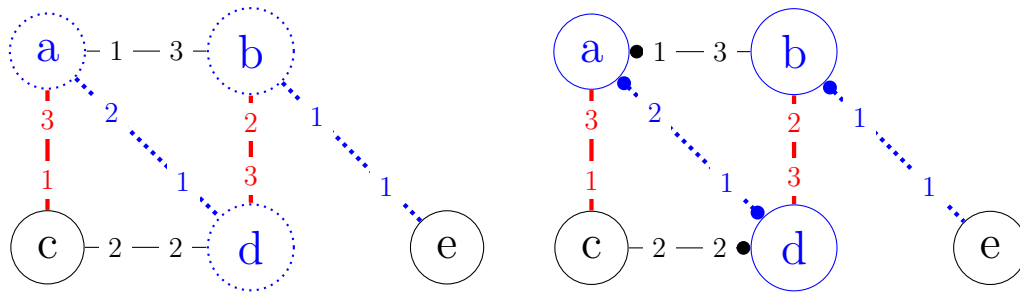


Figure 2: An example of Inverse Popular Matching Problem. On the edges, the number indicates the preference (see the description of Fig. 1). On the left, the path $\langle a, d, b, e \rangle$ is a blocking structure that is eliminated by any non-empty subset of $\{a, d, b\}$. The blocking nodes and edges are colored in blue, while the matching edges are red. On the right, if a circle on an edge is near an endpoint, it indicates that the edge is linked to that node.

more popular than M . In other words, more nodes prefer M' over M . Therefore, there is at least one connected component of the graph $(V, M \cup M')$, where more nodes prefer M' over M . Then, this component has one of the structures described in the statement of the Theorem. $\square$

The structures in Theorem 7 are called **blocking structures**. By Theorem 7, eliminative sets are those that eliminate all the blocking structures in the graph. Thus, in the remainder of this paper, our goal is to find a minimum cost set of nodes that eliminates all blocking structures.

Definition 8. A node v is considered a **blocking node** if it is a matched node and there is a blocking edge connected to it. (See an example in Figure 2.)

For a set of nodes N , let G_M^N denote the graph obtained from G by changing the preference list at every node in N according to Observation 5 and deleting the obtained negative edges.

Definition 9. A set N eliminates a blocking edge uv if it is not blocking in G_M^N . A zero edge or a blocking structure is eliminated by a set N if it is not part of G_M^N . (An edge or a blocking structure is eliminated by a node v if it is eliminated by $\{v\}$.)

Let us note that a complete ordering of preferences is not essential for our results. For each node, it is sufficient to know, for every incident non-matching edge, whether that edge is more preferred over or less preferred than its matching edge. The following definition formalizes this information. We refer to an edge uv in G_M as **linked** to v if $M(v) \neq \emptyset$ and $M(v) <_v u$. Every edge is linked to its endpoint(s) that can eliminate it. Note that a blocking edge connected to an unmatched node is only linked to the matched endpoint. (See an example in Figure 2.)

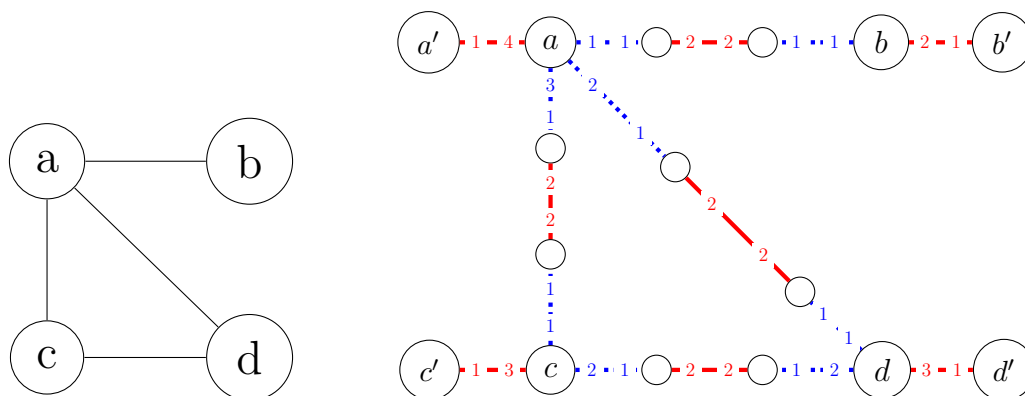


Figure 3: An example of the Minimum Vertex Cover problem and its corresponding instance of Problem 3. The matching edges are red. On the edges, the number indicates the preference (see the description of Fig. 1).

2.1 NP-hardness of the Inverse Popular Matching Problem

In this section, we consider a stronger version of Problem 3, where the matching is a perfect matching and the cost function is uniform. We demonstrate the NP-hardness of this problem. In other words, the following theorem will be proven.

Theorem 10. *The Inverse Popular Matching problem is NP-hard even if the input matching is a perfect matching and $c \equiv 1$.*

Proof. We will establish the theorem from the NP-hardness of the minimum size vertex cover problem [10]. Let $G = (V, E)$ be an undirected graph. A subset of nodes $C \subseteq V$ is a vertex cover if every edge has at least one endpoint in C .

Problem 11 (Minimum Vertex Cover problem). Input: an undirected graph $G = (V, E)$.

Find a vertex cover C that minimizes

$$|C|.$$

Let $G = (V, E)$ be an instance of the Minimum Vertex Cover problem. In order to reduce it to the inverse popular matching problem, we create an auxiliary graph $G^* = (V^*, E^*)$ from G as follows. (See Fig. 3.)

Let us start with a copy of G . We add a new node v' for each node $v \in V$ and connect it with a matching edge vv' . Then, we subdivide every original edge uv using two new points w_{uv} and w'_{uv} , where $w_{uv}w'_{uv}$ is a matching edge for each $uv \in E$. Let us now define the preference lists. For every node, the matching edge is the least preferred edge, and the non-matching edges can be arranged in any order. The cost function $c(u)$ is equal to 1 for any node $u \in V^*$. In conclusion, we obtain the following instance of Problem 3 with

a perfect matching.

$$\begin{aligned} V^* &= \{v, v' \mid v \in V\} \cup \{w_{uv}, w'_{uv} \mid uv \in E\} \\ E^* &= \{vv' \mid v \in V\} \cup \{uw_{uv}, w_{uv}w'_{uv}, w'_{uv}v \mid uv \in E\} \\ M^* &= \{vv' \mid v \in V\} \cup \{w_{uv}w'_{uv} \mid uv \in E\} \end{aligned}$$

It is clear that M^* is a perfect matching in $G^* = (V^*, E^*)$ and all non-matching edge are blocking edges. The blocking structures are only blocking paths $\langle u, w_{uv}, w'_{uv}, v \rangle$ for any edge $uv \in E$.

Claim 12. *There is a minimal eliminative set of nodes that is a subset of V .*

Proof of Claim. By the construction, every edge uw_{uv} or $w'_{uv}v$ is included in exactly one blocking structure, which is the blocking path $\langle u, w_{uv}, w'_{uv}, v \rangle$. Therefore, instead of modifying the preferences at w_{uv} or at w'_{uv} , changing the preference list at u or v eliminates this path as well. By Observation 4, a minimal eliminative set with respect to containment does not contain v' for any $v \in V$. Thus, there is a minimal eliminative set containing only original nodes. ■

Claim 13. *There exists an eliminative set of nodes with cost k in G^* if and only if there is a vertex cover of size k in G .*

Proof of Claim. Let $F \subseteq V'$ be an eliminative set of nodes. Due to Claim 12, we can assume that $F \subseteq V$ holds. Let $uv \in E$ be an arbitrary edge in the original graph G . Since the blocking path $\langle u, w_{uv}, w'_{uv}, v \rangle$ is eliminated by F , either $u \in F$ or $v \in F$ holds. Thus, the set F is a vertex cover in G . Similarly, it can be obtained that any vertex cover is an eliminative set of nodes in $(G^*, M^*, <, 1)$. ■

Changing the preference list at any node in V does not cause any edge to become negative. Furthermore, this modification does not affect the type of edges that are not connected to the altered node. Consequently, any blocking path $\langle u, w_{uv}, w'_{uv}, v \rangle$ will remain a blocking path unless one of its nodes is modified.

Thus, solving Problem 3 in G^* is equivalent to solving the minimum vertex cover problem in G . This finishes the proof of Theorem 10. □

3 Inverse Bipartite Popular Matching Problem

In this section, we consider Problem 3 when the input graph is a bipartite graph $G = (S, T, E)$ and the cost function is uniform, and show that this problem is NP-hard. Namely, the following theorem will be proven.

Theorem 14. *The Inverse Popular Matching problem (Problem 3) is NP-hard even for a bipartite graph and $c \equiv 1$.*

We will reduce from a special version of the 3-SAT problem. In the 1-in-3-SAT+ problem, a 3-CNF formula without negated variables is given, and the aim is to decide whether there exists a satisfying assignment where exactly one variable is true in each clause. This problem is NP-complete [4].

Let Φ be an instance of the 1-in-3-SAT+ problem. Let us denote the variables of Φ by $X_1, X_2, \dots, X_n$ and the clauses by $C_1, C_2, \dots, C_k$, where $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$. Let $cl(i)$ be the set of indices of the clauses that contain X_i . The proof of Theorem 14 will result from Lemma 15. To state this lemma, we first create an auxiliary graph G in the following manner.

For every $i \in \{1, 2, \dots, n\}$, two nodes a_i, b_i and a matching edge $a_i b_i$ are added to G . This matching edge $a_i b_i$ will be the most preferred one for both a_i and b_i . The following nodes and edges are added to G for every $j \in cl(i)$. (See Fig. 4.)

- A path $\langle f_{ij}, g_{ij}, h_{ij}, l_{ij} \rangle$, where the first edge and the last edge are matching edges. The matching edge will be the least favourite one for all of f_{ij}, g_{ij}, h_{ij} and l_{ij} .
- Edge $b_i f_{ij}$ that is linked to f_{ij} . (As we defined before, the matching edge is the most preferred one for b_i , and it is the least favourite one for f_{ij} .)

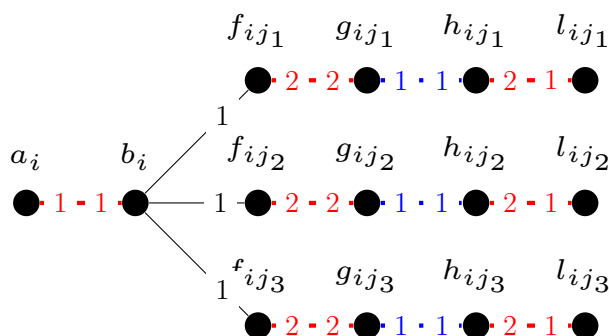


Figure 4: Subgraph of G , where $cl(i) = \{j_1, j_2, j_3\}$. The matching edges are red, while the zero edges are black. On the edges, the number indicates the preference (see the description of Fig. 1).

Next, two identical subgraphs of G are constructed by adding the following nodes and edges to G for any clause $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$. (See Fig 5.)

- Paths $\langle t_{i_k j}, u_{i_k j}, v_{i_k j} \rangle$ and $\langle t'_{i_k j}, u'_{i_k j}, v'_{i_k j} \rangle$ for any $k \in \{1, 2, 3\}$, where the first edges are matching edges, while $v_{i_k j}$ and $v'_{i_k j}$ are uncovered by M . The matching edge will be the most preferred one for $u_{i_k j}$ or $u'_{i_k j}$, and it will be the least favourite one for $t_{i_k j}$ and $t'_{i_k j}$.
- Edges $a_{i_k} t_{i_k j}$ and $a_{i_k} t'_{i_k j}$ that are linked to $t_{i_k j}$ and $t'_{i_k j}$ for any $k \in \{1, 2, 3\}$, respectively. (As we defined before, the matching edge is the most preferred one for a_i , and it is the least favourite one for t_{ij} and t'_{ij} .)

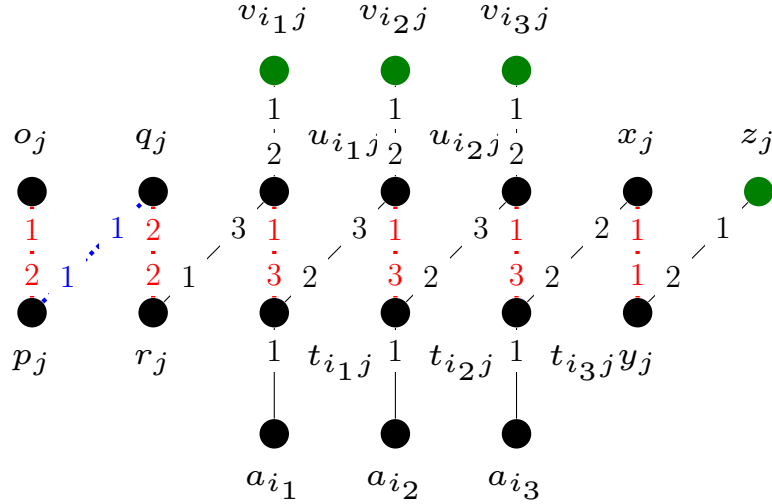


Figure 5: A part of G , where $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$. The matching edges are red, while the zero edges are black, and the blocking edges are blue. The unmatched nodes are green. On the edges, the number indicates the preference (see the description of Fig. 1).

- Edges $u_{i_kj}t_{i_{k-1}j}$ and $u'_{i_kj}t'_{i_{k-1}j}$ for $k \in \{3, 2\}$, that are linked to $t_{i_{k-1}j}$ and $t'_{i_{k-1}j}$, respectively.
- Paths $\langle x_j, y_j, z_j \rangle$ and $\langle x'_j, y'_j, z'_j \rangle$, where the first edges are matching edges and the nodes z_j, z'_j are unmatched. The matching edges x_jy_j and $x'_jy'_j$ will be the most preferred one for all of x_j, y_j, x'_j, y'_j .
- Edges $x_jt_{i_3j}$ and $x'_jt'_{i_3j}$, linked to t_{i_3j} and t'_{i_3j} , respectively.
- Paths $\langle o_j, p_j, q_j, r_j \rangle$ and $\langle o'_j, p'_j, q'_j, r'_j \rangle$, where the first edges and last edges are matching edges. Matching edges $o_jp_j, o'_jp'_j$ and $q_jr_j, q'_jr'_j$ will be the least preferred ones for all of their nodes.
- Edges $u_{i_1j}r_j$ and $u'_{i_1j}r'_j$, that are linked to r_j and r'_j , respectively.

This finishes the building of the auxiliary graph G . Let us consider the subgraph induced by the following nodes for any $i \in \{1, 2 \dots n\}$.

$$\{t_{ij}, t'_{ij}, u_{ij}, u'_{ij}v_{ij}, v'_{ij}, f_{ij}, g_{ij}, h_{ij}, l_{ij} \mid j \in cl(i)\} \cup \{a_i, b_i\}$$

This subgraph is called as being associated with the variable X_i and denoted by G_{X_i} . (See Fig. 6.)

We will define another subgraph within G that is related to a clause C_j and denote this subgraph by G_{C_j} . For a clause $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$, the subgraph G_{C_j} contains the following nodes and the existing edges between them. (See Fig. 7.)

$$\{t_{i_kj}, t'_{i_kj}, u_{i_kj}, u'_{i_kj}v_{i_kj}, v'_{i_kj}, a_{i_k}, b_{i_k}, f_{i_kj}, g_{i_kj}, h_{i_kj}, l_{i_kj} \mid k \in \{1, 2, 3\}\} \cup \{o_j, o'_j, p_j, p'_j, q_j, q'_j, r_j, r'_j, x_j, x'_j, y_j, y'_j, z_j, z'_j\}$$

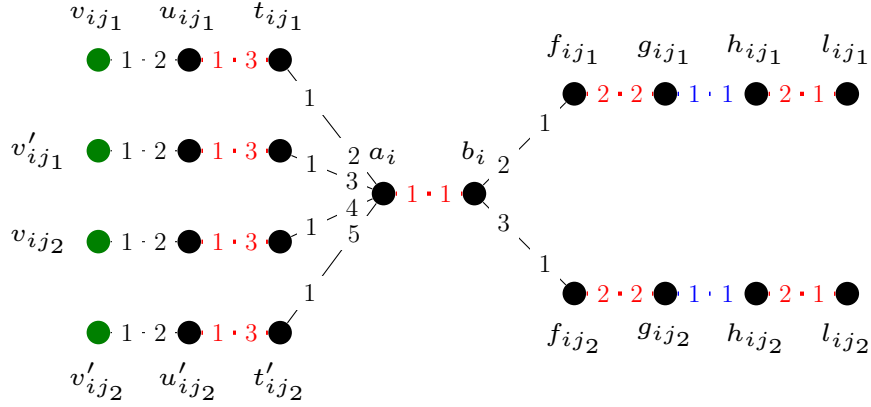


Figure 6: The subgraph G_{X_i} , where $cl(i) = \{j_1, j_2\}$. The matching edges are red, while the zero edges are black, and the blocking edges are blue. The unmatched nodes are green. On the edges, the number indicates the preference (see the description of Fig. 1).

As a result of the construction, the blocking edges of G are the followings.

$$\{g_{ij}h_{ij} \mid j \in \{1, 2, \dots, k, X_i \in C_j\} \cup \{p_jq_j, p'_jq'_j \mid j \in \{1, 2, \dots, k\}\} \quad (1)$$

Finally, the unmatched nodes of G are the following ones.

$$\{v_{ij}, v'_{ij}, z_j, \mid X_i \in C_j\} \quad (2)$$

In order to prove Theorem 14, we will establish the following lemma.

Lemma 15. *The minimum size eliminative set of nodes in G has a size of $4k$ if and only if Φ can be satisfied.*

Proof. First, we will start with the following claim.

Claim 16. *The auxiliary graph G contains only the following blocking structures.*

A blocking path in $G_{C_j} \cup G_{C_l}$ from an unmatched node v_{ij} or v'_{ij} to a blocking edge $g_{il}h_{il}$.

$$\langle v_{ij}^{(i)}, u_{ij}^{(i)}, t_{ij}^{(i)}, a_i, b_i, f_{il}, g_{il}, h_{il} \rangle \quad \text{for any } i \in \{1, 2, \dots, n\}, j, l \in cl(i) \quad (3)$$

A blocking path in $G_{C_j} \cup G_{C_l}$ from an unmatched node $v_{i_1j}^{(i)}$ or $v_{i_2j}^{(i)}$ to a blocking edge $g_{i_2l}h_{i_2l}$ or $g_{i_3l}h_{i_3l}$, respectively.

$$\langle v_{i_k-1j}^{(i)}, u_{i_k-1j}^{(i)}, t_{i_k-1j}^{(i)}, u_{i_kj}^{(i)}, t_{i_kj}^{(i)}, a_{i_k}, b_{i_k}, f_{i_kl}, g_{i_kl}, h_{i_kl} \rangle \quad (4)$$

for any $i \in \{1, 2, \dots, n\}, j, l \in cl(i_k), k \in \{2, 3\}$

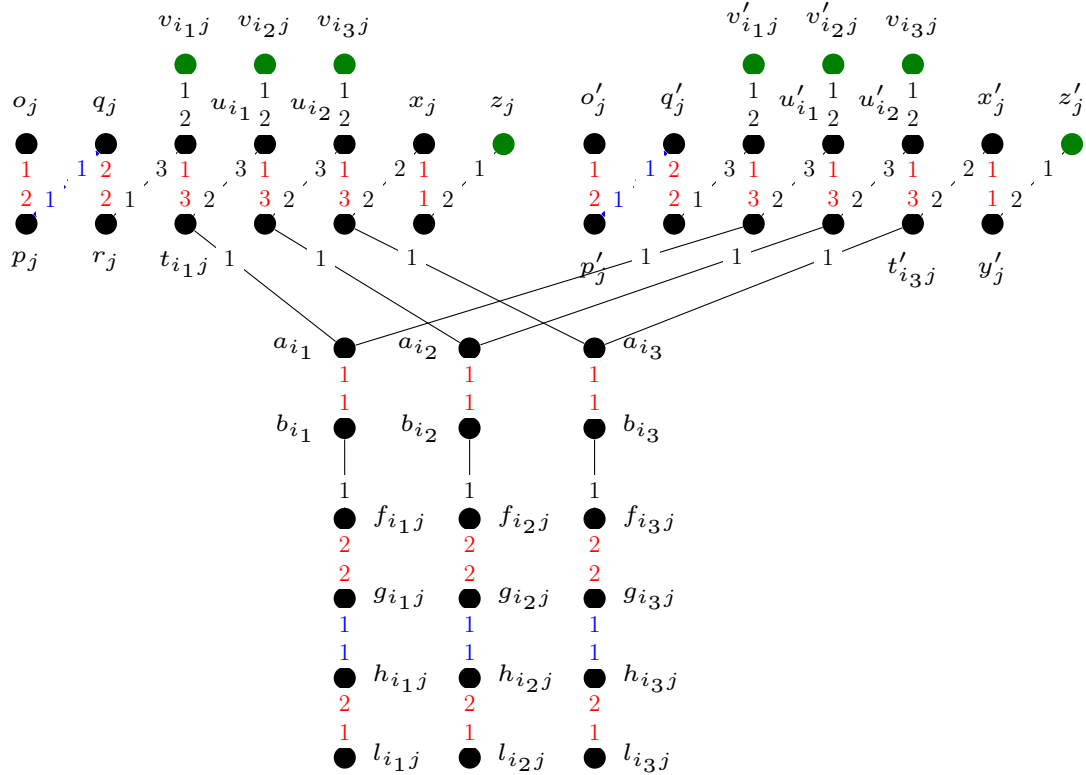


Figure 7: Subgraph G_{C_j} , where $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$. The matching edges are red, while the zero edges are black, and the blocking edges are blue. The unmatched nodes are green. On the edges, the number indicates the preference (see the description of Fig. 1).

A blocking path in $G_{C_j} \cup G_{C_l}$ from an unmatched node v_{i_1j} or v'_{i_1j} to a blocking edge $g_{i_3l}h_{i_3l}$.

$$\langle v_{i_1j}^{(l)}, u_{i_1j}^{(l)}, t_{i_1j}^{(l)}, u_{i_2j}^{(l)}, t_{i_2j}^{(l)}, u_{i_3j}^{(l)}, t_{i_3j}^{(l)}, a_{i_3}, b_{i_3}, f_{i_3l}, g_{i_3l}, h_{i_3l} \rangle \quad \text{for any } j, l \in cl(i_3) \quad (5)$$

A blocking path in $G_{C_j} \cup G_{C_l}$ from a blocking edge p_jq_j or $p'_jq'_j$ to another blocking edge $g_{il}h_{il}$.

$$\langle p_j^{(l)}, q_j^{(l)}, r_j^{(l)}, u_{i_1j}^{(l)}, t_{i_1j}^{(l)}, a_{i_1}, b_{i_1}, f_{i_1l}, g_{i_1l}, h_{i_1l} \rangle \quad \text{for any } j, l \in cl(i_1) \quad (6)$$

$$\langle p_j^{(l)}, q_j^{(l)}, r_j^{(l)}, u_{i_1j}^{(l)}, t_{i_1j}^{(l)}, u_{i_2j}^{(l)}, t_{i_2j}^{(l)}, a_{i_2}, b_{i_2}, f_{i_2l}, g_{i_2l}, h_{i_2l} \rangle \quad \text{for any } j, l \in cl(i_2) \quad (7)$$

$$\langle p_j^{(l)}, q_j^{(l)}, r_j^{(l)}, u_{i_1j}^{(l)}, t_{i_1j}^{(l)}, u_{i_2j}^{(l)}, t_{i_2j}^{(l)}, u_{i_3j}^{(l)}, t_{i_3j}^{(l)}, a_{i_3}, b_{i_3}, f_{i_3l}, g_{i_3l}, h_{i_3l} \rangle \quad \text{for any } j, l \in cl(i_3) \quad (8)$$

A blocking path in G_{C_j} from a blocking edge p_jq_j or $p'_jq'_j$ to an unmatched node z_j or z'_j , respectively.

$$\langle p_j^{(l)}, q_j^{(l)}, r_j^{(l)}, u_{i_1j}^{(l)}, t_{i_1j}^{(l)}, u_{i_2j}^{(l)}, t_{i_2j}^{(l)}, u_{i_3j}^{(l)}, t_{i_3j}^{(l)}, x_j^{(l)}, y_j^{(l)}, z_j^{(l)} \rangle \quad \text{for any } \quad (9)$$

Proof of Claim. It is clear that these paths are blocking paths.

Let B be a blocking structure that contains $a_i b_i$ for any $i \in \{1, 2, \dots, n\}$. By the construction, $a_i b_i$ is an internal edge in B . Then, there is a $j \in cl(i)$, such that B also contains $t_{ij} a_i$ or $t'_{ij} a_i$. Similarly, there is $l \in cl(i)$, such that $b_i f_{il}$ is also part of B . Therefore, B ends in the blocking edge $g_{il} h_{il}$. Because every path that starts with the edge $h_{il} g_{il}$ and ends at $z_j^{(n)}$ has even length (including the edge $h_{il} g_{il}$), there is no alternating path between them. Therefore, the other end of B is either the blocking edge $p_j^{(n)} q_j^{(n)}$ or an unmatched node $v_{i_k j}^{(n)}$ for $X_{i_k} \in C_j$. Thus, B is one of the blocking paths above, and there is no other blocking structure in the graph that contains $a_i b_i$ for any $i \in \{1, 2, \dots, n\}$.

Let B be a blocking structure in G , such that it does not contain $a_i b_i$ for any $i \in \{1, 2, \dots, n\}$. Every blocking structure containing $g_{ij} h_{ij}$ for some $i \in \{1, 2, \dots, n\}, j \in \{1, 2, \dots, k\}$ also includes the matching edge $a_i b_i$. Thus, B does not contain $g_{ij} h_{ij}$. It is clear that every blocking structure contains at least one blocking edge, i.e., at least one edge from (1). Therefore, B contains either $p_j q_j$ or $p'_j q'_j$, where $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$.

Since there is no cycle in G containing $p_j q_j$ or $p'_j q'_j$, there is no blocking cycle in the graph. Therefore, B is a path that starts with $p_j^{(n)} q_j^{(n)}$. Since $a_i b_i \notin B$, it ends at an unmatched node that is either $z_j^{(n)}$ or $v_{i_k j}^{(n)}$ for $k \in \{1, 2, 3\}$. Because every path that starts with the edge $p_j^{(n)} q_j^{(n)}$ and ends at $v_{i_k j}^{(n)}$ has even length (including the edge $p_j^{(n)} q_j^{(n)}$) for any $k \in \{1, 2, 3\}$, there is no alternating path between them. Therefore, P ends at $z_j^{(n)}$ and it is a path described in (9). ■

As a consequence, we get the following.

Observation 17. Let B be a blocking structure in the auxiliary graph G . Then, B contains either $\langle t_{ij}, a_i, b_i, f_{il}, g_{il}, h_{il} \rangle$ or $\langle t'_{ij}, a_i, b_i, f_{il}, g_{il}, h_{il} \rangle$ as a subpath for some $j, l \in cl(i)$, or B is one of the following blocking paths, where $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$.

$$\begin{aligned} &\langle p_j, q_j, r_j, u_{i_1 j}, t_{i_1 j}, u_{i_2 j}, t_{i_2 j}, u_{i_3 j}, t_{i_3 j}, x_j, y_j, z_j \rangle \\ &\langle p'_j, q'_j, r'_j, u'_{i_1 j}, t'_{i_1 j}, u'_{i_2 j}, t'_{i_2 j}, u'_{i_3 j}, t'_{i_3 j}, x'_j, y'_j, z'_j \rangle \end{aligned} \quad (10)$$

Let us examine the minimal eliminative sets within G .

Claim 18. There exists a minimum size eliminative set of nodes in G that is part of

$$\{t_{ij}, t'_{ij}, f_{ij} \mid i \in \{1, 2, \dots, n\}, j \in cl(i)\}.$$

Proof of Claim. By Observation 4, a minimum size eliminative set does not contain the following nodes.

$$\{a_i, b_i, u_{ij}^{(n)}, v_{ij}^{(n)}, k_{ij}, o_j^{(n)}, r_j^{(n)}, x_j^{(n)}, y_j^{(n)}, z_j^{(n)} \mid i \in \{1, 2, \dots, n\}, j \in cl(i)\}.$$

By Observation 17, if a blocking structure contains the blocking edge $g_{ij} h_{ij}$, then $b_i f_{ij}$ is also part of this blocking structure for any $i \in \{1, 2, \dots, n\}, j \in cl(i)$. Therefore, f_{ij} eliminates the same blocking structures as g_{ij} or h_{ij} . Then we can assume that a minimum size eliminative set of nodes does not contain g_{ij} or h_{ij} for any $i \in \{1, 2, \dots, n\}, j \in cl(i)$.

Similarly, by Claim 16, if a blocking structure contains the blocking edge $p_j q_j$ or $p'_j q'_j$, then t_{i1j} or t'_{i1j} is also part of it and t_{i1j} or t'_{i1j} eliminates this blocking structure. Thus, we can assume that $p_j^{(i)}, q_j^{(i)}$ are not part of a minimum size eliminative set for any $j \in \{1, 2, \dots, k\}$. ■

From now, let K be a minimal eliminative set of nodes, such that

$$K \subseteq \{t_{ij}, t'_{ij}, f_{ij} \mid i \in \{1, 2, \dots, n\}, j \in cl(i)\}. \quad (11)$$

Claim 19. *At least one of the following holds true for any $i \in \{1, 2, 3 \dots n\}$.*

$$\{t_{ij}, t'_{ij} \mid j \in cl(i)\} \subseteq K \quad (12)$$

or

$$\{f_{ij} \mid j \in cl(i)\} \subseteq K \quad (13)$$

Proof of Claim. Let us assume that $\{f_{ij} \mid j \in cl(i)\} \not\subseteq K$ for an index $i \in \{1, 2, \dots, n\}$, otherwise, the Lemma holds true. Thus, there exists an index $l \in cl(i)$, such that $f_{il} \notin K$. To finish the proof, it is enough to show that $\{t_{ij}, t'_{ij} \mid j \in cl(i)\} \subseteq K$. Let us indirectly assume the existence of an index $j \in cl(i)$, such that $t_{ij} \notin K$. The situation when $t'_{ij} \notin K$ can be handled similarly. Let us consider the following blocking path.

$$P = \langle v_{ij}, u_{ij}, t_{ij}, a_i, b_i, f_{il}, g_{il}, h_{il} \rangle$$

By the indirect assumption and the condition of (11), the path P does not intersect K . Thus, P is not eliminated by K , which is a contradiction, since K is an eliminative set of nodes. ■

Claim 20. *Let $K_j = K \cap G_{C_j}$, where $j \in \{1, 2, 3 \dots k\}$. Then $|K_j| \geq 4$ holds and $|K_j| = 4$ if and only if K_j is one of the following sets.*

$$\{t_{i1j}, t'_{i1j}, f_{i2j}, f_{i3j}\} \quad (14)$$

$$\{f_{i1j}, t_{i2j}, t'_{i2j}, f_{i3j}\} \quad (15)$$

$$\{f_{i1j}, f_{i2j}, t_{i3j}, t'_{i3j}\} \quad (16)$$

Proof of Claim. Since K is an eliminative set of nodes, it eliminates all the blocking structures within G_{C_j} . In particular, it eliminates the following four disjoint blocking structures in G_{C_j} .

- $\langle v_{i1j}, u_{i1j}, t_{i1j}, a_{i1}, b_{i1}, f_{i1j}, g_{i1j}, h_{i1j} \rangle$
- $\langle v_{i2j}, u_{i2j}, t_{i2j}, a_{i2}, b_{i2}, f_{i2j}, g_{i2j}, h_{i2j} \rangle$
- $\langle v_{i3j}, u_{i3j}, t_{i3j}, a_{i3}, b_{i3}, f_{i3j}, g_{i3j}, h_{i3j} \rangle$
- $\langle p'_j, q'_j, r'_j, u'_{i1j}, t'_{i1j}, u'_{i2j}, t'_{i2j}, u'_{i3j}, t'_{i3j}, x'_j, y'_j, z'_j \rangle$

Therefore, K_j contains at least one node from each of these paths. This proves the first part of the claim.

For the second part, let us assume that $|K_j| = 4$. Due to Claim 19, t_{ikj}, t'_{ikj} or f_{ikj} are part of K_j for any $k \in \{1, 2, 3\}$. Let q be the size of $\{f_{i1j}, f_{i2j}, f_{i3j}\} \cap K_j$.

Case 1: $q = 3$

Thus, $\{f_{i1j}, f_{i2j}, f_{i3j}\} \subseteq K_j$ holds. The following two blocking paths are disjoint and do not include any of these three nodes.

- $\langle p_j, q_j, r_j, u_{i1j}, t_{i1j}, u_{i2j}, t_{i2j}, u_{i3j}, t_{i3j}, x_j, y_j, z_j \rangle$
- $\langle p'_j, q'_j, r'_j, u'_{i1j}, t'_{i1j}, u'_{i2j}, t'_{i2j}, u'_{i3j}, t'_{i3j}, x'_j, y'_j, z'_j \rangle$

Since K_j is eliminative, it contains at least one node from each of these paths and $|K_j| \geq 5$. This is a contradiction, because $|K_j| = 4$.

Case 2: $q \leq 1$

Without loss of generality, we can assume that $f_{i1j}, f_{i2j} \notin K_j$. By Claim 19, we get that $\{t_{i1j}, t'_{i1j}, t_{i2j}, t'_{i2j}\} \subset K_j$. Moreover, either $\{t_{i3j}, t'_{i3j}\} \subset K_j$ or $f_{i3j} \in K_j$ holds. Thus, the size of K_j is at least 5, which is a contradiction.

Therefore, the only possible case is when $q = 2$. Then, by Claim 19, K_j is one of the sets described in (14)-(16). It is clear that these subsets eliminate the blocking structures in G_{C_j} . ■

Using (11) and Claim 20, we obtain the following.

Corollary 21. *The subsets $K \cap G_{C_j}$ and $K \cap G_{C_l}$ are disjoint for any $j, l \in \{1, 2, 3 \dots k\}$ and*

$$|K| \geq 4k,$$

where k is the number of clauses in Φ .

Using Claim 20 and Corollary 21, we obtain the following.

Observation 22. *If K is a minimal eliminative set of nodes as in (11), $|K| = 4k$, then $K \cap G_{C_j}$ is one of the subsets in (14)-(16) for any $j \in \{1, 2, 3 \dots k\}$.*

Claim 23. *If there exists an eliminative set of size $4k$, then Φ can be satisfied.*

Proof of Claim. Let K be an eliminative set of size $4k$. By Claim 18, let us assume that (11) holds.

Let us fix an index $i \in \{1, 2, 3 \dots n\}$. Due to Observation 22, whenever $t_{ij}, t'_{ij} \in K$ holds for some $j \in cl(i)$, then it follows that $f_{ij} \notin K$. Thus, using Claim 18, we have that (12) is satisfied. This implies that for any $j \in cl(i)$, we have that $t_{ij}, t'_{ij} \in K$ and $f_{ij} \notin K$ according to Observation 22. A similar statement is true whenever $f_{ij} \in K$ holds for some $j \in cl(i)$.

Therefore, exactly one of the conditions (12) and (13) holds for any $i \in \{1, 2, 3 \dots n\}$. Let us define the following assignment to the variables in Φ . If condition (12) holds, then the variable X_i is assigned to "TRUE"; otherwise, it is assigned to "FALSE". By Observation 22, $K \cap G_{C_j}$ is one of the subsets (14)-(16) for any $j \in \{1, 2, 3 \dots k\}$. Therefore,

exactly one variable is assigned to TRUE in each clause, and the assignment defined above is a satisfying assignment for Φ . ■

Claim 24. *If there exists a satisfying assignment for Φ , then there exists an eliminative set of nodes of size $4k$.*

Proof of Claim. Let us consider a satisfying assignment for Φ and define the following two subsets of indices.

$$\begin{aligned} T &= \{i \mid X_i \text{ is assigned to "TRUE"}\} \\ F &= \{i \mid X_i \text{ is assigned to "FALSE"}\} \end{aligned}$$

Let us consider the following set of nodes in G .

$$K = \{t_{ij}, t'_{ij} \mid i \in T, j \in cl(i)\} \cup \{f_{ij} \mid i \in F, j \in cl(i)\}$$

Since there is exactly one variable assigned to “TRUE” in every clause, we have $|K \cap G_{C_j}| = 4$ for any $j \in \{1, 2, \dots, k\}$. Since $K \cap G_{C_j}$ and $K \cap G_{C_l}$ are disjoint for any $j, l \in \{1, 2, 3 \dots k\}$, the size of K is $4k$.

In the following, we will show that K is an eliminative set of nodes in G . Let B be a blocking structure in G , and we show that it is eliminated by K .

Due to Observation 17, if B contains the edge $a_i b_i$ for $i \in \{1, 2, \dots, n\}$, then $t_{ij}^{(\iota)} a_i$ and $b_i f_{il}$ are also part of B for some $j, l \in cl(i)$. By the definition of K , either $f_{il} \in K$ or $t_{ij}, t'_{ij} \in K$ holds depending on whether X_i is assigned to TRUE or FALSE. Therefore, B is eliminated by K .

The remaining case is when B does not contain $a_i b_i$ for any $i \in \{1, 2, \dots, n\}$. By Observation 17, B is one of the paths described in (10).

Since one of $X_{i_1}, X_{i_2}, X_{i_3}$ is assigned to “TRUE”, we have that one of $\{t_{i_1 j}^{(\iota)}, t_{i_2 j}^{(\iota)}, t_{i_3 j}^{(\iota)}\}$ is part of K and B is eliminated. Therefore, K eliminates every blocking structure in G .

In summary, the set K is an eliminative set of size $4k$, and this finishes the proof of Claim 24. ■

By combining Corollary 21 with Claims 23 and 24, we obtain Lemma 15, i. e., there is an eliminative set of size $4k$ if and only if Φ can be satisfied. □

Proof of Theorem 14. Lemma 15, provides a polynomial-time reduction from the 1-in-3-SAT+ problem to Problem 3 with a bipartite graph and a uniform cost function. As the 1-in-3-SAT+ problem is NP-hard, it proves the NP-hardness of the Popular Matching Problem with a bipartite graph and a uniform cost function, i. e., Theorem 14 is proven. □

Let us note that, the path $\langle v_{i_3 j}, u_{i_3 j}, t_{i_3 j}, x_j, y_j, z_j \rangle$ is an augmenting path for the constructed matching for any $C_j = (X_{i_1} \vee X_{i_2} \vee X_{i_3})$. (see Fig. 5.) Thus, we get the following observation.

Observation 25. *The constructed matching is not a maximum size matching in the graph G .*

Finally, let us consider the Planar Cubic 1-in-3 SAT+ problem. In this problem, a 3-CNF formula without negated variables is given, where each variable appears in exactly three clauses, and the incidence graph is planar. The objective is to determine whether a satisfying assignment exists such that exactly one variable is true in each clause. This problem is known to be NP-hard [2]. When every variable appears in exactly three clauses, the maximum degree of our constructed auxiliary graph is 7. Consequently, the following statement can be established analogously to Theorem 14, using the NP-hardness of Planar Cubic 1-in-3 SAT+. SAT+ problem.

Corollary 26. *The Inverse Popular Matching problem (Problem 3) is NP-hard even when restricted to bipartite graphs with maximum degree 7 and $c \equiv 1$.*

4 Inverse Bipartite Popular Matching Problem with Maximal Matching

In the proof of Theorem 14, the constructed matching is not a maximum size matching in the graph. This section presents a polynomial algorithm to find a minimum cost eliminative set in bipartite graphs provided that the input matching is of maximum size. Namely, we solve the following problem.

Problem 27 (Inverse Bipartite Popular Matching problem with maximal matching). Input: bipartite graph $G = (S, T, E)$, maximum size matching $M \subseteq E$, preference list $<_u$ for each $u \in S \cup T$ and cost function $c : S \cup T \rightarrow \mathbb{N}$. Find a minimum cost eliminative set of nodes.

To begin with, we examine the problem and the modifications that occur on the graph G_M when a preference list is altered. Then, the problem is converted into a more practical equivalent form to discuss the algorithm (see Sec. 4.1). Subsequently, the algorithm is addressed in Section 4.2. The core idea of this method is to create a network graph with a source s and a sink t , then establish that the minimum s - t cut of this graph is equivalent to the optimal solution of the above mentioned transformed version of the Problem 27.

Let $(G, M, <, c)$ be an instance of Problem 27, where $G = (S \cup T, E)$ is a bipartite graph, $M \subseteq E$ is a maximum size matching in G and $c(v) > 0$ for any $v \in S \cup T$. By Theorem 7, a matching M is popular if and only if there is no blocking structure in G_M . Due to Observation 4, only the matched nodes are considered to be part of an eliminative set. Let us recall that G_M^K denotes the graph obtained by modifying the preferences in K according to Observation 5 and deleting the negative edges. Let us examine the differences between G_M and G_M^K . When an edge becomes negative, it is deleted from the graph G_M . A zero edge is removed if its endpoints are matched and the preference list of the linked endpoint is changed. Note that a zero edge connected to an unmatched node is not linked to any of its endpoints, and it can never be deleted. A blocking edge is deleted from

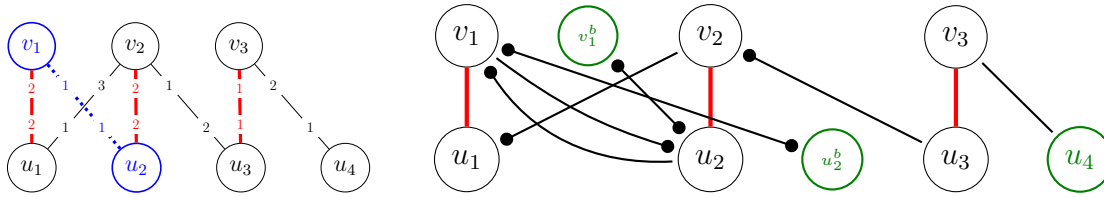


Figure 8: An example of Inverse Bipartite Popular Matching Problem with maximal matching and its transformed problem. In the original problem (left), the blocking nodes and edges are colored in blue, while the matching edges are red. On the edges, the number indicates the preference (see the description of Fig. 1). In the transformed problem (right), the nodes in $X_S \cup X_T$ are colored in green, while the matching edges are red. If a black circle on an edge is near an endpoint, it indicates that the edge is linked to that node.

G_M when both of its endpoints are matched, and they are part of K . A blocking edge uv turns into a zero edge if the preference list is modified at a matched endpoint. If the other endpoint is also a matched node, the obtained zero edge is linked to it. In summary, changing the preferences at a matched node v results in the following modifications on G_M .

- Zero edges linked to v are deleted from G_M .
- A blocking edge uv is transformed into a zero edge. If u is matched, the obtained zero edge uv is linked to u . If u is unmatched, the obtained zero edge is not linked to any of its endpoints.

For a set of nodes K , the graph G_M^K is obtained by applying the modifications described above to G_M for each node in K one by one. Using these observations, we can rephrase the problem without preferences as follows.

4.1 Transformation of the problem

In this section, we will transform an instance of Problem 27 into a more useful form for the algorithm. The goal is to formulate an equivalent problem without preferences, only using the information that which endpoint(s) of an edge can eliminate it, presented through linking. To define this converted problem, an auxiliary graph G' is created below for $(G, M, <, c)$. (See Fig. 8)

Let B and U denote the blocking and unmatched nodes of G , respectively. Note that B and U are disjoint and $B \subseteq V(M)$. The process of forming the auxiliary graph G' begins with copying all nodes from G and including all matching edges from M . Let us denote them as v' for any $v \in S \cup T$ and M' , respectively. The blocking nodes are duplicated, and their duplicate is denoted by v^b . The key is to represent each blocking node v by a pair of nodes (v', v^b) in G' and show that an eliminative set in G can be represented by a set in G' containing at most one node from each pair. The cost of each

node in G' is equal to the cost of the original node in G .

$$\begin{aligned}c'(v') &= c(v) \text{ for any } v \in S \cup T \\c'(v^b) &= c(v) \text{ for any } v \in B\end{aligned}$$

Hence, the nodes and matching of G' are the following.

$$\begin{aligned}S' &= \{v' \mid v \in S\} \cup \{v^b \mid v \in S \cap B\} \\T' &= \{u' \mid u \in T\} \cup \{u^b \mid u \in T \cap B\} \\M' &= \{u'v' \mid uv \in M\}\end{aligned}$$

The following notations for specific sets of nodes in $S' \cup T'$ will be useful.

$$\begin{aligned}B'_S &= \{v^b \mid v \in S \cap B\} \\B'_T &= \{v^b \mid v \in T \cap B\} \\U'_S &= \{v' \mid v \in S \cap U\} \\U'_T &= \{v' \mid v \in T \cap U\}\end{aligned}$$

We will call $B'_S \cup B'_T$ the blocking nodes of G' . The non-matching edges in G' are linked to one, both or none of their endpoints, and they are defined as follows. Recall that the purpose of linking is to sign those endpoint(s) that can eliminate the edge from G_M by the change of preferences.

- For any zero edge $uv \in E$:
 - Edge $u'v'$ is linked to v' if $u, v \in V(M)$ and uv is linked to v .
 - Edge $u'v'$ is linked to none of its endpoints if $v \in U$.
- For any blocking edge $uv \in E$:
 - Two parallel edges $u'v'$, one is linked to v' and the other $u'v'$ is linked to u' and edges $u'v^b$ and u^bv' linked to both endpoints if $u, v \in V(M)$.
 - Edge $u'v'$ is linked to v' , edge $u'v'$ linked to none of its endpoints and edge $u'v^b$ linked to v^b if $u \in U$.

In summary, the non-matching edges of G' are the following. (Linkings are indicated by underlines.)

$$\begin{aligned}E' &= \{u'\underline{v'} \mid u, v \in V(M) \text{ and } \underline{uv} \in E \text{ is a zero edge}\} \cup \\&\quad \{u'v' \mid v \in U \text{ and } uv \in E \text{ is a zero edge}\} \cup \\&\quad \{u'\underline{v'}, \underline{u'}v', \underline{u'}v^b, \underline{u^b}v' \mid u, v \in V(M) \text{ and } \underline{uv} \in E \text{ is a blocking edge}\} \cup \\&\quad \{u'\underline{v'}, u'v', u'\underline{v^b} \mid u \in U \text{ and } \underline{uv} \in E \text{ is a blocking edge}\}\end{aligned}$$

Now we discuss how the popularity of M can be captured in G' .

Due to the construction of G' and the fact that there is no alternating path connecting two unmatched nodes in G , we have the following observation.

Observation 28. *Because of the maximality of M , there is no alternating path from U'_S to U'_T .*

Let X'_S and X'_T be the following sets of nodes.

$$\begin{aligned} X'_S &= B'_S \cup U'_S \\ X'_T &= B'_T \cup U'_T \end{aligned}$$

Claim 29. *There is a blocking structure in G_M if and only if there is an alternating path in G' connecting X'_S and X'_T .*

Proof of Claim. Let R be a blocking structure in G_M . Let us suppose that R is a blocking path and

$$R = \langle v_1, v_2, v_3 \dots v_k \rangle.$$

Without loss of generality, we can assume that $v_1 v_2$ is a blocking edge. Then, by the construction, we have the followings.

$$\begin{aligned} v_1^b v_2' &\in E' \text{ and } v_1^b \in X'_S \cup X'_T \\ v_{2i}' v_{2i+1}' &\in M' \text{ and } v_{2i+1}' v_{2i+2}' \in E' \text{ for any } i \in \{1, 2 \dots \frac{k}{2} - 1\} \end{aligned}$$

Finally, if $v_k \in U$, then $v_k' \in X_S \cup X_T$ holds. Otherwise, $v_{k-1} v_k$ is a blocking edge, thus $v_{k-1}' v_k^b \in E'$ and $v_k^b \in X'_S \cup X'_T$. Therefore, $\langle v_1^b, v_2' \dots v_k' / v_k^b \rangle$ is an alternating path in G' . When R is a blocking cycle $\langle v_1, v_2, v_3 \dots v_k, v_1 \rangle$, it can be proven similarly that there is an alternating path $\langle v_k^b, v_1', v_2' \dots v_k', v_1^b \rangle$ in G' connecting X'_S and X'_T .

Similarly, a blocking structure in G can be constructed from an alternating path P . By Observation 28, at least one endpoint of P is part of $B'_S \cup B'_T$. Let $P = \langle v_1^b, v_2', v_3' \dots v_k', (v_{k+1}^b) \rangle$, where the last node v_k' or v_{k+1}^b is part of $X'_S \cup X'_T$.

Let $v_1, v_2, \dots, v_k, v_{k+1}$ be the equivalent nodes in G . If either v_1 or v_{k+1} is equal to an internal node v_i , then $\langle v_1, v_2, \dots, v_{i-1} \rangle$ or $\langle v_{i+1}, v_{i+2}, \dots, v_{k+1} \rangle$ is a blocking cycle. Otherwise, $\langle v_1, v_2, v_3 \dots v_k, (v_{k+1}) \rangle$ is a blocking path. ■

Note that the cost of a set $K' \subseteq S' \cup T'$ is defined as the sum of the cost of the elements within the subset, i.e., $c'(K')$ is equal to $\sum_{v \in K'} c'(v)$. For any $K' \subseteq S' \cup T'$, let $G'_{K'}$ be the graph that is obtained from G' by deleting all non-matching edges linked to a node $v \in K'$. A set of nodes $K' \subseteq S' \cup T'$ is called a **linked cut** if there is no alternating path from X'_S to X'_T in $G'_{K'}$. The transformed problem is to find the minimum cost linked cut in G' . In the following, we will show that the minimum cost eliminative set in G can be determined by using a minimum cost linked cut for the transformed problem. Namely, we prove the following theorem.

Theorem 30. *Let $(G, M, <, c)$ be an instance of Problem 27. Let K' be a minimum cost linked cut for the transformed graph G' described above. Then*

$$K = \{v \mid v' \in K' \text{ or } v^b \in K'\} \tag{17}$$

is a minimum cost eliminative set of nodes in G .

Proof. First, let us show the following.

Claim 31. *Let K' be a linked cut in G' . Then the set in (17) is an eliminative set of nodes in G .*

Proof of Claim. Let us suppose, to the contrary, that K is not eliminative. Then, by Theorem 7, there exists a blocking structure R in G_M^K , and let us assume that it is a blocking path. (The case when R is a blocking cycle can be finished similarly.) If the path contains only one edge, then it is a blocking edge uv connecting to an unmatched node $u \in U$. Then there is a non-matching edge between v^b and u' , which is an alternating path connecting X'_S and X'_T . Thus, let

$$R = \langle v_1, v_2, \dots, v_k, (v_{k+1}) \rangle,$$

where either v_k is an unmatched node or $v_k v_{k+1}$ is a blocking edge. Since $v_1 v_2$ is a blocking edge in G_M^K , we get $v_1, v_2 \notin K$. Thus, we have

$$v_1^b, v_2' \notin K' \text{ and } v_1^b v_2' \in E'(G'_{K'}).$$

By the construction, we have

$$v'_{2i} v'_{2i+1} \in M' \text{ for any } i \in \{1, 2, \dots, \left\lfloor \frac{k}{2} \right\rfloor - 1\}.$$

Since $v_{2i+1} v_{2i+2}$ is a non-matching edge in G_M^K , we get

$$v'_{2i+1} v'_{2i+2} \in E'(G'_{K'}) \text{ for any } i \in \{1, 2, \dots, \left\lfloor \frac{k}{2} \right\rfloor - 1\}.$$

Finally, if $v_k \in U$, then $v'_{k-1} v'_k$ is a non-matching edge in $G'_{K'}$, as it is linked to none of its endpoints. Otherwise, $v_k v_{k+1}$ is a blocking edge, and it can be obtained that $v'_k v'_{k+1} \in E'(G'_{K'})$, similarly to edge $v_1^b v_2'$. Therefore, $\langle v_1^b, v'_2, v'_3, \dots, v'_k, (v'_{k+1}) \rangle$ is an alternating path in $G'_{K'}$ connecting X'_S and X'_T , which contradicts the condition of K' being a linked cut. ■

As a consequence of Claim 31, every linked cut in G' corresponds to an eliminative set in G of at most the same cost. To finish the proof of Theorem 30, we will demonstrate that, for every eliminative set of nodes in G , a linked cut of equal cost can be constructed in G' . Namely, we prove the following statement.

For any $K_1 \subseteq V, K_2 \subseteq B$, let us define the following set of nodes in G' .

$$K'(K_1, K_2) = \{v' \mid v \in K_1\} \cup \{v^b \mid v \in K_2\} \quad (18)$$

Claim 32. *Let $K_1 \subseteq V, K_2 \subseteq B$, such that $K_0 = K'(K_1, K_2)$ is a linked cut. If $v \in K_1 \cap K_2$, then either $K'(K_1 \setminus \{v\}, K_2)$ or $K'(K_1, K_2 \setminus \{v\})$ is also a linked cut in G' .*

Proof of Claim. Suppose to the contrary that neither of $K_0 \setminus \{v'\}$ nor $K_0 \setminus \{v^b\}$ is a linked cut.

On one hand, as $K_0 \setminus \{v'\}$ is not a linked cut, there is an alternating path P' from X'_S to X'_T in $G'_{K_0 \setminus \{v'\}}$. Since K_0 is a linked cut, P' is not part of G'_{K_0} . Thus, there is an edge $v'_{-1}v' \in P'$ linked to v' and it is not linked to any other node in $K_0 \setminus \{v'\}$. Let $P'_{rv'}$ denote the part of P' that starts with v' and a matching edge. This is an alternating path starting at v' and ending at $r \in X'_S \cup X'_T$. Since any edge in $P' - v'_{-1}v'$ is part of G'_{K_0} , we have $P'_{rv'} \subseteq G'_{K_0}$.

On the other hand, there is an alternating path P' from X'_S to X'_T in $G'_{K_0 \setminus \{v^b\}}$ that is not part of G'_{K_0} . Therefore, it starts at v^b and let v^bw' be the first edge in P' . Thus, we have

$$w' \notin K_0 \text{ and } vw \text{ is a blocking edge in } G. \quad (19)$$

Let $P_{w'z}$ denote the path that we obtain from P' by deleting v^b . This is an alternating path from w' that starts with a matched edge and ends in $z \in X'_S \cup X'_T$. Since any edge in $P - v^bw'$ is part of G'_{K_0} , we get that $P_{w'z} \subseteq G'_{K_0}$.

Finally, by (19), there is a non-matching edge $v'w'$ part of G'_{K_0} . In conclusion, every edge in $P_{rv'} + v'w' + P_{w'z}$ is part of G'_{K_0} . Therefore, the walk obtained by combining $P_{rv'}, v'w', P_{w'z}$ is an alternating walk connecting X'_S and X'_T . Since G' is bipartite, it contains an alternating path from X'_S to X'_T . This is a contradiction, as K_0 was a good selection. ■

Claim 33. *Let K be an eliminative set in G . The set K can be partitioned into two sets $K_1 \cup K_2 = K$, $K_1 \cap K_2 = \emptyset$, $K_2 \subseteq B$, such that*

$$K'(K_1, K_2)$$

is a linked cut in G' .

Proof of Claim. First, let us define the following set of nodes $K'' = K'(K, K \cap B)$. First, we will show that K'' is a linked cut in G' . Suppose to the contrary that there exists an alternating path P' from X'_S to X'_T in $G'_{K''}$. According to Observation 28, at least one endpoint is part of $B'_S \cup B'_T$. Let us suppose that

$$P = \langle v_1^b, v'_2, v'_3 \dots v'_k \rangle,$$

where $v_1 \in B'_S$ and $v_k \in U'_T$. (When P' is connecting B'_S and B'_T , the proof can be finished similarly.) Since $v_1^bv'_2$ is a non-matching edge in $G'_{K''}$ linked to both endpoints, edge v_1v_2 is originally a blocking edge in G and

$$v_1^b, v'_2 \notin K''.$$

Thus, $v_1, v_2 \notin K$ and v_1v_2 remains blocking in G_M^K . Similarly, we get

$$v_{2i}v_{2i+1} \in M \text{ and } v_{2i+1}v_{2i+2} \in E(G_M^K) \setminus M$$

for any $i \in \{1, 2 \dots \frac{k}{2} - 1\}$. If there is an index $i \in \{2, 3 \dots k - 1\}$, such that $v'_i = v'_1$, then $\langle v_1, v_2 \dots v_{i-1}, v_i = v_1 \rangle$ is a blocking cycle. (Note that $v_i \neq v_j$, if $i \neq j$ and $i, j \notin \{1, k\}$, since $v'_i, v'_j \in (S' \cup T') \setminus (X'_S \cup X'_T)$ and there is a bijection between $(S' \cup T') \setminus (X'_S \cup X'_T)$ and $S \cup T \setminus U$.) Finally, as $v_k \in U$, the path of $\langle v_1, v_2 \dots v_k \rangle$ is a blocking path in G_M^K . This contradicts to the condition that K is an eliminative set. In summary, we got that

$$K'(K, K \cap B) = K''$$

is a linked cut in G' . By Claim 32, we can divide K into two sets by removing either v' or v^b from K'' one at a time for any blocking node $v \in K$. This finishes the proof of the claim. $\blacksquare$

Due to Claims 31 and 33, the cost of a minimal eliminative set in G is equal to the cost of a minimal linked cut in G' , and these solutions are equivalent. This finishes the proof of Theorem 30. $\square$

To sum up everything, finding a minimum cost linked cut in the transformed problem is sufficient to solve Problem 27. The following section will discuss an algorithm for this purpose.

4.2 Algorithm for computing a minimum linked cut in G'

In this section, an auxiliary network graph G^* with terminal nodes s, t will be defined, and we will show that the minimum s - t cut of this graph is equivalent to a minimum linked cut in G' . The idea of the following model is to create a graph in which deleting an arc corresponds to a node being part of a linked cut.

Let us start by defining this auxiliary graph for the transformed problem (G', M', c') . Every node v' or v^b from $(S' \cup T') \setminus (U'_S \cup U'_T)$ is duplicated as v or v_b . Two extra nodes, n_S and n_T , are created to represent U'_S and U'_T , respectively. A new node m_{uv} is created for every matching edge $u'v' \in M'$. We call these nodes the matching nodes. Finally, let us define two terminal nodes s, t . In summary, the nodes of G^* are the following.

$$V(G^*) = \{v \mid v' \in V(M')\} \cup \{v_b \mid v^b \in B'_S \cup B'_T\} \cup \{m_{uv} \mid u'v' \in M'\} \cup \{n_S, n_T, s, t\}$$

For the sake of simplicity, $\overline{X'_S}$ and $\overline{X'_T}$ denote the sets $S' \setminus X'_S$ and $T' \setminus X'_T$, respectively. Then the following arcs are given to G^* . (See Fig. 9.)

- For any $u'v' \in M'$, where $v' \in S'$ and $u' \in T'$, arcs $v \rightarrow m_{uv}$ and $m_{uv} \rightarrow u$ with capacities $c^*(vm_{uv}) = c'(v)$ and $c^*(m_{uv}u) = c'(u)$, respectively.
- For any non-matching edge $u'v'$:
 - Arc $m_{uM(v)} \rightarrow v$ with capacity ∞ , if $v' \in \overline{X'_S}, u' \in \overline{X'_T}$ and $u'v'$ is linked to v' .
 - Arc $u \rightarrow m_{M(v)v}$ with capacity ∞ , if $v' \in \overline{X'_S}, u' \in \overline{X'_T}$ and $u'v'$ is linked to u' .
 - Arc $n_T \rightarrow m_{M(v)v}$ with capacity ∞ , if $u' \in U'_T$.

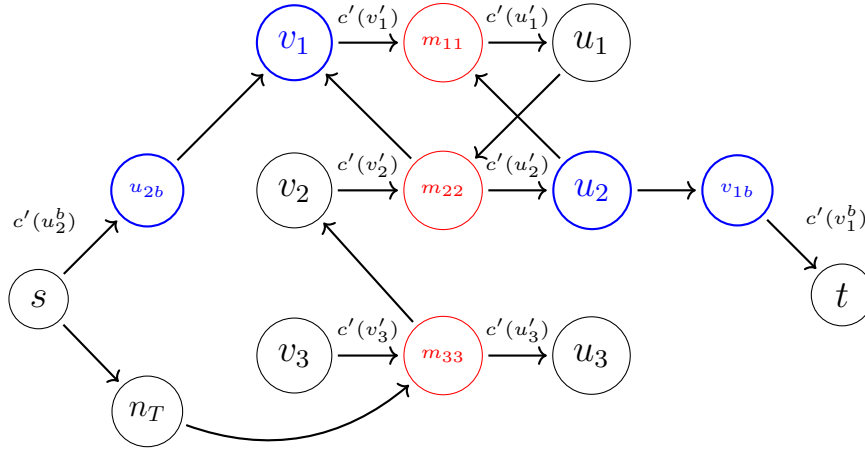


Figure 9: The corresponding auxiliary graph for the transformed problem shown in Fig. 8. The blocking nodes are blue, while the matching nodes are red. The unlabelled arcs have infinite capacities.

- Arc $m_{uM(u)} \rightarrow n_S$ with capacity ∞ , if $v' \in U'_S$.
- For any non-matching edge $u^b v'$:
 - Arc $u_b \rightarrow v$ with capacity ∞ , if $v' \in \overline{X'_S}$ and $u^b \in B'_T$.
 - Arc $v \rightarrow u_b$ with capacity ∞ , if $v' \in \overline{X'_T}$ and $u^b \in B'_S$.
 - Arc $u_b \rightarrow n_S$ with capacity ∞ , if $v' \in U'_T$ and $u^b \in B'_S$.
 - Arc $n_T \rightarrow u_b$ with capacity ∞ , if $v' \in U'_S$ and $u^b \in B'_T$.
- For any blocking node $u^b \in B'_T$ and $v^b \in B'_S$, arcs $s \rightarrow u_b$ and $v_b \rightarrow t$ with capacities $c^*(su_b) = c'(u^b)$ and $c^*(v_b t) = c'(v^b)$, respectively.
- Arcs $s \rightarrow n_T$ and $n_S \rightarrow t$ with capacity ∞ .

We will prove that computing the minimum cut in G^* defines a minimum cost linked cut in G' . The aim is to show that deleting arc $v \rightarrow m_{uv}$ or $m_{uv} \rightarrow u$ is equivalent to adding v' or u' to K' , while removing the arc $s \rightarrow v_b$ or $v_b \rightarrow t$ is equal to including v^b or u^b . For this, a corresponding set is defined for any finite cut in G^* . Note that $\{su_b \mid u^b \in B'_T\} \cup \{v_b t \mid v^b \in B'_S\}$ is a finite cut in G^* as there is no alternating path between U'_S and U'_T in G' . Let F be an arbitrary finite cut in G^* , and $K'(F) \subset V'$ is defined the following way.

- $v' \in K'(F)$, if $v' \in S'$ and $v \rightarrow m_{uv} \in F$
- $u' \in K'(F)$, if $u' \in T'$ and $m_{uv} \rightarrow u \in F$
- $v^b \in K'(F)$, if $v^b \in X'_S$ is a blocking node and $v_b \rightarrow t \in F$

- $u^b \in K'(F)$, if $u^b \in X'_T$ is a blocking node and $s \rightarrow u_b \in F$

Similarly, we can define a cut $F(K')$ corresponding to a set of nodes K in G' .

Note that any minimum s - t cut in G^* only contains arcs with finite capacities. Thus, if F is a minimum s - t cut, then the capacity of this cut is equal to $\sum_{w \in K'(F)} c'(w)$.

Lemma 34. *Let $F \in E(G^*)$ be a minimum s - t cut in G^* and K' a minimum cost linked cut in G' . Then*

$$\sum_{e \in F} c^*(e) = \sum_{w \in K'} c'(w).$$

Proof. First, we will show that $\sum_{e \in F} c^*(e) \geq \sum_{w \in K'} c'(w)$, and then the other direction will be proven.

$$\sum_{e \in F} c^*(e) \geq \sum_{w \in K'} c'(w) :$$

Let $K'(F)$ be defined as above. Arguing indirectly, assume there is an alternating path P' from X'_S to X'_T in $G'_{K'(F)}$. First, let us consider the case when P' has only one edge, meaning there is an edge between $v^b \in B'_S \cup B'_T$ and $u' \in U'_S \cup U'_T$. Since $v^b u'$ is part of $G'_{K'(F)}$, we have $v^b \notin K'(F)$. Therefore, $b_v \rightarrow t \notin F$ and the path of $s \rightarrow n_T \rightarrow b_v \rightarrow t$ is part of $G^* - F$. This is a contradiction.

Now, let us discuss the case when P' has more than one edge. First, let us suppose that it is connecting $B'_S \cup B'_T$ and $U'_S \cup U'_T$. Without loss of generality, we assume that its endpoints are $u_1^b \in B'_T$ and $v'_k \in U'_S$ and

$$P' = u_1^b, v'_1, u'_2, v'_2 \dots u'_k, v'_k.$$

As $u_1^b v'_1 \in G'_{K'(F)}$, we have that u_1^b and v'_1 are not a part of $K'(F)$. This is equivalent to $s \rightarrow u_{1b} \notin F$ and $v_1 \rightarrow m_{u_2 v_1} \notin F$. This means that

$$s \rightarrow u_{1b} \rightarrow v_1 \rightarrow m_{u_2 v_1} \subseteq G^* - F.$$

Since $u'_2 v'_2 \in G'_{K'(F)}$, it is linked to an endpoint that is not a part of $K'(F)$. If it is still linked to u'_2 , then $u'_2 \notin K'(F)$ and $m_{u_2 v_1} \rightarrow u_2 \notin F$. Thus, $m_{u_2 v_1} \rightarrow u_2 \rightarrow m_{u_3 v_2}$ is a path in $G^* - F$. Otherwise, the edge $u'_2 v'_2$ is linked to v'_2 and $m_{u_2 v_1} \rightarrow v_2 \rightarrow m_{u_3 v_2}$ is a path from $m_{u_2 v_1}$ to $m_{u_3 v_2}$ in $G^* - F$.

By repeating this argument, we obtain that

$$m_{u_i v_{i-1}} \rightarrow u_i \rightarrow m_{u_{i+1} v_i} \subseteq G^* - F \quad \text{or} \quad m_{u_i v_{i-1}} \rightarrow v_i \rightarrow m_{u_{i+1} v_i} \subseteq G^* - F$$

holds for any $i \in \{2, 3 \dots k\}$. Finally, as $u'_k v'_k$ is a non-matching edge in G' and $v'_k \in U'_S$, there is an arc $m_{u_k v_{k-1}} \rightarrow n_S$ in the auxiliary graph G^* . Thus,

$$m_{u_k v_{k-1}} \rightarrow n_S \rightarrow t$$

is clearly part of $G^* - F$, as its arcs have infinite capacities. Therefore, there is an st -path in $G^* - F$, which is a contradiction.

When the endpoints of P' are blocking nodes $v_1^b, v_k^b \in B'_S \cup B'_T$, the proof can be finished similarly.

Thus, $K'(F)$ is a linked cut and

$$\sum_{e \in F} c^*(e) = \sum_{w \in K'(F)} c'(w) \geq \sum_{w \in K'} c'(w)$$

This finishes the first part of the proof.

$\sum_{e \in F} c^*(e) \leq \sum_{w \in K'} c'(w)$: Let $F(K')$ be a set of arcs defined above. Let us suppose to the contrary that there is an st -path P in $G^* - F(K')$. In cases where there is no matching node P , the path will take the form of $s \rightarrow n_T \rightarrow b_v \rightarrow t$ or $s \rightarrow u_b \rightarrow n_S \rightarrow t$. We will discuss the case when the remaining path is $s \rightarrow n_T \rightarrow b_v \rightarrow t$, and the other case can be managed similarly. Since $b_v \rightarrow t \notin F(K')$, then $v^b \notin K'$ holds. As $n_T \rightarrow b_v \in E(G^*)$, there is a non-matching edge $u'v^b$ linked to v^b in G'_K , where $u' \in U'_T$ and $v^b \in B'_S$. Thus, this edge is an alternating path from X'_S to X'_T , which contradicts the requirement for K' to be a good set.

When there are matching nodes in P , let us denote the matching nodes of this path by $m_{u_2v_1}, m_{u_3v_2}, \dots, m_{u_kv_{k-1}}$ in the same order as they appear in the path. The first part of P is either $s \rightarrow u_b \rightarrow v_1 \rightarrow m_{u_2v_1}$ or $s \rightarrow n_T \rightarrow m_{u_2v_1}$. In the first case, there is a non-matching edge $u^bv'_1$ in G' by the construction. Since $s \rightarrow u_b, v_1 \rightarrow m_{u_1v_1} \notin F(K')$, then $u^b, v'_1 \notin K'$ and $u^bv_1 \in E'(G'_{K'})$. In the second case, there is a non-matching edge $u'v'_1 \in E'(G')$ by the construction. Since $u' \in U'_T$, the edge $u'v'_1$ is linked to none of its endpoints and it is part of $G'_{K'}$. Therefore, we get that

$$u^bv_1 \in E'(G'_{K'}) \quad \text{or} \quad u'v_1 \in E'(G'_{K'}),$$

where $u^b \in B'_T$ or $u' \in U'_T$. It is straightforward that

$$v'_i u'_{i+1} \in M'$$

holds for any $i \in \{1, 2, \dots, k\}$. Then, the directed path from $m_{u_i v_{i-1}}$ to $m_{u_{i+1} v_i}$ represents a non-matching $u'_i v'_i$ in G' . By the construction, this part of the path is either $m_{u_2v_1} \rightarrow u_2 \rightarrow m_{u_3v_2}$ or $m_{u_2v_1} \rightarrow v_2 \rightarrow m_{u_3v_2}$. If it contains u_2 , then $m_{u_2v_1} \rightarrow u_2 \notin F$ and $u'_2 \notin K'$. Otherwise, $v_2 \rightarrow m_{u_3v_2} \notin F$ and $v'_2 \notin K'$. In both cases, we get that $u'_2 v'_2$ is part of the $G'_{K'}$. Similarly, it can be obtained that

$$u'_i v'_i \in E'(G'_{K'})$$

holds for any $i \in \{1, 2, \dots, k-1\}$. Finally, it can be established that there is a node $v' \in U'_S$ or $v^b \in B'_S$, such that

$$u'_k v' \in E'(G'_{K'}) \quad \text{or} \quad u'_k v^b \in E'(G'_{K'}),$$

with a proof that is similar to u' or u^b . Hence, $\langle u'/u^b, v'_1, u'_2, v'_2 \dots v'_{k-1}, u'_k, v'/v^b \rangle$ is an alternating walk in $G'_{K'}$ between X'_S and X'_T . It contains an alternating path between X'_S and X'_T , which is a contradiction. Therefore, there is no st -path in $G^* - F(K')$.

Thus, $F(K')$ is a cut in G^* and

$$\sum_{e \in F} c^*(e) \leq \sum_{e \in F(K')} c^*(e) = \sum_{w \in K'} c'(w)$$

This finishes the proof of Lemma 34. $\square$

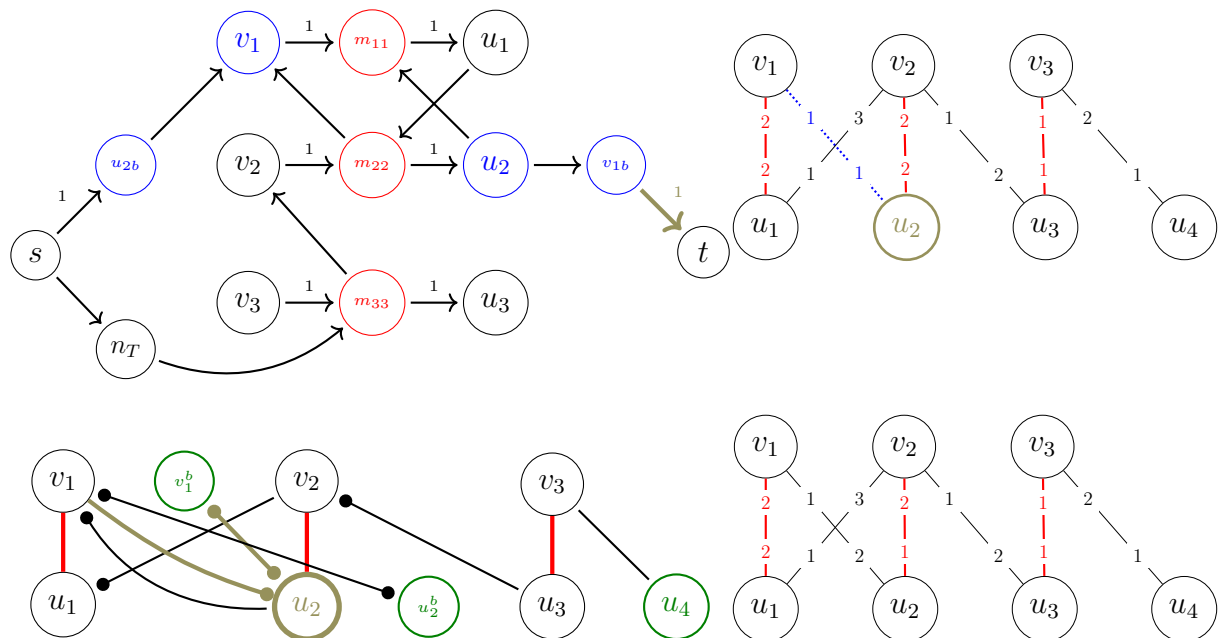


Figure 10: On the left, the auxiliary graph G^* with cost $c \equiv 1$ and the transformed problem G' . The yellow arc shows a minimum s - t cut in G^* , which corresponds to the yellow node in G' . After deleting the linked (yellow) edges from G' , there will be no alternating path between X_S and X_T (green nodes).

On the right, the original problem with the original preference lists (above) and with the modified ones (below) according to the min-cut.

In summary, the minimum cut in G^* corresponds to a minimum-cost linked cut in G' . According to Theorem 30, this linked cut is equivalent to a minimum-cost eliminative set of nodes in $G = (V, E)$. Therefore, Problem 27 with a maximum-size matching can be solved using Algorithm 1. The running time of this algorithm is determined by step 3, which involves computing a minimum cut in the directed graph G^* . The complexity of this step depends on the size of G^* . The auxiliary graph $G^* = (V^*, A^*)$ contains $O(|V|)$ nodes and $O(|V| + |E|)$ arcs. Consequently, by employing Orlin's $O(|V^*||A^*|)$ max-flow algorithm [3], the running time of Algorithm 1 is $O(|V|^2 + |V||E|)$, which is polynomial in the size of the original problem instance.

In problem 27, we assume that the cost function is positive. Algorithm 1 can be used with an arbitrary cost function $c : S \cup T \rightarrow \mathbb{N} \cup \{\infty\}$ as follows. First, the preference lists are altered at every node v such that $c^*(v) \leq 0$ according to Observation 5. The

Algorithm 1 Minimum cost eliminative set in a bipartite graph with maximum size matching

- 1: Input: $(G, M, <, c)$ an instance of Problem 27
 - 2: Create the auxiliary graph G^* as described above.
 - 3: Let F be a minimum cut of G^* .
 - 4: RETURN $K = \{v \mid m_{uv}v \in F \text{ or } v_b t \in F\} \cup \{u \mid um_{uv} \in F \text{ or } su_b \in F\}$
-

obtained problem is an instance of Problem 27, thus, Algorithm 1 solves it. Note that the algorithm works in the same way if a node has ∞ cost.

Another interesting special case is when a node set $V' \subseteq V$ is given, where it is possible to change the preferences. In other words, there are some nodes $v \in V \setminus V'$, where we cannot alter the preference list. This special version can be solved by choosing $c^*(v) = \infty$ for any $v \in V \setminus V'$ and applying Algorithm 1. (Note that if the minimum cut is infinite, the original problem is not feasible.)

4.3 Minmax theorem

In the previous section, it was established that the minimum eliminative set of nodes is equivalent to a minimum cut in a specific auxiliary graph denoted as G^* . An s - t path in G^* can be transformed into a blocking structure similarly to the method demonstrated in the proofs of Lemma 34 and Claim 29. Thus, an s - t flow in G^* can be converted into some blocking structure. By directly applying the well-known max-flow min-cut theorem to this graph, we get that the size of the minimum eliminative set is equal to the number of the obtained blocking structures. In the following, we present a better-formulated minmax theorem for Problem 27 with cost function $c \equiv 1$. For a general integer cost function, a similar result can be proven in the same manner.

Let $(G, M, <, 1)$ be an instance of Problem 27, where M is a maximum size matching. Let us consider a blocking structure $R = \langle v_1, v_2, \dots, v_k \rangle$ in G_M , where $v_i \neq v_j$ holds for any $i \neq j$, $(i, j) \neq (1, k)$. Note that if R is a blocking cycle, then $v_1 = v_k$ holds. We will say that a structure R covers the following nodes.

$$C(R) = \{v_i \in V(M) \cap V(R) \mid v_{i-1}v_i \text{ or } v_i v_{i+1} \text{ is linked to } v_i\} \quad (20)$$

If there is no internal blocking edge, the blocking structure R is eliminated by a matched node that has a linked edge from R . Therefore, to eliminate a blocking structure without internal blocking edges, at least one of its covered nodes must be part of the eliminative set. When internal blocking edges are present, an eliminative set has to include more than one node from R to eliminate it. Let us denote the number of blocking edges in R by $B(R)$. Let $w(R)$ be the minimum number of nodes that eliminate a blocking structure R . It is straightforward to verify the following.

$$w(R) = \begin{cases} B(R), & \text{if } R \text{ is a cycle} \\ B(R), & \text{if } R \text{ is a path and one endpoint is unmatched} \\ B(R) - 1, & \text{if } R \text{ is a path with no unmatched node} \end{cases}$$

In conclusion, let $\mathcal{R}$ be a set of blocking structures in G_M , such that $C(R) \cap C(R') = \emptyset$ for any $R, R' \in \mathcal{R}$. Since they cover different nodes, only distinct nodes can eliminate them. Therefore, at least $\sum_{R \in \mathcal{R}} w(R)$ nodes must be part of an eliminative set. We will demonstrate that this is strict by proving the following theorem.

Theorem 35. *Let $(G, M, <, \mathbb{1})$ be an instance of Problem 27. Then, the size of the minimal eliminative set is equal to the following maximum.*

$$\max \left\{ \sum_{R \in \mathcal{R}} w(R) \mid \mathcal{R} \text{ is a set of blocking structures, } C(R) \cap C(R') = \emptyset \quad \forall R, R' \in \mathcal{R} \right\} \quad (21)$$

Proof. In previous observations, we established that the expression 21 is a lower bound for the minimum size eliminative set. To prove the theorem, it is enough to find an eliminative set and disjoint blocking structures for which the equality holds. For this purpose, the min-cut model of Problem 27 is used. First, we will show that an integer flow in the auxiliary graph is equivalent to a set of disjoint blocking structures in G_M .

Claim 36. *Let x be an integer s - t flow of value k . Then there is a set of blocking structures $\mathcal{R}$, such that $C(R) \cap C(R') = \emptyset$ for any $R, R' \in \mathcal{R}$ and $\sum_{R \in \mathcal{R}} w(R) \geq k$.*

Proof of Claim. Since x is an integer s - t flow, it consists of k st -paths along with some directed cycles. Let us denote these paths by $P_1, P_2, \dots, P_k$. For any path P_i , an alternating walk between X'_S and X'_T can be constructed in the transformed problem similarly as in the proof of Lemma 34. Then, the obtained walks are equivalent to alternating walks by replacing every node in the walk with the original one.

Therefore, $P_1, P_2, \dots, P_k$ are equivalent to k alternating walks $P'_1, P'_2, \dots, P'_k$ that start or end with a blocking edge. The opposite end of each walk is either a blocking edge or an unmatched node. We refer to the union of the edges in them as $\mathcal{E}$. If P'_i contains an edge uv that is linked to u and $u \in T$, then P'_i contains su_b or $m_{uv}u$. Similarly, if P'_i contains an edge uv that is linked to v and $v \in S$, then P'_i contains $v_b t$ or vm_{uv} . Since the capacities of these edges are 1, every node is linked to at most two edges in $\mathcal{E}$.

Let us consider a node u that is linked to two edges from $\mathcal{E}$ and denote these edges by uv and uv' . One of these edges, called uv , is the starting blocking edge of an alternating walk denoted as P'_i . Consequently, u is an internal node within a different alternating walk P'_j .

$$\begin{aligned} P'_i &= \langle u, v, u_1^i, v_1^i, u_2^i, \dots, u_l^i, v_l^i \rangle \\ P'_j &= \langle u_1^j, v_1^j, u_2^j, \dots, u_r^j = u, v_r^j = v' \dots u_s^j, v_s^j \rangle \end{aligned}$$

The following two new alternating walks are established. The first one P''_i follows P'_j until reaching node u , at which point uv' is removed and P''_i continues along uv on the walk P'_i . Let us consider the case when there is an internal blocking edge in the remaining walk P'_j after the node u . Meaning there is an index $r + 1 \leq q' < s$, such that $u_{q'}^j v_{q'}^j$ is a blocking edge in P'_j . Then, P''_i starts in the first blocking edge and follows the alternating

walk P'_j until reaching the end. Otherwise, the other obtained alternating walk P''_j is an empty walk. Thus, we get the following blocking walks

$$P''_i = \langle u_1^j, v_1^j, u_2^j \dots u_r^j = u, v, u_1^i, v_1^i, u_2^i, \dots u_l^i, v_l^i \rangle$$

$$P''_j = \langle u_q^j, v_q^j, u_{q+1}^j \dots u_s^j, v_s^j \rangle \text{ or } P''_j = \emptyset,$$

where $u_q^j v_q^j$ is a blocking edge and $u_{q'}^j v_{q'}^j$ is not blocking for any $r + 1 \leq q' < q$.

After creating P''_i and P''_j from P'_i and P'_j , it can be easily confirmed that

$$w(P'_i) + w(P'_j) = w(P''_i) + w(P''_j)$$

holds. Repeating the merging procedure, we eventually get that there is no two edges within the union of $\{P''_1, P''_2 \dots P''_k\}$ that are linked to the same node. Furthermore, the value of the flow was k and $w(P_i) \geq 1$ for each $i \in \{1, 2 \dots k\}$. Therefore, we have the following.

$$\sum_{1 \leq i \leq k} w(P''_i) = \sum_{1 \leq i \leq k} w(P'_i) \geq k$$

$$C(P''_i) \cap C(P''_j) = \emptyset \quad \forall i, j \in \{1, 2, \dots k\}$$

Finally, the blocking structures are established in the following manner using the obtained alternating walk. If a walk P''_i contains a cycle with a blocking edge, then this cycle is deleted from the walk and serves as a blocking structure itself. The other cycles without blocking edges are also deleted. The remaining walk is simplified by deleting the matched nodes from the ends that are not connected to a blocking edge. (The resulting walk is a blocking path.) Let us denote these blocking structures by $R_1, R_2, \dots R_l$. By the formation of these blocking structures, we get the following.

$$\sum_{1 \leq i \leq k} w(R_i) = \sum_{1 \leq i \leq k} w(P''_i) \geq k$$

Therefore, $R_1, R_2, \dots R_l$ form a set of blocking structures that satisfies the conditions of the lemma. ■

Due to well-know max-flow min-cut theorem, the minimum cut in G^* is equal to the maximum integer flow. By Lemma 34, the minimum cut is equivalent to the minimum size eliminative set of nodes. According to Claim 36, the maximum flow provides a set of blocking structures that meet the conditions of (21), and the equation holds. This finishes the proof of Theorem 35. □

For a positive, integer cost function, the following theorem can be proven similarly. For a set of blocking structures $\mathcal{R}$, let $\tilde{C}_{\mathcal{R}}(v)$ denote the number of blocking structures R in $\mathcal{R}$, such that $v \in C(R)$ for any $v \in S \cup T$.

Theorem 37. *Let $(G, M, <, c)$ be an instance of Problem 27. Then, the cost of the minimal eliminative set is equal to the following maximum*

$$\max \left\{ \sum_{R \in \mathcal{R}} w(R) \mid R \text{ is a blocking structure, } \tilde{C}_{\mathcal{R}}(v) \leq c(v) \text{ for any } v \in S \cup T \right\} \quad (22)$$

5 Conclusion and open questions

We showed that the Inverse Popular Matching problem is NP-hard, even if the cost function $c \equiv 1$ is uniform and the input matching is perfect. In this scenario, the vertex cover problem is reduced to the problem. Furthermore, we established that the inverse popular matching problem remains NP-hard in a bipartite graph with a uniform cost function and maximum degree of 7. Our proof relies on the NP-hardness of the 1-in-3-SAT+ problem and the construction of an auxiliary graph. Finally, we presented a polynomial time algorithm to determine the minimum cost of an eliminative set in bipartite graphs if the desired matching is maximal. First, we transformed the problem into an equivalent version with linked edges. This transformation was fundamentally based on duplicating specific nodes and showing that a minimal linked cut included only one instance from each pair of replicated nodes. Subsequently, we created a flow graph and showed that the minimum cut of it is equivalent to the minimum cost linked cut in the transformed problem. Finally, using the above-mentioned flow model, we established a minmax characterization for an integer cost function.

We conclude with open questions that we hope will inspire further study and potentially lead to discoveries in this field. In our paper, we define a preference modification as an entire rearrangement. Is the problem polynomial if a predefined cost is associated with swapping two consecutive elements in a preference list? Our goal was to make a fixed matching popular. Can we solve the problem with alternative goal variants, such as modifying the preferences to ensure the existence of a popular matching that includes particular edges? The linked cut problem may be analyzed independently, without reference to instances of the Popular Matching problem. Consider an arbitrary bipartite graph with a fixed matching, where each non-matching edge is linked to one, both, or neither of its endpoints. A linked cut between two predetermined disjoint node sets is defined as in 4.1. Is it possible to solve the minimum cost linked cut problem in general? Furthermore, can the problem be addressed without a matching by identifying a linked cut K that eliminates all paths from X_S to X_T in the graph G_K ?

Acknowledgement

This research was supported by the National Research Excellence Programme of the Hungarian National Research, Development and Innovation Office under the STARTING.25 scheme ID: STARTING 153002.

The authors would like to acknowledge the valuable suggestions of the anonymous reviewers.

References

- [1] C.-C. Huang and T. Kavitha, “Popular matchings in the stable marriage problem,” *Information and Computation*, vol. 222, pp. 180–194, 2013. [doi:10.1016/j.ic.2012.10.012](https://doi.org/10.1016/j.ic.2012.10.012)

- [2] C. Moore and J. M. Robson, “Hard tiling problems with simple tiles,” *Discrete & Computational Geometry*, vol. 26, no. 4, pp. 573–590, 2001. [doi:10.1007/s00454-001-0047-6](https://doi.org/10.1007/s00454-001-0047-6)
- [3] J. B. Orlin, “Max flows in $O(nm)$ time, or better,” in *Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC 2013)*, pp. 765–774, 2013. [doi:10.1145/2488608.2488705](https://doi.org/10.1145/2488608.2488705)
- [4] T. J. Schaefer, “The complexity of satisfiability problems,” in *Proceedings of the 10th Annual ACM Symposium on Theory of Computing (STOC 1978)*, pp. 216–226, 1978. [doi:10.1145/800133.804350](https://doi.org/10.1145/800133.804350)
- [5] Á. Cseh, “Popular matchings,” in *Trends in Computational Social Choice*, U. Endriss, Ed. AI Access, 2017, ch. 6, pp. 105–122.
- [6] D. Gale and L. S. Shapley, “College admissions and the stability of marriage,” *The American Mathematical Monthly*, vol. 69, no. 1, pp. 9–15, 1962. [doi:10.1080/00029890.1962.11989827](https://doi.org/10.1080/00029890.1962.11989827)
- [7] P. Gärdenfors, “Match making: Assignments based on bilateral preferences,” *Behavioral Science*, vol. 20, no. 3, pp. 166–173, 1975. [doi:10.1002/bs.3830200304](https://doi.org/10.1002/bs.3830200304)
- [8] T. Kavitha, “Popular roommates in simply exponential time,” *Algorithmica*, vol. 87, pp. 292–320, 2025. [doi:10.1007/s00453-024-01287-4](https://doi.org/10.1007/s00453-024-01287-4)
- [9] D. F. Manlove, *Algorithmics of Matching Under Preferences*. World Scientific, 2013. [doi:10.1142/8591](https://doi.org/10.1142/8591)
- [10] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: W. H. Freeman, 1979.
- [11] K. Bérczi, L. M. Mendoza-Cadena, and K. Varga, “Inverse optimization problems with multiple weight functions,” *Discrete Applied Mathematics*, vol. 327, pp. 134–147, 2023. [doi:10.1016/j.dam.2022.12.004](https://doi.org/10.1016/j.dam.2022.12.004)
- [12] Y. Faenza, T. Kavitha, V. Powers, and X. Zhang, “Popular matchings and limits to tractability,” in *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2019)*, pp. 2790–2809, 2019. [doi:10.1137/1.9781611975482.173](https://doi.org/10.1137/1.9781611975482.173)
- [13] P. Biró, R. W. Irving, and D. F. Manlove, “Popular matchings in the marriage and roommates problems,” in *Algorithms and Complexity: Proceedings of CIAC 2010*, Lecture Notes in Computer Science, vol. 6078, pp. 97–108, Springer, 2010. [doi:10.1007/978-3-642-13073-1_10](https://doi.org/10.1007/978-3-642-13073-1_10)
- [14] T. Kavitha, “A size-popularity tradeoff in the stable marriage problem,” *SIAM Journal on Computing*, vol. 43, no. 1, pp. 52–71, 2014. [doi:10.1137/120902562](https://doi.org/10.1137/120902562)
- [15] S. Gupta, P. Misra, S. Saurabh, and M. Zehavi, “Popular matching in roommates setting is NP-hard,” *ACM Transactions on Computation Theory*, vol. 13, no. 2, Article 9, pp. 1–20, 2021. [doi:10.1145/3442354](https://doi.org/10.1145/3442354)

- [16] K. Bérczi, G. Csáji, and T. Király, “Manipulating the outcome of stable marriage and roommates problems,” *Games and Economic Behavior*, vol. 147, pp. 407–428, 2024. doi:[10.1016/j.geb.2024.08.010](https://doi.org/10.1016/j.geb.2024.08.010)
- [17] N. Boehmer, R. Bredereck, K. Heeger, and R. Niedermeier, “Bribery and control in stable marriage,” *Journal of Artificial Intelligence Research*, vol. 71, pp. 993–1048, 2021. doi:[10.1613/jair.1.12755](https://doi.org/10.1613/jair.1.12755)
- [18] T. Inoshita, R. W. Irving, K. Iwama, S. Miyazaki, and T. Nagase, “Improving man-optimal stable matchings by minimum change of preference lists,” *Algorithms*, vol. 6, no. 2, pp. 371–382, 2013. doi:[10.3390/a6020371](https://doi.org/10.3390/a6020371)
- [19] M. Demange and J. Monnot, “An introduction to inverse combinatorial problems,” in *Paradigms of Combinatorial Optimization: Problems and New Approaches*, V. T. Paschos, Ed. Wiley, 2013, ch. 17, pp. 547–586. doi:[10.1002/9781118600207.ch17](https://doi.org/10.1002/9781118600207.ch17)
- [20] T. Ayer, “Inverse optimization for assessing emerging technologies in breast cancer screening,” *Annals of Operations Research*, vol. 230, no. 1, pp. 57–85, 2015. doi:[10.1007/s10479-013-1520-3](https://doi.org/10.1007/s10479-013-1520-3)
- [21] T. C. Y. Chan, T. Craig, T. Lee, and M. B. Sharpe, “Generalized inverse multiobjective optimization with application to cancer therapy,” *Operations Research*, vol. 62, no. 3, pp. 680–695, 2014. doi:[10.1287/opre.2014.1267](https://doi.org/10.1287/opre.2014.1267)
- [22] J. R. Birge, A. Hortaçsu, and J. M. Pavlin, “Inverse optimization for the recovery of market structure from market outcomes: An application to the MISO electricity market,” *Operations Research*, vol. 65, no. 4, pp. 837–855, 2017. doi:[10.1287/opre.2017.1606](https://doi.org/10.1287/opre.2017.1606)
- [23] T. Crönert, L. Martin, S. Minner, and C. S. Tang, “Inverse optimization of integer programming games for parameter estimation arising from competitive retail location selection,” *European Journal of Operational Research*, vol. 312, no. 3, pp. 938–953, 2024. doi:[10.1016/j.ejor.2023.06.041](https://doi.org/10.1016/j.ejor.2023.06.041)
- [24] S. Allen, S. A. Gabriel, and J. P. Dickerson, “Using inverse optimization to learn cost functions in generalized Nash games,” *Computers & Operations Research*, vol. 142, Article 105721, 2022. doi:[10.1016/j.cor.2022.105721](https://doi.org/10.1016/j.cor.2022.105721)
- [25] L. Chen, Y. Chen, and A. Langevin, “An inverse optimization approach for a capacitated vehicle routing problem,” *European Journal of Operational Research*, vol. 295, no. 3, pp. 1087–1098, 2021. doi:[10.1016/j.ejor.2021.03.031](https://doi.org/10.1016/j.ejor.2021.03.031)
- [26] P. Xu, T.-L. Liu, Q. Tian, B. Si, W. Liu, and H.-J. Huang, “Estimation of schedule preference and crowding perception in urban rail corridor commuting: An inverse optimization method,” *Transportation Research Part B: Methodological*, vol. 189, Article 103023, 2024. doi:[10.1016/j.trb.2024.103023](https://doi.org/10.1016/j.trb.2024.103023)
- [27] J. Y. J. Chow and W. W. Recker, “Inverse optimization with endogenous arrival time constraints to calibrate the household activity pattern problem,” *Transportation Research Part B: Methodological*, vol. 46, no. 3, pp. 463–479, 2012. doi:[10.1016/j.trb.2011.11.005](https://doi.org/10.1016/j.trb.2011.11.005)

- [28] R. Fernández-Blanco, J. M. Morales, S. Pineda, and Á. Porras, “Inverse optimization with kernel regression: Application to the power forecasting and bidding of a fleet of electric vehicles,” *Computers & Operations Research*, vol. 134, Article 105405, 2021. [doi:10.1016/j.cor.2021.105405](https://doi.org/10.1016/j.cor.2021.105405)
- [29] A. E. Roth and E. Peranson, “The redesign of the matching market for American physicians: Some engineering aspects of economic design,” *American Economic Review*, vol. 89, no. 4, pp. 748–780, 1999. [doi:10.1257/aer.89.4.748](https://doi.org/10.1257/aer.89.4.748)
- [30] A. Abdulkadiroğlu and T. Sönmez, “School choice: A mechanism design approach,” *American Economic Review*, vol. 93, no. 3, pp. 729–747, 2003. [doi:10.1257/000282803322157061](https://doi.org/10.1257/000282803322157061)
- [31] A. E. Roth, T. Sönmez, and M. U. Ünver, “Efficient kidney exchange: Coincidence of wants in markets with compatibility-based preferences,” *American Economic Review*, vol. 97, no. 3, pp. 828–851, 2007. [doi:10.1257/aer.97.3.828](https://doi.org/10.1257/aer.97.3.828)