Popular Matchings and Limits to Tractability

Yuri Faenza* Telikepalli Kavitha† Vladlena Powers* Xingyu Zhang*

Abstract

We consider *popular matching* problems in both bipartite and non-bipartite graphs with strict preference lists. It is known that every stable matching is a min-size popular matching. A subclass of max-size popular matchings called *dominant matchings* has been well-studied in bipartite graphs: they always exist and there is a simple linear time algorithm to find one.

We show that it is NP-complete to decide if a bipartite graph admits a popular matching that is neither stable nor dominant. This gives rise to the anomaly that though it is easy to find min-size and max-size popular matchings in bipartite graphs, it is NP-complete to decide if there exists any popular matching whose size is sandwiched between the two extremes. We also show a number of related hardness results, such as (tight) 1/2-inapproximability of the maximum cost popular matching problem when costs are nonnegative. In non-bipartite graphs, we show a strong negative result: it is NP-hard to decide whether a popular matching exists or not, and the same result holds if we replace *popular* with *dominant*. On the positive side, we show the following results in any graph:

- we identify a subclass of dominant matchings called *strongly dominant* matchings and show a linear time algorithm to decide if a strongly dominant matching exists or not;
- we show an efficient algorithm to compute a popular matching of minimum cost in a graph with edge costs and bounded treewidth, or decide there is no popular matching.

1 Introduction

The marriage problem introduced by Gale and Shapley [8] is arguably the most relevant two-sided market model, and has been studied and applied in many areas of mathematics, computer science, and economics. The classical model assumes that the input is a complete bipartite graph, and that each node is endowed with a strict preference list over the set of nodes of the opposite color class. The goal is to find a matching that respects a concept of fairness called *stability*. An easy extension deals with incomplete lists, i.e., it assumes that the input graph is bipartite but need not be complete. In this setting, the problem enjoys elegant structural properties, that lead to fast algorithms for many optimization problems (a classical reference here is [12]).

In order to investigate more realistic scenarios, extensions of the model above have been studied. On one hand, one can change the structure of the *input*, admitting e.g. more complex preference patterns, or allowing the graph to be non-bipartite (see e.g. [23] for a collection of extensions). On the other hand, one can change the requirements of the *output*, i.e., ask for a matching that satisfies properties other than stability. For instance, relaxing the stability condition to *popularity* allows one to overcome one of the main drawbacks of stable matchings: its size. Indeed, the restriction that blocking pairs are forbidden may constrain the size of a stable matching to be as little as half the size of a maximum matching (note that a stable matching is maximal, so its size is at least half the size of a maximum matching). Popularity is a natural relaxation of stability: roughly speaking, a matching M is *popular* if the number of nodes that prefer M to any matching M' is at least the number of nodes that prefer M' to M. One can show that stable matchings are popular matchings of minimum size, and a maximum size popular matching can be twice as large as a stable matching. Hence, popularity allows for matchings of larger size while still guaranteeing a certain fairness condition.

Popular matchings (and variations thereof) have been extensively studied in the discrete optimization community, see e.g. [2, 6, 13, 14, 16, 17, 18], but there are still large gaps on what we know on the tractability of optimization problems over this family. Interestingly, all tractability results in popular matchings rely on connections with stable matchings. Kavitha [17] showed that a max-size popular matching can be found efficiently by a combination of the Gale-Shapley algorithm and *promotion* of "proposers" rejected by all neighbors. Cseh and Kavitha [6] showed that a pair of nodes is matched together in some popular matching if and only if it is matched in some stable or *dominant* matching.

*IEOR Department, Columbia University, New York City, USA, email: yf2414, vp2342, xz2464@columbia.edu

†Tata Institute of Fundamental Research, Mumbai, India, email: kavitha@tifr.res.in

Dominant matchings are a subclass of max-size popular matchings and the image (under a simple linear map) of stable matchings in a larger graph. Recently, Kavitha [19] showed that when there are costs on edges, finding a max-cost popular matching is NP-hard.

The notion of popular matchings can be immediately extended to non-bipartite graphs. Popular matchings need not always exist in a non-bipartite graph and it was not known if one can efficiently decide if a popular matching exists or not in a given non-bipartite graph.

Our contribution. In this paper, we show NP-hardness, inapproximability results, and polynomial-time algorithms for many problems on popular matchings, settling (mostly in the negative) most of the open questions in the area (see e.g. [5, 6, 14, 18, 23]), as well as introducing restrictions and concepts that make these problems tractable again. We solve a decade-old problem and show it is NP-complete to decide if a popular matching exists in a non-bipartite graph[1] (see Theorem 4.6). We also show that the problem stays NP-complete if we replace *popular* with *dominant* (see Theorem 4.5). In bipartite graphs, we show the following, quite surprising result:

> it is NP-complete to decide if a bipartite graph has a popular matching that is neither a *min-size* nor a *max-size* popular matching.

In particular, this is achieved by showing that it is NP-complete to decide if a bipartite graph has a popular matching that is neither stable nor dominant (see Theorem 3.1). Finding a min-size (similarly, max-size) popular matching is easy as stable matchings (resp., dominant matchings) always exist in bipartite graphs and there are linear time algorithms to compute these matchings. We are not aware of any other natural combinatorial optimization problem where finding elements of min-size (similarly, max-size) is easy but to decide whether there exists *any* element that is neither min-size nor max-size is NP-hard.

Still in bipartite graphs, we prove other complexity results, including the NP-completeness of deciding if there exists a popular matching with or without two given edges (see Theorem 4.1), and that, unless P=NP, the maximum cost popular matching problem with nonnegative costs cannot be approximated better than a factor $1/2$, and this is tight, since a $1/2$-approximation follows by known results (see Theorem 4.2).

All together, those negative results cast a dark shadow on the tractability of popular matchings. The second part of the paper is devoted to finding conditions under which (some of) those problems become tractable again. It is known that stable matchings are a tractable subclass of popular matchings in non-bipartite graphs [15]. The fact that stable matchings and dominant matchings are the only tractable subclasses of popular matchings in bipartite graphs prompts the following question: is there any non-trivial subclass of dominant matchings that is tractable in *all* graphs?

We show the answer to the above question is "yes" by identifying a subclass called *strongly dominant* matchings (see Definition 5): in bipartite graphs, these two classes coincide. We show a simple linear time algorithm for the problem of deciding if a strongly dominant matching exists in a given graph and if so, finding one. We also show that a popular matching of minimum cost (*with no restriction* on the signs of the cost function) in bipartite *and* non-bipartite graphs can be found efficiently if the treewidth of the input graph is bounded. This assumption can be of interest e.g. for balancing loads between and within server clusters, a problem that is solved using some concept of fairness (like stability [22]) and whose associated graphs often exhibit bounded treewidth [21].

Background and related results. Algorithmic questions for popular matchings were first studied in the domain of *one-sided* preference lists [1] in bipartite instances where it is only nodes on one side that have preferences over their neighbors. Popular matchings need not always exist here.

Popular matchings always exist in a bipartite instance with two-sided strict preference lists [10], and polynomial time algorithms to compute a max-size popular matching here were given in [13, 17]. A polynomial time algorithm to solve the max-cost popular matching problem in a complete bipartite graph was given in [6]. In non-bipartite graphs, it was shown in [19] that it is NP-hard to find a max-size popular matching (even when a stable matching exists) and in [14] that it is UGC-hard to compute a $\Theta(1)$-approximation of a max-cost popular matching.

It was very recently shown [20] that given a bipartite graph G along with a parameter $k \in (\mathsf{min}, \mathsf{max})$, where min is the size of a stable matching and max is the size of a dominant matching, it is NP-hard to decide whether G admits a popular matching of size k or not. Note that our NP-hardness result is a much stronger statement as we show that the problem of deciding whether G admits a popular matching of *any* intermediate size (rather than a particular size k) is NP-hard. In a subsequent work, Cseh and Kavitha [7] extended our proof of hardness of the popular roommates problem to show that the problem remains hard in complete graphs.

[1] Another, independently obtained, and different proof of this result also appears in the present volume [11].

Organization of the paper. After settling basic definitions, in Section 2.1 we present an LP framework of popular matchings in bipartite graphs proposed in [16] and further developed in [18, 19]. This framework allows to certify popularity (also stability, dominance) of matchings through the existence of certain vectors $\vec{\alpha}$, that correspond to dual variables of the linear relaxation of an integer programming formulation of popular matchings. This framework is crucial for the proof of our NP-hardness results. The main reduction is presented in Section 3: given an input formula ϕ to 1-in-3 SAT, we build a bipartite graph G such that any popular matching in G that is neither stable nor dominant has to match all nodes in G except one particular pair. We show that such a popular matching exists if and only if ϕ is 1-in-3 satisfiable. The other hardness results obtained here for bipartite and non-bipartite graphs can be obtained as consequences of the above reduction (see Section 4).

The algorithm for finding a strongly dominant matching is given in Section 5 and relies on Irving's classical algorithm for computing a stable matching in a non-bipartite graph [15] and on the characterization of strongly dominant matchings in terms of certain dual certificates $\vec{\alpha}$. Broadly speaking, this algorithm generalizes the dominant matching algorithm from [17] to all graphs. Note that the algorithm from [17] was exclusively for bipartite graphs: vertices of the two color classes played different roles there.

In Section 6, we prove that a popular matching of minimum cost can be found efficiently in graphs of bounded treewidth. The main tool here is a combinatorial characterization of popular matching in terms of certain forbidden structures in an appropriate labelled subgraph G_M [13].

2 Preliminaries

Throughout the paper, we will consider problems where our input is a graph G, together with a collection of rankings, one per node of G, with each node ranking its neighbors. *Each node* v *of* G has a preference list, and all preference lists are *strict*. We will denote an edge of G between nodes u and v as (u, v) or uv. If G is bipartite, we will denote it by $G(A \cup B, E)$, where A, B are the two color classes in which we partition the vertices of G. A matching M in G is *stable* if there is no *blocking edge* with respect to M, i.e. an edge whose both endpoints strictly prefer each other to their respective assignments in M. It follows from the classical work of Gale and Shapley [8] that a stable matching always exists when G is bipartite and such a matching can be computed in linear time.

The notion of popularity was introduced by Gärdenfors [10] in 1975. We say a node u *prefers* matching M to matching M' if either (i) u is matched in M and unmatched in M' or (ii) u is matched in both M, M' and u prefers $M(u)$ to $M'(u)$, where $M(u)$ is the partner of u in M. For any two matchings M and M', let $\phi(M, M')$ be the number of nodes that prefer M to M'.

Definition 1. *A matching* M *is* popular *if* $\phi(M, M') \geq \phi(M', M)$ *for every matching* M' *in* G*, i.e.,* $\Delta(M, M') \geq 0$ *where* $\Delta(M, M') = \phi(M, M') - \phi(M', M)$*.*

Thus, there is no matching M' that would defeat a popular matching M in an election between M and M', where each node casts a vote for the matching that it prefers. Since there is no matching where more nodes are *better-off* than in a popular matching, a popular matching can be regarded as a "globally stable matching". Equivalently, popular matchings are weak *Condorcet winners* [4] in the voting instance where nodes are voters and all feasible matchings are the candidates.

Though (weak) Condorcet winners need not exist in a general voting instance, popular matchings always exist in bipartite graphs, since every stable matching is popular [10]. Popular matchings have been extensively studied in bipartite graphs, in particular, a subclass of max-size popular matchings called *dominant matchings* is well-understood [6, 13, 17].

Definition 2. *A popular matching* M *is dominant in* G *if* M *is more popular than any larger matching in* G*, i.e.,* $\Delta(M, M') > 0$ *for any matching* M' *such that* $|M'| > |M|$*.*

Dominant matchings always exist in a bipartite graph and such a matching can be computed in linear time [17]. Every polynomial time algorithm currently known to find a popular matching in a bipartite graph finds either a stable matching [8] or a dominant matching [13, 17, 6].

In some problems, together with the graph G and the preference lists, we will also be given a cost function $c : E \to \mathbb{R}$. The *cost* of a matching M of G (with respect to c) is defined as $c(M) := \sum_{e \in M} c(e)$.

2.1 Labeled edges and dual certificates We present here an LP framework of popular matchings in bipartite graphs from [16, 18, 19]. Let M be any matching in G. We can associate *labels* to edges from $E \setminus M$ as follows:

- an edge (u, v) is $(-, -)$ if both u and v prefer their respective partners in M to each other;

- $(u,v) = (+,+)$ if u and v prefer each other to their partners in M;
- $(u,v) = (+,-)$ if u prefers v to its partner in M and v prefers its partner in M to u.

Blocking edges introduced earlier coincide with $(+,+)$ edges. Let $\tilde{G}$ be the graph G augmented with *self-loops*, i.e., every node is its own last choice. Corresponding to any matching M in G, there is a perfect matching $\tilde{M}$ in $\tilde{G}$ defined as follows: $\tilde{M} = M \cup \{(u,u) : u$ is left unmatched in $M\}$. We can therefore define an edge weight function wt_M in $\tilde{G}$ as follows.

$$\mathsf{wt}_M(u,v) = \begin{cases} 2 & \text{if } (u,v) \text{ is labeled } (+,+) \\ -2 & \text{if } (u,v) \text{ is labeled } (-,-) \\ 0 & \text{otherwise} \end{cases}$$

Thus $\mathsf{wt}_M(e) = 0$ for every $e \in M$. We need to define wt_M on self-loops as well: for any node u, let $\mathsf{wt}_M(u,u) = 0$ if u is unmatched in M, else let $\mathsf{wt}_M(u,u) = -1$. It is easy to see that for any matching N in G, $\Delta(N,M) = \mathsf{wt}_M(\tilde{N})$, where $\Delta(N,M) = \phi(N,M) - \phi(M,N)$ (see Definition 1). Thus M is popular if and only if every perfect matching in the graph $\tilde{G}$ has weight at most 0.

Theorem 2.1. ([16, 18]) *Let M be any matching in $G = (A \cup B, E)$. The matching M is popular if and only if there exists a* witness, *i.e. a vector $\vec{\alpha} \in \{0, \pm 1\}^n$ (where $n = |A \cup B|$) such that $\sum_{u \in A \cup B} \alpha_u = 0$ and*

$$\begin{aligned} \alpha_a + \alpha_b &\geq \mathsf{wt}_M(a,b) \quad \forall\, (a,b) \in E \\ \alpha_u &\geq \mathsf{wt}_M(u,u) \quad \forall\, u \in A \cup B. \end{aligned}$$

Call any $e \in E$ a *popular edge* if there is some popular matching in G that contains e. The popular subgraph F_G is a useful subgraph of G defined in [19].

Definition 3. *The* popular subgraph $F_G = (A \cup B, E_F)$ *is the subgraph of $G = (A \cup B, E)$ whose edge set E_F is the set of popular edges in E.*

Let $\mathcal{C}_1, \ldots, \mathcal{C}_h$ be the various components in F_G. Let M be a popular matching in G and let $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M. Call a node *unstable* if it is not contained in one (equivalently, any) stable matching.

Lemma 2.1. ([19]) *For any connected component $\mathcal{C}_i$ in F_G, either $\alpha_u = 0$ for all nodes $u \in \mathcal{C}_i$ or $\alpha_u \in \{\pm 1\}$ for all nodes $u \in \mathcal{C}_i$. Moreover, if $\mathcal{C}_i$ contains one or more unstable nodes, either all these unstable nodes are matched in M or none of them is matched in M.*

Next definition marks the state of each connected component $\mathcal{C}_i$ in F_G as "zero" or "unit" in $\vec{\alpha}$ — this classification will be useful in our hardness reduction.

Definition 4. *A connected component $\mathcal{C}_i$ in F_G is in* zero state *in $\vec{\alpha}$ if $\alpha_u = 0$ for all nodes $u \in \mathcal{C}_i$; it is in* unit state *in $\vec{\alpha}$ if $\alpha_u \in \{\pm 1\}$ for all nodes $u \in \mathcal{C}_i$.*

The vector $\vec{\alpha}$ as given in Theorem 2.1 is an optimal solution to the LP that is dual to the max-weight perfect matching LP in $\tilde{G}$ (with edge weight function wt_M). For any popular matching M, a vector $\vec{\alpha}$ as given in Theorem 2.1 will be called a *witness* to M.

A stable matching has the all-zeros vector $\vec{0}$ as a witness. It follows from [6] that a dominant matching M has a witness $\vec{\alpha}$ where $\alpha_u \in \{\pm 1\}$ for all nodes u matched in M and $\alpha_u = 0$ for all nodes u left unmatched in M. Call $s \in V$ a *stable node* if it is matched in some (equivalently, all) stable matching(s) [9]. Every popular matching has to match all stable nodes [13]. A node that is not stable is called an *unstable node*.

Let M be a popular matching in $G = (A \cup B, E)$ and let $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M. Lemma 2.2 given below follows from complementary slackness conditions. The first part of this lemma will be useful to us in identifying those edges in the graph G constructed in Section 3 that are *not* popular edges.

Lemma 2.2. ([19]) *For any popular edge (a,b), we have $\alpha_a + \alpha_b = \mathsf{wt}_M(a,b)$. For any unstable node u in G, if u is left unmatched in M, then $\alpha_u = 0$ else $\alpha_u = -1$.*

3 Finding a popular matching that is neither stable nor dominant

This section is devoted to proving the following result.

Theorem 3.1. *Given a bipartite instance $G = (A \cup B, E)$ with preference lists, the problem of deciding if G admits a popular matching that is neither stable nor dominant is NP-hard.*

Our reduction will be from 1-in-3 SAT. Recall that 1-in-3 SAT is the set of 3CNF formulas with no negated variables such that there is a satisfying assignment that makes *exactly one* variable true in each clause. Given an input formula ϕ, to determine if ϕ is 1-in-3 satisfiable or not is NP-hard [24]. We will now build a bipartite instance $G = (A \cup B, E)$. For each variable in our 1-in-3 SAT formula ϕ, we construct a variable gadget (in level 1), and for each clause in ϕ, we construct three gadgets in level 0, three in level 2, and one in level 3. There will be 4 more nodes: z, z', a_0, and b_0. If M is any popular matching in G that is neither stable nor dominant and $\vec{\alpha}$ is any witness of M, we will show that the following property holds for every clause c in ϕ:

- if $c = X_i \vee X_j \vee X_k$ then among the gadgets corresponding to X_i, X_j, X_k (in level 1), *exactly one* is in *unit state* in $\vec{\alpha}$ (see Definition 4).

Thus a popular matching in G that is neither stable nor dominant will yield a 1-in-3 satisfying assignment to ϕ by setting to true those variables that correspond to gadgets in unit state. Conversely, we show how to build a popular matching in G that is neither stable nor dominant, if ϕ is 1-in-3 satisfiable. We describe our gadgets below.

Level 1 nodes. Gadgets in level 1 are variable gadgets. Corresponding to a variable X_i, we will have the gadget in Fig. 1 on the 4 nodes x_i, y_i, x'_i, y'_i. Their preference lists are as follows:

$$x_i\colon\ y_i \succ y'_i \succ z \succ \cdots \qquad y_i\colon\ x_i \succ x'_i \succ z' \succ \cdots$$
$$x'_i\colon\ y_i \succ y'_i \succ \cdots \qquad y'_i\colon\ x_i \succ x'_i \succ \cdots$$

The node y_i is the top choice of both x_i, x'_i and the node y'_i is the second choice of both x_i, x'_i. The node x_i is the top choice of both y_i, y'_i and the node x'_i is the second choice of both y_i, y'_i.

The nodes in the gadget corresponding to X_i are also adjacent to nodes in the clause gadgets: these neighbors belong to the "$\cdots$" part of the preference lists. Note that the order among the nodes in the "$\cdots$" part in the above preference lists does not matter.

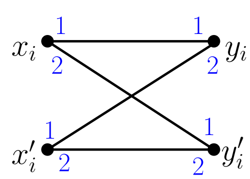

Figure 1: A level 1 gadget.

Let $c = X_i \vee X_j \vee X_k$ be a clause in ϕ. We will describe the gadgets that correspond to c. For the sake of readability, when we describe preference lists below, we drop the superscript c from all the nodes appearing in gadgets corresponding to clause c.

Level 0 nodes. There will be three level 0 gadgets, each on 4 nodes, corresponding to clause c. We describe below the preference lists of $a^c_1, b^c_1, a^c_2, b^c_2$ that belong to the gadget in Fig. 2.

$$a_1\colon\ b_1 \succ \underline{y'_j} \succ b_2 \succ \underline{z} \qquad a_2\colon\ b_2 \succ b_1$$
$$b_1\colon\ a_2 \succ \underline{x'_k} \succ a_1 \succ \underline{z'} \qquad b_2\colon\ a_1 \succ a_2$$

Neighbors that are outside this gadget are underlined. The preferences of nodes in the other two gadgets in level 0 corresponding to c (a^c_t, b^c_t for $t = 3, 4$ and a^c_t, b^c_t for $t = 5, 6$) are analogous. In particular, the node a^c_3's second choice is y'_k and b^c_3's is x'_i; similarly, a^c_5's is y'_i and b^c_5's is x'_j.

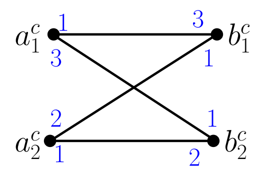

Figure 2: A level 0 gadget.

Level 2 nodes. There are three level 2 gadgets, each on 6 nodes, corresponding to clause c. The preference lists of the nodes p^c_t, q^c_t for $0 \le t \le 2$ that belong to the gadget in Fig. 3 are described below.

$$p_0\colon\ q_0 \succ q_2 \qquad q_0\colon\ p_0 \succ p_2 \succ \underline{z'} \succ \underline{s_0}$$
$$p_1\colon\ q_1 \succ q_2 \succ \underline{z} \qquad q_1\colon\ p_1 \succ p_2$$
$$p_2\colon\ q_0 \succ \underline{y_j} \succ q_1 \succ q_2 \succ \cdots \qquad q_2\colon\ p_1 \succ \underline{x_k} \succ p_0 \succ p_2 \succ \cdots$$

The "$\cdots$" in the preference lists of p_2 and q_2 above are to nodes $t^{c_i}_0$ and $s^{c_i}_0$ respectively (in a level 3 gadget), for *all* clauses c_i. The order among these neighbors is not important.

Let us note that p_2 and q_2 are each other's fourth choices. p_2 regards q_0 as its top choice, y_j as its second choice, and q_1 as its third choice. q_2 regards p_1 as its top choice, x_k as its second choice, and p_0 as its third choice. The preferences of nodes p^c_t, q^c_t for $3 \le t \le 8$ are given below. Again, the "$\cdots$" in the preference lists of p_5 and q_5 above are to nodes $t^{c_i}_0$ and $s^{c_i}_0$ respectively, for all clauses c_i.

$$p_3\colon\ q_3 \succ q_5 \qquad q_3\colon\ p_3 \succ p_5 \succ \underline{z'} \succ \underline{s_0}$$
$$p_4\colon\ q_4 \succ q_5 \succ \underline{z} \succ \underline{t_0} \qquad q_4\colon\ p_4 \succ p_5$$
$$p_5\colon\ q_3 \succ \underline{y_k} \succ q_4 \succ q_5 \succ \cdots \qquad q_5\colon\ p_4 \succ \underline{x_i} \succ p_3 \succ p_5 \succ \cdots$$

The preferences of nodes p^c_t, q^c_t for $6 \le t \le 8$ are described below. The "$\cdots$" in the preference lists of p_8 and q_8 above are to nodes $t^{c_i}_0$ and $s^{c_i}_0$ respectively, for all clauses c_i.

$$p_6\colon\ q_6 \succ q_8 \qquad q_6\colon\ p_6 \succ p_8 \succ \underline{z'}$$
$$p_7\colon\ q_7 \succ q_8 \succ \underline{z} \succ \underline{t_0} \qquad q_7\colon\ p_7 \succ p_8$$
$$p_8\colon\ q_6 \succ \underline{y_i} \succ q_7 \succ q_8 \succ \cdots \qquad q_8\colon\ p_7 \succ \underline{x_j} \succ p_6 \succ p_8 \succ \cdots$$

Level 3 nodes. Gadgets in level 3 are again clause gadgets. There is exactly one level 3 gadget on 8 nodes s^c_i, t^c_i, for $0 \le i \le 3$, corresponding to clause c. As before, we omit the superscript c in describing the

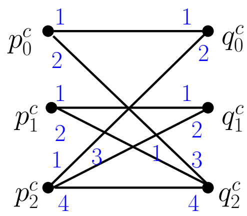

Figure 3: A level 2 gadget.

preference lists. For $1 \leq i \leq 3$, the preference list of s_i is $t_i \succ t_0$ and the preference list of t_i is $s_i \succ s_0$. The preference lists of s_0 and t_0 are given below.

$$s_0 : t_1 \succ \underline{q_0} \succ t_2 \succ \underline{q_3} \succ t_3 \succ \cdots$$

$$t_0 : s_3 \succ \underline{p_7} \succ s_2 \succ \underline{p_4} \succ s_1 \succ \cdots$$

Among neighbors in this gadget, s_0's order is $t_1 \succ t_2 \succ t_3$ while t_0's order is $s_3 \succ s_2 \succ s_1$. Also, s_0's order is interleaved with $q_0 \succ q_3$ (from level 2 gadgets) and t_0's order is interleaved with $p_7 \succ p_4$.

The "$\cdots$" in the preference lists of s_0 and t_0 above are to neighbors in levels 1 and 2. Let n_0 be the number of variables in ϕ. All the nodes $y'_1, \ldots, y'_{n_0}$ along with $q_2^{c_i}, q_5^{c_i}, q_8^{c_i}$ for all clauses c_i will be at the tail of the preference list of s_0^c and the order among all these nodes is not important. Similarly, all the nodes $x'_1, \ldots, x'_{n_0}$ along with $p_2^{c_i}, p_5^{c_i}, p_8^{c_i}$ for all clauses c_i will be at the tail of the preference list of t_0^c and the order among all these nodes is also not important.

There are four more nodes in G. These are $a_0, z' \in A$ and $b_0, z \in B$. Thus we have

$$\begin{aligned} A &= \cup_c\{a_i^c : 1 \leq i \leq 6\} \cup_i \{x_i, x'_i\} \cup_c \{p_i^c : 0 \leq i \leq 8\} \\ &\quad \cup_c\{s_i^c : 0 \leq i \leq 3\} \cup \{a_0, z'\} \\ B &= \cup_c\{b_i^c : 1 \leq i \leq 6\} \cup_i \{y_i, y'_i\} \cup_c \{q_i^c : 0 \leq i \leq 8\} \\ &\quad \cup_c\{t_i^c : 0 \leq i \leq 3\} \cup \{b_0, z\}. \end{aligned}$$

The neighbors of a_0 are b_0, z and the neighbors of b_0 are a_0, z'. The node a_0's preference list is $b_0 \succ z$ and the node b_0's preference list is $a_0 \succ z'$. The set of neighbors of z is $\{a_0\} \cup_i \{x_i\} \cup_c \{a_1^c, a_3^c, a_5^c\} \cup_c \{p_1^c, p_4^c, p_7^c\}$ and the set of neighbors of z' is $\{b_0\} \cup_i \{y_i\} \cup_c \{b_1^c, b_3^c, b_5^c\} \cup_c \{q_0^c, q_3^c, q_6^c\}$. The preference lists of z and z' are as follows (here k is the number of clauses in ϕ):

$$\begin{aligned} z &: x_1 \succ \cdots \succ x_{n_0} \succ p_1^{c_1} \succ \cdots \succ p_7^{c_k} \succ a_0 \succ \cdots \\ z' &: y_1 \succ \cdots \succ y_{n_0} \succ q_0^{c_1} \succ \cdots \succ q_6^{c_k} \succ b_0 \succ \cdots \end{aligned}$$

Thus, z prefers neighbors in level 1 to neighbors in level 2, then comes a_0, and then neighbors in level 0. Analogously, for z' (with b_0 replacing a_0). The order among neighbors in level i (for $i = 0, 1, 2$) in the preference lists of z and z' does not matter.

3.1 Some stable/dominant matchings in G It is helpful to see some stable matchings and dominant matchings in the instance G described above. It will be convenient to refer to nodes in A as *men* and nodes in B as *women*.

- The men-optimal stable matching S in G includes (a_0, b_0) and in the level 0 gadgets, for all clauses c, the edges (a_i^c, b_i^c) for $1 \leq i \leq 6$.
  - In the level 1 gadgets, the edges (x_i, y_i) and (x'_i, y'_i) are included for all $i \in [n_0]$.
  - In the level 2 gadgets, for all clauses c, the edges (p_i^c, q_i^c) for $0 \leq i \leq 8$ are included.
  - In the level 3 gadgets, for all clauses c, the edges (s_i^c, t_i^c) for $1 \leq i \leq 3$ are included.
  - The nodes z, z' and s_0^c, t_0^c for all clauses c are left unmatched in S.

- The women-optimal stable matching S' in G includes (a_0, b_0) and the same edges as S in all level 1, 2, 3 gadgets.
  - In level 0, S' includes the edges $(a_1^c, b_2^c), (a_2^c, b_1^c), (a_3^c, b_4^c), (a_4^c, b_3^c), (a_5^c, b_6^c), (a_6^c, b_5^c)$ for all c.

- The dominant matching M^* as computed by the algorithm in [17] will be as follows:
  - M^* contains the edges $(a_0, z), (z', b_0)$, and in the level 0 gadgets, for all clauses c, the edges $(a_1^c, b_2^c), (a_2^c, b_1^c), (a_3^c, b_4^c), (a_4^c, b_3^c), (a_5^c, b_6^c), (a_6^c, b_5^c)$.
  - In the level 1 gadgets, the edges (x_i, y'_i) and (x'_i, y_i) are included for all $i \in [n_0]$.
  - In the level 2 gadgets, for each clause c, the edges $(p_0^c, q_2^c), (p_1^c, q_1^c), (p_2^c, q_0^c)$ are included from the gadget in Fig. 3. Analogous edges are included from the other two level 2 gadgets corresponding to c.
  - In the level 3 gadgets, the edges $(s_0^c, t_1^c), (s_1^c, t_0^c), (s_2^c, t_2^c), (s_3^c, t_3^c)$ for all clauses c are included.

Note that M^* is a perfect matching as it matches all nodes in G. We can show the above matching M^* to be popular by assigning the following witness $\vec{\alpha} \in \{\pm 1\}^n$ to M^*:

- $\alpha_{a_0} = \alpha_{b_0} = 1$ while $\alpha_z = \alpha_{z'} = -1$.

- $\alpha_{a_i^c} = 1$ and $\alpha_{b_i^c} = -1$ for $i = 1, \ldots, 6$ and all clauses c.
- $\alpha_{x_i} = \alpha_{y_i} = 1$ while $\alpha_{x'_i} = \alpha_{y'_i} = -1$ for all $i \in [n_0]$.
- $\alpha_{p_0^c} = \alpha_{q_0^c} = \alpha_{p_1^c} = 1$ while $\alpha_{q_1^c} = \alpha_{p_2^c} = \alpha_{q_2^c} = -1$ for all clauses c. Similarly for the other 2 level 2 gadgets corresponding to c and all other clauses.
- $\alpha_{s_1^c} = \alpha_{t_1^c} = \alpha_{s_2^c} = \alpha_{s_3^c} = 1$ while $\alpha_{s_0^c} = \alpha_{t_0^c} = \alpha_{t_2^c} = \alpha_{t_3^c} = -1$ for all clauses c.

It can be checked that we have $\alpha_u + \alpha_v = 0$ for every edge $(u, v) \in M^*$. We also have $\alpha_u + \alpha_v \geq \mathsf{wt}_{M^*}(u, v)$ for every edge (u, v) in the graph. In particular, the endpoints of every blocking edge to M^*, such as (a_0, b_0), (x_i, y_i) for all i, (p_{3j}^c, q_{3j}^c) for all c and $j \in \{0, 1, 2\}$, and (s_1^c, t_1^c) for all c, have their α-value equal to 1.

There are many other dominant matchings in this instance G:

- The edges (a_i^c, b_i^c) may be included for $i \in \{1, \cdots, 6\}$ and all clauses c.
- Edges such as $(p_0^c, q_0^c), (p_1^c, q_2^c), (p_2^c, q_1^c)$ (see Fig. 3) may be included from a level 2 gadget.
- From the level 3 gadget corresponding to c, the edges $(s_0^c, t_2^c), (s_1^c, t_1^c), (s_2^c, t_0^c), (s_3^c, t_3^c)$ or the edges $(s_0^c, t_3^c), (s_1^c, t_1^c), (s_2^c, t_2^c), (s_3^c, t_0^c)$ may be included.

3.2 Useful properties of G Recall the popular subgraph F_G from Section 2, whose edge set is the set of popular edges in G.

LEMMA 3.1. *Let C be any level i gadget in G, where $i \in \{0, 1, 2, 3\}$. All the nodes in C belong to the same connected component in F_G.*

Proof. Consider a level 0 gadget in G, say on $a_1^c, b_1^c, a_2^c, b_2^c$ (see Fig. 2). The men-optimal stable matching S in G contains the edges (a_1^c, b_1^c) and (a_2^c, b_2^c) while the women-optimal stable matching S' contains the edges (a_1^c, b_2^c) and (a_2^c, b_1^c). Thus there are popular edges among these 4 nodes and so these 4 nodes belong to the same connected component in F_G.

Consider a level 1 gadget in G, say on x_i, y_i, x'_i, y'_i (see Fig. 1). Every stable matching in G contains (x_i, y_i) and (x'_i, y'_i) while the dominant matching M^* contains (x_i, y'_i) and (x'_i, y_i). Thus there are popular edges among these 4 nodes and so these 4 nodes belong to the same connected component in F_G.

Consider a level 2 gadget in G, say on p_i^c, q_i^c for $i = 0, 1, 2$ (see Fig. 3). The dominant matching M^* contains the cross edges (p_0^c, q_2^c) and (p_2^c, q_0^c). There is also another dominant matching in G that contains the cross edges (p_1^c, q_2^c) and (p_2^c, q_1^c). Thus there are popular edges among these 6 nodes and so these 6 nodes belong to the same connected component in F_G.

Consider a level 3 gadget in G, say on s_i^c, t_i^c for $i = 0, 1, 2, 3$. The dominant matching M^* contains (s_0^c, t_1^c), and (s_1^c, t_0^c). There is another dominant matching in G that contains (s_0^c, t_2^c) and (s_2^c, t_0^c). There is yet another dominant matching in G that contains (s_0^c, t_3^c) and (s_3^c, t_0^c). Thus there are popular edges among these 8 nodes and so these 8 nodes belong to the same connected component in F_G.

The following theorem on the popular subgraph F_G of G will be proved in Section 3.5.

THEOREM 3.2. *Every level i gadget, for $i \in \{0, 1, 2, 3\}$, forms a distinct connected component in the graph F_G. Nodes a_0, b_0, z, and z' form a separate connected component of F_G.*

Let M be any popular matching in G. Note that M either matches *both* z and z' or leaves both these nodes unmatched. This is because both z and z' are unstable nodes in the same connected component in F_G (by Theorem 3.2), so either both are matched or both are unmatched in M (by Lemma 2.1). It is similar with nodes s_0^c and t_0^c for any clause c: any popular matching either matches both s_0^c and t_0^c or leaves both these nodes unmatched.

A dominant matching in G is perfect and a stable matching in G matches all nodes, except z, z' and s_0^c, t_0^c for all c. Theorem 3.2 allows us to show the following two lemmas.

LEMMA 3.2. *A popular matching in G that matches either z or z' is dominant.*

Proof. Let M be as in the hypothesis of the lemma and $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M. It follows from Theorem 3.2 that (a_0, z) and (z', b_0) are in M. Since z and z' prefer their neighbors in level 1 to a_0 and b_0 respectively while these neighbors prefer their partners in M to z and z' (by Theorem 3.2), we have $\mathsf{wt}_M(x_i, z) = \mathsf{wt}_M(z', y_i) = 0$.

The nodes z and z' are unstable in G and M matches them, so $\alpha_z = \alpha_{z'} = -1$ (by Lemma 2.2). Since $\alpha_{x_i} + \alpha_z \geq 0$ and $\alpha_{z'} + \alpha_{y_i} \geq 0$, it follows that $\alpha_{x_i} = \alpha_{y_i} = 1$ for all i. Thus all nodes in level 1 have α-values equal to ± 1 (by Lemma 2.1). In particular, $\alpha_{x'_i} = \alpha_{y'_i} = -1$. This is due to the fact that $\mathsf{wt}_M(x_i, y'_i) = 0$ (by Theorem 3.2) and $\alpha_{x_i} + \alpha_{y'_i} = \mathsf{wt}_M(x_i, y'_i)$ (by Lemma 2.2) as (x_i, y'_i) is a popular edge. Similarly, with (x'_i, y_i).

Suppose s_0^c, t_0^c are unmatched in M. Then $\mathsf{wt}_M(s_0^c, y'_i) = \mathsf{wt}_M(x'_i, t_0^c) = 0$. This is because s_0^c and

t_0^c prefer to be matched to any neighbor than be unmatched while (by Theorem 3.2) y_i' and x_i' prefer their partners in M to s_0^c and t_0^c, respectively. Since we assumed s_0^c, t_0^c to be unmatched in M, $\alpha_{s_0^c} = \alpha_{t_0^c} = 0$ (by Lemma 2.2). So this implies that $\alpha_{s_0} + \alpha_{y_i'} = 0 - 1 < \mathsf{wt}_M(s_0^c, y_i')$, i.e., the edge (s_0^c, y_i') (similarly, (x_i', t_0^c)) is not covered by the sum of α-values of its endpoints, a contradiction. Thus s_0^c, t_0^c are forced to be matched in M.

Thus s_0^c and t_0^c for all clauses c are matched in M. The other nodes in G are stable and hence they have to be matched in M. Thus M is a perfect matching and also popular, so it is a dominant matching in G.

LEMMA 3.3. *A popular matching in G that leaves either s_0^c or t_0^c unmatched for some c is stable.*

Proof. We will repeatedly use Theorem 3.2 here. Let M be as in the hypothesis of the lemma and $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M. Since the nodes s_0^c and t_0^c are unmatched in M, we have $\alpha_{s_0^c} = \alpha_{t_0^c} = 0$ (by Lemma 2.2). Also $\mathsf{wt}_M(s_0^c, q_2^c) = \mathsf{wt}_M(p_2^c, t_0^c) = 0$ since both s_0^c and t_0^c prefer to be matched than be unmatched while q_2^c and p_2^c prefer their partners in M to s_0^c and t_0^c, respectively (by Theorem 3.2). So $\alpha_{p_2^c} \geq 0$ and $\alpha_{q_2^c} \geq 0$ for all c.

We also have $\alpha_{p_2^c} + \alpha_{q_2^c} = \mathsf{wt}_M(p_2^c, q_2^c) \leq 0$ since any popular matching matches p_2^c to a partner at least as good as q_2^c and similarly, q_2^c to a partner at least as good as p_2^c (by Theorem 3.2). This means that $\alpha_{p_2^c} = \alpha_{q_2^c} = 0$. The same argument can be used for every $p_{3j+2}^{c_i}$ and $q_{3j+2}^{c_i}$ (for any clause c_i and $j = 0, 1, 2$) to show that $\alpha_{p_{3j+2}^{c_i}} = \alpha_{q_{3j+2}^{c_i}} = 0$. Thus all level 2 nodes have α-values equal to 0 (by Lemmas 2.1 and 3.1).

The fact that all level 2 nodes have α-values equal to 0 immediately implies that all level 3 nodes also have α-values equal to 0. This is because if M matches $s_0^{c_i}$ and $t_0^{c_i}$ for some clause c_i then at least one of $s_0^{c_i}, t_0^{c_i}$ is not matched to its top choice neighbor in its gadget. So either $\mathsf{wt}_M(s_0^{c_i}, q_0^{c_i}) = 0$ or $\mathsf{wt}_M(p_7^{c_i}, t_0^{c_i}) = 0$. Since $\alpha_{q_0^{c_i}} = \alpha_{p_7^{c_i}} = 0$ (these are level 2 nodes) and $\alpha_{s_0^{c_i}} = \alpha_{t_0^{c_i}} = -1$ (by Lemma 2.2), we have a contradiction. Thus $\alpha_u = 0$ for every level 3 node u (by Lemma 2.1).

Similarly, $\alpha_{x_i'} \geq 0$ and $\alpha_{y_i'} \geq 0$ for all i as the edges (x_i', t_0^c) and (s_0^c, y_i') would not be covered otherwise. Also, $\alpha_{x_i'} + \alpha_{y_i'} = \mathsf{wt}_M(x_i', y_i')$ (by Lemma 2.2) as (x_i', y_i') is a popular edge and $\mathsf{wt}_M(x_i', y_i') \leq 0$ since M matches x_i' to either y_i or y_i' and similarly, y_i' to either x_i or x_i' (by Theorem 3.2). Thus $\alpha_{x_i'} = \alpha_{y_i'} = 0$. This means that all level 1 nodes have α-values equal to 0 (by Lemma 2.1).

Since all level 1 nodes have α-values equal to 0, we have $\alpha_z = \alpha_{z'} = 0$; otherwise the edges (x_i, z) and (z', y_i) would not be covered. This is because $\mathsf{wt}_M(x_i, z) = \mathsf{wt}_M(z', y_i) = 0$ (by Theorem 3.2). So in order to cover the edges (x_i, z) and (z', y_i), we need to have $\alpha_z \geq 0$ and $\alpha_{z'} \geq 0$, i.e., $\alpha_z = \alpha_{z'} = 0$ (by Lemma 2.2). Thus $\alpha_{a_0} = \alpha_{b_0} = 0$ (by Theorem 3.2 and Lemma 2.1).

Moreover, $\alpha_z = \alpha_{z'} = 0$ also implies that all their neighbors in level 0 have their α-values at least 0. For instance, consider a_1^c and b_1^c: in order to cover the edges (a_1^c, z) and (z', b_1^c), we have $\alpha_{a_1^c} \geq 0$ and $\alpha_{b_1^c} \geq 0$. Since either $(a_1^c, b_1^c) \in M$ or $(a_1^c, b_2^c), (a_2^c, b_1^c)$ are in M (by Theorem 3.2), we have $\mathsf{wt}_M(a_1^c, b_1^c) = 0$. Because (a_1^c, b_1^c) is a popular edge, this means $\alpha_{a_1^c} + \alpha_{b_1^c} = 0$ (by Lemma 2.2). Thus $\alpha_{a_1^c} = \alpha_{b_1^c} = 0$.

Similarly, $\alpha_{a_{2i-1}^c} = \alpha_{b_{2i-1}^c} = 0$ for $i = 1, 2, 3$ and all clauses c. Thus all level 0 nodes have α-values equal to 0. So $\vec{\alpha} = \vec{0}$, i.e., $\mathsf{wt}_M(e) \leq 0$ for all edges e. In other words, there is no blocking edge to M. Thus M is a stable matching.

It follows from Lemmas 3.2 and 3.3 that if M is a popular matching in G that is neither stable nor dominant, then M has to match nodes s_0^c, t_0^c for all c and leave z, z' unmatched. Since a popular matching has to match all *stable* nodes [13], M has to match all nodes except z and z'.

Conversely, if M is a popular matching in G that matches all nodes except z and z' then M is neither a max-size popular matching nor a min-size popular matching, i.e., M is neither dominant nor stable. Thus we can conclude the following theorem.

THEOREM 3.3. *The graph G admits a popular matching that is neither stable nor dominant if and only if G admits a popular matching that matches all nodes except z and z'.*

3.3 Desired popular matchings in G We will call a matching M in G that matches all nodes except z and z' a *desired popular matching* here. Let M be such a matching and let $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M, where n is the number of nodes in G.

Recall Definition 4 from Section 2. We say a gadget is in *unit* (similarly, *zero*) state in $\vec{\alpha}$ if for any node u in this gadget, we have $\alpha_u \in \{\pm 1\}$ (resp., $\alpha_u = 0$). The following two observations will be important here.

1. All level 3 gadgets have to be in *unit* state in $\vec{\alpha}$.

2. All level 0 gadgets have to be in *zero* state in $\vec{\alpha}$.

The nodes s_0^c and t_0^c, for all clauses c, are left unmatched in any stable matching in G. Since M has to match the unstable nodes s_0^c and t_0^c for all clauses c, $\alpha_{s_0^c} = \alpha_{t_0^c} = -1$ for all c (by Lemma 2.2). Thus the first observation follows from Lemmas 2.1 and 3.1.

The proof of the second observation was already seen in the last two paragraphs of the proof of Lemma 3.3, where it was shown that $\alpha_z = \alpha_{z'} = 0$ implies that $\alpha_{a^c_{2i-1}} = \alpha_{b^c_{2i-1}} = 0$ for $i = 1, 2, 3$ and all clauses c. Thus all level 0 gadgets have to be in zero state in $\vec{\alpha}$.

Lemmas 3.4-3.6 are easy to show and are crucial to our proof. Let $c = X_i \vee X_j \vee X_k$ be a clause in ϕ. Below, we are omitting the superscript c from node names for the sake of readability. Recall that $\vec{\alpha} \in \{0, \pm 1\}^n$ is a witness of our "desired popular matching" M.

Lemma 3.4. *For every clause c in ϕ, at least two of the three level 2 gadgets corresponding to c have to be in unit state in $\vec{\alpha}$.*

Proof. Let c be any clause in ϕ. We know from observation 1 above that the level 3 gadget corresponding to c is in *unit* state in $\vec{\alpha}$. So $\alpha_{s_0} = \alpha_{t_0} = -1$. Also, one of the following three cases holds: (1) (s_0, t_1) and (s_1, t_0) are in M, (2) (s_0, t_2) and (s_2, t_0) are in M, (3) (s_0, t_3) and (s_3, t_0) are in M.

- In case (1), the node t_0 prefers p_4 and p_7 to its partner s_1 in M. Thus $\mathsf{wt}_M(p_4, t_0) = \mathsf{wt}_M(p_7, t_0) = 0$. Since $\alpha_{t_0} = -1$, we need to have $\alpha_{p_4} = \alpha_{p_7} = 1$ so that $\alpha_{p_4} + \alpha_{t_0} \geq \mathsf{wt}_M(p_4, t_0)$ and $\alpha_{p_7} + \alpha_{t_0} \geq \mathsf{wt}_M(p_7, t_0)$. Thus the level 2 gadgets on p_i, q_i for $3 \leq i \leq 8$ corresponding to c have to be in unit state in $\vec{\alpha}$.
- In case (2), the node t_0 prefers p_7 to its partner s_2 in M and the node s_0 prefers q_0 to its partner t_2 in M. Thus $\alpha_{p_7} = \alpha_{q_0} = 1$ so that $\alpha_{p_7} + \alpha_{t_0} \geq \mathsf{wt}_M(p_7, t_0)$ and $\alpha_{s_0} + \alpha_{q_0} \geq \mathsf{wt}_M(s_0, q_0)$. Thus the level 2 gadgets on p_i, q_i for $0 \leq i \leq 2$ and for $6 \leq i \leq 8$ corresponding to c have to be in unit state in $\vec{\alpha}$.
- In case (3), the node s_0 prefers q_0 and q_3 to its partner t_3 in M. Thus $\alpha_{q_0} = \alpha_{q_3} = 1$ so that $\alpha_{s_0} + \alpha_{q_0} \geq \mathsf{wt}_M(s_0, q_0)$ and $\alpha_{s_0} + \alpha_{q_3} \geq \mathsf{wt}_M(s_0, q_3)$. Thus the level 2 gadgets on p_i, q_i for $0 \leq i \leq 5$ corresponding to c have to be in unit state in $\vec{\alpha}$.

Lemma 3.5. *For any clause c in ϕ,* at least one *of the level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.*

Proof. We showed in Lemma 3.4 that at least two of the three level 2 gadgets corresponding to c are in unit state in $\vec{\alpha}$. We have three cases here:

(i) the gadgets on p_i, q_i for $0 \leq i \leq 5$ are in unit state in $\vec{\alpha}$, (ii) the gadgets on p_i, q_i for $0 \leq i \leq 2$ and $6 \leq i \leq 8$ are in unit state in $\vec{\alpha}$, (iii) the gadgets on p_i, q_i for $3 \leq i \leq 8$ are in unit state in $\vec{\alpha}$.

Let us consider case (i) first. It follows from the proof of Lemma 3.4 that $\alpha_{q_0} = \alpha_{q_3} = 1$. This also forces $\alpha_{p_1} = \alpha_{p_4} = 1$. This is because α_{p_1} and α_{p_4} have to be non-negative since p_1 and p_4 are neighbors of the unmatched node z. And so by Lemma 2.1, $\alpha_{p_1} = 1$ and $\alpha_{p_4} = 1$.

As q_0 and p_1 are the most preferred neighbors of p_2 and q_2 while p_2 and q_2 are the least preferred neighbors of q_0 and p_1 in M (by Theorem 3.2), we have $\mathsf{wt}_M(p_2, q_0) = \mathsf{wt}_M(p_1, q_2) = 0$. Since (p_2, q_0) and (p_1, q_2) are popular edges, it follows from Lemma 2.2 that $\alpha_{p_2} + \alpha_{q_0} = 0$ and $\alpha_{p_1} + \alpha_{q_2} = 0$. Thus $\alpha_{p_2} = \alpha_{q_2} = -1$ and so $(p_2, q_2) \notin M$. So either $(p_2, q_0), (p_0, q_2)$ are in M or $(p_2, q_1), (p_1, q_2)$ are in M (by Theorem 3.2). This means that either $\mathsf{wt}_M(p_2, y_j) = 0$ or $\mathsf{wt}_M(x_k, q_2) = 0$. That is, either $\alpha_{y_j} = 1$ or $\alpha_{x_k} = 1$.

Similarly, $\mathsf{wt}_M(p_5, q_3) = \mathsf{wt}_M(p_4, q_5) = 0$ and we can conclude that $\alpha_{p_5} = \alpha_{q_5} = -1$. Thus $(p_5, q_5) \notin M$ and either $(p_5, q_3), (p_3, q_5)$ are in M or $(p_5, q_4), (p_4, q_5)$ are in M (by Theorem 3.2). This means that either $\mathsf{wt}_M(p_5, y_k) = 0$ or $\mathsf{wt}_M(x_i, q_5) = 0$. That is, either $\alpha_{y_k} = 1$ or $\alpha_{x_i} = 1$. Thus either (1) the gadgets corresponding to variables X_i and X_j are in unit state or (2) the gadget corresponding to X_k is in unit state in $\vec{\alpha}$. Thus in this case at least *one* of the level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.

The proofs of case (ii) and case (iii) are quite similar. Let us consider case (ii) next. It follows from the proof of Lemma 3.4 that $\alpha_{q_0} = \alpha_{p_7} = 1$. This also forces $\alpha_{p_1} = \alpha_{q_6} = 1$ and $\alpha_{p_2} = \alpha_{q_2} = -1$. By the same reasoning as in case (i), we have either $\mathsf{wt}_M(p_2, y_j) = 0$ or $\mathsf{wt}_M(x_k, q_2) = 0$. That is, either $\alpha_{y_j} = 1$ or $\alpha_{x_k} = 1$. Similarly, $\alpha_{p_8} = \alpha_{q_8} = -1$ and either $\mathsf{wt}_M(p_8, y_i) = 0$ or $\mathsf{wt}_M(x_j, q_8) = 0$, i.e., $\alpha_{y_i} = 1$ or $\alpha_{x_j} = 1$.

So either (1) the gadgets corresponding to variables X_i and X_k are in unit state or (2) the gadget corresponding to X_j is in unit state in $\vec{\alpha}$. Thus in this case also at least one of the level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.

In case (iii), it follows from the proof of Lemma 3.4 that $\alpha_{p_4} = \alpha_{p_7} = 1$. This forces $\alpha_{q_3} = \alpha_{q_6} = 1$ and $\alpha_{p_5} = \alpha_{q_5} = -1$. By the same reasoning as in case (i), we have either $\mathsf{wt}_M(p_5, y_k) = 0$ or $\mathsf{wt}_M(x_i, q_5) = 0$. That is, either $\alpha_{y_k} = 1$ or $\alpha_{x_i} = 1$. Similarly, either $\alpha_{y_i} = 1$ or $\alpha_{x_j} = 1$. Thus either (1) the gadgets corresponding to variables X_j and X_k are in unit state or (2) the gadget corresponding to X_i is in unit state in $\vec{\alpha}$. Thus in this case also at least one of the level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.

Lemma 3.6. *For any clause c in ϕ,* at most one *of the level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.*

Proof. We know from observation 2 made at the start of this section that all the three level 0 gadgets corresponding to c are in zero state in $\vec{\alpha}$. So $\alpha_{a_t} = \alpha_{b_t} = 0$ for $1 \leq t \leq 6$. We know from Theorem 3.2 that either $(a_1, b_1), (a_2, b_2)$ are in M or $(a_1, b_2), (a_2, b_1)$ are in M. So either $\mathsf{wt}_M(a_1, y'_j) = 0$ or $\mathsf{wt}_M(x'_k, b_1) = 0$. So either $\alpha_{y'_j} \geq 0$ or $\alpha_{x'_k} \geq 0$.

Consider any variable X_r. We know from Theorem 3.2 that either $\{(x_r, y'_r), (x'_r, y_r)\} \subseteq M$ or $\{(x_r, y_r), (x'_r, y'_r)\} \subseteq M$. It follows from Lemma 2.2 that $\alpha_{x_r} + \alpha_{y'_r} = \mathsf{wt}_M(x_r, y'_r) = 0$ and $\alpha_{x'_r} + \alpha_{y_r} = \mathsf{wt}_M(x'_r, y_r) = 0$. Also due to the nodes z and z', we have $\alpha_{x_r} \geq 0$ and $\alpha_{y_r} \geq 0$. Thus $\alpha_{y'_r} \leq 0$ and $\alpha_{x'_r} \leq 0$.

Hence we can conclude that either $\alpha_{y'_j} = 0$ or $\alpha_{x'_k} = 0$. In other words, either the gadget corresponding to X_j or the gadget corresponding to X_k is in zero state. Similarly, by analyzing the level 0 gadget on nodes a^c_t, b^c_t for $t = 3, 4$, we can show that either the gadget corresponding to X_k or the gadget corresponding to X_i is in zero state. Also, by analyzing the level 0 gadget on nodes a^c_t, b^c_t for $t = 5, 6$, either the gadget corresponding to X_i or the gadget corresponding to X_j is in zero state.

Thus at least 2 of the 3 gadgets corresponding to variables in clause c are in zero state in $\vec{\alpha}$. Hence at most 1 of the 3 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$.

THEOREM 3.4. *If G admits a desired popular matching then ϕ has a 1-in-3 satisfying assignment.*

Proof. Let M be a desired popular matching in G. That is, M matches all nodes except z and z'. Let $\vec{\alpha} \in \{0, \pm 1\}^n$ be a witness of M.

We will now define a $\mathsf{true}/\mathsf{false}$ assignment for the variables in ϕ. For each variable X_r in ϕ do:

- If the level 1 gadget corresponding to X_r is in *unit* state in $\vec{\alpha}$, i.e., if $\alpha_{x_r} = \alpha_{y_r} = 1$ and $\alpha_{x'_r} = \alpha_{y'_r} = -1$ or equivalently, if (x_r, y'_r) and (x'_r, y_r) are in M, then set X_r to true.

- Else set X_r to false, i.e., the level 1 gadget corresponding to X_r is in *zero* state in $\vec{\alpha}$ or equivalently, (x_r, y_r) and (x'_r, y'_r) are in M.

Since M is our desired popular matching, it follows from Lemmas 3.5 and 3.6 that for every clause c in ϕ, *exactly one* of the three level 1 gadgets corresponding to variables in c is in unit state in $\vec{\alpha}$. When the gadget X_r is in unit state, we have $\alpha_{x_r} = \alpha_{y_r} = 1$ and $\alpha_{x'_r} = \alpha_{y'_r} = -1$. This is due to the fact that $\alpha_{x_r} \geq 0$ and $\alpha_{y_r} \geq 0$ because of the nodes z and z', respectively.

Thus X_r is in unit state in $\vec{\alpha}$ if and only if the edges (x_r, y'_r) and (x'_r, y_r) are in M. Hence for each clause c in ϕ, exactly one of the variables in c is set to true. Hence this is a 1-in-3 satisfying assignment for ϕ.

3.4 The converse

Suppose ϕ admits a 1-in-3 satisfying assignment. We will now use this assignment to construct a desired popular matching M in G. The edge (a_0, b_0) is in M. For each variable X_r in ϕ do:

- if $X_r = \mathsf{true}$ then include the edges (x_r, y'_r) and (x'_r, y_r) in M;

- else include the edges (x_r, y_r) and (x'_r, y'_r) in M.

Consider a clause $c = X_i \vee X_j \vee X_k$. We know that exactly one of X_i, X_j, X_k is set to true in our assignment. Assume without loss of generality that $X_j = \mathsf{true}$. We will include the following edges in M from all the gadgets corresponding to c. Corresponding to the level 0 gadgets for c:

- Add the edges $(a^c_1, b^c_1), (a^c_2, b^c_2)$ and $(a^c_5, b^c_6), (a^c_6, b^c_5)$ to M.

  We will select $(a^c_3, b^c_3), (a^c_4, b^c_4)$ from the gadget on $a^c_3, b^c_3, a^c_4, b^c_4$.

  (Note that we could also have selected $(a^c_3, b^c_4), (a^c_4, b^c_3)$ from this gadget.)

Corresponding to the level 2 gadgets for c:

- Add the edges $(p^c_0, q^c_0), (p^c_2, q^c_1), (p^c_1, q^c_2)$ from the first gadget, $(p^c_3, q^c_3), (p^c_4, q^c_4), (p^c_5, q^c_5)$ from the second gadget, and $(p^c_6, q^c_8), (p^c_7, q^c_7), (p^c_8, q^c_6)$ from the third gadget to M.

Since the first and third level 2 gadgets are dominant, we will include (s^c_0, t^c_2) and (s^c_2, t^c_0) in M. Hence

- Add the edges $(s^c_0, t^c_2), (s^c_1, t^c_1), (s^c_2, t^c_0), (s^c_3, t^c_3)$ to M.

Thus M matches all nodes except z and z'. We will show the following theorem now.

THEOREM 3.5. *The matching M described above is a popular matching in G.*

Proof. We will prove M's popularity by describing a witness $\vec{\alpha} \in \{0, \pm 1\}^n$. That is, $\sum_{u \in A \cup B} \alpha_u$ will be 0 and every edge will be covered by the sum of α-values of its endpoints, i.e., $\alpha_u + \alpha_v \geq \mathsf{wt}_M(u, v)$ for all edges (u, v) in E. We will also have $\alpha_u \geq \mathsf{wt}_M(u, u)$ for all nodes u.

Set $\alpha_{a_0} = \alpha_{b_0} = \alpha_z = \alpha_{z'} = 0$. Also set $\alpha_u = 0$ for all nodes u in the gadgets with *no* "blocking edges". This includes all level 0 gadgets, and the gadgets in level 1 that correspond to variables set to false, and also the level 2 gadgets such as the gadget with nodes $p^c_3, q^c_3, p^c_4, q^c_4, p^c_5, q^c_5$ since we assumed $X_j = \mathsf{true}$.

For every variable X_r assigned to true: set $\alpha_{x_r} = \alpha_{y_r} = 1$ and $\alpha_{x'_r} = \alpha_{y'_r} = -1$. For every clause, consider the level 2 gadgets corresponding to this clause with "blocking edges": for our clause c, these are the gadgets on p_i, q_i for $0 \le i \le 2$ and $6 \le i \le 8$ (since we assumed $X_j = \mathsf{true}$).

Recall that we included in M the edges $(p_0^c, q_0^c), (p_2^c, q_1^c), (p_1^c, q_2^c)$ from the first gadget. We will set $\alpha_{q_0^c} = \alpha_{p_1^c} = \alpha_{q_1^c} = 1$ and $\alpha_{p_0^c} = \alpha_{p_2^c} = \alpha_{q_2^c} = -1$. We also included in M the edges $(p_6^c, q_8^c), (p_7^c, q_7^c), (p_8^c, q_6^c)$ from the third gadget. We will set $\alpha_{p_6^c} = \alpha_{q_6^c} = \alpha_{p_7^c} = 1$ and $\alpha_{q_7^c} = \alpha_{p_8^c} = \alpha_{q_8^c} = -1$.

In the level 3 gadget corresponding to c, we included in M the edges $(s_0^c, t_2^c), (s_1^c, t_1^c), (s_2^c, t_0^c)$, (s_3^c, t_3^c). We will set $\alpha_{t_1^c} = \alpha_{s_2^c} = \alpha_{t_2^c} = \alpha_{s_3^c} = 1$ and $\alpha_{s_0^c} = \alpha_{t_0^c} = \alpha_{s_1^c} = \alpha_{t_3^c} = -1$.

The claim below shows that $\vec{\alpha}$ is indeed a valid witness for M. Thus M is a popular matching.

Claim 1. *The vector $\vec{\alpha}$ defined above is a witness to M's popularity.*

Proof. For any edge $(u, v) \in M$, we have $\alpha_u + \alpha_v = 0$, also $\alpha_z = \alpha_{z'} = 0$. Thus $\sum_{u \in A \cup B} \alpha_u = 0$. For any neighbor v of z or z', we have $\alpha_v \ge 0$. Thus all edges incident to z or z' are covered by the sum of α-values of their endpoints. It is also easy to see that for every intra-gadget edge (u, v), we have $\alpha_u + \alpha_v \ge \mathsf{wt}_M(u, v)$.

In particular, the endpoints of every blocking edge to M have their α-value set to 1. When $X_j = \mathsf{true}$, the edge (x_j, y_j) is a blocking edge to M and so are $(p_1^c, q_1^c), (p_6^c, q_6^c), (s_2^c, t_2^c)$ in the gadgets involving clause c. We will now check that the edge covering constraint holds for all edges (u, v) where u and v belong to different levels.

- Consider edges in G between a level 0 gadget and a level 1 gadget. When $X_j = \mathsf{true}$, the edges (a_1^c, y_j') and (x_j', b_5^c) are the most interesting as they have one endpoint in a gadget with α-values 0 and another endpoint in a gadget with α-values equal to ± 1.

  Observe that both these edges are labeled $(-,-)$. This is because a_1^c prefers its partner b_1^c to y_j' and symmetrically, y_j' prefers its partner x_j to a_1^c. Thus $\mathsf{wt}_M(a_1^c, y_j') = -2 < \alpha_{a_1^c} + \alpha_{y_j'} = 0 - 1$. Similarly, b_5^c prefers its partner a_6^c to x_j' and symmetrically, x_j' prefers its partner y_j to b_5^c. Thus $\mathsf{wt}_M(x_j', b_5^c) = -2 < \alpha_{x_j'} + \alpha_{b_5^c} = -1 + 0$.

- We will now consider edges in G between a level 1 gadget and a level 2 gadget. We have $\mathsf{wt}_M(p_2^c, y_j) = 0$ since p_2^c prefers y_j to its partner q_1^c while y_j prefers its partner x_j' to p_2^c. We have $\alpha_{p_2^c} + \alpha_{y_j} = -1 + 1 = \mathsf{wt}_M(p_2^c, y_j) = 0$. The edge (x_k, q_2^c) is labeled $(-,-)$ and we have $\alpha_{x_k} = 0$ and $\alpha_{q_2} = -1$.

  Similarly, the edge (p_8^c, y_i) is labeled $(-,-)$ and so this is covered by the sum of α-values of its endpoints. We have $\mathsf{wt}_M(x_j, q_8^c) = 0 = 1 - 1 = \alpha_{x_j} + \alpha_{q_8^c}$. We also have $\mathsf{wt}_M(p_5^c, y_k) = 0$ and $\alpha_{p_5^c} = \alpha_{y_k} = 0$. Similarly, $\mathsf{wt}_M(x_i, q_5^c) = 0$ and $\alpha_{x_i} = \alpha_{q_5^c} = 0$. Thus all these edges are covered.

- We will now consider edges in G between a level 2 gadget and a level 3 gadget. First, consider the edges $(s_0^c, q_0^c), (s_0^c, q_3^c), (p_7^c, t_0^c), (p_4^c, t_0^c)$. We have $\mathsf{wt}_M(s_0^c, q_0^c) = 0$ and $\alpha_{s_0^c} = -1, \alpha_{q_0^c} = 1$, so this edge is covered. Similarly, $\mathsf{wt}_M(p_7^c, t_0^c) = 0$ and $\alpha_{p_7^c} = 1, \alpha_{t_0^c} = -1$. The edges (s_0^c, q_3^c) and (p_4^c, t_0^c) are labeled $(-,-)$, so they are also covered. Next consider the edges $(s_0^c, q_{3j+2}^{c_i})$ and $(p_{3j+2}^{c_i}, t_0^c)$ for any clause c_i and $j \in \{0, 1, 2\}$. It is easy to see that these edges are labeled $(-,-)$, so these edges are also covered.

- Finally consider the edges between a level 1 gadget and a level 3 gadget. Corresponding to clause c, these edges are (s_0^c, y_i') and (x_i', t_0^c) for any $i \in [n_0]$. It is again easy to see that these edges are labeled $(-,-)$ and so they are covered. Thus it follows that $\vec{\alpha}$ is a witness to M's popularity. $\Diamond$

We have shown a polynomial time reduction from 1-in-3 SAT to the problem of deciding if $G = (A \cup B, E)$ admits a popular matching that matches all nodes except z and z'. That is, we have shown the following theorem.

Theorem 3.6. *The instance $G = (A \cup B, E)$ admits a popular matching that matches all nodes except z and z' if and only if ϕ is in 1-in-3 SAT.*

3.5 Proof of Theorem 3.2

Let $c = X_i \vee X_j \vee X_k$ be a clause in ϕ. We will show in the following claims that no edge between 2 different gadgets can be popular.

Claim 2. *No edge between a level 0 node and a level 1 node is popular.*

Proof. Consider any such edge in G, say (a_1^c, y_j'). In order to show this edge cannot be present in any popular matching, we will show a popular matching S along with a witness $\vec{\alpha}$ such that $\alpha_{a_1^c} + \alpha_{y_j'} > \mathsf{wt}_S(a_1^c, y_j')$. Then it will immediately follow from the *slackness* of this edge that (a_1^c, y_j') is not used in any popular matching (by Lemma 2.2).

Let S be the men-optimal stable matching in G. The vector $\vec{\alpha} = \vec{0}$ is a witness to S. The edges (a_1^c, b_1^c) and (x_j', y_j') belong to S, so we have $\mathsf{wt}(a_1^c, y_j') = -2$ while $\alpha_{a_1^c} = \alpha_{y_j'} = 0$. Thus (a_1^c, y_j') is not a popular edge. We can similarly show that (x'_k, b^c_1) is not a popular edge by considering the women-optimal stable matching S'. $\Diamond$

CLAIM 3. *No edge between a level 1 node and a level 2 node is popular.*

Proof. Consider any such edge in G, say (p^c_2, y_j). Consider the dominant matching M^* that contains the edges $(p^c_0, q^c_2), (p^c_2, q^c_0)$ and also $(x_j, y'_j), (x'_j, y_j)$. Note that $\mathsf{wt}_{M^*}(p^c_2, y_j) = -2$.

A witness $\vec{\beta}$ to M^* sets $\beta_{p^c_2} = \beta_{q^c_2} = -1$ and $\beta_{x_j} = \beta_{y_j} = 1$. This is because (x_j, y_j) and (p^c_0, q^c_0) are blocking edges to M^*, so $\beta_{x_j} = \beta_{y_j} = 1$ and similarly, $\beta_{p^c_0} = \beta_{q^c_0} = 1$ (this makes $\beta_{p^c_2} = \beta_{q^c_2} = -1$). So $\beta_{p^c_2} + \beta_{y_j} = 0$ while $\mathsf{wt}_{M^*}(p^c_2, y_j) = -2$. Thus this edge is slack and so it cannot be a popular edge.

We can similarly show that the edge (x_k, q^c_2) is not popular by considering the dominant matching that includes the edges (p^c_1, q^c_2) and (p^c_2, q^c_1). $\Diamond$

CLAIM 4. *No edge between a node in level 3 and a node in levels 1 or 2 is popular.*

Proof. Consider the edge (s^c_0, q^c_0). The dominant matching M^* includes the edges $(s^c_0, t^c_1), (s^c_1, t^c_0)$, and (p^c_2, q^c_0). So we have $\mathsf{wt}_{M^*}(s^c_0, q^c_0) = -2$ while $\beta_{s^c_0} = -1$ and $\beta_{q^c_0} = 1$, where $\vec{\beta}$ is a witness to M^*. Hence (s^c_0, q^c_0) is not a popular edge. It can similarly be shown for any edge $e \in \{(s^c_0, q^c_3), (p^c_4, t^c_0), (p^c_7, t^c_0)\}$ that e is not a popular edge.

Suppose the edge (s^c_0, u) for some $u \in \{y'_i : i \in [n_0]\} \cup \{q^{c_i}_2, q^{c_i}_5, q^{c_i}_8 : c_i \text{ is a clause}\}$ belongs to a popular matching M. In order to show a contradiction, consider the edge (s^c_0, t^c_1). We have $\mathsf{wt}_M(s^c_0, t^c_1) = 0$ since s^c_0 prefers t^c_1 to u while t^c_1 prefers its partner in M (this is s^c_1) to s^c_0. Since $\alpha_{s^c_0} = -1$, it has to be the case that $\alpha_{t^c_1} = 1$.

Since s^c_0 is matched to a node outside $\{t^c_1, t^c_2, t^c_3\}$, the node t^c_0 also has to be matched to a node outside $\{s^c_1, s^c_2, s^c_3\}$ — otherwise one of the 3 stable nodes t^c_1, t^c_2, t^c_3 would be left unmatched in M; however as M is popular, every stable node has to be matched in M. We have also seen earlier that neither (p^c_4, t^c_0) nor (p^c_7, t^c_0) is popular. Thus t^c_0 has to be matched to a neighbor worse than s^c_1.

Thus $\mathsf{wt}_M(s^c_1, t^c_0) = 0$ and so $\alpha_{s^c_1} = 1$. Since (s^c_1, t^c_1) is a popular edge, it follows from Lemma 2.2 that $\alpha_{s^c_1} + \alpha_{t^c_1} = \mathsf{wt}_M(s^c_1, t^c_1)$. However $\mathsf{wt}_M(s^c_1, t^c_1) = 0$ since $(s^c_1, t^c_1) \in M$ and we have just shown that $\alpha_{s^c_1} = \alpha_{t^c_1} = 1$. This is a contradiction and thus $(s^c_0, u) \notin M$. $\Diamond$

It is easy to see that the nodes z and z' are in the same connected component of F_G as the dominant matching M^* contains the edges (a_0, z) and (z', b_0) while any stable matching in G contains (a_0, b_0). We will now show that any popular matching that matches z and z' has to match these nodes to a_0 and b_0, respectively.

LEMMA 3.7. *If M is a popular matching in M that matches z and z' then $\{(a_0, z), (z', b_0)\} \subseteq M$.*

Proof. Suppose $(x_i, z) \in M$ for some $i \in [n_0]$. We know from the above claims that there is no popular edge between x_i's gadget and any neighbor in levels 0, 2, or 3. So $(x_i, z) \in M$ implies that $(z', y_i) \in M$ since all the 4 nodes x_i, y_i, x'_i, y'_i have to be matched in M and there is no other possibility of a popular edge incident to either y_i or y'_i. Hence $(x'_i, y'_i) \in M$. Thus (x_i, y'_i) and (x'_i, y_i) are blocking edges to M.

This means that $\alpha_{x_i} = \alpha_{y_i} = \alpha_{x'_i} = \alpha_{y'_i} = 1$. Note that (x'_i, y'_i) is a popular edge and so $\alpha_{x'_i} + \alpha_{y'_i} = \mathsf{wt}_M(x'_i, y'_i)$. However $\mathsf{wt}_M(x'_i, y'_i) = 0$ while $\alpha_{x'_i} = \alpha_{y'_i} = 1$. This is a contradiction and hence $(x_i, z) \notin M$ for any $i \in [n_0]$. We can similarly show that neither (p^c_{3j+1}, z) nor (z', q^c_{3j}) is in M, for any $j \in \{0, 1, 2\}$ and any clause c.

So if z and z' are matched in M then it has to be either with a_0, b_0 or with some level 0 neighbors. Observe that if $(a^c_{2i-1}, z) \in M$ then $(z', b^c_{2i-1}) \in M$ as the 4 nodes $a^c_{2i-1}, b^c_{2i-1}, a^c_{2i}, b^c_{2i}$ have to be matched in M and there is no other possibility of a popular edge incident to any of these 4 nodes (by Claim 2).

Suppose (a^c_{2i-1}, z) and (z', b^c_{2i-1}) are in M for some c and $i \in \{1, 2, 3\}$. Since z prefers a_0 to any level 0 neighbor, we have $\mathsf{wt}_M(a_0, z) = 0$. Similarly, $\mathsf{wt}_M(z', b_0) = 0$. Since $\mathsf{wt}_M(a^c_{2i-1}, b^c_{2i-1}) = 2$, this implies that $\alpha_{a^c_{2i-1}} = \alpha_{b^c_{2i-1}} = 1$. Hence, $\alpha_z = \alpha_{z'} = -1$ and we have $\alpha_{a_0} = \alpha_{b_0} = 1$. Note that $\alpha_{a_0} + \alpha_{b_0}$ has to be equal to $\mathsf{wt}_M(a_0, b_0)$ as (a_0, b_0) is a popular edge.

If a_0 is not matched to z (and so b_0 is not matched to z'), then $(a_0, b_0) \in M$. So $\mathsf{wt}_M(a_0, b_0) = 0$, however $\alpha_{a_0} + \alpha_{b_0} = 2$, a contradiction. Thus if M matches z and z' then $\{(a_0, z), (z', b_0)\} \subseteq M$.

Thus the above claims and Lemma 3.7 show that every level i gadget (for $i = 0, 1, 2, 3$) forms a distinct connected component in the graph F_G and the 4 nodes a_0, b_0, z, z' belong to their own connected component. This finishes the proof of Theorem 3.2.

4 Related hardness results

The construction from the previous section can be adapted to show further hardness results and also hardness of approximability results.

4.1 Popular matchings with forced / forbidden edges In the *popular matching problem in* $G = (A \cup B, E)$ *with forced/forbidden elements* (`pmffe`), our input also consists of some forced (resp., forbidden) edges E_1 (resp., E_0), and/or some forced (resp. forbidden) nodes U_1 (resp. U_0). The goal is to compute a popular matching M in G where all forced elements are included in M and no forbidden element is included in M. When $|E_1| = 1$ and $E_0 = U_1 = U_0 = \emptyset$, `pmffe` is the *popular edge problem*, which can be solved in polynomial time [6].

THEOREM 4.1. *The popular matching problem in* $G = (A \cup B, E)$ *with forced/forbidden element set* $\langle E_0, E_1, U_0, U_1 \rangle$ *is NP-hard when (i)* $|E_0| = 2$, *(ii)* $|E_1| = 2$, *(iii)* $|E_0| = |E_1| = 1$, *(iv)* $|U_0| = |U_1| = 1$, *(v)* $|U_0| = 1$ *and* $|E_0 \cup E_1| = 1$, *and (vi)* $|U_1| = 1$ *and* $|E_0 \cup E_1| = 1$.

Proof. The proof for $|U_0| = |U_1| = 1$ follows from our instance G in Section 3 with $U_0 = \{z\}$ and $U_1 = \{s_0^c\}$. It follows from the proof of Lemma 3.3 that the nodes s_0^c, t_0^c for all clauses c have to be matched in such a popular matching M. So M has to match all nodes in G except z and z'. Theorem 3.6 showed that finding such a popular matching M is NP-hard.

In order to show that the case with $|U_1| = 1$ and $|E_0 \cup E_1| = 1$ is also NP-hard, consider $U_1 = \{s_0^c\}$ and either $E_1 = \{(a_0, b_0)\}$ or $E_0 = \{(a_0, z)\}$. The forced node set forces the nodes s_0^c, t_0^c for all clauses c to be matched in our popular matching M while E_1 (similarly, E_0) forces z and z' to be unmatched in M.

In order to show the NP-hardness of the variant with $|E_0| = 2$ or $|E_1| = 2$, we will augment our instance G in Section 3 with an extra level 1 gadget X_0 (see the gadget in Fig. 1). Call the new instance G_0. The gadget X_0 has 4 nodes x_0, x_0', y_0, y_0' with the following preferences:

$$
\begin{aligned}
x_0 &: \; y_0 \succ y_0' \\
y_0 &: \; x_0 \succ x_0' \\
x_0' &: \; y_0 \succ y_0' \succ t_0^{c_1} \succ \cdots \succ t_0^{c_k} \\
y_0' &: \; x_0 \succ x_0' \succ s_0^{c_1} \succ \cdots \succ s_0^{c_k}
\end{aligned}
$$

Thus nodes in the gadget X_0 are not adjacent to z or z' or to any node in levels 0, 1, or 2 — however x_0' is adjacent to t_0^c for all clauses c and y_0' is adjacent to s_0^c for all clauses c. For each clause c, the node y_0' is at the bottom of s_0^c's preference list and the node x_0' is at the bottom of t_0^c's preference list.

A stable matching in the instance G_0 includes the edges (x_0, y_0) and (x_0', y_0') while there is a dominant matching in this instance with the edges (x_0, y_0') and (x_0', y_0). It is easy to extend Theorem 3.2 to show that the popular subgraph for the instance G_0 is the popular subgraph F_G along with an extra connected component with 4 nodes x_0, y_0, x_0', y_0'.

We now show that the problem is hard for the instance G_0 with $E_1 = \{(a_0, b_0), (x_0, y_0')\}$. Let M be a popular matching in G_0 that includes the edges (a_0, b_0) and (x_0, y_0'). Since M has to contain (a_0, b_0), it means that the nodes z and z' are unmatched in M.

Since $(x_0, y_0') \in M$, it means that $(x_0', y_0) \in M$. So (x_0, y_0) is a blocking edge to M and we have $\alpha_{x_0} = \alpha_{y_0} = 1$ and $\alpha_{x_0'} = \alpha_{y_0'} = -1$. This forces s_0^c, t_0^c for all c to be matched in M. The argument is the same as in Lemma 3.3 since the edges (s_0^c, y_0') and (x_0', t_0^c) would not be covered otherwise. This is because $\alpha_{x_0'} = \alpha_{y_0'} = -1$ and if s_0^c, t_0^c are unmatched then $\alpha_{s_0^c} = \alpha_{t_0^c} = 0$ and $\mathtt{wt}_M(s_0^c, y_0') = \mathtt{wt}_M(x_0', t_0^c) = 0$. This would make $\alpha_{s_0^c} + \alpha_{y_0'} < \mathtt{wt}_M(s_0^c, y_0')$ and similarly, $\alpha_{x_0'} + \alpha_{t_0^c} < \mathtt{wt}_M(x_0', t_0^c)$.

Thus M has to be a popular matching that matches all nodes in G_0 except z and z'. It is easy to see that the proof of Theorem 3.6 implies that finding such a popular matching in G_0 is NP-hard.

We can similarly show that the popular matching problem in G_0 with a given forbidden edge set E_0 is NP-hard for $|E_0| = 2$. For this, we will take $E_0 = \{(a_0, z), (x_0, y_0)\}$. This is equivalent to setting $E_1 = \{(a_0, b_0), (x_0, y_0')\}$.

We can similarly show that this problem with a given forced set E_1 and a given forbidden edge set E_0 is NP-hard for $|E_0| = |E_1| = 1$. For this, we will take $E_0 = \{(a_0, z)\}$ and $E_1 = \{(x_0, y_0')\}$. This will force s_0^c, t_0^c for all c to be matched in M while z, z' are unmatched in M.

Finally, the variant with $|U_0| = 1$ and $|E_0 \cup E_1| = 1$ follows by taking $U_0 = \{z\}$ and $E_0 = \{(x_0, y_0)\}$ or $E_1 = \{(x_0, y_0')\}$.

Some easy cases. Recall that when $|E_1| = 1$ and $E_0 = U_0 = U_1 = \emptyset$, `pmffe` reduces to the *popular edge problem*, that can be solved in polynomial time [6]. Now suppose $|E_0| = 1$ and $E_1 = U_0 = U_1 = \emptyset$: we show below that a polynomial-time algorithm for `pmffe` follows from the algorithm for the popular edge problem. Define E_s and $\overline{E}_s$ as:

$$
\begin{aligned}
E_s &= \{e \in E : \exists \text{ stable matching } M \text{ s.t. } e \in M\}, \\
\overline{E}_s &= \{e \in E : \exists \text{ stable matching } M \text{ s.t. } e \notin M\}.
\end{aligned}
$$

$E_d, \overline{E}_d$ (resp., $E_p, \overline{E}_p$) are defined similarly, by replacing "stable" with "dominant" (resp., popular). It was proved in [6] that $E_p = E_s \cup E_d$. We now argue that $\overline{E}_p = \overline{E}_s \cup \overline{E}_d$.

Consider any $e \in \overline{E}_s \cup \overline{E}_d$. Since there is a stable or dominant matching that does not contain e, it follows that $e \in \overline{E}_p$. We conclude that $\overline{E}_p \supseteq \overline{E}_s \cup \overline{E}_d$. We will now show that $\overline{E}_p \subseteq \overline{E}_s \cup \overline{E}_d$.

Consider any $e = (i,j) \in \overline{E}_p$. There is a popular matching M s.t. $e \notin M$, and i or j is matched (if i and j are unmatched then $(i,j) = (+,+)$, and M is not popular). Without loss of generality assume $(j,k) \in M$. It follows from $E_p = E_s \cup E_d$ that there exists a stable or dominant matching M' s.t. $(j,k) \in M'$, hence $(i,j) \notin M'$. Thus $\overline{E}_p \subseteq \overline{E}_s \cup \overline{E}_d$.

Since $\overline{E}_p = \overline{E}_s \cup \overline{E}_d$, we can solve the forbidden edge problem by checking if $e \in \overline{E}_s$ or $e \in \overline{E}_d$. For stable matchings, this can be done in polynomial time, see e.g. [12]. For dominant matchings, it immediately follows from results from [6].

4.2 Popular matching of maximum cost In this section, we are given, together with the usual bipartite graph and the rankings, a nonnegative edge cost vector $c \geq 0$. First, consider the problem of finding a popular matching of *minimum* cost wrt c (`min-wp`). Recall that `pmffe` with $|E_0| = 2$, $U_1 = U_0 = E_1 = \emptyset$ is NP-hard, as shown by the instance with graph G_0 and $E_0 = \{(a_0, z), (x_0, y_0)\}$ (see Section 4.1). Let the costs of (a_0, z) and (x_0, y_0) be equal to 1, and let all other costs be 0. We can conclude the following.

Corollary 4.1. *`min-wp` is NP-Hard.*

Moreover, since (unless P=NP) we cannot distinguish efficiently instances of `min-wp` that have optimum value 0 from those that have value strictly greater than 0, `min-wp` cannot be approximated to any factor. Below, we denote the problem of finding a popular matching in a bipartite graph of *maximum* cost wrt a non-negative edge cost function by `max-wp`.

Theorem 4.2. *Unless P=NP, `max-wp` cannot be approximated in polynomial time to a factor better than $\frac{1}{2}$. On the other hand, there is a polynomial-time algorithm that computes a $\frac{1}{2}$-approximation to `max-wp`.*

Proof. Consider the instance G_0 again. The hardness of approximating `max-wp` to a factor better than $\frac{1}{2}$ immediately follows from Theorem 4.1 with $|E_1| = 2$ by setting the costs of edges in E_1 to 1 and all other edge costs to 0.

We will now show that a popular matching in G of cost at least $c(M^*)/2$ can be computed in polynomial time, where M^* is a max-cost popular matching in G. It was shown in [6] that any popular matching M in $G = (A \cup B, E)$ can be partitioned into $M_0 \cup M_1$ such that $M_0 \subseteq S$ and $M_1 \subseteq D$, where S is a stable matching and D is a dominant matching in G.

Consider the following algorithm.

1. Compute a max-cost stable matching S^* in G.
2. Compute a max-cost dominant matching D^* in G.
3. Return the matching in $\{S^*, D^*\}$ with larger cost.

Since all edge costs are non-negative, either the max-cost stable matching in G or the max-cost dominant matching in G has cost at least $c(M^*)/2$. Thus Steps 1-3 compute a $\frac{1}{2}$-approximation for max-cost popular matching in $G = (A \cup B, E)$.

Regarding the implementation of this algorithm, both S^* and D^* can be computed in polynomial time [6]. Thus our algorithm runs in polynomial time.

4.3 Popular and Dominant matchings in non-bipartite graphs In this section, we consider the popular (also, dominant) matching problem in general graphs. The only modification to the input with respect to the previous paragraphs, is that our input graph G is not required to be bipartite. As usual, together with G, we are given a collection of rankings, one per node of G, with each node ranking its neighbors in a strict order of preference. The goal is to decide if G admits a popular (resp., dominant) matching. In this section, we use an extension to the non-bipartite case of the dual certificates from Section 2.1, as well as combinatorial certificates for popular matchings. We start by presenting those tools.

4.3.1 Dual certificates in non-bipartite graphs Our next theorem shows that the sufficient condition in Theorem 2.1 also certifies popularity in non-bipartite graphs.

Theorem 4.3. *Let M be any matching in $G = (V, E)$. The matching M is popular if there exists $\vec{\alpha} \in \mathbb{R}^{|V|}$ such that $\sum_{u \in V} \alpha_u = 0$ and*

$$\begin{aligned} \alpha_u + \alpha_v \quad &\geq \quad \mathsf{wt}_M(u,v) \quad \forall\, (u,v) \in E \\ \alpha_u \quad &\geq \quad \mathsf{wt}_M(u,u) \quad \forall\, u \in V. \end{aligned}$$

The proof of Theorem 4.3 follows by considering the max-weight perfect fractional matching LP in the graph $\tilde{G}$ with edge weight function wt_M as the primal LP. It is easy to see that if there exists a vector $\vec{\alpha} \in \mathbb{R}^{|V|}$ as given above then the optimal value of the dual LP is at most 0, equivalently, $\mathsf{wt}_M(\tilde{N}) \leq 0$ for all matchings N in G, i.e., M is a popular matching. If M is a popular matching that admits $\vec{\alpha} \in \mathbb{R}^{|V|}$ satisfying the conditions in Theorem 4.3, we will call $\vec{\alpha}$ a *witness* of M.

Note that any stable matching in G has $\vec{0}$ as a witness. The witness of the matching M =

$\{(d_0, d_1), (d_2, d_3)\}$ in the instance described in Fig. 4 on the nodes d_0, d_1, d_2, d_3 is $\vec{\alpha}$ where $\alpha_{d_1} = \alpha_{d_3} = 1$ and $\alpha_{d_0} = \alpha_{d_2} = -1$.

4.3.2 Combinatorial characterization of popular matchings Fix a matching M of G. We refer e.g. to [23] for the standard concepts of M-*exposed* and M-*covered* nodes, M-*alternating* paths and cycles, and M-*augmenting* paths. The graph G_M is defined as the subgraph of G obtained by deleting edges that are labeled $(-,-)$, and by attaching to other edges not in M the appropriate labels as defined in Section 2.1. Observe that M is also a matching of G_M, hence definitions of M-alternating path and cycle apply in G_M as well. These definitions can be used to obtain a characterization of popular matchings in terms of forbidden substructures of G_M, as shown in [13].

THEOREM 4.4. *A matching M of G is popular if and only if G_M does not contain any of the following:*

(i) an M-alternating cycle with a $(+,+)$ edge.

(ii) an M-alternating path that starts and ends with two distinct $(+,+)$ edges.

(iii) an M-alternating path that starts from an M-exposed node and ends with a $(+,+)$ edge.

4.3.3 The reductions Theorem 3.6 showed that that the problem of deciding if a bipartite instance $G = (A \cup B, E)$ admits a popular matching that matches *exclusively* a given set $S \subset A \cup B$ is NP-hard (i.e., the nodes outside S have to be left unmatched). Let us call this the *exclusive popular set* problem. We will now use the hardness of the exclusive popular set problem in the instance G from Section 3 to show that the dominant matching problem in non-bipartite graphs is NP-hard. In order to show this, *merge* the nodes z and z' in the instance G from Section 3 into a single node z. Call the new graph G'. The preference list of the node z in G' is all its level 1 neighbors in some order of preference, followed by all its level 2 neighbors in some order of preference, followed by a_0, b_0, and then level 0 neighbors. The order among level i neighbors (for $i = 0, 1, 2$) in this list does not matter.

LEMMA 4.1. *A popular matching N in G' is dominant if and only if N matches all nodes in G' except z.*

Proof. Let N be any popular matching in G'. Any popular matching has to match all stable nodes in G' [13], thus N matches all stable nodes in G'. Suppose some unstable node other than z (say, s_0^c) is left unmatched in N. We claim that t_0^c also has to be left unmatched in N. Indeed, if N is a matching that leaves s_0^c unmatched but matches t_0^c, then there is an N-alternating path in G_N with s_0^c as an endpoint that ends with a $(+,+)$ edge. This is a contradiction to the popularity of N (by Theorem 4.4, part (iii)).

Since s_1^c and t_1^c have no other neighbors, the edge $(s_1^c, t_1^c) \in N$ and so there is an augmenting path $\rho = s_0^c$-t_1^c-s_1^c-t_0^c with respect to N. Observe that N is *not* more popular than $N \oplus \rho$, a larger matching. Thus N is not a dominant matching in G'.

Conversely, suppose N is a popular matching in G' that matches all nodes except z. Then there is no larger matching than N in G', thus N is a dominant matching in G'.

Thus a dominant matching exists in G' if and only if there is a popular matching in G' that matches all nodes except z. This is equivalent to deciding if there exists a popular matching in G that matches all nodes in G except z and z'. This concludes the proof of the following theorem.

THEOREM 4.5. *Given a (non-bipartite) graph $G = (V, E)$ with preference lists, the problem of deciding if G admits a dominant matching or not is NP-hard. Moreover, this hardness holds even when G admits a stable matching.*

We will now show the following.

THEOREM 4.6. *Given a (non-bipartite) graph $G = (V, E)$ with strict preference lists, the problem of deciding if G admits a popular matching or not is NP-hard.*

For this, we will augment the graph G' with the gadget D given in Fig. 4. Call the new graph H.

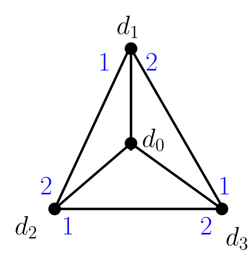

Figure 4: Each of d_1, d_2, d_3 is a top choice neighbor for another node here and d_0 is the last choice of d_1, d_2, d_3.

The gadget D. There will be 4 nodes d_0, d_1, d_2, d_3 that form the gadget D (see Fig. 4). The preferences of nodes in D are given below.

$d_1\colon\ d_2 \succ d_3 \succ d_0$ $\qquad$ $d_2\colon\ d_3 \succ d_1 \succ d_0$

$d_3\colon\ d_1 \succ d_2 \succ d_0$ $\qquad$ $d_0\colon\ d_1 \succ d_2 \succ d_3 \succ \cdots$

The node d_0 will be adjacent to all nodes in H, except z. The neighbors of d_0 that are not in D are in the "$\cdots$" part of d_0's preference list and the order among these nodes does not matter. The node d_0 will be at the bottom of preference lists of all its neighbors.

LEMMA 4.2. *For any popular matching M in H, the following properties hold:*

(1) either $\{(d_0, d_1), (d_2, d_3)\}$ or $\{(d_0, d_2), (d_1, d_3)\}$ is a subset of M.

(2) M matches all nodes in H except z.

Proof. Since each of d_1, d_2, d_3 is a top choice neighbor for some node in H, a popular matching in H cannot leave any of these 3 nodes unmatched. Since these 3 nodes have no neighbors outside themselves other than d_0, a popular matching has to match d_0 to one of d_1, d_2, d_3. Thus d_0, d_1, d_2, d_3 are matched among themselves in M. The only possibilities for M when restricted to d_0, d_1, d_2, d_3 are the pair of edges $(d_0, d_1), (d_2, d_3)$ and $(d_0, d_2), (d_1, d_3)$. The third possibility $(d_0, d_3), (d_1, d_2)$ is "less popular than" $(d_0, d_1), (d_2, d_3)$ as d_0, d_2, and d_3 prefer the latter to the former. This proves part (1) of the lemma.

Consider any node $v \neq z$ outside the gadget D. If v is left unmatched in M then we either have an alternating path $\rho_1 = (v, d_0)$-(d_0, d_1)-(d_1, d_3) or an alternating path $\rho_2 = (v, d_0)$-(d_0, d_2)-(d_2, d_1) with respect to M: the middle edge in each of these alternating paths belongs to M and the third edge is a *blocking edge* with respect to M. Both ρ_1 and ρ_2 are M-alternating paths in G_M that start from an M-exposed node and end with a $(+, +)$ edge — this is a forbidden structure for a popular matching (see Theorem 4.4, case (iii)). Hence every node $v \neq z$ in H has to be matched in M. This proves part (2).

Since the total number of nodes in H is odd, at least 1 node has to be left unmatched in any matching in H. Lemma 4.2 shows that the node z will be left unmatched in any popular matching in H. For any popular matching M in H, the matching M restricted to G' (recall that G' is $H \setminus D$) has to be popular on G', otherwise it would contradict the popularity of M in H. We will now show the following converse of Lemma 4.2.

LEMMA 4.3. *If G' admits a popular matching that matches all its nodes except z then H admits a popular matching.*

Proof. Let N be a popular matching in G' that matches all its nodes except z. Let G'_N be the subgraph obtained by removing all edges labeled $(-, -)$ with respect to N from G'. Since N is popular in G', it satisfies the necessary and sufficient conditions for popularity given in Theorem 4.4.

We claim $M = N \cup \{(d_0, d_1), (d_2, d_3)\}$ is a popular matching in H. We will now show that M obeys those conditions in the subgraph H_M obtained by deleting edges labeled $(-, -)$ with respect to M. There is no edge in H_M between D and any node in G' since every edge in H between D and a node in G' is $(-, -)$. This is because for any such edge (d_0, v), the node d_0 prefers d_1 (its partner in M) to v and similarly, v prefers each of its neighbors in G' to d_0. Since $v \neq z$, we know that N (and thus M) matches v to one of its neighbors in G'. Thus the graph H_M is the graph G'_N along with some edges within the gadget D.

It is easy to check that $\{(d_0, d_1), (d_2, d_3)\}$ satisfies the 3 conditions from Theorem 4.4 in the subgraph of D obtained by pruning $(-, -)$ edges. We know that N satisfies the 3 conditions from Theorem 4.4 in G'_N. Thus M satisfies all these 3 conditions in H_M. Hence M is popular in H.

Thus we have shown that H admits a popular matching if and only if G' admits a matching that matches all nodes except z. Since the latter problem is NP-hard, so is the former problem. This concludes the proof of Theorem 4.6.

5 Strongly dominant matchings

Given the hardness of finding dominant matchings in all graphs (see Theorem 4.5), our goal now is to identify a subclass of dominant matchings that is tractable in all graphs. Dominant matchings in bipartite graphs were first seen in [13] and they were defined using Definition 5 given below. The goal in [13] was to compute a max-size popular matching in a bipartite graph and it was observed that it was enough to construct a matching that fulfilled Definition 5. It was shown in [17] that if M satisfies Definition 5 then M satisfies Definition 2, i.e., such a matching M is dominant.

DEFINITION 5. *A matching M is strongly dominant in $G = (V, E)$ if there is a partition (L, R) of the node set V such that (i) $M \subseteq L \times R$, (ii) M matches all nodes in R, (iii) every $(+, +)$ edge is in $R \times R$, and (iv) every edge in $L \times L$ is $(-, -)$.*

Consider the complete graph on 4 nodes d_0, d_1, d_2, d_3 where d_0's preference list is $d_1 \succ d_2 \succ d_3$, d_1's preference list is $d_2 \succ d_3 \succ d_0$, while d_2's preference list is $d_3 \succ d_1 \succ d_0$, and d_3's preference list is $d_1 \succ d_2 \succ d_0$. This instance (see Fig. 4) has no stable matching. The matching $M = \{(d_0, d_1), (d_2, d_3)\}$ is a strongly dominant matching here with $L = \{d_0, d_2\}$ and

$R = \{d_1, d_3\}$. $M \subseteq L \times R$ and it is a perfect matching. Moreover, the edge $(d_0, d_2) \in L \times L$ is $(-,-)$ and there is only one $(+,+)$ edge here, which is $(d_1, d_3) \in R \times R$.

In bipartite graphs, every dominant matching is strongly dominant [6]. However this is not so in non-bipartite graphs, as shown by the next example. Consider the following graph on 4 nodes a, b, c, d where a's preference list is $b \succ c \succ d$, while b's preference list is $a \succ c$ and c's preference list is $a \succ b$ and d's only neighbor is a. It is simple to check that the matching $\{(a, d), (b, c)\}$ is popular; moreover it is a perfect matching and hence it is dominant. However it is *not* strongly dominant as both (a, b) and (a, c) are $(+,+)$ edges and one of b, c has to be in L. Also note that this matching does not admit any witness as given in Theorem 4.3.

A new instance. We now show a surprisingly simple reduction from the strongly dominant matching problem in $G = (V, E)$ to the stable matching problem in a new graph $G' = (V, E')$. So Irving's algorithm [15], which efficiently solves the stable matching problem in all graphs, when run in the new instance G', solves our problem. Thus our result generalizes the close relationship between strongly dominant matchings and stable matchings shown in [6] for bipartite graphs to all graphs.

The graph G' can be visualized as the bidirected graph corresponding to G. The node set of G' is the same as that of G. For every $(u, v) \in E$, there will be 2 edges in G' between u and v: one directed from u to v which will be denoted by (u^+, v^-) or (v^-, u^+) and the other directed from v to u which will be denoted by (u^-, v^+) or (v^+, u^-).

For any $u \in V$, if u's preference list in G is $v_1 \succ v_2 \succ \cdots \succ v_k$ then u's preference list in G' is $v_1^- \succ v_2^- \succ \cdots \succ v_k^- \succ v_1^+ \succ v_2^+ \succ \cdots \succ v_k^+$. The neighbor v_i^+ corresponds to the edge (u^-, v_i^+) and the neighbor v_i^- corresponds to the edge (u^+, v_i^-). Thus u prefers *outgoing* edges to *incoming* edges: among outgoing edges (similarly, incoming edges), its order is its original preference order.

- A matching M' in G' is a subset of E' such that for each $u \in V$, M' contains at most one edge incident to u, i.e., at most one edge in $\{(u^+, v^-), (u^-, v^+) : v \in \mathsf{Nbr}(u)\}$ is in M', where $\mathsf{Nbr}(u)$ is the set of u's neighbors in G.

- For any matching M' in G', define the *projection* of M' to be the following matching M of G:

$$M = \{(u, v) : (u^+, v^-) \text{ or } (u^-, v^+) \text{ is in } M'\}.$$

Definition 6. *A matching M' is stable in G' if for every edge $(u^+, v^-) \in E' \setminus M'$: either (i) u is matched in M' to a neighbor ranked better than v^- or (ii) v is matched in M' to a neighbor ranked better than u^+.*

We now present our algorithm to find a strongly dominant matching in $G = (V, E)$.

1. Build the bidirected graph $G' = (V, E')$.
2. Run Irving's stable matching algorithm in G'.
3. If a stable matching M' is found in G' then return the projection M of M'.

   Else return "G has no strongly dominant matching".

Note that running Irving's algorithm in the bidirected graph G' is exactly the same as running Irving's algorithm in the simple undirected graph H that has *three* copies of each node $u \in V$: these are u^+, u^-, and $d(u)$. Corresponding to each edge (u, v) in G, there will be the two edges (u^+, v^-) and (u^-, v^+) in H and for each $u \in V$, the graph H also has the edges $(u^+, d(u))$ and $(u^-, d(u))$. If u's preference list in G is $v_1 \succ v_2 \succ \cdots \succ v_k$ then u^+'s preference list in H will be $v_1^- \succ v_2^- \succ \cdots \succ v_k^- \succ d(u)$ and u^-'s preference list will be $d(u) \succ v_1^+ \succ v_2^+ \succ \cdots \succ v_k^+$. The preference list of $d(u)$ will be $u^+ \succ u^-$. Thus in any stable matching in H, one of u^+, u^- has to be matched to $d(u)$.

Correctness of our algorithm. The correctness of our strongly dominant matching algorithm is based on the following characterization of strongly dominant matchings in terms of their witnesses, whose proof is deferred to the journal version of the paper.

Theorem 5.1. *A matching M is strongly dominant in G if and only if there exists $\vec{\alpha}$ that satisfies Theorem 4.3 such that $\alpha_u = \pm 1$ for all u matched in M and $\alpha_u = 0$ for all u unmatched in M.*

The following two lemmas conclude the proof of correctness of our algorithm.

Lemma 5.1. *If M' is a stable matching in G' then the projection of M' is a strongly dominant matching in G.*

Sketch of the proof. Let M be the projection of M'. In order to show that M is a strongly dominant matching in G, we will construct a witness $\vec{\alpha}$ as given in Theorem 5.1. Set $\alpha_u = 0$ for all nodes u left unmatched in M. For each node u matched in M, if $(u^+, *) \in M'$ then set $\alpha_u = 1$; else set $\alpha_u = -1$. Note that $\sum_{u \in V} \alpha_u = 0$ since for each edge $(a, b) \in M$ (so either (a^+, b^-) or (a^-, b^+) is in M'), we have $\alpha_a + \alpha_b = 0$ and for each node u that is unmatched in M, we have $\alpha_u = 0$. Also, $\alpha_u \geq \mathsf{wt}_M(u, u)$ for all $u \in V$ since

(i) $\alpha_u = 0 = \mathtt{wt}_M(u,u)$ for all u left unmatched in M and (ii) $\alpha_u \geq -1 = \mathtt{wt}_M(u,u)$ for all u matched in M.

We defer the proof that for every $(a,b) \in E$, $\alpha_a + \alpha_b \geq \mathtt{wt}_M(a,b)$ to the journal version of the paper. Once this is proved, since $\vec{\alpha}$ satisfies the conditions in Theorem 5.1, we can conclude that M is a strongly dominant matching in G.

LEMMA 5.2. *If G admits a strongly dominant matching then G' admits a stable matching.*

Sketch of the proof. Let M be a strongly dominant matching in $G = (V,E)$. Let $\vec{\alpha}$ be a witness of M as given in Theorem 5.1. That is, $\alpha_u = 0$ for u unmatched in M and $\alpha_u = \pm 1$ for u matched in M. As done in the proof of Theorem 5.1, we can interpret $(\tilde{M}, \vec{\alpha})$ as a pair of optimal primal and dual solutions. Hence, for each $(u,v) \in M$, $\alpha_u + \alpha_v = \mathtt{wt}_M(u,v) = 0$ by complementary slackness on the dual LP, so (α_u, α_v) is either $(1,-1)$ or $(-1,1)$. Construct a stable matching M' in G' as follows. For each $(u,v) \in M$, if $(\alpha_u, \alpha_v) = (1,-1)$ then add (u^+, v^-) to M'; else add (u^-, v^+) to M'. The statement follows by showing that no edge in $E' \setminus M'$ blocks M'. The proof of the latter fact is postponed to the journal version of the paper.

Since Irving's algorithm runs in linear time, we can conclude the following theorem.

THEOREM 5.2. *There is a linear time algorithm to determine if a graph $G = (V,E)$ admits a strongly dominant matching or not and if so, to return one.*

6 Popular matchings of minimum cost in bounded treewidth graphs

In this section, we give an high-level view of the proof of the following theorem. Full details are deferred to the journal version of the paper.

THEOREM 6.1. *There exists a function $f(n,\omega) = O(n^{3\omega+7})$ such that, given as input a graph G with n nodes with strict preference lists, treewidth ω, and a cost vector c on the edges of G, in time $f(n,\omega)$ we can either conclude that G has no popular matching, or find a popular matching of G of minimum cost.*

6.1 Locally popular matchings We assume that the reader is familiar with basic graph theory terminology, tree decompositions and related concepts (see e.g. [3]). We fix an input graph G and its preference lists, and assume that no two matchings of G have the same cost. This can be achieved efficiently by standard perturbation techniques. Bounded treewidth is a classical assumption that often turns intractable problems into tractable ones. A typical example is Maximum Independent (Stable) Set (`MIS`), for which a polynomial-time algorithm exists in bounded treewidth graphs [3]. `MIS` enjoys two nice properties. The first is *monotonicity*: if S is an independent set in a graph G and G' is a subgraph of G, then the solution induced by S on G' is also feasible. The second is *locality*: S is an independent set if and only if, for each node of S, any node of its neighborhood does not belong to S.

Similar properties do not hold for popular matchings. Indeed, popularity is not a local condition, since it may depend on how nodes far away in the graph are matched, see Theorem 4.4. Moreover, if we take a graph G and a popular matching M, the subset of M contained in an induced subgraph of G need not be popular (see Fig. 5, bottom graph).

We overcome those problems by investigating the class of (S,X)-locally popular matchings, where S is a vertex separator of G and X is a connected component of $G \setminus S$ (possibly $S = \emptyset$ and $X = V$). Let M be a matching of G and $U \subseteq V$. If $e \cap U \neq \emptyset$ for all $e \in M$, we say that M is a *U-matching*. We say that a $(X \cup S)$-matching M of G is *(S,X)-locally popular* if none of the structures (i), (ii), and (iii) from Theorem 4.4 is a subgraph of $G_M[X \cup S]$.

We remark that, in the graph $G_M[X \cup S]$, for a node $v \in S$ matched in M to a neighbor outside $X \cup S$, the labels of edges of $G_M[X \cup S]$ incident to v are a function of the edge $(v, M(v))$ (however note that the edge $(v, M(v))$ is not in $G[X \cup S]$). In particular, a node that is not matched in $G_M[X \cup S]$ may still be M-covered (hence not M-exposed). If this happens, such a node cannot be the M-exposed node in the path (iii) from Theorem 4.4. See Fig. 5 for an example. From the definition, it follows that the condition of being locally popular only depends on $G[X \cup S \cup N(S)]$.

The following easy lemma shows a certain monotonicity property of (S,X)-locally popular matchings, and that popular matchings arise as a special case of locally popular. In particular, if M is a popular matching, for S, X as above we have that $M_{X \cup S}$ is (S,X)-locally popular, where for a matching M of G and $U \subseteq V$, we let $M_U := \{e \in M : e \cap U \neq \emptyset\}$.

LEMMA 6.1. *Let S, X be as above, S' be a vertex separator of G, and X' be one of the connected components of $G \setminus S'$ such that $X' \cup S' \supseteq X \cup S$. Then:*

1. *Let M be a matching in G. Then M is $(\emptyset, V)$-locally popular if and only if it is popular.*

2. *Let M' be a (S',X')-locally popular matching. Then $M'_{X \cup S}$ is an (S,X)-locally popular matching.*

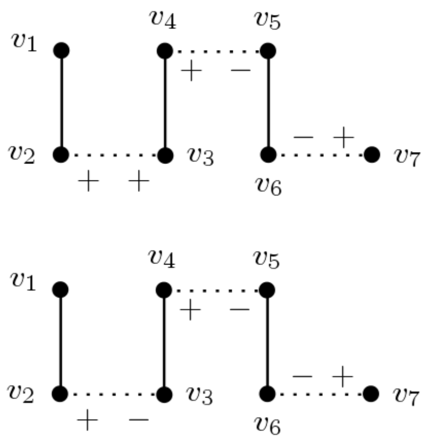

Figure 5: An example of popular vs locally-popular. In each graph, let M be the matching given by bold edges. *Top graph.* M is not popular, since there is an M-alternating path in G_M with a $(+,+)$ edge and an unmatched node. But if we take $X = \{v_1, v_2\}$ and $S = \{v_3, v_4\}$, matching $M_{X\cup S}$ is (S,X)-locally popular. *Bottom graph.* M is popular, hence it is (X,S)-locally popular for every choice of S and X. But if we restrict M to the subgraph G' of G induced by v_1, v_2, v_3, v_4, v_5, the matching we obtain is not popular in G', since there is a $(+,+)$ edge incident to the now unmatched node v_5.

6.2 Tipping points (S,X)-locally popular matchings can be exponentially many, even for fixed S, X. The following lemma shows how to partition (S,X)-locally popular matchings for each S and X into a "small" number of sets with the following property: for each of those sets $\mathcal{T}$, in order to compute the popular matching of G of minimum cost, we only need to store one matching.

LEMMA 6.2. *We can partition, for each vertex separator S and connected component X of $G \setminus S$, the set of (S,X)-locally popular matchings into a family $(\mathcal{T}_1^{S,X}, \ldots, \mathcal{T}_{k^{S,X}}^{S,X})$ with the following properties:*

1. *$k^{S,X} \leq g(|S|)|V|^{|S|}$ for some universal function $g : \mathbb{N} \to \mathbb{N}$.*

2. *For $i = 1, \ldots, k^{S,X}$, call the matching of minimum cost from $\mathcal{T}_i^{S,X}$ the $\mathcal{T}_i^{S,X}$-leader[2]. Consider vertex separators S', S'' of G and connected component X' (resp. X'') of $G \setminus S'$ (resp. $G \setminus S''$) with $X'' \supseteq X'$ and $S'' \cup X'' \supseteq S' \cup X'$ (possibly $S'' = \emptyset$ and $X'' = V$). Assume that M' is the $\in \mathcal{T}_j^{S'',X''}$-leader, for some j. Let $M := M'_{X'\cup S'}$. By Lemma 6.1, part 2, M is (S',X')-locally popular, hence $M \in \mathcal{T}_i^{S',X'}$ for some i. Then M is the $\mathcal{T}_i^{S',X'}$-leader.*

[2]This is uniquely defined, because of the initial perturbation of costs.

Call sets $\mathcal{T}_1^{S,X}, \ldots, \mathcal{T}_{k^{S,X}}^{S,X}$ as in the previous lemma *tipping points* for (S,X), and, for $M \in \mathcal{T} \in \{\mathcal{T}_1^{S,X}, \ldots, \mathcal{T}_{k^{S,X}}^{S,X}\}$, we say that $\mathcal{T}$ is the tipping point of M. It follows from the definition of $\mathcal{T}$-leader and Lemma 6.1, part 1 that the popular matching of minimum cost is the $\mathcal{T}$-leader of minimum cost, where $\mathcal{T}$ ranges over all tipping points for $(\emptyset, V)$. We now present Algorithm 1 and Algorithm 2, that show how to iteratively obtain the $\mathcal{T}$-leaders by building on a tree decomposition of G. The bound in Theorem 6.1 follows from an analysis of the running times of those algorithms.

6.3 Tree decompositions and the algorithm Denote a tree decomposition of G by $(T, \mathcal{B})$ where T is a tree and $\mathcal{B} = \{B(i) : i \in V(T)\}$ are the bags. Assume wlog that, for each pair of bags B, B' adjacent in T, $B \cap B'$ is a vertex separator of G. Pick an arbitrary vertex of T as the root and orient the remaining edges towards the root, as to obtain a *directed tree decomposition.* In the directed tree, each non-root node B has exactly one *successor* $S(B)$, i.e., there exists exactly one node $S(B)$ such that $(B, S(B))$ is an arc of the directed tree decomposition. If B is the root, we set $S(B) = \emptyset$. We say that B is a *predecessor* of $S(B)$. We assume wlog that $(T, \mathcal{B})$ is *dichotomic*, i.e. each bag has at most two predecessors. For a bag B different from the root, let T_B be the unique subtree of T containing B and all its predecessors, but not containing $S(B)$. When B is the root, set $T_B = T$. Note that, for each $B \in \mathcal{B}$, the removal of B from T_B partitions the latter in at most two subtrees T_{B_1}, T_{B_2} of T, where B_1 (and possibly B_2) is a predecessor of B. We also let $V(T_B)$ the nodes of G contained in the union of bags of T_B.

For each bag B, Algorithm 1 recursively computes and stores in $\mathcal{L}_B$ $\mathcal{T}$-leaders for all tipping points $\mathcal{T}$ for (S,X), with $S = B \cap S(B)$ and $X = V(T_B) \setminus S$, starting from leaves of T. Note that $|S| \leq \omega$. By Lemma 6.1, part 1, when B is the root, the matching from $\mathcal{L}_B$ of minimum cost is the popular matching of minimum cost, see Step 21. When B is a leaf of T, the bounded treewidth assumption implies that the leaders can be computed by complete enumeration, see steps 4–7. The function `Update`, given in Algorithm 2, guarantees that at each time we keep stored at most one matching for each (S,X)-tipping point. If conversely B is not a leaf, then Lemma 6.2, part 2, guarantees that $\mathcal{L}_B$ can be constructed from set(s) $\mathcal{L}_{B'}$ for all predecessors B' of B (recall that they are at most 2), see steps 10–17. Also, if we deduce $\mathcal{L}_B = \emptyset$ for some B, Lemma 6.1 guarantees that G has no popular matching, see Step 18.

Algorithm 1

Require: A graph G, together with, for each node $v \in V$, a ranking of the neighbors of v. A dichotomic directed tree decomposition $(T, \mathcal{B})$ of G.

1: For all $B \in \mathcal{B}$, label B as *unflagged*.
2: Choose an unflagged bag B whose predecessors are flagged, and flag B.
3: Set $\mathcal{L}_B = \emptyset$, $S = B \cap S(B)$, $X = V(T_B) \setminus S$.
4: **if** B has no predecessor in T **then**
5:   **for** all B-matchings M^* of G **do**
6:     **if** M^* is an (S, X)-locally popular matching **then** `Update`(M^*, $\mathcal{L}_B$).
7:     **end if**
8:   **end for**
9: **else**
10:   Let $S_1 = B \cap B_1$ and (possibly) $S_2 = B \cap B_2$, where B_1 and (poss.) B_2 are predecessors of B.
11:   **for** all B-matchings M of G, all $M_1 \in \mathcal{L}_{B_1}$ and (possibly) $M_2 \in \mathcal{L}_{B_2}$ **do**
12:     **if** $(M_1)_{S_1} = M_{S_1}$ and (possibly) $(M_2)_{S_2} = M_{S_2}$ **then**
13:       Let $M^* = M \cup M_1 \cup M_2$.
14:       **if** M^* is an (S, X)-locally popular matching of G **then** `Update`(M^*, $\mathcal{L}_B$).
15:     **end if**
16:   **end for**
17: **end if**
18: **if** $\mathcal{L}_B = \emptyset$ **output** G has no popular matching.
19: **else if** $\exists B \in \mathcal{B}$ unflagged, **then** go to Step 2.
20: **end if**
21: Let B be the head of T. Output the matching of minimum cost from $\mathcal{L}_B$.

Algorithm 2 `Update`

Require: M^*, $\mathcal{L}_B$.
Let $\mathcal{T}$ be the (S, X)-tipping point of M^*.
if there exists $M' \in \mathcal{L}_B$ whose tipping point is $\mathcal{T}$ **then**
  if $c(M^*) < c(M')$ **then**
    Set $\mathcal{L}_B = \mathcal{L}_B \setminus \{M'\} \cup \{M^*\}$.
  end if
  else Set $\mathcal{L}_B = \mathcal{L}_B \cup \{M^*\}$.
end if

References

[1] D.J. Abraham, R.W. Irving, T. Kavitha, and K. Mehlhorn. *Popular matchings.* SIAM Journal on Computing, 37(4): 1030–1045, 2007.

[2] P. Biró, R. W. Irving, and D. Manlove. *Popular Matchings in the Marriage and Roommates Problems.* Proceedings of CIAC '10: 97–108, 2010.

[3] H. Bodlaender. *A tourist guide through treewidth.* Acta Cybernetica, 11, 1993.

[4] `https://en.wikipedia.org/wiki/Condorcet_method`

[5] A. Cseh. *Popular matchings.* In *Trends in computational social choice.* AI Access: 105–122, 2017.

[6] Á. Cseh and T. Kavitha. *Popular edges and dominant matchings.* Mathematical Programming, 172(1): 209–229, 2018.

[7] Á. Cseh and T. Kavitha. *Popular matchings in complete graphs.* To appear in FSTTCS 2018.

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

[9] D. Gale and M. Sotomayor. *Some remarks on the stable matching problem.* Discrete Applied Mathematics, 11: 223-232, 1985.

[10] P. Gärdenfors. *Match making: assignments based on bilateral preferences.* Behavioural Sciences, 20(3): 166–173, 1975.

[11] S. Gupta, P. Misra, S. Saurabh, and M. Zehavi. *Popular Matching in Roommates Setting is NP-hard.* Proceedings of SODA 2019.

[12] D. Gusfield and R. W. Irving. *The Stable marriage problem - structure and algorithms.* MIT Press, 1989.

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

[14] C.-C. Huang and T. Kavitha. *Popularity, mixed matchings, and self-duality.* Proceedings of SODA '17: 2294–2310, 2017.

[15] R. W. Irving. *An efficient algorithm for the stable roommates problem.* Journal of Algorithms, 6: 577–595, 1985.

[16] T. Kavitha, J. Mestre, and M.Nasre. *Popular mixed matchings.* Theoretical Computer Science, 412(24): 2679–2690, 2011.

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

[18] T. Kavitha. *Popular half-integral matchings.* Proceedings of ICALP '16: 22.1-22.13, 2016.

[19] T. Kavitha. *Max-size popular matchings and extensions.* `http://arxiv.org/abs/1802.07440`

[20] T. Kavitha. *Popular matchings of desired size.* Proceedings of WG '18: 306-317, 2018.

[21] R. Klasing, A. Kosowski, and A. Navarra. *Cost minimization in wireless networks with a bounded and unbounded number of interfaces.* Networks, 53 (3): 266–275, 2009.

[22] B. M. Maggs and R. K. Sitaraman. *Algorithmic Nuggets in Content Delivery.* Computer Communication Review, 45 (3): 52–66, 2015.

[23] D. F. Manlove. *Algorithmics of Matching Under Preferences.* World Scientific Publishing Company, 2004.

[24] T. J. Schaefer. The complexity of satisfiability problems. Proceedings of STOC '78: 216–226, 1978.