ECONSTOR
Make Your Publications Visible.

A Service of
ZBW
Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Cseh, Ágnes; Faenza, Yuri; Kavitha, Telikepalli; Powers, Vladlena

Working Paper

Understanding popular matchings via stable matchings

CERS-IE Working Papers, No. CERS-IE WP - 2020/3

Provided in Cooperation with:
Institute of Economics, Centre for Economic and Regional Studies, Hungarian Academy of Sciences

Suggested Citation: Cseh, Ágnes; Faenza, Yuri; Kavitha, Telikepalli; Powers, Vladlena (2020) : Understanding popular matchings via stable matchings, CERS-IE Working Papers, No. CERS-IE WP - 2020/3, Hungarian Academy of Sciences, Institute of Economics, Centre for Economic and Regional Studies, Budapest

This Version is available at:
https://hdl.handle.net/10419/222072

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

INSTITUTE OF ECONOMICS
Centre for Economic and Regional Studies

KRTK KTI

Közgazdaság- és Regionális Tudományi Kutatóközpont
KÖZGAZDASÁG-TUDOMÁNYI INTÉZET

CERS-IE WORKING PAPERS | KRTK-KTI MŰHELYTANULMÁNYOK

Understanding popular matchings via stable matchings

ÁGNES CSEH – YURI FAENZA – TELIKEPALLI KAVITHA –

VLADLENA POWERS

CERS-IE WP – 2020/3

January 2020

https://www.mtakti.hu/wp-content/uploads/2020/01/CERSIEWP202003.pdf

CERS-IE Working Papers are circulated to promote discussion and provoque comments, they have not been peer-reviewed.
Any references to discussion papers should clearly state that the paper is preliminary.
Materials published in this series may be subject to further publication.

INSTITUTE OF ECONOMICS, CENTRE FOR ECONOMIC AND REGIONAL STUDIES,
BUDAPEST, 2020

ABSTRACT

An instance of the marriage problem is given by a graph G together with, for each vertex of G, a strict preference order over its neighbors. A matching M of G is popular in the marriage instance if M does not lose a head-to-head election against any matching where vertices are voters. Every stable matching is a min-size popular matching; another subclass of popular matchings that always exist and can be easily computed is the set of dominant matchings. A popular matching M is dominant if M wins the head-to-head election against any larger matching. Thus every dominant matching is a max-size popular matching and it is known that the set of dominant matchings is the linear image of the set of stable matchings in an auxiliary graph. Results from the literature seem to suggest that stable and dominant matchings behave, from a complexity theory point of view, in a very similar manner within the class of popular matchings.

The goal of this paper is to show that indeed there are differences in the tractability of stable and dominant matchings, and to investigate further their importance for popular matchings. First, we show that it is easy to check if all popular matchings are also stable, however it is co-NP-hard to check if all popular matchings are also dominant. Second, we show how some new and recent hardness results on popular matching problems can be deduced from the NP-hardness of certain problems on stable matchings, also studied in this paper, thus showing that stable matchings can be employed not only to show positive results on popular matching (as is known), but also most negative ones. Problems for which we show new hardness results include finding a min-size (resp. max-size) popular matching that is not stable (resp. dominant). A known result for which we give a new and simple proof is the NP-hardness of finding a popular matching when G is non-bipartite.

JEL codes: C63, C78
Keywords: popular matching, NP-completeness, polynomial algorithm, stable matching

Ágnes Cseh
Centre for Economic and Regional Studies, Institute of Economics, Tóth Kálmán utca 4., Budapest, 1097, Hungary
e-mail: cseh.agnes@krtk.mta.hu

Yuri Faenza
IEOR, Columbia University, New York, USA;
e-mail: yf2414@columbia.edu

Telikepalli Kavitha
Tata Institute of Fundamental Research, Mumbai, India
e-mail: kavitha@tifr.res.in

Vladlena Powers
IEOR, Columbia University, New York, USA;
e-mail: yf2414@columbia.edu

Népszerű párosítások megértése stabil párosítások segítségével

ÁGNES CSEH – YURI FAENZA – TELIKEPALLI KAVITHA –

VLADLENA POWERS

ÖSSZEFOGLALÓ

Adott egy páros gráf G = (A u B, E), ahol minden csúcs szigorú listában rangsorolja a szomszédjait. Egy M párosítást akkor nevezünk népszerűnek, ha nincsen olyan M' párosítás, hogy több csúcs részesíti előnyben M'-t M-mel szemben, mint fordítva. Minden stabil párosítás minimális méretű népszerű párosítás, és egy másik, könnyen kiszámolható alosztálya a népszerű párosításoknak az úgynevezett domináns párosítások halmaza. Egy népszerű párosítást akkor nevezünk dominánsnak, ha az megnyer minden szavazást egy nálánál nagyobb méretű párosítással szemben. Ebből következik, hogy minden domináns párosítás egyben legnagyobb méretű népszerű párosítás is. Az is ismert, hogy a domináns párosítások megkaphatók egy kibővített input stabil párosításainak a lineáris leképezéseiként.

Cikkünkben megmutatjuk, hogy a stabil és a domináns párosításokkal kapcsolatos bonyolultsági eredmények igen különbözőek a két osztályra. Elsőként bebizonyítjuk, hogy könnyű ellenőrizni, hogy minden népszerű párosítás stabil-e, viszont coNP-nehéz eldönteni, hogy minden népszerű párosítás domináns-e. Ezután bemutatunk néhány NP-teljességi bizonyítást népszerű párosításokra, amik stabil párosításokra bizonyított eredményekből következnek. Ezzel demonstráljuk, hogy a stabilitásra bizonyított eredmények nemcsak a pozitív, de a nehézségi bizonyításokban is hasznosnak bizonyulnak. Az újonnan nehéznek besorolt problémák közt megtalálható a legkisebb nem stabil és a legnagyobb nem domináns párosítás találása. Népszerű párosítás létezésére nem páros gráfon adunk egy új és egyszerű bizonyítást.

JEL: C63, C78
Kulcsszavak: népszerű párosítás, NP-teljesség, polinomiális algoritmus, stabil párosítás

Understanding popular matchings via stable matchings

Ágnes Cseh[1], Yuri Faenza[2], Telikepalli Kavitha[3]*, and Vladlena Powers[2]

[1] Centre for Economic and Regional Studies, Institute of Economics, Budapest, Hungary; cseh.agnes@krtk.mta.hu

[2] IEOR, Columbia University, New York, USA; {yf2414, vp2342}@columbia.edu

[3] Tata Institute of Fundamental Research, Mumbai, India; kavitha@tifr.res.in

Abstract. An instance of the marriage problem is given by a graph $G = (A \cup B, E)$, together with, for each vertex of G, a strict preference order over its neighbors. A matching M of G is *popular* in the marriage instance if M does not lose a head-to-head election against any matching where vertices are voters. Every stable matching is a *min-size* popular matching; another subclass of popular matchings that always exist and can be easily computed is the set of *dominant* matchings. A popular matching M is dominant if M wins the head-to-head election against any larger matching. Thus every dominant matching is a *max-size* popular matching and it is known that the set of dominant matchings is the linear image of the set of stable matchings in an auxiliary graph. Results from the literature seem to suggest that stable and dominant matchings behave, from a complexity theory point of view, in a very similar manner within the class of popular matchings.

The goal of this paper is to show that indeed there are differences in the tractability of stable and dominant matchings, and to investigate further their importance for popular matchings. First, we show that it is easy to check if all popular matchings are also stable, however it is co-NP hard to check if all popular matchings are also dominant. Second, we show how some new and recent hardness results on *popular* matching problems can be deduced from the NP-hardness of certain problems on *stable* matchings, also studied in this paper, thus showing that stable matchings can be employed not only to show positive results on popular matching (as is known), but also most negative ones. Problems for which we show new hardness results include finding a min-size (resp. max-size) popular matching that is not stable (resp. dominant). A known result for which we give a new and simple proof is the NP-hardness of finding a popular matching when G is non-bipartite.

1 Introduction

Consider a bipartite graph $G = (A \cup B, E)$ on n vertices and m edges where each vertex has a strict ranking of its neighbors. Such a graph supplied with preference lists, also called a *marriage* instance, is an extensively studied model in two-sided matching markets. The problem of computing a *stable* matching in G is classical. A matching M is stable if there is no *blocking edge* with respect to M, i.e., an edge whose endpoints prefer each other to their respective assignments in M. The notion of stability was introduced by Gale and Shapley [11] in 1962 who showed that stable matchings always exist in G and there is a simple linear time algorithm to find one.

Stable matchings in an instance with an underlying bipartite graph G are well-understood [14], with efficient algorithms [19, 27, 9, 24, 10, 26] to solve several optimization problems that have many applications in economics, computer science, and mathematics. Here we study a related and more relaxed notion called *popularity*. This notion was introduced by Gärdenfors [12] in 1975 who showed that every stable matching is also popular. For any vertex u, its preference over neighbors extends naturally to a preference over matchings as follows: u prefers M to M' either if (i) u is matched in M and unmatched in M' or (ii) u is matched in both and prefers its partner in M to its partner in M'. Let $\psi(M, M')$ be the number of vertices that prefer M to M'.

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

* Work done while visiting Max-Planck-Institut für Informatik, Saarbrücken, Germany.

Hence, in a voting-based context, vertices constitute the set of voters, and each matching in the instance is an alternative. In a head-to-head election between two matchings, each vertex casts a vote for the matching that it prefers and it abstains from voting if its assignment is the same in both matchings. A popular matching, by definition, never loses such a head-to-head election against another matching. Hence, a popular matching is a weak *Condorcet winner* [3, 4] in the corresponding voting instance. It is easy to show that a stable matching is a min-size popular matching [16]. Thus larger matchings and more generally, matchings that achieve more social good, are possible by relaxing the constraint of stability to popularity.

Algorithmic questions for popular matchings in bipartite graphs have been well-studied in the last decade [2, 16, 20, 21, 17, 6, 8]. We currently know efficient algorithms for the following problems in bipartite graphs: (i) min-size popular matchings, (ii) max-size popular matchings, and (iii) finding a popular matching with a given edge. All these algorithms compute either a stable matching or a *dominant* matching.

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 $|M'| > |M|$.*

Thus a dominant matching defeats every larger matching in a head-to-head election, so it immediately follows that a dominant matching is a popular matching of maximum size. The example in Fig. 1 (from [16]) demonstrates the differences between stable, dominant, and max-size matchings.

In the graph $G = (A \cup B, E)$ here, we have $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$. The same preferences are depicted as numbers on the edges and as lists to the left of the drawn graph. Vertex b_1 is the top choice for all a_i's, b_2 is the second choice for a_1 and a_2, and b_3 is the third choice for a_1 alone. The preference lists of the b_i vertices are symmetric. There are two popular matchings here, both of them have the same cardinality: $M_1 = \{(a_1, b_1), (a_2, b_2)\}$ and $M_2 = \{(a_1, b_2), (a_2, b_1)\}$.

The matching M_1 is stable, but not dominant, since it is *not* more popular than the larger matching $M_3 = \{(a_1, b_3), (a_2, b_2), (a_3, b_1)\}$. Observe that in an election between M_1 and M_3, the vertices a_1, b_1 vote for M_1, the vertices a_3, b_3 vote for M_3, and the vertices a_2, b_2 are indifferent between M_1 and M_3; thus $\Delta(M_1, M_3) = 2 - 2 = 0$. The matching M_2 is dominant since M_2 is more popular than M_3: observe that $\Delta(M_2, M_3) = 4 - 2 = 2$ since a_1, b_1, a_2, b_2 prefer M_2 to M_3 while a_3, b_3 prefer M_3 to M_2. However M_2 is not stable, since (a_1, b_1) blocks it.

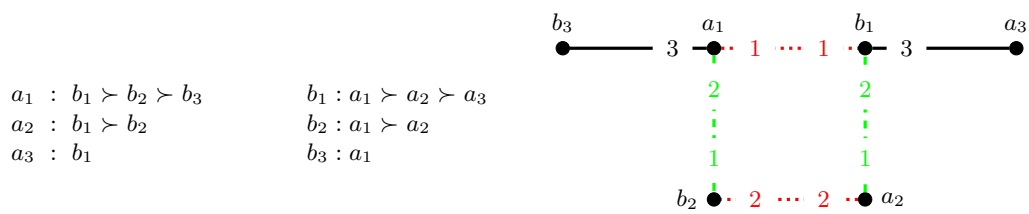

Fig. 1. The above instance admits two popular matchings. The stable matching M_1 is marked by dotted red edges, while the dominant matching M_2 is marked by dashed green edges.

Dominant matchings always exist in a bipartite graph [16] and a dominant matching can be computed in linear time [20]. Moreover, dominant matchings are the linear image of stable matchings in an auxiliary instance [6], hence oftentimes an optimization problem over the set of dominant matchings (e.g. finding one of maximum weight) boils down to solving the same problem on the set of stable matchings. Very recently, the following rather surprising result was shown [8]: it is NP-hard to decide if a bipartite graph admits a popular matching that is *neither stable nor dominant.*

1.1 Our problems, results, and techniques

Everything known so far about stable and dominant matchings seemed to suggest that those classes play somehow symmetric roles in popular matching problems in bipartite graphs: both classes are

always non-empty and one is a tractable subclass of *min-size popular matchings* while the other is a tractable subclass of *max-size popular matchings*. Our first set of results shows that this symmetry is not always the case.

Our starting point is an investigation of the complexity of the following two natural and easy-to-ask questions on popular matchings in a bipartite graph G with strict preference lists:

(1) is *every* popular matching in G also stable?
(2) is *every* popular matching in G also dominant?

Both these questions are trivial to answer in instances that admit popular matchings of more than one size. Then the answer to both questions is "no" since $\{\text{dominant matchings}\} \cap \{\text{stable matchings}\} = \emptyset$ in such graphs as dominant matchings are max-size popular matchings while stable matchings are min-size popular matchings. Thus, in this case, a dominant matching is an *unstable* popular matching and a stable matching is a *non-dominant* popular matching in G. However, when all popular matchings in G have the same size, these questions are non-trivial.

Moreover, it is useful to ask these questions because when there are edge utilities, the problem of finding a max-utility popular matching is NP-hard in general [22] and also hard to approximate to a factor better than 2 [8]; however if every popular matching is stable (similarly, dominant), then the max-utility popular matching problem can be solved in polynomial time. Thus a "yes" answer to either of these questions has applications. We show the following dichotomy here: though both these questions seem analogous, *only one* is easy-to-answer.

($*$) *There is an $O(m^2)$ algorithm to decide if every popular matching in $G = (A \cup B, E)$ is stable, however it is co-NP complete to decide if every popular matching in $G = (A \cup B, E)$ is dominant.*

The first step in proving ($*$) is to show that questions (1) and (2) are equivalent to the following:[4]

(1′) is every *dominant* matching in G also stable?
(2′) is every *stable* matching in G also dominant?

In Section 3, we give a combinatorial algorithm that solves (1′) in time $O(m^2)$. We settle the complexity of (2) and (2′) in Section 4, showing that the problem is co-NP hard. We deduce the latter from the hardness of finding a stable matching with a certain augmenting path: a result that is shown in this paper.

Our hardness reduction is surprisingly simple when compared to those that appeared in recent publications on popular matchings [8, 13, 22], and establishes a new connection between hardness of problems for stable matchings and hardness of problems for popular matchings. This connection turns out to be very fertile: we exploit it further to show NP-hardness of the following new decision problems for a bipartite graph G (in particular, these hardness results are not implied by the reductions from [8, 13, 22]):

(3) is there a stable matching in G that is dominant?
(4) is there a max-size popular matching in G that is not dominant?
(5) is there a min-size popular matching in G that is not stable?

A general graph (not necessarily bipartite) with strict preference lists is called a *roommates* instance. Popular matchings need not always exist in a roommates instance and the popular roommates problem is to decide if a given instance admits one or not. The complexity of the popular roommates problem was open for close to a decade and very recently, two independent proofs of NP-hardness [8, 13] of this problem were shown. Both these proofs are rather lengthy and technical.

We use the hardness result for problem (3) to show a short and simple proof of NP-hardness of the popular roommates problem. Moreover, the hardness result for (5) shows an alternative and much simpler proof of NP-hardness (compared to [8]) of the following decision problem in a marriage

[4] Although similar in spirit, the arguments leading from (1) to (1′) and from (2) to (2′) are not the same, see Lemma 2 and Lemma 3.

instance $G = (A \cup B, E)$ equipped with strict preference lists: is there a popular matching in G that is neither stable nor dominant?[5]

Algorithms for computing min-size/max-size popular matchings and for the popular edge problem compute either stable matchings or dominant matchings. Dominant matchings in G are stable matchings in a related graph G' (see Section 2) and so the machinery of stable matchings is used to solve dominant matching problems. Thus all positive results in the domain of popular matchings can be attributed to stable matchings. Conversely, all hardness results proved in this paper rely on the fact that it is hard to find stable matchings that have / do not have certain augmenting paths. Hence, properties of stable matchings are also responsible for, and provide a unified approach to, the hardness of many popular matching problems.

1.2 Background and related results

In all problems considered in this paper, all vertices of a graph have strict preference lists over their neighbors. The first algorithmic question studied in the domain of popular matchings was in the one-sided preference lists model in bipartite graphs: here, unlike in our setting, one side of the graph consists of agents who have preferences over their neighbors while vertices on the other side are objects with no preferences. Popular matchings need not always exist here and an efficient algorithm was shown in [1] to determine if one exists or not.

Popular matchings always exist in bipartite graphs when every vertex has a strict preference list [12]. However when preference lists are not strict, the problem of deciding if a popular matching exists or not is NP-hard [2, 5]. The first non-trivial algorithms designed for computing popular matchings in bipartite graphs with strict preference lists were the max-size popular matching algorithms [16, 20]. These algorithms compute dominant matchings and the term *dominant matching* was formally defined a few years later in [6] to solve the "popular edge" problem.

As mentioned earlier, it was recently shown that it is NP-hard to decide if a marriage instance admits a popular matching that is neither stable nor dominant [8]. This hardness result was shown by a reduction from 1-in-3 SAT and a consequence of this hardness result was the hardness of the popular roommates problem. The NP-hardness of popular roommates problem shown in [13] was by a reduction from a problem called the *partitioned vertex cover* problem.

There are several efficient algorithms to solve the stable roommates problem [18, 25, 26]. In contrast, the *dominant roommates* problem, i.e., the problem of deciding whether a roommates instance admits a dominant matching or not, is NP-hard [8]. Our NP-hardness proof of the popular roommates problem also proves the hardness of the dominant roommates problem and it is much simpler than the proof in [8].

Organization of the paper. Section 2 contains known facts on popular, stable, and dominant matchings, that will be used throughout the paper. In Section 3, we present an algorithm to decide if G has an unstable popular matching. Section 4 has our co-NP hardness result, which is deduced from the problem of deciding whether stable matchings *with* certain augmenting paths exist, whose hardness is also proved in Section 4. In Section 5 we show how the problem of deciding whether a stable matching *without* certain augmenting paths exists implies new and known hardness results, in particular the hardness of deciding if there exists a matching that is both stable and dominant.

2 Preliminaries

Let $G = (A \cup B, E)$ be the graph in our input. We will often refer to vertices in A and B as *men* and *women*, respectively. We will always assume that each vertex has a strict preference order over his/her neighbors, and the set of these lists is denoted by $\mathcal{P}$. An instance of our problem consists of G and $\mathcal{P}$, but we will omit to explicitly mention $\mathcal{P}$ when it is clear from the context. We often

[5] The reduction in [8] also showed that it was NP-hard to decide if G admits a popular matching that is neither a min-size nor a max-size popular matching. Our reduction does not imply this.

abbreviate $n = |A \cup B|$ and $m = |E|$. We now sketch four important tools developed for popular matchings in earlier papers. Each of these will be used in later parts of this paper.

1) Characterization of popular matchings. Let M be any matching in G. For any edge $(a, b) \notin M$, define $\mathsf{vote}_a(b, M)$ as follows (here $M(a)$ is a's partner in the matching M and $M(a) = \mathsf{null}$ if a is unmatched in M):

$$\mathsf{vote}_a(b, M) = \begin{cases} + & \text{if } a \text{ prefers } b \text{ to } M(a); \\ - & \text{if } a \text{ prefers } M(a) \text{ to } b. \end{cases}$$

We can similarly define $\mathsf{vote}_b(a, M)$. Label every edge $(a, b) \notin M$ by $(\mathsf{vote}_a(b, M), \mathsf{vote}_b(a, M))$. Thus every edge outside M has a label in $\{(\pm, \pm)\}$. Note that an edge e is labeled $(+, +)$ if and only if e is a blocking edge to M.

Let G_M be the subgraph of G obtained by deleting edges labeled $(-, -)$ from G. The following theorem characterizes popular matchings. This characterization holds in non-bipartite graphs as well.

Theorem 1 ([16]). *Matching M is popular in instance $G, \mathcal{P}$ if and only if G_M does not contain any of the following with respect to M:*

(i) an alternating cycle with a $(+, +)$ edge;
(ii) an alternating path with two distinct $(+, +)$ edges;
(iii) an alternating path with a $(+, +)$ edge and an unmatched vertex as an endpoint.

The following theorem characterizes dominant matchings.

Theorem 2 ([6]). *A popular matching M is dominant iff there is no M-augmenting path in G_M.*

2) The graph G'. Dominant matchings in G are equivalent to stable matchings in a related graph G': this equivalence was first used in [6], later simplified in [8]. The graph G' is the bidirected graph corresponding to G. The vertex set of G' is the same as the vertex set $A \cup B$ of G and every edge (a, b) in G is replaced by 2 edges in G': one directed from a to b denoted by (a^+, b^-) and the other directed from b to a denoted by (a^-, b^+). Let $u \in A \cup B$ and suppose $v_1 \succ v_2 \cdots \succ v_k$ is u's preference order in G. Then u's preference order in G' is:

$$v_1^- \succ v_2^- \cdots \succ v_k^- \succ v_1^+ \succ v_2^+ \cdots \succ v_k^+.$$

That is, every vertex prefers outgoing edges to incoming edges and among outgoing edges, it maintains its original preference order and among incoming edges, it maintains again its original preference order. Observe that vertex preferences in G' are expressed on incident edges rather than on neighbors. However it is easy to see that stable matchings in G' are equivalent to stable matchings in the following conventional graph that has 3 vertices $u^+, u^-, d(u)$ corresponding to each vertex u in G'. The preference order of u^+ is $v_1^- \succ v_2^- \cdots \succ v_k^- \succ d(u)$, the preference order of u^- is $d(u) \succ v_1^+ \succ v_2^+ \cdots \succ v_k^+$, and the preference order of $d(u)$ is $u^+ \succ u^-$.

It was shown in [6, 8] that any stable matching M' in G' projects to a dominant matching M in G by setting $(a, b) \in M$ if and only if either (a^+, b^-) or (a^-, b^+) is in M', and conversely, any dominant matching in G can be realized as a stable matching in G'.

3) The set of matched vertices. We will be using the *Rural Hospitals Theorem* for stable matchings (see e.g. [14]) that states that all stable matchings in G match the same subset of vertices. Note that every dominant matching in G matches the same subset of vertices (via the Rural Hospitals Theorem for stable matchings in G'). More generally, the following fact is true, where $V(N)$ is the set of vertices matched in a matching N.

Lemma 1 ([15, 16]). *Let S be a stable, M be a popular, and D be a dominant matching in a marriage instance $G = (A \cup B, E), \mathcal{P}$. Then $V(S) \subseteq V(M) \subseteq V(D)$.*

Thus, in particular, in instances where stable matchings have the same size as dominant matchings, all popular matchings match the same subset of vertices.

4) Witness of a popular matching. Let $\tilde{G}$ be the graph G augmented with self-loops. That is, we assume each vertex is its own last choice neighbor. So we can henceforth regard any matching M in G as a perfect matching $\tilde{M}$ in $\tilde{G}$ by including self-loops at all vertices left unmatched in M. The following edge weight function wt_M in $\tilde{G}$ will be useful to us. For any edge (a, b) in G, define:

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

We need to define wt_M for self-loops as well: let $\mathsf{wt}_M(u, u) = 0$ if u is matched to itself in $\tilde{M}$, else $\mathsf{wt}_M(u, u) = -1$. For any matching N in G, it is easy to see that $\mathsf{wt}_M(\tilde{N}) = \Delta(N, M)$ and so M is popular in G if and only if every perfect matching in $\tilde{G}$ has weight at most 0. Theorem 3 follows from LP-duality and the fact that G is a bipartite graph.

Theorem 3 ([23, 21]). *A matching M in $G = (A \cup B, E), \mathcal{P}$ is popular if and only if there exists a vector $\boldsymbol{\alpha} \in \{0, \pm 1\}^n$ such that $\sum_{u \in A \cup B} \alpha_u = 0$,*

$$\alpha_a + \alpha_b \;\geq\; \mathsf{wt}_M(a,b) \quad \forall\, (a,b) \in E \qquad \textit{and} \qquad \alpha_u \;\geq\; \mathsf{wt}_M(u,u) \quad \forall\, u \in A \cup B.$$

For any popular matching M, a vector $\boldsymbol{\alpha}$ as given in Theorem 3 will be called M's *witness* or *dual certificate*. A popular matching may have several witnesses. Any stable matching has $\mathbf{0}$ as a witness.

Call an edge e *popular* if there is a popular matching in G that contains e. For any popular edge (a, b), it was shown in [7] (using complementary slackness) that $\alpha_a + \alpha_b = \mathsf{wt}_M(a, b)$. This will be a useful fact for us. Another useful fact from [7] (again by complementary slackness) is that if vertex u is left unmatched in some popular matching then $\alpha_u = \mathsf{wt}_M(u, u)$ for every popular matching M; thus $\alpha_u = 0$ if u is left unmatched in M, else $\alpha_u = -1$.

We now illustrate each of the above four tools on our example instance $G, \mathcal{P}$ from Fig. 1.

1) Recall the matching $M_1 = \{(a_1, b_1), (a_2, b_2)\}$. We will test if M_1 is popular/dominant using Theorems 1 and 2. First we label each edge (a, b) not in M_1 by $(\mathsf{vote}_a(b, M_1(a)), \mathsf{vote}_b(a, M_1(b)))$, see Fig.2. Since no edge is labeled $(-,-)$, the subgraph $G_{M_1} = G$. Observe that G_{M_1} has no forbidden alternating path/cycle as given in Theorem 1. Thus M_1 is popular. However there is an M_1-augmenting path $\langle b_3, a_1, b_1, a_3 \rangle$ in G_{M_1}, so M_1 is not dominant (by Theorem 2).

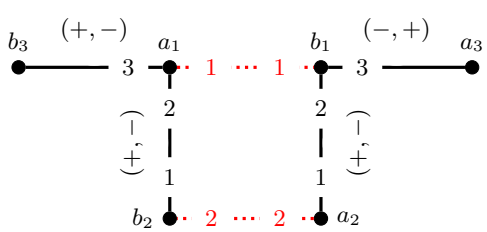

Fig. 2. Edge labels with respect to $M_1 = \{(a_1, b_1), (a_2, b_2)\}$.

2) The bidirected graph G' corresponding to G is given in Fig. 3. Observe that in the transformed set of preference lists, each vertex ranks its outgoing edges in their original order, followed by the incoming copies of the same edges in their original order. The only stable matching in G' is $\{(a_1^+, b_2^-), (a_2^-, b_1^+)\}$ (marked by dashed green edges in Fig. 3) and it projects to $M_2 = \{(a_1, b_2), (a_2, b_1)\}$—this is the only dominant matching in G.

$a_1 : b_1^- \succ b_2^- \succ b_3^- \succ b_1^+ \succ b_2^+ \succ b_3^+$
$a_2 : b_1^- \succ b_2^- \succ b_1^+ \succ b_2^+$
$a_3 : b_1^- \succ b_1^3$

$b_1 : a_1^- \succ a_2^- \succ a_3^- \succ a_1^+ \succ a_2^+ \succ a_3^+$
$b_2 : a_1^- \succ a_2^- \succ a_1^+ \succ a_2^+$
$b_3 : a_1^- \succ a_1^+$

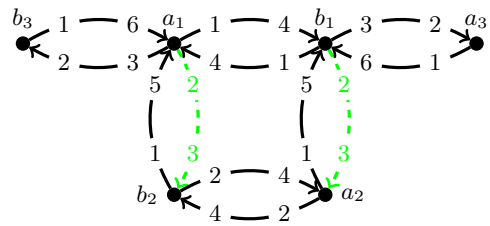

Fig. 3. The bidirected graph G' corresponding to G, and the transformed lists.

3) The set of matched vertices is the same for all popular matchings in the above instance: $V(M_1) = V(M_2) = \{(a_1, a_2, b_1, b_2\}$.
4) We will construct a witness $\boldsymbol{\alpha}$ for M_2, see Fig. 4. So $\boldsymbol{\alpha} \in \{0, \pm 1\}^6$. Since M_2 leaves a_3 and b_3 unmatched, it has to be the case that $\alpha_{a_3} = \alpha_{b_3} = 0$. We also know that $\alpha_{a_1} = \alpha_{b_1} = 1$ because $\alpha_{a_1} + \alpha_{b_1} \geq \mathsf{wt}_{M_2}(a_1, b_1) = 2$. Since $\sum_{u \in A \cup B} \alpha_u = 0$, the only possibility for the remaining two vertices is $\alpha_{a_2} = \alpha_{b_2} = -1$. We have $\alpha_u \geq \mathsf{wt}_{M_2}(u, u)$ for all vertices u since $\mathsf{wt}_{M_2}(a_3, a_3) = \mathsf{wt}_{M_2}(b_3, b_3) = 0$ and $\mathsf{wt}_{M_2}(v, v) = -1$ for other vertices v.
Observe that $\alpha_{a_1} + \alpha_{b_3} = 1 > 0 = \mathsf{wt}_{M_2}(a_1, b_3)$ and $\alpha_{a_3} + \alpha_{b_1} = 1 > 0 = \mathsf{wt}_{M_2}(a_3, b_1)$. For the remaining 4 edges, the corresponding constraint is tight. That is, $\alpha_a + \alpha_b = \mathsf{wt}_{M_2}(a, b)$ for $a \in \{a_1, a_2\}$ and $b \in \{b_1, b_2\}$.

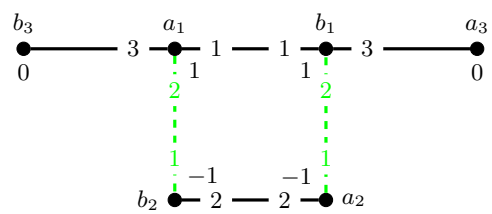

Fig. 4. A witness $\boldsymbol{\alpha}$ constructed for $M_2 = \{(a_1, b_2), (a_2, b_1)\}$ is denoted by the red labels next to the vertices.

3 Finding an unstable popular matching

We are given $G = (A \cup B, E)$ with strict preference lists and the problem is to decide if every popular matching in G is also stable, i.e., if {popular matchings} = {stable matchings} or not in G. In this section we show an efficient algorithm to answer this question.

Problem 1 *Input: A bipartite graph $G = (A \cup B, E)$ with strict preference lists.*
Decide: If there is an unstable popular matching in G.

Let G admit an unstable popular matching M and let $\boldsymbol{\alpha} \in \{0, \pm 1\}^n$ be M's witness. Let A_0 be the set of vertices $a \in A$ with $\alpha_a = 0$ and let B_0 be the set of vertices $b \in B$ with $\alpha_b = 0$.

Let M_0 be the set of edges $(a, b) \in M$ such that $\alpha_a = \alpha_b = 0$. Let M_1 be the set of edges $(a, b) \in M$ such that $\alpha_a, \alpha_b \in \{\pm 1\}$. Note that $M = M_0 \uplus M_1$, since the parities of α_a and α_b have to be the same for any popular edge due to the tightness of the constraint $\alpha_a + \alpha_b = \mathsf{wt}_M(a, b) = 0$.

The construction can be followed in Fig. 5. $A \setminus A_0$ has been further split into $A_1 \cup A_{-1}$: A_1 is the set of those vertices a with $\alpha_a = 1$ and A_{-1} is the set of those vertices a with $\alpha_a = -1$; similarly, $B \setminus B_0$ has been further split into $B_1 \cup B_{-1}$: B_1 is the set of those vertices b with $\alpha_b = 1$ and B_{-1} is the set of those vertices b with $\alpha_b = -1$. We have $M_1 \subseteq (A_1 \times B_{-1}) \cup (A_{-1} \times B_1)$ since for any edge (a, b) in M, we have $\alpha_a + \alpha_b = \mathsf{wt}_M(a, b) = 0$.

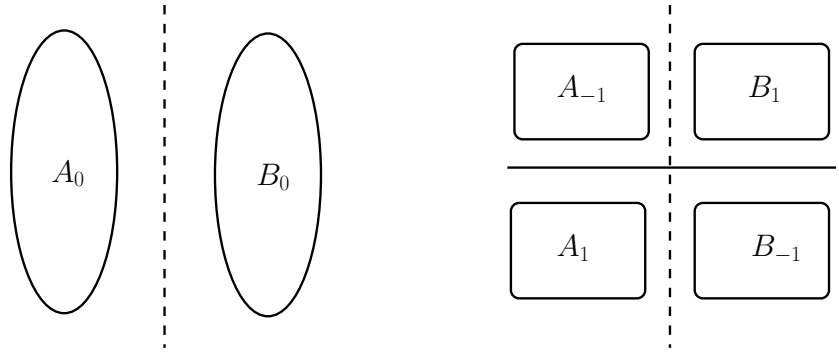

Fig. 5. M_0 is the matching M restricted to vertices in $A_0 \cup B_0$ and M_1 is the matching M restricted to remaining vertices.

Now we run a transformation $M_0 \rightsquigarrow D$ as given in [6]: let G_0 be the graph G restricted to $A_0 \cup B_0$. Run Gale-Shapley algorithm in the graph G'_0 with the starting matching $M'_0 = \{(u^+, v^-) : (u, v) \in M_0 \text{ and } u \in A_0, v \in B_0\}$, where G'_0 is the graph obtained from G_0 as described in Section 2. That is, instead of starting with the empty matching, we start with the matching M'_0 in G'_0; unmatched men in A_0 propose in decreasing order of preference and whenever a woman receives a proposal from a neighbor that she prefers to her current partner (her preferences as given in G'_0), she rejects her current partner and accepts this proposal. This results in a stable matching in G'_0, equivalently, a dominant matching D in G_0, see Section 2. It was moreover shown in [6] that $M^* = M_1 \cup D$ is a dominant matching in G. We include a new and much simpler proof of this fact below.

Claim 1 *The matching $M^* = M_1 \cup D$ is dominant in G.*

Proof. The dominant matching D was obtained as the linear image of a stable matching (call it D') in G'_0. Let A'_1 be the set of $a \in A_0$ such that $(a^+, b^-) \in D'$ for some $b \in B_0$ and $A'_{-1} = A_0 \setminus A'_1$. Similarly, let B'_1 be the set of $b \in B_0$ such that $(a^-, b^+) \in D'$ for some $a \in A_0$ and $B'_{-1} = B_0 \setminus B'_1$. See Fig. 6.

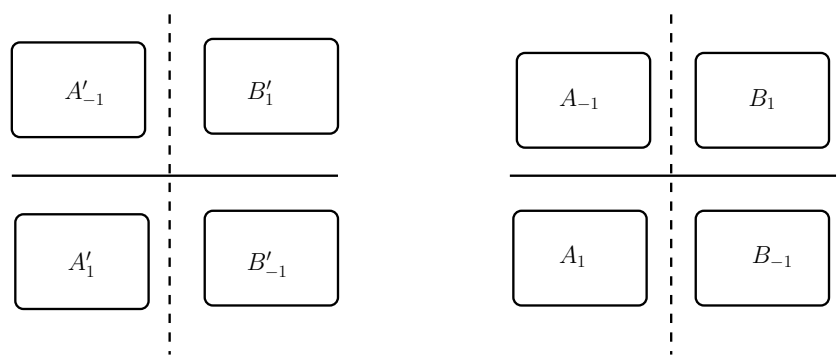

Fig. 6. We transformed the stable matching M_0 on $A_0 \cup B_0$ to the dominant matching D: this partitions A_0 into $A'_1 \cup A'_{-1}$ and B_0 into $B'_1 \cup B'_{-1}$. We also have $A \setminus A_0 = A_1 \cup A_{-1}$ and $B \setminus B_0 = B_1 \cup B_{-1}$.

Observe that every vertex in $A'_1 \cup B'_1$ is matched in D, however not every vertex in $A'_{-1} \cup B'_{-1}$ is necessarily matched in D. That is, $A'_{-1} \cup B'_{-1}$ may contain unmatched vertices.

The popular matching M has a witness $\boldsymbol{\alpha} \in \{0, \pm 1\}^n$: recall that this was the witness used to partition $A \cup B$ into $A_0 \cup B_0$ and $(A \setminus A_0) \cup (B \setminus B_0)$. We have $\alpha_u = 0$ for all $u \in A_0 \cup B_0$ and $\alpha_u \in \{\pm 1\}$ for all $u \in (A \setminus A_0) \cup (B \setminus B_0)$.

In order to prove the popularity of $M^* = M_1 \cup D$, we will show a witness $\boldsymbol{\beta}$ as follows. For the vertices outside $A_0 \cup B_0$, set $\beta_u = \alpha_u$ since nothing has changed for these vertices in the transformation $M \rightsquigarrow M^*$. For the vertices in $A_0 \cup B_0$, set $\beta_u = 1$ for $u \in A'_1 \cup B'_1$ and $\beta_u = -1$ for every matched vertex $u \in A'_{-1} \cup B'_{-1}$. For every unmatched vertex u, set $\beta_u = 0$.

Thus $\beta_u \geq \mathsf{wt}_{M^*}(u,u)$ for all u and $\sum_{u:A\cup B} \beta_u = 0$: this is because $D \subseteq (A'_1 \times B'_{-1}) \cup (A'_{-1} \times B'_1)$, so for every edge $(a,b) \in D$, we have $\beta_a + \beta_b = 0$; recall that $\alpha_a + \alpha_b = 0$ for all $(a,b) \in M_1$. What is left to show is that $\beta_a + \beta_b \geq \mathsf{wt}_{M^*}(a,b)$ for every edge (a,b).

The correctness of the Gale-Shapley algorithm in G'_0 to compute popular matchings in G_0 immediately implies that every edge with both endpoints in $A_0 \cup B_0$ is covered by the sum of β-values of its endpoints (see [8]). We will now show that every edge with one endpoint in $A_0 \cup B_0$ and the other endpoint in $(A \setminus A_0) \cup (B \setminus B_0)$ is also covered.

- Every edge in $A'_1 \times B_1$ is covered since $\beta_a + \beta_b = 2 \geq \mathsf{wt}_{M^*}(a,b)$. Similarly every edge in $A_1 \times B'_1$ is also covered.
- Consider any edge $(a,b) \in A'_1 \times B_{-1}$. We have $\alpha_a = 0$ and $\alpha_b = -1$ and so $\mathsf{wt}_M(a,b) \leq -1$, i.e., $\mathsf{wt}_M(a,b) = -2$ since this is a value in $\{0, \pm 2\}$. This means b prefers $M(b) = M^*(b)$ to a. Thus $\mathsf{wt}_{M^*}(a,b) \leq 0$. Since $\beta_a = 1$ and $\beta_b = -1$, we have $\beta_a + \beta_b \geq \mathsf{wt}_{M^*}(a,b)$. Similarly every edge in $A_{-1} \times B'_1$ is also covered.
- Consider any edge (a,b) in $A_{-1} \times B'_{-1}$. We have $\mathsf{wt}_M(a,b) \leq -1$ which means that $\mathsf{wt}_M(a,b) = -2$. That is, both a and b prefer their partners in M to each other. We will now show that $\mathsf{wt}_{M^*}(a,b) = -2$. Since $M(a) = M^*(a)$, nothing has changed for a and so a prefers $M^*(a)$ to b. We claim that $M^*(b) \succeq_b M(b)$, i.e., b is no worse in M^* than in M.
  This is because we ran Gale-Shapley algorithm in G'_0 with M'_0 as the starting matching. So if $b \in B'_{-1}$ changed its partner from u^+ in M'_0 to v^+ in D' then b prefers v to u. Thus every $b \in B'_{-1}$ has at least as good a partner in D as in M_0. Hence $\mathsf{wt}_{M^*}(a,b) = -2 = \beta_a + \beta_b$. By the same reasoning, we can argue that every edge in $A_1 \times B'_{-1}$ is also covered.
- Consider any edge (a,b) in $A'_{-1} \times B_{-1}$. We have $\alpha_a = 0$ and $\alpha_b = -1$ and so $\mathsf{wt}_M(a,b) \leq -1$, i.e., $\mathsf{wt}_M(a,b) = -2$. So b prefers $M(b) = M^*(b)$ to a. We claim that $M^*(a) \succeq_a M(a)$. This will imply that $\mathsf{wt}_{M^*}(a,b) = -2$.
  Recall that D' is a stable matching in G'_0: here every vertex prefers outgoing edges to incoming edges, thus every $a \in A'_{-1}$ gets at least as good a partner as $S^*(a)$ in D, where S^* is the men-optimal stable matching in G_0; in turn, $S^*(a) \succeq_a M_0(a)$ for all $a \in A$. Hence $M^*(a) \succeq_a M(a)$ for every $a \in A'_{-1}$. Thus $\mathsf{wt}_{M^*}(a,b) = -2 = \beta_a + \beta_b$. We can similarly show that every edge in $A'_{-1} \times B_1$ is also covered.

Thus M^* is a popular matching in G. We will now show that M^* is a *dominant* matching in G. Observe that $\beta_u \in \{\pm 1\}$ for every matched vertex u: we will use this fact to show that M^* is dominant. Recall that $\beta_u = 0$ for all unmatched vertices u. Let $\rho = \langle a_0, b_1, a_1, \ldots, b_k, a_k, b_{k+1} \rangle$ be any M^*-augmenting path in G. We have $\beta_{a_0} = \beta_{b_{k+1}} = 0$, hence $\beta_{b_1} = \beta_{a_k} = 1$, and so $\beta_{a_1} = \beta_{b_k} = -1$.

Either (i) $\beta_{b_2} = -1$ or (ii) $\beta_{b_2} = 1$ which implies that $\beta_{a_2} = -1$. It is now easy to see that in the path ρ, for some $i \in \{1, \ldots, k-1\}$ the edge (a_i, b_{i+1}) has to be labeled $(-,-)$. That is, ρ is *not* an augmenting path in G_{M^*}. Thus there is no M^*-augmenting path in G_{M^*}, hence M^* is a dominant matching in G (by Theorem 2). □

Since M is an unstable matching, there is an edge (a,b) that blocks M. Since $\mathsf{wt}_M(a,b) = 2$, the endpoints of a blocking edge (a,b) have to satisfy $\alpha_a = \alpha_b = 1$; so $a \in A_1$ and $b \in B_1$. The edge (a,b) blocks M^* as well since the matching M_1 was unchanged by this transformation of M_0 to D, so $M^*(a) = M_1(a)$ and $M^*(b) = M_1(b)$, thus a and b prefer each other to their respective partners in M^*. So M^* is an unstable dominant matching and Lemma 2 follows.

Lemma 2. *If $G, \mathcal{P}$ has an unstable popular matching then $G, \mathcal{P}$ admits an unstable dominant matching.*

Hence in order to answer the question of whether every popular matching in G is stable or not, we need to decide if there exists a dominant matching M in G with a blocking edge. We present a simple combinatorial algorithm for this problem.

Our algorithm is based on the equivalence between dominant matchings in G and stable matchings in G'. Our task is to determine if there exists a stable matching in G' that includes a pair of edges (a^+, v^-) and (u^-, b^+) such that a and b prefer each other to v and u, respectively, in G. It is easy to decide in $O(m^3)$ time whether such a stable matching exists or not in G'.

– For every pair of edges $e_1 = (a, v)$ and $e_2 = (u, b)$ in G such that a and b prefer each other to v and u, respectively: determine if there is a stable matching in G' that contains the pair of edges (a^+, v^-) and (u^-, b^+).

In the graph G', we modify Gale-Shapley algorithm so that b rejects proposals from all neighbors ranked worse than u^- and v rejects all proposals from neighbors ranked worse than a^+. If the resulting algorithm returns a stable matching that contains the edges (a^+, v^-) and (u^-, b^+), then we have the desired matching; else G' has no stable matching that contains this particular pair of edges.

In order to determine if there exists an unstable dominant matching, we may need to go through all pairs of edges $(e_1, e_2) \in E \times E$. Since we can determine in linear time if there exists a stable matching in G' with any particular pair of edges [14], the running time of this algorithm is $O(m^3)$.

A faster algorithm. It is easy to improve the running time to $O(m^2)$. For each $(a, b) \in E$ we check the following.

($\circ$) Does there exist a stable matching in G' such that (1) a^+ is matched to a neighbor that is ranked worse than b^- in a's list, and (2) b^+ is matched to a neighbor that is ranked worse than a^- in b's list?

We modify the Gale-Shapley algorithm in G' so that (1) b rejects all offers from superscript $+$ neighbors, i.e., b accepts proposals only from superscript $-$ neighbors, and (2) every neighbor of a that is ranked better than b^- rejects proposals from a^+.

Suppose ($\circ$) holds. Then this modified Gale-Shapley algorithm returns among all such stable matchings, the most men-optimal and women-pessimal one [14]. Thus among all stable matchings that match a^+ to a neighbor ranked worse than b^- and that include some edge $(*, b^+)$, the matching returned by the above algorithm matches b to its least preferred neighbor and a to its most preferred neighbor.

Hence if the modified Gale-Shapley algorithm returns a matching that is (i) unstable or (ii) includes an edge $(a^-, *)$ or (iii) matches b^+ to a neighbor better than a^-, then there is no dominant matching M in G such that the pair (a, b) blocks M. Else we have the desired stable matching in G', call this matching M'.

The projection of the matching M' on to the edge set of G will be a dominant matching in G with (a, b) as a blocking edge. Since we may need to go through all edges in E and the time taken for any edge (a, b) is $O(m)$, the entire running time of this algorithm is $O(m^2)$. We have thus shown the following theorem.

Theorem 4. *Given $G = (A \cup B, E)$ on m edges and strict preference lists of its vertices, we can decide in $O(m^2)$ time whether every popular matching in G is stable or not; if not, we can return an unstable popular matching.*

4 Finding a non-dominant popular matching

Given an instance $G = (A \cup B, E), \mathcal{P}$, the problem we consider here is to decide if every popular matching is also dominant, i.e., to decide if {popular matchings} = {dominant matchings} or not in G.

Problem 2 *Input: A bipartite graph $G = (A \cup B, E)$ with strict preference lists.*
Decide: If there is a non-dominant popular matching in G.

In this section, we show the following.

Theorem 5. *Given $G = (A \cup B, E)$ with strict preference lists, it is NP-complete to decide if G admits a popular matching that is not dominant.*

We start with the following lemma, that is the counterpart of Lemma 2.

Lemma 3. *If $G, \mathcal{P}$ has a non-dominant popular matching M then $G, \mathcal{P}$ admits a non-dominant stable matching N. If M is given, then N can be found efficiently.*

Proof. Let M be a non-dominant popular matching in G and $\boldsymbol{\alpha} \in \{0, \pm 1\}^n$ its witness (see Theorem 3). We will use the decomposition illustrated in Fig. 5 to show the existence of a non-dominant stable matching in G. As per the decomposition in Fig. 5, $M = M_0 \uplus M_1$. Since M is not dominant, there exists an M-augmenting path ρ in G_M (by Theorem 2). The endpoints of ρ (call them u and v) are unmatched in M, hence $\alpha_u = \alpha_v = 0$ (see Section 2) and so u and v are in $A_0 \cup B_0$.

In the graph G_M, the vertices in $B_{-1} \cup A_{-1}$ (these vertices have α-value -1) are adjacent only to vertices in $A_1 \cup B_1$ as all other edges incident to vertices in $B_{-1} \cup A_{-1}$ are labeled $(-,-)$, and these are not present in G_M. Suppose the M-alternating path ρ leaves the vertex set $A_0 \cup B_0$, i.e., suppose it contains a non-matching edge between $A_0 \cup B_0$ and $A_1 \cup B_1$. Since the partners of vertices in $A_1 \cup B_1$ are in $B_{-1} \cup A_{-1}$, the path ρ can never return to $A_0 \cup B_0$. However we know that the last vertex v of ρ is in $A_0 \cup B_0$. Thus ρ never leaves $A_0 \cup B_0$, i.e., ρ is an M_0-augmenting path in $G^*_{M_0}$, where G^* is the graph G restricted to vertices in $A_0 \cup B_0$.

We now run a transformation $M_1 \rightsquigarrow S$ as given in [6] to convert M_1 into a stable matching S as follows. The matching S is obtained by running Gale-Shapley algorithm in the subgraph G_1, which is the graph G restricted to $(A \setminus A_0) \cup (B \setminus B_0)$: however, rather than starting with the empty matching, we start with the matching given by $M_1 \cap (A_{-1} \times B_1)$. So unmatched men (these are vertices in A_1 to begin with) propose in decreasing order of preference and whenever a woman receives a proposal from a neighbor that she prefers to her current partner (her preferences as given in G), she rejects her current partner and accepts this proposal. This results in a stable matching S in G_1.

It was shown in [6] that $N = M_0 \cup S$ is a stable matching in G. We include a new and simple proof of this below (see Claim 2). Since ρ is an M_0-augmenting path in $G^*_{M_0}$ and M_0 is a subset of N, it follows that ρ is an N-augmenting path in G_N. Thus N is a non-dominant stable matching in G. □

Claim 2 *The matching $N = M_0 \cup S$ is a stable matching in G.*

Proof. The matching S is obtained by running Gale-Shapley algorithm in the graph G_1 on vertex set $(A \setminus A_0) \cup (B \setminus B_0)$. We did not compute the matching S from scratch — we started with edges of the matching M_1 restricted to $A_{-1} \times B_1$. So in the resulting matching S, it is easy to see the following two useful properties:

- $S(b) \succeq_b M_1(b)$ for every $b \in B_1$. This is because to begin with, every $b \in B_1$ is matched to $M_1(b)$ and b will change her partner only if she receives a proposal from a neighbor better than $M_1(b)$.
- $S(a) \succeq_a M_1(a)$ for every $a \in A_1$. This is because all vertices in B_{-1} are unmatched in our starting matching and every $b \in B_{-1}$ prefers her partner in M_1 to any neighbor in A_{-1} (since every edge in $A_{-1} \times B_{-1}$ is labeled $(-,-)$ with respect to M_1). Thus in the matching S, a will get accepted either by $M_1(a)$ or a better neighbor.

It is now easy to show that N is a stable matching. We already know that M_0 is a stable matching on $A_0 \cup B_0$ and S is a stable matching on $(A \setminus A_0) \cup (B \setminus B_0)$. It is left to show that no edge (a, b) with one endpoint in $A_0 \cup B_0$ and another endpoint in $(A \setminus A_0) \cup (B \setminus B_0)$ *blocks* N, i.e., we need to show that $\mathsf{wt}_N(a, b) \leq 0$ for every such edge (a, b).

Suppose $a \in A_0$ and $b \in B_{-1}$. Let $\boldsymbol{\alpha}$ be the witness of M used to partition $A \cup B$ into $A_0 \cup B_0$ and $(A \setminus A_0) \cup (B \setminus B_0)$. So $\alpha_a = 0$ and $\alpha_b = -1$, hence $\mathsf{wt}_M(a, b) \leq -1$, i.e., $\mathsf{wt}_M(a, b) = -2$. Thus a prefers $M_0(a) = N(a)$ to b. Hence $\mathsf{wt}_N(a, b) \leq 0$. We can similarly show that $\mathsf{wt}_N(a, b) \leq 0$ for $a \in A_{-1}$ and $b \in B_0$.

Suppose $a \in A_0$ and $b \in B_1$. Then $\alpha_a = 0$ and $\alpha_b = 1$ and so $\mathsf{wt}_M(a, b) \leq 1$, i.e., $\mathsf{wt}_M(a, b) \leq 0$. So if a prefers $M_0(a) = N(a)$ to b then we can immediately conclude that $\mathsf{wt}_N(a, b) \leq 0$. Else b prefers $M_1(b)$ to a and we have noted above that $S(b) \succeq_b M_1(b)$ for every $b \in B_1$. Thus b prefers $N(b) = S(b)$ to a and so $\mathsf{wt}_N(a, b) \leq 0$. We can similarly argue that $\mathsf{wt}_N(a, b) \leq 0$ for every $a \in A_1$ and $b \in B_0$. This finishes the proof of this claim. □

Our problem now is to decide if there exists a non-dominant stable matching N in G, i.e., a stable matching N with an N-augmenting path in G_N (see Theorem 2). We show a reduction from 3SAT. Given a 3SAT formula ϕ, we transform $\phi = C_1 \wedge \cdots \wedge C_m$ as follows: let $X_1, \ldots, X_n$ be the n variables in ϕ. For each i:

- replace all occurrences of $\neg X_i$ in ϕ with X_{n+i} (a single new variable);
- add the clauses $X_i \vee X_{n+i}$ and $\neg X_i \vee \neg X_{n+i}$ to capture $\neg X_i \iff X_{n+i}$.

Thus, the updated formula is $\phi = C_1 \wedge \cdots \wedge C_m \wedge C_{m+1} \wedge \cdots \wedge C_{m+n} \wedge D_{m+n+1} \wedge \cdots \wedge D_{m+2n}$, where $C_1, \ldots, C_m$ are the original m clauses with negated literals substituted by new variables and for $1 \leq i \leq n$: C_{m+i} is the clause $X_i \vee X_{n+i}$ and D_{m+n+i} is the clause $\neg X_i \vee \neg X_{n+i}$. Corresponding to the above formula ϕ, we will construct an instance $G, \mathcal{P}$ whose high-level picture is shown in Fig. 7.

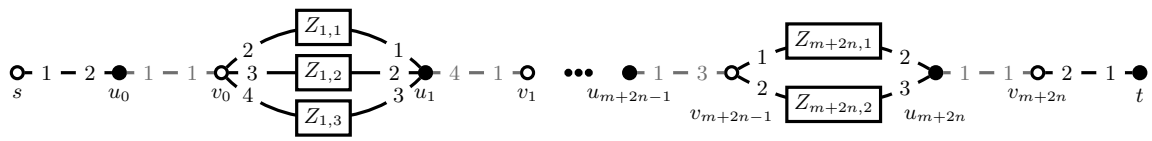

Fig. 7. The high-level picture of the instance $G, \mathcal{P}$.

G is the series composition of an edge (s, u_0), a gadget for each clause i that starts with node v_{i-1} and ends with node u_i, and finally, an edge (v_{m+2n}, t). The gadget corresponding to clause i contains the parallel composition of disjoint gadgets $Z_{i,j}$ for each literal j in clause i. Note that gadgets $Z_{i,j}$ associated with positive literals (so $i \leq m+n$) are different from those associated with negated literals (here $i \geq m+n+1$).

For a variable x, we use a_i, b_i, a'_i, b'_i to denote the 4 vertices in the gadget of x in C_i and c_k, d_k, c'_k, d'_k to denote the 4 vertices in the gadget of $\neg x$ in D_k. The edge (a_i, d_k) will be called a *consistency* edge. This connects the gadget of a positive literal to the (unique) corresponding negative literal (see Fig. 8). Definitions of preference lists are given in Section 4.1. Here, we explain the main steps of the proof. Define $F := \{(u_i, v_i) : 0 \leq i \leq m+2n\}$ to be the set of *basic edges*.

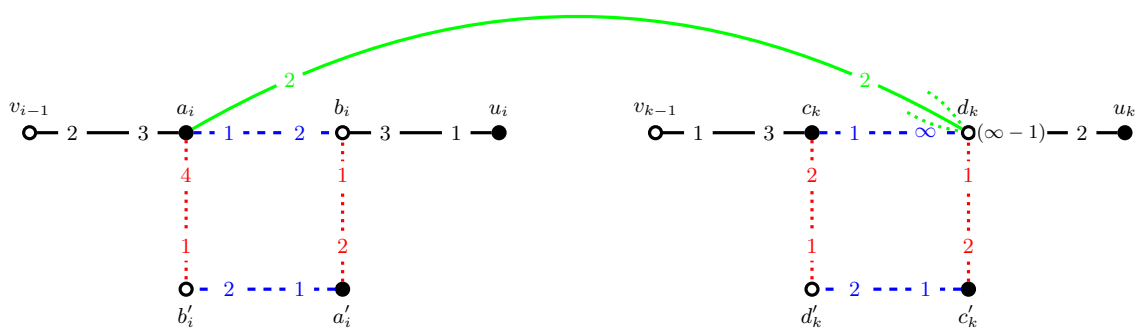

Fig. 8. Suppose x occurs in clause C_i (gadget on the left) and $\neg x$ occurs in clause D_k (gadget on the right). The rank ∞ on the edge (d_k, c_k) denotes that c_k is d_k's *last* choice neighbor and the rank $\infty - 1$ on the edge (d_k, u_k) denotes that u_k is d_k's *last but one* choice neighbor.

Lemma 4. *Let S be any stable matching in $G, \mathcal{P}$. Then*

1. *S leaves s, t unmatched, $F \subseteq S$, and S contains no consistency edge.*

Also, for every variable $x \in \{X_1, \ldots, X_{2n}\}$, we have:

2. *From the gadget of $\neg x$, either (i) $(c_k, d_k), (c'_k, d'_k) \in S$ or (ii) $(c_k, d'_k), (c'_k, d_k) \in S$.*
   — If (i) happens, we say that the gadget is in true state*, else that it is in* false state*.*
3. *From a gadget of x in, say, the i-th clause, either the pair of edges (i) $(a_i, b'_i), (a'_i, b_i)$ or (ii) $(a_i, b_i), (a'_i, b'_i)$ is in S.*
   — If (i) happens, we say that the gadget is in true state*, else that it is in* false state*.*
4. *If the gadget of $\neg x$ is in true state then all the gadgets of x are in false state.*

Conversely, any matching S of G that satisfies 1-4 above is stable.

Lemma 4 implies that each stable matching can be mapped to a true/false assignment as follows:

– For each x, if the gadget of $\neg x$ is in true state, then set $x = \mathsf{false}$, else set $x = \mathsf{true}$.

Conversely, for any true/false assignment A to the variables in ϕ, we define an associated matching M_A as follows. First, include all basic edges in M_A. For any $x \in \{X_1, \ldots, X_{2n}\}$:

– If $x = \mathsf{true}$ then set all gadgets corresponding to x in true state, and the gadget corresponding to $\neg x$ in false state. Thus the *dotted red* edges (see Fig. 8) from all these gadgets get added to M_A.
– Otherwise, set all gadgets corresponding to x in false state and the gadget corresponding to $\neg x$ in true state. Thus the *dashed blue* edges (see Fig. 8) from all these gadgets get added to M_A.

Lemma 4 implies that M_A is stable. The next fact concludes the reduction.

Lemma 5. *Let S be a stable matching in $G, \mathcal{P}$. If there is an augmenting path ρ in G_S, then ρ goes from s to t, does not use any consistency edge, and passes, in each clause, in exactly one gadget, which is in true state. In particular, the assignment corresponding to S is feasible. Conversely, if A is a feasible assignment, in each clause i there is a gadget Z_{i,j_i} in true state, and there exists an augmenting path in G_S that passes through Z_{i,j_i} for all i.*

In the rest of the section, we give the missing details from the construction, and prove Lemma 4 and Lemma 5.

4.1 The preference lists

The gadgets corresponding to an occurrence of x in the i-th clause and the occurrence of $\neg x$ are given in Fig. 8. Let D_k be the clause that contains the unique occurrence of $\neg x$ in the transformed formula ϕ.

Note that the labels of vertices a_i, b_i, a'_i, b'_i should depend on x; however, for the sake of readability, we omit the dependency on x, since it is always clear from the context. Similarly for the labels of vertices c_k, d_k, c'_k, d'_k.

We now describe the preference lists of the 4 vertices in the gadget of x in the clause C_i.

$$a_i : \ b_i \succ d_k \succ v_{i-1} \succ b'_i \qquad\qquad a'_i : \ b'_i \succ b_i$$
$$b_i : \ a'_i \succ a_i \succ u_i \qquad\qquad b'_i : \ a_i \succ a'_i$$

We now describe the preference lists of the 4 vertices in the gadget of $\neg x$.

$$c_k : \ d_k \succ d'_k \succ v_{k-1} \qquad\qquad c'_k : \ d'_k \succ d_k$$
$$d_k : \ c'_k \succ a_i \succ a_j \succ \cdots \succ u_k \succ c_k \qquad\qquad d'_k : \ c_k \succ c'_k$$

Here $a_i, a_j, \ldots$ are the occurrences of the a-vertex in all the gadgets corresponding to literal x in the formula ϕ. That is, the literal x occurs in clauses $i, j, \ldots$ The order among the vertices $a_i, a_j, \ldots$ in d_k's preference list does not matter.

We now describe the preference lists of vertices u_i and v_i. The preference list of u_0 is $v_0 \succ s$ and for $1 \leq i \leq m+n$, the preference list of u_i is as given on the left (these correspond to *positive* clauses C_i) and for $m+n+1 \leq i \leq m+2n$, the preference list of u_i is as given on the right (these correspond to *negative* clauses D_i):

$$u_i: \quad b_{i1} \succ b_{i2} \succ b_{i3} \succ \underline{v_i} \qquad \text{or} \qquad u_i: \quad \underline{v_i} \succ d_{i1} \succ d_{i2},$$

where b_{ij} (similarly, d_{ij}) is the b-vertex (resp., d-vertex) that appears in the gadget corresponding to the j-th literal in the i-th clause. If the i-th clause (a positive one) consists of only 2 literals then there is no b_{i3} here. The vertex v_i is underlined.

The preference list of v_{m+2n} is $u_{m+2n} \succ t$ and for $1 \leq i \leq m+n$, the preference list of v_{i-1} is as given on the left (these correspond to *positive* clauses C_i) and for $m+n+1 \leq i \leq m+2n$, the preference list of v_{i-1} is as given on the right (these correspond to *negative* clauses D_i):

$$v_{i-1}: \quad \underline{u_{i-1}} \succ a_{i1} \succ a_{i2} \succ a_{i3} \qquad \text{or} \qquad v_{i-1}: \quad c_{i1} \succ c_{i2} \succ \underline{u_{i-1}},$$

where a_{ij} (similarly, c_{ij}) is the a-vertex (resp., c-vertex) that appears in the gadget corresponding to the j-th literal in the i-th clause. If the i-th clause (a positive one) consists of only 2 literals then there is no a_{i3} here. The vertex u_{i-1} is underlined.

4.2 Proof of Lemma 4

We first derive some useful properties. Let S be the matching given by the union of the basic set and the pair of dashed blue edges from the gadget of every literal. One easily checks that S is stable. Thus s and t are unstable vertices, i.e., they remain unmatched in any stable matching. Suppose an edge e is labeled $(-,-)$ with respect to some stable matching. It is an easy fact [14] that e is an *unstable edge*, i.e., no stable matching contains e. The following lemma is based on this fact.

Lemma 6. *Let S be any stable matching in $G, \mathcal{P}$. Then (i) S does not contain any consistency edge and (ii) S is a superset of the basic set.*

Proof. Consider the matching N given by the union of the basic set, the pair of dashed blue edges from the gadget of every positive literal, and the pair of dotted red edges from the gadget of every negative literal. It is easy to see that N is a stable matching. Every consistency edge is labeled $(-,-)$ with respect to N. Thus every consistency edge is unstable, proving (i).

Observe that every edge (v_{i-1}, a_i) is labeled $(-,-)$ with respect to N. Similarly, every edge (d_k, u_k) is also labeled $(-,-)$ with respect to N. Thus no stable matching can contain these edges. Recall that all the u-vertices and v-vertices are stable, so this immediately implies that v_{i-1} is matched to u_{i-1} for all $i \leq m+n$ and u_k is matched v_k for all $k \geq m+n+1$. Also recall that the 4 vertices in each literal gadget are stable. This implies u_{m+n} and v_{m+n} are also matched to each other in every stable matching, proving (ii). □

We can now conclude the proof of Lemma 4. Let S be any stable matching in G. Statement 1 follows from Lemma 6. Statements 2 and 3 are shown by statement 1 and stability. As for statement 4, if the dashed blue pair of edges from $\neg x$'s gadget is included in S then the stability of S implies that S has to contain the dashed blue pair of edges from every gadget of x—otherwise some consistency edge would be a *blocking edge* to S. Finally, it is easy to see that any matching that satisfies statements 1-4 is stable.

4.3 Proof of Lemma 5

We first show the second part of Lemma 5. Let S be a stable matching of G. If there is a gadget in true state in each clause, then an M-augmenting path starting at s and ending at t in G_M is easily constructed as follows. First, take edge (s, u_0), then all edges (u_i, v_i) and:

- for each clause C_i with gadget Z set to true, edges $(v_{i-1}, a_i), (a_i, b'_i), (b'_i, a'_i), (a'_i, b_i), (b_i, u_i)$;

– for each clause D_k with gadget Z set to true, edges $(v_{k-1}, c_k), (c_k, d_k), (d_k, u_k)$.

Last, take edge (v_{m+2n}, t). In order to prove the other direction, we start with some auxiliary facts.

Claim 3 *All popular matchings in G have the same size and match the same set of vertices. In particular, they leave unmatched only s and t.*

Proof. Consider the matching M given by the union of the basic set with the pair of dashed blue edges from the gadget of every literal. Note that M is a stable matching in G. We claim that M is also a dominant matching in G. Since a dominant (resp. stable) matching is a popular matching of maximum (resp. minimum) size, the first claim follows. The second is then implied by Lemma 1. Consider the edge (v_0, a_1): this is labeled $(-, -)$ with respect to M since both v_0 and a_1 prefer their partners in M to each other. Thus there is no s-t path in the graph G_M. As s and t are the only unmatched vertices, there is no M-augmenting path in G_M. So M is a dominant matching in G by Theorem 2. □

Claim 4 *Let S be a stable matching in G and ρ an augmenting path in G_S. Then ρ goes from s to t and does not use any consistency edge.*

Proof. By Claim 3, stable matchings in G leave only 2 vertices unmatched: these are s and t. So ρ must go from s to t. Recall that consistency edges cannot be included in S, see Lemma 4. By parity reasons, traversing ρ from s to t, a consistency edge can only occur in ρ if it leads ρ back to an earlier clause. That is, a consistency edge (a_i, d_k) has to be traversed in ρ in the direction $d_k \to a_i$. Let $e = (d_k, a_i)$ be the first consistency edge traversed in ρ. Observe that ρ cannot reach t, because nodes v_{i-1} and u_i must have already been traversed by ρ. Hence, consistency edges cannot appear in ρ. □

We can now conclude the proof of Lemma 5. Assume that there is an augmenting path ρ in G_S, starting at s and terminating at t. Because of Claim 5, ρ goes from s to t, traverses all clauses, and for each clause i, there is a path in $G_{\phi,S}$ between v_{i-1} and u_i. The literal whose gadget provides this $v_{i-1} \to u_i$ path is set to true and thus the i-th clause is satisfied. As this holds for each i, this means that ϕ has a satisfying assignment.

4.4 Max-size popular matchings

A non-dominant popular matching trivially exists if the size of a stable matching differs from the size of a dominant matching in an instance. Our next result is tailored for such instances. We will now show that it is NP-hard to decide if $G, \mathcal{P}$ admits a *max-size* popular matching that is not dominant.

Problem 3 *Input: A bipartite graph $G = (A \cup B, E)$ with strict preference lists.*
Decide: If there is a non-dominant matching among max-size popular matchings in G.

Given a 3SAT formula ϕ, we will transform it as described earlier and build the graph G_ϕ. Recall that all popular matchings in G_ϕ have the same size. We proved in Theorem 5 that it is NP-hard to decide if G_ϕ admits a popular matching that is not dominant. Consider the graph $H = G_\phi \cup \rho$ where ρ is the path $\langle p_0, q_0, p_1, q_1 \rangle$ with p_1 and q_0 being each other's top choices. There are no edges between a vertex in ρ and a vertex in G_ϕ. A max-size popular matching in H consists of the pair of edges $(p_0, q_0), (p_1, q_1)$, and a popular matching in G_ϕ.

Hence a max-size popular matching in H that is *not* dominant consists of $(p_0, q_0), (p_1, q_1)$, and a popular matching in G_ϕ that is non-dominant. Thus the desired result immediately follows.

Theorem 6. *Given $G = (A \cup B, E)$ and $\mathcal{P}$ where in $\mathcal{P}$, every vertex has a strict ranking over its neighbors, it is NP-complete to decide if G admits a max-size popular matching that is not dominant.*

5 Hardness of finding a stable matching that is dominant and consequences for popular matchings

Given $G = (A \cup B, E)$ and a set of strictly ordered lists $\mathcal{P}$, we first consider here the problem of deciding if G admits a matching that is *both* stable and dominant.

Problem 4 *Input: A bipartite graph $G = (A \cup B, E)$ with strict preference lists.*
Decide: If there is a stable matching in G that is also dominant.

In instances where all popular matchings have the same size, such a matching M is desirable as there are no blocking edges with respect to M; moreover, M defeats any larger matching in a head-to-head election.

In Section 5.1, we show that the above problem is NP-complete. We will then use this hardness to show the NP-completeness of deciding if G admits a min-size popular matching that is unstable and also to give a short proof of NP-hardness of the popular roommates problem.

5.1 Finding a matching that is both stable and dominant

We know from Theorem 2 that Problem 4 is equivalent to the problem of deciding if there exists a stable matching M such that M has no augmenting path in G_M. We will show a simple reduction from 3SAT to this stable matching problem. Note that the graph G here (and hence the reduction) is different from the instance given in Section 4. Given a 3SAT formula ϕ, we will transform ϕ as done in Section 4 so that there is exactly one occurrence of $\neg x$ in ϕ for every variable x.

Corresponding to the above formula ϕ, we will construct an instance $G, \mathcal{P}$ whose high-level picture is shown in Fig. 9. There are two "unwanted" vertices s, t along with u_i^ℓ, v_i^ℓ for every clause ℓ in ϕ, where i is the number of literals in clause ℓ, along with one gadget for each literal in every clause.

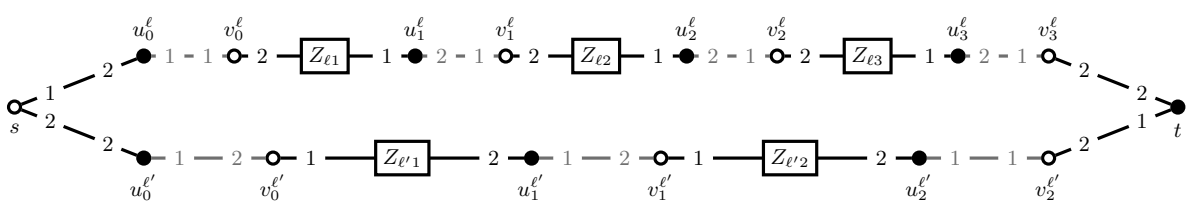

Fig. 9. The high-level picture of the gadgets corresponding to clauses ℓ with three literals, and ℓ' with two negated literals in the instance $G, \mathcal{P}$. The vertices s and t are common to all clauses.

In Fig. 9, $Z_{\ell j}$ is the gadget corresponding to the j-th literal of clause ℓ, where each clause has 2 or 3 literals. As before, we have a separate gadget for every occurrence of each literal. That is, there is a separate gadget for each occurrence of $x \in \{X_1, \ldots, X_{2n}\}$ in ϕ and another gadget for the unique occurrence of $\neg x$ in ϕ.

As done in Section 4, vertices from a gadget corresponding to x are denoted by a_i, a_i', b_i, b_i', while vertices from a gadget corresponding to $\neg x$ are denoted by c_k, c_k', d_k, d_k'; however adjacency lists and preferences are different here, see again Fig 9. As before, we have a *consistency edge* between x's gadget and $\neg x$'s gadget: this is now between the b-vertex of x's gadget and c-vertex of $\neg x$'s gadget.

Again, we postpone the definition of preference lists, and instead state the following lemma, analogous to Lemma 4. The proof of Lemma 7 is analogous to the proof of Lemma 4 and hence omitted.

Lemma 7. *Let S be any stable matching in $G, \mathcal{P}$. Then*

1. *S leaves s and t unmatched and it does not contain any consistency edge, while the edges (u_i^ℓ, v_i^ℓ) for all clauses ℓ and indices i are in S.*

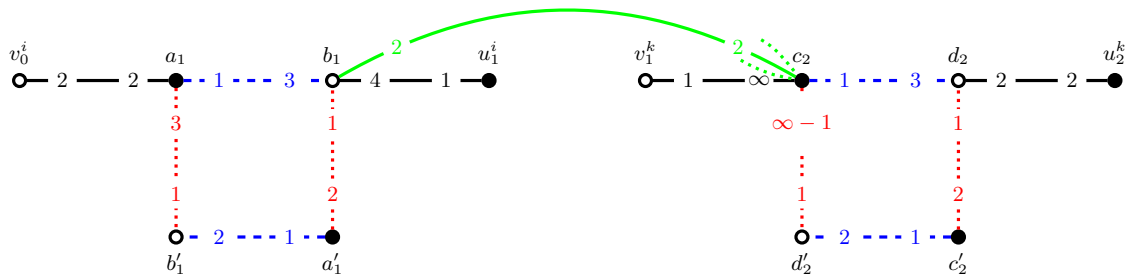

Fig. 10. Suppose x occurs as the first literal in the i-th clause and $\neg x$ occurs as the second literal in the k-th clause. For convenience, we use a_1, b_1, a_1', b_1' to denote the 4 vertices in this gadget of x and c_2, d_2, c_2', d_2' to denote the 4 vertices in the gadget of $\neg x$. As before, ∞ and $\infty - 1$ denote last choice neighbor and last but one choice neighbor, respectively.

Also, for every variable $x \in \{X_1, \ldots, X_{2n}\}$, we have:

2. *From the gadget of $\neg x$, either the pair (i) $(c_k, d_k), (c_k', d_k')$ or (ii) $(c_k, d_k'), (c_k', d_k)$ is in S.*
   — If (i) happens, we say that the gadget is in false state*, else that it is in* true state*.*
3. *From a gadget of x, say, its gadget in the i-th clause, either the pair of edges (i) $(a_i, b_i), (a_i', b_i')$ or (ii) $(a_i, b_i'), (a_i', b_i)$ is in S.*
   — If (i) happens, we say that the gadget is in true state*, else that it is in* false state*.*
4. *If the gadget of $\neg x$ is in true state then all the gadgets of x are in false state.*

Conversely, any matching S of G that satisfies 1-4 above is stable.

Corresponding to any stable matching M, define a true/false assignment A_M as follows:

– If the dashed blue pair (c_k, d_k) and (c_k', d_k') from $\neg x$'s gadget belongs to M then A_M sets x to true; else A_M sets x to false.

Conversely, for any true/false assignment A to the variables in ϕ, we define an associated matching M_A as follows. For any $x \in \{X_1, \ldots, X_{2n}\}$:

– If $x =$ true, then set every gadget corresponding to x in true state, and the gadget corresponding to $\neg x$ in false state. Thus the *dashed blue* edges (see Fig. 10) from all these gadgets get added to M_A.
– Otherwise, set every gadget corresponding to x in false state, and the gadget corresponding to $\neg x$ in true state. Thus the *dotted red* edges (see Fig. 10) from all these gadgets get added to M_A.

Finally, include the edges (u_i^ℓ, v_i^ℓ) for all ℓ and i. Using Lemma 7, it is easy to see that M_A is a stable matching.

Our goal is to show that ϕ is satisfiable if and only if G has a stable matching S without an S-augmenting path in G_S. This is achieved by the following lemma, whose proof is given at the end of this section.

Lemma 8. *If there is a stable matching M such that G_M has no augmenting path, then A_M satisfies ϕ. Conversely, if there exists a satisfiable assignment A, then there is no augmenting path with respect to M_A in G_{M_A}.*

We can therefore conclude the following.

Theorem 7. *Given $G = (A \cup B, E), \mathcal{P}$, it is NP-complete to decide if G admits a matching that is both stable and dominant.*

Preference lists We now describe the preference lists of the 4 vertices a_1, b_1, a'_1, b'_1 in the gadget of x in the i-th clause. We assume x to be the first literal in this clause. Note that the consistency edge now exists between the b-vertex in x's gadget and the c-vertex in $\neg x$'s gadget (see Fig. 10).

$$a_1 : \ b_1 \succ v_0^i \succ b'_1 \qquad\qquad a'_1 : \ b'_1 \succ b_1$$
$$b_1 : \ a'_1 \succ c_k \succ a_1 \succ u_1^i \qquad\qquad b'_1 : \ a_1 \succ a'_1$$

We now describe the preference lists of the 4 vertices c_2, d_2, c'_2, d'_2 in the gadget of $\neg x$ (see Fig. 10). We assume $\neg x$ to be the second literal in this clause (let this be the k-th clause).

$$c_2 : \ d_2 \succ b_i \succ b_j \succ \cdots \succ d'_2 \succ v_1^k \qquad\qquad c'_2 : \ d'_2 \succ d_2$$
$$d_2 : \ c'_2 \succ u_2^k \succ c_2 \qquad\qquad d'_2 : \ c_2 \succ c'_2$$

Here $b_i, b_j, \ldots$ are the occurrences of the b-vertex in all the gadgets corresponding to literal x in the formula ϕ. The order among these vertices in c_2's preference list is not important.

We now describe the preference lists of vertices u_i^ℓ and v_i^ℓ for $0 \le i \le r$ in clause ℓ, where $r \in \{2, 3\}$ is the number of literals in this clause. The preference list of u_0^ℓ is $v_0^\ell \succ s$ and the preference list of v_r^ℓ is $u_r^\ell \succ t$.

Let $1 \le j \le r$ and let a_j, b_j, a'_j, b'_j be the 4 vertices in the gadget of the j-th literal in this clause. The preference list of u_j^ℓ is as given on the left (resp. right) for positive (resp. negative) clauses.

$$u_j^\ell : \ \ b_j \succ v_j^\ell \qquad\qquad \text{or} \qquad\qquad u_j^\ell : \ \ v_j^\ell \succ d_j.$$

The preference list of v_{j-1}^ℓ is as given on the left (resp. right) for positive (resp. negative) clauses.

$$v_{j-1}^\ell : \ \ u_{j-1}^\ell \succ a_j \qquad\qquad \text{or} \qquad\qquad v_{j-1}^\ell : \ \ c_j \succ u_{j-1}^\ell.$$

Observe that s and t are the last choices for each of their neighbors. The preferences of s and t are not relevant and it is easy to see the following.

Claim 5 *Both s and t are unstable vertices.*

Proof of Lemma 8 Take any stable matching M. Suppose A_M does not satisfy, say, the r-th clause. We show that G_M has an augmenting path with respect to M, concluding the proof of one direction. Note that all gadgets in the r-th clause are in false state in M. Construct the augmenting path in G_M follows:

- Let the r-th clause be a positive clause $x \vee y \vee z$. Let a_1, b_1, a'_1, b'_1 be the 4 vertices in x's gadget, a_2, b_2, a'_2, b'_2 be the 4 vertices in y's gadget, and a_3, b_3, a'_3, b'_3 be the 4 vertices in z's gadget. Thus we have the following augmenting path with respect to M:

$$s-(u_0^r, v_0^r)-(a_1, b'_1)-(a'_1, b_1)-(u_1^r, v_1^r)-(a_2, b'_2)-(a'_2, b_2)-(u_2^r, v_2^r)-(a_3, b'_3)-(a'_3, b_3)-(u_3^r, v_3^r)-t.$$

- Let the r-th clause be a negative clause $\neg x \vee \neg y$. So both x and y are set to true and M contains the dashed blue pair of edges from $\neg x$'s gadget and also from $\neg y$'s gadget.
  Let c_1, d_1, c'_1, d'_1 be the 4 vertices in $\neg x$'s gadget and let c_2, d_2, c'_2, d'_2 be the 4 vertices in $\neg y$'s gadget. Thus we have the following augmenting path with respect to M:

$$s - (u_0^r, v_0^r) - (c_1, d_1) - (u_1^r, v_1^r) - (c_2, d_2) - (u_2^r, v_2^r) - t.$$

Conversely, assume that ϕ has a satisfiable assignment A. We show that there is no M_A-augmenting path in G_{M_A}. Let ℓ be any positive clause. Suppose A sets the j-th literal in this clause to true. Let a_j, b_j, a'_j, b'_j be the 4 vertices in the gadget corresponding to the j-th literal.

So $(a_j, b_j) \in M_A$ and hence there is no edge (v^ℓ_{j-1}, a_j) in G_{M_A}. Thus there is no alternating path between s and t such that all the intermediate vertices on this path correspond to the ℓ-th clause.

However there are consistency edges jumping across clauses—so there may be an M_A-augmenting path that begins with vertices corresponding to the ℓ-th clause and then uses a consistency edge. Let ρ be an M_A-alternating path in G_{M_A} with s as an endpoint and let (s, u^ℓ_0) be the first edge in ρ. So the prefix of ρ consists of vertices that belong to the ℓ-th clause.

Let b_i be the last vertex of the ℓ-th clause in this prefix of ρ. Let a_i, b_i, a'_i, b'_i be the 4 vertices in the gadget of b_i (let this correspond to variable x) in the ℓ-th clause. Observe that (a_i, b'_i) and (a'_i, b_i) are in M_A—otherwise b_i is not reachable from u^ℓ_0 in G_{M_A} via a path of vertices in the ℓ-th clause.

Assume the consistency edge (b_i, c_k) belongs to ρ, where c_k is a vertex in the gadget of $\neg x$. Let c_k, d_k, c'_k, d'_k be the vertices in the gadget of $\neg x$ and suppose this literal occurs in the r-th clause. Since the dotted red pair in x's gadget in the ℓ-th clause is in M_A, the dotted red pair (c_k, d'_k) and (c'_k, d_k) has to be in M_A. Since c'_k and d'_k are degree 2 vertices, ρ has to contain the subpath $c_k - d'_k - c'_k - d_k$. However ρ is now stuck at d_k in the graph G_{M_A}. The path cannot go back to c_k. There is no edge between d_k and u^r_k in G_{M_A} since this is a $(-,-)$ edge as both d_k and u^r_k prefer their respective partners in M_A to each other. Thus the alternating path ρ has to terminate at d_k.

The case when ℓ is a negative clause is even simpler since in this case ρ cannot leave the vertices of the ℓ-th clause using a consistency edge. We know that the assignment A sets some literal in the ℓ-th clause to true: let this be the k-th literal, so $(c'_k, d_k) \in M_A$ and hence there is no edge (d_k, u^ℓ_k) in G_{M_A}. Thus there is no M_A-augmenting path in G_{M_A} in this case also, concluding the proof.

5.2 Min-size popular matchings

In this section we investigate the counterpart of Theorem 6, i.e., the complexity of determining if $G = (A \cup B, E)$ admits a *min-size* popular matching that is not stable.

Problem 5 *Input: A bipartite graph $G = (A \cup B, E)$ with strict preference lists.*
Decide: If there is an unstable matching among min-size popular matchings in G.

Given a 3SAT formula ϕ, we transform it as described in Section 4 and build the graph G as described in Section 5.1. We now augment the bipartite graph G into bipartite graph H as follows:

- Add a new vertex w, which is adjacent to each d'-vertex in $\neg x$'s gadget for every variable $x \in \{X_1, \ldots, X_{2n}\}$.
- Add a square $\langle t, t', r', r\rangle$ at the t-end of the graph (see Fig. 11), where t', r', r are new vertices.

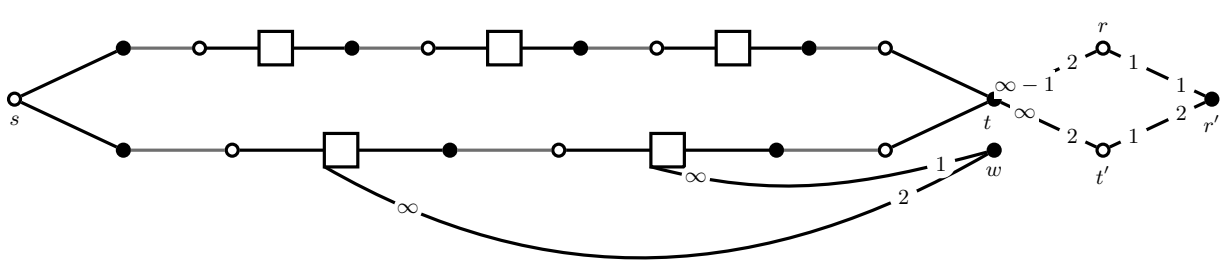

Fig. 11. We add a square $\langle t, t', r', r\rangle$ at the end of the graph G, and a vertex w as shown in the figure above. Recall that each s-t path in G corresponds to a clause in ϕ.

The preference lists of the vertices in $\{r, r', t', t\}$ are as follows:

$$r : r' \succ t \qquad r' : r \succ t' \qquad t' : r' \succ t \qquad t : \cdots \succ r \succ t'.$$

Recall that the vertex t is adjacent to the last v^ℓ-vertex in every clause gadget ℓ. The vertex t prefers all its v-neighbors (the order among these is not important) to its neighbors in the square which are r and t' and t prefers r to t'.

Regarding the vertex w, this vertex is the last choice for all its neighbors and w's preference list is some permutation of its neighbors (the order among these neighbors is not important). This finishes the description of the graph H. We denote the collection of all preference lists again by $\mathcal{P}$. The proof of Lemma 9 is analogous to the proof of Lemma 4.

Lemma 9. *Let S be any stable matching in $H, \mathcal{P}$. Then S contains the edges (u_i^ℓ, v_i^ℓ) for all clauses ℓ and all i and the pair of edges $(r, r'), (t, t')$. Moreover, S does not contain any consistency edge.*

Corollary 1. *For any stable matching S in H, P and any variable $x \in \{X_1, \ldots, X_{2n}\}$, we have:*

1. *From the gadget of $\neg x$, either the pair (i) $(c_k, d_k), (c'_k, d'_k)$ or (ii) $(c_k, d'_k), (c'_k, d_k)$ is in S.*
2. *From a gadget of x, say its gadget in the i-th clause, either the pair of edges (i) $(a_i, b_i), (a'_i, b'_i)$ or (ii) $(a_i, b'_i), (a'_i, b_i)$ is in S.*

It follows from Lemma 9 and Corollary 1 that a stable matching in H matches all vertices except s and w. A max-size popular matching in H is a perfect matching: it includes the edges $(s, u_0^\ell), (v_0^\ell, c_k), (d'_k, w)$, (d_k, c'_k) for some negative clause ℓ. We now prove the following theorem.

Theorem 8. *$H, \mathcal{P}$ admits an unstable min-size popular matching if and only if G admits a stable matching that is dominant.*

Proof. Suppose G admits a stable matching N that is also dominant. We claim that $M = N \cup \{(r, t), (r', t')\}$ is a popular matching in H. Note that there is a blocking edge (r, r') with respect to M. Since M matches exactly the stable vertices, this would make M an unstable min-size popular matching.

Recall Theorem 1 (from Section 2) that characterizes popular matchings. We will now show that the matching M satisfies the three sufficient conditions for popularity as given in Theorem 1. Note that (r, r') is the only blocking edge with respect to M. Thus property (ii) from Theorem 1 obviously holds. Property (i) holds since the edge (t, t') is labeled $(-, -)$ with respect to M. Thus there is no alternating cycle in H_M with the edge (r, r'). We will now show property (iii) also holds in H_M, thus M is a popular matching in H.

There are two unmatched vertices in M: s and w. We need to check that the edge (r, r') is not reachable via an M-alternating path from either s or w in H_M. Since the matching N is dominant in G, there is no M-alternating path between s and t in H_M. Thus the blocking edge (r, r') is not reachable from s by an M-alternating path in H_M.

Consider the vertex w and any of its neighbors, say d'_k (see Fig. 10). We know that N includes either the dotted red pair of edges $(c_k, d'_k), (c'_k, d_k)$ or the dashed blue pair of edges $(c_k, d_k), (c'_k, d'_k)$. In both cases, the blocking edge (r, r') is not reachable in H_M by an M-alternating path with (w, d'_k) as a starting edge. This proves one side of the reduction.

We will now show the converse. Let M be a min-size popular matching in H that is unstable. Since M is a min-size popular matching, the set of vertices matched in M is the set of stable vertices. Consider the edges $(v_{i-1}^\ell, a_i), (d_k, u_k^\ell)$, and (b_i, c_k) for any ℓ, i, and k. For each of those edges, there is a stable matching in G (and thus in H) where all these edges are labeled $(-, -)$ (see the proof of Lemma 4). Thus these edges are *slack*[6] with respect to a popular matching and its witness, i.e. $\mathbf{0}$ here—so they are *unpopular* edges (see Section 2). Hence M has to contain the edges (u_i^ℓ, v_i^ℓ) for all clauses ℓ and all i.

The 4 vertices in a literal gadget are stable, hence matched among themselves in M (since they must all be matched, and we excluded other edges incident to them), i.e., either the dotted red pair or the dashed blue pair from each literal gadget belongs to M. Also, the consistency edge cannot be a *blocking edge* to M as this would make a blocking edge reachable via an M-alternating path in H_M from the unmatched vertex w, a contradiction to M's popularity in H (by Theorem 1).

[6] An edge (x, y) is slack with respect to a popular matching N and its witness $\boldsymbol{\alpha}$ if $\alpha_x + \alpha_y > \mathsf{wt}_N(x, y)$.

Since M is unstable, there is a blocking edge with respect to M. The only possibility is from within the square $\langle r, r', t', t\rangle$. Thus M has to contain the pair of edges (r, t) and (r', t'): this makes (r, r') a blocking edge with respect to M. Consider the matching $N = M \setminus \{(r, t), (r', t')\}$. We have already argued that N is a stable matching in G.

Suppose N is not dominant in G. Then there is an N-alternating path between s and t in G_N. Thus in the graph H_M, there is an M-alternating path from the unmatched vertex s to the blocking edge (r, r'). This contradicts the popularity of M in H by Theorem 1. Hence N is a dominant matching in G. □

We proved in Section 5.1 that the problem of deciding if G admits a stable matching that is dominant is NP-hard. Thus we can conclude the following theorem.

Theorem 9. *Given $G = (A \cup B, E)$ and $\mathcal{P}$, it is NP-complete to decide if G admits a min-size popular matching that is not stable.*

Our graph H is such that every popular matching here has size either $n/2$ or $n/2 - 1$, where n is the number of vertices in H. All popular matchings of size $n/2$ are dominant since they are perfect matchings. Thus a popular matching M in H is neither stable nor dominant if and only if M is an unstable popular matching of size $n/2 - 1$, i.e., M is an unstable min-size popular matching in H. So we have shown a new and simple proof of NP-hardness of the problem of deciding if a marriage instance admits a popular matching that is neither stable nor dominant.

5.3 A simple proof of NP-hardness of the popular roommates problem

We know that the popular roommates problem is NP-hard [8, 13]. Here we adapt the hardness reduction given in Section 5.1 to show a short and simple proof of hardness of this problem. We first mention some useful structural results from Section 2 that extend to the roommates case.

The first is the *Rural Hospitals Theorem*, see [14, Theorem 4.5.2]). That is, if $H = (V, E)$ admits stable matchings, then all stable matchings in H have to match the same subset of vertices. Another useful property is that if an edge e is labeled $(-, -)$ with respect to a stable matching in H then no popular matching in H can include e [22]. Last, we recall that the characterization of popular matchings given in Theorem 1 also holds when G is non-bipartite.

Problem 6 *Input: A (non-bipartite) graph $H = (V, E)$ with strict preference lists.*
Decide: If there is a popular matching in H.

Given a 3SAT formula ϕ, we transform it as described in Section 4 and build the graph G as described in Section 5.1. We now augment the bipartite graph G into a non-bipartite graph H, depicted in Fig. 12 as follows:

- Add edges between s and the d'-vertex in $\neg x$'s gadget for every variable x.
- At the other end of the graph G, add an edge (t, r) along with a triangle $\langle r, r', r''\rangle$, where r, r', r'' are new vertices.

The preference lists of the vertices in $\{r, r', r'', t\}$ are as follows:

$$r : r' \succ r'' \succ t \qquad r' : r'' \succ r \qquad r'' : r \succ r' \qquad t : \cdots \succ r.$$

The vertex t is adjacent to the last v^ℓ-vertex in every clause gadget ℓ. The vertex t prefers all its v-neighbors (the order among these is not important) to r. We denote the collection of all preference lists again by $\mathcal{P}$.

The vertex s is the last choice neighbor for all its neighbors. The preference list of s is not relevant. Recall the vertex w used in Section 5.2: now we have merged the vertices s and w. This creates odd cycles, which is allowed here since the graph H is non-bipartite.

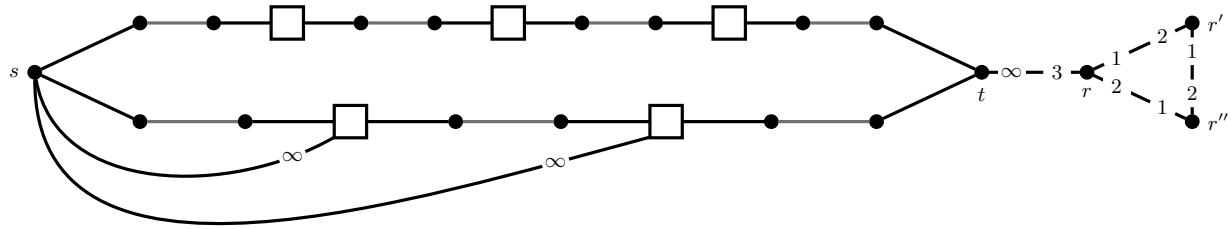

Fig. 12. We add a new triangle $\langle r, r', r'' \rangle$ to t, and connect s with all d' vertices as shown in the figure above.

Popular matchings in H Let M be a popular matching in H. Observe that M has to match r, r', and r'' since each of these vertices is a top choice neighbor for some vertex. If one of these vertices is left unmatched in M then there would be a blocking edge incident to an unmatched vertex, a contradiction to its popularity (see Theorem 1, condition (iii)). Since t is the only *outside* neighbor of r, r', r'', the matching M has to contain the pair of edges $(t, r), (r', r'')$. Note that the edge (r, r'') blocks M.

Let $H^* = H \setminus \{t, r, r', r''\}$. Consider the matching $N = M \setminus \{(t, r), (r', r'')\}$ in H^*. Since M is popular in H, the matching N has to be popular in H^*. We claim N has to match all vertices in H^* except s. This is because H^* admits a stable matching: consider $S = \cup_{\ell,i}\{(u_i^\ell, v_i^\ell)\} \cup \{$dashed blue edges in every literal gadget$\}$—this is a stable matching in H^*. We know that N has to match all stable vertices [16]. Since the number of vertices in H^* is odd, the vertex s is left unmatched in N.

Consider any consistency edge. We claim this is an *unpopular* edge in H^*. This is because there is a stable matching in H^* where this edge is labeled $(-, -)$ (see the proof of Lemma 4), hence this edge cannot be used in any popular matching in H^* [22]. Since all vertices in H^* except s have to be matched in N, the following 3 observations hold:

1. N contains the edges (u_i^ℓ, v_i^ℓ) for all clauses ℓ and all i.
2. From the gadget of $\neg x$, either the pair (i) $(c_k, d_k), (c'_k, d'_k)$ or (ii) $(c_k, d'_k), (c'_k, d_k)$ is in N.
3. From a gadget of x, say, its gadget in the i-th clause, either the pair of edges (i) $(a_i, b_i), (a'_i, b'_i)$ or (ii) $(a_i, b'_i), (a'_i, b_i)$ is in N.

We are now ready to prove Theorem 10, whose proof is similar to the proof of Theorem 8.

Theorem 10. *$H, \mathcal{P}$ admits a popular matching if and only if G admits a stable matching that is dominant.*

Proof. Suppose G admits a stable matching S that is also dominant. We claim that $M = S \cup \{(t, r), (r', r'')\}$ is a popular matching in H. We will again use Theorem 1 to prove the popularity of M in H. There is exactly one blocking edge with respect to M: this is the edge (r, r'').

Observe that properties (i) and (ii) from Theorem 1 are easily seen to hold. We will now show that property (iii) from Theorem 1 also holds. We need to check that the edge (r, r'') is not reachable via an M-alternating path from s in H_M.

Since the matching S is dominant in G, there is no S-alternating path between s and t in G_S. Thus the blocking edge (r, r'') is not reachable from s by an M-alternating path in G_M. We now need to show that the blocking edge (r, r'') is not reachable from s by an M-alternating path in H_M, i.e., when the first edge of the alternating path is (s, d'_k) for some d'_k. We know that M includes either the dotted red pair of edges $(c_k, d'_k), (c'_k, d_k)$ or the dashed blue pair of edges $(c_k, d_k), (c'_k, d'_k)$ from this gadget. In both cases, it can be checked that the blocking edge (r, r') is not reachable in H_M by an M-alternating path with (s, d'_k) as the starting edge. This proves one side of the reduction.

We will now show the converse. Suppose H admits a popular matching M. We argued above that the pair of edges (t, r) and (r', r'') is in M. Consider the matching $N = M \setminus \{(t, r), (r', r'')\}$. We claim that N is a stable matching in G.

From observations 1-3 given above, it follows that the only possibility of a blocking edge to N is from a consistency edge. However a consistency edge cannot block N as this would make a blocking edge reachable by an M-alternating path in H_M from the unmatched vertex s and this contradicts M's popularity (by Theorem 1).

We now claim that N is a dominant matching in G. Suppose not. Then there is an N-alternating path between s and t in G_N. Thus in the graph H_M, there is an M-alternating path from the unmatched vertex s to the blocking edge (r, r''). This contradicts the popularity of M in H by Theorem 1. Hence N is a dominant matching in G. □

We know that the problem of deciding if G admits a stable matching that is dominant is NP-hard. This completes our new proof of the NP-hardness of the popular roommates problem. Observe that every popular matching in H is a max-size matching (since only the vertex s is left unmatched), hence every popular matching in H is dominant. Thus our reduction above also shows a simple proof of NP-hardness of the dominant roommates problem.

Conclusions and an open problem We considered popular matching problems in a bipartite graph $G = (A \cup B, E)$ with strict preferences. We showed a simple $O(|E|^2)$ algorithm for deciding if there exists a popular matching in G that is *not* stable. An open problem is to improve the running time of this algorithm. We showed that the problems of deciding if a bipartite graph admits a stable matching that is (i) dominant, (ii) *not* dominant are NP-hard. These results imply many new hardness results for popular matchings in bipartite graphs, including the hardness of finding (1) a popular matching that is *not* dominant, (2) a min-size popular matching that is *not* stable, and (3) a max-size popular matching that is *not* dominant.

References

1. D. J. Abraham, R. W. Irving, T. Kavitha, and K. Mehlhorn. Popular matchings. *SIAM Journal on Computing*, 37:1030–1045, 2007.
2. P. Biró, R. W. Irving, and D. F. Manlove. Popular matchings in the marriage and roommates problems. In *Proceedings of CIAC '10: the 7th International Conference on Algorithms and Complexity*, volume 6078 of *Lecture Notes in Computer Science*, pages 97–108, 2010.
3. M. Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. L'Imprimerie Royale, 1785.
4. Condorcet method. `https://en.wikipedia.org/wiki/Condorcet_method`.
5. Á. Cseh, C.-C. Huang, and T. Kavitha. Popular matchings with two-sided preferences and one-sided ties. *SIAM Journal on Discrete Mathematics*, 31(4):2348–2377, 2017.
6. Á. Cseh and T. Kavitha. Popular edges and dominant matchings. *Mathematical Programming*, 172(1):209–229, Nov 2018.
7. Y. Faenza and T. Kavitha. Quasi-popular matchings, optimality, and extended formulations. In *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms*, 2020.
8. Y. Faenza, T. Kavitha, V. Powers, and X. Zhang. Popular matchings and limits to tractability. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2790–2809. SIAM, 2019.
9. T. Feder. A new fixed point approach for stable networks and stable marriages. *Journal of Computer and System Sciences*, 45:233–284, 1992.
10. T. Feder. Network flow and 2-satisfiability. *Algorithmica*, 11:291–319, 1994.
11. D. Gale and L. S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.
12. P. Gärdenfors. Match making: assignments based on bilateral preferences. *Behavioural Science*, 20:166–173, 1975.
13. S. Gupta, P. Misra, S. Saurabh, and M. Zehavi. Popular matching in roommates setting is NP-hard. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2810–2822. SIAM, 2019.
14. D. Gusfield and R. W. Irving. *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, 1989.

15. M. Hirakawa, Y. Yamauchi, S. Kijima, and M. Yamashita. *On The Structure of Popular Matchings in The Stable Marriage Problem - Who Can Join a Popular Matching?* In the 3rd International Workshop on Matching Under Preferences (MATCH-UP), 2015.
16. C.-C. Huang and T. Kavitha. Popular matchings in the stable marriage problem. *Information and Computation*, 222:180–194, 2013.
17. C.-C. Huang and T. Kavitha. Popularity, mixed matchings, and self-duality. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2294–2310. SIAM, 2017.
18. R. Irving. An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms*, 6:577–595, 1985.
19. R. W. Irving, P. Leather, and D. Gusfield. An efficient algorithm for the "optimal" stable marriage. *Journal of the ACM*, 34:532–543, 1987.
20. T. Kavitha. A size-popularity tradeoff in the stable marriage problem. *SIAM Journal on Computing*, 43:52–71, 2014.
21. T. Kavitha. Popular half-integral matchings. In *the 43rd International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 22, pages 1–13, 2016.
22. T. Kavitha. Max-size popular matchings and extensions. *arXiv preprint arXiv:1802.07440*, 2018.
23. T. Kavitha, J. Mestre, and M. Nasre. Popular mixed matchings. *Theoretical Computer Science*, 412:2679–2690, 2011.
24. U. G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54:57–67, 1992.
25. A. Subramanian. A new approach to stable matching problems. *SIAM Journal on Computing*, 23(4):671–700, 1994.
26. C.-P. Teo and J. Sethuraman. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23:874–891, 1998.
27. J. H. Vande Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.