Popular Matchings in Complete Graphs

Ágnes Cseh[1]
Hungarian Academy of Sciences, Budapest, Hungary
cseh.agnes@krtk.mta.hu

Telikepalli Kavitha[2]
Tata Institute of Fundamental Research, Mumbai, India
kavitha@tcs.tifr.res.in

Abstract

Our input is a complete graph $G = (V, E)$ on n vertices where each vertex has a strict ranking of all other vertices in G. The goal is to construct a matching in G that is "globally stable" or *popular*. A matching M is popular if M does not lose a head-to-head election against any matching M': here each vertex casts a vote for the matching in $\{M, M'\}$ where it gets a better assignment. Popular matchings need not exist in the given instance G and the popular matching problem is to decide whether one exists or not. The popular matching problem in G is easy to solve for odd n. Surprisingly, the problem becomes NP-hard for even n, as we show here.

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis

Keywords and phrases popular matching, complete graph, complexity, linear programming

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2018.17

Related Version A full version of the paper is available at [9], `https://arxiv.org/pdf/1807.01112.pdf`.

Acknowledgements Thanks to Chien-Chung Huang for asking us about the complexity of the popular roommates problem with complete preference lists.

1 Introduction

Consider a complete graph $G = (V, E)$ on n vertices where each vertex ranks all other vertices in a strict order of preference. Such a graph is called a *roommates* instance with complete preferences. The problem of computing a stable matching in G is a classical and well-studied problem. Recall that a matching M is stable if there is no *blocking pair* with respect to M, i.e., a pair (u, v) where both u and v prefer each other to their respective assignments in M.

Stable matchings need not always exist in a roommates instance. For example, the instance given in Fig. 1 on 4 vertices d_0, d_1, d_2, d_3 has no stable matching. (Here d_0's top choice is d_1, then d_2, and finally d_3, and so on.)

Irving [17] gave an efficient algorithm to decide if G admits a stable matching or not. In this paper we consider a notion that is more relaxed than stability: this is the notion of *popularity*. For any vertex u, a ranking over neighbors can be extended naturally to a

---

[1] Supported by the Cooperation of Excellences Grant (KEP-6/2017), by the Ministry of Human Resources under its New National Excellence Programme (ÚNKP-18-4-BME-331), the Hungarian Academy of Sciences under its Momentum Programme (LP2016-3/2016), its János Bolyai Research Fellowship, and OTKA grant K128611.

[2] This work was done while visiting the Hungarian Academy of Sciences, Budapest.

38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2018).
Editors: Sumit Ganguly and Paritosh Pandya; Article No. 17; pp. 17:1–17:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$$\begin{array}{lcl} d_0 & : & d_1 > d_2 > d_3 \\ d_1 & : & d_2 > d_3 > d_0 \\ d_2 & : & d_3 > d_1 > d_0 \\ d_3 & : & d_1 > d_2 > d_0 \end{array}$$

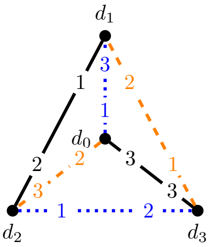

Figure 1 An instance that admits two popular matchings – marked by dotted blue and dashed orange edges – but no stable matching. The preference list of each vertex is illustrated by the numbers on its edges: a lower number indicates a more preferred neighbor.

ranking over matchings as follows: u prefers matching M to matching M' if (i) u is matched in M and unmatched in M' or (ii) u is matched in both and u prefers $M(u)$ to $M'(u)$. For any two matchings M and M', let $\phi(M, M')$ be the number of vertices that prefer M to M'.

▶ **Definition 1.** Let M be any matching in G. M is *popular* if $\phi(M, M') \geq \phi(M', M)$ for every matching M' in G.

Suppose an election is held between M and M' where each vertex casts a vote for the matching that it prefers. So $\phi(M, M')$ (similarly, $\phi(M', M)$) is the number of votes for M (resp., M'). A popular matching M never loses an election to another matching M' since $\phi(M, M') \geq \phi(M', M)$: thus it is a weak *Condorcet winner* [6, 1] in the corresponding voting instance. So popularity can be regarded as a natural notion of “global stability”.

The notion of popularity was first introduced in bipartite graphs in 1975 by Gärdenfors – popular matchings always exist in bipartite graphs since stable matchings always exist here [11] and every stable matching is popular [12]. The proof that every stable matching is popular holds in non-bipartite graphs as well [5]; in fact, it is easy to show that every stable matching is a min-size popular matching [14]. Relaxing the constraint of stability to popularity allows us to find globally stable matchings that may exist in instances that do not admit stable matchings; moreover, even when stable matchings exist, there may be popular matchings that achieve more “social good” (such as larger size) in many applications.

Observe that the instance in Fig. 1 has two popular matchings: $M_1 = \{(d_0, d_1), (d_2, d_3)\}$ and $M_2 = \{(d_0, d_2), (d_1, d_3)\}$. However as was the case with stable matchings, popular matchings also need not always exist in the given instance G. The *popular roommates problem* is to decide if G admits a popular matching or not. When the graph is not complete, it is known that the popular roommates problem is NP-hard [10, 13]. Here we are interested in the complexity of the popular matching problem when the input instance is complete.

Interestingly, several popular matching problems that are intractable in bipartite graphs become tractable in *complete bipartite* graphs. The min-cost popular matching problem in bipartite graphs is such a problem – this is NP-hard in a bipartite graph with incomplete lists [10], however it can be solved in polynomial time in a bipartite graph with complete lists [8]. The difference is due to the fact that while there is no efficient description of the convex hull of all popular matchings in a general bipartite graph, this polytope has a compact extended formulation in a complete bipartite graph.

It is a simple observation (see Section 2) that when n is *odd*, a matching in a complete graph G on n vertices is popular only if it is stable. Since there is an efficient algorithm to decide if G admits a stable matching or not, the popular roommates problem in a complete graph G can be efficiently solved when n is odd. We show the following result here.

▶ **Theorem 2.** *Let G be a complete graph on n vertices, where n is even. The problem of deciding whether G admits a popular matching or not is NP-hard.*

So the popular roommates problem with complete preference lists is NP-hard for even n while it is easy to solve for odd n. Note that the popular roommates problem is non-trivial for every $n \geq 5$, i.e., there are both "yes instances" and "no instances" of size n. It is rare and unusual for a natural decision problem in combinatorial optimization to be efficiently solvable when n has one parity and become NP-hard when n has the other parity. We are not aware of any natural optimization problem on graphs that is non-trivially tractable when the cardinality of the vertex set has one parity, which becomes intractable for the other parity.

1.1 Background and related work

The first polynomial time algorithm for the stable roommates problem was by Irving [17] in 1985. Roommates instances that admit stable matchings were characterized in [25]. New polynomial time algorithms for the stable roommates problem were given in [24, 26].

Algorithmic questions for popular matchings in bipartite graphs have been well-studied in the last decade [3, 8, 14, 16, 18, 19, 20]. Not much was known on popular matchings in non-bipartite graphs. Biró et al. [3] proved that validating whether a given matching is popular can be done in polynomial time, even when ties are present in the preference lists. It was shown in [15] that every roommates instance $G = (V, E)$ admits a matching with *unpopularity factor* $O(\log |V|)$ and that it is NP-hard to compute a least unpopularity factor matching. It was shown in [16] that computing a max-weight popular matching in a roommates instance with edge weights is NP-hard, and more recently, that computing a max-size popular matching in a roommates instance is NP-hard [21].

The complexity of the popular roommates problem was open for several years [3, 7, 15, 16, 22] and two independent NP-hardness proofs [10, 13] of this problem were announced very recently. Interestingly, both these hardness proofs need "incomplete preference lists", i.e., the underlying graph is *not* complete. The reduction in [13] is from a variant of the vertex cover problem called the *partitioned vertex cover* problem and we discuss the reduction in [10] in Section 1.2 below. So the complexity status of the popular roommates problem in a complete graph was an open problem and we resolve it here.

Computational hardness for instances with complete lists has been investigated in various matching problems under preferences. An example is the three-sided stable matching problem with cyclic preferences: this involves three groups of participants, say, men, women, and dogs, where dogs have weakly ordered preferences over men only, men have preferences over women only, and finally, women only list the dogs. If these preferences are allowed to be incomplete, the problem of finding a *weakly* stable matching is known to be NP-complete [4]. It is one of the most intriguing open questions in stable matchings [22, 27] as to whether the same problem becomes tractable when lists are complete.

1.2 Techniques

The 1-in-3 SAT problem is a well-known NP-hard problem [23]: it consists of a 3-SAT formula ϕ with no negated literals and the problem is to find a truth assignment to the variables in ϕ such that every clause has exactly one variable set to `true`. We show a polynomial time reduction from 1-in-3 SAT to the popular roommates problem with complete lists.

Our construction is based on the reduction in [10] that proved the NP-hardness of the popular roommates problem. However there are several differences between our reduction and the reduction in [10]. The reduction in [10] considered a popular matching problem in bipartite graphs called the "exclusive popular set" problem and showed it to be NP-hard – when preference lists are complete, this problem can be easily solved. Thus the reduction in [10] needs incomplete preference lists.

The exclusive popular set problem asks if there is a popular matching in the given bipartite graph where the set of matched vertices is S, for a given even-sized subset S. A key step in the reduction in [10] from this problem in bipartite graphs to the popular matching problem in non-bipartite graphs merges all vertices outside S into a single node. Thus the total number of vertices in the non-bipartite graph used in [10] is *odd*. Moreover, the fact that popular matchings always exist in bipartite graphs is crucially used in this reduction. However in our setting, the whole problem is to decide if *any* popular matching exists in the given graph – thus there are no popular matchings that "always exist" here.

The reduction in [10] primarily uses the LP framework of popular matchings in bipartite graphs from [18, 19, 21] to analyze the structure of popular matchings in their instance. The LP framework characterizing popular matchings in non-bipartite graphs is more complex [21], so we use the combinatorial characterization of popular matchings [14] in terms of forbidden alternating paths/cycles to show that any popular matching in our instance will yield a 1-in-3 satisfying assignment for ϕ. To show the converse, we use a dual certificate similar to the one used in [10] to prove the popularity of the matching that we construct using a 1-in-3 satisfying assignment for ϕ.

2 Preliminaries

Let M be any matching in $G = (V, E)$. For any pair $(u, v) \notin M$, define $\mathsf{vote}_u(v, M)$ as follows: (here $M(u)$ is u's partner in M and $M(u) = \mathsf{null}$ if u is unmatched in M)

$$\mathsf{vote}_u(v, M) = \begin{cases} + & \text{if } u \text{ prefers } v \text{ to } M(u); \\ - & \text{if } u \text{ prefers } M(u) \text{ to } v. \end{cases}$$

Label every edge (u, v) that does not belong to M by the pair $(\mathsf{vote}_u(v, M), \mathsf{vote}_v(u, M))$. Thus every non-matching edge has a label in $\{(\pm, \pm)\}$. Note that an edge is labeled $(+, +)$ if and only if it 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 in G.

▶ **Theorem 3** ([14]). *M is popular in G if and only if G_M does not contain any of the following with respect to M:*

(1) *an alternating cycle with a $(+, +)$ edge;*
(2) *an alternating path with two distinct $(+, +)$ edges;*
(3) *an alternating path with a $(+, +)$ edge and an unmatched vertex as an endpoint.*

Using the above characterization, it can be easily checked whether a given matching is popular or not [14]. Thus our NP-hardness result implies that the popular roommates problem is NP-complete.

When n is odd. Recall the claim made in Section 1 that when n is odd, every popular matching in G has to be stable. A simple proof of this statement is included below.

▶ **Observation 4** ([2]). *Let G be a complete graph on n vertices, where n is odd. Any popular matching in G has to be stable.*

x_i : $y_i > y'_i > D > \dots$
x'_i : $y_i > y'_i > D > \dots$
y_i : $x_i > x'_i > D > \dots$
y'_i : $x_i > x'_i > D > \dots$

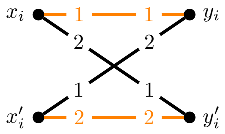

Figure 2 The variable gadget in level 1.

Proof. Suppose not. Let M be a popular matching in G that is not stable. So there is a blocking edge (u, v) to M. Because n is odd, we know that there is an unmatched vertex. If one of u, v is unmatched, then the edge (u, v) is a forbidden alternating path for popularity (by Theorem 3, part (3)). So let the unmatched vertex be $x \notin \{u, v\}$.

Then the path x - $(M(u), u)$ - $(v, M(v))$ with respect to M is again a forbidden alternating path for popularity (by Theorem 3, part (3)). Thus M is not a popular matching. ◀

3 The graph G

Recall that ϕ is the input formula to 1-in-3 SAT. The graph G that we construct here consists of gadgets in 4 levels along with 2 special gadgets that we will call the D-gadget and Z-gadget. Gadgets in level 1 correspond to variables in the formula ϕ while gadgets in levels 0, 2, and 3 correspond to clauses in ϕ. Variants of the gadgets in levels 0-3 and the D-gadget were used in [10] while the Z-gadget is new.

We will now describe these gadgets: along with a figure, we provide the preference lists of vertices in this gadget. The tail of each list consists of all vertices not listed yet, in an arbitrary order. Even though the preference lists are complete, the structure of the gadgets and the preference lists will ensure that inter-gadget edges will not belong to any popular matching, as we will show in Section 4.

The D-gadget. The D-gadget is on 4 vertices d_0, d_1, d_2, d_3 and the preference lists of these vertices are as given in Fig. 1 with all vertices outside the D-gadget at the tail of each list (in an arbitrary order). Recall that this gadget admits no stable matching.

We describe gadgets from level 1 first, then levels 0, 2, 3, and finally, the Z-gadget. The stable matchings within the gadgets are highlighted by colors in the figures. The gray elements in the preference lists denote vertices that are outside this gadget. We will assume that D in a preference list stands for $d_0 > d_1 > d_2 > d_3$.

Level 1. For each variable X_i in the formula ϕ, we construct a gadget on four vertices as shown in Fig. 2. The bottom vertices x'_i and y'_i will be preferred by some vertices in level 0 to vertices in their own gadget, while the top vertices x_i and y_i will be preferred by some vertices in level 2 to vertices in their own gadget. All four vertices in a level 1 gadget prefer to be matched among themselves, along the four edges drawn than be matched to any other vertex in the graph. This gadget has a unique stable matching $\{(x_i, y_i), (x'_i, y'_i)\}$.

Level 0. To each clause $c = X_i \vee X_j \vee X_k$ in the formula ϕ, we create *6 gadgets* in level 0. One of these can be seen in Fig. 3. The top two vertices, i.e. a^c_1 and b^c_1, rank y'_j and x'_k in level 1, as their respective second choices. Recall that indices j and k are well-defined in the

$$
\begin{aligned}
a_1 &: b_1 > y'_j > b_2 > D > \dots\\
a_2 &: b_2 > b_1 > D > \dots\\
b_1 &: a_2 > x'_k > a_1 > D > \dots\\
b_2 &: a_1 > a_2 > D > \dots
\end{aligned}
$$

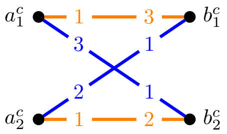

Figure 3 A clause gadget in level 0.

$$
\begin{aligned}
p_0 &: q_0 > q_2 > D > \dots\\
p_1 &: q_1 > q_2 > D > \dots\\
p_2 &: q_0 > y_j > q_1 > q_2 > D > \dots\\
q_0 &: p_0 > p_2 > D > \dots\\
q_1 &: p_1 > p_2 > D > \dots\\
q_2 &: p_1 > x_k > p_0 > p_2 > D > \dots
\end{aligned}
$$

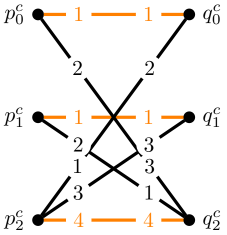

Figure 4 A clause gadget in level 2.

clause $c = X_i \vee X_j \vee X_k$. Within this level 0 gadget on $a_1^c, b_1^c, a_2^c, b_2^c$, both $\{(a_1^c, b_1^c), (a_2^c, b_2^c)\}$ and $\{(a_1^c, b_2^c), (a_2^c, b_1^c)\}$ are stable matchings. In the preference lists below (and also for gadgets in levels 2 and 3), we have omitted the superscript c in their lists for the sake of readability.

The gadget on vertices $\{a_3^c, a_4^c, b_3^c, b_4^c\}$ is built analogously: the vertex a_3^c ranks y'_k as its second choice, while b_3^c ranks x'_i second. In the third gadget, the vertex a_5^c ranks y'_i second, while b_5^c ranks x'_j second. Observe the shift in i, j, k indices as second choices for vertices a_1^c, a_3^c, a_5^c (and similarly, for b_1^c, b_3^c, b_5^c).

The fourth, fifth and sixth gadgets are analogous to the first, second, and third gadgets, respectively, but there is a slight twist. More precisely, the preferences of $a'^c_1, a'^c_2, b'^c_1, b'^c_2$ in the fourth gadget are analogous to the preferences in Fig. 3, except that a'^c_1 ranks y'_k second, while b'^c_1 ranks x'_j second. Similarly, the second choice of a'^c_3 is y'_i, the second choice of b'^c_3 is x'_k, and finally, a'^c_5 ranks y'_j second, while b'^c_5 ranks x'_i second. Observe the change in *orientation* of the indices i, j, k as second choice neighbors when comparing the first three level 0 gadgets of c with its last three level 0 gadgets. This will be important to us later.

Level 2. To each clause $c = X_i \vee X_j \vee X_k$ in the formula ϕ, we create *6 gadgets* in level 2. The first gadget in level 2 is on vertices $p_0^c, p_1^c, p_2^c, q_0^c, q_1^c, q_2^c$ and their preference lists are described in Fig. 4. Note that p_2^c ranks y_j from level 1 as its second choice, while q_2^c ranks x_k from level 1 second.

The second gadget in level 2 is on vertices $p_3^c, p_4^c, p_5^c, q_3^c, q_4^c, q_5^c$ and it is built analogously. That is, p_3^c and q_3^c are each other's top choices and similarly, p_4^c and q_4^c are each other's top choices, and so on. The preference list of p_5^c is $q_3^c > y_k > q_4^c > q_5^c > D > \dots$ and the preference list of q_5^c is $p_4^c > x_i > p_3^c > p_5^c > D > \dots$

The third gadget in level 2 is on vertices $p_6^c, p_7^c, p_8^c, q_6^c, q_7^c, q_8^c$ and it is built analogously. In particular, the preference list of p_8^c is $q_6^c > y_i > q_7^c > q_8^c > D > \dots$ and the preference list of q_8^c is $p_7^c > x_j > p_6^c > p_8^c > D > \dots$

The fourth gadget in level 2 is on vertices $p'^c_0, p'^c_1, p'^c_2, q'^c_0, q'^c_1, q'^c_2$ and it is totally analogous to the first gadget in level 2. That is, p'^c_0 and q'^c_0 are each other's top choices and similarly, p'^c_1 and q'^c_1 are each other's top choices, and so on. In particular, the preference list of p'^c_2 is $q'^c_0 > y_j > q'^c_1 > q'^c_2 > D > \dots$ and the preference list of q'^c_2 is $p'^c_1 > x_k > p'^c_0 > p'^c_2 > D > \dots$

$s_0 \ : \ t_1 > q_0 > t_2 > q_3 > t_3 > D > t_0 > \ldots$

$t_0 \ : \ s_3 > p_7 > s_2 > p_4 > s_1 > D > s_0 > \ldots$

$s_1 \ : \ t_1 > t_0 > D > \ldots$

$t_1 \ : \ s_1 > s_0 > D > \ldots$

$s_2 \ : \ t_2 > t_0 > D > \ldots$

$t_2 \ : \ s_2 > s_0 > D > \ldots$

$s_3 \ : \ t_3 > t_0 > D > \ldots$

$t_3 \ : \ s_3 > s_0 > D > \ldots$

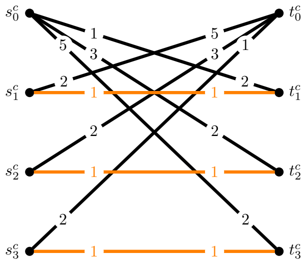

Figure 5 A clause gadget in level 3.

$z_0 \ : \ z_4 > z_5 > \cup_i \{x_i, y_i\} > \cup_{c,i} \{p^c_{3i+1}, q^c_{3i}, p'^c_{3i+1}, q'^c_{3i}\} > \cup_{c,i} \{a^c_i, b^c_i, a'^c_i, b'^c_i\} > z_1 > z_2 > z_3 > D > \ldots$

$z_1 \ : \ z_5 > z_4 > \cup_i \{x_i, y_i\} > \cup_{c,i} \{p^c_{3i+1}, q^c_{3i}, p'^c_{3i+1}, q'^c_{3i}\} > \cup_{c,i} \{a^c_i, b^c_i, a'^c_i, b'^c_i\} > z_0 > z_3 > z_2 > D > \ldots$

$z_2 \ : \ z_0 > z_1 > z_3 > z_4 > z_5 > D > \ldots$

$z_3 \ : \ z_1 > z_0 > z_2 > z_5 > z_4 > D > \ldots$

$z_4 \ : \ z_2 > z_3 > z_5 > z_0 > z_1 > D > \ldots$

$z_5 \ : \ z_3 > z_2 > z_4 > z_1 > z_0 > D > \ldots$

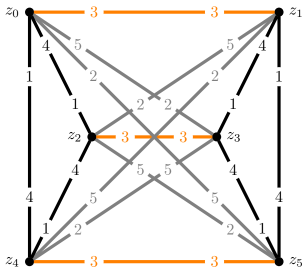

Figure 6 The Z-gadget.

Similarly, the fifth gadget in level 2 is on vertices $p'^c_3, p'^c_4, p'^c_5, q'^c_3, q'^c_4, q'^c_5$ and it is totally analogous to the second gadget in level 2. Also, the sixth gadget in level 2 is on vertices $p'^c_6, p'^c_7, p'^c_8, q'^c_6, q'^c_7, q'^c_8$ and it is totally analogous to the third gadget in level 2.

Level 3. To each clause $c = X_i \vee X_j \vee X_k$ in the formula ϕ, we create *2 gadgets* in level 3. The first gadget is on vertices $s^c_0, s^c_1, s^c_2, s^c_3, t^c_0, t^c_1, t^c_2, t^c_3$ and the preference lists of these vertices are described in Fig. 5.

The second gadget in level 3 is on $s'^c_0, s'^c_1, s'^c_2, s'^c_3, t'^c_0, t'^c_1, t'^c_2, t'^c_3$ and their preference lists are absolutely analogous to the preference lists of the first gadget in level 3.

The Z-gadget. The Z-gadget is on 6 vertices $z_0, z_1, z_2, z_3, z_4, z_5$ and the preference lists of these vertices are given in Fig. 6. The vertices in a set stand for all these vertices in an arbitrary order. For example, $\cup_i \{x_i, y_i\}$ denotes all the "top" vertices belonging to variable gadgets in an arbitrary order.

Note that G is a complete graph on an even number of vertices and so every popular matching in G has to be a perfect matching.

4 Popular edges in G

Call an edge e in G *popular* if there is a popular matching M in G such that $e \in M$. In this section we identify edges that are not popular and show that every popular edge is an *intra-gadget* edge, connecting two vertices of the same gadget. All missing proofs are in the full version of our paper on the arxiv [9].

▶ **Lemma 5.** *For any clause c, no popular matching in G can match s^c_0 (similarly, t^c_0) to a neighbor worse than t^c_0 (resp., s^c_0). An analogous statement holds for s'^c_0 and t'^c_0.*

▶ **Lemma 6.** *Every popular matching matches the vertices in the D-gadget among themselves.*

The gadget D admits 2 popular matchings: $\{(d_0, d_1), (d_2, d_3)\}$ and $\{(d_0, d_2), (d_1, d_3)\}$. So if M is a popular matching then either $\{(d_0, d_1), (d_2, d_3)\} \subset M$ or $\{(d_0, d_2), (d_1, d_3)\} \subset M$.

▶ **Lemma 7.** *Let (u, v) be an edge in G where both u and v prefer d_0 to each other. Then (u, v) cannot be a popular edge.*

▶ **Corollary 8.** *The edges (s^c_0, t^c_0) and (s'^c_0, t'^c_0) are not popular edges for any clause c.*

Corollary 8 follows from Lemma 7 by setting u and v to s^c_0 and t^c_0 (similarly, s'^c_0 and t'^c_0), respectively. Let us call u a level i vertex if u belongs to a level i gadget.

▶ **Lemma 9.** *No edge between a level i vertex and a level $i+1$ vertex is popular, for $0 \leq i \leq 2$.*

▶ **Lemma 10.** *All popular matchings match the 6 vertices of the Z-gadget among themselves.*

Proof. Let M be any popular matching in G. It follows from Lemma 7 that M has to pair each of z_2, z_3, z_4, and z_5 to a vertex in the Z-gadget. Let us now show that z_0 also has to be matched within the Z-gadget. Then it immediately follows that z_1 also has to be matched within the Z-gadget. We have the following 3 cases:

(1) Suppose z_0 is matched in M to a level 0 neighbor, say b^c_1. Then (a^c_1, b^c_1) is a blocking edge to M. Lemmas 6, 7, and 9 ensure that a^c_1 is either matched to z_1 or to b^c_2. We investigate these two cases below.
  - $(a^c_1, z_1) \in M$: Here both z_0 and z_1 are matched to vertices they prefer to all their neighbors inside the Z-gadget, except for z_4 and z_5. We know that z_4 and z_5 must be matched inside the Z-gadget. There are 3 subcases and in each case there is an alternating cycle in G_M with a blocking edge (a^c_1, b^c_1): a contradiction to M's popularity (by Theorem 3).
    - $(z_4, z_2) \in M$: the alternating cycle is $(b^c_1, z_0) \overset{(+,-)}{-} (z_4, z_2) \overset{(+,-)}{-} (z_1, a^c_1) \overset{(+,+)}{-} (b^c_1, z_0)$.
    - $(z_4, z_3) \in M$: the alternating cycle is $(b^c_1, z_0) \overset{(+,-)}{-} (z_4, z_3) \overset{(+,-)}{-} (z_1, a^c_1) \overset{(+,+)}{-} (b^c_1, z_0)$.
    - $(z_4, z_5) \in M$: the alternating cycle is $(b^c_1, z_0) \overset{(+,-)}{-} (z_4, z_5) \overset{(-,+)}{-} (z_1, a^c_1) \overset{(+,+)}{-} (b^c_1, z_0)$.
  - $(a^c_1, b^c_2) \in M$: Lemmas 6, 7, and 9 ensure that a^c_2 is matched to z_1 (recall that M is perfect). This leads to the same 3 subcases as above, except that instead of the edge (z_1, a^c_1), there is the path $(z_1, a^c_2) - (b^c_2, a^c_1)$ in G_M: here (a^c_2, b^c_2) is labeled $(+, -)$.

(2) Suppose z_0 is matched in M to a level 1 neighbor, say y_i.
This case is similar to the previous case. Here the edge (x_i, y_i) becomes the blocking edge to M. It follows from Lemmas 6, 7, and 9 that x_i is either matched to z_1 or to y'_i. The latter case leaves x'_i unmatched and the subcases that arise in the former case are analogous to the ones in case (1).

(3) Suppose z_0 is matched in M to a level 2 neighbor, say q^c_0.
It follows from Lemmas 6, 7, and 9 that $(p^c_0, q^c_2), (p^c_2, q^c_1)$, and (p^c_1, z_1) are in M. Consider the alternating path $(z_0, q^c_0) - (p^c_2, q^c_1) - (p^c_1, z_1)$: it has two blocking edges (p^c_2, q^c_0) and (p^c_1, q^c_1). This is again a contradiction to M's popularity.

Recall that Lemma 6 showed that all vertices of D must be matched within the gadget. Thus z_0 cannot be matched to a vertex in the D-gadget. The case where z_0 is matched in M to a level 3 neighbor does not arise as such an edge would violate Lemma 7. This finishes our proof that any popular matching M matches the 6 vertices of the Z-gadget among themselves. ◀

It follows from Lemmas 6, 7, 9, and 10 that every popular edge is an intra-gadget edge. Lemma 11 (proof in [9]) shows that there is only one possibility for a popular matching within the Z-gadget. Thus every popular matching in G contains $(z_0, z_1), (z_2, z_3), (z_4, z_5)$.

▶ **Lemma 11.** *The only popular matching inside the Z-gadget is $\{(z_0, z_1), (z_2, z_3), (z_4, z_5)\}$.*

5 Stable states versus unstable states

In this section we will show how to obtain a 1-in-3 satisfying assignment for the input ϕ from any popular matching in G. The following definition will be useful to us.

▶ **Definition 12.** A gadget A in $G = (V, E)$ is said to be in *unstable state* with respect to matching M if there is a blocking edge $(u, v) \in V(A) \times V(A)$ with respect to M. If there is no such blocking edge to M then we say A is in *stable state* with respect to M.

In Figures 2-6 depicting our gadgets, corresponding to matchings that consist of colored edges within the gadget, the relevant gadget is in *stable* state. A level 1 gadget in unstable state will encode the corresponding variable being set to true while a level 1 gadget in stable state will encode the corresponding variable being set to false. We will now analyze what gadgets are in stable/unstable state with respect to any popular matching M in G. This will lead to the proof that for any clause c, exactly one of the level 1 gadgets corresponding to the 3 variables in c is in unstable state.

▶ **Lemma 13.** *For any clause c, the following statements hold:*
- *all its 6 level 0 gadgets are in stable state with respect to M;*
- *both its level 3 gadgets in G are in unstable state with respect to M.*

Proof. Consider a level 0 gadget corresponding to clause c, say the one on vertices $a_1^c, b_1^c, a_2^c, b_2^c$. Lemmas 6, 7, 9, and 10 imply that either $\{(a_1^c, b_1^c), (a_2^c, b_2^c)\} \subset M$ or $\{(a_1^c, b_2^c), (a_2^c, b_1^c)\} \subset M$. Thus there is no blocking edge within this gadget. As this holds for every level 0 gadget corresponding to c and for every clause c, the first part of the lemma follows.

We will now prove the second part of the lemma. Since M is a perfect matching, the vertices s_0^c, t_0^c (also $s_0'^c, t_0'^c$) have to be matched in M, for all clauses c. It follows from Lemmas 6 and 7 that both s_0^c and t_0^c (similarly, $s_0'^c$ and $t_0'^c$) have to be matched to neighbors that are better than d_0. Lemma 9 showed that there is no popular edge between a level 3 vertex and a level 2 vertex. Thus s_0^c is matched to t_i^c for some $i \in \{1, 2, 3\}$.

If s_0^c is matched to t_i^c then s_i^c has to be matched to t_0^c – otherwise Lemma 7 would be violated by s_i^c and its partner. So (s_i^c, t_i^c) blocks M and this holds for every clause c. Similarly, there is a blocking edge $(s_i'^c, t_i'^c)$ for some $i \in \{1, 2, 3\}$ for every clause c. ◀

▶ **Lemma 14.** *For any clause c, at least* one *of the following two conditions has to hold:*
- *two or more of its first three level 2 gadgets are in unstable state with respect to M;*
- *two or more of its last three level 2 gadgets are in unstable state with respect to M.*

The proof of Lemma 14 is given in [9]. Recall that there are three level 1 gadgets associated with any clause c: these gadgets correspond to the three variables in c.

▶ **Lemma 15.** *Let $c = X_i \vee X_j \vee X_k$. At least one of the level 1 gadgets corresponding to X_i, X_j, X_k is in unstable state with respect to M.*

Proof. Suppose not. That is, assume that for some clause c, all three of its level 1 gadgets are in stable state. Let $c = X_i \vee X_j \vee X_k$. So (x_r, y_r) and (x_r', y_r') are in M for all $r \in \{i, j, k\}$.

We know from Lemma 14 that either two or more of the *first* three level 2 gadgets corresponding to c are in unstable state with respect to M or two or more of the *last* three level 2 gadgets corresponding to c are in unstable state with respect to M. Assume without loss of generality that the first and second gadgets, i.e., those on p_i^c, q_i^c, for $0 \le i \le 5$, are in unstable state with respect to M.

We know from our lemmas in Section 4 that there is no popular edge across gadgets. Thus M matches the 6 vertices of a level 2 gadget with each other. In particular, it follows from Lemma 7 that for the level 2 gadget on p_i^c, q_i^c for $i = 0, 1, 2$, we have (i) $(p_0^c, q_0^c), (p_1^c, q_1^c), (p_2^c, q_2^c)$ in M or (ii) $(p_0^c, q_2^c), (p_1^c, q_1^c), (p_2^c, q_0^c)$ in M or (iii) $(p_0^c, q_0^c), (p_1^c, q_2^c), (p_2^c, q_1^c)$ in M.

There are two unstable states for each level 2 gadget, i.e., either (ii) or (iii) above for the gadget on p_i^c, q_i^c for $i = 0, 1, 2$. A level 2 gadget can be in either of these two unstable states in M – without loss of generality assume that M contains $(p_0^c, q_0^c), (p_1^c, q_2^c), (p_2^c, q_1^c)$ and $(p_3^c, q_5^c), (p_4^c, q_4^c), (p_5^c, q_3^c)$. Observe that p_2^c likes y_j more than q_1^c and similarly, q_5^c likes x_i more than p_3^c. Consider the following alternating path ρ with respect to M:

$$(q_2^c, p_1^c) \overset{(+,+)}{-} (q_1^c, p_2^c) \overset{(+,-)}{-} (y_j, x_j) \overset{(-,+)}{-} (z_0, z_1) \overset{(+,-)}{-} (y_i, x_i) \overset{(-,+)}{-} (q_5^c, p_3^c) \overset{(+,+)}{-} (q_3^c, p_5^c).$$

Note that M has to contain (z_0, z_1) (by Lemma 11). Observe that ρ is an alternating path in G_M with *two* blocking edges (p_1^c, q_1^c) and (p_3^c, q_3^c). This is a contradiction to M's popularity (by Theorem 3) and the lemma follows. ◀

We can also show that (see [9]) *at most one* of the level 1 gadgets corresponding to X_i, X_j, X_k is in unstable state with respect to M. So *exactly one* of the level 1 gadgets corresponding to X_i, X_j, X_k is in unstable state with respect to M. This allows us to set a 1-in-3 satisfying assignment to instance ϕ. For each variable X_i in ϕ do:

– if the gadget corresponding to X_i is in *unstable* state then set $X_i = \mathsf{true}$ else set $X_i = \mathsf{false}$.

It follows from our above discussion that this is a 1-in-3 satisfying assignment for ϕ. We have thus shown the following result.

▶ **Theorem 16.** *If G admits a popular matching then ϕ has a 1-in-3 satisfying assignment.*

6 The converse

We will now show the converse of Theorem 16, i.e., if ϕ has a 1-in-3 satisfying assignment S then G admits a popular matching. We will use S to construct a popular matching M in G as follows. To begin with, $M = \emptyset$.

Level 1. For each variable X_i do:

- if X_i is set to true in S then add (x_i, y_i') and (x_i', y_i) to M;
- else add (x_i, y_i) and (x_i', y_i') to M.

For each clause $c = X_i \vee X_j \vee X_k$, we know that exactly one of X_i, X_j, X_k is set to true in S. Assume without loss of generality that $X_k = \mathsf{true}$ in S. For the level 0, 2, and 3 gadgets corresponding to c, we do as follows:

Level 0. Recall that there are *six* level 0 gadgets that correspond to c. For the first 3 gadgets (these are on vertices a_i^c, b_i^c for $i = 1, \ldots, 6$) do:

- include $(a_1^c, b_2^c), (a_2^c, b_1^c)$ from the first gadget;
- include $(a_3^c, b_3^c), (a_4^c, b_4^c)$ from the second gadget;
- choose either $(a_5^c, b_5^c), (a_6^c, b_6^c)$ or $(a_5^c, b_6^c), (a_6^c, b_5^c)$ from the third gadget.

Observe that since the third variable X_k of c was set to be `true`, cross edges are fixed in the first gadget (see Fig. 3), while the other stable matching (horizontal edges) is chosen in the second gadget.

For the fourth and fifth gadgets, we will do exactly the opposite. Also, it will not matter which stable pair of edges is chosen from the third and sixth gadgets. So for the last 3 level 0 gadgets corresponding to c (these are on vertices $a_i'^c, b_i'^c$ for $i = 1, \dots, 6$) do:

- include $(a_1'^c, b_1'^c), (a_2'^c, b_2'^c)$ from the fourth gadget;
- include $(a_3'^c, b_4'^c), (a_4'^c, b_3'^c)$ from the fifth gadget.
- choose either $(a_5'^c, b_5'^c), (a_6'^c, b_6'^c)$ or $(a_5'^c, b_6'^c), (a_6'^c, b_5'^c)$ from the sixth gadget.

Level 2. Recall that there are *six* level 2 gadgets that correspond to c. For the first 3 gadgets (these are on vertices p_i^c, q_i^c for $i = 0, \dots, 8$) do:

- include $(p_0^c, q_2^c), (p_1^c, q_1^c), (p_2^c, q_0^c)$ from the first gadget
- include $(p_3^c, q_3^c), (p_4^c, q_5^c), (p_5^c, q_4^c)$ from the second gadget
- include $(p_6^c, q_6^c), (p_7^c, q_7^c), (p_8^c, q_8^c)$ from the third gadget

In the first three gadgets, because $X_k =$ `true`, the third one is set to parallel edges, reaching the stable state, while the first one is blocked by the top horizontal edge and the second one is blocked by the middle horizontal edge. Include isomorphic edges (to the above ones) from the last three level 2 gadgets corresponding to c, i.e., include $(p_0'^c, q_2'^c), (p_1'^c, q_1'^c), (p_2'^c, q_0'^c)$ from the fourth gadget, and so on. On this level, the last three gadgets mimic the matching edges from the first three gadgets, unlike in level 0.

Level 3. For the first level 3 gadget corresponding to c do:

- include $(s_0^c, t_3^c), (s_1^c, t_1^c), (s_2^c, t_2^c), (s_3^c, t_0^c)$ in M.

Since the third variable in c was set to be `true`, the vertices s_0^c and t_0^c are matched to t_3^c and s_3^c, respectively – thus the bottom horizontal edge (s_3^c, t_3^c) blocks M. Include isomorphic edges (to the above ones) for the second level 3 gadget corresponding to c, i.e., include $(s_0'^c, t_3'^c), (s_1'^c, t_1'^c), (s_2'^c, t_2'^c), (s_3'^c, t_0'^c)$ in M. Once again, the second gadget mimics the matching edges on the first gadget.

***Z*-gadget and *D*-gadget.** Finally include the edges $(z_0, z_1), (z_2, z_3), (z_4, z_5)$ from the Z-gadget in M. By Lemma 11, every popular matching in G has to include these edges. Also include $(d_0, d_1), (d_2, d_3)$ from the D-gadget in M.

6.1 The popularity of M

We will now prove the popularity of the above matching M via the LP framework of popular matchings initiated in [18] for bipartite graphs. This framework generalizes to provide a sufficient condition for popularity in non-bipartite graphs [10]. This involves showing a witness $\vec{\alpha} \in \{0, \pm 1\}^{|V|}$ such that $\vec{\alpha}$ is a *certificate* of M's popularity. In order to define the constraints that $\vec{\alpha}$ has to satisfy so as to certify M's popularity, let us define an edge weight function w_M as follows.

For any edge (u, v) in G do:

- if (u, v) is labeled $(-, -)$ then set $w_M(u, v) = -2$;
- if (u, v) is labeled $(+, +)$ then set $w_M(u, v) = 2$;
- else set $w_M(u, v) = 0$. (So $w_M(e) = 0$ for all $e \in M$.)

Let N be any perfect matching in G. It is easy to see from the definition of the edge weight function w_M that $w_M(N) = \phi(N,M) - \phi(M,N)$.

Let the max-weight perfect fractional matching LP in the graph G with edge weight function w_M be our primal LP. This is LP1 defined below.

$$\text{maximize} \sum_{e\in E} w_M(e)x_e \qquad \text{(LP1)}$$

$$\text{subject to}$$

$$\textstyle\sum_{e\in\delta(u)} x_e \;=\; 1 \;\; \forall\, u \in V \qquad \text{and} \qquad x_e \geq 0 \;\; \forall\, e \in E.$$

If the primal optimal value is at most 0 then $w_M(N) \leq 0$ for all perfect matchings N in G, i.e., $\phi(N,M) \leq \phi(M,N)$. This means $\phi(M',M) \leq \phi(M,M')$ for *all* matchings M' in G, since G is a complete graph on an even number of vertices (so $M' \subseteq$ some perfect matching). That is, M is a popular matching in G.

Consider the LP that is dual to LP1. This is LP2 given below in variables α_u, where $u \in V$.

$$\text{minimize} \sum_{u\in V} \alpha_u \qquad \text{(LP2)}$$

$$\text{subject to}$$

$$\alpha_u + \alpha_v \;\geq\; w_M(u,v) \;\; \forall\,(u,v) \in E.$$

If we show a dual feasible solution $\vec{\alpha}$ such that $\sum_{u\in V}\alpha_u = 0$ then the primal optimal value is at most 0, i.e., M is a popular matching.

In order to prove the popularity of M, we define $\vec{\alpha}$ as follows. For each variable X_r do:

- if X_r was set to `true` then set $\alpha_{x_r} = \alpha_{y_r} = 1$ and $\alpha_{x'_r} = \alpha_{y'_r} = -1$;
- else set $\alpha_{x_r} = \alpha_{y_r} = \alpha_{x'_r} = \alpha_{y'_r} = 0$.

Let clause $c = X_i \vee X_j \vee X_k$. Recall that we assumed that $X_i = X_j =$ `false` and $X_k =$ `true`. For the vertices in clauses corresponding to c, we will set α-values as follows.

- For every level 0 vertex v do: set $\alpha_v = 0$.
- For the first three level 2 gadgets corresponding to c do:
  - set $\alpha_{p^c_0} = \alpha_{q^c_0} = 1$, $\alpha_{p^c_1} = 1, \alpha_{q^c_1} = -1$, and $\alpha_{p^c_2} = \alpha_{q^c_2} = -1$;
  - set $\alpha_{p^c_3} = -1, \alpha_{q^c_3} = 1$, $\alpha_{p^c_4} = \alpha_{q^c_4} = 1$, and $\alpha_{p^c_5} = \alpha_{q^c_5} = -1$;
  - set $\alpha_{p^c_6} = \alpha_{q^c_6} = \alpha_{p^c_7} = \alpha_{q^c_7} = \alpha_{p^c_8} = \alpha_{q^c_8} = 0$.

The setting of α-values is analogous for vertices in the last three level 2 gadgets corresponding to c. For the first level 3 gadget corresponding to c do:

- set $\alpha_{s^c_0} = \alpha_{t^c_0} = -1$, $\alpha_{s^c_1} = -1, \alpha_{t^c_1} = 1$, $\alpha_{s^c_2} = -1$, $\alpha_{t^c_2} = 1$, and $\alpha_{s^c_3} = \alpha_{t^c_3} = 1$.

The setting of α-values is analogous for vertices in the other level 3 gadget corresponding to c. For the z-vertices do: set $\alpha_u = 0$ for all $u \in \{z_0,\ldots,z_5\}$. For the d-vertices do:

- set $\alpha_{d_0} = \alpha_{d_2} = -1$ and $\alpha_{d_1} = \alpha_{d_3} = 1$.

Properties of $\vec{\alpha}$. For every $(u,v) \in M$, either $\alpha_u = \alpha_v = 0$ or $\{\alpha_u, \alpha_v\} = \{-1, 1\}$; so $\alpha_u + \alpha_v = 0$. Since M is a perfect matching, we have $\sum_{u\in V}\alpha_u = 0$. The claims stated below (proofs are in [9]) show that $\vec{\alpha}$ is a feasible solution to LP2. This will prove the popularity of M.

We need to show that every edge (u,v) is *covered*, i.e., $\alpha_u + \alpha_v \geq w_M(u,v)$. We have already observed that for any $(u,v) \in M$, $\alpha_u + \alpha_v = 0 = w_M(u,v)$.

▶ **Claim 17.** *Let* (u, v) *be a blocking edge to* M*. Then* $\alpha_u + \alpha_v = 2 = w_M(u, v)$*.*

▶ **Claim 18.** *Let* (u, v) *be an intra-gadget edge that is non-blocking. Then* $\alpha_u + \alpha_v \geq w_M(u, v)$*.*

▶ **Claim 19.** *Let* (u, v) *be any inter-gadget edge. Then* $\alpha_u + \alpha_v \geq w_M(u, v)$*.*

Thus we have shown the following theorem.

▶ **Theorem 20.** *If* ϕ *has a 1-in-3 satisfying assignment then* G *admits a popular matching.*

Theorem 2 stated in Section 1 follows from Theorems 16 and 20. Thus the popular matching problem in a roommates instance on n vertices with complete preference lists is NP-hard for even n.

References

1 Condorcet method. Accessed 3 October 2018. URL: `https://en.wikipedia.org/wiki/Condorcet_method`.

2 C.-C. Huang. Personal communication.

3 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. Springer, 2010.

4 P. Biró and E. McDermid. Three-sided stable matchings with cyclic preferences. *Algorithmica*, 58(1):5–18, 2010.

5 K.S. Chung. On the Existence of Stable Roommate Matchings. *Games and Economic Behavior*, 33(2):206–230, 2000.

6 M.-J.-A.-N. de C. (Marquis de) Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix.* L'Imprimerie Royale, 1785.

7 Ágnes Cseh. Popular matchings. *Trends in Computational Social Choice*, page 105, 2017.

8 Ágnes Cseh and Telikepalli Kavitha. Popular edges and dominant matchings. *Mathematical Programming*, pages 1–21, 2017. online first.

9 Ágnes Cseh and Telikepalli Kavitha. Popular Matchings in Complete Graphs. *CoRR*, abs/1807.01112, 2018.

10 Yuri Faenza, Telikepalli Kavitha, Vladlena Powers, and Xingyu Zhang. Popular Matchings and Limits to Tractability. *arXiv preprint arXiv:1805.11473*, 2018.

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 Sushmita Gupta, Pranabendu Misra, Saket Saurabh, and Meirav Zehavi. Popular Matching in Roommates Setting is NP-hard. *arXiv preprint arXiv:1803.09370*, 2018.

14 C.-C. Huang and T. Kavitha. Popular Matchings in the Stable Marriage Problem. In *Proceedings of ICALP '11: the 38th International Colloquium on Automata, Languages and Programming*, volume 6755 of *Lecture Notes in Computer Science*, pages 666–677. Springer, 2011.

15 C.-C. Huang and T. Kavitha. Near-popular matchings in the Roommates problem. *SIAM Journal on Discrete Mathematics*, 27(1):43–62, 2013.

16 Chien-Chung Huang and Telikepalli Kavitha. Popularity, mixed matchings, and self-duality. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2294–2310. Society for Industrial and Applied Mathematics, 2017.

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

18 T. Kavitha, J. Mestre, and M. Nasre. Popular Mixed Matchings. In *Proceedings of ICALP '09: the 36th International Colloquium on Automata, Languages and Programming*, volume 5555 of *Lecture Notes in Computer Science*, pages 574–584. Springer, 2009.

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

20 Telikepalli Kavitha. Popular half-integral matchings. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 55. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

21 Telikepalli Kavitha. Max-size popular matchings and extensions. *arXiv preprint arXiv:1802.07440*, 2018.

22 David F. Manlove. *Algorithmics of Matching Under Preferences*, volume 2 of *Series on Theoretical Computer Science*. WorldScientific, 2013. `doi:10.1142/8591`.

23 Thomas J Schaefer. The complexity of satisfiability problems. In *Proceedings of the tenth annual ACM symposium on Theory of computing*, pages 216–226. ACM, 1978.

24 A. Subramanian. A New Approach to Stable Matching Problems. *SIAM Journal on Computing*, 23(4):671–700, 1994.

25 J.J.M. Tan. A necessary and sufficient condition for the existence of a complete stable matching. *Journal of Algorithms*, 12:154–178, 1991.

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

27 Gerhard J Woeginger. Core stability in hedonic coalition formation. In *International Conference on Current Trends in Theory and Practice of Computer Science*, pages 33–50. Springer, 2013.