Popular matchings with two-sided preferences and one-sided ties

Ágnes Cseh[1]⋆, Chien-Chung Huang[2], and Telikepalli Kavitha[3]

[1] TU Berlin, Germany. cseh@math.tu-berlin.de
[2] Chalmers University, Sweden. huangch@chalmers.se
[3] Tata Institute of Fundamental Research, India. kavitha@tcs.tifr.res.in

Abstract. We are given a bipartite graph $G = (A \cup B, E)$ where each vertex has a preference list ranking its neighbors: in particular, every $a \in A$ ranks its neighbors in a strict order of preference, whereas the preference lists of $b \in B$ may contain ties. A matching M is *popular* if there is no matching M' such that the number of vertices that prefer M' to M exceeds the number that prefer M to M'. We show that the problem of deciding whether G admits a popular matching or not is NP-hard. This is the case even when every $b \in B$ either has a strict preference list or puts all its neighbors into a single tie. In contrast, we show that the problem becomes polynomially solvable in the case when each $b \in B$ puts all its neighbors into a single tie. That is, all neighbors of b are tied in b's list and and b desires to be matched to any of them. Our main result is an $O(n^2)$ algorithm (where $n = |A \cup B|$) for the popular matching problem in this model. Note that this model is quite different from the model where vertices in B have no preferences and do *not* care whether they are matched or not.

1 Introduction

We are given a bipartite graph $G = (A \cup B, E)$ where the vertices in A are called applicants and the vertices in B are called posts, and each vertex has a preference list ranking its neighbors in an order of preference. Here we assume that vertices in A have strict preferences while vertices in B are allowed to have ties in their preference lists. Thus each applicant ranks all posts that she finds interesting in a strict order of preference, while each post need not come up with a total order on all interested applicants – here applicants may get grouped together in terms of their suitability, thus equally competent applicants are tied together at the same rank.

Our goal is to compute a *popular* matching in G. The definition of popularity uses the notion of each vertex casting a "vote" for one matching versus another. A vertex v *prefers* matching M to matching M' if either v is unmatched in M' and matched in M or v is matched in both matchings and $M(v)$ (v's partner in M) is ranked better than $M'(v)$ in v's preference list. In an election between

⋆ Work done while visiting TIFR, supported by the Deutsche Telekom Stiftung.

matchings M and M', each vertex v votes for the matching that it prefers or it abstains from voting if M and M' are equally preferable to v. Let $\phi(M, M')$ be the number of vertices that vote for M in an election between M and M'.

Definition 1. *A matching M is* popular *if $\phi(M, M') \geq \phi(M', M)$ for every matching M'.*

If $\phi(M', M) > \phi(M, M')$, then we say M' is *more popular* than M and denote it by $M' \succ M$; else $M \succeq M'$. Observe that popular matchings need not always exist. Consider an instance where $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$ and for $i = 1, 2, 3$, each a_i has the same preference list which is b_1 followed by b_2 followed by b_3 while each b_i ranks a_1, a_2, a_3 the same, i.e. a_1, a_2, a_3 are tied together in b_i's preference list. It is easy to see that for any matching M here, there is another matching M' such that $M' \succ M$, thus this instance admits no popular matching.

The popular matching problem is to determine if a given instance $G = (A \cup B, E)$ admits a popular matching or not, and if so, to compute one. This problem has been studied in the following two models.

- *1-sided model:* here it is only vertices in A that have preferences and cast votes; vertices in B are objects with no preferences or votes.
- *2-sided model:* vertices on both sides have preferences and cast votes.

Popular matchings need not always exist in the 1-sided model and the problem of whether a given instance admits one or not can be solved efficiently using the characterization and algorithm from [1]. In the 2-sided model when all preference lists are strict, it can be shown that any stable matching is popular [3]; thus a popular matching can be found in linear time using the Gale-Shapley algorithm. However when ties are allowed in preference lists on both sides, Biró, Irving, and Manlove [3] showed that the popular matching problem is NP-complete. In this paper we focus on the following variant:

∗ it is only vertices in A that have preference lists ranking their neighbors, however vertices on *both* sides cast votes.

That is, vertices in B have no preference lists ranking their neighbors – however b desires to be matched to any of its neighbors. Thus in an election between two matchings, b abstains from voting if it is matched in both or unmatched in both, else it votes for the matching where it is matched. An intuitive understanding of such an instance is that A is a set of applicants and B is a set of tasks – while each applicant has a preference list over the tasks that she is interested in, each task just cares to be assigned to anyone who is interested in performing it. We will see in Section 2 that the above problem is significantly different from the popular matching problem in the 1-sided model where vertices in B do not cast votes. We show the following results here, complementing our polynomial time algorithm in Theorem 1 with our hardness result in Theorem 2.

Theorem 1. *Given a bipartite graph $G = (A \cup B, E)$ where each $a \in A$ has a strict preference list over its neighbors while each $b \in B$ puts all its neighbors into a single tie, the popular matching problem in G can be solved in $O(n^2)$ time, where $|A \cup B| = n$.*

Theorem 2. *The popular matching problem is* NP*-complete in* $G = (A \cup B, E)$ *where each* $a \in A$ *has a strict preference list while each* $b \in B$ *either has a strict preference list or puts all its neighbors into a single tie.*

Note that our NP-hardness reduction needs B to have $\Omega(|B|)$ vertices with strict preference lists and $\Omega(|B|)$ vertices with single ties as their preference lists. Theorem 2 follows from a simple reduction from the (2,2)-E3-SAT problem which is NP-complete [2]. Our reduction shows that the 2-sided popular matching problem in $G = (A \cup B, E)$ where every vertex in A has a strict preference list of length 2 or 4 and every vertex in B has either a strict preference list of length 2 or a single tie of length 2 or 3 as a preference list is NP-complete.

We show Theorem 1 by partitioning the set B into three sets: the first set X is a subset of top posts and, roughly speaking, the second set Y consists of *mid-level* posts, and the third set Z consists of *unwanted* posts (see Figure 1). Applicants get divided into two sets: the set of those with one or more neighbors in the set Z (call this set $\mathsf{nbr}(Z)$) and the rest (this set is $A \setminus \mathsf{nbr}(Z)$).

Our algorithm performs the partition of B into X, Y, and Z over several iterations. Initially $X = F$, where F is the set of top posts, $Y = B \setminus F$, and $Z = \emptyset$. In each iteration, certain top posts get *demoted* from X to Y and certain non-top posts get demoted from Y to Z. With new posts entering Z, we also have applicants moving from $A \setminus \mathsf{nbr}(Z)$ to $\mathsf{nbr}(Z)$. Using the partition $\langle X, Y, Z \rangle$ of B, we will build a graph H where each applicant keeps at most two edges: either to its most preferred post in X and also in Y or to its most preferred post in Z and also in Y. Some dummy posts may be included in Y.

We prove that G admits a popular matching if and only if H admits an A-complete matching, i.e., one that matches all vertices in A. We show that corresponding to any popular matching in G, there is a partition $\langle L_1, L_2, L_3 \rangle$ of B into *top posts*, *mid-level posts*, and *unwanted posts* such that $X \supseteq L_1$ and $Z \subseteq L_3$, where $\langle X, Y, Z \rangle$ is the partition computed by our algorithm. This allows us to show that if H does not admit an A-complete matching, then G has no popular matching. In fact, not every popular matching in G becomes an A-complete matching in H. However it will be the case that if G admits popular matchings, then at least one of them becomes an A-complete matching in H.

Background. Popular matchings have been well-studied in the 1-sided model [1, 9–15] where only vertices of A have preferences and cast votes. Abraham et al. [1] gave polynomial time algorithms to determine if a given instance admits a popular matching or not – their algorithm also works when preference lists of vertices in A admit ties. Gärdenfors [5], who introduced the notion of popular matchings, considered this problem in the domain of 2-sided preference lists. In any instance $G = (A \cup B, E)$ with 2-sided strict preference lists, a stable matching is actually a minimum size popular matching and polynomial algorithms for computing a maximum size popular matching were given in [7, 8].

Organization of the paper. Section 2 has preliminaries, Section 3 has our algorithm and our proof of correctness. Due to the space constraints, certain proofs (incl. the proof of Theorem 2) have been omitted from this version of the paper. These proofs will be included in the full version of the paper.

2 Preliminaries

For any $a \in A$, let $f(a)$ denote a's most desired post. Let $F = \{f(a) : a \in A\}$ be the set of top posts. We will refer to posts in F as f-posts and to those in $B \backslash F$ as non-f-posts. For any $a \in A$, let r_a be the rank of a's most preferred non-f-post in a's preference list; when all of a's neighbors are in F, we set $r_a = \infty$. The following theorem characterizes popular matchings in the 1-sided voting model.

Theorem 3 (from [1]). *Let $G = (A \cup B, E)$ be an instance of the 1-sided popular matching problem, where each $a \in A$ has a strict preference list. Let M be any matching in G. M is popular if and only if the following two properties are satisfied:*

(i) M matches every $b \in F$ to some applicant a such that $b = f(a)$;
(ii) M matches each applicant a to either $f(a)$ or its neighbor of rank r_a.

Thus the only applicants that may be left unmatched in a popular matching here are those $a \in A$ that satisfy $r_a = \infty$.

Let us consider the following example where $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$: both a_1 and a_2 have the same preference list which is $b_1 > b_2$ (b_1 followed by b_2) while a_3's preference list is $b_1 > b_2 > b_3$. Assume first that only applicants cast votes. The only posts that any of a_1, a_2, a_3 can be matched to in a popular matching here are b_1 and b_2. As there are three applicants and only two possible partners in a popular matching, there is no popular matching here. However in our 2-sided voting model, where posts also care about being matched and all neighbors are in a single tie, we have a popular matching $\{(a_1, b_1), (a_2, b_2), (a_3, b_3)\}$. Note that b_3 is ranked third in a_3's preference list, which is worse than $r_{a_3} = 2$, however such edges are permitted in popular matchings in our 2-sided model.

Consider the following example: $A = \{a_0, a_1, a_2, a_3\}$ and $B = \{b_0, b_1, b_2, b_3\}$; both a_1 and a_2 have the same preference list which is $b_1 > b_2$ while a_3's preference list is $b_1 > b_0 > b_2$ and a_0's preference list is $b_0 > b_3$. There is again no popular matching here in the 1-sided model, however in our 2-sided voting model, we have a popular matching $\{(a_0, b_3), (a_1, b_1), (a_2, b_2), (a_3, b_0)\}$. Note that $b_0 \in F$ and here it is matched to a_3 and $f(a_3) \neq b_0$; also a_3 is matched to its second ranked post: this is neither its top post nor its r_{a_3}-th ranked post ($r_{a_3} = 3$ here).

Thus popular matchings in our 2-sided voting model are quite different from the characterization given in Theorem 3 for popular matchings in the 1-sided model. Our algorithm (presented in Section 3) uses the following decomposition.

Dulmage-Mendelsohn decomposition [4]. Let M be a maximum matching in a bipartite graph $G = (A \cup B, E)$. Using M, we can partition $A \cup B$ into three disjoint sets: a vertex v is *even* (similarly, *odd*) if there is an even (resp., odd) length alternating path (with respect to M) from an unmatched vertex to v. Similarly, a vertex v is *unreachable* if there is no alternating path from an unmatched vertex to v. Denote by $\mathcal{E}$, $\mathcal{O}$, and $\mathcal{U}$ the sets of even, odd, and unreachable vertices, respectively. The following properties (proved in [6]) will be used in our algorithm and analysis.

- $\mathcal{E}$, $\mathcal{O}$, and $\mathcal{U}$ are pairwise disjoint. Let M' be any maximum matching in G and let $\mathcal{E}'$, $\mathcal{O}'$, and $\mathcal{U}'$ be the sets of even, odd, and unreachable vertices with respect to M', respectively. Then $\mathcal{E} = \mathcal{E}'$, $\mathcal{O} = \mathcal{O}'$, and $\mathcal{U} = \mathcal{U}'$.
- Every maximum matching M matches all vertices in $\mathcal{O} \cup \mathcal{U}$ and has size $|\mathcal{O}| + |\mathcal{U}|/2$. In M, every vertex in $\mathcal{O}$ is matched with some vertex in $\mathcal{E}$, and every vertex in $\mathcal{U}$ is matched with another vertex in $\mathcal{U}$.
- The graph G has no edge in $\mathcal{E} \times (\mathcal{E} \cup \mathcal{U})$.

3 Finding popular matchings in a 2-sided voting model

The input is $G = (A \cup B, E)$ where each applicant $a \in A$ has a strict preference list while each post $b \in B$ has a single tie as its preference list. Our algorithm below builds a graph H using a partition $\langle X, Y, Z \rangle$ of B that is constructed in an iterative manner. Initialize $X = F$, $Y = B \setminus F$, and $Z = \emptyset$.

For any $a \in A$, recall that r_a is the rank of a's most preferred non-f-post. For any $U \subseteq B$, let $\mathsf{nbr}(U)$ (similarly, $\mathsf{nbr}_H(U)$) denote the set of neighbors in G (resp., in H) of the vertices in U. Note that our algorithm will maintain $\mathsf{nbr}_H(X) \cap \mathsf{nbr}(Z) = \emptyset$ by ensuring that $\mathsf{nbr}_H(X) \subseteq A \setminus \mathsf{nbr}(Z)$.

(I) While true do
  0. H is the empty graph on $A \cup B$.
  1. For each $a \in A \setminus \mathsf{nbr}(Z)$ do:
     – if $f(a) \in X$ then add the edge $(a, f(a))$ to H.
  2. For every $b \in X$ that is isolated in H do:
     – delete b from X and add b to Y.
  3. For each $a \in A$ do:
     – let b be a's most preferred post in the set Y; if the rank of b in a's preference list is $\leq r_a$ (i.e., r_a or better), then add (a, b) to H.
  4. Consider the graph H constructed in steps 1-3. Compute a maximum matching in H. *[This is to identify "even" posts in H.]*
     – If there exist even posts in Y then delete all even posts from Y and add them to Z.
     – Else quit the While-loop.

(II) Every $a \in \mathsf{nbr}(Z)$ adds the edge (a, b) to H where b is a's most preferred post in the set Z.

(III) Add all posts in $D = \{\ell(a) : a \in A \text{ and } r_a = \infty\}$ to Y, where $\ell(a)$ is the *dummy* last resort post of applicant a. For every applicant a such that $\mathsf{nbr}(\{a\}) \subseteq X$, add the edge $(a, \ell(a))$ to H.

Note that if a matching M includes the edge $(a, \ell(a))$, it means a is unmatched in M. The condition for exiting the While-loop ensures that all posts in Y, and hence all in $X \cup Y$, are odd/unreachable in the subgraph of H with the set of posts restricted to *real* posts in $X \cup Y$ (i.e., the non-dummy ones). So starting with a maximum matching in this subgraph and augmenting it after adding the edges on posts in Z in Step (II) and the edges on dummy posts in Step (III), we get a maximum matching in H that matches all real posts in $X \cup Y$. After the construction of H, our algorithm for the popular matching problem in G is given below.

- If H admits an A-complete matching, then return one that matches all real posts in $X \cup Y$; else output "G has no popular matching".

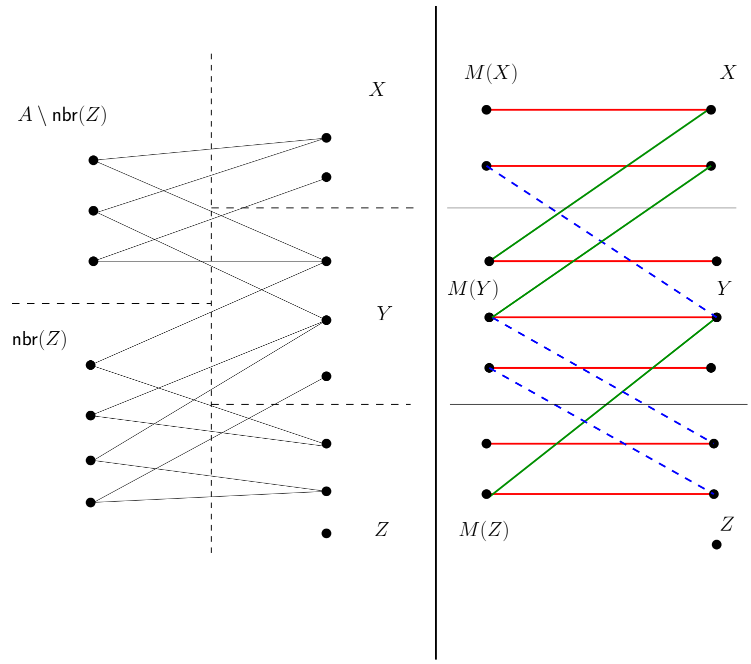

Fig. 1. The set B gets partitioned into X, Y, and Z. We have $\mathsf{nbr}_H(X) \cap \mathsf{nbr}(Z) = \emptyset$. In the figure on the right, the horizontal edges belong to M. Only the edges of $(M(Y) \times X) \cup (M(Z) \times (X \cup Y))$ can be labeled $+1$.

In the rest of this section, we prove the following theorem.

Theorem 4. *G admits a popular matching if and only if H admits an A-complete matching, i.e., one that matches all vertices in A.*

The sufficient part. We first show that if H admits an A-complete matching, then G admits a popular matching. We have already observed that if H admits an A-complete matching, then H has an A-complete matching that matches all real posts in $X \cup Y$. Call this matching M; other than the dummy last resort posts, all posts that are unmatched in M have to be in Z.

A useful observation is that $Z \subseteq B \setminus F$. This is because in Step 4 of the While-loop in our algorithm, all f-posts in Y are odd/unreachable in H as they are the only neighbors in H of applicants who regard them as f-posts.

We now assign edge labels in $\{\pm 1\}$ to all edges in $G \setminus M$: for an edge (a, b) in $G \setminus M$, if a prefers b to $M(a)$, then we label this edge $+1$, else we label this -1. The label of (a, b) is basically a's vote for b vs $M(a)$. Figure 1 is helpful here.

For any $U \in \{X, Y, Z\}$, let $M(U) \subseteq A$ be the set of applicants matched in M to posts in U. The following lemma is important.

Lemma 1. *Every edge of G in $M(X) \times Y$ is labeled -1; similarly, every edge in $M(Y) \times Z$ is labeled -1. Any edge labeled $+1$ has to be either in $M(Y) \times X$ or in $M(Z) \times (X \cup Y)$.*

Proof. Every edge of $\mathsf{nbr}(X) \times X$ that is present in H is a top ranked edge. Since M belongs to H, the edges of M from $\mathsf{nbr}(X) \times X$ are top ranked edges. Thus it is clear that every edge of G in $M(X) \times Y$ is labeled -1. Regarding $M(Y) \times Z$, every edge of $\mathsf{nbr}(Y) \times Y$ that is present in the graph H is an edge (a, b) where the rank of b in a's preference list is $\leq r_a$ (i.e., r_a or better); on the other hand, every edge of $\mathsf{nbr}(Z) \times Z$ that is present in the graph H is an edge (a, b') where the rank of b' in a's preference list is $\geq r_a$ (because $b' \in B \setminus F$). Since M belongs to H, the edges of M from $\mathsf{nbr}(Y) \times Y$ are ranked better than edges of $\mathsf{nbr}(Z) \times Z$. Thus every edge of G in $M(Y) \times Z$ is labeled -1.

We now show that any edge labeled $+1$ has to be in either $M(Y) \times X$ or $M(Z) \times (X \cup Y)$ (see Figure 1). Consider any edge $(a, b) \notin M$ such that $b \in U$ and $a \in M(U)$, where $U \in \{X, Y, Z\}$. It follows from the construction of the graph H that a vertex in $\mathsf{nbr}(U)$ can be adjacent in H to only its most preferred post in U. Thus any edge $(a, b) \notin M$ where $b \in U$ and $a \in M(U)$ is ranked -1. We have already seen that all edges in $M(X) \times Y$ and in $M(Y) \times Z$ are labeled -1. There are no edges in $M(X) \times Z$ since $M(X) \subseteq A \setminus \mathsf{nbr}(Z)$. Thus any edge labeled $+1$ has to be in either $M(Y) \times X$ or $M(Z) \times (X \cup Y)$. $\square$

Let M' be any matching in G. The symmetric difference of M' and M is denoted by $M' \oplus M$: this consists of alternating paths and alternating cycles – note that edges here alternate between M and M'. It will be convenient to assume that last resort posts are used only in M and not in M'. The claim that $M \succeq M'$ follows easily from Lemma 2. This proves the popularity of M.

Lemma 2. *Consider $M' \oplus M$. The following three properties hold:*

(i) in any alternating cycle in $M' \oplus M$, the number of edges that are labeled -1 is at least the number of edges that are labeled $+1$.
(ii) in any alternating path in $M' \oplus M$, the number of edges that are labeled $+1$ is at most two plus *the number of edges that are labeled -1; in case one of the endpoints of this path is a last resort post, then the number of edges labeled $+1$ is at most* one plus *the number of edges labeled -1.*
(iii) in any even length alternating path in $M' \oplus M$, the number of edges that are labeled -1 is at least the number of edges that are labeled $+1$; in case one of the endpoints of this path is a last resort post, then the number of edges labeled -1 is at least one plus *the number of edges labeled $+1$.*

The necessary part. We now show the other side of Theorem 4. That is, if G admits a popular matching, then H admits an A-complete matching. Let M^* be a popular matching in G. Lemma 3 will be useful to us.

Lemma 3. *If $(a,b) \in M^*$ and $b \in F$, then b has rank better than r_a in a's preference list.*

Label the edges of $G \setminus M^*$ by $+1$ or -1: the label of an edge (a,b) in $G \setminus M^*$ is the vote of a for b vs $M^*(a)$. In case a is not matched in M^*, then $\mathsf{vote}(a,b) = +1$ for any neighbor b of a. Due to the popularity of M^*, the following two properties hold on these edge labels (otherwise $M^* \oplus \rho \succ M^*$).

- $(*)$ there is no alternating path ρ such that the edge labels in $\rho \setminus M^*$ are $\langle +1, +1, +1, \cdots \rangle$, i.e., no three consecutive non-matching edges labeled $+1$.
- $(**)$ there is no alternating path ρ where the edge labels in $\rho \setminus M^*$ are $\langle +1, +1, -1, +1, +1, \cdots \rangle$.

From the matching M^* and the edge labels on $G \setminus M^*$, we partition B into $L_1 \cup L_2 \cup L_3$ as follows. This partition uses property $(*)$ in a crucial way.

0. Initialize $L_1 = L_2 = \emptyset$ and $L_3 = \{b \in B : b \text{ is unmatched in } M^*\}$. We now add more posts to the sets L_1, L_2, L_3 as described below.
1. Any alternating path with respect to M^* can have at most two consecutive non-matching edges that are labeled $+1$. For each length-5 alternating path $\rho = a_0\text{-}b_0\text{-}a_1\text{-}b_1\text{-}a_2\text{-}b_2$ where $(a_0,b_0),(a_1,b_1),(a_2,b_2) \in M^*$ and both (a_1,b_0) and (a_2,b_1) are marked $+1$, add b_{i-1} to L_i, for $i = 1,2,3$.
2. Now consider those $b \in B$ that are matched in M^* but b is not a part of any length-5 alternating path where both the non-matching edges are labeled $+1$. We repeat the following two steps till there are no more posts to be added to either L_2 or L_3 via these rules:
   - suppose $M^*(b)$ has no $+1$ edge incident on it: if $M^*(b) \in \mathsf{nbr}(L_3)$, then add b to L_2.
   - if $M^*(b)$ has a $+1$ edge to a vertex in L_2, then add b to L_3.
3. For each b such that $M^*(b)$ has no $+1$ edge incident on it:
   – if $M^*(b) \notin \mathsf{nbr}(L_3)$, then add b to L_1.
4. For each b not yet in $L_2 \cup L_3$ and $M^*(b)$ has a $+1$ edge to a vertex in L_1:
   – add b to L_2.

Lemma 4. *The above partition $\langle L_1, L_2, L_3 \rangle$ satisfies the following properties:*

(1) $F \subseteq L_1 \cup L_2$, where F is the set of top posts.
(2) $M^*(L_1) \cap \mathsf{nbr}(L_3) = \emptyset$.

We will use the partition $\langle L_1, L_2, L_3 \rangle$ of B to build the following subgraph $G' = (A \cup B, E')$ of G. For each $a \in A$, include the following edges in E':

- (i) if $a \notin \mathsf{nbr}(L_3)$, then add the edge $(a, f(a))$ to E'.
- (ii) if a has a neighbor of rank $\leq r_a$ in L_2, then add the edge (a,b) to E', where b is a's most preferred neighbor in L_2.
- (iii) if $a \in \mathsf{nbr}(L_3)$, then add the edge (a,b) to E', where b is a's most preferred neighbor in L_3.

Lemma 5. *Every edge of the matching M^* belongs to the graph G'.*

Proof. The set B has been partitioned into $L_1 \cup L_2 \cup L_3$. We will now show that for each post b_0 that is matched in M^*, the edge $(M^*(b_0), b_0)$ belongs to G'.

– *Case 1.* The post $b_0 \in L_1$. Hence there is no $+1$ edge incident on $a_0 = M^*(b_0)$, in other words, $b_0 = f(a_0)$. Lemma 4.2 tells us that $M^*(L_1) \cap \mathsf{nbr}(L_3) = \emptyset$; hence a_0 has no neighbor in L_3 and by rule (i) above, the edge $(a_0, f(a_0)) = (a_0, b_0)$ belongs to the edge set of G'.

– *Case 2.* Next we consider the case when $b_0 \in L_2$. It is easy to see that b_0 has to be a_0's most preferred post in L_2, where $a_0 = M^*(b_0)$. Otherwise there would have been an edge (a_0, b_1) labeled $+1$ with $b_1 \in L_2$, where b_1 is a_0's most preferred post in L_2. Then either $b_1 \in L_1$ or $b_0 \in L_3$ (from how we construct the sets L_1, L_2, L_3), a contradiction. We now have to show that the rank of b_0 in a_0's preference list is $\leq r_a$, otherwise the edge (a_0, b_0) does not belong to G'.

Suppose $b_0 \in F$. Since the edge $(a_0, b_0) \in M^*$, which is a popular matching, it follows from Lemma 3 that b_0 is ranked better than r_{a_0} in a_0's preference list; thus the edge (a_0, b_0) would belong to G'. So the case left is when $b_0 \notin F$. If b_0 is not a_0's most preferred post outside F, then there is the length-5 alternating path $\rho = b_0$-a_0-b_1-a_1-$f(a_1)$-$M^*(f(a_1))$, where b_1 is the most preferred post of a_0 outside F and $a_1 = M^*(b_1)$. The alternating path ρ has two consecutive non-matching edges (a_0, b_1) and $(a_1, f(a_1))$ that are labeled $+1$. This contradicts the presence of b_0 in L_2 as such a post would have to be in L_3. Thus if $b_0 \notin F$, then b_0 has to be a_0's most preferred post outside F, i.e. b_0 has rank r_{a_0} in a_0's preference list.

– *Case 3.* We finally consider the case when the post $b_0 \in L_3$. We need to show that b_0 is the most preferred post of $a_0 = M^*(b_0)$ in L_3. Suppose not. Let b_1 be a_0's most preferred post in L_3. Since $b_1 \in L_3$ while $F \cap L_3 = \emptyset$ (by Lemma 4.1), we know that there is an edge labeled $+1$ incident on $a_1 = M^*(b_1)$. Let this edge be (a_1, b_2) and let a_2 be $M^*(b_2)$. So there is a length-5 alternating path $p = b_0$-a_0-b_1-a_1-b_2-a_2 where both the non-matching edges (a_0, b_1) and (a_1, b_2) are labeled $+1$. This contradicts the presence of b_1 in L_3 as such a post would have to be in L_2. Thus b_0 is a_0's most preferred post in L_3. $\square$

The following lemma shows the relationship between the partition $\langle L_1, L_2, L_3 \rangle$ and the partition $\langle X, Y, Z \rangle$ constructed by our algorithm earlier.

Lemma 6. *The set $X \supseteq L_1$ and the set $Z \subseteq L_3$, where X and Z are the sets in the partition $\langle X, Y, Z \rangle$ constructed by our algorithm that builds the graph H.*

The matching M^* need not be A-complete. However it would help us to assume that M^* is A-complete, so we augment M^* by adding $(a, \ell(a))$ edges for every $a \in A$ that is unmatched in M^*. Recall that $\ell(a)$ is the dummy last resort post of a. However the augmented matching M^* need not belong to the graph G' any longer – hence we augment G' also by adding some dummy vertices and some edges as described below.

The augmentation of G' is analogous to Step (III) of our algorithm – we augment G' as follows: let $L_2 = L_2 \cup D$, where $D = \{\ell(a) : a \in A \text{ and } r_a = \infty\}$; if $\mathsf{nbr}(\{a\}) \subseteq L_1$, then add $(a, \ell(a))$ to G'. Thus when compared to G', the augmented G' has some new vertices (all these are dummy last resort posts) and some new edges – each new edge is of the form $(a, \ell(a))$ where $\ell(a)$ is a's only neighbor in $L_2 \cup L_3$. These new edges are enough to show the following lemma.

Lemma 7. *The augmented matching M^* belongs to the augmented graph G'.*

Since the augmented M^* is an A-complete matching, it follows from Lemma 7 that the augmented graph G' admits an A-complete matching. Theorem 5 uses Lemma 6 to show that if the augmented graph G' admits an A-complete matching, then so does the graph H constructed by our algorithm.

Theorem 5. *If H does not admit an A-complete matching, then the augmented graph G' cannot admit an A-complete matching.*

Proof. We will use G' to refer to the *augmented* graph G' in this proof. The rules for adding edges in H and in G' are exactly the same – the only difference is in the partition $\langle X, Y, Z\rangle$ on which H is based vs the partition $\langle L_1, L_2, L_3\rangle$ on which G' is based. If $\langle X, Y, Z\rangle = \langle L_1, L_2, L_3\rangle$, then the graphs H and G' are exactly the same.

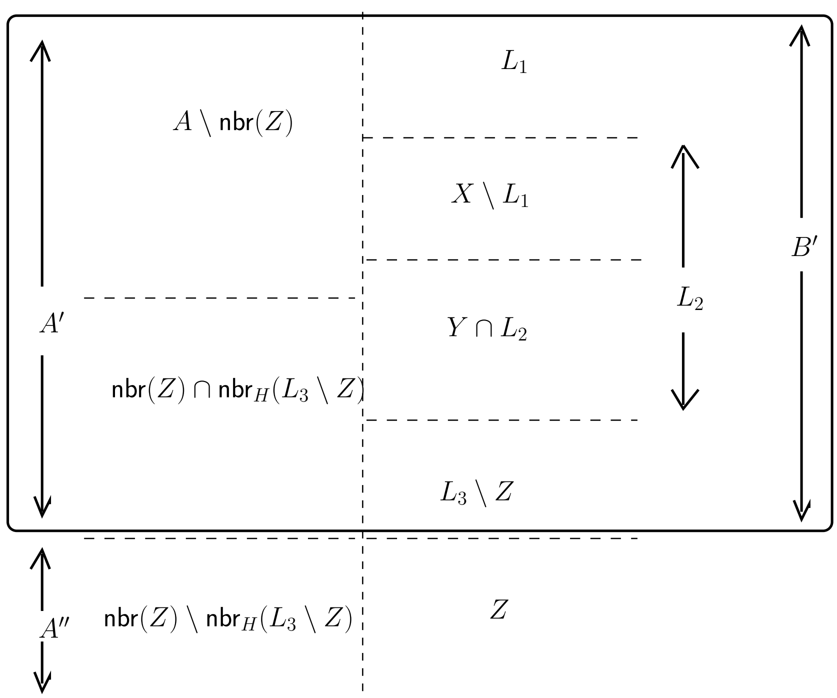

Fig. 2. The part of G' inside the box will be called G'_0. The graph G' has no edge between any applicant in A' and any post in Z.

Refer to Figure 2. This denotes how the partition $\langle X, Y, Z \rangle$ can be modified to the partition $\langle L_1, L_2, L_3 \rangle$. We know from Lemma 6 that $X \supseteq L_1$ and $Z \subseteq L_3$. Consider the subgraph G'_0 of G' induced on the vertex set $A' = (A \setminus \mathsf{nbr}(Z)) \cup (\mathsf{nbr}(Z) \cap \mathsf{nbr}_H(L_3 \setminus Z))$ and $B' = X \cup Y$. This is the part bounded by the box in Figure 2. In our analysis, we can essentially separate G' into G'_0 and the part outside G'_0 due to the following claim that says G' has no edges between A' and Z.

Claim 1 *G' has no edge (a, b) where $a \in A'$ and $b \in Z$.*

Proof. Any applicant $a \in A'$ has to belong to either $A \setminus \mathsf{nbr}(Z)$ or to $\mathsf{nbr}(Z) \cap \mathsf{nbr}_H(L_3 \setminus Z)$ (see Figure 2). There is obviously no edge in G between a vertex in $A \setminus \mathsf{nbr}(Z)$ and any vertex in Z. So suppose $a \in \mathsf{nbr}(Z) \cap \mathsf{nbr}_H(L_3 \setminus Z)$. For $b \in L_3$, if the edge (a, b) is in G', then b has to be a's most preferred post in L_3. We will now show that $b \in L_3 \setminus Z$, equivalently $b \notin Z$. Thus G' has no edge (a, b) where $a \in A'$ and $b \in Z$.

Since $a \in \mathsf{nbr}_H(L_3 \setminus Z)$, the graph H contains an edge between a and some $b' \in L_3 \setminus Z$. Recall that an element of $L_3 \setminus Z$ is a real post in Y. By the rules of including edges in H, it follows that the rank of b' in a's preference list is $\leq r_a$. The entire set L_3 cannot contain any post of rank better than r_a for any $a \in A$ since any post of rank better than r_a in a's list belongs to F while $L_3 \cap F = \emptyset$ (by Lemma 4.1). So b' has rank r_a in a's list. Thus a's most preferred neighbor in L_3 belongs to $L_3 \setminus Z$. □

Let G_0 be the subgraph of G'_0 obtained by deleting from G'_0 the edges that are absent in H. Thus G_0 is a subgraph of both G' and H. The following claim will be useful to us.

Claim 2 *All posts in $(X \setminus L_1) \cup (L_3 \setminus Z)$ are odd/unreachable in G_0. Moreover, every edge (a, b) in G' that is missing in H satisfies $b \in (X \setminus L_1) \cup (L_3 \setminus Z)$.*

Consider the graph G_1 whose edge set is the intersection of the edge sets of G' and H. Equivalently, G_1 can be constructed by adding to the edge set of G_0 the edges incident on $A'' = \mathsf{nbr}(Z) \setminus \mathsf{nbr}_H(L_3 \setminus Z)$ that are present in both G' and H (see Fig. 2). This is due to the fact that G' has no edge in $A' \times Z$.

We claim that all posts in $(X \setminus L_1) \cup (L_3 \setminus Z)$ are odd/unreachable in G_1. This is because Claim 2 tells us that each post in this set is odd/unreachable in G_0 and due to the absence of $A' \times Z$ edges in G', the graph G_1 has no *new* edge (new when compared to G_0) incident on the set A' of applicants in G_0. Hence all posts in $(X \setminus L_1) \cup (L_3 \setminus Z)$ remain odd/unreachable in G_1.

Claim 2 also tells us that all edges in G' that are missing in H are incident on posts in $(X \setminus L_1) \cup (L_3 \setminus Z)$. We know that all these posts are odd/unreachable in G_1, hence G' has no *new* edge (new when compared to G_1) on posts that are *even* in G_1. Thus the size of a maximum matching in G' equals the size of a maximum matching in G_1. This is at most the size of a maximum matching in H, since G_1 is a subgraph of H. Hence if H has no A-complete matching, then neither does G'. □

Theorem 5, along with Lemma 7, finishes the proof of the necessary part of Theorem 4 and this completes the proof of correctness of our algorithm.

It is easy to see that each iteration of our algorithm takes $O(n)$ time (where $|A \cup B| = n$) since it involves finding a maximum matching in a subgraph where each vertex in A has degree at most 2. Thus the running time of our algorithm is $O(n^2)$ and Theorem 1 stated in Section 1 follows.

Conclusions and an open problem. We gave an $O(n^2)$ algorithm for the popular matching problem in $G = (A \cup B, E)$ where vertices in A have strict preference lists while each vertex in B puts all its neighbors into a single tie and $n = |A \cup B|$. Our algorithm needs the preference lists of vertices in A to be strict and the complexity of this problem when ties are allowed in the preference lists of vertices in A is currently unknown.

References

1. D.J. Abraham, R.W. Irving, T. Kavitha, and K. Mehlhorn. Popular matchings. SIAM Journal on Computing, Vol. 37, No. 4, pages 1030–1045, 2007.
2. P. Berman, M. Karpinski, and Alexander D. Scott. Approximation hardness of short symmetric instances of MAX-3SAT. Electronic Colloquium on Computational Complexity Report, number 49, 2003.
3. P. Biró, R.W. Irving, and D.F. Manlove. Popular matchings in the Marriage and Roommates problems. In *Proceedings of the 7th CIAC*: pages 97–108, 2010.
4. A. Dulmage and N. Mendelsohn. Coverings of bipartite graphs. In *Canadian Journal of Mathematics*, 10: pages 517-534, 1958.
5. P. Gärdenfors. Match making: assignments based on bilateral preferences. *Behavioural Science*, 20: pages 166–173, 1975.
6. R.L. Graham, M. Grötschel, and L. Lovasz, editors. *The Handbook of Combinatorics*, chapter 3, Matchings and Extensions, by W. R. Pulleyblank, pages 179–232. North Holland, 1995.
7. C.-C. Huang and T. Kavitha. Popular matchings in the stable marriage problem. In *Proceedings of the 38th ICALP*: pages 666–677, 2011.
8. T. Kavitha. Popularity vs Maximum cardinality in the stable marriage setting. In *Proceedings of the 23rd SODA*: pages 123-134, 2012.
9. T. Kavitha, J. Mestre, and M. Nasre. Popular mixed matchings. In *Proceedings of the 36th ICALP*: pages 574–584, 2009.
10. T. Kavitha and M. Nasre. *Note: Optimal popular matchings.* Discrete Applied Mathematics, 157(14): pages 3181–3186, 2009.
11. M. Mahdian. Random popular matchings. In *Proceedings of the 7th EC*: pages 238–242, 2006.
12. D.F. Manlove and C.T.S. Sng. Popular matchings in the Capacitated House Allocation problem. In *Proceedings of the 14th ESA*: pages 492–503, 2006.
13. R.M. McCutchen. The least-unpopularity-factor and least-unpopularity-margin criteria for matching problems with one-sided preferences. In *Proceedings of the 8th LATIN:* pages 593-604, 2008.
14. E. McDermid and R.W. Irving. Popular matchings: Structure and algorithms. In *Proceedings of the 15th COCOON*: pages 506–515, 2009.
15. J. Mestre. Weighted popular matchings. In *Proceedings of the 33rd ICALP*: pages 715–726, 2006.