Semi-Popular Matchings and Copeland Winners

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

Rohit Vaish
Indian Institute of Technology Delhi
New Delhi, India
rvaish@iitd.ac.in

ABSTRACT

Given a graph $G = (V, E)$ where every vertex has a weak ranking over its neighbors, we consider the problem of computing an *optimal* matching as per agent preferences. Classical notions of optimality such as *stability* and its relaxation *popularity* could fail to exist when G is non-bipartite. In light of the non-existence of a popular matching, we consider its relaxations that satisfy universal existence. We find a positive answer in the form of *semi-popularity*. A matching M is semi-popular if for a majority of the matchings N in G, M does not lose a head-to-head election against N. We show that a semi-popular matching always exists in any graph G and complement this existence result with a fully polynomial-time randomized approximation scheme (FPRAS).

A special subclass of semi-popular matchings is the set of *Copeland winners*—the notion of Copeland winner is classical in social choice theory and a Copeland winner always exists in any voting instance. We study the complexity of computing a matching that is a Copeland winner and show there is no polynomial-time algorithm for this problem unless P = NP.

KEYWORDS

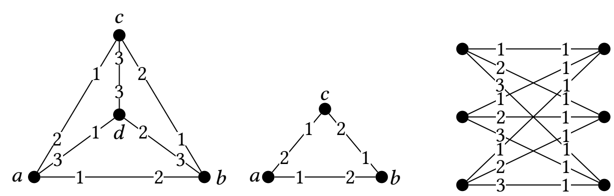
Matchings under preferences, Popularity, FPRAS

ACM Reference Format:
Telikepalli Kavitha and Rohit Vaish. 2023. Semi-Popular Matchings and Copeland Winners. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 9 pages.

1 INTRODUCTION

Matching problems with preferences are of central importance in economics, computer science, and operations research [20, 27, 34]. Over the years, these problems have found several real-world applications such as in school choice [1, 2], labor markets [32, 33], and dormitory assignment [31]. The input in such problems is typically a graph $G = (V, E)$ where the vertices correspond to *agents*, each with a weak ranking of its neighbors. The goal is to divide the agents into pairs, i.e., find a matching in G, while optimizing some criterion of agent satisfaction based on their preferences.

A classical criterion of agent satisfaction in the matching literature is *stability* which requires that there is no *blocking edge*, i.e., no pair of agents simultaneously prefer each other over their prescribed matches [16]. Stability is an intuitively appealing notion, but it can be too demanding in the context of general (i.e., not necessarily bipartite) graphs, also known as *roommates instances*. Indeed, there exist simple roommates instances that do not admit any stable matching (Fig. 1a and Fig. 1b).



(a) (b) (c)

Figure 1: In each figure, the vertices denote the agents, and the number closer to a vertex denotes its rank for the other vertex, where a lower number denotes a more preferred neighbor. (a) An instance with no stable matching but with two popular matchings $\{(a,d),(b,c)\}$ and $\{(a,c),(b,d)\}$. (b) A roommates instance without a popular matching. (c) A bipartite instance (with ties) without a popular matching.

Popularity is a meaningful relaxation of stability that captures welfare in a collective sense [17]. Intuitively, popularity asks for a matching that is not *defeated* by any matching in a head-to-head comparison. More concretely, consider an election in which the matchings play the role of candidates and the agents/vertices act as voters. Given a pair of matchings M and N, a vertex prefers M to N (resp., N to M) if it gets a more preferred partner in M (resp., N).

In the M-vs-N election, every vertex votes for the matching in $\{M, N\}$ that it prefers and it abstains from voting if it is indifferent. Note that being left unmatched is the worst choice for any voter. In this M-vs-N election, let $\phi(M, N)$ be the number of votes for M and let $\phi(N, M)$ be the number of votes for N. Further, let $\Delta(M, N) := \phi(M, N) - \phi(N, M)$. We say that the matching N *defeats* (or is *more popular* than) the matching M if $\Delta(N, M) > 0$. A *popular* matching is one such that there is *no* "more popular" matching.

*Definition 1.1 (**Popular matching**).* A matching M is popular if there is no matching that is more popular than M, i.e., $\Delta(M, N) \geq 0$ for all matchings N.

Thus, a popular matching is a *weak Condorcet winner* in the underlying election among matchings [10].[1] Note that under strict preferences, a stable matching is also popular [7], but a popular matching can exist in instances with no stable matching; e.g., there are two popular matchings $\{(a,d),(b,c)\}$ and $\{(a,c),(b,d)\}$ in Fig. 1a. Thus popularity is a more relaxed criterion than stability—it ensures "collective stability" as there is no matching that makes more agents better off than those who are worse off.

[1] A matching M is a *Condorcet winner* if $\Delta(M, N) > 0$ for every matching $N \neq M$ in G, and a *weak Condorcet winner* if $\Delta(M, N) \geq 0$ for every matching N in G.

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

Unfortunately, the popularity criterion also suffers from similar limitations as stability and more. First, although popular matchings always exist in a bipartite graph with strict preferences [17], they could *fail to exist* with weak rankings, i.e., when preferences include ties (Fig. 1c) or when the graph is non-bipartite (Fig. 1b). Second, determining the existence of a popular matching is known to be NP-hard in roommates instances with strict preferences [14, 19] and in bipartite instances with weak rankings [4, 11].

1.1 Relaxations of Popularity

The non-existence and intractability results for popular matchings motivate the study of relaxations in search of positive results. A natural relaxation of popularity is *low unpopularity* and two of the well-known measures for quantifying the unpopularity of a matching are *unpopularity margin* and *unpopularity factor* [28]. The former bounds the additive gap and the latter bounds the multiplicative gap of the worst pairwise defeat suffered by the matching. Specifically, a matching N that minimizes $\max_M \Delta(M, N)$ is a least unpopularity margin matching and a matching N that minimizes $\max_M(\phi(M, N)/\phi(N, M))$ is a least unpopularity factor matching. A popular matching has unpopularity margin 0 and unpopularity factor exactly 1 (where we interpret $0/0$ as 1).

A matching M with low unpopularity margin/factor may lose many elections, however M can be considered to be *approximately popular* because there are no heavy defeats. It is easy to construct a bipartite instance on n vertices with weak rankings (analogous to the one in Fig. 1c) where every matching has unpopularity margin/factor $\Omega(n)$. So it can be the case that every matching suffers a heavy defeat against some other matching.

An intriguing alternative is to ask for a matching that does not suffer many defeats. If M is a popular matching, the constraints $\Delta(M, N) \geq 0$ have to be satisfied for all matchings N. Finding a matching that satisfies the maximum number of these constraints is NP-hard since that would solve the popular matching problem. What would be the complexity of seeking a matching that violates only a small fraction of these constraints?

Note that there are four matchings in the instance in Fig. 1b (this includes the empty matching) and each of these four matchings loses to at least one matching. Is there always a matching that is guaranteed to *not lose* against a good fraction of matchings, say a majority of the matchings (i.e., not lose against at least $\mu/2$ matchings, where μ is the total number of matchings)? This is precisely the notion of *semi-popularity* that was introduced in [25].

*Definition 1.2 (**Semi-popular matching**).* A matching M is semi-popular if $\Delta(M, N) \geq 0$ for a majority of matchings N in G.

Thus, matching M is *semi-popular* if M is undefeated by a majority of matchings, i.e., M loses to at most $\mu/2$ matchings, where μ is the total number of matchings in G. Note that the three matchings $M_1 = \{(a, b)\}, M_2 = \{(b, c)\}$, and $M_3 = \{(c, a)\}$ in Fig. 1b are semi-popular. Popularity relies on the notion of *majority*, i.e., there is no matching that is preferred to a popular matching by a majority of non-indifferent agents. Semi-popularity takes the notion of majority a step further by asking for a matching undefeated by a majority of matchings.

Regarding a matching undefeated by many matchings to be *approximately popular* is in the same spirit as regarding a matching that does not have many blocking edges to be *approximately stable*. When stable matchings do not exist in a roommates instance $G = (V, E)$, the complexity of finding a matching with the smallest number of blocking edges was studied in [3], where this problem was shown to be NP-hard. Further, even under strict preferences, this problem cannot be approximated within a factor of $|V|^{1-\varepsilon}$ for any $\varepsilon > 0$, unless P = NP [3].

Semi-popular matchings were introduced in [25] to design an efficient bicriteria approximation algorithm for the min-cost popular matching problem in a bipartite instance with strict preferences (this is an NP-hard problem [14]). Stable matchings, and therefore semi-popular matchings, always exist in such instances, but what about general instances? We consider the following questions here.

> *Does a semi-popular matching always exist in any roommates instance with weak rankings? If so, is it easy to find one?*

We show a positive answer to the first question above and an *almost* positive answer to the second question above. Our first observation is the following.

Proposition 1.3. *Every roommates instance where agents have weak rankings admits a semi-popular matching.*

The proof of Proposition 1.3 uses an averaging argument over the space of all matchings and is non-constructive. Thus, the above existence result for semi-popular matchings does not automatically provide an efficient algorithm for computing such a matching. So though we know that semi-popular matchings always exist, the complexity of finding one remains elusive. However we are able to show an efficient randomized algorithm that with high probability (specifically, with probability at least $1 - 1/|V|$) finds an *almost* semi-popular matching.

Theorem 1.4 (**FPRAS for a semi-popular matching**). *Given a roommates instance $G = (V, E)$ with weak rankings and any $\varepsilon > 0$, we can compute in $\text{poly}(|V|, 1/\varepsilon)$ time a matching M such that $\Delta(M, N) \geq 0$ for at least $1/2 - \varepsilon$ fraction of all matchings N in G with high probability.*

Though we do not know how to find a semi-popular matching in polynomial time, we can find in $\text{poly}(|V|, 1/\varepsilon)$ time a matching M that is undefeated by at least $1/2 - \varepsilon$ fraction of the matchings with high probability. It is relevant to note that our algorithm works for *any* input graph G (not necessarily bipartite) and can also accommodate *weak rankings* (i.e., preferences with ties) and, more generally, partial order preferences. Thus, the notion of semi-popularity satisfies universal existence (Proposition 1.3) and there is an efficient algorithm for computing an arbitrarily close approximation to it (Theorem 1.4) in the general roommates model.

For any matching M in the graph G, let $\text{wins}(M)$ (resp., $\text{ties}(M)$) be the number of matchings that are defeated by (resp., tie with) M in their head-to-head election. Deciding if there exists a matching M that satisfies $\text{wins}(M) + \text{ties}(M) = \mu$ (where μ is the total number of matchings) is NP-hard [4] as this is precisely the popular matching problem. In contrast to this, there is a polynomial time algorithm [5] to decide whether there exists a matching M such that $\text{wins}(M) = \mu - 1$, (so M defeats every other matching). Such a matching M is a

Condorcet winner[2] (aka a *strongly popular* matching). Of course, such a matching need not exist in the given instance and this motivates the following natural question:

> *Is there an efficient algorithm to find a matching M that maximizes* $\mathsf{wins}(M)$*?*

Copeland winners. The above question is closely connected to a classical notion called *Copeland winner* in social choice theory. The Copeland rule is a well-known *Condorcet-consistent* voting rule (i.e., it selects the Condorcet winner whenever one exists) that has a long history starting from the 13th century and is named after Arthur H. Copeland [8, 21]. Copeland's method is a natural extension of the Condorcet method and has been called "perhaps the simplest modification" of the Condorcet method [12]. Variants of the Copeland rule are used in sports leagues around the world. Below we define this method in the setting of matchings.

The *Copeland score* of a matching M is defined as $\mathsf{score}(M) := \mathsf{wins}(M) + \mathsf{ties}(M)/2$. That is, the Copeland rule assigns one point for every win, half a point for every tie (this includes comparing the matching against itself), and none for a loss in a head-to-head comparison.

Definition 1.5. A matching with the maximum value of $\mathsf{score}(\cdot)$ is a *Copeland winner*.

Social choice theory tells us that a Copeland winner satisfies many standard desirable properties such as Condorcet-consistency, monotonicity, majority [6, 9, 30], and most importantly, a Copeland winner always exists. It is easy to show that every Copeland winner is semi-popular (see Corollary 2.3). However, unlike semi-popular matchings where we do not know the complexity of computing an *exact* solution, it can be shown that finding an exact Copeland winner is computationally intractable.

THEOREM 1.6 (**HARDNESS OF COPELAND WINNER**). *Unless* $\mathsf{P} = \mathsf{NP}$*, there is no polynomial-time algorithm for finding a Copeland winner in a roommates instance with weak rankings.*

The class of Copeland$^\alpha$ rules generalizes the Copeland rule where wins/ties/losses get the weights of $1/\alpha/0$ for some $0 \leq \alpha \leq 1$ [15, 35]. Let us call a matching with the maximum value of $\mathsf{wins}(M) + \alpha \cdot \mathsf{ties}(M)$ a Copeland$^\alpha$ winner. So a Copeland^0 winner is a matching M that maximizes $\mathsf{wins}(M)$.

It is easy to extend our proof of Theorem 1.6 to show that unless $\mathsf{P} = \mathsf{NP}$, there is no polynomial-time algorithm in a roommates instance with weak rankings to find a Copeland$^\alpha$ winner for any $0 \leq \alpha < 1$. Thus in spite of the tractability of testing if there exists a matching M with $\mathsf{wins}(M) = \mu - 1$ and finding one if so [5], our earlier question on the tractability of finding a matching M that maximizes $\mathsf{wins}(M)$ has a negative answer. That is, under standard complexity-theoretic assumptions, there is no polynomial-time algorithm for finding a matching M that maximizes $\mathsf{wins}(M)$.

Background and related work. There are polynomial-time algorithms known for deciding if a roommates instance with strict preferences admits a stable matching [23]. As mentioned earlier, it is NP-hard to decide if a popular matching exists in bipartite graphs with weak rankings or in non-bipartite graphs with strict preferences [4, 11, 14, 19]. See Fig. 2 for an illustration of relaxations among the various notions mentioned here.

[2]Recall that a Condorcet winner is one that wins every head-to-head election (see footnote 1).

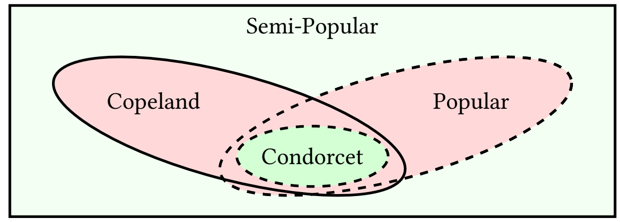


Figure 2: Relationship among the various notions mentioned in this paper for the setting of roommates instances with weak preferences. A solid (resp., dashed) border indicates that the property is guaranteed to exist (resp., could fail to exist). Computational tractability (resp., intractability) is indicated via green (resp., red) color. We use a lighter shade of green for the outer box to denote tractability of the "almost" variant of the problem.

Algorithmic aspects of popular matchings have been extensively studied in the last fifteen years within theoretical computer science and combinatorial optimization literature and we refer to [10] for a survey. The special case of popular matchings in bipartite graphs with strict preferences has been of particular interest, where such matchings always exist, and the work that is closest to ours here is [25], where semi-popular matchings were introduced.

1.2 Our Techniques

We establish our algorithmic result (Theorem 1.4) by using a sampling-based procedure (Algorithm 1). Sampling matchings from a near-uniform distribution is well-studied in theoretical computer science [29], however it has not really been explored much in computational social choice. We use the sampling approach to search for an almost semi-popular matching in the exponentially large space of all matchings in $G = (V, E)$. Specifically, we draw two independent samples, each containing $\Theta(\log |V|/\varepsilon^2)$ matchings, from a distribution that is $\varepsilon/4$-close to the uniform distribution in total variation distance.[3] By the seminal result of Jerrum and Sinclair [24], there is an algorithm with running time $\mathrm{poly}(|V|, \log(1/\varepsilon))$ for generating a sample from such a distribution.

We then pit the two random samples against each other by evaluating all head-to-head elections between pairs of matchings, one matching from each sample, and pick the one with the highest Copeland score in these elections. It is easy to see that the chosen matching is semi-popular 'on the sample' (Lemma 2.4). By a standard concentration argument, we are able to show that this matching is almost semi-popular with respect to *all* the matchings in the given instance with high probability (Lemma 2.6).

Our hardness result for Copeland winners (Theorem 1.6) uses a reduction from VERTEX COVER.[4] At a high level, our construction is inspired by a construction in [13] that used a far simpler instance to show that the extension complexity of the bipartite

[3]Informally, the total variation distance between two probability distributions is the largest possible difference between the probabilities that the two probability distributions can assign to the same event.

[4]In the VERTEX COVER problem, the input is a graph $G = (V, E)$ and an integer k, and the goal is to determine if there is a subset $S \subseteq V$ of at most k vertices such that every edge in E is incident to some vertex in S. This is an NP-hard problem.

popular matching polytope is near-exponential. Our construction, on the other hand, is considerably more involved and we construct a non-bipartite instance G with weak rankings. What makes our construction particularly tricky is that *every* Copeland winner in G has to correspond to a minimum vertex cover in the input instance H. In general, we do not know how to characterize Copeland winners in G. In fact, we do not even know how to test if a given matching is a Copeland winner or not. This makes our reduction challenging. We use the LP framework for popular matchings to analyze Copeland winners and this leads to our hardness proof. This proof is given in Section 3.

2 COMPUTING AN ALMOST SEMI-POPULAR MATCHING

Our input is a roommates instance $G = (V, E)$ on n vertices where every vertex has a weak ranking over its neighbors. While it is easy to construct roommates instances that admit no popular matchings (see Fig. 1b), semi-popular matchings are always present in any instance G, as we show below.

Let μ be the total number of matchings in G. For any matching M in the graph G, recall that $\mathsf{wins}(M)$ (resp., $\mathsf{ties}(M)$) is the number of matchings that are defeated by (resp., tie with) M in their head-to-head election. A matching M is popular if and only if $\mathsf{wins}(M) + \mathsf{ties}(M) = \mu$ and M is semi-popular if and only if $\mathsf{wins}(M) + \mathsf{ties}(M) \geq \mu/2$. Recall that $\mathsf{score}(M) = \mathsf{wins}(M) + \mathsf{ties}(M)/2$. The following lemma immediately implies Proposition 1.3.

Lemma 2.1. *Every roommates instance (where agents have weak rankings) admits a matching M with $\mathsf{score}(M) \geq \mu/2$.*

Lemma 2.1 is a straightforward consequence of the following observation which, in turn, follows easily from pigeonhole principle.[5]

Proposition 2.2. *In any directed multigraph with n vertices and m edges, there exists some vertex with outdegree at least $\lceil m/n \rceil$.*

Proof. (of Lemma 2.1) Construct a directed multigraph whose vertices correspond to the matchings. Between any pair of vertices (u, v) in this multigraph, add *two* directed edges from u to v if the matching corresponding to u defeats the one corresponding to v. Otherwise, if the corresponding matchings are tied, add one directed edge from u to v and another from v to u.

Finally, add a directed edge (self-loop) from every vertex to itself.

Observe that a vertex in the multigraph constructed above has outdegree d if and only if the corresponding matching has score $d/2$. The multigraph has $\mu(\mu-1)+\mu$ edges. Thus, by Proposition 2.2, there must exist a vertex with outdegree at least μ. Then, the corresponding matching, say M, must have $\mathsf{score}(M)$ at least $\mu/2$. □

Corollary 2.3. *Every Copeland winner is a semi-popular matching.*

Proof. Let M be a Copeland winner. Then $\mathsf{score}(M) \geq \mu/2$ (by Lemma 2.1). Since $\mathsf{score}(M) = \mathsf{wins}(M) + \mathsf{ties}(M)/2 \leq \mathsf{wins}(M) + \mathsf{ties}(M)$, we have $\mathsf{wins}(M)+\mathsf{ties}(M) \geq \mu/2$. Thus M is semi-popular. □

[5] We thank an anonymous reviewer for suggesting to use this fact to prove Lemmas 2.1 and 2.4.

ALGORITHM 1: An FPRAS for Semi-Popular Matchings.

Input: A graph $G = (V, E)$ on n vertices and a set of weak rankings for every vertex $v \in V$.
Parameters: $\varepsilon > 0$.
Output: A matching in G.

Produce two independent samples $\mathcal{S}_0$ and $\mathcal{S}_1$ of $k = \lceil (32 \ln n/\varepsilon^2) \rceil$ matchings where each matching is chosen from a distribution that is $\varepsilon/4$-close to the uniform distribution (on all matchings in G) in total variation distance.
foreach *matching* $M \in \mathcal{S}_0 \cup \mathcal{S}_1$ **do**
  Initialize $\mathsf{wins}'_M = \mathsf{ties}'_M = 0$.
foreach *matching* $M \in \mathcal{S}_0$ **do**
  foreach *matching* $N \in \mathcal{S}_1$ **do**
    if $\Delta(M, N) > 0$ then $\mathsf{wins}'_M = \mathsf{wins}'_M + 1$.
    if $\Delta(M, N) = 0$ then $\mathsf{ties}'_M = \mathsf{ties}'_M + 1$ and $\mathsf{ties}'_N = \mathsf{ties}'_N + 1$.
    if $\Delta(M, N) < 0$ then $\mathsf{wins}'_N = \mathsf{wins}'_N + 1$.
return *a matching* $S \in \mathcal{S}_0 \cup \mathcal{S}_1$ *with the maximum value of* $\mathsf{wins}'_S + \mathsf{ties}'_S/2$.

Our algorithm. We will now show an FPRAS for computing a semi-popular matching. In fact, we will construct a matching M with $\mathsf{score}(M) \geq (1 - \varepsilon) \cdot \mu/2$ with high probability. In order to construct such a matching, as mentioned earlier, we will use the classical result from [24] that shows an algorithm with running time $\mathsf{poly}(n, \log(1/\varepsilon))$ to sample matchings from a distribution ε-close to the uniform distribution in total variation distance (see [24, Corollary 4.3]).

Our algorithm is presented as Algorithm 1. The input to our algorithm is a roommates instance $G = (V, E)$ on n vertices along with a parameter $\varepsilon > 0$. It computes two independent samples $\mathcal{S}_0$ and $\mathcal{S}_1$ of $k = \lceil (32 \ln n/\varepsilon^2) \rceil$ matchings—each from a distribution $\varepsilon/4$-close to the uniform distribution (on all the matchings in G) in total variation distance.

For $M \in \mathcal{S}_0$ (resp., $\mathcal{S}_1$), let wins'_M be the number of matchings in $\mathcal{S}_1$ (resp., $\mathcal{S}_0$) that M wins against and let ties'_M be the number of matchings in $\mathcal{S}_1$ (resp., $\mathcal{S}_0$) that M ties with.

Our algorithm computes $\mathsf{score}'_S = \mathsf{wins}'_S + \mathsf{ties}'_S/2$ for each $S \in \mathcal{S}_0 \cup \mathcal{S}_1$. It returns a matching in $\mathcal{S}_0 \cup \mathcal{S}_1$ with the maximum value of score'. We will now show that such a matching has a high Copeland score on the sample. Recall that $|\mathcal{S}_0| = |\mathcal{S}_1| = k$.

Lemma 2.4. *If S^* is the matching returned by Algorithm 1, then* $\mathsf{score}'_{S^*} \geq k/2$.

Proof. Consider a bipartite graph whose left and right vertex sets correspond to the matchings in $\mathcal{S}_0$ and $\mathcal{S}_1$, respectively; thus, there are $2k$ vertices overall. If the i^{th} matching in $\mathcal{S}_0$ defeats (resp., is defeated by) the j^{th} matching in $\mathcal{S}_1$, then add two directed edges from the i^{th} left vertex to the j^{th} right vertex (resp., from the j^{th} right vertex to the i^{th} left vertex). If the two matchings are tied, add two directed edges—one in either direction—between the corresponding vertices.

Observe that a vertex in the bipartite graph constructed above has outdegree d if and only if the corresponding matching, say M, has $\mathsf{score}'_M \geq d/2$. There are $2k^2$ edges in the graph. Thus, by Proposition 2.2, there must exist a vertex with outdegree at least k, and therefore a matching, say M, with $\mathsf{score}'_M \geq k/2$. Thus, the matching S^* returned by Algorithm 1 has $\mathsf{score}'_{S^*} \geq \mathsf{score}'_M \geq k/2$. □

We will show that the on-sample guarantee of Lemma 2.4 carries over, with high probability, to the entire set of matchings. This proof makes use of a tail bound for the random variable score'_S corresponding to the on-sample Copeland score of any fixed matching $S \in \mathcal{S}_0 \cup \mathcal{S}_1$.

Lemma 2.5. *Let $S \in \mathcal{S}_0 \cup \mathcal{S}_1$ be any fixed matching sampled by Algorithm 1. Then the probability that $\mathsf{score}'_S \geq k \cdot (\mathsf{score}(S)/\mu + \varepsilon/2)$ is at most $1/n$.*

Proof. Assume without loss of generality that $S \in \mathcal{S}_1$. If $\mathcal{S}_0$ was a set of k matchings chosen uniformly at random from the set of all matchings in G, then the probability that S defeats any matching in $\mathcal{S}_0$ is $\mathsf{wins}(S)/\mu$ and the probability that it ties with any matching in $\mathcal{S}_0$ is $\mathsf{ties}(S)/\mu$.

However, the matchings in $\mathcal{S}_0$ are sampled from a distribution $\varepsilon/4$-close to the uniform distribution in total variation distance. So, the probability that S defeats any matching in $\mathcal{S}_0$ is at most $\mathsf{wins}(S)/\mu + \varepsilon/4$ and the probability that it ties with any matching in $\mathcal{S}_0$ is at most $\mathsf{ties}(S)/\mu + \varepsilon/4$.

Observe that $\mathsf{score}'_S = X_1 + \ldots + X_k$, where, for each i, the random variable $X_i \in \{0, \frac{1}{2}, 1\}$ denotes whether S loses/ties/wins against the i-th matching in $\mathcal{S}_0$. Note that $\mathbb{E}[X_i] \leq \mathsf{wins}(S)/\mu + \mathsf{ties}(S)/2\mu + 3\varepsilon/8$ for each i. Since $\mathsf{score}(S) = \mathsf{wins}(S) + \mathsf{ties}(S)/2$, linearity of expectation gives

$$\mathbb{E}\left[\mathsf{score}'_S\right] \;\leq\; k \cdot \left(\frac{\mathsf{score}(S)}{\mu} + \frac{3\varepsilon}{8}\right). \tag{1}$$

In light of Equation (1), to prove the lemma it suffices to bound the probability of the event that

$$\mathsf{score}'_S - \mathbb{E}[\mathsf{score}'_S] \geq k \cdot \varepsilon/8,$$

or, equivalently,

$$\mathsf{score}'_S/k - \mathbb{E}[\mathsf{score}'_S/k] \geq \varepsilon/8.$$

For this, we will use Hoeffding's inequality [22]. Recall that if $X_1, \ldots, X_k$ are bounded independent random variables such that $X_i \in [0,1]$ for all $i \in [k]$ and $Y := (X_1 + \cdots + X_k)/k$, then Hoeffding's inequality says that for any $t \geq 0$,

$$\Pr\left[Y - \mathbb{E}[Y] \;\geq\; t\right] \;\leq\; e^{-2kt^2}.$$

Applying the above inequality for $Y := \mathsf{score}'_S/k$ and $t := \varepsilon/8$, we get that

$$\Pr\left[\mathsf{score}'_S/k - \mathbb{E}[\mathsf{score}'_S/k] \;\geq\; \varepsilon/8\right] \;\leq\; e^{-k\varepsilon^2/32}. \tag{2}$$

Substituting $k = \lceil (32 \ln n/\varepsilon^2) \rceil$ in Equation (2), we get that the right-hand side is at most $1/n$. Thus, with probability at least $1 - 1/n$, we have $\mathsf{score}'_S < \mathbb{E}[\mathsf{score}'_S] + k \cdot \varepsilon/8$. Using the upper bound in Equation (1), it follows that $\mathsf{score}'_S < k \cdot (\mathsf{score}(S)/\mu + \varepsilon/2)$ with probability at least $1 - 1/n$. □

By Lemmas 2.4 and 2.5, the matching S^* returned by our algorithm satisfies $k/2 \;\leq\; \mathsf{score}'_{S^*} \;<\; k \cdot (\mathsf{score}(S^*)/\mu + \varepsilon/2)$ with high probability. Thus Lemma 2.6 follows.

Lemma 2.6. *If S^* is the matching returned by Algorithm 1, then $\mathsf{score}(S^*) > (1 - \varepsilon) \cdot \mu/2$ with high probability.*

So our algorithm computes a matching whose Copeland score is more than $(1 - \varepsilon) \cdot \mu/2$ with high probability. Its running time is polynomial in n and $1/\varepsilon$. Since $\mathsf{wins}(S^*) + \mathsf{ties}(S^*) \geq \mathsf{score}(S^*)$, Theorem 1.4 follows.

Theorem 1.4 (**FPRAS for a semi-popular matching**). *Given a roommates instance $G = (V, E)$ with weak rankings and any $\varepsilon > 0$, we can compute in $\mathsf{poly}(|V|, 1/\varepsilon)$ time a matching M such that $\Delta(M, N) \geq 0$ for at least $1/2 - \varepsilon$ fraction of all matchings N in G with high probability.*

3 FINDING A COPELAND WINNER: A HARDNESS RESULT

In Section 3.1 we will first give a high-level overview of the proof of Theorem 1.6 which states that under standard complexity-theoretic assumptions, there is no polynomial-time algorithm for finding a Copeland winner. Several details are given in Section 3.2 and the remaining details can be found in the full version of the paper [26].

3.1 A High-Level Overview

Given a VERTEX COVER instance $H = (V_H, E_H)$, we will construct a roommates instance $G = (V, E)$ such that any Copeland winner in G will correspond to a minimum vertex cover in H. We will assume that the vertices in the VERTEX COVER instance are indexed as $1, 2, \ldots, n$, i.e., $V_H = \{1, \ldots, n\}$. Specifically,

- for every vertex $i \in V_H$, there is a gadget Z_i in G on 4 *main* vertices a_i, b_i, a'_i, b'_i and 100 *auxiliary* vertices $u_i^0, \ldots, u_i^{99}$, and
- for every edge $e \in E_H$, there is a gadget Y_e in G on 6 *main* vertices $s_e, t_e, s'_e, t'_e, s''_e, t''_e$ along with 8 *auxiliary* vertices $v_e, v'_e, w_e, w'_e, c_e, d_e, c'_e, d'_e$ (see Fig. 3).

The gadgets. The preferences of the vertices in the vertex gadget Z_i and the edge gadget Y_e are shown in Fig. 3. Observe that in the vertex gadget Z_i, all the vertices $u_i^0, \ldots, u_i^{99}$ are tied at the third position (which is the last acceptable position) in a_i's preference order. Similarly, in the edge gadget Y_e (where $e = (i, j)$), the vertices b_i and c_e are tied in d_e's preference list and the vertices b_j and c'_e are tied in d'_e's preference list.

Red state vs blue state. Let M be a matching in G. We will say a vertex gadget Z_i is in red state in M if $\{(a_i, b_i), (a'_i, b'_i)\} \subset M$ and in blue state if $\{(a_i, b'_i), (a'_i, b_i)\} \subset M$.

A high-level overview of our hardness reduction. We will show that any Copeland winner M in G has the following properties.

- M does not use any *inter-gadget* edge, i.e., a shared edge between a vertex gadget and an edge gadget.
- For any vertex $i \in V_H$, its vertex gadget Z_i is either in red state or in blue state in M.
- For any edge $e = (i, j)$, at least one of Z_i, Z_j has to be in blue state in M.
- The vertices i with vertex gadgets Z_i in blue state in M form a *minimum vertex cover* in H.

The above properties will imply Theorem 1.6. For the sake of readability, we do not include in Section 3.2 the proofs of all the lemmas that together imply Theorem 1.6; the proofs of lemmas

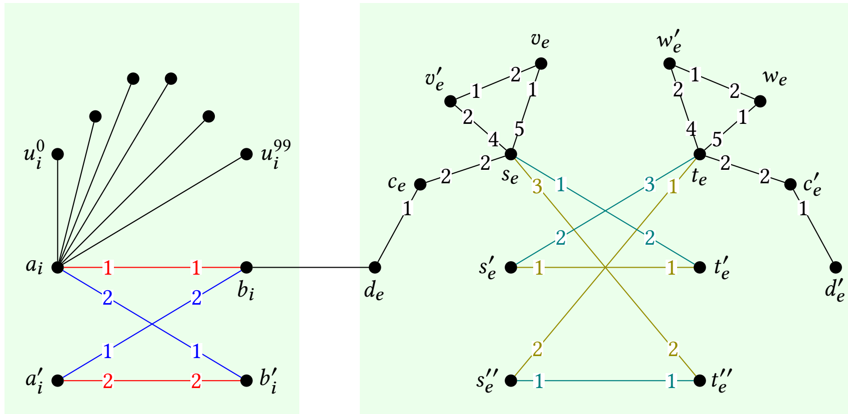


Figure 3: (Left) The vertex gadget Z_i. The red colored edges within Z_i indicate a stable matching. (Right) The edge gadget Y_e on 14 vertices. All unnumbered edges incident to a vertex should be interpreted as "tied for the last acceptable position". The edges (c_e, d_e), (v_e, v_e'), (w_e, w_e'), (c_e', d_e') along with the olive-colored (resp., teal-colored) edges define the matching F_e (resp., L_e). Here $e = (i, j)$ and (b_i, d_e) is an inter-gadget edge.

marked by an asterisk (★) are given in the full version of the paper [26]. The hardness of finding a Copeland winner will follow from Lemmas 3.7 and 3.8 which are proved in Section 3.2.

Theorem 1.6 (**Hardness of Copeland winner**). *Unless* P = NP*, there is no polynomial-time algorithm for finding a Copeland winner in a roommates instance with weak rankings.*

3.2 Proof of Theorem 1.6

Let $e \in E$. In the edge gadget Y_e (see Fig. 3), we will find it convenient to define the matchings

$$F_e = \{(s_e, t_e''), (s_e', t_e'), (s_e'', t_e), (v_e, v_e'), (w_e, w_e'), (c_e, d_e), (c_e', d_e')\}$$

and

$$L_e = \{(s_e, t_e'), (s_e', t_e), (s_e'', t_e''), (v_e, v_e'), (w_e, w_e'), (c_e, d_e), (c_e', d_e')\}.$$

These are highlighted with olive and teal colors in Fig. 3, respectively. The following two lemmas will be very useful to us. Lemma 3.1 discusses the number of matchings that tie with the matching F_e (or L_e) in the subgraph restricted to the edge gadget Y_e (see Fig. 3).

Lemma 3.1 (★). *In the subgraph restricted to Y_e, there are exactly 10 matchings that are tied with F_e and no matching defeats F_e. Furthermore, an analogous statement holds for the matching L_e.*

Next, Lemma 3.2 shows that there is *no* matching within the subgraph restricted to Y_e that "does better" than F_e or L_e in terms of the number of matchings that defeat or tie with it.

Lemma 3.2 (★). *For any matching T_e in the subgraph restricted to Y_e, there are at least 10 matchings within this subgraph that either defeat or tie with T_e.*

Recall the red/blue states of a vertex gadget described in Section 3. Lemma 3.3 stated below shows that a Copeland winner matching that does not use any inter-gadget edge must have each vertex gadget in either red or blue state.

Lemma 3.3 (★). *Let M be a Copeland winner in G. If M does not use any inter-gadget edge, then in any vertex gadget Z_i, either $\{(a_i, b_i), (a_i', b_i')\} \subset M$ or $\{(a_i, b_i'), (a_i', b_i)\} \subset M$.*

Observation 1. *Consider the subgraph induced on Z_i.*

- *The red matching $R_i = \{(a_i, b_i), (a_i', b_i')\}$ is tied with 2 matchings in this subgraph. These are R_i itself and the blue matching $B_i = \{(a_i, b_i'), (a_i', b_i)\}$.*
- *The blue matching $B_i = \{(a_i, b_i'), (a_i', b_i)\}$ is tied with 3 matchings in this subgraph: these are B_i itself, the red matching R_i, and the matching $\{(a_i, b_i)\}$.*

Moreover, no matching in this subgraph defeats either the red matching R_i or the blue matching B_i.

It is straightforward to verify the above observation. For any gadget X, let $M \cap X$ denote the edges of matching M in the subgraph restricted to X.

Lemma 3.4 (★). *Let $e = (i, j) \in E$. Let M be any matching in G such that both Z_i and Z_j are in red state in M. Then there are at least 100 matchings within $Y_e \cup Z_i \cup Z_j$ that defeat or tie with $M \cap (Y_e \cup Z_i \cup Z_j)$.*

Next, we will show in Lemma 3.6 that in a Copeland winner matching that does not use any inter-gadget edge, for any edge gadget, at least one of its adjacent vertex gadgets must be in blue state. The proof of Lemma 3.6 will make use of the following key technical lemma.

Lemma 3.5 (★). *Let M^* be any matching in G that satisfies the following three conditions:*

1. *Every vertex gadget is either in red or blue state.*
2. *For every edge gadget, at least one of its adjacent vertex gadgets is in blue state.*
3. *For each edge $e = (i, j)$ where $i < j$, if the vertex gadget Z_i is in blue state, then $M^* \cap Y_e = F_e$ otherwise $M^* \cap Y_e = L_e$.*

Then (i) M^* is popular in G and (ii) any matching that contains an inter-gadget edge loses to M^*.

We showed in Lemma 3.3 that the first property stated in Lemma 3.5 is obeyed by any Copeland winner that does not use any inter-gadget edge. Lemma 3.6 stated below will show that the second property stated in Lemma 3.5 is also obeyed by any Copeland winner that does not use any inter-gadget edge.

Lemma 3.6. *Let M be a Copeland winner in G. If M does not use* any *inter-gadget edge, then, for every edge $e = (i, j)$, at least one of Z_i, Z_j has to be in blue state in M.*

Proof. Suppose, for contradiction, that both Z_i and Z_j are in red state in M for some edge $e = (i, j)$. Let $\mathcal{X}$ denote the set of all vertex and edge gadgets in the matching instance. Consider a partitioning of $\mathcal{X}$ into *single* gadgets and *auxiliary* gadgets. Each auxiliary gadget is a triple of an edge and two adjacent vertex gadgets (Y_e, Z_i, Z_j) where $e = (i, j)$ is an edge such that both Z_i and Z_j are in red state in M. While there is such an edge e with both its vertex gadgets in red state and these vertex gadgets are "unclaimed" by any other edge, the edge e claims both its vertex gadgets and makes an auxiliary gadget out of these three gadgets. All the remaining vertex and edge gadgets are classified as single gadgets, including those edges one or both of whose endpoints have been claimed by some other edge(s). Observe that

$$\text{loss}(M) + \text{ties}(M) \geq \Pi_{X \in \mathcal{X}}(\text{loss}(M \cap X) + \text{ties}(M \cap X))$$

where X is any single or auxiliary gadget, and loss$(M \cap X)$ (resp., ties$(M \cap X)$) is the number of matchings that defeat (resp., tie with) $M \cap X$ within the gadget X. Also, loss(M) (resp., ties(M)) is the number of matchings that defeat (resp., tie with) the matching M.

Consider any auxiliary gadget $X = (Y_e, Z_i, Z_j)$. We know that loss$(M \cap X)$ + ties$(M \cap X)$ is at least 100 (by Lemma 3.4). Thus, we have

$$\text{loss}(M) + \text{ties}(M) \;\geq\; 2^{n'} \cdot 3^{n''} \cdot 10^{m'} \cdot 100^t, \quad (3)$$

where n' (resp., n'') is the number of vertices present as single gadgets in red (resp., blue) state, m' is the number of edges that are present as single gadgets, and t is the number of auxiliary gadgets in the aforementioned partition. Note that we used Lemma 3.2 here in bounding loss$(M \cap Y_e)$ + ties$(M \cap Y_e)$ by 10 for every edge gadget Y_e that is present as a single gadget, and used Lemma 3.3 and Observation 1 in bounding loss$(M \cap Z_i)$ + ties$(M \cap Z_i)$ by 2 (resp., 3) for every vertex gadget Z_i (acting as single gadget) that is in red (resp., blue) state.

We will now construct another matching with Copeland score higher than that of M to establish the desired contradiction. Let M^* be the matching obtained by converting both Z_i and Z_j in each auxiliary gadget (Y_e, Z_i, Z_j) in our partition above into blue state and let $M^* \cap Y_e$ be either F_e or L_e (it does not matter which). For every edge $e' = (i', j')$ such that $Y_{e'}$ is present as a single gadget, observe that at least one of i', j' is either in blue state or in an auxiliary gadget. If it is $\min\{i', j'\}$ that is in blue state/auxiliary gadget, then let $M^* \cap Y_{e'} = F_{e'}$, else let $M^* \cap Y_{e'} = L_{e'}$. Any vertex gadget that is present as a single gadget remains in its original red or blue state.

Notice that M^* satisfies the conditions in Lemma 3.5. So M^* is popular in G, i.e., loss$(M^*) = 0$, and any matching that ties with M^* must not use any inter-gadget edge. Therefore, the number of matchings that tie with M^* across the entire graph G is simply the product of matchings that tie with it on individual single and auxiliary gadgets. Hence,

$$\text{loss}(M^*) + \text{ties}(M^*) \;=\; \text{ties}(M^*) \;=\; 2^{n'} \cdot 3^{n''} \cdot 10^{m'} \cdot 90^t, \quad (4)$$

where n', n'', m', and t are defined as in (3). Note that the bound of 90 for an auxiliary gadget in (4) follows from taking the product of 3, 3, and 10, which is the number of matchings that tie with M^* in the two vertex gadgets and their common edge gadget (see Observation 1 and Lemma 3.1).

Recall that score(N) = wins(N) + ties$(N)/2 = \mu$ − loss(N) − ties$(N)/2$ for any matching N. Comparing (3) and (4) along with the fact that loss$(M) \geq 0 =$ loss(M^*), we have score$(M^*) >$ score(M), as long as there is even a single edge (i, j) such that both Z_i and Z_j are in red state in M. Indeed,

$$\text{loss}(M^*) + \text{ties}(M^*)/2 \;=\; 2^{n'-1} \cdot 3^{n''} \cdot 10^{m'} \cdot 90^t, \text{ and}$$

$$\begin{aligned}\text{loss}(M) + \text{ties}(M)/2 \;&\geq\; ((\text{loss}(M) + \text{ties}(M))\,/2 \\ &\geq\; 2^{n'-1} \cdot 3^{n''} \cdot 10^{m'} \cdot 100^t.\end{aligned}$$

This contradicts the fact that M is a Copeland winner. Hence, for every edge $e = (i, j)$, at least one of Z_i, Z_j must be in blue state in M. This proves Lemma 3.6. □

In Lemma 3.6, we showed that for any Copeland winner M that does not use any inter-gadget edge, the vertices whose gadgets are in blue state in M constitute a vertex cover in H. The next result shows that the set of such vertices is, in fact, a *minimum* vertex cover.

Lemma 3.7. *Let M be a Copeland winner in G. If M does not use* any *inter-gadget edge, then the vertices whose gadgets are in blue state in M constitute a minimum vertex cover in H.*

Proof. We know from Lemma 3.6 that loss(M) + ties$(M) \geq \Pi_S(\text{loss}(M \cap S) + \text{ties}(M \cap S))$ where the product is over all gadgets S. Further, from Lemma 3.3, Lemma 3.2, and Observation 1, we know that the right hand side in the above inequality is at least $2^{n-k} \cdot 3^k \cdot 10^m$, where k is the number of vertex gadgets in blue state and n (resp., m) is the number of vertices (resp., edges) in the VERTEX COVER instance H. Thus, score$(M) = \mu$ − loss(M) − ties$(M)/2 \leq \mu - 2^{n-k-1} \cdot 3^k \cdot 10^m$. Moreover, $k \geq |C|$, where C is a minimum vertex cover in H (by Lemma 3.6).

Let us construct a matching M_C where the vertex gadgets corresponding to the vertices in the minimum vertex cover C are in blue state, those corresponding to the remaining vertices are in red state, and for every edge $e = (i, j)$, if $\min\{i, j\}$ is in blue state then let $M_C \cap Y_e = F_e$, else $M_C \cap Y_e = L_e$. Then, the matching M_C satisfies the conditions of Lemma 3.5. Therefore, by a similar argument as in the proof of Lemma 3.6, we get that score$(M_C) = \mu - 2^{n-c-1} \cdot 3^c \cdot 10^m$, where $|C| = c$. Thus score$(M_C) >$ score(M) if $c < k$. Since score(M) has to be the highest among all matchings, it follows that $c = k$. In other words, the set of vertices whose gadgets are in blue state in M constitute a minimum vertex cover in H. □

Finally, we show that our assumption that a Copeland winner does not use an inter-gadget edge always holds. Thus Theorem 1.6 stated in Section 1 follows.

Lemma 3.8. *If M is a Copeland winner in G, then M does not use any inter-gadget edge.*

Proof. The proof is similar to that of Lemma 3.6. Below, we will outline the main steps involved in the proof. Suppose, for contradiction, that M uses an inter-gadget edge, say (b_i, d_e) (see Fig. 3). Let $\mathcal{X}$ denote the set of all vertex and edge gadgets in the matching instance. Consider a partitioning of $\mathcal{X}$ into *single*, *double*, *triple*, and *auxiliary* gadgets as follows:

- Each *double* (resp., *triple*) gadget is a pair of an edge gadget and an adjacent vertex gadget (Y_e, Z_i) (resp., a triple of an edge and two adjacent vertex gadgets (Y_e, Z_i, Z_j)) where $e = (i, j)$ is an edge such that M contains an inter-gadget edge between Y_e and Z_i (resp., two inter-gadget edges between Y_e and each of Z_i and Z_j). While there is an edge gadget that shares an inter-gadget edge with one of (resp., both) its adjacent vertex gadgets, we make a double (resp., triple) gadget out of these two (resp., three) gadgets. Note that a vertex gadget can be included in at most one double/triple gadget in this manner.
- Next, if there is an edge gadget that is adjacent to two vertex gadgets both of which are in red state and are still unclaimed by any edge gadget, then this edge gadget claims both these vertex gadgets; such a triple of edge gadget and its adjacent vertex gadgets is classified as an *auxiliary* gadget. Note that such an edge gadget does not share an inter-gadget edge with either of the vertex gadgets.
- All remaining vertex and edge gadgets are classified as *single* gadgets.

Observe that

$$\text{loss}(M) + \text{ties}(M) \geq \Pi_{X \in \mathcal{X}}(\text{loss}(M \cap X) + \text{ties}(M \cap X)),$$

where X denotes any single, double, triple or auxiliary gadget.

Consider any double or triple gadget X. Since the matching M uses at least one inter-gadget edge (b_i, d_e) in X, the vertex a_i must be matched with either the vertex b'_i or one of the vertices in $u_i^0, \ldots, u_i^{99}$ (otherwise, if a_i is unmatched, then M will not be *Pareto optimal* and it is easy to see that every Copeland winner has to be Pareto optimal).[6] In both cases, the matching $\{(a_i, u_i^{k'}), (a'_i, b'_i)\}$ is tied with M where $k' \in \{0, \ldots, 99\}$. Thus, there are at least 100 matchings that defeat or tie with $M \cap (Y_e \cup Z_i)$. In other words, for every double or triple gadget X, the value of $\text{loss}(M \cap X) + \text{ties}(M \cap X)$ is at least 100. We therefore have

$$\text{loss}(M) + \text{ties}(M) \;\geq\; 2^{n'} \cdot 3^{n''} \cdot 10^{m'} \cdot 100^{t_2} \cdot 100^{t_3} \cdot 100^{a}, \quad (5)$$

where n' (resp., n'') is the number of vertices present as single gadgets in red (resp., blue) state, m' is the number of edges that are present as single gadgets, t_2 is the number of double gadgets, t_3 is the number of triple gadgets, and a is the number of auxiliary gadgets in the aforementioned partition. As done in Lemma 3.6, we once again used Lemma 3.2 in bounding $\text{loss}(M \cap Y_e) + \text{ties}(M \cap Y_e)$ by 10 for every edge gadget Y_e that is present as a single gadget, and used Observation 1 in bounding $\text{loss}(M \cap Z_i) + \text{ties}(M \cap Z_i)$ by 2 (resp., 3) for every vertex gadget Z_i (acting as single gadget) that is in red (resp., blue) state. Additionally, we used Lemma 3.4 to obtain the corresponding bound for an auxiliary gadget.

We will now construct an alternative matching M^* that has a higher Copeland score than M to derive the desired contradiction. Starting with M, let us remove any inter-gadget edges from each double/triple gadget and convert both $M \cap Z_i$ and $M \cap Z_j$ in the triple gadget (or just $M \cap Z_i$ in case of a double gadget) to blue state and replace $M \cap Y_e$ with F_e. Additionally, for each auxiliary gadget (Y_e, Z_i, Z_j), we convert both $M \cap Z_i$ and $M \cap Z_j$ to blue state and replace $M \cap Y_e$ with F_e.

Note that for any edge $e = (i, j)$ in a single gadget, either Z_i or Z_j is now in blue state. If it is $\min\{i, j\}$ that is in blue state, then let $M^* \cap Y_e = F_e$, else let $M^* \cap Y_e = L_e$. The rest of the gadgets are in the same state as under M.

Notice that M^* satisfies the conditions in Lemma 3.5. Thus, $\text{loss}(M^*) = 0$ and any matching that ties with M^* must not use any inter-gadget edge. Therefore, the number of matchings that tie with M^* across the entire graph G is simply the product of matchings that tie with it on individual single, double, triple, and auxiliary gadgets. Hence,

$$\begin{aligned} \text{loss}(M^*) + \text{ties}(M^*) &= \text{ties}(M^*) \\ &= 2^{n'} \cdot 3^{n''} \cdot 10^{m'} \cdot 30^{t_2} \cdot 90^{t_3} \cdot 90^{a}, \quad (6) \end{aligned}$$

where n', n'', m', t_2, t_3 and a are defined as in (5). For any double or triple gadget, by Observation 1 and Lemma 3.1, the entire contribution to $\text{ties}(M^*)$ due to these two gadgets is 30 (or 90 in case of three gadgets) and to $\text{loss}(M^*)$ is 0. Additionally, for an auxiliary gadget, the value of $\text{ties}(M^*)$ is equal to $3 \times 3 \times 10 = 90$.

Comparing (5) and (6) along with the fact that $\text{loss}(M) \geq 0 = \text{loss}(M^*)$, we get that $\text{score}(M^*) > \text{score}(M)$. Indeed,

$$\text{loss}(M^*) + \text{ties}(M^*)/2 \;=\; 2^{n'-1} \cdot 3^{n''} \cdot 10^{m'} \cdot 30^{t_2} \cdot 90^{t_3} \cdot 90^{a}, \text{ and}$$

$$\begin{aligned} \text{loss}(M) + \text{ties}(M)/2 \;&\geq\; (\text{loss}(M) + \text{ties}(M))/2 \\ &\geq\; 2^{n'-1} \cdot 3^{n''} \cdot 10^{m'} \cdot 100^{t_2} \cdot 100^{t_3} \cdot 100^{a}. \end{aligned}$$

This contradicts the fact that M is a Copeland winner. Thus, a Copeland winner must not use any inter-gadget edge. □

REMARK. *By reducing from a restricted version of VERTEX COVER on 3-regular graphs, which is also known to be* NP*-hard [18], the intractability stated in Theorem 1.6 can be shown to hold even when there are only a* constant *number of neighbors per vertex.*

4 CONCLUDING REMARKS

We adopted a voting-theoretic perspective on the matching-under-preferences problem, and examined some existential and computational questions in the context of relaxing popularity. Though we know that a semi-popular matching always exists and we showed an FPRAS to find an *almost* semi-popular matching, we do not know the computational complexity of finding an *exact* semi-popular matching. The main open question here is settle this complexity. We also showed that it is NP-hard to find a Copeland winner. Is there an FPRAS for an *approximate* Copeland winner?

Going forward, it will be very interesting to consider other voting rules that might facilitate tractability results while providing natural relaxations to well-studied solution concepts such as stability and popularity.

ACKNOWLEDGEMENTS

We are grateful to the anonymous reviewers for their helpful comments. TK acknowledges support from project no. RTI4001 of the

[6] A matching M is Pareto optimal if there is no matching N such that at least one vertex is better off in N than in M and no vertex is worse off in N.

Department of Atomic Energy, Government of India. RV acknowledges support from SERB grant no. CRG/2022/002621 and DST INSPIRE grant no. DST/INSPIRE/04/2020/000107.

REFERENCES

[1] A. Abdulkadiroğlu, P. Pathak, and A. E. Roth. The New York City High School Match. *American Economic Review*, 95(2): 364–367, 2005.

[2] A. Abdulkadiroğlu, P. Pathak, A. E. Roth, and T. Sönmez. The Boston Public School Match. *American Economic Review*, 95(2): 368–371, 2005.

[3] D. J. Abraham, P. Biro, and D. F. Manlove. Almost stable matchings in the Roommates problem. *Proceedings of the 3rd Workshop on Approximation and Online Algorithms (WAOA)*, 1–14, 2006.

[4] P. Biro, R. W. Irving, and D. F. Manlove. Popular Matchings in the Marriage and Roommates Problems. *Proceedings of the 7th International Conference on Algorithms and Complexity (CIAC)*, 97–108, 2010.

[5] F. Brandt and M. Bullinger. Finding and Recognizing Popular Coalition Structures. *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), 195–203, 2020.*

[6] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia. Handbook of Computational Social Choice. *Cambridge University Press*, 2016.

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

[8] A. H. Copeland. A "Reasonable" Social Welfare Function. Mimeo, University of Michigan, 1951.

[9] Copeland's Method. https://en.wikipedia.org/wiki/Copeland

[10] Á. Cseh. Popular Matchings. Chapter 6 in *Trends in Computational Social Choice*, 105(3), 2017.

[11] 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.

[12] P. Dasgupta and E. Maskin. The Fairest Vote of All. *Scientific American*, 290(3): 64–69, 2004.

[13] Y. Faenza and T. Kavitha. Quasi-Popular Matchings, Optimality, and Extended Formulations. *Mathematics of Operations Research*, 47(1): 427-457, 2022.

[14] Y. Faenza, T. Kavitha, V. Powers, and X. Zhang. Popular Matchings and Limits to Tractability. *Proceedings of the 30th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2790–2809, 2019.

[15] P. Faliszewski, E. Hemaspaandra, H. Schnoor. Copeland Voting: Ties Matter. *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2: 983–990, 2008.

[16] D. Gale and L. S. Shapley. College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, 69(1): 9–15, 1962.

[17] P. Gärdenfors. Match Making: Assignments Based on Bilateral Preferences. *Behavioral Science*, 20(3): 166–173, 1975.

[18] M. R. Garey, D. S. Johnson, L. Stockmeyer. Some Simplified NP-Complete Graph Problems. *Theoretical Computer Science*, 1(3): 237–267, 1976.

[19] S. Gupta, P. Misra, S. Saurabh, and M. Zehavi. Popular Matching in Roommates Setting is NP-Hard. *ACM Transactions on Computation Theory*, 13(2): Mar. 2021.

[20] D. Gusfield and R. W. Irving. The Stable Marriage Problem: Structure and Algorithms. *MIT Press*, 1989.

[21] G. Hägele and F. Pukelsheim. Llull's Writings on Electoral Systems. *Studia Lulliana*, 41(97): 3–38, 2001.

[22] W. Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *The Collected Works of Wassily Hoeffding*, 409–426, 1994.

[23] R. W. Irving. An Efficient Algorithm for the Stable Roommates Problem. *Journal of Algorithms*, 6: 577–595, 1985.

[24] M. Jerrum and A. Sinclair. Approximating the Permanent. *SIAM Journal on Computing*, 18(6): 1149–1178, 1989.

[25] T. Kavitha. Min-Cost Popular Matchings. *Proceedings of the 40th Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, 25:1–25:17, 2020.

[26] T. Kavitha and R. Vaish. Matchings and Copeland's Method. https://arxiv.org/abs/2105.13729

[27] D. Manlove. Algorithmics of Matching under Preferences. *World Scientific Publishing Company*, 2013.

[28] R. M. McCutchen. The Least-Unpopularity-Factor and Least-Unpopularity-Margin Criteria for Matching Problems with One-Sided Preferences. *Proceedings of the Latin American Symposium on Theoretical Informatics (LATIN)*, 593–604, 2008.

[29] R. Motwani and P. Raghavan. Randomized Algorithms. *Cambridge University Press*, 1995.

[30] E. Pacuit. Voting Methods. *The Stanford Encyclopedia of Philosophy.* https://plato.stanford.edu/archives/fall2019/entries/voting-methods, 2019

[31] N. Perach, J. Polak, and U. Rothblum. A Stable Matching Model with an Entrance Criterion Applied to the Assignment of Students to Dormitories at the Technion. *International Journal of Game Theory*, 36(3), 519–535, 2008.

[32] A. E. Roth. The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory. *Journal of Political Economy*, 92(6): 991–1016, 1984.

[33] A. E. Roth and E. Peranson. The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design. *American Economic Review*, 89(4): 748–780, 1999.

[34] A. E. Roth and M. Sotomayor. Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis (Econometric Society Monographs). *Cambridge University Press*, 1992.

[35] R. Vaish, N. Misra, S. Agarwal, and A. Blum. On the Computational Hardness of Manipulating Pairwise Voting Rules. *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS)*, 358–367, 2016.