

Strategic Voting in the Context of Stable-Matching of Teams

Leora Schmerler and Noam Hazon and Sarit Kraus

Abstract

In the celebrated stable-matching problem, there are two sets of agents M and W , and the members of M only have preferences over the members of W and vice versa. It is usually assumed that each member of M and W is a single entity. However, there are many cases in which each member of M or W represents a team that consists of several individuals with common interests. For example, students may need to be matched to professors for their final projects, but each project is carried out by a team of students. Thus, the students first form teams, and the matching is between teams of students and professors.

When a team is considered as an agent from M or W , it needs to have a preference order that represents it. A voting rule is a natural mechanism for aggregating the preferences of the team members into a single preference order. In this paper, we investigate the problem of strategic voting in the context of stable-matching of teams. Specifically, we assume that members of each team use the Borda rule for generating the preference order of the team. Then, the Gale-Shapley algorithm is used for finding a stable-matching, where the set M is the proposing side. We show that the single-voter manipulation problem can be solved in polynomial time, both when the team is from M and when it is from W . We show that the coalitional manipulation problem is computationally hard, but it can be solved approximately both when the team is from M and when it is from W .

1 Introduction

Matching is the process in which agents from different sets are matched with each other. The theory of matching originated with the seminal work of Gale and Shapley [9], and since then intensive research has been conducted in this field. Notably, the theory of matching has also been successfully applied to many real-world applications including college admissions and school matching [1], matching residents to hospitals [17], and kidney exchange [18]. A very common matching problem, which is also the problem that was studied by Gale and Shapley in their original paper, is the *stable-matching* problem. In this problem there are two equally sized disjoint sets of agents, M and W , and the members of M have preferences over only the members of W , and vice versa. The goal is to find a stable bijection (i.e., matching) from the agents of M to the agents of W , where the stability requirement is that no pair of agents prefers a match with each other over their matched partners. Many works have analyzed this setting, and they assume that each member of the sets M and W represents a single agent. However, there are many cases in which each member of M or W represents more than one individual [14].

For example, suppose that teams of students need to be matched with professors who will serve as their advisors in their final projects. It is common that students form their teams based on friendship connections and common interests and then approach the professors. Therefore, each team is considered to be a single agent for the matching process: the professors may have different preferences regarding which team they would like to mentor, and the teams may have preferences regarding which professor they would like as their mentor. Clearly, even though the team is considered to be a single agent for the matching process, it is still composed of several students, and they may have different opinions regarding the

appropriate mentor for their team. Thus, every team needs a mechanism that aggregates the students' opinions and outputs a single preference order that represents the team for the matching process, and a voting rule is a natural candidate.

Indeed, voters might benefit from reporting rankings different from their true ones, and this problem of manipulation also exists in the context of matching. For example, suppose that there are 4 possible professors, denoted by p_1, p_2, p_3 and p_4 and 4 teams. Now, suppose that one of the students, denoted r , who is a member of one of the teams, prefers p_1 over p_2 , p_2 over p_3 , and p_3 over p_4 . It is possible that r will gain an (unauthorized) access to the preferences of the professors and to the preferences of the other teams. Since the matching algorithm is usually publicly known, r might be able to reason that p_3 is matched with his team, but if r votes strategically and misreports his preferences then p_2 will be matched with his team.

In this paper, we investigate the problem of strategic voting in the context of stable-matching of teams. We assume that the members of each team use the Borda rule as a *social welfare function (SWF)*, which outputs a complete preference order. This preference order represents the team for the matching process. The agents then use the Gale-Shapley (GS) algorithm for finding a stable-matching. In the GS algorithm, one set of agents makes proposals to the other set of agents, and it is assumed that M is the proposing side and W is the proposed-to side. The proposing side and proposed-to side are commonly referred to as men and women, respectively. Note that the GS algorithm treats the men and women differently. Therefore, every manipulation problem in the context of stable-matching has two variants: one in which the teams are from the men's side, and another one in which the teams are from the women's side. Moreover, we analyze both manipulation by a single voter and coalitional manipulation. In a single voter manipulation, the goal is to find a preference order for a single manipulator such that his team will be matched by the GS algorithm with a specific preferred agent. In the coalitional manipulation setting, there are several voters who collude and coordinate their votes so that an agreed upon agent will be matched with their team.

We begin by studying manipulation from the men's side, and show that the single voter manipulation problem can be solved in polynomial time. We then analyze the coalitional manipulation problem, and show that the problem is computationally hard. However, we provide a polynomial-time algorithm with the following guarantee: given a manipulable instance with $|R|$ manipulators, the algorithm finds a successful manipulation with at most one additional manipulator. We then study manipulation from the women's side. Manipulation here is more involved, and we propose different algorithms, but with almost the same computational complexity as in manipulation from the men's side. That is, the single voter manipulation problem can be solved in polynomial time, and the coalitional manipulation problem is computationally hard. Indeed, we provide a polynomial-time algorithm with the following guarantee: given a manipulable instance with $|R|$ manipulators, the algorithm finds a successful manipulation with at most two additional manipulators.

The contribution of this work is twofold. First, it provides an analysis of a voting manipulation in the context of stable-matching of teams, a problem that has not been investigated to date. Second, our work concerns the manipulation of Borda as an SWF, which has scarcely been investigated.

2 Related Work

The computational analysis of voting manipulation has been vastly studied in different settings. We refer the reader to the survey provided by Faliszewski and Procaccia [8], and the more recent survey by Conitzer and Walsh [4]. However, most of the works on voting

manipulation analyze the problem with no actual context, and where a voting rule is used to output one winning candidate or a set of tied winning candidates (i.e., a social choice function). In this work, we investigate manipulation of Borda as a SWF, which outputs a complete preference order of the candidates, and analyze it within the context of stable-matching.

Indeed, there are a few papers that investigate the manipulation of SWFs. The first work that directly deals with the manipulation of SWF was by Bossert and Storcken [3], who assumed that a voter prefers one order over another if the former is closer to her own preferences than the latter according to the Kemeny distance. Bossert and Sprumont [2] assumed that a voter prefers one order over another if the former is strictly between the latter and the voter’s own preferences. Built on this definition, their work studies three classes of SWF that are not prone to manipulation (i.e., strategy-proof). Dogan and Lainé [6] characterized the conditions to be imposed on SWFs so that if we extend the preferences of the voters to preferences over orders in specific ways, the SWFs will not be prone to manipulation. Our work also investigates the manipulation of SWF, but we analyze the SWF in the specific context of stable-matching. Therefore, unlike all of the above works, the preferences of the manipulators are well-defined and no additional assumptions are needed. The work that is closest to ours is that of Schmerler and Hazon [19]. They assume that a positional scoring rule is used as a SWF, and study the manipulation of the SWF in the context of negotiation.

The strategic aspects of the GS algorithm have previously been studied in the literature. It was first shown that reporting the true preferences is a weakly dominant strategy for men, but women may have an incentive to misreport their preferences [7, 16, 10]. Teo et al. [22] provided a polynomial-time algorithm for computing the optimal manipulation by a woman. Shen et al. [21] generalized this result to manipulation by a coalition of women. For the proposing side, Dubins and Freedman [7] investigated the strategic actions of a coalition of men, and proved that there is no manipulation that is a strict improvement for every member of the coalition. Huang [13] studied manipulation that is a weak improvement for every member of a coalition of men. Hosseini et al. [11] introduced a new type of strategic action: manipulation through an accomplice. In this manipulation, a man misreports his preferences in behalf of a woman, and Hosseini et al. provided a polynomial time algorithm for computing an optimal accomplice manipulation, and they further generalized this model in [12]. All of these works consider the manipulation of the GS algorithm, while we study the manipulation of Borda as a SWF. Indeed, the output of the SWF is used (as part of the input) for the GS algorithm. As an alternative to the GS algorithm, Pini et al. [15] show how voting rules which are NP-hard to manipulate can be used to build stable-matching procedures, which are themselves NP-hard to manipulate.

3 Preliminaries

We assume that there are two equally sized disjoint sets of agents, M and W . Let $k = |M| = |W|$. The members of M have preferences over only the members of W , and vice versa. The preference of each $m \in M$, denoted by \succ_m , is a strict total order over the agents in W . The preference profile \succ_M is a vector $(\succ_{m_1}, \succ_{m_2}, \dots, \succ_{m_k})$. The preference order \succ_w and the preference profile \succ_W are defined analogously. We will refer to the agents of M as men and to the agents of W as women.

A matching is a mapping $\mu : M \cup W \rightarrow M \cup W$, such that $\mu(m) \in W$ for all $m \in M$, $\mu(w) \in M$ for all $w \in W$, and $\mu(m) = w$ if and only if $\mu(w) = m$. A stable-matching is a matching in which there is no blocking pair. That is, there is no man m and woman w such that $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$. The GS algorithm finds a stable-matching, and

it works as follows. There are multiple rounds, and each round is composed of a proposal phase followed by a rejection phase. In a proposal phase, each unmatched man proposes to his favorite woman from among those who have not yet rejected him (regardless of whether the woman is already matched). In the rejection phase, each woman tentatively accepts her favorite proposal and rejects all of the other proposals. The algorithm terminates when no further proposals can be made. Let $o(w)$ be the set of men that proposed to w in one of the rounds of the GS algorithm.

In our setting, (at least) one of the agents of M (W) is a team that runs an election for determining its preferences. That is, there is a man \hat{m} (woman \hat{w}), which is associated with a set of voters, V . The preference of each $v \in V$, denoted by ℓ_v , is a strict total order over W (M). The preference profile \mathcal{L} is a vector of the preference orders for each $v \in V$. The voters use the *Borda* rule as a SWF, denoted by \mathcal{F} , which is a mapping of the set of all preference profiles to a single strict preference order. Specifically, in the Borda rule, each voter v awards the candidate that is placed in the top-most position in ℓ_v a score of $k - 1$, the candidate in the second-highest position in ℓ_v a score of $k - 2$, etc. Then, for the output of \mathcal{F} , the candidate with the highest aggregated score is placed in the top-most position, the candidate with the second-highest score is placed in the second-highest position, etc. Since ties are possible, we assume that a lexicographical tie-breaking rule is used. Note that the output of \mathcal{F} is the preference order of \hat{m} (\hat{w}). That is, $\succ_{\hat{m}} = \mathcal{F}(\mathcal{L})$, and $\succ_{\hat{w}}$ is defined analogously.

Recall that the GS algorithm finds a stable matching, given \succ_M and \succ_W . Given a man $m \in M$, let \succ_{M-m} be the preference profile of all of the men besides m , and \succ_{W-w} is defined analogously. We consider a setting in which the input for the GS algorithm is $\succ_{M-\hat{m}}, \succ_{\hat{m}}$, and \succ_W , and thus $\mu(\hat{m})$ is the spouse that is the match of \hat{m} according to the output of the GS algorithm. We also consider a setting in which the input for the GS algorithm is $\succ_{W-\hat{w}}, \succ_{\hat{w}}$ and \succ_M , and thus $\mu(\hat{w})$ is the spouse that is the match of \hat{w} according to the output of the GS algorithm. In some circumstances, we would like to examine the output of the GS algorithm for different possible preference orders that represent a man $m \in M$. We denote by $\mu_x(m, \succ)$ the spouse that is the match of m when the input for the GS algorithm is \succ_{M-m}, \succ (instead of \succ_m), and \succ_W . We define $\mu_x(w, \succ)$ and $o_x(w, \succ)$ similarly.

We study the setting in which there exists a manipulator r among the voters associated with a man \hat{m} (woman \hat{w}), and his (her) preference order is ℓ_r . The preference order that represents \hat{m} (\hat{w}) is thus $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$. We also study the setting in which there is a set of R manipulators, their preference profile is $\mathcal{L}_R = \{\ell_1, \ell_2, \dots, \ell_{|R|}\}$, and the preference order that represents \hat{m} (\hat{w}) is thus $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. For clarity purposes we slightly abuse notation, and write $\mu(\hat{m}, \ell_r)$ for denoting the spouse that is the match of \hat{m} according to the output of the GS algorithm, given that its input is $\succ_{M-\hat{m}}, \mathcal{F}(\mathcal{L} \cup \{\ell_r\})$, and \succ_W . We define $\mu(\hat{w}, \ell_r)$, $o(\hat{w}, \ell_r)$, $\mu(\hat{m}, \mathcal{L}_R)$, $\mu(\hat{w}, \mathcal{L}_R)$ and $o(\hat{w}, \mathcal{L}_R)$ similarly.

Let $s(c, \ell_v)$ be the score of candidate c from ℓ_v . Similarly, let $s(c, \mathcal{L})$ be the total score of candidate c from \mathcal{L} , i.e., $s(c, \mathcal{L}) = \sum_{v \in V} s(c, \ell_v)$. Similarly, $s(c, \mathcal{L}, \ell_r) = \sum_{v \in V} s(c, \ell_v) + s(c, \ell_r)$, and $s(c, \mathcal{L}, \mathcal{L}_R) = \sum_{v \in V} s(c, \ell_v) + \sum_{r \in R} s(c, \ell_r)$. Since we use a lexicographical tie-breaking rule, we write that $(c, \ell) > (c', \ell')$ if $s(c, \ell) > s(c', \ell')$ or $s(c, \ell) = s(c', \ell')$ but c is preferred over c' according to the lexicographical tie-breaking rule. We define $(c, \mathcal{L}, \ell) > (c', \mathcal{L}, \ell')$ and $(c, \mathcal{L}, \mathcal{L}_R) > (c', \mathcal{L}, \mathcal{L}'_R)$ similarly. Note that due to space constraints, almost all of the proofs are deferred to the full version of the paper [20].

4 Men's Side

We begin by considering the variant in which a specific voter, or a coalition of voters, are associated with an agent \hat{m} , and they would like to manipulate the election so that a

preferred spouse w^* will be the match of \hat{m} .

4.1 Single Manipulator

With a single manipulator, the Manipulation in the context of Matching from the Men's side (MnM-m) is defined as follows:

Definition 1 (MnM-m). *We are given a man \hat{m} , the preference profile \mathcal{L} of the honest voters that associate with \hat{m} , the preference profile $\succ_{M-\hat{m}}$, the preference profile \succ_W , a specific manipulator r , and a preferred woman $w^* \in W$. We are asked whether a preference order ℓ_r exists such that $\mu(\hat{m}, \ell_r) = w^*$.*

We show that MnM-m can be decided in polynomial time by Algorithm 1, which works as follows. The algorithm begins by verifying that a preference order exists for \hat{m} , which makes w^* the match of \hat{m} . It thus iteratively builds a temporary preference order for \hat{m} , \succ_x in lines 4-7. Moreover, during the iterations in lines 4-7 the algorithm identifies a set B , which is the set of women that might prevent w^* from being \hat{m} 's match. Specifically, \succ_x is initialized as the original preference order of \hat{m} , $\succ_{\hat{m}}$. In each iteration, the algorithm finds the woman b , which is the match of \hat{m} given that \succ_x is the preference order of \hat{m} . If b is placed higher than w^* in \succ_x , then b is added to the set B , it is placed in \succ_x immediately below w^* , and the algorithm proceeds to the next iteration (using the updated \succ_x).

Now, if $b = \mu_x(\hat{m}, \succ_x)$ is positioned lower than w^* in \succ_x , then no preference order exists that makes w^* the match of \hat{m} , and the algorithm returns false. If $b = w^*$, then the algorithm proceeds to build the preference order for the manipulator, ℓ_r . Clearly, w^* is placed in the top-most position in ℓ_r . Then, the algorithm places all the women that are not in B in the highest available positions. Finally, the algorithm places all the women from B in the lowest positions in ℓ_r , and they are placed in a reverse order with regard to their order in $\mathcal{F}(\mathcal{L})$.

ALGORITHM 1: Manipulation by a single voter from the men's side

```

1  $B \leftarrow \emptyset$ 
2 set  $\succ_x$  to be  $\succ_{\hat{m}}$ 
3  $b \leftarrow \mu_x(\hat{m}, \succ_x)$ 
4 while  $b \succ_x w^*$  do
5   | add  $b$  to  $B$ 
6   | move  $b$  in  $\succ_x$  immediately below  $w^*$ 
7   |  $b \leftarrow \mu_x(\hat{m}, \succ_x)$ 
8 if  $b \neq w^*$  then
9   | return false
  // phase 1:
10  $\ell_r \leftarrow$  empty preference order
11 place  $w^*$  in the highest position in  $\ell_r$ 
12 for each  $w \in W \setminus (B \cup \{w^*\})$  do
13   | place  $w$  in the next highest available position in  $\ell_r$ 
  // phase 2:
14 while  $B \neq \emptyset$  do
15   |  $b \leftarrow$  the least preferred woman from  $B$  according to  $\mathcal{F}(\mathcal{L})$ 
16   | place  $b$  in the highest available position in  $\ell_r$ 
17   | remove  $b$  from  $B$ 
18 if  $\mu(\hat{m}, \ell_r) = w^*$  then
19   | return  $\ell_r$ 
20 return false

```

For proving the correctness of Algorithm 1 we use the following known results:

Theorem 1 (due to [16]). *In the Gale-Shapley matching procedure which always yields the optimal stable outcome for the set of the men agents, M , truthful revelation is a dominant strategy for all the agents in that set.*

Lemma 1 (due to [13]). *For man m , his preference list is composed of $(P_L(m), \mu(m), P_R(m))$, where $P_L(m)$ and $P_R(m)$ are respectively those women ranking higher and lower than $\mu(m)$. Let $A \subseteq W$ and let $\pi_r(A)$ be a random permutation from all $|A|!$ sets. For a subset of men $S \subseteq M$, if every member $m \in S$ submits a falsified list of the form $(\pi_r(P_L(m)), \mu(m), \pi_r(P_R(m)))$, then $\mu(m)$ stays m 's match.*

An immediate corollary of Theorem 1 is the following.

Corollary 1. *Given a man m with his preference order \succ_m , let $w_m = \mu(m)$. Let \succ'_m be a preference order for m such that if $w_m \succ_m w$ then $w_m \succ'_m w$. Then, w_m is also the match of m with the preference order \succ'_m , i.e., $\mu_x(m, \succ'_m) = w_m$.*

We begin the analysis of Algorithm 1 by showing that it is possible to verify (in polynomial time) whether a preference order exists for \hat{m} , which makes w^* the match of \hat{m} . We do so by showing that it is sufficient to check whether $w^* = \mu_x(\hat{m}, \succ_x)$, where \succ_x is the preference order that is built by Algorithm 1 in lines 4-7.

Lemma 2. *A preference order \succ_t for \hat{m} exists such that $w^* = \mu_x(\hat{m}, \succ_t)$ if and only if $w^* = \mu_x(\hat{m}, \succ_x)$.*

That is, if Algorithm 1 returns false in line 9 then there is no preference order for \hat{m} that makes w^* the match of \hat{m} (and thus no manipulation is possible for r).

We now show that the set B , which is identified by the algorithm in lines 4-7, is a set of woman that might prevent w^* from being \hat{m} 's match.

Lemma 3. *Given a preference order \succ_t for \hat{m} , if there exists $b \in B$ such that $b \succ_t w^*$ then $\mu_x(\hat{m}, \succ_t) \neq w^*$.*

That is, Algorithm 1 should place the women from B in the lowest position in ℓ_r and w^* in the highest position in ℓ_r , so that w^* will be preferred over every woman $b \in B$ in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$.

Theorem 2. *Algorithm 1 correctly decides the MnM-m problem in polynomial time.*

4.2 Coalitional Manipulation

We now study manipulation by a coalition of voters. The coalitional manipulation in the context of matching from the men's side is defined as follows:

Definition 2 (coalitional MnM-m). *We are given a man \hat{m} , the preference profile \mathcal{L} of the honest voters that associate with \hat{m} , the preference profile $\succ_{M-\hat{m}}$, the preference profile \succ_W , a coalition of manipulators R , and a preferred woman $w^* \in W$. We are asked whether a preference profile \mathcal{L}_R exists such that $\mu(\hat{m}, \mathcal{L}_R) = w^*$.*

We show that the coalitional MnM-m problem is computationally hard, even with two manipulators. The reduction is from the Permutation Sum problem (as defined by Davies et al. [5]) that is *NP*-complete [24].

Definition 3 (Permutation Sum). *Given q integers $X_1 \leq \dots \leq X_q$ where $\sum_{i=1}^q X_i = q(q+1)$, do two permutations σ and π of 1 to q exist such that $\sigma(i) + \pi(i) = X_i$ for all $1 \leq i \leq q$?*

Theorem 3. *Coalitional MnM-m is NP-Complete.*

Even though coalitional MnM-m is NP-complete, it might still be possible to develop an efficient heuristic algorithm that finds a successful coalitional manipulation. We use Algorithm 2, which is a generalization of Algorithm 1, that works as follows. Similar to

ALGORITHM 2: Manipulation by a coalition of voters from the men's side

```

1  $B \leftarrow \emptyset$ 
2 set  $\succ_x$  to be  $\succ_{\hat{m}}$ 
3  $b \leftarrow \mu_x(\hat{m}, \succ_x)$ 
4 while  $b \succ_x w^*$  do
5   | add  $b$  to  $B$ 
6   | place  $b$  in  $\succ_x$  immediately below  $w^*$ 
7   |  $b \leftarrow \mu_x(\hat{m}, \succ_x)$ 
8 if  $b \neq w^*$  then
9   | return false
10 for each  $r \in R$  do
11   |  $\ell_r \leftarrow$  empty preference order
12   | place  $w^*$  in the highest position in  $\ell_r$ 
13   | for each  $w \in W \setminus (B \cup \{w^*\})$  do
14     | place  $w$  in the next highest available position in  $\ell_r$ 
15   |  $B' \leftarrow B$ 
16   | while  $B' \neq \emptyset$  do
17     |  $b \leftarrow$  the least preferred woman from  $B'$  according to  $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$ 
18     | place  $b$  in the highest available position in  $\ell_r$ 
19     | remove  $b$  from  $B'$ 
20   | add  $\ell_r$  to  $\mathcal{L}_R$ 
21 if  $\mu(\hat{m}) = w^*$  then
22   | return  $\mathcal{L}_R$ 
23 return false

```

Algorithm 1, Algorithm 2 identifies a set B , which is the set of women that might prevent w^* from being \hat{m} 's match. In addition, it verifies that a preference order for \hat{m} exists, which makes w^* the match of \hat{m} . Then, Algorithm 2 proceeds to build the preference order of every manipulator $r \in R$ similarly to how Algorithm 1 builds the preference order for the single manipulator. Indeed, Algorithm 2 builds the preference order of each manipulator r in turn, and the order in which the women in B are placed depends on their order according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. That is, the order in which the woman in B are placed in each ℓ_r is not the same for each r , since \mathcal{L}_R is updated in each iteration. We refer to each of the iterations in Lines 10-20 as a *stage* of the algorithm. We now show that Algorithm 2 is an efficient heuristic that also has a theoretical guarantee. Specifically, the algorithm is guaranteed to find a coalitional manipulation in many instances, and we characterize the instances in which it may fail. Formally,

Theorem 4. *Given an instance of coalitional MnM-m,*

1. *If there is no preference profile making w^* the match of \hat{m} , then Algorithm 2 will return false.*
2. *If a preference profile making w^* the match of \hat{m} exists, then for the same instance with one additional manipulator, Algorithm 2 will return a preference profile that makes w^* the match of \hat{m} .*

That is, Algorithm 2 will succeed in any given instance such that the same instance but with one less manipulator is manipulable. Thus, it can be viewed as a 1-additive approximation algorithm (this approximate sense was introduced by Zuckerman et al. [25] when analyzing Borda as a social choice function (SCF)).

In order to prove Theorem 4 we use the following definitions. Let $D_0 = \{d_0\}$, where $(d_0, \mathcal{L}) > (w, \mathcal{L})$, and $d_0, w \in B$. For each $s = 1, 2, \dots$, let $D_s \subseteq B$ be $D_s = D_{s-1} \cup \{d \in B : d \text{ was ranked above some } d' \in D_{s-1} \text{ according to } \mathcal{F}(\mathcal{L}, \mathcal{L}_R) \text{ in some stage } l, 1 \leq l \leq |R|\}$. Now, let $D = \bigcup_{0 \leq s} D_s$. Note that $\forall s D_s \neq D_{s-1}$, and s does not necessarily equal $|R|$. Let $s_\ell(w)$ be the score of woman w in $\mathcal{F}(\mathcal{L}, \mathcal{L}_R)$ after stage ℓ .

The proof of Theorem 4 relies on Lemmata 4-8, and its general intuition is as follows. Consider the women in D : we show that if there exists a manipulation, then Algorithm 2 is able to determine the votes in \mathcal{L}_R such that the average score of the women in D is lower than the score of woman w^* . Moreover, a successful manipulation requires that w^* will be ranked higher than any women in D , and thus the algorithm may use one additional manipulator.

We begin with a basic lemma that clarifies where Algorithm 2 places the women of D .

Lemma 4. *The women in D are placed in each stage l , $1 \leq l \leq |R|$ in the $|D|$ lowest positions.*

We now show the relation between the score of w^* and the average score of the women in D , when there are $|R| - 1$ manipulators. In essence, the lemma characterizes the settings in which Algorithm 2 returns false and no manipulation exists.

Lemma 5. *Let $q(D)$ be the average score of women in D after $|R| - 1$ stages. That is, $q(D) = \frac{1}{|D|} \sum_{d \in D} s_{|R|-1}(d)$. If $s_{|R|-1}(w^*) < q(D)$ and there are $|R| - 1$ manipulators, then there is no manipulation that makes w^* the match of \hat{m} , and the algorithm will return false.*

The following lemma shows the relation between the score of w^* and the maximum score of a woman in D , when there are $|R|$ manipulators. In essence, the lemma characterizes the settings in which Algorithm 2 finds a successful manipulation.

Lemma 6. *Assume that a preference order for \hat{m} exists, which makes w^* the match of \hat{m} . If $\max_{d \in D} \{s_{|R|}(d)\} < s_{|R|}(w^*)$ and there are $|R|$ manipulators, then there is a manipulation that makes w^* the match of \hat{m} , and Algorithm 2 will find such a manipulation.*

Finally, we show that the highest score of a woman from D is not much higher than the average score of the women in D . We thus first show that the scores of the women in D are dense, as captured by the following definition.

Definition 4 (due to [25]). *A finite non-empty set of integers A is called 1-dense if, when sorting the set in a non-increasing order $a_1 \geq a_2 \geq \dots \geq a_i$ (such that $\{a_1, \dots, a_i\} = A$), $\forall j, 1 \leq j \leq i - 1$, $a_{j+1} \geq a_j - 1$ holds.*

Lemma 7. *Let D be as before. Then the set $\{s_{|R|-1}(d) : d \in D\}$ is 1-dense.*

Lemma 8. $\max_{d \in D} \{s_{|R|}(d)\} \leq q(D) + |D| - 1$.

Now we can prove the theorem.

Proof of Theorem 4. Clearly, if Algorithm 2 returns a preference profile \mathcal{L}_R , then it is a successful manipulation that will make w^* the match of \hat{m} . Suppose that a preference profile exists that makes w^* the match of \hat{m} with $|R| - 1$ manipulators. By Lemma 8, $\max_{d \in D} \{s_{|R|}(d)\} \leq q(D) + |D| - 1$. By Lemma 5, $q(D) + |D| - 1 \leq s_{|R|-1}(w^*) + |D| - 1$. Since $|D| \leq k - 1$, $s_{|R|-1}(w^*) + |D| - 1 < s_{|R|-1}(w^*) + k - 1 = s_{|R|}(w^*)$. Overall, $\max_{d \in D} \{s_{|R|}(d)\} < s_{|R|}(w^*)$, and by Lemma 6 the algorithm will find a preference profile that will make w^* the match of \hat{m} with $|R|$ manipulators. \square

5 Women's Side

We now consider the second variant, in which a specific voter, or a coalition of voters, are associated with an agent \hat{w} , and they would like to manipulate the election so that a preferred spouse m^* will be the match of \hat{w} . This variant is more involved, since manipulation of the GS algorithm is also possible by a single woman or a coalition of women. Indeed, there are notable differences between manipulation from the women's side and manipulation from the men's side. First, the manipulators from the women's side need to ensure that **two** men are positioned "relatively" high. In addition, the set B , which is the set of agents that are placed in low positions, is defined differently, and it is not built iteratively. Finally, in manipulation from the women's side, it is not always possible to place all the agents from B in the lowest positions.

5.1 Single Manipulator

With a single manipulator, the Manipulation in the context of Matching from the Women's side (MnM-w) is defined as follows:

Definition 5 (MnM-w). *We are given a woman \hat{w} , the preference profile \mathcal{L} of the honest voters that associate with \hat{w} , the preference profile \succ_M , the preference profile $\succ_{W-\hat{w}}$, a specific manipulator r , and a preferred man $m^* \in M$. We are asked whether a preference order ℓ_r exists such that $\mu(\hat{w}, \ell_r) = m^*$.*

ALGORITHM 3: Manipulation by a single voter from the women's side

```

1 for each  $m_{nd} \in M \setminus \{m^*\}$  do
   | // phase 1:
2   |  $\ell_r \leftarrow$  empty preference order
3   | place  $m_{nd}$  in the highest position in  $\ell_r$ 
4   | place  $m^*$  in the second-highest position in  $\ell_r$ 
5   | if  $(m_{nd}, \mathcal{L}, \ell_r) > (m^*, \mathcal{L}, \ell_r)$  then
6   | | place  $m^*$  in the highest position in  $\ell_r$ 
7   | | place  $m_{nd}$  in  $\ell_r$  in the highest position such that  $(m^*, \mathcal{L}, \ell_r) > (m_{nd}, \mathcal{L}, \ell_r)$ , if such
   | | position exists
8   | | if no such position exists then
9   | | | continue to the next iteration
10  | if  $\mu(\hat{w}, \ell_r) \neq m^*$  or  $m_{nd} \notin o(\hat{w}, \ell_r)$  then
11  | | continue to the next iteration
   | // phase 2:
12  | for each  $m \notin o(\hat{w}, \ell_r)$  do
13  | | place  $m$  in the highest available position in  $\ell_r$ 
   | // phase 3:
14  |  $B^{nd} \leftarrow o(\hat{w}, \ell_r) \setminus \{m^*, m_{nd}\}$ 
15  | while  $B^{nd} \neq \emptyset$  do
16  | |  $b \leftarrow$  the least preferred man from  $B^{nd}$  according to  $\mathcal{F}(\mathcal{L})$ 
17  | | place  $b$  in the highest available position in  $\ell_r$ 
18  | | remove  $b$  from  $B^{nd}$ 
19  | if  $\mu(\hat{w}, \ell_r) = m^*$  then
20  | | return  $\ell_r$ 
21 return false

```

Clearly, if $\mu(\hat{w}) = m^*$ then finding a preference order ℓ_r such that $\mu(\hat{w}, \ell_r) = m^*$ is trivial. We thus henceforth assume that $\mu(\hat{w}) \neq m^*$. The MnM-w problem can be decided

in polynomial-time, using Algorithm 3. The algorithm tries to identify a man $m_{nd} \in M$, and to place him and m^* in ℓ_r such that m_{nd} is ranked in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$ as high as possible while m^* is still preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$. In addition, the algorithm ensures (at the end of phase 1) that $\mu(\hat{w}, \ell_r) = m^*$ and $m_{nd} \in o(\hat{w}, \ell_r)$. Note that we compute $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$ even though ℓ_r is not a complete preference order, since we assume that all the men that are not in ℓ_r get a score of 0 from ℓ_r . If phase 1 is successful (i.e., $\mu(\hat{w}, \ell_r) = m^*$ and $m_{nd} \in o(\hat{w}, \ell_r)$), the algorithm proceeds to phase 2, where it fills the preference order ℓ_r by placing all the men that are not in $o(\hat{w}, \ell_r)$ in the highest available positions. Finally, in phase 3, the algorithm places all the men from $o(\hat{w}, \ell_r)$ (except for m^* and m_{nd} that are already placed in ℓ_r) in the lowest positions in ℓ_r , and they are placed in a reverse order with regard to their order in $\mathcal{F}(\mathcal{L})$. If $\mu(\hat{w}, \ell_r) = m^*$ then we are done; otherwise, the algorithm iterates and considers another man.

For proving the correctness of Algorithm 3 we need the following result.

Lemma 9 (Swapping lemma, due to [23]). *Given a woman $w \in W$, let \succ'_w be a preference order that is derived from \succ_w by swapping the positions of an adjacent pair of men (m_i, m_j) and making no other changes. Then,*

1. *if $m_i \notin o(w)$ or $m_j \notin o(w)$, then $\mu_x(w, \succ'_w) = \mu(w)$.*
2. *if both m_i and m_j are not one of the two most preferred proposals among $o(w)$ according to \succ_w , then $\mu_x(w, \succ'_w) = \mu(w)$.*
3. *if m_i is the second preferred proposal among $o(w)$ according to \succ_w and m_j is the third preferred proposal among $o(w)$ according to \succ_w , then $\mu_x(w, \succ'_w) \in \{\mu(w), m_j\}$.*
4. *if $m_i = \mu(w)$ and m_j is the second preferred proposal among $o(w)$ according to \succ_w , then the second preferred proposal among $o(w)$ according to \succ'_w is m_i or m_j .*

If we use the swapping lemma sequentially, we get the following corollary.

Corollary 2. *Given a woman $w \in W$, let \succ'_w be a preference order for w such that $\succ_w \neq \succ'_w$. Let $m^* \in M$ be the most preferred man among $o(w)$ according to \succ_w . That is, $\mu(w) = m^*$. Let $m_{nd} \in M$ be the second most preferred man among $o(w)$ according to \succ_w . If m_{nd} is the most preferred man among $o(w) \setminus \{m^*\}$ according to \succ'_w , and $m^* \succ'_w m_{nd}$, then $o(w) = o_x(w, \succ'_w)$ and thus $\mu_x(w, \succ'_w) = \mu(w) = m^*$.*

Corollary 2 is the basis of our algorithm. Intuitively, the manipulator needs to ensure that m^* is among the set of proposals $o(\hat{w}, \ell_r)$, and that m^* is the most preferred men, according to $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$, among this set. That is, $m^* = \mu(\hat{w}, \ell_r)$. Thus, the algorithm searches for a man, denoted by m_{nd} , that serves as the second-best proposal. If such a man exists, then, according to Corollary 2, the position of every man $m \in o(\hat{w}, \ell_r)$ does not change \hat{w} 's match (which is currently m^*) if m_{nd} is preferred over m in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$. In addition, the position of every man $m \notin o(\hat{w}, \ell_r)$ does not change \hat{w} 's match at all.

Theorem 5. *Algorithm 3 correctly decides the MnM-w problem in polynomial time.*

5.2 Coalitional Manipulation

Finally, We study manipulation by a coalition of voters from the women's side.

Definition 6 (coalitional MnM-w). *We are given a woman \hat{w} , the preference profile \mathcal{L} of the honest voters that associate with \hat{w} , the preference profile \succ_M , the preference profile $\succ_{W-\hat{w}}$, a coalition of manipulators R , and a preferred man $m^* \in M$. We are asked whether a preference profile \mathcal{L}_R exists such that $\mu(\hat{w}, \mathcal{L}_R) = m^*$.*

Similar to the single manipulator setting, if $\mu(\hat{w}) = m^*$ then finding a preference profile \mathcal{L}_R such that $\mu(\hat{w}, \ell_R) = m^*$ is trivial. We thus henceforth assume that $\mu(\hat{w}) \neq m^*$. The coalitional MnM-w problem is computationally hard, even with two manipulators, and we again reduce from the Permutation Sum problem (Definition 3).

Theorem 6. *Coalitional MnM-w is NP-Complete.*

Similar to the coalitional MnM-m, the coalitional MnM-w also has an efficient heuristic algorithm that finds a successful manipulation. We use Algorithm 4, which works as follows. Similar to Algorithm 3, Algorithm 4 needs to identify a man $m_{nd} \in M$, such

ALGORITHM 4: Manipulation by a coalition of voters from the women's side

```

1 for each  $m_{nd} \in M \setminus \{m^*\}$  do
   // phase 1:
2    $gap \leftarrow s(m_{nd}, \mathcal{L}) - s(m^*, \mathcal{L})$ 
3   if  $m_{nd}$  is preferred over  $m^*$  according to the lexicographical tie breaking rule then
4      $gap = gap + 1$ 
5   if  $|R| \cdot (k - 1) < gap$  then
6     continue to the next iteration
7    $\mathcal{L}_R \leftarrow (\ell_1, \dots, \ell_{|R|})$  where each preference order is an empty one
8   if  $|R| \geq gap$  then
9     place  $m^*$  in the highest position and  $m_{nd}$  in the second highest position, in
        $\max(gap + \lceil (|R| - gap)/2 \rceil, 0)$  preference orders of  $\mathcal{L}_R$ 
10    place  $m^*$  in the second highest position and  $m_{nd}$  in the highest position in all of the
       other preference orders of  $\mathcal{L}_R$ 
11   else
12     place  $m^*$  in the highest position in each  $\ell_r \in \mathcal{L}_R$ 
13      $s_{m_{nd}} \leftarrow |R| \cdot (k - 1) - gap$ 
14     place  $m_{nd}$  in  $(s_{m_{nd}} \bmod |R|)$  manipulators such that it gets a score of  $\lceil \frac{s_{m_{nd}}}{|R|} \rceil$  from
       each manipulator
15     place  $m_{nd}$  in the other manipulators such that it gets a score of  $\lfloor \frac{s_{m_{nd}}}{|R|} \rfloor$  from each
       manipulator
16   if  $\mu(\hat{w}, \mathcal{L}_R) \neq m^*$  or  $m_{nd} \notin o(\hat{w}, \mathcal{L}_R)$  then
17     continue to the next iteration
   // phase 2:
18    $B^{nd} \leftarrow o(\hat{w}, \mathcal{L}_R) \setminus \{m^*, m_{nd}\}$ 
19   for each  $r \in R$  do
20     for each  $m \in M \setminus (B^{nd} \cup \{m^*, m_{nd}\})$  do
21       place  $m$  in the next highest available position in  $\ell_r$ 
22        $B' \leftarrow B^{nd}$ 
23       while  $B' \neq \emptyset$  do
24          $b \leftarrow$  the least preferred man from  $B'$  according to  $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$ 
25         place  $b$  at the highest available position in  $\ell_r$ 
26         remove  $b$  from  $B'$ 
27   if  $\mu(\hat{w}, \mathcal{L}_R) = m^*$  then
28     return  $\mathcal{L}_R$ 
29 return false

```

that m_{nd} is ranked in $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$ as high as possible while m^* is still preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. In addition, the algorithm needs to ensure that $\mu(\hat{w}, \mathcal{L}_R) = m^*$ and $m_{nd} \in o(\hat{w}, \mathcal{L}_R)$, which is done at the end of phase 1. Indeed, finding such a man $m_{nd} \in M$, and placing him and m^* in every $\ell_r \in \mathcal{L}_R$ is not trivial. The algorithm considers every $m \in M \setminus \{m^*\}$, and computes the difference between the score of m from \mathcal{L} and the

score of m^* from \mathcal{L} . Clearly, if this gap is too big, m cannot be m_{nd} (line 5). Otherwise, there are two possible cases. If there are many manipulators, specifically, $|R| \geq gap$, then the algorithm places m^* and m in the two highest positions in every ℓ_r (lines 9-10). On the other hand, if $|R| < gap$, then the algorithm places m^* in the highest position in every ℓ_r . The algorithm places m such that he gets a total score of $|R| \cdot (k - 1) - gap$ from the manipulators. Moreover, the algorithm tries to place m in almost the same position in every ℓ_r . If phase 1 is successful, the algorithm proceeds to fill the preference orders of \mathcal{L}_R iteratively in phase 2. The algorithm first defines the set B^{nd} , which consists of all the men from $o(\hat{w}, \mathcal{L}_R)$, except for m^* and m_{nd} . Note that at the beginning of phase 2, in which B^{nd} is defined, only m^* and m are positioned in every $\ell_r \in \mathcal{L}_R$. Then, in every $\ell_r \in \mathcal{L}_R$, the algorithm places all the men that are not in B^{nd} (except for m^* and m_{nd} that are already placed in \mathcal{L}_R) in the highest available positions. The algorithm places all the men from B^{nd} in the lowest positions in ℓ_r , and they are placed in a reverse order with regard to their current order in $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. Note that since \mathcal{L}_R is updated in every iteration, the order in which the men from B^{nd} are placed in each ℓ_r is not the same for each r . If $\mu(\hat{w}, \ell_R) = m^*$ then we are done; otherwise, the algorithm iterates and considers another man. We refer to each of the iterations in Lines 19-26 as a *stage* of the algorithm.

We now show that Algorithm 4 will succeed in any given instance such that the same instance but with two less manipulators is manipulable. That is, unlike the coalitional MnM-m, the coalitional MnM-w admits a 2-additive approximation algorithm. Formally,

Theorem 7. *Given an instance of coalitional MnM-w,*

1. *If there is no preference profile making m^* the match of \hat{w} exists, then Algorithm 4 will return false.*
2. *If a preference profile making m^* the match of \hat{w} exists, then for the same instance with two additional manipulators, Algorithm 4 will return a preference profile that makes m^* the match of \hat{w} .*

6 Conclusion and Future Work

In this paper, we initiate the analysis of strategic voting in the context of stable matching of teams. Specifically, we assume that the Borda rule is used as a SWF, which outputs an order over the agents that is used as an input in the GS algorithm. Note that in the standard model of manipulation of Borda, the goal is that a specific candidate will be the winner. In our setting, the algorithms need also to ensure that specific candidates will not be ranked too high. Similarly, in the standard model of manipulation of the GS algorithm, the goal is simply to achieve a more preferred match. In our setting, the algorithms for manipulation need also to ensure that a less preferred spouse is matched to a specific agent. Therefore, even though the manipulation of the Borda rule and the manipulation of the GS algorithm have already been studied, our analysis of the manipulation of Borda rule in the context of GS stable matching provides a better understanding of both algorithms.

Interestingly, our algorithms for the single manipulator settings are quite powerful. They provide exact solutions for the single manipulator case, and their generalizations provide approximate solutions to the coalitional manipulation settings, both when the manipulators are on the men's side or on the women's side.

For future work, we would like to extend our analysis and study additional voting rules as SWFs. It is also worth studying the destructive manipulation objective. In addition, it will be interesting to examine a new type of manipulation, which is only relevant in our setting, in which there is a coalition of manipulators, but every manipulator is associated with a different agent.

Acknowledgments

This research has been partly supported by the Israel Science Foundation under grant 1958/20, by the EU Project TAILOR under grant 952215, and by the Ministry of Science, Technology & Space (MOST), Israel.

References

- [1] Atila Abdulkadiroğlu, Parag A Pathak, and Alvin E Roth. The new york city high school match. *American Economic Review*, 95(2):364–367, 2005.
- [2] Walter Bossert and Yves Sprumont. Strategy-proof preference aggregation: Possibilities and characterizations. *Games and Economic Behavior*, 85:109–126, 2014.
- [3] Walter Bossert and Ton Storcken. Strategy-proofness of social welfare functions: the use of the kemeny distance between preference orderings. *Social Choice and Welfare*, 9(4):345–360, 1992.
- [4] Vincent Conitzer and Toby Walsh. Barriers to manipulation in voting. In Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors, *Handbook of Computational Social Choice*, pages 127–145. Cambridge University Press, 2016.
- [5] Jessica Davies, George Katsirelos, Nina Narodytska, and Toby Walsh. Complexity of and algorithms for borda manipulation. In *The Twenty-Fifth AAAI Conference on Artificial Intelligence*, pages 657–662, 2011.
- [6] Onur Dogan and Jean Lainé. Strategic manipulation of social welfare functions via strict preference extensions. In *The 13th Meeting of the Society for Social Choice and Welfare*, page 199, 2016.
- [7] Lester E Dubins and David A Freedman. Machiavelli and the gale-shapley algorithm. *The American Mathematical Monthly*, 88(7):485–494, 1981.
- [8] Piotr Faliszewski and Ariel D Procaccia. Ai’s war on manipulation: Are we winning? *AI Magazine*, 31(4):53–64, 2010.
- [9] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.
- [10] David Gale and Marilda Sotomayor. Ms. machiavelli and the stable matching problem. *The American Mathematical Monthly*, 92(4):261–268, 1985.
- [11] Hadi Hosseini, Fatima Umar, and Rohit Vaish. Accomplice manipulation of the deferred acceptance algorithm. In *The Thirtieth International Joint Conference on Artificial Intelligence*, pages 231–237, 2021.
- [12] Hadi Hosseini, Fatima Umar, and Rohit Vaish. Two for one & one for all: Two-sided manipulation in matching markets. *arXiv preprint arXiv:2201.08774*, 2022.
- [13] Chien-Chung Huang. Cheating by men in the gale-shapley stable matching algorithm. In *European Symposium on Algorithms*, pages 418–431. Springer, 2006.
- [14] Nitsan Perach and Shoshana Anily. Stable matching of student-groups to dormitories. *European Journal of Operational Research*, 2022.

- [15] Maria Silvia Pini, Francesca Rossi, K Brent Venable, and Toby Walsh. Manipulation complexity and gender neutrality in stable marriage procedures. *Autonomous Agents and Multi-Agent Systems*, 22(1):183–199, 2011.
- [16] Alvin E Roth. The economics of matching: Stability and incentives. *Mathematics of operations research*, 7(4):617–628, 1982.
- [17] Alvin E Roth. The nrmp as a labor market: understanding the current study of the match. *Journal of the American Medical Association*, 275(13):1054–1056, 1996.
- [18] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Kidney exchange. *The Quarterly journal of economics*, 119(2):457–488, 2004.
- [19] Leora Schmerler and Noam Hazon. Strategic voting in negotiating teams. In *International Conference on Algorithmic Decision Theory*, pages 209–223. Springer, 2021.
- [20] Leora Schmerler, Noam Hazon, and Sarit Kraus. Strategic voting in the context of stable-matching of teams. *arXiv preprint arXiv:2207.04912*, 2022.
- [21] Weiran Shen, Yuan Deng, and Pingzhong Tang. Coalitional permutation manipulations in the gale-shapley algorithm. *Artificial Intelligence*, 301:103577, 2021.
- [22] Chung-Piaw Teo, Jay Sethuraman, and Wee-Peng Tan. Gale-shapley stable marriage problem revisited: Strategic issues and applications. *Management Science*, 47(9):1252–1267, 2001.
- [23] Rohit Vaish and Dinesh Garg. Manipulating gale-shapley algorithm: Preserving stability and remaining inconspicuous. In *The Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 437–443, 2017.
- [24] Wenci Yu, Han Hoogeveen, and Jan Karel Lenstra. Minimizing makespan in a two-machine flow shop with delays and unit-time operations is NP-hard. *Journal of Scheduling*, 7(5):333–348, 2004.
- [25] Michael Zuckerman, Ariel D Procaccia, and Jeffrey S Rosenschein. Algorithms for the coalitional manipulation problem. *Artificial Intelligence*, 173(2):392–412, 2009.

Leora Schmerler
Ariel University
Ariel, Israel
Email: leoras@ariel.ac.il

Noam Hazon
Ariel University
Ariel, Israel
Email: noamh@ariel.ac.il

Sarit Kraus
Bar-Ilan University
Ramat-Gan, Israel
Email: sarit@cs.biu.ac.il

A Table of Notations

M/W	The set of men / women
k	$ M $ and $ W $
\succ_m / \succ_w	The linear preference order of man m / woman w
\succ_M / \succ_W	The set of preference orders of the men / women
$\succ_{M-m} / \succ_{W-w}$	The set of preference orders of the men / women without m / w
$\mu(m) / \mu(w)$	The match m / w received
\hat{m} / \hat{w}	A man / woman that represents a voting group
V	The set of honest voters associated with \hat{w} or \hat{m}
\mathcal{L}	The set of preference order of V
$\mathcal{F}(\mathcal{L})$	The output of the SWF on the preferences set \mathcal{L}
w^* / m^*	The preferred woman / man according to the preference of r
$\mu(\hat{w}) / \mu(\hat{m})$	The match \hat{w} / \hat{m} received according to \hat{w} 's / \hat{w} 's preference order $\mathcal{F}(\mathcal{L})$
$\mu(\hat{w}, \ell_r) / \mu(\hat{m}, \ell_r)$	The match \hat{w} / \hat{m} received according to \hat{w} 's / \hat{m} 's preference order $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$
$o(\hat{w})$	The set of proposals \hat{w} received according to \hat{w} 's preference order $\mathcal{F}(\mathcal{L})$
$o(\hat{w}, \ell_r)$	The set of proposals \hat{w} received according to \hat{w} 's preference order $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$
r	The manipulator
ℓ_r	The linear preference order of r
R	The set of manipulators
$\mathcal{L}_R = \{\ell_1, \dots, \ell_{ R }\}$	The preference profile of R
$\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$	The output of the SWF on the preferences set $\mathcal{L} \cup \ell_r$
$s(c, \ell_v)$	The score of c from the preferences of v according to the voting rule
$s(c, \mathcal{L})$	The score of c from the preferences set \mathcal{L} , i.e., $s(c, \mathcal{L}) = \sum_{v \in V} s(c, v)$
$s(c, \mathcal{L}, \ell_r)$	The score of c from the preferences set $\mathcal{L} \cup \{\ell_r\}$
$(c, \ell) > (c', \ell')$	$s(c, \ell) > s(c', \ell')$ or $s(c, \ell) = s(c', \ell')$ but c is preferred over c' according to a lexicographical tie-breaking rule
$(c, \mathcal{L}, \ell) > (c', \mathcal{L}, \ell')$	$s(c, \mathcal{L}, \ell) > s(c', \mathcal{L}, \ell')$ or $s(c, \mathcal{L}, \ell) = s(c', \mathcal{L}, \ell')$ but c is preferred over c' according to a lexicographical tie-breaking rule
$c >_\ell c'$	$(c, \ell) \succ (c', \ell)$

Table 1: Table of Notations.

B Omitted Proofs

B.1 Proof of Corollary 1

Proof. Assume by contradiction that $w'_m = \mu_x(m, \succ'_m) \neq w_m$. There are two possible cases:

1. $w'_m \succ_m w_m$. That is, m will have an incentive to strategically report the preference order \succ'_m instead of his sincere preference order \succ_m , in contradiction to Theorem 1.
2. $w_m \succ_m w'_m$. By the definition of \succ'_m , it is not possible that $w'_m \succ'_m w_m$. Consider the setting in which \succ_W and \succ_{M-m} are the same, but the sincere preference order of m is \succ'_m . However, $w_m \succ'_m w'_m$, and thus, in this setting, m will have an incentive to strategically report the preference order \succ_m instead of his sincere preference order \succ'_m , in contradiction to Theorem 1.

□

B.2 Proof of Lemma 2

Proof. One direction is trivial, so we prove the other direction. Assume that a preference order \succ_t for \hat{m} exists such that $w^* = \mu_x(\hat{m}, \succ_t)$. We show that $w^* = \mu_x(\hat{m}, \succ_x)$. Assume to contradiction that $\mu_x(\hat{m}, \succ_x) = w$ and $w \neq w^*$. It is not possible that $w \succ_x w^*$, since the condition in line 4 would have been true (where $b = w$), and then the algorithm would have placed w below w^* in \succ_x (in line 6). Now, consider the setting in which \succ_W and $\succ_{M-\hat{m}}$ are the same, but the sincere preference order of \hat{m} is \succ_x . In this setting, if \hat{m} is truthful, then his match is w . However, if \hat{m} strategically reports the preference order \succ_t , then his match will be w^* and $w^* \succ_x w$, in contradiction to Theorem 1. □

B.3 Proof of Lemma 3

Proof. Let $b \in B$ such that $b \succ_t w^*$. Since $b \in B$, then there is an iteration of Algorithm 1 in which b is moved immediately below w^* in \succ_x (Line 6). Let $\succ_x^{(1)}$ be the preference order \succ_x in this iteration, before the execution of Line 6. By definition, $\mu_x(\hat{m}, \succ_x^{(1)}) = b$. Let $\succ_x^{(2)}$ be $\succ_x^{(1)}$ when we move all the woman w such that $b \succ_t w \succ_t w^*$ to be below b and above w^* in $\succ_x^{(2)}$, and all the woman w such that $w^* \succ_t w$ to be below w^* in $\succ_x^{(2)}$. According to Corollary 1, $\mu_x(\hat{m}, \succ_x^{(2)}) = b$. Let $\succ_x^{(3)}$ be $\succ_x^{(2)}$ when we move all the woman w such that $w \succ_t b$ to be above b in $\succ_x^{(3)}$. Assume by contradiction that $\mu_x(\hat{m}, \succ_t) = w^*$. By Lemma 1, $\mu_x(\hat{m}, \succ_x^{(3)}) = w^*$. On the other hand, each woman w such that $w^* \succ_x^{(3)} w$ is also below w^* in $\succ_x^{(2)}$. Therefore, according to Corollary 1, $\mu_x(\hat{m}, \succ_x^{(2)}) = w^*$, which is a contradiction. Therefore, $\mu_x(\hat{m}, \succ_t) \neq w^*$. □

B.4 Proof of Theorem 2

Proof. Clearly, the algorithm runs in polynomial time since there are three loops, where the three loops together iterate at most $2k$ times, and the running time of the GS matching algorithm is in $O(k^2)$. In addition, if the algorithm returns a preference order, which is a manipulative vote for the manipulator r , then w^* will be the match of \hat{m} by the GS algorithm. We need to show that if there exists a preference order for the manipulator r that makes w^* the match of \hat{m} , then our algorithm will find such a preference order for r . Assume that a manipulative vote, ℓ_t , exists, which makes w^* the match of \hat{m} . That is, $\mu(\hat{m}, \ell_t) = w^*$. Then, by Lemma 2, the algorithm finds a preference order for \hat{m} that makes w^* his match (i.e., the preference order \succ_x), and thus it does not return false in line 9. We show that Algorithm 1 returns ℓ_r in line 19.

we begin by analyzing ℓ_r after phase 1. Since w^* is positioned in the highest position in ℓ_r , then $s(w^*, \ell_r) \geq s(w^*, \ell_t)$. Since ℓ_t is a successful manipulation, then w^* is preferred over every $b \in B$ in $\mathcal{F}(\mathcal{L} \cup \{\ell_t\})$ according to Lemma 3. In addition, every $b \in B$ gets a score of 0 from ℓ_r , and thus w^* is preferred over every $b \in B$ in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$. Note that when Algorithm 1 changes $\mathcal{F}(\mathcal{L})$ to \succ_x it only re-position the women $b \in B$. That is, for every $w \in W \setminus B$ such that $w^* \succ_x w$, w is also less preferred than w^* in $\mathcal{F}(\mathcal{L})$. However, since w^* is positioned in the highest position in ℓ_r , then for every $w \in W \setminus B$ such that $w^* \succ_x w$, w is also less preferred than w^* in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$. Overall, every w such that $w^* \succ_x w$ is also less preferred than w^* in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$, and thus according to Corollary 1, $\mu(\hat{m}, \ell_r) = w^*$.

We now show that Algorithm 1 (lines 14-17) can assign scores to all the women $b \in B$ such that $\mu(\hat{m}, \ell_r) = w^*$ in line 19 (i.e., ℓ_r is a successful manipulation). For any $b \in B$, if $s(b, \ell_r) \leq s(b, \ell_t)$ then $s(b, \ell_r) \leq s(b, \mathcal{L}, \ell_t)$. Since $(w^*, \mathcal{L}, \ell_t) > (b, \mathcal{L}, \ell_t)$ then

$(w^*, \mathcal{L}, \ell_r) > (b, \mathcal{L}, \ell_r)$. Otherwise, let $b \in B$ be a woman such that $s(b, \ell_r) > s(b, \ell_t)$ and let $s = s(b, \ell_r)$. There are s women from B below b in ℓ_r . According to the pigeonhole principle, at least one of the women from B , denoted b' , gets a score of at least s from ℓ_t . That is, $s(b', \ell_t) \geq s(b, \ell_r)$. By the algorithm construction, all of the women $b'' \in B$ that are positioned lower than b in ℓ_r are positioned higher than b in $\mathcal{F}(\mathcal{L})$. That is, $(b', \mathcal{L}) > (b, \mathcal{L})$. However, $(b', \mathcal{L}, \ell_t) < (w^*, \mathcal{L}, \ell_t)$ and thus $(b, \mathcal{L}, \ell_r) < (w^*, \mathcal{L}, \ell_r)$. Overall, after placing the women from B in ℓ_r , $\forall b \in B$, $(b, \mathcal{L}, \ell_r) < (w^*, \mathcal{L}, \ell_r)$. That is, every w such that $w^* \succ_x w$ is also less preferred than w^* in $\mathcal{F}(\mathcal{L} \cup \{\ell_r\})$, and thus according to Corollary 1, $\mu(\hat{m}, \ell_r) = w^*$. \square

B.5 Proof of Theorem 3

Proof. Clearly, the coalitional MnM-m problem is in NP . Given an instance of the Permutation Sum problem, we build an instance of the coalitional MnM-m problem as follows. There are $q + 3$ men and $q + 3$ women. The men are: m_1, \dots, m_{q+2} and \hat{m} . The women are: w_1, \dots, w_{q+2} and w^* . We assume that w^* is preferred over the other women according to the lexicographical tie-breaking rule. The q women, w_1, \dots, w_q , correspond to the integers X_1, \dots, X_q . By Lemma 1 from [5], we can construct an election in which the non-manipulators cast votes such that:

$$\hat{m} = (w_{q+1}, w_1, \dots, w_q, w^*, w_{q+2}),$$

and the corresponding scores are:

$$(2q + 4 + C, 2q + 4 + C - X_1, \dots, 2q + 4 + C - X_q, C, z),$$

where C is a constant and $z \leq C$. The preference orders of the men beside \hat{m} are the same, and are as follows:

$$(w^*, w_1, \dots, w_{q+2})$$

The preference orders of the women are the same, and are as follows:

$$(\hat{m}, m_1, \dots, m_{q+2})$$

We show that two manipulators can make the man \hat{m} the match of woman w^* iff the Permutation Sum problem has a solution.

(\Leftarrow) Suppose we have two permutations σ and π of 1 to q such that $\sigma(i) + \pi(i) = X_i$. Let σ^{-1} be the inverse function of σ , i.e., $i = \sigma^{-1}(w)$ if $w = \sigma(i)$. We define $\pi^{-1}(w)$ similarly. We construct the following two manipulative votes:

$$(w^*, w_{q+2}, w_{\sigma^{-1}(q)}, \dots, w_{\sigma^{-1}(1)}, w_{q+1})$$

$$(w^*, w_{q+2}, w_{\pi^{-1}(q)}, \dots, w_{\pi^{-1}(1)}, w_{q+1})$$

Since $\sigma(i) + \pi(i) = X_i$ and $z \leq C$, the preference order of \hat{m} , $\mathcal{F}(\mathcal{L}_R, \mathcal{L})$ is:

$$(w^*, w_1, \dots, w_q, w_{q+1}, w_{q+2})$$

since the corresponding scores are:

$$(2q + 4 + C, 2q + 4 + C - X_1 + X_1, \dots, 2q + 4 + C - X_q + X_q,$$

$$2q + 4 + C, 2(q + 1) + z).$$

Therefore, w^* is the match of \hat{m} .

(\Rightarrow) Assume we have a successful manipulation. Such manipulation must ensure that none of the women w_1, \dots, w_{q+2} are placed in the first positions in the preference order of \hat{m} . That is, to ensure that woman w^* is ranked higher than woman w_{q+1} , both manipulators have to place w^* in the highest position in their preferences, and w_{q+1} in the lowest position in their preferences. Thus, the score of woman w^* in the preference of \hat{m} will be $2q + 4 + C$. Let $\sigma(i)$ be a function that determines the score where the first manipulator is assigned to woman w_i . $\pi(i)$ is defined similarly for the second manipulator. Since the manipulation is successful, for every i , $1 \leq i \leq q$,

$$2q + 4 + C - X_i + \sigma(i) + \pi(i) \leq 2q + 4 + C,$$

and thus,

$$\sigma(i) + \pi(i) \leq X_i.$$

Since $\sum_{i=1}^q X_i = q(q+1)$,

$$\sum_{i=1}^q \sigma(i) + \pi(i) \leq q(q+1).$$

On the other hand, since w_{q+1} is placed in the lowest position by both manipulators,

$$\sum_{i=1}^q \sigma(i) \geq \frac{q(q+1)}{2}$$

and

$$\sum_{i=1}^q \pi(i) \geq \frac{q(q+1)}{2}.$$

Therefore, $\sum_{i=1}^q \sigma(i) + \pi(i) = q(q+1)$, and $\sum_{i=1}^q \sigma(i) = \sum_{i=1}^q \pi(i) = \frac{q(q+1)}{2}$. That is, σ and π are permutations of 1 to q . Moreover, since there is no slack in the inequalities,

$$\sigma(i) + \pi(i) = X_i.$$

That is, there is a solution to the Permutation Sum problem. \square

B.6 Proof of Lemma 4

Proof. Zuckerman et al. [25] define a set of candidates G_W , and show that their algorithm places them at the $|G_W|$ lowest positions (Lemma 3.5 in [25]). Even though our definition of the set D is slightly different, our algorithm handles the set D in the same way that their algorithm handles the set G_W . Thus, the women in D are placed in each stage in the $|D|$ lowest positions. \square

B.7 Proof of Lemma 5

Proof. Assume that there is a successful manipulation \mathcal{L}_T with $|R| - 1$ manipulators. Recall that $s(w, \mathcal{L}, \mathcal{L}_T)$ is the Borda score of a woman w in $(\mathcal{L}, \mathcal{L}_T)$. Since Algorithm 2 places the woman w^* at the highest positions, then:

$$s_{|R|-1}(w^*) = (s(w^*, \mathcal{L}) + \sum_{j=1}^{|R|-1} k - 1) \geq s(w^*, \mathcal{L}, \mathcal{L}_T) \quad (1)$$

Since Algorithm 2 (as proved in Lemma 4) places all of the women $d \in D$ at the $|D|$ lowest positions, then:

$$q(D) = \frac{1}{|D|} \left(\sum_{d \in D} s(d, \mathcal{L}) + \sum_{j=1}^{|R|-1} \sum_{i=0}^{|D|-1} i \right) \leq \quad (2)$$

$$\frac{1}{|D|} \sum_{d \in D} s(d, \mathcal{L}, \mathcal{L}_T) =: q^T(D)$$

Combining the equations and the assumption that $s_{|R|-1}(w^*) < q(D)$, we get that $q^T(D) > s(w^*, \mathcal{L}, \mathcal{L}_T)$. Since $q^T(D)$ is an average, there is at least one $d \in D$ such that $s(d, \mathcal{L}, \mathcal{L}_T) \geq q^T(D)$. Therefore, $s(d, \mathcal{L}, \mathcal{L}_T) > s(w^*, \mathcal{L}, \mathcal{L}_T)$. However, since d is positioned above w^* , then \hat{m} 's match will not be w^* , according to Lemma 3, which is a contradiction to the assumption that \mathcal{L}_T is a successful manipulation. Therefore, there is no manipulation that makes w^* the match of \hat{m} . Clearly, if Algorithm 2 returns \mathcal{L}_R , then it is a successful manipulation. Thus, if there is no manipulation, then the algorithm will return false. \square

B.8 Proof of Lemma 6

Proof. Since Algorithm 2 builds the preference order \succ_x in an identical way to Algorithm 1, then according to Lemma 2, Algorithm 2 will find a preference order \succ_x such that $w^* = \mu_x(\hat{m}, \succ_x)$. Recall that for every $w \in W \setminus \{B\}$ such that $w^* \succ_x w$, w is also less preferred than w^* in $\mathcal{F}(\mathcal{L})$. However, since w^* is positioned in the highest position in each $\ell_r \in \mathcal{L}_R$, then for every $w \in W \setminus \{B\}$ such that $w^* \succ_x w$, w is also less preferred than w^* in $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. We now show that for all $b \in B \setminus D$, it holds that $s_{|R|}(b) \leq \min_{d \in D} \{s_{|R|}(d)\}$. Assume by contradiction that there exists $b \in B \setminus D$ and $d \in D$, such that $s_{|R|}(b) > s_{|R|}(d)$. Then, by the definition of D , $b \in D$, which contradicts the choice of b . Since $\max_{d \in D} \{s_{|R|}(d)\} < s_{|R|}(w^*)$, then for all $b \in B$, it holds that $s_{|R|}(b) < s_{|R|}(w^*)$. Overall, we get that for all $w \in W$ such the $w^* \succ_x w$ w is also less preferred over w^* according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. Therefore, by Corollary 1, w^* is \hat{m} 's match with the preference $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. That is, the algorithm finds a manipulation that makes w^* the match of \hat{m} . \square

B.9 Proof of Lemma 7

Proof. Zuckerman et al. [25] define a set of candidates G_W , and show that the scores of the candidates in G_W are 1-dense (Lemma 3.12 in [25]). Even though our definition of the set D is slightly different, the set of scores $\{s_{|R|-1}(d) : d \in D\}$ is essentially identical to the set of scores of the candidates in G_W . Thus, it is 1-dense. \square

B.10 Proof of Lemma 8

Proof. Sort the members of D by their scores after stage $|R| - 1$ in an increasing order, i.e., $D = \{d_1, \dots, d_{|D|}\}$ such that for all $1 \leq t \leq |D| - 1$, $s_{|R|-1}(d_t) \leq s_{|R|-1}(d_{t+1})$. Thus, $d_1 = \arg \min_{d \in D} \{s_{|R|-1}(d)\}$. Denote for $1 \leq t \leq |D|$, $g_t = s_{|R|-1}(d_t) + |D| - t$, and let $G = \{g_1, \dots, g_{|D|}\}$. Note that according to Algorithm 2, for any t , $s_{|R|}(d_t) = g_t$. Now, since the set $\{s_{|R|-1}(d) : d \in D\}$ is 1-dense (according to Lemma 7) then for any $1 \leq t \leq |D| - 1$, $g_t \geq g_{t+1}$. Thus, for any $t > 1$, $g_1 \geq g_t$. That is, $g_1 = \max_{d \in D} \{s_{|R|}(d)\} = s_{|R|-1}(d_1) + |D| - 1 = \min_{d \in D} \{s_{|R|-1}(d)\} + |D| - 1$. Clearly, $q(D) \geq \min_{d \in D} \{s_{|R|-1}(d)\}$, and thus $\max_{d \in D} \{s_{|R|}(d)\} \leq q(D) + |D| - 1$. \square

B.11 Proof of Corollary 2

Proof. We construct the preference order \succ'_w by starting from \succ_w and performing a sequence of swaps of two adjacent men till the resulting preference order is \succ'_w . We show that each swap does not change the set of proposals, by repeatedly invoking Lemma 9.

We begin by positioning the most preferred man according to \succ'_w , using swaps of two adjacent men. That is, if m_{st} is the most preferred man according to \succ'_w , we swap pairs (m, m_{st}) , $m \in M$, until m_{st} is placed in the first position in \succ'_w . We call these swaps the

swaps of m_{st} . We then position the second preferred man using his swaps, and so on. Clearly, this process terminates since the number of men is finite. Let $\succ_w^{(t)}$ be \succ_w after t swaps. That is, $\succ_w^{(0)}$ is \succ_w , $\succ_w^{(1)}$ is \succ_w after one swap, and $\succ_w^{(t+1)}$ is $\succ_w^{(t)}$ after one swap. We show that for every $t \geq 0$, $o_x(w, \succ_w^{(t)}) = o_x(w, \succ_w^{(t+1)})$ and thus $o(w) = o_x(w, \succ_w^{(0)}) = o_x(w, \succ_w')$. Let (m_i, m_j) be the pair of adjacent men that swap their positions when moving from $\succ_w^{(t)}$ to $\succ_w^{(t+1)}$. That is, $m_i \succ_w^{(t)} m_j$ and $m_j \succ_w^{(t+1)} m_i$. Recall that for every $m_1, m_2 \in M$, if $m_1 \succ_w' m_2$, then all the swaps of m_1 are executed before all the swaps of m_2 . In addition, since m_{nd} is the most preferred man among $o(w) \setminus \{m^*\}$ according to \succ_w' and $m^* \succ_w' m_{nd}$, then the following cases are not possible:

1. $m_i = m^*$ and $m_j \in o_x(w, \succ_w^{(t)})$.
2. $m_i \in o_x(w, \succ_w^{(t)})$ and $m_j = m^*$.
3. $m_i = m_{nd}$ and $m_j \in o_x(w, \succ_w^{(t)})$.
4. $m_i \in o_x(w, \succ_w^{(t)})$ and $m_j = m_{nd}$.
5. $m_i = m^*$ and $m_j = m_{nd}$.

We thus need to consider only the following two cases:

1. $m_i \notin o_x(w, \succ_w^{(t)})$ or $m_j \notin o_x(w, \succ_w^{(t)})$. According to the GS algorithm, a swap of such m_i and m_j cannot change the response of w (either an acceptance or rejection). Therefore, $o_x(w, \succ_w^{(t)}) = o_x(w, \succ_w^{(t+1)})$.
2. $m_i, m_j \in o_x(w, \succ_w^{(t)}) \setminus \{m^*, m_{nd}\}$. We use case 2 of Lemma 9 for this case. Assume to contradiction that $o_x(w, \succ_w^{(t)}) \neq o_x(w, \succ_w^{(t+1)})$. There are two possible cases:
 - (a) There exists a man $o \in o_x(w, \succ_w^{(t)})$ such that $o \notin o_x(w, \succ_w^{(t+1)})$. By case 2 of Lemma 9, $m^* = \mu_x(w, \succ_w^{(t)}) = \mu_x(w, \succ_w^{(t+1)})$. Let \succ^o be $\succ_w^{(t+1)}$ such that o is positioned above m^* . We can construct \succ^o from $\succ_w^{(t+1)}$ by swaps of o . Since $o \notin o_x(w, \succ_w^{(t+1)})$, then by case 1 of Lemma 9, $\mu_x(w, \succ_w^{(t+1)}) = \mu_x(w, \succ^o)$. We now swap m_j and m_i in \succ^o , and thus $m_i \succ^o m_j$ as in $\succ_w^{(t)}$. Let $Pre^o \subset o_x(w, \succ_w^{(t)})$ be the set of proposals that w receives before she receives the proposal o . Note that all the men $o \in Pre^o$ are in the same order in $\succ_w^{(t)}$ and in \succ^o . Therefore, the response of woman w is the same for all the proposals $o \in Pre^o$ and thus $o \in o_x(w, \succ^o)$. Therefore, $m^* \neq \mu_x(w, \succ^o)$, which is a contradiction.
 - (b) There exists a man $o \notin o_x(w, \succ_w^{(t)})$ such that $o \in o_x(w, \succ_w^{(t+1)})$. Using a similar argument to case (a) above (i.e., we now construct \succ^o from $\succ_w^{(t)}$) we get that in this case also $o_x(w, \succ_w^{(t)}) = o_x(w, \succ_w^{(t+1)})$.

□

B.12 Proof of Theorem 5

Proof. Clearly, the algorithm runs in polynomial time since there are three loops, where the three loops together iterate at most k^2 times, and the running time of the GS matching algorithm is in $O(k^2)$. In addition, if the algorithm returns a preference order, which is a manipulative vote for the manipulator r , then m^* will be the match of \hat{w} by the GS algorithm. We need to show that if there exists a preference order for the manipulator r

that makes m^* the match of \hat{w} , then our algorithm will find such a preference order for r . Assume that a manipulative vote, ℓ_t , exists, which makes m^* the match of \hat{w} . That is, $\mu(\hat{w}, \ell_t) = m^*$. We show that Algorithm 3 returns ℓ_r in line 20. Let $\ell_{r(1)}$ be the preference order ℓ_r after phase 1 of the algorithm. $\ell_{r(2)}$ and $\ell_{r(3)}$ are defined similarly. Note that $\ell_{r(3)}$ is the preference order ℓ_r that is returned by the algorithm in line 20.

Algorithm 3 iterates over all $m_{nd} \in M \setminus \{m^*\}$, and thus there exists an iteration in which m_{nd} is the second preferred proposal among $o(\hat{w}, \ell_t)$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_t\})$. Let $\ell_{t(1)}$ be the preference order ℓ_t where m^* and m_{nd} are placed in the same positions as in $\ell_{r(1)}$. Note that m^* and m_{nd} are placed in $\ell_{r(1)}$ such that m^* is preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(1)}\})$, and thus m^* is preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(1)}\})$. In addition, m_{nd} is positioned in $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(1)}\})$ not lower than in $\mathcal{F}(\mathcal{L} \cup \{\ell_t\})$. Therefore, m_{nd} is the most preferred man among $o(\hat{w}, \ell_t) \setminus \{m^*\}$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(1)}\})$. By Corollary 2, $o(\hat{w}, \ell_t) = o(\hat{w}, \ell_{t(1)})$ and $\mu(\hat{w}, \ell_{t(1)}) = m^*$. Thus, m_{nd} is the second preferred proposal among $o(\hat{w}, \ell_{t(1)})$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(1)}\})$.

Let $\ell_{t(2)}$ be the preference order $\ell_{t(1)}$ where the men $m \notin o(\hat{w}, \ell_{t(1)})$ are placed in the highest positions in $\ell_{t(2)}$ without changing the positions of m^* and m_{nd} (similar to the positioning of the men $m \notin o(\hat{w}, \ell_{r(2)})$ after phase 2 of the algorithm). That is, m^* is preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(2)}\})$ and m_{nd} is the most preferred man among $o(\hat{w}, \ell_{t(1)}) \setminus \{m^*\}$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(2)}\})$. We can thus use (again) Corollary 2 to get that $o(\hat{w}, \ell_{t(1)}) = o(\hat{w}, \ell_{t(2)})$ and $\mu(\hat{w}, \ell_{t(2)}) = m^*$.

Recall that at the end of phase 1 of Algorithm 3, m^* and m_{nd} are placed in $\ell_{r(1)}$ such that m^* is preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(1)}\})$. In addition, m_{nd} is positioned in $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(1)}\})$ not lower than in $\mathcal{F}(\mathcal{L} \cup \{\ell_{t(2)}\})$, since they are placed in the same position in $\ell_{r(1)}$ and $\ell_{t(2)}$ and the other men in $\ell_{r(1)}$ get a score of 0 from $\ell_{r(1)}$. Specifically, the men $m \in o(\hat{w}, \ell_{t(2)})$ also get a score of 0 from $\ell_{r(1)}$ and thus m_{nd} is the most preferred man among $o(\hat{w}, \ell_{t(2)}) \setminus \{m^*\}$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(1)}\})$. By Corollary 2, $o(\hat{w}, \ell_{t(2)}) = o(\hat{w}, \ell_{r(1)})$ and $\mu(\hat{w}, \ell_{r(1)}) = m^*$. Since in phase 2 of Algorithm 3 we place only men $m \notin o(\hat{w}, \ell_{r(1)})$, then, we can (again) use Corollary 2 to show that $o(\hat{w}, \ell_{t(2)}) = o(\hat{w}, \ell_{r(2)})$, $\mu(\hat{w}, \ell_{r(2)}) = m^*$, and m_{nd} is the second preferred proposal among $o(\hat{w}, \ell_{r(2)})$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(2)}\})$.

We now show that Algorithm 3 (lines 14-18) can assign scores to all the men $m \in B^{nd}$ such that ℓ_r is a successful manipulation. For any $m \in B^{nd}$, if $s(m, \ell_{r(3)}) \leq s(m, \ell_{t(2)})$ then $s(m, \mathcal{L}, \ell_{r(3)}) \leq s(m, \mathcal{L}, \ell_{t(2)})$. Since $(m_{nd}, \mathcal{L}, \ell_{t(2)}) > (m, \mathcal{L}, \ell_{t(2)})$ and $s(m_{nd}, \mathcal{L}, \ell_{t(2)}) = s(m_{nd}, \mathcal{L}, \ell_{r(3)})$ then $(m_{nd}, \mathcal{L}, \ell_{r(3)}) > (m, \mathcal{L}, \ell_{r(3)})$. Otherwise, let $m \in B^{nd}$ be a man such that $s(m, \ell_{r(3)}) > s(m, \ell_{t(2)})$ and let $s = s(m, \ell_{r(3)})$. By the algorithm construction, there are s men from B^{nd} below m in $\ell_{r(3)}$. According to the pigeonhole principle, at least one of the men from B^{nd} , denoted m' , gets a score of at least s from $\ell_{t(2)}$. That is, $s(m', \ell_{t(2)}) \geq s(m, \ell_{r(3)})$. By the algorithm construction, all the men $m'' \in B^{nd}$ that are positioned lower than m in $\ell_{r(3)}$ are positioned higher than m in $\mathcal{F}(\mathcal{L})$. That is, $(m', \mathcal{L}) > (m, \mathcal{L})$. However, $(m', \mathcal{L}, \ell_{t(2)}) < (m_{nd}, \mathcal{L}, \ell_{t(2)})$ and thus $(m, \mathcal{L}, \ell_{r(3)}) < (m_{nd}, \mathcal{L}, \ell_{r(3)})$. Overall, after placing the men from B^{nd} in $\ell_{r(3)}$, $\forall m \in B^{nd}$, $(m, \mathcal{L}, \ell_{r(3)}) < (m_{nd}, \mathcal{L}, \ell_{r(3)})$. That is, m_{nd} is the most preferred man among $o(\hat{w}, \ell_{r(2)}) \setminus \{m^*\}$ according to $\mathcal{F}(\mathcal{L} \cup \{\ell_{r(2)}\})$. In addition $(m_{nd}, \mathcal{L}, \ell_{r(3)}) < (m^*, \mathcal{L}, \ell_{r(3)})$ and thus by Corollary 2, $\mu(\hat{w}, \ell_{r(3)}) = m^*$. \square

B.13 Proof of Theorem 6

Proof. Clearly, the coalitional MnM-w problem is in *NP*. Given an instance of the Permutation Sum problem we build an instance of the coalitional MnM-w problem as follows. There are $q + 3$ men and $q + 3$ women. The men are: m_1, \dots, m_{q+2} and m^* . The women are: w_1, \dots, w_{q+2} and \hat{w} . We assume that m^* is preferred over the other women according to the lexicographical tie-breaking rule. The q men, m_1, \dots, m_q , correspond to the integers X_1, \dots, X_q . By Lemma 1 from [5], we can construct an election in which the

non-manipulators cast votes such that:

$$\hat{w} = (m_{q+1}, m_1, \dots, m_q, m^*, m_{q+2}),$$

and the corresponding scores are:

$$(2q + 4 + C, 2q + 4 + C - X_1 \dots, 2q + 4 + C - X_q, C, z),$$

where C is a constant and $z \leq C$. The preference orders of the women beside \hat{w} are the same, and are as follows:

$$(m^*, m_1, \dots, m_{q+2})$$

The preference orders of the men are the same, and are as follows:

$$(\hat{w}, w_1, \dots, w_{q+2})$$

We show that two manipulators can make the woman \hat{w} the match of man m^* iff the Permutation Sum problem has a solution.

(\Leftarrow) Suppose we have two permutations σ and π of 1 to q such that $\sigma(i) + \pi(i) = X_i$. Let σ^{-1} be the inverse function of σ , i.e., $i = \sigma^{-1}(m)$ if $m = \sigma(i)$. We define $\pi^{-1}(m)$ similarly. We construct the following two manipulative votes:

$$(m^*, m_{q+2}, m_{\sigma^{-1}(q)}, \dots, m_{\sigma^{-1}(1)}, m_{q+1})$$

$$(m^*, m_{q+2}, m_{\pi^{-1}(q)}, \dots, m_{\pi^{-1}(1)}, m_{q+1})$$

Since $\sigma(i) + \pi(i) = X_i$ and $z \leq C$, the preference order of \hat{w} , $\mathcal{F}(\mathcal{L}_R, \mathcal{L})$ is:

$$(m^*, m_1, \dots, m_{q+2})$$

since the corresponding scores are:

$$(2q + 4 + C, 2q + 4 + C - X_1 + X_1, \dots, 2q + 4 + C - X_q + X_q, \\ 2q + 4 + C, 2(q + 1) + z).$$

Therefore, m^* is the match of \hat{w} .

(\Rightarrow) Assume we have a successful manipulation. Such a manipulation must ensure that none of the men m_1, \dots, m_{q+2} are placed in the first positions in the preference order of \hat{w} . That is, to ensure that man m^* is ranked higher than man m_{q+1} , both manipulators have to place m^* in the highest position in their preferences, and m_{q+1} in the lowest position in their preferences. Thus, the score of man m^* in the preference of \hat{w} will be $2q + 4 + C$. Let $\sigma(i)$ be a function that determines the score where the first manipulator is assigned to man m_i . $\pi(i)$ is defined similarly for the second manipulator. Since the manipulation is successful, for every i , $1 \leq i \leq q$,

$$2q + 4 + C - X_i + \sigma(i) + \pi(i) \leq 2q + 4 + C,$$

and thus,

$$\sigma(i) + \pi(i) \leq X_i.$$

Since $\sum_{i=1}^q X_i = q(q + 1)$,

$$\sum_{i=1}^q \sigma(i) + \pi(i) \leq q(q + 1).$$

On the other hand, since m_{q+1} is placed in the lowest position by both manipulators,

$$\sum_{i=1}^q \sigma(i) \geq \frac{q(q+1)}{2}$$

and

$$\sum_{i=1}^q \pi(i) \geq \frac{q(q+1)}{2}.$$

Therefore, $\sum_{i=1}^q \sigma(i) + \pi(i) = q(q+1)$, and $\sum_{i=1}^q \sigma(i) = \sum_{i=1}^q \pi(i) = \frac{q(q+1)}{2}$. That is, σ and π are permutations of 1 to q . Moreover, since there is no slack in the inequalities,

$$\sigma(i) + \pi(i) = X_i.$$

That is, there is a solution to the Permutation Sum problem. \square

B.14 Proof of Theorem 7

In order to prove Theorem 7 we use the following definitions, which correspond to the definitions we used for proving Theorem 4. Given $m_{nd} \in M \setminus \{m^*\}$, let $D_0^{nd} = \{d_0\}$, where $(d_0, \mathcal{L}) > (m, \mathcal{L})$, and $d_0, m \in B^{nd}$. For each $s = 1, 2, \dots$, let $D_s^{nd} \subseteq B^{nd}$ be $D_s^{nd} = D_{s-1}^{nd} \cup \{d : d \in B^{nd} \text{ was ranked above some } d' \in D_{s-1}^{nd} \text{ according to } \mathcal{F}(\mathcal{L}, \mathcal{L}_R) \text{ in some stage } l, 1 \leq l \leq |R|\}$. Now, let $D^{nd} = \bigcup_{0 \leq s} D_s^{nd}$. Note that $\forall s D_s^{nd} \neq D_{s-1}^{nd}$, and s does not necessarily equal $|R|$. Let $s_\ell(m)$ be the score of man m in $\mathcal{F}(\mathcal{L}, \mathcal{L}_R)$ after stage ℓ . Let \mathcal{L}_{R-2} be a preference profile of $|R| - 2$ manipulators.

The proof of Theorem 7 relies on several Lemmata, which reminiscent Lemmata 4-8, and its general intuition is also somehow similar to the intuition behind the proof of Theorem 4.

We begin with a basic lemma that clarifies where Algorithm 4 places the men of D^{nd} .

Lemma 10. *Given $m_{nd} \in M \setminus \{m^*\}$, the men in D^{nd} are placed in each stage l , $1 \leq l \leq |R|$ among the $|D^{nd}| + 1$ lowest positions.*

Proof. Zuckerman et al. [25] define a set of candidates G_W , and show that their algorithm places them at the $|G_W|$ lowest positions (Lemma 3.5 in [25]). Even though our definition of the set D^{nd} is slightly different, our algorithm handles the set D^{nd} in almost the same way that their algorithm handles the set G_W . Indeed, the algorithm might place m_{nd} in one of the $|D^{nd}|$ lowest positions in phase 1. Thus, the men in D^{nd} are placed in each stage in the $|D^{nd}| + 1$ lowest positions. \square

We now show that if there exists a successful manipulation, then Algorithm 4 will successfully finish phase 1. That is, the algorithm will find a man m_{nd} such that $\mu(\hat{w}, \mathcal{L}_R) = m^*$ and $m_{nd} \in o(\hat{w}, \mathcal{L}_R)$ at the end of phase 1.

Lemma 11. *Assume that a preference profile \mathcal{L}_T with $|R|$ manipulators exists such that $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Let $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ be the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$. If the input to Algorithm 4 consists of $|R|$ manipulators, then in the iteration in which $m_{nd} = m'_{nd}$, $(m_{nd}, \mathcal{L}, \mathcal{L}_R) < (m^*, \mathcal{L}, \mathcal{L}_R)$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$. Similarly, if \mathcal{L}_T consists of $|R| - 2$ manipulators, and the input to Algorithm 4 consists of $|R| - 2$ manipulators, then in the iteration in which $m_{nd} = m'_{nd}$, $(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) < (m^*, \mathcal{L}, \mathcal{L}_{R-2})$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2})$.*

Proof. Clearly, since m^* is preferred over m'_{nd} , then for the iteration $m_{nd} = m'_{nd}$ it satisfies $|R|(k-1) \geq \text{gap}$. Otherwise, even if each manipulator would place m^* in the highest

position and m_{nd} in the lowest position m_{nd} would be preferred over m^* which contradicts the existence of \mathcal{L}_T .

Next we show that m^* is preferred over m_{nd} according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$. We consider the two cases of the algorithm:

- $|R| \geq \text{gap}$. We show that $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$ and $(m_{nd}, \mathcal{L}, \mathcal{L}_R) < (m^*, \mathcal{L}, \mathcal{L}_R)$.

According to the algorithm, $s(m^*, \mathcal{L}, \mathcal{L}_R) = s(m^*, \mathcal{L}) + \max(\text{gap} + \lceil (|R| - \text{gap})/2 \rceil, 0)(k-1) + (|R| - \max(\text{gap} + \lceil (|R| - \text{gap})/2 \rceil, 0))(k-2)$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + \max(\text{gap} + \lceil (|R| - \text{gap})/2 \rceil, 0)(k-2) + (|R| - \max(\text{gap} + \lceil (|R| - \text{gap})/2 \rceil, 0))(k-1)$. We show that $s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) \geq 0$. $s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) =$

$s(m^*, \mathcal{L}) + 2 \max(\text{gap} + \lceil (|R| - \text{gap})/2 \rceil, 0) - |R| - s(m_{nd}, \mathcal{L}) \geq s(m^*, \mathcal{L}) + \max(|R| + \text{gap}, 0) - |R| - s(m_{nd}, \mathcal{L})$. We consider the two cases:

1. $|R| + \text{gap} > 0$. Then, $s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) \geq s(m^*, \mathcal{L}) - |R| + |R| + \text{gap} - s(m_{nd}, \mathcal{L}) \geq 0$. In addition, $s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) \leq 1$. $s(m^*, \mathcal{L}) + \max(|R| + \text{gap}, 0) - |R| - s(m_{nd}, \mathcal{L})$ If one of the manipulators that positioned m^* above m_{nd} , would change the positions of m_{nd} and m^* , then m^* would be preferred over m_{nd} .
2. $|R| + \text{gap} \leq 0$. $s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m^*, \mathcal{L}) - |R| + 0 - s(m_{nd}, \mathcal{L}) \geq 0$. Then in each manipulator m_{nd} is positioned in the highest position.

Thus, it is not possible to give m_{nd} a higher score. That is, $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$.

- $|R| < \text{gap}$. We show that $(m_{nd}, \mathcal{L}, \mathcal{L}_R) < (m^*, \mathcal{L}, \mathcal{L}_R)$.

According to the algorithm, $s(m^*, \mathcal{L}, \mathcal{L}_R) = s(m^*, \mathcal{L}) + |R|(k-1)$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (s_{m_{nd}} \bmod |R|) \lceil \frac{|R|(k-1) - \text{gap}}{|R|} \rceil + (|R| - (s_{m_{nd}} \bmod |R|)) \lfloor \frac{|R|(k-1) - \text{gap}}{|R|} \rfloor$.

$s(m^*, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m^*, \mathcal{L}) + |R|(k-1) - s(m_{nd}, \mathcal{L}) - (s_{m_{nd}} \bmod |R|) \lceil \frac{|R|(k-1) - \text{gap}}{|R|} \rceil - (|R| - (s_{m_{nd}} \bmod |R|)) \lfloor \frac{|R|(k-1) - \text{gap}}{|R|} \rfloor \geq 0$.

Since, $|R|(k-1) \geq \text{gap}$.

Since there exists $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$, there is no option to award m_{nd} a higher score such that it will not overcome m^* according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_R)$ then according to line 16, $m_{nd} \in o(\hat{w}, \mathcal{L}_R)$ after phase 1 and $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) \geq s(m_{nd}, \mathcal{L}, \mathcal{L}_T)$. □

Corollary 3. *Assume that a preference profile \mathcal{L}_T with $|R|$ manipulators exists such that $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Let $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ be the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$. If the input to Algorithm 4 consists of $|R|$ manipulators, then in the iteration in which $m_{nd} = m'_{nd}$ the algorithm will successfully finish phase 1. Similarly, if \mathcal{L}_T consists of $|R| - 2$ manipulators, and the input to Algorithm 4 consists of $|R| - 2$ manipulators, then in the iteration in which $m_{nd} = m'_{nd}$ the algorithm will successfully finish phase 1.*

Proof. According to Lemma 11, $(m_{nd}, \mathcal{L}, \mathcal{L}_R) < (m^*, \mathcal{L}, \mathcal{L}_R)$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$. And therefore in the end of phase 1, $\forall m \in o(\hat{w}, \mathcal{L}_R)$, $(m, \mathcal{L}, \mathcal{L}_R) < (m_{nd}, \mathcal{L}, \mathcal{L}_R)$. Since $\mu(\hat{w}, \mathcal{L}_T) = m^*$ and $m_{nd} \in o(\hat{w}, \mathcal{L}_T)$, then by Corollary 2, $\mu(\hat{w}, \mathcal{L}_R) = m^*$ and $m_{nd} \in o(\hat{w}, \mathcal{L}_R)$ in the end of phase 1. That is, Algorithm 4 successfully finishes phase 1. Clearly, the proof is similar for manipulation with $|R| - 2$ manipulators. □

We now show the relation between the score of m_{nd} and the average score of the men in D^{nd} , when there are $|R| - 2$ manipulators.

Lemma 12. *Assume that a preference profile \mathcal{L}_T with $|R| - 2$ manipulators exists such that $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Let $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ be the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$. Assume that the input to Algorithm 4 consists of $|R| - 2$ manipulators, and let $q(D^{nd})$ be the average score of men in D^{nd} . That is, $q(D^{nd}) = \frac{1}{|D^{nd}|} \sum_{d \in D^{nd}} s(d, \mathcal{L}, \mathcal{L}_{R-2})$. Then, in the iteration in which $m_{nd} = m'_{nd}$, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{|R|-2}) \geq q(D^{nd})$.*

Proof. Let $q^T(D^{nd}) = \frac{1}{|D^{nd}|} \sum_{d \in D^{nd}} s(d, \mathcal{L}, \mathcal{L}_T)$ be the average score of men in D^{nd} in $(\mathcal{L}, \mathcal{L}_T)$. As a corollary of Lemma 10, if $|R| \geq \text{gap}$, then all the men in D^{nd} are placed among the $|D^{nd}|$ lowest positions in the preference profile of each manipulator. That is,

$$q(D^{nd}) = \frac{1}{|D^{nd}|} (\sum_{d \in D^{nd}} s(d, \mathcal{L}) + \sum_{j=1}^{|R|-2} \sum_{i=0}^{|D^{nd}|-1} i) \leq \frac{1}{|D^{nd}|} \sum_{d \in D^{nd}} s(d, \mathcal{L}, \mathcal{L}_T) = q^T(D^{nd}).$$

Now, if $|R| - 2 < \text{gap}$, but $\lfloor \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rfloor \geq |D^{nd}|$ then again all the men in D^{nd} are placed among the $|D^{nd}|$ lowest positions in the preference profile of each manipulator, and $q(D^{nd}) \leq q^T(D^{nd})$. Finally, consider the case in which $|R| - 2 < \text{gap}$, and $\lfloor \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rfloor < |D^{nd}|$. Note that for each $\ell_r \in \mathcal{L}_r$, only men from D^{nd} are placed below m_{nd} , and all the men from D^{nd} are placed among the $|D^{nd}| + 1$ lowest positions. Therefore, if for every $1 \leq i \leq |R| - 2$, m_{nd} is positioned in $\ell_i \in \mathcal{L}_T$ at the same position as in $\ell_i \in \mathcal{L}_R$ or at a lower position, then obviously $q(D^{nd}) \leq q^T(D^{nd})$. If it is not the case, then there is a subset of manipulators $\mathcal{L}_T^+ \subset \mathcal{L}_T$, and a corresponding subset $\mathcal{L}_R^+ \subset \mathcal{L}_{R-2}$, such that m_{nd} is positioned in $\ell_i \in \mathcal{L}_T^+$ at a higher position than in $\ell_i \in \mathcal{L}_R^+$. That is, $\Delta = s(m_{nd}, \mathcal{L}_T^+) - s(m_{nd}, \mathcal{L}_R^+)$ is greater than 0, and thus $\sum_{d \in D^{nd}} s(d, \mathcal{L}_T^+) - s(d, \mathcal{L}_R^+)$ is at most Δ . However, since $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2})$ according to Lemma 11, then there must be a non-empty subset of manipulators $\mathcal{L}_T^- \subset \mathcal{L}_T$, and a corresponding subset $\mathcal{L}_R^- \subset \mathcal{L}_{R-2}$, such m_{nd} is positioned in $\ell_i \in \mathcal{L}_T^-$ at a lower position than in $\ell_i \in \mathcal{L}_R^-$. Moreover, $s(m_{nd}, \mathcal{L}_T^-) - s(m_{nd}, \mathcal{L}_R^-) \geq \Delta$ and thus $\sum_{d \in D^{nd}} s(d, \mathcal{L}_T^-) - s(d, \mathcal{L}_R^-)$ is at least Δ . Therefore, $q(D^{nd}) \leq q^T(D^{nd})$.

Note that $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ is the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$, and $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Therefore, according to Lemma 11, in the iteration in which $m_{nd} = m'_{nd}$, $(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) < (m^*, \mathcal{L}, \mathcal{L}_{R-2})$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2})$. In addition, after phase 1 of the algorithm, $M \setminus \{m_{nd}, m^*\}$ are not placed yet in \mathcal{L}_{R-2} , and thus at the beginning of phase 2, $B^{nd} = o(\hat{w}, \mathcal{L}_T) \setminus \{m^*, m_{nd}\}$ according to Corollary 2. Since $\mu(\hat{w}, \mathcal{L}_T) = m^*$, then for all $d \in o(\hat{w}, \mathcal{L}_T) \setminus \{m^*, m_{nd}\}$, $s(d, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_T)$. Since $D^{nd} \subseteq B^{nd}$, then $q^T(D^{nd}) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_T)$. Since we showed that $q(D^{nd}) \leq q^T(D^{nd})$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2})$, we get that $q(D^{nd}) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2})$. \square

The following lemma shows the relation between the score of m_{nd} and the maximum score of a man in D^{nd} , when there are $|R|$ manipulators. In essence, the lemma characterizes the settings in which Algorithm 4 finds a successful manipulation.

Lemma 13. *Assume that a preference profile \mathcal{L}_T with $|R|$ manipulators exists such that $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Let $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ be the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$. If the input to Algorithm 4 consists of $|R|$ manipulators, and in the iteration in which $m_{nd} = m'_{nd}$, $\max_{d \in D^{nd}} \{s(d, \mathcal{L}, \mathcal{L}_R)\} < s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$, then Algorithm 4 finds a successful manipulation.*

Proof. Since there exists a preference profile \mathcal{L}_T that ensures that $\mu(\hat{w}, \mathcal{L}_T) = m^*$. Then, according to Lemma 11 the algorithm can place m^* and m_{nd} such that in the end of phase 1 $\mu(\hat{w}, \mathcal{L}_R) = m^*$ and $s(m_{nd}, \mathcal{L}, \mathcal{L}_T) \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$.

Even though the set D^{nd} in the proof of Theorem 7 is defined slightly differently than in the proof of Theorem 4, and this Lemma is slightly different than Lemma 6, the proof of this lemma is essentially identical to the proof of Lemma 6. And therefore $\forall m \in o(\hat{w}, \mathcal{L}_R) \setminus \{m^*, m_{nd}\}$ in the end of phase 1, $(m, \mathcal{L}, \mathcal{L}_R) < (m_{nd}, \mathcal{L}, \mathcal{L}_R)$ in the end of phase 2, and according to Corollary 2 $\mu(\hat{w}, \mathcal{L}_R) = m^*$ in the end of phase 2. Thus, Algorithm 4 will find a successful manipulation. \square

Next, we show that the highest score of a man from D^{nd} , when there are $|R|$ manipulators, is not much higher than the average score of the men in D^{nd} , when there are $|R| - 2$ manipulators. We thus first show that the scores of men in D^{nd} according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_{R-2})$ and $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_{R-1})$ are almost 1-dense, as captured by the following definition.

Definition 7. A finite non-empty set of integers B is called almost 1-dense if, when sorting the set in a non-increasing order $b_1 \geq b_2 \geq \dots \geq b_i$ (such that $\{b_1, \dots, b_i\} = B$), there exists at most one index, j , such that $1 \leq j \leq i - 1$, $b_{j+1} = b_j - 2$ and $\forall t \neq j, 1 \leq t \leq i - 1$, $b_{t+1} \geq b_t - 1$ holds.

Lemma 14. Let $D^{nd} = \bigcup_{0 \leq s} D_s^{nd}$, as before. Then for all $s \geq 1$ and $d \in D_s^{nd} \setminus D_{s-1}^{nd}$ there exist $d' \in D_{s-1}^{nd}$, $X \subseteq B^{nd}$ and $j, 0 \leq j \leq |R|$, s.t. $\{s_j(d), s_j(d')\} \cup s_j(X)$ is almost 1-dense.

Proof. Let $s \geq 1$ and $d \in D_s^{nd} \setminus D_{s-1}^{nd}$. By definition, there exist $d' \in D_{s-1}^{nd}$ and a minimal $j, 0 \leq j \leq |R|$, such that d was ranked below d' in stage j , that is $s(d, \ell_j) < s(d', \ell_j)$. We distinguish between two cases: case 1: $j > 1$. In this case d was ranked above d' in stage $j - 1$. So we have:

$$s_{j-1}(d) \geq s_{j-1}(d') \quad (3)$$

$$s_{j-2}(d) \leq s_{j-2}(d') \quad (4)$$

d was ranked above d' in stage $j - 1$, and hence $s(d, \ell_{j-1}) > s(d', \ell_{j-1})$. Denote $g = s(d, \ell_{j-1}) - s(d', \ell_{j-1}) - 1$. We distinguish into two cases:

- Let $d' = d_0, d_1, \dots, d_{g+1} = d$ be the men that got in stage $j - 1$ the points $s(d', \ell_{j-1}), s(d', \ell_{j-1}) + 1, \dots, s(d', \ell_{j-1}) + g + 1$, respectively. In this case m_{nd} was not placed between the men $\{d_0, d_1, \dots, d_{g+1}\}$, but higher than d or lower than d' in stage $j - 1$. This case corresponds to the 1-dense case as proved in Zuckerman et al. [25]. Since 1-dense is almost 1-dense by definition.

- Let $d' = d_0, d_1, \dots, d_g = d$ be the men that got in stage $j - 1$ the points $s(d', \ell_{j-1}), s(d', \ell_{j-1}) + 1, \dots, s(d', \ell_{j-1}) + g + 1$, respectively. In this case m_{nd} was placed between the men $\{d_0, d_1, \dots, d_g\}$, but higher than d' and lower than d in stage $j - 1$. Our purpose is to show that $\{s_{j-1}(d_0), \dots, s_{j-1}(d_g)\}$ is almost 1-dense. By definition of the algorithm,

$$s_{j-2}(d_0) \geq s_{j-2}(d_1) \geq \dots \geq s_{j-2}(d_g) \quad (5)$$

Let m_{nd} be placed between d_i and d_{i+1} , $i \in \{0, \dots, g\}$. That is, $s(d_i, \ell_{j-1}) + 2 = s(d_{i+1}, \ell_{j-1})$. Denote $u_t = s_{j-2}(d_t) + s(d', \ell_{j-1})$ for $0 \leq t \leq g$. Then

$$\forall t, 0 \leq t \leq i, s_{j-1}(d_t) = u_t + t. \quad (6)$$

$$\forall t, i + 1 \leq t \leq g, s_{j-1}(d_t) = u_t + t + 1. \quad (7)$$

We need to show that $\{s_{j-1}(d_t) | 0 \leq t \leq g\}$ is almost 1-dense. It is enough to show that:

1. For all $t, 0 \leq t < i$, if $u_t + t < u_0$, then there exists $t', t < t' \leq i$ s.t. $u_t + t < u_{t'} + t' \leq u_t + t + 1$

2. For all t , $i + 1 \leq t < g$, if $u_t + t + 1 < u_0$, then there exists t' , $t < t' \leq g$ s.t. $u_t + t < u_{t'} + t' \leq u_t + t + 1$
3. For all t , $0 < t \leq i$, if $u_t + t > u_0$, then there exists t' , $0 \leq t' < t$ s.t. $u_t + t - 1 \leq u_{t'} + t' < u_t + t$
4. For all t , $i + 1 < t \leq g$, if $u_t + t + 1 > u_0$, then there exists t' , $i + 1 \leq t' < t$ s.t. $u_t + t - 1 \leq u_{t'} + t' < u_t + t$
5. For $t = i$ and for $y = i + 1$ that either $u_t + t < u_0$ or $u_y + y + 1 > u_0$. There are 1 of 2 possible cases:
 - (a) If $u_t + t < u_0$, then there exists t' , $t < t' \leq g$ s.t. $u_t + t < u_{t'} + t' + 1 = u_t + t + 2$ and, if $u_y + y + 1 > u_0$, then there exists y' , $0 \leq y' < y$ s.t. $u_y + y + 1 - 1 \leq u_{y'} + y' < u_y + y + 1$
 - (b) If $u_t + t < u_0$, then there exists t' , $t < t' \leq g$ s.t. $u_t + t < u_{t'} + t' + 1 = u_t + t + 1$ and, if $u_y + y + 1 > u_0$, then there exists y' , $0 \leq y' < y$ s.t. $u_y + y + 1 - 2 \leq u_{y'} + y' < u_y + y + 1$

Proof of 1: From (5) we get

$$u_0 \geq \dots \geq u_g \quad (8)$$

Also from (3) and (6) we have $u_0 \leq u_g + g + 1$. Let $0 \leq t < i$ s.t. $u_t + t < u_0$. Let us consider the sequence $u_t + t, u_{t+1} + t + 1, \dots, u_i + i$. Since $u_t + t < u_0 \leq u_i + i$, it follows that there is a minimal index t' , $t < t' \leq i$ s.t. $u_t + t < u_{t'} + t'$. Then $u_{t'-1} + t' - 1 \leq u_t + t$, and thus

$$u_{t'-1} + t' \leq u_t + t + 1. \quad (9)$$

From (8) $u_{t'} \leq u_{t'-1}$, and then

$$u_{t'} + t' \leq u_{t'-1} + t'. \quad (10)$$

Combining (9) and (10) together, we get $u_{t'} + t' \leq u_t + t + 1$. This concludes the proof of 1. for $0 \leq t < i$.

Let $i + 1 \leq t < g$ s.t. $u_t + t + 1 < u_0$. Let us consider the sequence $u_t + t + 1, u_{t+1} + t + 2, \dots, u_g + g + 1$. Since $u_t + t + 1 < u_0 \leq u_g + g + 1$, it follows that there is a minimal index t' , $t < t' \leq g$ s.t. $u_t + t + 1 < u_{t'} + t' + 1$. Then

$$u_{t'-1} + t' \leq u_t + t + 1. \quad (11)$$

From (8) $u_{t'} \leq u_{t'-1}$, and then

$$u_{t'} + t' \leq u_{t'-1} + t'. \quad (12)$$

Combining (11) and (12) together, we get $u_{t'} + t' \leq u_t + t + 1$. This concludes the proof of 1.

The proof of 2. is similar to the proof of 1., by replacing the use of Equation (7) instead of Equation (6). Rest of the equations are the same.

The proof of 3. is analogous to the proof of 1., by choosing t' to be the maximal index such that $u_{t'} + t' < u_t + t$.

The proof of 4. is analogous to the proof of 2., by choosing t' to be the maximal index such that $u_{t'} + t' < u_t + t$.

The proof of 5. considers the position of m_{nd} in ℓ_j between d_i and d_{i+1} . The proof of 5.(a), for the first part let us consider the sequence $u_t + t, u_{t+1} + t + 2, \dots, u_g + g + 1$. Since $u_t + t < u_0 \leq u_g + g + 1$, it follows that there is a minimal index t' , $t < t' \leq g$ s.t. $u_t + t < u_{t'} + t' + 1$. Then $u_{t'-1} + t' \leq u_t + t$, and thus we conclude with the same equation as Equation 9.

From (8) $u_{t'} \leq u_{t'-1}$, and then we have again Equation 10

Combining (9) and (10) together, we get $u_{t'} + t' < u_t + t + 1$. The second part of the proof, is of the proof of 3. This concludes the proof of 5.(a)

The proof of 5.(b) the first part is of the proof of 1. The second part is analogous to 5.(a), by choosing t' to be the maximal index such that $u_{t'} + t' < u_t + t$.

Case 2: $j = 1$. We proceed by essentially reducing this case to Case 1. In Case 2 we have that $s \geq 2$, because otherwise, if $s = 1$, then $d' \in D_0^{nd}$: therefore $s_0(d) \geq s_0(d') = \max_{b \in B^{nd}} \{s_0(b)\} \rightarrow d \in D_0^{nd}$, a contradiction. $d' \notin D_{s-2}^{nd}$, because otherwise, by definition, $d \in D_{s-1}^{nd}$. Therefore there exists $d'' \in D_{s-2}^{nd}$ s.t. d' was ranked below d'' in some stage j' , i.e., $s_{j'-1}(d) \leq s_{j'-1}(d'')$. By combining the last arguments, we get that $s_{j'-1}(d) \leq s_{j'-1}(d')$. Let j_0 be minimal s.t. $s_{j_0}(d) \geq s_{j_0}(d')$. As in stage 1, d was ranked below d' , it holds that $s_0(d) \geq s_0(d')$. If $j_0 = 0$ then $s_0(d) = s_0(d')$, hence $\{s_0(d), s_0(d')\}$ is 0-dense, and in particular 1-dense and almost 1-dense.

Otherwise ($j_0 \neq 0$) it holds that $s_{j_0-1}(d) > s_{j_0-1}(d')$ by the minimality of j_0 . So, we have that

$$s_{j_0}(d') \geq s_{j_0}(d) \quad (13)$$

and

$$s_{j_0-1}(d') \leq s_{j_0-1}(d) \quad (14)$$

These two inequalities are analogous to (3) and (4), with $j - 1$ replaced by j_0 , and the roles of d and d' exchanged. From this point we can proceed exactly as in Case 1, keeping in mind these cosmetic changes. \square

Lemma 15. *Let $H \subseteq B$ such that $s_j(H)$ is almost 1-dense for some $0 \leq j \leq |R|$. Then there exists $H', H \subseteq H' \subseteq B$ such that $s_{|R|}(H')$ is almost 1-dense.*

Proof. The proof is the same as in [25], proof of Lemma 3.11. \square

Lemma 16. *The set D^{nd} is almost 1-dense.*

Proof. The proof is the similar to the one in [25], proof of Lemma 3.12. We Note that here for two almost 1-dense sets A, A' , if $A \cap A' \neq \emptyset$, then $A \cup A'$ is also almost 1-dense, since the almost 1-dense occurs after a manipulator places m_{nd} between to men from B . Since this can happen only once in each manipulator. Clearly, either the set A or the set A' is 1-dense. Otherwise $A \cap A'$ is already almost 1-dense, and this concludes, that $A \cup A'$ is almost 1-dense. \square

Lemma 17. *Assume that the input to Algorithm 4 consists of $|R|$ manipulators. If there exists $m_{nd} \in M \setminus \{m^*\}$ such that the condition in Line 16 does not hold, then $\max_{d \in D^{nd}} \{s(d, \mathcal{L}, \mathcal{L}_R)\} \leq q(D^{nd}) + \frac{3|D^{nd}|}{2} + 1$.*

Proof. Recall that $q(D^{nd})$ is the average score of men in D^{nd} when there are $|R| - 2$ manipulators. Let $q(D^{nd}, R)$ be the average score of men in D^{nd} when there are $|R|$ manipulators. That is, $q(D^{nd}, R) = \frac{1}{|D^{nd}|} \sum_{d \in D^{nd}} s(d, \mathcal{L}, \mathcal{L}_R)$. We define $q(D^{nd}, R - 1)$ similarly. According to Lemma 16, the set $\{s_{|R|}(d) : d \in D^{nd}\}$ is almost 1-dense, and therefore $\max_{d \in D^{nd}} \{s(d, \mathcal{L}, \mathcal{L}_R)\} \leq q(D^{nd}, R) + \frac{|D^{nd}|}{2}$. According to Lemma 4, in each stage each man from D^{nd} is placed among the $|D^{nd}| + 1$ lowest positions. Therefore, $q(D^{nd}, R) + \frac{|D^{nd}|}{2} \leq q(D^{nd}, R - 1) + |D^{nd}| + \frac{1}{2}$. Using this Lemma again we get $q(D^{nd}, R - 1) + |D^{nd}| + \frac{1}{2} \leq q(D^{nd}, R - 2) + \frac{3|D^{nd}|}{2} + 1$ \square

Finally, we show that by adding two manipulators, we can increase the score of m_{nd} by at least $2k - 3$.

Lemma 18. *Assume that the input to Algorithm 4 consists of $|R| - 2$ manipulators. If there exists $m_{nd} \in M \setminus \{m^*\}$ such that the condition in Line 5 does not hold, then when adding 2 additional manipulators, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) + 2k - 3 \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$.*

Proof. Clearly, if in line 5, $(|R| - 2)(k - 1) \geq \text{gap}$, then $|R|(k - 1) \geq \text{gap}$. Let $t = \text{gap} + \lceil (|R| - \text{gap})/2 \rceil$ and let $t_{-2} = \text{gap} + \lceil (|R| - 2 - \text{gap})/2 \rceil$. Note that $t - t_{-2} = 1$. There are three possible cases:

- $|R| - 2 \geq \text{gap}$ and $|R| > \text{gap}$.

When there are $|R|$ manipulators, according to Algorithm 4, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (k - 2) \max(t, 0) + (k - 1)(|R| - \max(t, 0))$. Similarly, when there are $|R| - 2$ manipulators, according to Algorithm 4, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (k - 2) \max(t_{-2}, 0) + (k - 1)(|R| - 2 - \max(t_{-2}, 0))$.

Consider the following three possible cases:

1. $t_{-2} \geq 0$ and $t \geq 0$.

That is, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (k - 2)t + (k - 1)(|R| - t) = s(m_{nd}, \mathcal{L}) - t + (k - 1)|R|$. Similarly, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (k - 2)t_{-2} + (k - 1)(|R| - 2 - t_{-2}) = s(m_{nd}, \mathcal{L}) - t_{-2} + (k - 1)|R| - 2(k - 1)$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = -1 + 2(k - 1) = 2k - 3$.

2. $t_{-2} < 0$ and $t \geq 0$.

That is, $\max(t_{-2}, 0) = 0$, and thus $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (k - 1)(|R| - 2)$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = -t + 2(k - 1)$. Recall that $t - t_{-2} = 1$, and since we assume that $t_{-2} < 0$ and $t \geq 0$, it must be that $t = 0$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = 2(k - 1) \geq 2k - 3$.

3. $t_{-2} < 0$ and $t < 0$.

That is, $\max(t, 0) = \max(t_{-2}, 0) = 0$, and thus $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (k - 1)|R|$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = 2(k - 1) \geq 2k - 3$.

- $|R| - 2 < \text{gap}$ and $|R| \geq \text{gap}$.

That is, $\text{gap} = |R|$ or $\text{gap} = |R| - 1$. Therefore, $t = |R|$. Recall that $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (k - 2) \max(t, 0) + (k - 1)(|R| - \max(t, 0))$. That is, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (k - 2)|R|$. In addition, according to Algorithm 4, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (\lceil \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rceil)((|R|-2)(k-1)-\text{gap}) \bmod (|R|-2)) + (\lfloor \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rfloor)(|R|-2 - (((|R|-2)(k-1)-\text{gap}) \bmod (|R|-2)))$. Clearly, $(\lceil \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rceil)((|R|-2)(k-1)-\text{gap}) \bmod (|R|-2) + (\lfloor \frac{(|R|-2)(k-1)-\text{gap}}{|R|-2} \rfloor)(|R|-2 - (((|R|-2)(k-1)-\text{gap}) \bmod (|R|-2))) = (|R|-2)(k-1) - \text{gap}$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (|R|-2)(k-1) - \text{gap}$.

Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = (k-2)|R| - (|R|-2)(k-1) + \text{gap} = -|R| + 2k - 2 + \text{gap}$. Since $\text{gap} = |R|$ or $\text{gap} = |R| - 1$, then $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) \geq 2k - 3$.

- $|R| - 2 < \text{gap}$ and $|R| < \text{gap}$.

According to Algorithm 4, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (\lceil \frac{(|R|)(k-1)-\text{gap}}{|R|} \rceil)((|R|(k-1) - \text{gap}) \bmod |R|) + (\lfloor \frac{(|R|)(k-1)-\text{gap}}{|R|} \rfloor)(|R| - (|R|(k-1) - \text{gap}) \bmod |R|)$. Clearly, $(\lceil \frac{(|R|)(k-1)-\text{gap}}{|R|} \rceil)((|R|(k-1) - \text{gap}) \bmod |R|) + (\lfloor \frac{(|R|)(k-1)-\text{gap}}{|R|} \rfloor)(|R| - (|R|(k-1) - \text{gap}) \bmod |R|) = (|R|)(k-1) - \text{gap}$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) = s(m_{nd}, \mathcal{L}) + (|R|)(k-1) - \text{gap}$.

In addition, according to Algorithm 4, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (\lceil \frac{(|R|-2)(k-1)-gap}{|R|-2} \rceil)((|R|-2)(k-1)-gap) \bmod (|R|-2) + (\lfloor \frac{(|R|-2)(k-1)-gap}{|R|-2} \rfloor)(|R|-2 - (((|R|-2)(k-1)-gap) \bmod (|R|-2)))$. Clearly, $(\lceil \frac{(|R|-2)(k-1)-gap}{|R|-2} \rceil)((|R|-2)(k-1)-gap) \bmod (|R|-2) + (\lfloor \frac{(|R|-2)(k-1)-gap}{|R|-2} \rfloor)(|R|-2 - (((|R|-2)(k-1)-gap) \bmod (|R|-2))) = (|R|-2)(k-1) - gap$. Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (|R|-2)(k-1) - gap$.

Therefore, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) = s(m_{nd}, \mathcal{L}) + (|R|)(k-1) - gap - s(m_{nd}, \mathcal{L}) - (|R|-2)(k-1) + gap = 2k - 2 > 2k - 3$

Overall, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) - s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) \geq 2k - 3$. \square

Now we can prove the theorem.

Proof of Theorem 7. Clearly, if Algorithm 4 returns a preference profile \mathcal{L}_R , then it is a successful manipulation that will make m^* the match of \hat{w} . Suppose that a preference profile \mathcal{L}_T exists that makes m^* the match of \hat{w} with $|R| - 2$ manipulators. Let $m'_{nd} \in o(\hat{w}, \mathcal{L}_T)$ be the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}_T)$. We divide into two cases:

1. $B^{nd} = \emptyset$. According to Corollary 3, in the iteration in which $m_{nd} = m'_{nd}$, the algorithm successfully finishes phase 1. Since $B^{nd} = \emptyset$, then according to Corollary 2, in the iteration in which $m_{nd} = m'_{nd}$, the algorithm successfully finishes phase 2. That is, the algorithm will find a preference profile that will make w^* the match of \hat{m} with $|R| - 2$ manipulators.
2. $B^{nd} \neq \emptyset$. Clearly, a preference profile \mathcal{L}'_T exists that makes m^* the match of \hat{w} with $|R|$ manipulators, in which $m'_{nd} \in o(\hat{w}, \mathcal{L}'_T)$ and it is the second most preferred man according to $\mathcal{F}(\mathcal{L} \cup \mathcal{L}'_T)$. According to Corollary 3, in the iteration in which $m_{nd} = m'_{nd}$, the algorithm successfully finishes phase 1. Clearly, when there are $|R|$ manipulators, in the iteration in which $m_{nd} = m'_{nd}$, the algorithm will still successfully finish phase 1. According to Lemma 17, $\max_{d \in D^{nd}} \{s(d, \mathcal{L}, \mathcal{L}_R)\} \leq q(D^{nd}) + \frac{3|D^{nd}|}{2} + 1$. According to Lemma 12, $q(D^{nd}) + \frac{3|D^{nd}|}{2} + 1 \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) + \frac{3|D^{nd}|}{2} + 1$. Since $|D^{nd}| \leq k - 2$, then $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) + \frac{3|D^{nd}|}{2} + 1 \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) + \frac{3k}{2} - 2$. According to Lemma 18, $s(m_{nd}, \mathcal{L}, \mathcal{L}_{R-2}) + \frac{3k}{2} - 2 \leq s(m_{nd}, \mathcal{L}, \mathcal{L}_R) + 3 - 2k + \frac{3k}{2} - 2$. Finally, since $B^{nd} \neq \emptyset$, then $D^{nd} \neq \emptyset$, therefore $k \geq 3$. Thus, $s(m_{nd}, \mathcal{L}, \mathcal{L}_R) + 3 - 2k + \frac{3k}{2} - 2 < s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$. Overall, $\max_{d \in D^{nd}} \{s(d, \mathcal{L}, \mathcal{L}_R)\} < s(m_{nd}, \mathcal{L}, \mathcal{L}_R)$, and according to Lemma 13 the algorithm will find a preference profile that will make w^* the match of \hat{m} with $|R|$ manipulators.

\square