# Chapter 5

# Manipulating Gale-Shapley Algorithm

## 5.1 Introduction

The theory of two-sided matching (Roth and Sotomayor, 1992) has been a cornerstone of market design, inspiring a wide array of applications such as school choice (Abdulkadiroğlu et al., 2005a,b), entry-level labor markets (Roth and Peranson, 1999), and kidney exchange (Roth et al., 2004). All of these markets are built around a centralized matching procedure, which solicits the preferences of the participating agents (typically in the form of rank ordered lists) and selects a matching outcome based on these preferences (i.e., decides who gets matched to whom).

Arguably the most famous of such procedures is the Gale-Shapley algorithm (Gale and Shapley, 1962), which takes as input the preferences of two sets of agents over each other (commonly referred to as *men* and *women*), and after a sequence of proposal and rejection steps, outputs a matching that is *stable* in the sense that no pair of agents prefer each other over their assigned partners. Over the years, the notion of *stability* has emerged as the most significant predictor of a market's long term success—while markets with stable matching procedures have successfully persisted, the unstable ones have failed and are out of use (Roth, 2002).

A frequent concern in such markets, however, is that of strategic behavior by *individual* agents. Indeed, Roth (1982) has shown that no stable matching mechanism makes it a dominant strategy for every agent to announce its true preferences. This, in particular, means that the Gale-Shapley algorithm is also vulnerable to manipulation.

Our interest in this chapter is to study the conditions under which the Gale-Shapley matching for the manipulated instance remains stable with respect to the true preferences. It is known from the results of Dubins and Freedman (1981) and Roth (1982) that the proposing side (say, the men) in the algorithm has no incentive to manipulate. Therefore, any strategic behavior must occur on the proposed-to side (i.e., the women). Our starting point, therefore, is the following natural question:

> Suppose a woman can manipulate by permuting her true preference list while everyone else is truthful. Then, is the resulting Gale-Shapley matching stable with respect to the true preferences?

Our first result (Theorem 23) shows that the answer to the above question is YES if the manipulation is *optimal*. That is, if, by misreporting, the manipulator secures the best possible partner (according to her true preferences), then the resulting Gale-Shapley matching is stable with respect to the true preferences of the agents. This is an encouraging result for both the market designer (who cares about stability with respect to the true preferences) and the manipulator (who cares about finding the optimal partner). We complement this result by showing that sub-optimal manipulations can sometimes lead to instability (Example 5).

Besides optimality, the manipulator might also want to avoid being suspected of strategic behavior. Indeed, it is reasonable to expect that the market designer has a coarse idea of the true preferences of the agents, say from past runs of the algorithm or from survey data. In such a setting, strategic misreporting can be easily detected if the announced list of an agent looks significantly different from the market designer's estimate. Our second result (Theorem 24) shows that this is an avoidable concern for the manipulator: For *any* optimal manipulation, there exists an equivalent inconspicuous manipulation which results in the *same* matched partner, and can be derived from the true list by promoting only one man (and making no other changes). In other words, an optimal manipulation is *nearly indistinguishable* from the true list.

Our third result (Theorem 25) strengthens the stability implication of Theorem 23 for inconspicuous optimal manipulations. Specifically, we show that *every* matching (including the Gale-Shapley matching) that is stable with respect to the manipulated instance is also stable with respect to the true preferences. We complement this result by giving an example of a *non*-inconspicuous optimal manipulation, where a matching that is stable with respect to the manipulated instance is not stable with respect to the true preferences (Example 6).

*Why study permutation manipulations?* By far, the most commonly studied model of manipulation in the stable matching literature has been that of *truncation* (Gale and Sotomayor, 1985; Roth and Rothblum, 1999; Coles and Shorrer, 2014; Jaramillo et al., 2014). This model allows the manipulator to remove a tail of her true preference list and report the rest of the list unshuffled. An especially attractive feature of truncation-based manipulations is *exhaustiveness*, which means that any possible misrepresentation can be replicated or improved by a truncation strategy (Roth and Vate, 1991; Jaramillo et al., 2014). However, an optimal truncation manipulation by an agent can look *very* different from her true preference list. In fact, Coles and Shorrer (2014) have shown that when the true preferences of all the other agents are generated uniformly at random, the optimal truncation strategy for the manipulator involves the removal of a *large* fraction of her true list. This phenomenon is particularly severe for large markets, where the ratio of the length of the reported list to that of the true list goes to zero as the size of the market grows. By contrast, as our results show, an optimal permutation manipulation can be nearly identical to the true list of the manipulator. We consider this feature of permutation manipulations to be an important one, both from a theoretical and a practical standpoint.

## 5.2 Related Work

The literature on permutation manipulations of Gale-Shapley algorithm has focused primarily on computational questions (Teo et al., 2001; Kobayashi and Matsui, 2009, 2010; Deng et al., 2015; Gupta et al., 2016). The earliest result in this direction is by Teo et al. (2001), who gave an $\mathcal{O}(n^3)$ algorithm for computing an optimal permutation manipulation of Gale-Shapley algorithm by a single manipulator. In conjunction with our result (Theorem 23), this means that not only are optimal manipulations stability-preserving, but are also efficiently computable.

The work of Teo et al. (2001) was generalized to the coalitional manipulation setting by Deng et al. (2015). For permutation manipulations, they provide an $\mathcal{O}(n^6)$ algorithm for computing a pareto-optimal manipulation for any fixed coalition. Additionally, they show that any such manipulation can be made inconspicuous. There are two important ways in which the work of Deng et al. (2015) differs from ours: First, their model only allows for manipulations that are stability-preserving in the first place. On the other hand, we allow the manipulator to pick *any* permutation (not just a stability-preserving one), and identify a class of manipulations that induce stability. Second, their result on the inconspicuousness of coalitional manipulation involves a reduction to the stable roommates problem (Irving, 1985), and uses sophisticated combinatorial structures like rotations and suitor graphs (Kobayashi and Matsui, 2010). Our proof, on the other hand, is direct and much simpler.

Another set of relevant works is by Kobayashi and Matsui (2009, 2010), who study the following extension version of the manipulation problem: Suppose we are given the preference lists of all men, and a complete/partial matching. The goal is to compute a set (if it exists) of the preference lists of all women such that the men-proposing Gale-Shapley matching contains the given matching. They show that this problem is NP-complete in general (when the given matching is partial), but polynomial time solvable if the given matching is complete.

Finally, the recent work of Gupta et al. (2016) studies the question of *total stability* of a manipulation strategy when all women are strategic individuals and all men are truthful. A manipulation strategy is totally stable if (i) it is stability-preserving, and (ii) it constitutes a stable Nash equilibrium, meaning that any unilateral deviation from the strategy is either non-improving or induces instability. It is shown that checking a given strategy for total stability can be done in polynomial time.

## 5.3 Preliminaries

**Problem setup** An instance $\langle M, W, \succ\rangle$ of the *stable marriage problem* consists of a set $M$ of $n$ men, a set $W$ of $n$ women, and a *preference profile* $\succ = \{\succ_{m_1}, \dots, \succ_{m_n}, \succ_{w_1}, \dots, \succ_{w_n}\}$ consisting of the *preference lists* of all men and women. The preference list of each man $m \in M$, denoted by $\succ_m$, is a strict total order over the set of all women (the preference lists of women are defined analogously). We will use the shorthand $w_1 \succeq_m w_2$ to denote 'either $w_1 \succ_m w_2$ or $w_1 = w_2$'. We let $\succ_{-w}$ denote the preference lists of all men and women except woman $w$. Thus, $\succ = \{\succ_{-w}, \succ_w\}$.

**Stable matchings** A *matching* refers to a function $\mu : M \cup W \rightarrow M \cup W$, where $\mu(m) \in W$ 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 matching $\mu$ admits a *blocking pair* with respect to $\succ$ if there is a pair of agents $(m, w)$ such that $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$. A matching $\mu$ is *stable* if it admits no blocking pair with respect to $\succ$. We let $S_\succ$ denote the set of all matchings that are stable with respect to $\succ$.

**Gale-Shapley algorithm** A *matching algorithm* takes as input a preference profile and outputs a matching. In this chapter, we study the well-known *deferred acceptance* algorithm (Gale and Shapley, 1962), also known as *Gale-Shapley algorithm*. In particular, we will focus on the *men-proposing* version of this algorithm, abbreviated as GS algorithm. We let $\mu = \text{GS}(\succ)$ denote the matching output by the men-proposing Gale-Shapley algorithm for the input profile $\succ$.

Briefly, GS algorithm proceeds in rounds, and each round consists of two phases: (i) a *proposal phase*, where each single man proposes to his favorite woman from among those who haven't rejected him yet, and (ii) a *rejection phase*, where each woman with multiple proposals in hand rejects all proposals except the one that she likes best. The algorithm terminates when no single agents remain.

Gale and Shapley (1962) showed that GS algorithm always terminates with a stable matching. Moreover, this matching is *simultaneously* the best for all men from among all stable matchings (and, as McVitie and Wilson (1971) later observed, the worst for all women). Theorem 21 recalls these results.

**Theorem 21** (Men-optimality of GS). *Let $\succ$ be a preference profile, and let $\mu = \text{GS}(\succ)$. Then, $\mu \in S_\succ$. Moreover, for any $\mu' \in S_\succ$, $\mu(m) \succeq_m \mu'(m)$ for all $m \in M$ and $\mu'(w) \succeq_w \mu(w)$ for all $w \in W$.*

We let $\text{Prop}(w, \succ)$ denote the set of all men who propose to $w$ during the run of GS algorithm on $\succ$. Further, we let $\text{Prop}(w, \succ, i)$ denote the $i^{\text{th}}$ favorite man of $w$ (according to $\succ_w$) in the set $\text{Prop}(w, \succ)$. Thus, $\text{Prop}(w, \succ, 1) = \mu(w)$ for $\mu = \text{GS}(\succ)$.

**Stable lattice** Given a preference profile $\succ$ and two matchings $\mu$ and $\mu'$, define the *join* function $\mu_\vee = \mu \vee \mu'$ as follows: for each $m \in M$ and $w \in W$,

$$\mu_\vee(m) = \begin{cases} \mu(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu'(m) & \text{otherwise,} \end{cases} \quad \text{and} \quad \mu_\vee(w) = \begin{cases} \mu'(w) & \text{if } \mu(w) \succ_w \mu'(w) \\ \mu(w) & \text{otherwise.} \end{cases}$$

Similarly, define the *meet* function $\mu_\wedge = \mu \wedge \mu'$ for all $m \in M$ and $w \in W$ as:

$$\mu_\wedge(m) = \begin{cases} \mu'(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu(m) & \text{otherwise,} \end{cases} \quad \text{and} \quad \mu_\wedge(w) = \begin{cases} \mu(w) & \text{if } \mu(w) \succ_w \mu'(w) \\ \mu'(w) & \text{otherwise.} \end{cases}$$

The following result from (Knuth, 1997), attributed to John Conway, shows that the *join* and *meet* of any pair of stable matchings are also stable.

**Theorem 22** (Lattice of stable matchings)**.** *Let $\succ$ be a preference profile and let $\mu, \mu' \in S_\succ$. Then, $\mu_\vee, \mu_\wedge \in S_\succ$.*

**Manipulation** A matching algorithm is said to be *manipulable by an agent* $w$ if there exists a pair of preference profiles $\succ$ and $\succ'$, differing only in the preferences of $w$, such that $\mu'(w) \succ_w \mu(w)$, where $\mu$ and $\mu'$ are the matchings before and after the manipulation respectively. The agent $w$ is referred to as the *manipulator*. In this chapter, we focus on *permutation manipulations*, i.e., $\succ'_w$ is a permutation of $\succ_w$. Besides, we focus only on the manipulation of GS algorithm, and the manipulator is assumed to be a woman, who knows the preferences of all the other agents.

We will often use $\succ' = (\succ_{-w}, \succ'_w)$ and $\mu' = \text{GS}(\succ')$ to denote the manipulated profile (with respect to $\succ$) and the resulting GS matching respectively. The matching $\mu'$ and the set $S_{\succ'}$ will be referred to as the *induced matching* and the *induced lattice* respectively. The set $S_\succ$ will be referred to as the *original lattice*. We will say that a manipulation $\succ'_w$ with respect to $\succ$ is *stability-preserving* if $\mu' \in S_\succ$.

**Optimal manipulation** Given a preference profile $\succ$, an *optimal manipulation* of GS algorithm by an agent $w$ with respect to a profile $\succ$ refers to a preference list $\succ'_w$ such that (i) $\mu'(w) \succ_w \mu(w)$, and (ii) $\mu'(w) \succeq_w \mu''(w)$ for any other preference list $\succ''_w$, where $\mu' = \text{GS}(\succ_{-w}, \succ'_w)$ and $\mu'' = \text{GS}(\succ_{-w}, \succ''_w)$.

**Inconspicuous equivalent of a manipulation** Given a preference profile $\succ$, we call $\succ''_w$ an *inconspicuous equivalent* of a manipulation $\succ'_w$ (of GS algorithm by woman $w$) if (i) $\succ''_w$ can be derived from the true preference list $\succ_w$ by moving at most one man, and (ii) $\mu''(w) = \mu'(w)$, where $\mu' = \mathrm{GS}(\succ')$ and $\mu'' = \mathrm{GS}(\succ'')$.

## 5.4 Main Results

Our first result (Theorem 23) shows that optimal manipulation of GS algorithm by a single woman preserves stability.[1]

**Theorem 23** (Optimal manipulation is stability-preserving)**.** *Let $\succ'_w$ be an optimal manipulation with respect to $\succ$ for woman $w$ and let $\mu' = \mathrm{GS}(\succ')$. Then, $\mu' \in S_{\succ}$.*

This is a positive result, since the stability of the resulting matching is not affected by optimal strategic behaviour of a single agent. Example 5 complements this result by showing that sub-optimal strategic behavior can lead to instability.

**Example 5.** *(**Sub-optimal manipulation can be unstable**) Consider the stable marriage instance shown below.*

| True preferences of men | True preferences of women |
|---|---|
| $m_1$: 2 1 3 4 | $w_1$: 1 2 $\underline{3}$ 4 |
| $m_2$: 3 1 2 4 | $w_2$: 3 $\underline{1}$ 2 4 |
| $m_3$: 1 2 3 4 | $w_3$: 1 $\underline{2}$ 3 4 |
| $m_4$: 1 4 2 3 | $w_4$: 1 2 3 $\underline{4}$ |
| $w_1$ manipulates sub-optimally ($\succ'$) | $w_1$ manipulates optimally ($\succ''$) |
| $w_1$: $\underline{2}$ 4 3 1 | $w_1$: $\underline{1}$ 4 3 2 |
| $w_2$: $\underline{3}$ 1 2 4 | $w_2$: $\underline{3}$ 1 2 4 |
| $w_3$: $\underline{1}$ 2 3 4 | $w_3$: 1 $\underline{2}$ 3 4 |
| $w_4$: 1 2 3 $\underline{4}$ | $w_4$: 1 2 3 $\underline{4}$ |

*The top row shows the true preferences of the agents, and the bottom row shows two different manipulations by $w_1$ with respect to $\succ$. The underlined numbers denote the matched partners of the women under the men-proposing Gale-Shapley algorithm.*

[1] After the publication of our paper (Vaish and Garg, 2017), it was pointed out to us by Yannai Gonczarowski that a similar result was stated in (Gonczarowski and Friedgut, 2013).

*Notice that the manipulation under $\succ'$ is sub-optimal for $w_1$ because she matches to a strictly better partner (according to her true preference list $\succ_{w_1}$) under $\succ''$. Besides, the matching induced under $\succ'$ is not stable with respect to the true preferences $\succ$ since $(m_1, w_1)$ is a blocking pair. On the other hand, the manipulation under $\succ''$ is optimal, and the induced matching is stable with respect to the true preferences.*

As mentioned earlier in Section 5.1, it is still possible that the list reported by the manipulator (as part of optimal manipulation) might look very different from her true list. Indeed, for the stable marriage instance in Example 5, notice that except for the most preferred man (namely $m_1$), the optimal manipulation $\succ''_{w_1}$ by $w_1$ is a complete reversal of her true preference list $\succ_w$. Theorem 24 addresses this concern by showing that any optimal manipulation has an inconspicuous equivalent which results in the same matched partner for the manipulator (thus maintaining optimality) while differing from the true list in the positioning of only one man (thus making the manipulation hard to be detected).

**Theorem 24** (Inconspicuous equivalent of optimal manipulation). *Let $\succ'_w$ be an optimal manipulation with respect to $\succ$ for woman $w$. Let $\succ''_w$ be another preference list derived from her true list $\succ_w$ by promoting the man $\text{Prop}(w, \succ', 2)$ to the position right after $\text{Prop}(w, \succ', 1)$ while making no other changes. Then, $\succ''_w$ is the inconspicuous equivalent of $\succ'_w$, i.e., $\mu''(w) = \mu'(w)$, where $\mu' = \text{GS}(\succ')$ and $\mu'' = \text{GS}(\succ'')$.*

Going back to Example 5, it can be observed that $w_1$ can achieve her optimal partner (i.e., $m_1$) by reporting the list $m_1 \succ m_4 \succ m_2 \succ m_3$, derived from her true list $\succ_w$ by promoting the man $m_4$ and making no other changes.

Our final result strengthens the stability implication of Theorem 23 for inconspicuous optimal manipulations. Recall from Theorem 23 that the men-proposing Gale-Shapley matching for the optimally manipulated instance is stable with respect to the true preferences. This result can be alternatively interpreted in the lattice terminology as follows: Theorem 23 shows that the "men-optimal extreme" of the lattice $S_{\succ'}$ always lies inside the original lattice $S_{\succ}$. One might wonder whether the same holds for the rest of the manipulation lattice as well. Theorem 25 shows that this is indeed the case for inconspicuous optimal manipulations.

**Theorem 25** (Lattice containment result). *Let $\succ'_w$ be an optimal manipulation with respect to $\succ$ for woman $w$, and let $\succ''_w$ be its inconspicuous equivalent (as described in Theorem 24). Then, $S_{\succ''} \subseteq S_{\succ}$.*

Example 6 complements Theorem 25 by showing that the implication does not extend to non-inconspicuous optimal manipulations.

| True preferences of men | True preferences of women |
|---|---|
| $m_1$: 2 1 3 4 5 | $w_1$: 1 2 3 $\underline{4}$ 5 |
| $m_2$: 4 1 5 3 2 | $w_2$: 3 2 4 5 $\underline{1}$ |
| $m_3$: 3 1 4 5 2 | $w_3$: 4 2 1 5 $\underline{3}$ |
| $m_4$: 1 3 4 5 2 | $w_4$: 5 1 4 3 $\underline{2}$ |
| $m_5$: 1 5 4 3 2 | $w_5$: 1 2 3 4 $\underline{5}$ |
| Opt + Non-Inconsp manipulation ($\succ'$) | Opt + Inconsp manipulation ($\succ''$) |
| $w_1$: $\underline{2}$ 5 4 3 1 | $w_1$: 1 $\underline{2}$ 5 3 4 |
| $w_2$: 3 2 4 5 $\underline{1}$ | $w_2$: 3 2 4 5 $\underline{1}$ |
| $w_3$: $\underline{4}$ 2 1 5 3 | $w_3$: $\underline{4}$ 2 1 5 3 |
| $w_4$: 5 1 4 $\underline{3}$ 2 | $w_4$: 5 1 4 $\underline{3}$ 2 |
| $w_5$: 1 2 3 4 $\underline{5}$ | $w_5$: 1 2 3 4 $\underline{5}$ |

**Example 6.** *Consider the following stable marriage instance*

*The top row shows the true preferences of the agents, while the bottom row shows two different optimal manipulations $\succ'$ and $\succ''$ for the manipulator $w_1$. The underlined numbers denote the matched partners of the women under the men-proposing Gale-Shapley algorithm.*

*First, we briefly describe why $\succ'$ and $\succ''$ are both optimal. Observe that during the run of GS algorithm on the true profile $\succ$, $w_1$ receives proposals only from $m_4$ and $m_5$. Therefore, any manipulation by $w_1$ must involve swapping the relative ordering of $m_4$ and $m_5$. By rejecting $m_4$ in favor of $m_5$, $w_1$ forces $m_4$ to propose to $w_3$, thereby displacing $m_3$. As a result, $m_3$ is forced to propose to $w_1$. This already is an improvement for $w_1$ over her original partner $m_4$. However, $w_1$ can do even better by pretending to prefer $m_5$ over $m_3$ in the manipulated list. This forces $m_3$ to propose to $w_4$, thereby displacing $m_2$, who in turn is forced to propose to $w_1$, giving her a more preferred match. Notice that $w_1$ can do no better, since rejecting $m_2$ in favor of $m_5$ will not displace $m_1$ in the preference list of $w_2$. Thus, $m_2$ is the optimal partner for $w_1$.*

*Let us now consider the matching $\phi$ defined as $\phi = (m_1, w_5), (m_2, w_1), (m_3, w_2), (m_4, w_3), (m_5, w_4)$. It is easy to verify that $\phi \in S_{\succ'}$ but $\phi \notin S_{\succ''}$ and $\phi \notin S_{\succ}$ since $(m_1, w_1)$ constitutes a blocking pair in each case. Therefore, the stable lattice induced by an optimal manipulation (in this case $\succ'$) need not be completely contained inside the original lattice.* □

## 5.5 Proofs of Main Results

### 5.5.1 Proof of Stability of Optimal Manipulation (Theorem 23)

We start with a lemma (Lemma 29) that will be useful in the proof of Theorem 23. The lemma states that if a fixed man $m$ proposes to a fixed woman $w$ during the run of GS algorithm, then the same holds when $w$ moves that man up/down in her list while making no other changes. Notice that the lemma applies only to the fixed man $m$ who was moved in the preference list of $w$, and not to any other man who proposed to $w$ but was not moved.

**Lemma 29.** *Let $\succ$ and $\succ'$ be two preference profiles that are completely identical except for the preferences of a fixed woman $w$. Let $\succ'_w$ be derived from $\succ_w$ by moving a fixed man $m$ and making no other changes. Then, $m \in \text{Prop}(w, \succ) \Rightarrow m \in \text{Prop}(w, \succ')$.*

*Proof.* Suppose, for contradiction, that $m \in \text{Prop}(w, \succ) \setminus \text{Prop}(w, \succ')$. Let $R$ and $R'$ denote the runs of GS algorithm on $\succ$ and $\succ'$ respectively, and let $\mu = \text{GS}(\succ)$ and $\mu' = \text{GS}(\succ')$. Since $m$ proposes to $w$ during $R$ but not during $R'$, it must be that he is rejected by $\mu'(m)$ during $R$. In other words, the matchings $\mu$ and $\mu'$ must be non-identical.

Since $\mu \neq \mu'$, the runs $R$ and $R'$ must differ at some round if we run the GS algorithm in parallel on the two profiles. Let $t$ be the *earliest* round at which this happens. The following observations will help us identify the point of departure of the runs $R$ and $R'$.

1. *The proposal phase in round 1 is identical for $R$ and $R'$.*
   *Reason*: Proposals made in the first round of GS algorithm depend only on the preferences of the men, which are identical between $\succ$ and $\succ'$.

2. *If $R$ and $R'$ have an identical rejection phase for each round until (and including) round $i$, then the proposal phases for the two runs in round $(i+1)$ are also identical.*
   *Reason*: Since the preferences of the men are identical between the two profiles, the proposals made at round $(i+1)$ depend only on the rejections made until (and including) round $i$, which are the same for the two runs.

Therefore, for $R$ and $R'$ to differ for the first time at round $t$, all proposal and rejection phases until (and including) the proposal phase at round $t$ must be identical between the two runs, while the rejection phases at round $t$ must be different. We already know that the two instances differ only in the positioning of $m$ in the preference list of $w$. Therefore, for the rejection phases at round $t$ to differ, $w$ must have in hand a proposal from $m$ in at least one of the runs $R$ and $R'$. This means that $m$ must have proposed to $w$ at (or before) round $t$ of that run. Since the proposal phases for $R$ and

$R'$ are identical upto that point, it must be that $m$ proposes to $w$ during both $R$ and $R'$, which is a contradiction. □

We will now provide the proof of Theorem 23.

**Theorem 23** (Optimal manipulation is stability-preserving). *Let $\succ'_w$ be an optimal manipulation with respect to $\succ$ for woman $w$ and let $\mu' = \text{GS}(\succ')$. Then, $\mu' \in S_{\succ}$.*

*Proof.* Suppose, for contradiction, that $\mu' \notin S_{\succ}$. Then, there must exist a pair $(m', w')$ that blocks $\mu'$ with respect to $\succ$. That is, $w' \succ_{m'} \mu'(m')$ and $m' \succ_{w'} \mu'(w')$.

We first claim that $w' = w$. Indeed, by the stability of GS algorithm, we have $\mu' \in S_{\succ'}$. Hence, either $\mu'(m') \succ'_{m'} w'$ or $\mu'(w') \succ'_{w'} m'$ or both. If $w' \neq w$, then by construction $\succ'_{w'} = \succ_{w'}$. Similarly, $\succ'_{m'} = \succ_{m'}$. Therefore, we have that either $\mu'(m') \succ_{m'} w'$ or $\mu'(w') \succ_{w'} m'$ or both, which contradicts the blocking pair condition. Hence, $w' = w$.

Next, consider a preference list $\succ''_w$ for $w$ derived from $\succ'_w$ by *promoting* the man $m'$ to the top position and making no other changes. Let $\succ'' = (\succ_{-w}, \succ''_w)$ and $\mu'' = \text{GS}(\succ'')$. Since $w' = w$, we have that $m' \succ_w \mu'(w)$ and $w \succ_{m'} \mu'(m')$. Moreover, since $\succ'_{m'} = \succ_{m'}$, we have that $w \succ'_{m'} \mu'(m')$. Therefore, during the run of GS algorithm on $\succ'$, it must be that $m'$ proposes to and is rejected by $w$ before he is matched to $\mu'(m')$. That is, $m' \in \text{Prop}(w, \succ')$. By applying Lemma 29 with $\succ'$ and $\succ''$ as the *old* and the *new* profiles respectively, we get that $m' \in \text{Prop}(w, \succ'')$. Since $m'$ is at the top position in $\succ''_w$, we must have $\mu''(w) = m'$. This, however, contradicts the optimality of $\succ'_w$, since $m' \succ_w \mu'(w)$ according to the blocking pair condition. □

### 5.5.2 Proof of Inconspicuous Equivalent of Optimal Manipulation (Theorem 24)

Our proof of Theorem 24 crucially relies on the *swapping lemma* (Lemma 30). This lemma provides a complete description of the matched partners that the manipulator can secure by swapping any pair of adjacent men in her preference list. We state the lemma below and defer its proof to Section 5.5.3. It will be convenient to denote the set of non-proposing men by Non-Prop$(w, \succ)$. Thus, Non-Prop$(w, \succ) = M \setminus \text{Prop}(w, \succ)$.

**Lemma 30** (Swapping lemma). *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \text{GS}(\succ)$ and $\mu' = \text{GS}(\succ')$. Let $\succ'_w$ be 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 $m_i \in$ Non-Prop$(w, \succ)$ or $m_j \in$ Non-Prop$(w, \succ)$, then $\mu'(w) = \mu(w)$.*

*(2) if* $m_i, m_j \notin \{\mathrm{Prop}(w, \succ, 1), \mathrm{Prop}(w, \succ, 2)\}$*, then* $\mu'(w) = \mu(w)$*.*

*(3) if* $m_i = \mathrm{Prop}(w, \succ, 2)$ *and* $m_j = \mathrm{Prop}(w, \succ, 3)$*, then* $\mu'(w) \in \{\mu(w), m_j\}$*.*

*(4) if* $m_i = \mathrm{Prop}(w, \succ, 1)$ *and* $m_j = \mathrm{Prop}(w, \succ, 2)$*, then* $\mathrm{Prop}(w, \succ', 2) \in \{m_i, m_j\}$*.*

In words, case (1) shows that swapping two non-proposers or a proposer and a non-proposer cannot give a different matched partner. Case (2) shows the same result for swapping a pair of proposers outside the first and the second best. Case (3) shows that the matched partner either stays the same or becomes worse (according to the true preference list) if the second and third-best proposers are swapped. Finally, case (4) shows that swapping the first and the second-best proposers can lead to at most one new proposal that is better than the old partner (according to the true preference list).

**Theorem 24** (Inconspicuous equivalent of optimal manipulation). *Let* $\succ'_w$ *be an optimal manipulation with respect to* $\succ$ *for woman* $w$*. Let* $\succ''_w$ *be another preference list derived from her true list* $\succ_w$ *by promoting the man* $\mathrm{Prop}(w, \succ', 2)$ *to the position right after* $\mathrm{Prop}(w, \succ', 1)$ *while making no other changes. Then,* $\succ''_w$ *is the inconspicuous equivalent of* $\succ'_w$*, i.e.,* $\mu''(w) = \mu'(w)$*, where* $\mu' = \mathrm{GS}(\succ')$ *and* $\mu'' = \mathrm{GS}(\succ'')$*.*

*Proof.* We construct the preference list $\succ''_w$ by starting from $\succ'_w$ and performing a sequence of *massaging* operations on it in order to make it resemble the original preference list $\succ_w$, except for the placement of the man $\mathrm{Prop}(w, \succ', 2)$. Each such operation involves swapping a pair of adjacent men in the current list. By repeatedly invoking swapping lemma (Lemma 30), we will argue that $w$ continues to receive a proposal from $\mu'(w)$ at each intermediate step, giving us the desired result. We start by describing the construction of the list $\succ''_w$ followed by arguing the correctness of this construction. For notational convenience, we will use $p = \mathrm{Prop}(w, \succ', 1)$ and $q = \mathrm{Prop}(w, \succ', 2)$.

(1) *Constructing* $\succ''_w$: Starting from $\succ'_w$, we construct a sequence of preference lists $\succ^{(1)}_w, \succ^{(2)}_w, \ldots$ culminating in $\succ''_w$ as follows (refer Figure 5.1):

   (a) *Promoting* $q$: The list $\succ^{(1)}_w$ is obtained from $\succ'_w$ by promoting $q$ to the position right after $p$. Since any men between $p$ and $q$ in the list $\succ'_w$ can only be non-proposers, it follows from case (1) of swapping lemma (Lemma 30) that the run of the algorithm on the profile $\succ^{(1)} = (\succ_{-w}, \succ^{(1)}_w)$ is identical to that on $\succ'$. Thus, $\mathrm{Prop}(w, \succ^{(1)}, 1) = p$ and $\mathrm{Prop}(w, \succ^{(1)}, 2) = q$. Notice that any man above $p$ in the list $\succ^{(1)}_w$ must be a non-proposer.

   (b) *Fixing the part of the list above* $p$: Our goal in this step will be to make the part of the list $\succ^{(1)}_w$ above and including $p$ resemble that in the list $\succ_w$. We achieve this by replacing the

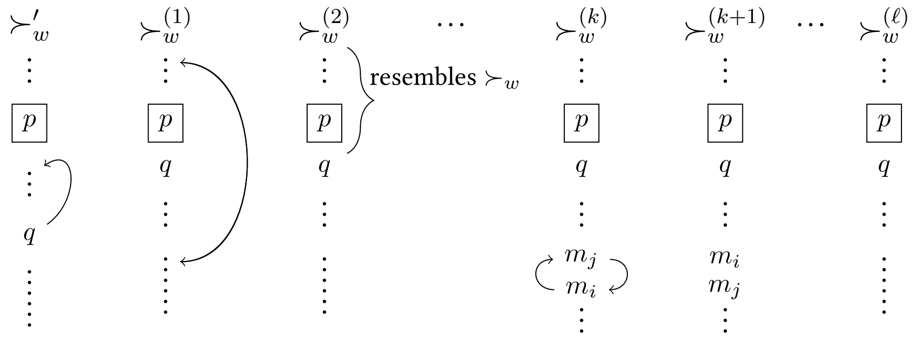

**Figure 5.1:** *The sequence of preference lists constructed in the proof of Theorem 24. Here,* $p = \text{Prop}(w, \succ', 1) = \mu'(w)$ *and* $q = \text{Prop}(w, \succ', 2)$.

set of non-proposers above $p$ in $\succ^{(1)}_w$ with the set of men above $p$ in the true list $\succ_w$. It is easy to see that no man in the latter set can be a proposer to $w$ at any stage, or else the optimality of the manipulation is violated. Thus, our task involves shuffling around a set of non-proposers in the list $\succ^{(1)}_w$, which, by case (1) of swapping lemma (Lemma 30), does not affect the run of GS algorithm. We call the resulting preference list $\succ^{(2)}_w$.

It is easy to verify that after this step, the new list $\succ^{(2)}_w$ resembles the true preference list $\succ_w$ for all positions above and including $p$, and that $\text{Prop}(w, \succ^{(2)}, 1) = p$ and $\text{Prop}(w, \succ^{(2)}, 2) = q$, where $\succ^{(2)} = (\succ_{-w}, \succ^{(2)}_w)$.

(c) *Fixing the part of the list below* $q$: The final step in our construction involves a sequence of preference lists $\{\succ^{(3)}_w, \succ^{(4)}_w, \dots\}$. The list $\succ^{(k+1)}_w$ is derived from $\succ^{(k)}_w$ by *swapping* a pair of adjacent men $(m_i, m_j)$ in $\succ^{(k)}_w$ such that (i) $\text{Prop}(w, \succ^{(k)}, 2) \succ^{(k)}_w m_i$ and $\text{Prop}(w, \succ^{(k)}, 2) \succ^{(k)}_w m_j$, and (ii) $m_i \succ_w m_j$ and $m_j \succ^{(k)}_w m_i$. In words, each new list in the sequence is derived from the previous list by swapping a pair of adjacent men that are (i) both positioned below the second-favorite proposer according to the previous list, and (ii) are incorrectly ordered with respect to the true preference list $\succ_w$. No other changes are made. Notice that this sequence of preference lists must be finite since there can only be a finite number of pairs of men that are incorrectly ordered with respect to the true list. Let $\succ^{(\ell)}_w$ be the final list in this sequence, and let $\succ^{(k)} = (\succ_{-w}, \succ^{(k)}_w)$ and $\mu^{(k)} = \text{GS}(\succ^{(k)})$ denote the preference profile and the corresponding GS matching at each step. This finishes the construction of the sequence of preference lists.

(2) *Correctness*: We will now show that Theorem 24 holds for $\succ''_w = \succ^{(\ell)}$. That is, we will show that:

(a) The list $\succ^{(\ell)}_w$ can be obtained from the true list $\succ_w$ by promoting $q$ to the position right after $p$ while making no other changes, and

(b) $\text{Prop}(w, \succ^{(\ell)}, 1) = \mu^{(\ell)}(w) = p$.

Case (a) follows easily from the above construction. Indeed, the steps (1a) and (1b) ensure that $\succ_w^{(\ell)}$ resembles $\succ_w$ for all positions above and including $p$, while step (1c) incrementally corrects for pairs that are out of order with respect to $\succ_w$, thus eventually terminating with a list $\succ_w^{(\ell)}$ that is identical to $\succ_w$ except for the positioning of the man $q$.

We prove case (b) by induction. The base case consists of showing that $\mu^{(3)}(w) = p$. Indeed, let $(m_i, m_j)$ be the pair of adjacent men in $\succ_w^{(2)}$ that are swapped in $\succ_w^{(3)}$. By construction, we know that $m_i, m_j \notin \{\text{Prop}(w, \succ^{(2)}, 1), \text{Prop}(w, \succ^{(2)}, 2)\}$. Therefore, from cases (1) and (2) of swapping lemma (Lemma 30), we have that $\mu^{(3)}(w) = p$.

We now proceed to the induction hypothesis. That is, we now assume that $\mu^{(k)}(w) = p$ for all $3 < k \leq K$, and show that $\mu^{(K+1)}(w) = p$.

As before, let $(m_i, m_j)$ be the pair of adjacent men that are swapped in $\succ_w^{(K)}$ to obtain $\succ_w^{(K+1)}$. By construction, we have that $m_i, m_j \notin \{\text{Prop}(w, \succ^{(K)}, 1), \text{Prop}(w, \succ^{(K)}, 2)\}$. Therefore, from cases (1) and (2) of swapping lemma (Lemma 30), we have that $\mu^{(K+1)}(w) = \mu^{(K)}(w)$. Finally, using the induction hypothesis, we get that $\mu^{(K+1)}(w) = p$.

Hence, by induction, we have that $\mu^{(\ell)}(w) = p$. This finishes the proof of case (b) and, as a result, of Theorem 24. □

### 5.5.3 Proof of Swapping Lemma (Lemma 30)

Recall the statement of Swapping Lemma.

**Lemma 30** (Swapping lemma). *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \text{GS}(\succ)$ and $\mu' = \text{GS}(\succ')$. Let $\succ'_w$ be 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 $m_i \in \text{Non-Prop}(w, \succ)$ or $m_j \in \text{Non-Prop}(w, \succ)$, then $\mu'(w) = \mu(w)$.*

*(2) if $m_i, m_j \notin \{\text{Prop}(w, \succ, 1), \text{Prop}(w, \succ, 2)\}$, then $\mu'(w) = \mu(w)$.*

*(3) if $m_i = \text{Prop}(w, \succ, 2)$ and $m_j = \text{Prop}(w, \succ, 3)$, then $\mu'(w) \in \{\mu(w), m_j\}$.*

*(4) if $m_i = \text{Prop}(w, \succ, 1)$ and $m_j = \text{Prop}(w, \succ, 2)$, then $\text{Prop}(w, \succ', 2) \in \{m_i, m_j\}$.*

*Proof.* Case (1) follows from the contrapositive of Lemma 29, since the list $\succ'_w$ is derived from $\succ_w$ by changing the position of a fixed non-proposer (namely, either $m_i$ or $m_j$), which does not affect the run of the algorithm before and after the change. Cases (2), (3) and (4) follow from Lemmas 33, 34 and 35 respectively, which we state and prove below. We remark that only the cases (1) and (2) are relevant to the proof of Theorem 24; cases (3) and (4) are listed for completeness. □

The remainder of this section will focus on the proofs of Lemmas 33 to 35. Before stating the lemmas, we describe two other intermediate results—Lemmas 31 and 32—that will be useful in proving these results.

We will start with the *precedence lemma* (Lemma 31), which says that a necessary condition for the manipulator to make a previously non-proposing man propose to her in some round of the *new* run is to reject her *old* partner at an earlier round. In other words, a "new proposal" event must be preceded by the "rejection of old partner" event. This result explains why the manipulator must promote one (or more) of the previously non-favorite proposers above her *old* partner in order to secure a better partner.

**Lemma 31** (Precedence Lemma). *Let $\succ$ and $\succ'$ be a pair of preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \mathrm{GS}(\succ)$ and $\mu' = \mathrm{GS}(\succ')$. Let $m \in \mathrm{Prop}(w, \succ') \setminus \mathrm{Prop}(w, \succ)$. Then, during the run of GS algorithm on $\succ'$, the event "$m$ proposes to $w$" must be preceded in time by the event "$\mu(w)$ is rejected by $w$".*

*Proof.* Let $R$ and $R'$ denote the runs of the Gale-Shapley algorithm on the profiles $\succ$ and $\succ'$ respectively. For the ease of exposition, we will refer to $\mu$ and $\mu'$ as the *old* and the *new* matching respectively.

We know that $\succ_m = \succ'_m$ by construction. Therefore, the reason $m$ proposes to $w$ during $R'$ but not during $R$ is because he is rejected by his *old* partner (say $w_2 = \mu(m)$) during $R'$. That is, $w_2 \succ'_m w$, and therefore $w_2 \succ_m w$.

Similarly, $w_2$ rejects $m$ during $R'$ because she receives one or more proposals that she strictly prefers over him. Let $m_2$ denote the *earliest* such proposal received by $w_2$ during $R'$. Thus, $m_2 \succ'_{w_2} m$. Moreover, since $w_2 \neq w$, we also have that $m_2 \succ_{w_2} m$.

We now claim that $m_2$ *does not propose to* $w_2$ *during* $R$. Indeed, if $m_2$ does propose to $w_2$ during $R$, then $w_2$ must eventually be matched under $R$ to a man she weakly prefers over $m_2$, which contradicts the above condition that $m_2 \succ_{w_2} m$ (since $\mu(w_2) = m$).

Thus, $m_2$ does not propose to $w_2$ during $R$, but does so during $R'$. By a similar reasoning as above, we know that this is possible only if he is rejected by his *old* partner (say $w_3 = \mu(m_2)$) during $R'$ in favor of one or more better proposals. Let $m_3$ denote the *earliest* such proposal received by $w_3$ during $R'$.

Proceeding in this manner, we obtain a sequence of women and men $(w_i, m_i)$, starting with $w_1 = w$ and $m_1 = m$, such that (i) for $i \geq 1$, $m_i$ is rejected by his *old* partner $w_{i+1}$ during $R'$,

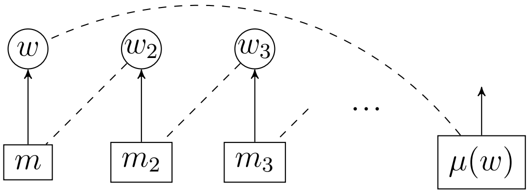

**Figure 5.2:** *The sequence of events arising in the proof of precedence lemma (Lemma 31). The rectangles and circles correspond to men and women respectively. A dashed edge stands for* "rejection by old partner during $R'$"*, while a solid edge denotes* "the earliest proposal received during $R'$ that is better than the old partner" *(except for the edge $m \to w$, which simply denotes a proposal event without the requirement of being the earliest such).*

and (ii) for $i \geq 2$, $m_i$ is the *earliest* proposal received by $w_i$ during $R'$ that she strictly prefers over her *old* partner $m_{i-1}$. This way, each pair of consecutive agents in the sequence, namely $(w_i, m_i)$ or $(m_i, w_{i+1})$, is associated with either a proposal or a rejection event during $R'$. Furthermore, a proposal event involving $(w_i, m_i)$ is preceded in time by a rejection event involving $(m_i, w_{i+1})$. Similarly, a rejection event involving $(m_i, w_{i+1})$ is preceded in time by a proposal event involving $(m_{i+1}, w_{i+1})$, and so on. This sequence is pictorially depicted in Figure 5.2.

Since the number of men and women is finite, some member of this sequence must eventually repeat. If the first agent to repeat is a man (say $m_k$), then we obtain that the event "*$m_k$ proposes to $w_k$*" is both preceded and succeeded in time by another event "*$m_k$ is rejected by his old partner $w_{k+1}$ during $R'$*", which is absurd. Therefore, the first agent to repeat in this sequence must be a woman. By a similar observation, it can be argued that no woman other than $w$ can be the first to repeat. This implies that the last agent in the sequence before a repeat happens is $\mu(w)$, as shown in Figure 5.2.

Thus, putting together the temporal implications of this sequence, we get that the event "*$m$ proposes to $w$ during $R'$* (and not during $R$)" is preceded in time by the event "*$\mu(w)$ is rejected by $w$*", as desired. □

Another interpretation of precedence lemma (Lemma 31) is that the manipulator cannot secure a better partner by simply permuting the part of her preference list below the *old* proposer, while keeping all men above and including the *old* proposer intact. Lemma 32 formalizes this.

**Lemma 32** (Shuffling men below the favorite proposer). *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \mathrm{GS}(\succ)$ and $\mu' = \mathrm{GS}(\succ')$. Let $\succ'_w$ be derived from $\succ_w$ by arbitrarily permuting the positions of the men below $\mu(w)$ while keeping the positions of all men above and including $\mu(w)$ intact. Then, $\mu(w) \succeq_w \mu'(w)$.*

*Proof.* Suppose, for contradiction, that $\mu'(w) \succ_w \mu(w)$. Let $R$ and $R'$ denote the runs of the Gale-Shapley algorithm on the profiles $\succ$ and $\succ'$ respectively. Notice that, by construction, we have

$\mu'(w) \succ_w \mu(w) \Rightarrow \mu'(w) \succ'_w \mu(w)$. That is, $w$ is matched under $\mu'$ to a man she strictly prefers over $\mu(w)$ according to $\succ'_w$. This means that $w$ receives one or more proposals during $R'$ that she strictly prefers over $\mu(w)$ according to $\succ'_w$. Let $t$ denote the *earliest* round during $R'$ at which $w$ receives a proposal that she prefers over $\mu(w)$ according to $\succ'_w$, and let $m_1$ denote the corresponding proposer.

Thus, $m_1 \in \text{Prop}(w, \succ') \setminus \text{Prop}(w, \succ)$. By precedence lemma (Lemma 31), we have that during $R'$, the event "$m_1$ *proposes to* $w$" is preceded in time by the event "$\mu(w)$ *is rejected by* $w$". Moreover, $\mu(w)$ is rejected by $w$ during $R'$ because, at an earlier round, she receives a strictly better proposal (say from $m_2$) according to $\succ'_w$. This, however, contradicts our assumption that $m_1$ is the earliest such proposal received by $w$ during $R'$. Hence, $\mu(w) \succeq_w \mu'(w)$. □

Lemma 32 rules out the option of 'permuting the men below the *old* partner' as a way of securing a better match. It is, however, possible that permuting the list in this manner might actually make the manipulator worse off. Our next result (Lemma 33) shows that this does not happen as long as the shuffle is confined to the men below the second-favorite proposer.

**Lemma 33** (Swapping proposers other than the top two)**.** *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \text{GS}(\succ)$ and $\mu' = \text{GS}(\succ')$. Let $\succ'_w$ be derived from $\succ_w$ by swapping the positions of an adjacent pair of men $(m_i, m_j)$ such that $m_i, m_j \in \text{Prop}(w, \succ) \setminus \{\text{Prop}(w, \succ, 1), \text{Prop}(w, \succ, 2)\}$ and making no other changes. Then, $\mu'(w) = \mu(w)$.*

*Proof.* Notice that the list $\succ'_w$ is derived from $\succ_w$ by swapping a pair of men below $\mu(w)$ while keeping the positions of all men above and including $\mu(w)$ intact. Therefore, from Lemma 32, we have that $\mu(w) \succeq_w \mu'(w)$. If $\mu'(w) = \mu(w)$, then Lemma 33 follows. Therefore, we will assume here onwards that $\mu(w) \succ_w \mu'(w)$. By construction, this also means that $\mu(w) \succ'_w \mu'(w)$.

We now claim that $m_i, m_j \in \text{Prop}(w, \succ')$. Indeed, the profiles $\succ$ and $\succ'$ differ only in the preference list of $w$, and the list $\succ'_w$ is derived from $\succ_w$ by changing the position of a fixed man $m_i$ (or, equivalently, $m_j$). Therefore, from Lemma 29, we have that $m_i, m_j \in \text{Prop}(w, \succ) \Rightarrow m_i, m_j \in \text{Prop}(w, \succ')$.

Let us now assume that $m_i \succ'_w m_j$ (the argument for $m_j \succ'_w m_i$ is analogous). Since $m_i, m_j \in \text{Prop}(w, \succ')$, $w$ must weakly prefer her matched partner under $\mu'$ over the man $m_i$ (according to $\succ'_w$). That is, $\mu'(w) \succeq'_w m_i$.

If $\mu'(w) \succ'_w m_i$, then the list $\succ_w$ can be thought of as having been obtained from the list $\succ'_w$ by swapping a pair of non-favorite proposers (namely $m_i$ and $m_j$) while keeping the positions of all the men above and including the favorite proposer, namely $\mu'(w)$, intact. Thus, by applying Lemma 32 with $\succ'$ and $\succ$ as the *old* and the *new* profile respectively, we get that $\mu'(w) \succeq'_w \mu(w)$, which contradicts the earlier condition $\mu(w) \succ'_w \mu'(w)$. Thus, $\mu'(w) = \mu(w)$. □

**Lemma 34** (Swapping the second and third-best proposers). *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \mathrm{GS}(\succ)$ and $\mu' = \mathrm{GS}(\succ')$. Let $\succ'_w$ be derived from $\succ_w$ by swapping the positions of an adjacent pair of men $(m_2, m_3)$ such that $m_2 = \mathrm{Prop}(w, \succ, 2)$ and $m_3 = \mathrm{Prop}(w, \succ, 3)$ and making no other changes. Then, $\mu'(w) \in \{\mu(w), m_3\}$.*

*Proof.* Our proof of Lemma 34 closely follows the proof of Lemma 33. Observe that $\succ'_w$ is derived from $\succ_w$ by shuffling the positions of a pair of men below $\mu(w)$ while keeping the positions of all men above and including $\mu(w)$ intact. Therefore, from Lemma 32, we have that $\mu(w) \succeq_w \mu'(w)$. If $\mu'(w) = \mu(w)$, the statement of Lemma 34 follows. Therefore, we will assume here onwards that $\mu(w) \succ_w \mu'(w)$. By construction, we also have $\mu(w) \succ'_w \mu'(w)$.

We now claim that $m_2, m_3 \in \mathrm{Prop}(w, \succ')$. Indeed, the list $\succ'_w$ is derived from $\succ_w$ by changing the position of a fixed man $m_2$ (or $m_3$). Therefore, it follows from Lemma 29 that $m_2, m_3 \in \mathrm{Prop}(w, \succ) \Rightarrow m_2, m_3 \in \mathrm{Prop}(w, \succ')$.

Since $m_2, m_3 \in \mathrm{Prop}(w, \succ')$ and $m_3 \succ'_w m_2$ by construction, it must be that $w$ is matched under $\mu'$ to someone she weakly prefers over $m_3$ (according to $\succ'_w$). That is, $\mu'(w) \succeq'_w m_3$.

If $\mu'(w) \succ'_w m_3$, then we can think of the list $\succ_w$ as having been obtained from $\succ'_w$ by swapping a pair of non-favorite proposers (namely $m_2$ and $m_3$) while keeping everyone above and including the favorite proposer, namely $\mu'(w)$, intact. Therefore, by applying Lemma 32 with $\succ'$ and $\succ$ as the *old* and the *new* profile respectively, we get that $\mu'(w) \succeq'_w \mu(w)$, which contradicts the condition $\mu(w) \succ'_w \mu'(w)$ obtained above. Therefore, $\mu'(w)$ must be $m_3$. □

**Lemma 35** (Swapping the first and second-best proposers). *Let $\succ$ and $\succ'$ be two preference profiles differing only in the preferences of a fixed woman $w$, and let $\mu = \mathrm{GS}(\succ)$ and $\mu' = \mathrm{GS}(\succ')$. Let $\succ'_w$ be derived from $\succ_w$ by swapping the positions of an adjacent pair of men $(m_1, m_2)$ such that $m_1 = \mathrm{Prop}(w, \succ, 1)$ and $m_2 = \mathrm{Prop}(w, \succ, 2)$ and making no other changes. Then,* $\mathrm{Prop}(w, \succ', 2) \in \{m_1, m_2\}$.

*Proof.* Our proof of Lemma 35 closely resembles our proof of Lemma 32. Let $m' = \mathrm{Prop}(w, \succ', 1)$ and $m'' = \mathrm{Prop}(w, \succ', 2)$. Suppose, for contradiction, that $m'' \notin \{m_1, m_2\}$. We claim that this condition implies that $m'' \succ'_w m_2$. Indeed, let $R$ and $R'$ denote the runs of GS algorithm on the profiles $\succ$ and $\succ'$ respectively. Notice that the profiles $\succ$ and $\succ'$ differ only in the preference list of $w$, and the list $\succ'_w$ is derived from $\succ_w$ by changing the position of a fixed man $m_1$ (or, equivalently, $m_2$). Therefore, from Lemma 29, we have that $m_1, m_2 \in \mathrm{Prop}(w, \succ) \Rightarrow m_1, m_2 \in \mathrm{Prop}(w, \succ')$. Hence, for $m''$ to be different from both $m_1$ and $m_2$, we must have $m'' \succ'_w m_2$, since, by construction, we have that $m_2 \succ'_w m_1$.

Observe that the condition $m'' \succ'_w m_2$ implies that $m' \succ'_w m'' \succ'_w m_2 \succ'_w m_1$. Thus, there is at least one man (namely $m''$) who is strictly preferred over $m_1$ by $w$ according to $\succ'_w$ and who proposes to $w$ during $R'$ but not during $R$. That is, $m'' \in \text{Prop}(w, \succ') \setminus \text{Prop}(w, \succ)$. Let $p$ be the *earliest* such proposal received by $w$ during $R'$. Clearly, $p \in \text{Prop}(w, \succ') \setminus \text{Prop}(w, \succ)$. By precedence lemma (Lemma 31), we know that during $R'$, the event "*$p$ proposes to $w$*" is preceded in time by the event "*$m_1$ is rejected by $w$*". The latter event, however, happens because at an earlier round, $w$ receives one or more proposals that she strictly prefers over $m_1$ according to $\succ'_w$. This contradicts the assumption that $p$ is the earliest such proposal. Therefore, $m'' \in \{m_1, m_2\}$. □

### 5.5.4 Proof of Lattice Containment (Theorem 25)

**Theorem 25** (Lattice containment result). *Let $\succ'_w$ be an optimal manipulation with respect to $\succ$ for woman $w$, and let $\succ''_w$ be its inconspicuous equivalent (as described in Theorem 24). Then, $S_{\succ''} \subseteq S_{\succ}$.*

*Proof.* Let us assume, for contradiction, that there exists a matching $\phi$ such that $\phi \in S_{\succ''} \setminus S_{\succ}$. Then, there must exist a pair $(m, w)$ that blocks $\phi$ with respect to $\succ$ (it follows from the argument in the proof of Theorem 23 that one of the blocking agents must be the manipulator $w$). Thus, $m \succ_w \phi(w)$ and $w \succ_m \phi(m)$.

Consider the set $S_{\succ''}$. Since $\phi \in S_{\succ''}$, it follows from Theorem 22 that $\phi(w) \succeq''_w \mu''(w)$, where $\mu'' = \text{GS}(\succ'')$. Moreover, from Theorem 24, we know that $\mu'(w) = \mu''(w)$ for $\mu' = \text{GS}(\succ')$. Therefore, we also have that $\phi(w) \succeq''_w \mu'(w)$.

By inconspicuousness of $\succ''_w$, we have that $\phi(w) \succeq_w \mu'(w)$. Combining this with the blocking pair condition above, we get that $m \succ_w \phi(w) \succeq_w \mu'(w)$. Once again, by inconspicuousness of $\succ''_w$, we have that $m \succ''_w \phi(w)$.

We also know that $\succ_m = \succ''_m$ by construction. Therefore, from the blocking pair condition, we have that $w \succ''_m \phi(m)$. Combining this with the condition $m \succ''_w \phi(w)$, we get that the pair $(m, w)$ blocks $\phi$ with respect to $\succ''$, which contradicts the assumption $\phi \in S_{\succ''}$. Therefore, $S_{\succ''} \subseteq S_{\succ}$. □

## 5.6 Concluding Remarks

We studied the problem of manipulation of Gale-Shapley algorithm by a single agent, and identified a class of manipulations called inconspicuous manipulations which are optimal, stability-preserving, and nearly identical to the manipulator's true preference list.

Several directions for future work remain. First, extensions of the current model to a setting with *several* self-interested manipulators (seeking an equilibrium strategy) will be interesting to study. It will also be interesting to find out if our techniques—in particular, swapping lemma—can lead to *faster* algorithms for optimal manipulation by a single agent or a coalition of agents; the current best algorithms for these problems are $\mathcal{O}(n^3)$ (Teo et al., 2001) and $\mathcal{O}(n^6)$ (Deng et al., 2015) respectively.

Finally, it would be of interest to conduct a similar analysis for *other* stable matching algorithms, such as ones that compute *minimum regret* (Knuth, 1997; Gusfield, 1987), *egalitarian* (Irving et al., 1987; Teo and Sethuraman, 1998) or *median* (Teo and Sethuraman, 1998) stable matchings.