

Manipulating Gale-Shapley Algorithm: Preserving Stability and Remaining Inconspicuous

Rohit Vaish

Indian Institute of Science, India
rohit.vaish@csa.iisc.ernet.in

Dinesh Garg

IIT Gandhinagar, India
dgarg@iitgn.ac.in

Abstract

We study the problem of manipulation of the men-proposing Gale-Shapley algorithm by a single woman via permutation of her true preference list. Our contribution is threefold: First, we show that the matching induced by an optimal manipulation is stable with respect to the true preferences. Second, we identify a class of optimal manipulations called *inconspicuous manipulations* which, in addition to preserving stability, are also nearly identical to the true preference list of the manipulator (making the manipulation hard to be detected). Third, for optimal inconspicuous manipulations, we strengthen the stability result by showing that the *entire* stable lattice of the manipulated instance is contained inside the original lattice.

1 Introduction

The theory of two-sided matching has been a cornerstone of market design, inspiring a wide array of applications such as school choice [Abdulkadiroğlu *et al.*, 2005a; Abdulkadiroğlu *et al.*, 2005b], 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 their true preferences. This, in particular, means that the Gale-Shapley algorithm is also vulnerable to manipulation.

Our interest in this paper 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 3) 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 1).

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 4) 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 5) strengthens the stability implication of Theorem 3 for inconspicuous optimal manipula-

tions. 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 2).

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.

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; Kobayashi and Matsui, 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 3), 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, which we consider an important contribution of this work.

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.

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 μ 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 μ 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 paper, 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 1 recalls these results.

Theorem 1 (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^{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 μ and μ' , define the *join* function $\mu_{\vee} = \mu \vee \mu'$ as follows: for each $m \in M$ and $w \in W$,

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

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

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

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

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

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 2 (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 μ and μ' are the matchings before and after the manipulation respectively. The agent w is referred to as the *manipulator*. In this study, we only consider *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 μ' 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 $\mu' = \text{GS}(\succ')$ and $\mu'' = \text{GS}(\succ'')$.

4 Main Results

Our first result (Theorem 3) shows that optimal manipulation of GS algorithm by a single woman preserves stability.

Theorem 3 (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, $\mu' \in S_{\succ}$.*

This is a positive result, since the stability of the resulting matching is not affected by optimal strategic behaviour of a single agent. Example 1 complements this result by showing that sub-optimal strategic behavior can lead to instability.

Example 1. (Sub-optimal manipulation can be unstable) *Consider the stable marriage instance shown below.*

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

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.

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 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 1, 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 4 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 4 (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 1, 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 3 for inconspicuous optimal manipulations. Recall from Theorem 3 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 3 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 5 shows that this is indeed the case for inconspicuous optimal manipulations.

Theorem 5 (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 4). Then, $S_{\succ''} \subseteq S_{\succ}$.*

Example 2 complements Theorem 5 by showing that the implication does not extend to non-inconspicuous optimal manipulations.

Example 2. *Consider the following stable marriage instance*

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

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 ϕ 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. \square

5 Proofs of Main Results

5.1 Proof of Stability of Optimal Manipulation (Theorem 3)

We start by stating a lemma (Lemma 6) that will be useful in the proof of Theorem 3. 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. The proof is omitted due to space constraints. (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 6. *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, $m \in \text{Prop}(w, \succ) \Rightarrow m \in \text{Prop}(w, \succ')$.*

We now provide the proof of Theorem 3.

Proof. (of Theorem 3) Suppose, for contradiction, that $\mu' \notin S_{\succ}$. Then, there must exist a pair (m', w') that blocks μ' 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 6 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. \square

5.2 Proof of Inconspicuous Equivalent of Optimal Manipulation (Theorem 4)

Our proof of Theorem 4 crucially relies on the *swapping lemma* (Lemma 7). 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 omit its proof due to space constraints. It will be convenient to denote the set of non-proposing men by $\text{Non-Prop}(w, \succ)$. Thus, $\text{Non-Prop}(w, \succ) = M \setminus \text{Prop}(w, \succ)$.

Lemma 7 (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 $\text{Prop}(w, \succ', 2) \in \{m_i, m_j\}$.

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).

Proof. (of Theorem 4) 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 $\text{Prop}(w, \succ', 2)$. Each such operation involves swapping a pair of adjacent men in the current list. By repeatedly invoking swapping lemma (Lemma 7), 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 $p = \text{Prop}(w, \succ', 1)$ and $q = \text{Prop}(w, \succ', 2)$.

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

- (a) *Promoting q* : The list $\succ_w^{(1)}$ 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 7) that the run of the algorithm on the profile $\succ^{(1)} = (\succ_{-w}, \succ_w^{(1)})$ is identical to that on \succ' . Thus, $\text{Prop}(w, \succ^{(1)}, 1) = p$ and $\text{Prop}(w, \succ^{(1)}, 2) = q$. Notice that any man above p in the list $\succ_w^{(1)}$ 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 $\succ_w^{(1)}$ above and including p resemble that in the list \succ_w . We achieve this by replacing the set of non-proposers above p in $\succ_w^{(1)}$ 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 $\succ_w^{(1)}$, which, by case (1) of swapping lemma (Lemma 7), does not affect the run of GS algorithm. We call the resulting preference list $\succ_w^{(2)}$.

It is easy to verify that after this step, the new list $\succ_w^{(2)}$ 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 $\succ^{(2)} = (\succ_{-w}, \succ_w^{(2)})$.

- (c) *Fixing the part of the list below q* : The final step in our construction involves a sequence of preference lists $\{\succ_w^{(3)}, \succ_w^{(4)}, \dots\}$. The list $\succ_w^{(k+1)}$ is derived from $\succ_w^{(k)}$ by *swapping* a pair of adjacent men (m_i, m_j) in $\succ_w^{(k)}$ such that (i) $\text{Prop}(w, \succ^{(k)}, 2) \succ_w^{(k)} m_i$ and $\text{Prop}(w, \succ^{(k)}, 2) \succ_w^{(k)} m_j$, and (ii) $m_i \succ_w m_j$ and $m_j \succ_w^{(k)} m_i$. 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 $\succ_w^{(\ell)}$ be the final list in this sequence, and let $\succ^{(k)} = (\succ_{-w}, \succ_w^{(k)})$ 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 4 holds for

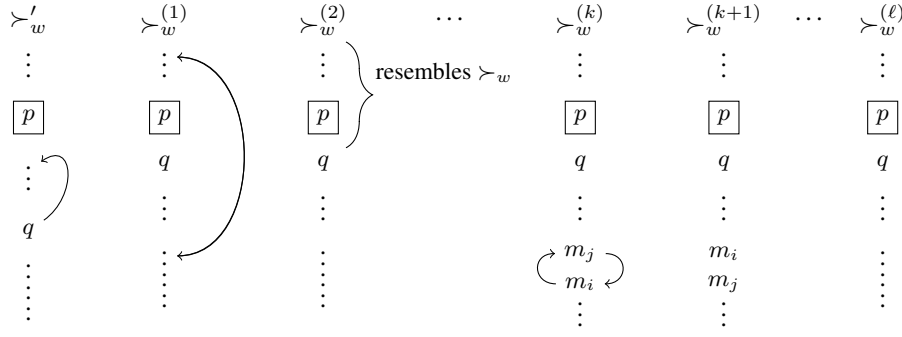


Figure 1: The sequence of preference lists constructed in the proof of Theorem 4. Here, $p = \text{Prop}(w, \succ', 1) = \mu'(w)$ and $q = \text{Prop}(w, \succ', 2)$.

$\succ_w'' = \succ^{(\ell)}$. That is, we will show that:

- The list $\succ_w^{(\ell)}$ can be obtained from the true list \succ_w by promoting q to the position right after p while making no other changes, and
- $\text{Prop}(w, \succ^{(\ell)}, 1) = \mu^{(\ell)}(w) = p$.

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 7), 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 7), 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 4. \square

5.3 Proof of Lattice Containment (Theorem 5)

Proof. (of Theorem 5) Let us assume, for contradiction, that there exists a matching ϕ such that $\phi \in S_{\succ''} \setminus S_{\succ}$. Then, there must exist a pair (m, w) that blocks ϕ with respect to \succ (it follows from the argument in the proof of Theorem 3 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 2 that $\phi(w) \succeq_w'' \mu''(w)$, where $\mu'' = \text{GS}(\succ'')$. Moreover, from Theorem 4, 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 ϕ with respect to \succ'' , which contradicts the assumption $\phi \in S_{\succ''}$. Therefore, $S_{\succ''} \subseteq S_{\succ}$. \square

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.

Our work motivates several directions for future work. 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.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. Part of this work was done at IBM Research, Bangalore, India. RV thanks Microsoft Research for a travel grant to present this work at IJCAI 2017.

References

- [Abdulkadiroğlu *et al.*, 2005a] Atila Abdulkadiroğlu, Parag A Pathak, and Alvin E Roth. The New York City High School Match. *American Economic Review*, pages 364–367, 2005.
- [Abdulkadiroğlu *et al.*, 2005b] Atila Abdulkadiroğlu, Parag A Pathak, Alvin E Roth, and Tayfun Sönmez. The Boston Public School Match. *American Economic Review*, pages 368–371, 2005.
- [Coles and Shorrer, 2014] Peter Coles and Ran Shorrer. Optimal Truncation in Matching Markets. *Games and Economic Behavior*, 87:591–615, 2014.
- [Deng *et al.*, 2015] Yuan Deng, Weiran Shen, and Pingzhong Tang. Coalition Manipulations of the Gale-Shapley Algorithm. *arXiv preprint arXiv:1502.07823*, 2015.
- [Dubins and Freedman, 1981] Lester E Dubins and David A Freedman. Machiavelli and the Gale-Shapley Algorithm. *The American Mathematical Monthly*, 88(7):485–494, 1981.
- [Gale and Shapley, 1962] David Gale and Lloyd S Shapley. College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.
- [Gale and Sotomayor, 1985] David Gale and Marilda Sotomayor. Ms. Machiavelli and the Stable Matching Problem. *The American Mathematical Monthly*, 92(4):261–268, 1985.
- [Gupta *et al.*, 2016] Sushmita Gupta, Kazuo Iwama, and Shuichi Miyazaki. Total Stability in Stable Matching Games. In *15th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2016)*, pages 23:1–23:12, 2016.
- [Gusfield, 1987] Dan Gusfield. Three Fast Algorithms for Four Problems in Stable Marriage. *SIAM Journal on Computing*, 16(1):111–128, 1987.
- [Irving *et al.*, 1987] Robert W Irving, Paul Leather, and Dan Gusfield. An Efficient Algorithm for the “Optimal” Stable Marriage. *Journal of the ACM (JACM)*, 34(3):532–543, 1987.
- [Irving, 1985] Robert W Irving. An Efficient Algorithm for the “Stable Roommates” Problem. *Journal of Algorithms*, 6(4):577–595, 1985.
- [Jaramillo *et al.*, 2014] Paula Jaramillo, Çağatay Kayı, and Flip Klijn. On the Exhaustiveness of Truncation and Dropping Strategies in Many-to-Many Matching Markets. *Social Choice and Welfare*, 42(4):793–811, 2014.
- [Knuth, 1997] Donald Ervin Knuth. *Stable Marriage and Its Relation to Other Combinatorial Problems: An Introduction to the Mathematical Analysis of Algorithms*, volume 10. American Mathematical Soc., 1997.
- [Kobayashi and Matsui, 2009] Hirotatsu Kobayashi and Tomomi Matsui. Successful Manipulation in Stable Marriage Model with Complete Preference Lists. *IEICE TRANSACTIONS on Information and Systems*, 92(2):116–119, 2009.
- [Kobayashi and Matsui, 2010] Hirotatsu Kobayashi and Tomomi Matsui. Cheating Strategies for the Gale-Shapley Algorithm with Complete Preference Lists. *Algorithmica*, 58(1):151–169, 2010.
- [McVitie and Wilson, 1971] David G McVitie and Leslie B Wilson. The Stable Marriage Problem. *Communications of the ACM*, 14(7):486–490, 1971.
- [Roth and Peranson, 1999] Alvin E Roth and Elliott Peranson. The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design. *American Economic Review*, 89(4):748–780, 1999.
- [Roth and Rothblum, 1999] Alvin E Roth and Uriel G Rothblum. Truncation Strategies in Matching Markets—In Search of Advice for Participants. *Econometrica*, 67(1):21–43, 1999.
- [Roth and Vate, 1991] Alvin E Roth and John H Vande Vate. Incentives in Two-Sided Matching with Random Stable Mechanisms. *Economic theory*, 1(1):31–44, 1991.
- [Roth *et al.*, 2004] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Kidney Exchange. *The Quarterly Journal of Economics*, 119(2):457–488, 2004.
- [Roth, 1982] Alvin E Roth. The Economics of Matching: Stability and Incentives. *Mathematics of operations research*, 7(4):617–628, 1982.
- [Roth, 2002] Alvin E Roth. The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics. *Econometrica*, 70(4):1341–1378, 2002.
- [Teo and Sethuraman, 1998] Chung-Piaw Teo and Jay Sethuraman. The Geometry of Fractional Stable Matchings and its Applications. *Mathematics of Operations Research*, 23(4):874–891, 1998.
- [Teo *et al.*, 2001] 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.