Some Tools for Simulating Deferred Acceptance

Justin Hadad*

This draft: November 2022

1 Introduction

Deferred acceptance has born several practical fixes and meaningful theoretical studies throughout the economics, operations research, and computer science literature. In this note we're interested in discussing the properties of deferred acceptance in simulation. Code will be included, which is prepared in Python and can be copied quite easily for replication. The full code is available at this GitHub link, and code is quite fast and amenable to any sorts of common or private value combinations which agents hold for members of the other side.

To begin, this short paper is meant to provide simulation (and straightforward applicability) for what has been a heavily researched theoretical and experimental project. The interested reader is encouraged to investigate [GS62] for the initial exposition, and [Rot08] for a more holistic overview of the practice.

Consider a market which consists of M men and W women. Each man m_i where $i = 1, \ldots, M$ has value m_i^j drawn uniformly from $(0, 1)$ for woman w_j, and each woman w_j has value w_j^i drawn uniformly from $(0, 1)$ for man m_i. m_i^j is independently and identically distributed from m_i^ℓ where $1 < \ell \neq j < W$; likewise, it is independently and identically distributed from m_k^j where $1 < k \neq i < M$. Values are independently and identically distributed for women in the same way. There is demand only for a single partner and preferences are strict. No utility is derived from an empty match.

The assignment mechanism analyzed here is men-proposing deferred acceptance: (0) Each person ranks those of the opposite gender in accordance with their preferences for a marriage partner. (1) Each man proposes to a woman. (2) If a woman receives more than one proposal, she "holds" one offer and rejects all others. (3) Any man rejected at step $k - 1$

*Written while at University of Zurich, Department of Economics; small edits made while at University of Oxford, Department of Economics.

makes a new proposal to his most preferred woman who hasn't yet rejected him. (4) Each woman holds her most preferred acceptable offer to date, and rejects the rest. The algorithm stops when no further proposals are made, at which point a match is created between each woman and the man whose proposal she is holding.

It can immediately be seen that men have no incentive to propose in some order different from their true preference ordering. The idea is quite straightforward: they may only miss out on a more preferred match if they choose to jump down their ordering. Women, however, have incentive to reject men for whom a cycle could be born that leads them to a more preferred match. To give some more intuition for this, let us explain the "man-optimal" stable match as opposed to the "woman-optimal" stable match.

We have just acknowledged that men can do no better than to propose down their preference ordering. There is no order of proposals which can make them better off, then, as it is already optimal in such a way. Women, if they proposed, would likewise get their woman-optimal match, then. It so happens that the men-optimal match is the "woman-pessimal" match, and vice versa—by nature of the proposing side proposing down their preference ordering, it necessarily must mean that the other side is receiving matches that are of their highest (least preferred, yet still stable) rank. Consider the stable matching which women get when they propose, versus when they accept; and consider briefly that they are different. Women can make manifest their more preferred match by treating all agents beneath this match as unacceptable; that is, reject all agents for whom they are less satisfied than they would be with their woman-optimal stable matching. This is termed in the literature as *truncating* their preferences.

Before investigating this strategic behavior more, we present the algorithm. It is written for men to propose, but it is quite easily transformed to have men propose (by just changing the order of the parameters).

2 Establishing the algorithm

We could define values that agents have for agents of the other side if we would like, or, alternatively, we can randomly generate them using the method described above: independently and identically drawn from the uniform distribution.

```
men_values = np.array([np.random.permutation(range(0, women))
    for _ in range(men)])
women_values = np.array([np.random.permutation(range(0, men))
    for _ in range(women)])
```

They key to the algorithm is establishing a series of arrays of offers, tentative acceptances, and permanent rejections. The logic is straightforward: we iterate through offers and reject, at first, all but the most preferred of them. We iterate through until the array of unmatched men are empty.

```
offers = [[] for woman in range(len(acc))] # women-indexed offers
matches = [-1 for woman in range(len(acc))] # women-indexed matches
unmatched = [i for i in range(len(prop))] # list of men who are unmatched
offered = [] # list of women who are offered men
```

The core of the logic follows: while there is some unmatched man, have him propose to his most preferred woman to whom he has not already proposed. Then, among all women to whom there have been proposals, choose one of that lot.

```
while unmatched:

    for man in unmatched:
        for idx, woman in reversed(list(enumerate(prop[man]))):
            if woman != -1:
                offers[woman].append(man)
                prop[man][idx] = -1
                if woman not in offered:
                    offered.append(woman)
                break

    unmatched = []
```

Once the proposals are made, each woman for whom there has been a proposal declines all but one. These rejected men are added to the unmatched pool.

```
    for woman in offered:
        current_man = matches[woman]
        for man in reversed(acc[woman]):
            if man == current_man:
                break
            if man in offers[woman]:
                if current_man != -1:
                    unmatched.append(current_man)
                matches[woman] = man
```

```
                break
        for man in offers[woman]:
            best_man = matches[woman]
            if man != best_man and man != current_man:
                unmatched.append(man)

    offers = [[] for woman in range(len(acc))] # women-indexed offers
    offered = []

return matches
```

This is the bulk of the logic. The GitHub link contains iterations of this where each proposal is printed, so the user can see how the algorithm works precisely, especially how differences manifest across men-proposing deferred acceptance ("MPDA") and women-proposing deferred acceptance ("WPDA").

We will now discuss some short notes on the simulations, each of which was originally a shorter, stand-alone piece: one on comparative welfare, another on truncation strategies, and a third on the value of competition which is not discussed here, but ideas from which have become Hadad (2023).

3 Comparative Welfare

Note 1. Averaging over one hundred iterations, we get these average ranks for men in MPDA and WPDA respectively, with market sizes up to ten. Ranks start at one in the best case, and increase to the size of the other market in the worst case (though we hardly get there; average rank at worst has an *ex ante* maximum of a half the number of agents on the other side of the market).

The average rank trend holds as we increase market size. The case of as many as 50 men and women is enlightening; see 4a and 4b for the average ranks of men and women in respective market sizes when either side proposes. They look similar! And this is the point, indeed: that average rank grows more disparate from proposing and accepting in the exactly balanced market, and otherwise there is just one stable match. The repercussions of this are well-documented: truncating does not do much in the unbalanced market, whereas it is quite useful in the balanced market. Vying for proposing rites is critical then when the number of agents is the same; we will cover that soon.

For a clearer exposition of what we just discussed, see 5. This captures the difference between Figures 4a and 4b. Following intuition, because we expect just one stable matching

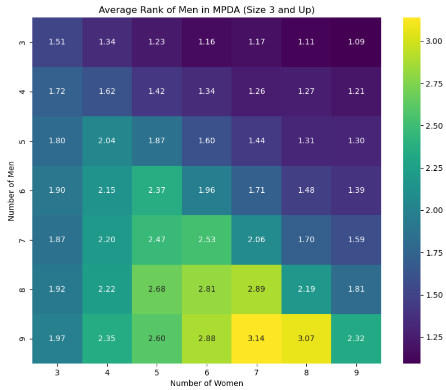
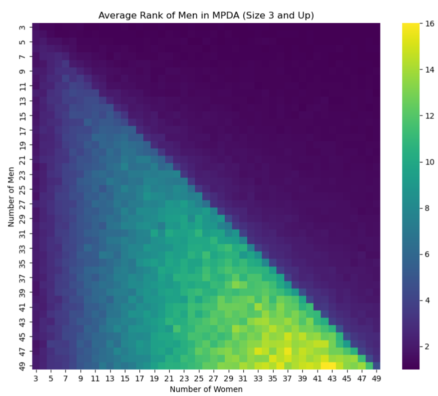


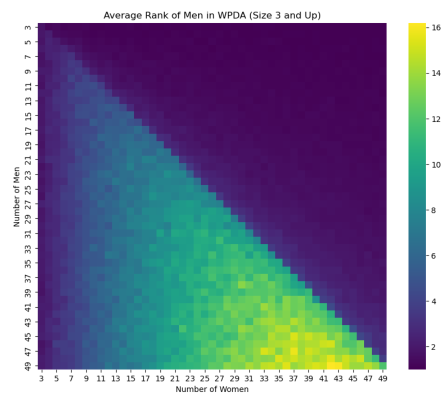
Figure 1: MPDA

Figure 2: WPDA

Figure 3: Average rank for men, up to market size ten



(a) MPDA



(b) WPDA

Figure 4: Average rank for men, up to market size 50

when the market is quite unbalanced, the gap in average rank is largest in the balanced case.

From Pittel (1989) (the author of which has done a considerable amount of work on the mathematics of such a problem) we know that in a balanced market of n men and n women, men's average matches approach woman $[\log(n) - 1]$, and women's approach man $\left[\frac{n}{\log(n)} - 1\right]$. It can be easily shown that our gap as in 5 abides by this. Theory enthusiasts are absolutely pointed to the very readable Ashlagi et al (2017) which discusses the effect of competition on matching markets. Simulations are included there, as well.

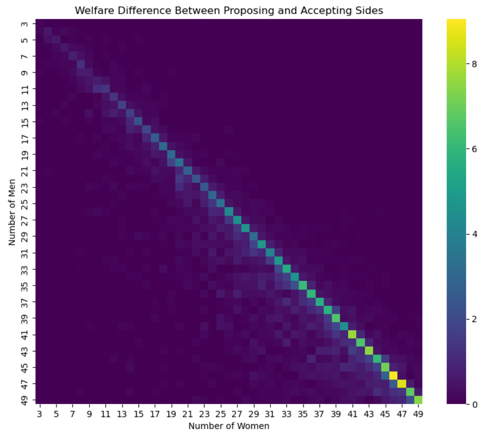


Figure 5: Difference between simulated average rank for men in MPDA and WPDA

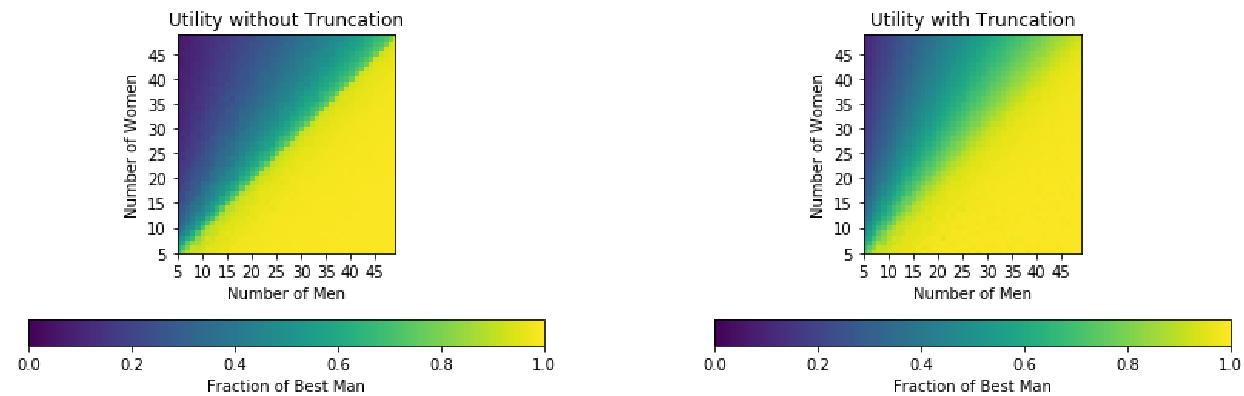
4 Truncation

As discussed in the introduction, deferred acceptance is strategy-proof for the proposing side but not for the accepting side. [CS14] find that truncation—the rejection of potential partners with whom a match does not meet the utility they would get in their optimal stable match—can improve utility for the accepting side. In this note I affirm the findings of [CS14] by demonstrating utility gains from truncating, and analyze if truncation is more useful than an increase in the size of the proposing side. For ease, consider men-proposing deferred acceptance, where women (the accepting side) have the option to truncate. As women are our focal point here, we place women on the x-axis for ease.

Let us visit a new metric briefly; the fraction of women who achieve their most preferred man. This is contained in Figure 6a; it can be viewed analogously to the average utility each woman achieves from a given market size. This and all other figures are created with differently sized markets, with number of men $M = (5, 50)$ and number of women $W = (5, 50)$, and with utilities averaged over 30 iterations.

As previously discussed, being on the short side of the market is strictly preferred to being on the long side. We now look at the optimal fraction of men to decline in order to get the most preferred stable matching. Like [CS14], if the utility gains from any set of truncations is the same, the highest level of truncation is chosen. I graph the average utility each woman receives from the level of truncation that is optimal given contemporary market conditions. Figure 6b shows the utility received from truncating up to their match they would otherwise receive from proposing.

Figure 7a shows the difference in utility between truncating and not truncating, or the



(a) Fraction of Women Receiving Their Best Man, MPDA with No Truncation

(b) Fraction of Women Receiving Their Best Man, MPDA with Optimal Truncation

Figure 6: Comparative Analysis of MPDA with and without Truncation

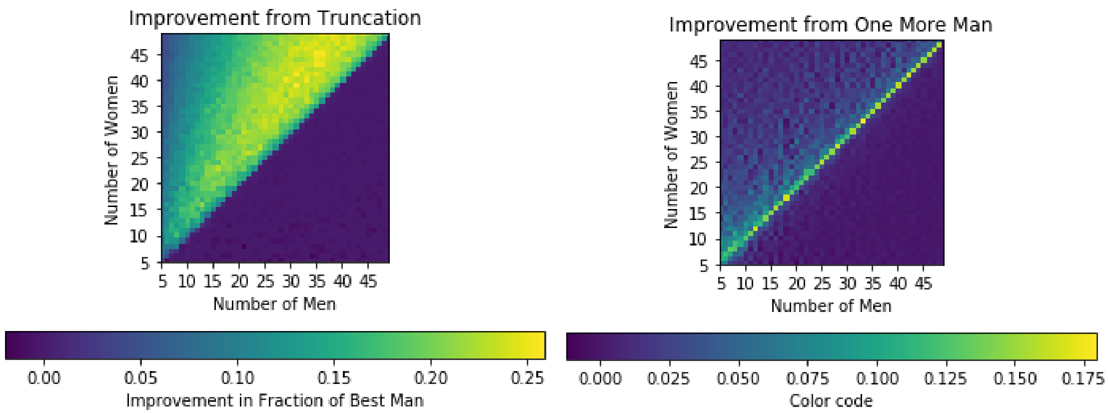
difference between Figures 6b and 6a . Note that optimally truncating never decreases utility; it is possible to truncate at a threshold utility less than the least-desired man, so that no match is changed. Truncating is a particularly useful strategy in balanced markets and when the number of women exceeds the number of men. Note this is the utility *difference*; there is little improvement from truncation when the number of men is greater than the number of women because women are expected to get the best possible outcome then even without truncation. Note additionally that this whole analysis is instructive, perhaps, but indeed redundant: we could merely have inspected the average utility difference between the proposing and accepting sides!

Now, there is another way for women to improve utility: increase their demand, or introduce *another man* into the market. Figure 7b shows the improvement in women's average utility from adding another man into the market.

We have already discussed that an increase in the number of men in a balanced market generates the biggest increase in women's average utility. Adding an extra man when utility already nears 1 (when the number of men is significantly higher than the number of women) will not drastically change women's utility, meanwhile gains from increased competition are expected when the number of women is significantly larger than the number of women, but are smaller than in a balanced market.

Truncation and increased competition strictly improve women's average utility. I now consider if truncating optimally or if increasing the number of men in the market is more optimal for women. 8 includes this comparison. This analogous to comparing the optimal match (from optimal truncation) to a slight increase in competition.

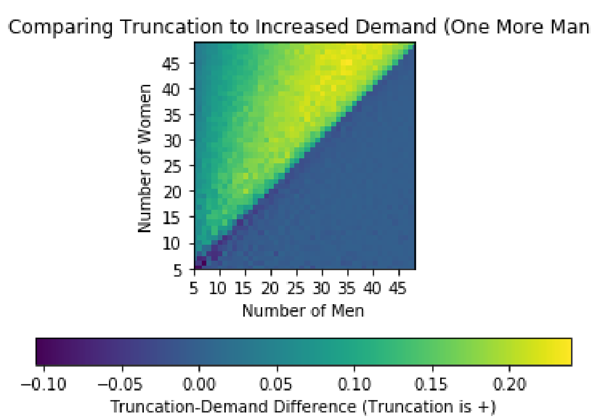
Of course, this begets the opportunity for more analysis, namely the level of competition



(a) Utility Gains from Truncation (b) Utility Gains from Adding One Man

Figure 7: Comparative Analysis of MPDA with and without Truncation

Figure 8: Difference in Utility: Truncation vs. One More Man



that agents prefer to their optimal match . A conversation around the theory in that line of thought is available in [Had23].

5 Final remark

The code linked in the GitHub contains the toolkit for the analyses above. The interested reader is encouraged to explore possible changes, for example in value distributions, conditional or costly rejections, and incomplete information.

References

[CS14] Peter Coles and Ran Shorrer. Optimal truncation in matching markets. *Games and Economic Behavior*, 87:591–615, 2014.

[GS62] David Gale and Lloyd Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 1962.

[Had23] Justin Hadad. On the analytical bounds for average rank in one-to-one two-sided matching markets. *Working Paper*, 2023.

[Rot08] Alvin Roth. Deferred acceptance algorithms: history, theory, practice, and open questions. *International Journal of Game Theory*, 36(3):537–569, March 2008.