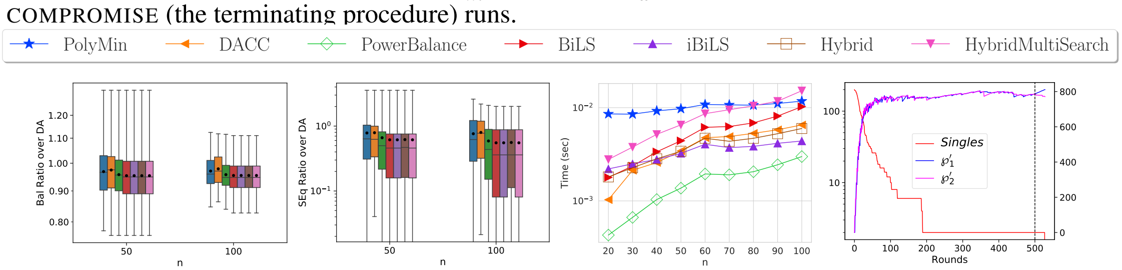
We thank all reviewers for the time they invested to review this paper and share their insights. In this letter, we respond to all reviewer comments, quoted verbatim, **bold in teal color**; content from the paper is quoted in blue.

Application of the proposed algorithm in practice; visualization of the matching process; real-world stable marriage data. We have conducted experiments on real-world data, yet could not include them within page limits. We extract distributions from the data of an online dating service [2]. The data consists of 17,359,346 anonymous ratings, on the $1-10$ scale, of 168,791 profiles made by 135,359 LibimSeTi users, along with gender information. We remove users of unknown gender and those who have not rated the opposite gender, and construct a 2D distribution of the frequency of each pair of ratings (i, j). Drawing from this distribution, we generate data of $n = 100$; the limited scale of ratings does not suffice to generate interesting preference lists at larger size. We resolve ties using 80% *randomness* and 20% *popularity* (P), i.e., the global ranking of agents by all ratings. We run 50 instances per size, and plot quality and runtime results. We visualize the process for PowerBalance with the instance that yields the *median* Sex-Equality Cost. In each round, we measure the number of *Single* agents and the sum of κ index values from men ($\wp'_m$) and women ($\wp'_w$), which dictate which side proposes in the following round. The left-side axis marks the scale of singles and the right-side axis marks the scale of $\wp'_m$ and women $\wp'_w$. The vertical black dashed line marks the round in which COMPROMISE (the terminating procedure) runs.



(a) Bal (b) SEq (c) Time (d) Visualization ($n = 100$)

Publication of the algorithm in an implemented code (e.g. Java as stated in Line 304). Please note that a link to code and data is provided in footnote 3 at Line 304, Page 7.

Pseudocode for the subroutines PROPOSE and COMPROMISE. The pseudocodes are given below.

```
procedure PROPOSE(p, μ)                              ▷ proposer p, matching μ
    if (μ(p) = ∅ ∧ κ_p < n) then
        q = ℓ_p[κ_p]                                 ▷ p wants to propose to q
        if accept(q, p) then
            if μ(q) ≠ ∅ then                         ▷ break up q if married
                r = μ(q); μ = μ \ {⟨q, r⟩}
            μ = μ ∪ {⟨p, q⟩}; κ_q = pr_q(p)          ▷ match p and q
        else κ_p = κ_p + 1                           ▷ q rejects p
```

```
function COMPROMISE(C, μ)                            ▷ side C, matching μ
    if (C == M) then F = W
    else F = M
    while (∃x ∈ C : μ(x) = ∅ ∧ κ_x < n) do
        for all x ∈ C do PROPOSE(x,μ)
    while (∃x ∈ F : μ(x) = ∅ ∧ κ_x < n) do
        for all x ∈ F do PROPOSE(x,μ)
    return μ
```

Relevance to real world matching problems unclear. As we report in Lines 33–34, citing recent results by Hassidim et al. [3] brought to our attention by Roth [4], the set of possible stable matchings is large in real-world markets [3].

What does "SEQ ratio over DA" mean? It means that we normalize cost results, dividing by the corresponding best cost the DA algorithm can obtain; lower cost values are better.

A characterization of under what cases DA leads to very unfair matchings. The most challenging cases are those of *symmetric* distributions on two sides, even more so when some choices are universally popular, as in dataset D.

Minor comments regarding language/notation/citations/etc. Thank you for these comments, we will heed them.

"10 million students" in a Chinese admissions market. We cite this fact as a motivation for research. In most countries the sizes of student admission markets are in the order of a few thousands.

Why not compare against an IP formulation of the problem? A mixed-integer linear programming formulation is indeed possible, e.g., minimizing an auxiliary variable X such that $SEq < X$ and $SEq > -X$. To our knowledge, such an IP-based solution has not been attempted to date. Sethuraman et al. [6] consider fairness, yet only in terms of a *median* stable matchng, not in terms of sex-equality cost. Sethuraman opined that an IP-based solution method for sex-equal stable marriage is "unlikely to have polynomial running time" [5]. In a recent study [1], Ágoston et al. were able to solve only a pruning-intensive variant of the stable matching problem using an IP technique on real data; other variants were infeasible for sizes larger than 100; matching 100 students to 20 schools with common quotas took 39,560 seconds (almost 11 hours) [1]. On the other hand, our quadratic-time algorithms surpass or match the quality that Approx achieves with all carefully tested values of ε for which a solution exists (Figure 3); thus, they achieve *near-optimal* solutions. We reconfirmed this fact in communication with the authors of Approx [7].

[1] Kolos Csaba Ágoston, Péter Biró, and Iain McBride. Integer programming methods for special college admissions problems. *Jnl Comb. Opt.*, 32(4):, Nov 2016. 1
[2] Lukas Brozovsky and Vaclav Petricek. Recommender system for online dating service. In *Proceedings of Conference Znalosti 2007*, Ostrava, 2007. VSB. 1
[3] Avinatan Hassidim, Assaf Romm, and Ran I. Shorrer. Need vs. merit: The large core of college admissions markets. https://ssrn.com/abstract=3071873. 1
[4] Alvin E. Roth. Private Communication, 2018. 1
[5] Jay Sethuraman. Private Communication, 2019. 1
[6] Jay Sethuraman, Chung-Piaw Teo, and Liwen Qian. Many-to-one stable matching: Geometry and fairness. *Math. Operations Research*, 31(3):581–596, 2006. 1
[7] Hiroki Yanagisawa and Shuichi Miyazaki. Private Communication, 2019. 1