



Two-Sided Matching with Indifferences: Using Heuristics to Improve Properties of Stable Matchings

Christian Haas¹ 

Accepted: 6 June 2020 / Published online: 1 July 2020
© The Author(s) 2020

Abstract

Two-Sided Matching is a widely used approach to allocate resources based on preferences. In Two-Sided Matching problems where indifferences are allowed in the preference lists, finding stable matchings with certain properties is known to be NP-hard and, in some instances, even NP-hard to approximate. This article, therefore, considers the use of heuristics in Two-Sided Matching scenarios with indifferences to find and explore potential solutions and their properties. Two heuristics, a Genetic Algorithm and Threshold Accepting, are compared to existing approaches with respect to their ability to generate alternative stable solutions based on initially stable matchings for scenarios with either complete or incomplete preferences. The evaluation shows that using these types of heuristics is a valid approach to obtain matchings with improved properties such as fairness or average matched rank, compared to existing algorithms. Overall, this approach allows for the exploration of alternative stable matchings and subsequent selection of a best solution given selected evaluation criteria.

Keywords Two-Sided Matching · Heuristics · Matching properties · Preferences with indifferences · Complete and incomplete preferences

1 Introduction

The allocation of resources based on non-monetary preferences is a widespread and important challenge in markets where monetary allocations are not desirable or possible. In such markets, participants express their valuations through preference rankings and an allocation mechanism determines which participants are being matched to each other based on these rankings. This general concept is applied in a variety of settings. In the US National Residents Matching Program (NRMP) and

✉ Christian Haas
christianhaas@unomaha.edu

¹ Department of Information Systems and Quantitative Analysis, University of Nebraska at Omaha, Omaha, NE 68182, USA

similar programs in Canada and Scotland, medical interns are allocated to hospitals by expressing their mutual preferences (Abdulkadiroglu and Sönmez 2013). In school choice problems, students and schools express their preferences over available options to determine student placement in cities such as Boston, Minneapolis, or Seattle (Roth and Sotomayor 1992; Abdulkadiroglu and Sönmez 2003). As not everyone can be allocated to their favorite choice, matching mechanisms are used to find allocations that satisfy certain desirable characteristics, such as stability, fairness, Pareto efficiency, or incentive compatibility. Further examples of preference-based matching include the allocation of (computational) resources being shared in a social setting (Caton et al. 2014), student-project allocations in university settings (Abraham et al. 2007), or the matching of mentors to mentees in professional development scenarios (Haas et al. 2018; Haas and Hall 2019).

For the previously mentioned examples, and the allocation of resources based on preferences in general, the field of Two-Sided Matching is a successful and established approach to find such allocations. Similarly to monetary markets, one side provides resources (e.g., school or college places) and the other side wants to “consume” the available resources. In contrast to monetary markets, however, the allocation decision has to be derived based on preference rankings, not on a willingness to pay. That is, each side ranks participants of the other side in an ordinal ranking, where a higher rank indicates that a match would be preferred.

Over time, a variety of different mechanisms have been developed for these matching markets that create allocations based on certain criteria. Some of these algorithms assume that preferences are strict, i.e., each participant is always able to provide a strict priority structure between two options. However, in many practical scenarios such a restriction can be overly prohibitive. For a variety of reasons participants might be indifferent between certain choices, and in markets with a large number of options the effort to rank each individual option separately and uniquely might be prohibitively expensive. In addition, legal reasons (e.g., affirmative action) can lead to providers (schools, hospitals, etc.) ranking applicants in separate indifference classes, thus creating ties in the preferences (Erdil and Ergin 2017).

The existence of indifferences between choices has been observed in several scenarios. In the residents matching programs, hospitals were observed to be indifferent between (a subset of) medical interns (Irving and Manlove 2008). Students in a school choice mechanism in Amsterdam sometimes provide the same numerical score to multiple schools even though they are allowed to provide a unique ranking for each school (de Haan et al. 2015). Preferences from conferences that use a Two-Sided Matching to allocate reviewers to papers exhibit substantial indifferences (Mattei and Walsh 2013). Similarly, preferences collected from applications such as Mentor-Mentee Matching also provide evidence that a substantial number of participants can be indifferent between multiple options (Haas and Hall 2019).

From an allocation perspective, the inclusion of indifferences and the addition to potentially incomplete rankings, i.e., participants not providing a ranking for each option, leads to computational and optimization challenges. If indifferences between options are allowed, finding an optimal allocation, in general, will be NP-hard and even difficult to approximate for certain allocation criteria (Halldórsson et al. 2007). On the other hand, while efficient mechanisms exist for

finding certain types of solutions, the most general form of Two-Sided Matching relies on approximation algorithms or heuristics to find matchings with desirable properties due to the mentioned NP-hardness.

Previously, a set of different approximation algorithms and local search procedures have been suggested for calculating allocations to the general Two-Sided Matching formulation when indifferences and incompleteness are allowed. While they offer guarantees about specific properties, e.g., the minimum number of matched participants, their actual (average) performance is unclear and they are designed for a specific combination of allocation objectives (such as finding a large stable match). Previous work shows that there can be many different allocations that satisfy the stability criterion but differ with respect to other criteria, and suggested mechanisms often focus on finding a stable allocation that optimize one or more other criteria. Finding allocations that are stable and improve upon other solutions with respect to additional criteria such as fairness or number of matched participants also directly translates into benefits for the participants in the matching market.

To determine and evaluate a wider set of potential allocations, this article proposes the use of heuristic algorithms for calculating solutions to the general Two-Sided Matching problem. Specifically, it considers two heuristic approaches, a Genetic Algorithm (GA) and a Threshold Accepting (TA) algorithm, as well as their combination with respect to their ability to find allocations with improved properties in the case of complete or incomplete preferences with indifferences. They also offer the flexibility to include multiple objectives when calculating allocations. Heuristic approaches have been used for matching problems before, e.g., GAs for strict (Kimbrough and Kuo 2010) and incomplete preferences (Haas 2014), and GAs to find fair solutions (Nakamura et al. 1995). In contrast to previous work in this field, this article specifically considers the case of non-strict, not necessarily complete preferences.

Overall, this article provides a systematic evaluation of the allocation properties provided by the suggested heuristic approaches compared to other alternative allocation mechanisms. Properties in this case refers to criteria such as stability, fairness, or number of matched participants. For the case of (both complete and incomplete) preferences with indifferences, the evaluation of the allocation properties shows that heuristics can find significantly improved allocations for generalized Two-Sided Matching problems by providing substantial improvements on secondary objectives such as the fairness or average matched rank characteristics of the solutions.

The structure of the article is as follows. Section 2 overviews related work. Section 3 describes the general Two-Sided Matching formulation and defines matching properties, and Sect. 4 describes several Two-Sided Matching algorithms and heuristics. After introducing the simulation environment used to systematically evaluate the algorithms in Sects. 5, 6 and 7 present the evaluation results which show that in many circumstances heuristics can improve upon previous mechanisms, especially for objectives such as average matched rank. Finally, Sect. 8 discusses the findings and future work.

2 Related Work

Since the seminal paper of Gale and Shapley (1962), who introduced the concept of Two-Sided Matching and proposed the Deferred Acceptance (DA) algorithm for complete preferences without indifferences, literature on Two-Sided Matching as well as its application areas has grown considerably [see Roth (2008) for a broad survey of application areas]. Generally, the literature on Two-Sided Matching can be categorized into several areas: (1) preference types, (2) solution objectives and computational complexity, and (3) incentive compatibility.

2.1 Mechanisms for Different Preference Types

Regarding preferences, the DA and much of the initial literature focuses on problems with strict preference orderings. If ties (indifferences) are introduced into the problem, certain characteristics of the algorithms can no longer be guaranteed. For example, in order to use the standard algorithms such as DA, the ties have to be broken first as the DA only allows strict preferences as input. Erdil and Ergin (2008, 2017) introduced an extension to the DA that can cope with ties in preferences. Their algorithm tries to find potential Pareto-improvement cycles in a given solution which might improve the number of matched participants. As Gusfield and Irving (1989, p 219) note, however, many of the strong results for DA and related algorithms depend upon strict preference orderings, and characterizing stable matchings under partial ordering remains a largely open problem.

2.2 Allocation Objectives

Considering solution objectives, the de facto standard objective for Two-Sided Matching solutions is stability, often defined as the lack of unstable pairs in the solution (see the definition of stability in Sect. 3). Stability is considered a must-have characteristic of solutions as even one unstable pair can lead to chaotic unraveling of the solution if agents are able to act upon their preference (Knuth 1997). However, it was shown early on that DA is biased as it finds the optimal stable matching for one side, and the pessimal stable matching for the other side (Knuth 1997). At the same time, depending on the actual preferences the number of stable matchings can be large, sometimes exponential in the size of the problem (Gusfield and Irving 1989; Knuth 1997). This existence of potentially many stable solutions implies that they can be evaluated based on additional metrics such as *fairness* of the solution to both sides or *average matched rank (AMR)* that measures how close to their most preferred option participants are matched with on average.

The existence of efficient algorithms for multi-objective versions of Two-Sided Matching depends on the preference assumptions. For strict and complete preferences, Irving et al. (1987) efficiently compute the best stable matching with respect to average matched rank, and Iwama et al. (2010) provide an approach for an approximately fairness-best solution. In the complete preference setting, heuristics

have been studied to obtain solutions, with the GA being a prominent example. For example, Nakamura et al. (1995) study whether a GA can yield stable matchings with higher fairness than the DA solutions, yet do not consider indifferences or other objectives. Aldershof and Carducci (1999) describe a GA to compute stable solutions from random initial assignments, with stability as the sole objective. Furthermore, both Vien and Chung (2006) and Kimbrough and Kuo (2010) compare a GA with multiple objectives to the standard algorithms, yet neither of them consider indifferences in preferences. However, when indifferences are allowed finding either AMR-optimal or fairness-optimal stable solutions becomes NP-hard (Manlove et al. 2002), and the corresponding scenario is referred to as Stable Matching with Ties (SMT). Axtell and Kimbrough (2008) discuss trade-offs between stability and matched ranks; Klaus and Klijn (2006b) study (procedural) fairness and stability; and Iwama et al. (2010) propose an algorithm that approximately yields the fairness-best stable matching. Other approaches that look at different or multiple objectives for certain matching problems include (Vien and Chung 2006; Klaus and Klijn 2006a); (Pais 2008; Pini et al. 2011); and (Boudreau 2011) which also consider economic criteria such as matched ranks.

The most general setting with incomplete preferences and indifferences, often referred to as Stable Matching with Ties and Incompleteness (SMTI), focuses on the common objective to find a large stable solution, i.e., a solution that matches as many participants as possible. This problem is also NP-hard, and sometimes even hard to approximate (Iwama et al. 1999; Halldórsson et al. 2007). Previous work in this setting has focused on approximation algorithms that yield non-trivial lower bounds for the size of the matching (Halldórsson et al. 2007; McDermid 2009; Király 2011; Paluch 2014), and local search heuristics that try to improve a given stable solution (Gelain et al. 2013). Recently, Diebold and Bichler (2017) evaluate different algorithms for one-sided and two-sided matching, yet do not consider approximation algorithms or heuristics which would be required for larger problem sizes.

2.3 Incentive Compatibility and Strategic Aspects

Besides allocation properties, potential strategic incentives by participants in different mechanisms has been explored as well. The impossibility result by Roth (1982) [see Roth and Sotomayor (1992) for elaboration] shows that there is no mechanism which is incentive compatible for both sides and at the same time always able to yield a stable outcome. Hence, by focusing on algorithms that provide stable solutions, at least some of the participants have a strategic incentive to manipulate the preferences they submit to the Two-Sided Matching market. Previous work on incentive compatibility and strategies are mainly focusing on the DA. The DA is strategy-proof for the 'proposing' side, yet due to the impossibility theorem not for both sides. Roth and Rothblum (1999) discuss truncation, i.e. the deliberate omission of certain admissible participants from one's preference list, as a potential manipulation strategy. This type of strategy has been subsequently studied from a theoretical perspective (Ehlers 2008; Pini et al. 2011; Coles and Shorrer 2014), and

experimental setting (Echenique et al. 2016). In addition, manipulation occurrence and its effects on the practical applications of college admissions and school choice problems (Abdulkadiroglu et al. 2009; Kesten 2012; Pathak and Sönmez 2013) and mentor-mentee applications (Haas and Hall 2019) have been studied as well. The latter, considering an SMTI scenario with several sets of preferences, analyzes that manipulation success (i.e., probability that a manipulation is successful) and manipulation effects on the participants. The analysis shows that manipulation effects can be very similar across different algorithms and that potential losses can outweigh potential gains. However, additional work is necessary to study preference manipulation in generalized SMTI scenarios.

Overall, while the one-sided strategy-proofness of the DA can be a beneficial feature in scenarios where one side is less likely to manipulate or has fewer incentives to do so, the fact that DA yields solutions with inferior properties with respect to secondary objectives such as number of matched participants led to the development of the mentioned specialized SMTI algorithms without a specific consideration of preference manipulation. In general, the comparison between the DA as one-sided strategy-proof mechanism and other mechanisms such as the Boston mechanism is complex as it involves a variety of trade-offs between different mechanism properties (de Haan et al. 2015). These trade-offs have to be investigated for a given scenario, yet the absence of strategy-proofness for participants might be outweighed by improvements to the resulting allocation properties.

3 Two-Sided Matching Model

In a Two-Sided Matching scenario, participants i of one side, X , need to be matched with participants of the other side, Y . In total, there are $n_X + n_Y$ participants in the market. For easier notation and without loss of generality, the index j will denote participants of the opposite side. Each participant i has a *preference profile* $P_i = (P_{i,j_1}, \dots, P_{i,j_n})$, $j \in Y$ over participants of the other side with whom they want to be matched, where $P_{i,j}$ denotes the preference rank that participant i has towards participant $j \in \{Y \cup \emptyset\}$. The preference towards \emptyset indicates the preference for being unmatched. Preference profiles are transitive and anti-symmetric. The preference profiles represent transitive priority structures $\succsim = (\succsim_{i \in X})$, where each participant of the opposite side is ranked according to its priority. The asymmetric part \succ_i indicates a strict priority, whereas the symmetric part indicates an indifference. All participants j with $j \succ_i \emptyset$ are said to be *acceptable* for participant i (and vice versa).

A preference profile is called *complete* if $j \succ \emptyset$ for all participants j . If $\emptyset \succ j$ for some participants j , the preference profile is called *incomplete*. This refers to a situation where participant i wants to remain unmatched rather than being matched to participant j . A preference profile is *strict* if $\forall j \in X$ ($j \in Y$), \succsim_j is asymmetric. If $j \sim k$ for some participants j and k , then the preference profile is said to have *indifferences*, or ties.

Given the representation of participants' preferences on both sides, the goal of Two-Sided Matching is to find a matching $\mu = \langle X, Y \rangle$ that defines which participants are matched together. $\langle X, Y \rangle$ consists of pairs $\langle x, y \rangle$ with $x \in X$ and $y \in Y$.

3.1 Matching Properties

A matching μ is characterized by certain properties. The concept of *blocking pairs* is the most fundamental evaluation criterion for Two-Sided Matching allocations. It considers the question of whether participants have the incentive to deviate from a given matching. A blocking pair is defined as a pair $\langle x, y \rangle$, $x \in X$, $y \in Y$, such that

1. $\mu(x) = \emptyset$ or $y \succ_x \mu(x)$, AND
2. $\mu^{-1}(y) = \emptyset$ is single or $x \succ_y \mu^{-1}(y)$, AND
3. x and y are mutually acceptable.

Blocking pairs are essential in characterizing the notion of *stability*. Stability in Two-Sided Matching is defined as the absence of blocking pairs in the allocation. As mentioned earlier, Gale and Shapley (1962) and Irving (1994) showed that there is at least one stable matching for a Two-Sided Matching problem, even with incomplete preferences and indifferences. Because of this, additional metrics need to be considered to quantify other aspects of the allocation.

In addition to stability, further commonly addressed properties are egalitarian matchings and fairness. Hence, the following additional matching properties are considered:¹

Number of Matched Pairs For problems with complete preferences, the algorithms considered in this article always yield the maximum number of matched pairs. In contrast, this property is lost by introducing incomplete preferences. In such cases, the number of matched pairs is used as an additional property for the matchings:

$$\text{NumPairs} = \sum_{\langle x, y \rangle} \{ \langle x, y \rangle | x \neq \emptyset \wedge y \neq \emptyset \} \quad (1)$$

Average Matched Rank As a measure of the average “satisfaction” of participants with the resulting matching, the Average Matched Rank (AMR) is defined as the overall rank of matched partners averaged over all participants. This corresponds to finding an ‘egalitarian’ solution. In formal terms:

$$\text{AMR} = \frac{\sum_{i,j \in \langle X, Y \rangle} P_{i,j} + P_{j,i}}{n_X + n_Y} \quad (2)$$

Note that lower numbers indicate better solutions, as the most preferred alternative has rank 1.

Fairness The previous egalitarian criterion considers the average rankings of matched partners for both sides. In comparison, fairness is measured as the difference of the average matched rank between the two sides of the matching mar-

¹ The definitions of average matched rank and fairness scores are adapted from Gusfield (1987) and Iwama et al. (2010).

ket. A higher fairness score reflects that participants of one side are, on average, matched to partners with a better rank than participants of the other side, whereas scores around 0 reflect a more equal distribution. Formally:

$$\text{Fairness} = \left| \frac{\sum_{i,j \in \langle X, Y \rangle} P_{i,j}}{n_X} - \frac{\sum_{i,j \in \langle X, Y \rangle} P_{j,i}}{n_Y} \right| \quad (3)$$

In case of incomplete preferences, finding the stable matching with the largest number of matched pairs is the most commonly considered combination of metrics. Even though finding the AMR- or fairness-best stable solution can also be goals in this case, the application of the AMR and fairness metrics to the incomplete preferences case requires the specification how unmatched participants are handled for the calculation of metrics. There is no standard in the literature how this case is handled, and most approximation algorithms focus on the mentioned combination of stability and matched pairs. In the subsequent evaluation, the AMR and fairness metrics will consider the matched participants only.

4 Stable Matching Algorithms

4.1 Complete Preferences

Algorithms found in the literature concentrate on finding stable matchings under certain conditions. For strict preferences, the Deferred Acceptance (DA) algorithm by Gale and Shapley (1962) can be used as it always yields a stable allocation. Additionally, in this case the AMR-optimal (AMRO) algorithm by Irving et al. (1987) yields the AMR-best stable solution in polynomial time, and for finding the most balanced (fair) solution the approximation algorithm by Iwama et al. (2010) can be used (henceforth called fairness-equal, FE).

In the case of indifferences, even the case of complete preferences becomes challenging. As a first step, a tie-breaking rule has to be applied in order to run the previously mentioned algorithms. The tie-breaking rule greatly affects the properties of the resulting matching and, in general, tie-breaking and applying the algorithms does not guarantee a solution with good properties. Therefore, Erdil and Ergin (2017) suggest two alternatives: the Worker-Optimal-Stable-Matching (WOSMA) which considers Pareto-improvements for one side (in their case called workers), and Efficient-Stable-Matching-Algorithm (ESMA) which considers Pareto-improvements of both sides. The former will be used in the case of incomplete preferences, as the focus of this scenario is the number of matched pairs, not the average matched rank.

4.2 Incomplete Preferences

For incomplete preferences, the common goal of Two-Sided Matching is to find a stable solution with the maximum number of matched pairs. Integer Programming

(IP) formulations have been suggested by several authors to calculate matchings with the largest number of matched pairs for given scenarios (Iwama et al. 2014; Kwanashie and Manlove 2014). The IP approach will be used to calculate an optimal matching with respect to the number of matched pairs objective, and can be used as a baseline to compare the matchings from other suggested algorithms.

Several approximation algorithms have been developed to find stable matchings with a large number of matched pairs.² For the evaluation in this article, approximation algorithms with the best runtime and guarantees on the number of matched pairs are used for the comparison with the proposed heuristics. As the trivial approximation ratio in this case is 2, i.e., the worst case number of matched pairs is half the size of the optimal solution, any approximation ratio smaller than 2 is desirable.

- *Shift* Halldórsson et al. (2007) describe an approximation algorithm for this case, in the following abbreviated as *Shift*. For certain preference structures, this algorithm provides non-trivial quality bounds for finding the stable match of maximum size. Shift operates through breaking indifferences in a systematic manner and applying the DA on the resulting set of strict preferences. In particular, if indifferences occur on both sides of the market, Shift guarantees non-trivial quality bounds if the length of indifferences is at most 2.
- Király Király (2011) presents a linear-time algorithm with a $5/3$ approximation ratio if ties are allowed on both sides and $3/2$ for one-sided ties.
- *McDermid* McDermid (2009) presents an algorithm with a $3/2$ approximation ratio, which is the best known approximation ratio for the general case without restrictions on tie lengths.
- *GSM* Paluch (2014) presents another algorithm with a $3/2$ approximation ratio and additionally a runtime improvement compared to the algorithm by McDermid (2009).

These algorithms are used in the subsequent comparison. If ties are only one-sided, Iwama et al. (2014) provide an algorithm that guarantees an approximation ratio of $25/17$. Irving and Manlove (2008) present algorithms to approximate stable marriage and hospital/residents problems. As the former puts a considerable restriction on the preferences, and the latter has a worse approximation ratio than McDermid, Király, and GSM, they are not considered in the evaluation.

4.3 Matching Heuristics

In addition to the mentioned algorithms, the use of heuristics to find solutions to the matching problem is considered as well. Two different heuristics have been adapted for the Two-Sided Matching scenario: a GA (Goldberg 1989) as an example of a evolutionary metaheuristic, and a Threshold Accepting algorithm as an example of a local search heuristic. In general, heuristics can be used to find (stable) allocations

² Note that approximation algorithms exist for this case, yet not for the case when AMR or fairness is optimized simultaneously with stability and the number of matched pairs.

from random initial assignments (see e.g. Aldershof and Carducci 1999), or to improve an initial stable matching by trying to retain stability and increasing other matching properties. This article focuses on the latter case: starting with (efficiently calculated) stable solutions, specific properties will be improved by the subsequent adjustments. This allows for the generation of additional potential matchings and an exploration of the solution space of potential allocations.

In this article, the first class of heuristics are Genetic Algorithms (GA) (Goldberg 1989; Holland 1992), which have been found to perform well in case of large search spaces (Kimbrough and Kuo 2010; Haas et al. 2013). The GA uses a population of chromosomes, each of which represents a solution to the matching problem (i.e., each chromosome describes a $\langle X, Y \rangle$). A chromosome consists of several genes, where each gene encodes a provider-requester match $\langle x, y \rangle$ of the solution. In other words, when a solution has m matches, the chromosome has m genes, and each gene consists of two identifiers, one for the provider, one for the requester. To determine the performance of a given chromosome, a fitness function is used. The performance metrics described in the previous section, or a combination thereof, are commonly used as fitness functions. In order to improve the fitness of the solutions, two genetic operators specifically adapted for Two-Sided Matching are applied in order to derive new, potentially better-performing solutions. The cycle crossover operator (Goldberg 1989), creates new potential solutions by combining two parent solutions. Specifically, the cycle crossover operator selects a sequence of two or more genes (pairs of participants $\langle x, y \rangle$) and switches either x or y for each pair in the sequence. This ensures that each participant remains matched to at most one other participant. The mutation operator, given a certain mutation probability, depends on the type of preferences. For complete preferences, it randomly selects two genes (matched pairs) of a given chromosome and exchanges either the requester or provider identifiers to create a new chromosome. For incomplete preferences, the operator tries to find Pareto-improvement cycles similar to the ones suggested by Erdil and Ergin (2017) to increase the number of matched participants. The population is evolved using these operators over a given number of rounds.

Following examples show the conceptual idea behind the cycle crossover and mutation operators. For the cycle crossover, two chromosomes C_1 and C_2 are selected and then adjusted to C_1^* and C_2^* by switching the y participants of the third and fourth pair. This is an example of a cycle of two pairs, but longer cycles are considered as well.

$$\begin{aligned} C_1 : (\langle x_2, y_1 \rangle, \langle x_1, y_3 \rangle, \langle x_3, y_2 \rangle, \langle x_4, y_5 \rangle) \quad C_2 : (\langle x_2, y_3 \rangle, \langle x_1, y_1 \rangle, \langle x_3, y_5 \rangle, \langle x_4, y_2 \rangle) \\ C_1^* : (\langle x_2, y_1 \rangle, \langle x_1, y_3 \rangle, \langle x_3, y_5 \rangle, \langle x_4, y_2 \rangle) \quad C_2^* : (\langle x_2, y_3 \rangle, \langle x_1, y_1 \rangle, \langle x_3, y_2 \rangle, \langle x_4, y_5 \rangle) \end{aligned}$$

For the mutation operator in the complete preferences case, a single chromosome is altered by switching the y participants of two pairs.

$$(\langle x_2, y_1 \rangle, \langle x_1, y_3 \rangle) \rightarrow (\langle x_2, y_3 \rangle, \langle x_1, y_1 \rangle)$$

Finally, in the incomplete preference case the mutation operator focuses on reducing the number of unmatched participants similar to the Pareto-improvement cycles

proposed by Erdil and Ergin (2017). In the following example, such a cycle can increase the number of matched pairs by one through matching an additional pair of participants.

$$(\langle \emptyset, y_4 \rangle, \langle x_1, y_3 \rangle), \langle x_3, \emptyset \rangle \rightarrow (\langle x_1, y_4 \rangle, \langle x_3, y_3 \rangle)$$

For the Two-Sided Matching application, the implemented GA focuses solely on stable allocations. Specifically, additional checks are included as part of the crossover and mutation operators that identify potential blocking pairs introduced by the adjustments, and only consider adjustments to be valid if they do not result in blocking pairs. This ensures that only valid, stable allocations are returned by the GA. The pseudocode for the adapted GA based on Goldberg (1989); Holland (1992); Haas (2014) is described in Algorithm 1.

Algorithm 1: Pseudocode of Genetic Algorithm adapted for Two-Sided Matching

Data: Preference Profiles: list of $\langle x, y \rangle$ pairs
Data: Parameters: Probabilities for cycle crossover and mutation, number of evolution rounds, size of population
Result: Match μ
begin
 Initial population $\leftarrow \emptyset$;
 for $i \leftarrow 1$ **to** *population size* **do**
 Generate initial solution based on a Two-Sided Matching algorithm (e.g., DA);
 Add initial solution to initial population;
 for $i \leftarrow 1$ **to** *number evolution rounds* **do**
 Temporary population \leftarrow population;
 Select pairs of solutions based on crossover probability and perform cycle crossover operation;
 for $\text{solution} \in \text{temporary population}$ **do**
 Mutate solution according to mutation probability;
 Population \leftarrow select best solutions from temporary population;
 $\mu \leftarrow$ best solution of population;

The second class of heuristics studied in this article are Threshold Accepting (TA) algorithms (Dueck and Scheuer 1990). TAs are an example of local search heuristics, where a given starting solution is improved by sequentially adjusting the solution and accepting adjustments within a certain threshold. Similar to the GA implementation, custom checks for blocking pairs only consider adjustments valid if they do not introduce blocking pairs to the matching. TAs are conceptually similar to Simulated Annealing approaches and have been successfully applied in many optimization problems. Depending on the definition of the solution adjustment per step, TAs try to improve a given solution and hence are suitable for finding (especially local) improvements. Compared to GAs which work on a population of different solutions in the solution space, the performance of a TA depends on the properties of the starting solution. However, it is more flexible on incrementally improving this solution than the GA. This is why its applicability in the case of Two-Sided

Matching is considered in the subsequent evaluation. Examples for adjustments are switching participants between pairs or running Pareto improvement cycles as described for the GA mutation operator.

The general procedure for TAs is as follows: given a starting allocation, a set of thresholds is defined. For each of these thresholds, a certain number of adjustments to the allocation are sequentially performed, e.g., changing a number of matched pairs. An adjustment is accepted as new (temporary) solution if it does not decrease the properties/metrics of the solution (compared to the current solution) by more than the threshold. Thresholds are reduced over time such that convergence to a (local) optimum becomes more likely, whereas the initial thresholds are set to avoid being stuck in a local optimum too soon. The pseudocode for the adapted TA based on Dueck and Scheuer (1990); Haas (2014) is described in Algorithm 2.

Algorithm 2: Pseudocode of Threshold Accepting Algorithm adapted for Two-Sided Matching

Data: Preference Profiles: list of $\langle x, y \rangle$ pairs
Data: Parameters: Number of thresholds, size of thresholds, number of repetitions per threshold
Result: Match μ

```

begin
   $\mu_{current} \leftarrow$  Generate initial solution based on a Two-Sided Matching
    algorithm (e.g., DA, GA, etc.);
  for  $i \leftarrow 1$  to number of thresholds do
    for  $j \leftarrow 1$  to number of repetitions per threshold do
       $\mu_{adj} \leftarrow$  apply adjustment to current solution  $\mu_{current}$  (e.g.,
        switching participants, Pareto-improvement cycles);
      if  $properties(\mu_{adj}) + current\ threshold \geq properties(\mu_{current})$ 
        then
           $\mu_{current} \leftarrow \mu_{adj}$ ;
     $\mu \leftarrow \mu_{current}$ ;

```

As GAs tend to sample (especially large) search spaces better than local search heuristics, the combination of the two approaches is also considered. In this case, the GA is used to find a good starting allocation for the TA, which then tries to further improve this solution. This algorithm, abbreviated GATA, aims to combine the advantages of GA and TA.

5 Simulation

To evaluate the allocations found by the heuristics in comparison to existing allocation mechanisms, a simulation-based evaluation approach is used. In all subsequently described simulation scenarios, randomly created sets of preferences for the participants are used, and the process of preference creation is similar to Gent and Prosser (2002) and Gelain et al. (2013). The scenarios consider 10–500 participants per side (i.e., 20–1000 participants in total). Each of the simulation

Table 1 Simulation input parameters

Parameter	Range	Description
n_X	{10, 20, 50, 100, 200, 500}	Number of participants X on side 1
n_Y	{10, 20, 50, 100, 200, 500}	Number of participants Y on side 2
l	{1.0, 0.5, 0.3, 0.1, 0.05}	Relative length of preference lists
ψ	{2, 5, 10}	Maximum length of ties
ξ	{25, 100}	Percentage of participants j that are correlated in the preferences of participants i

scenarios was independently repeated 100 times with different preference profiles in each repetition, and the results refer to the averages of these runs.

To obtain robust results and study their dependency on certain input parameters, a systematic simulation plan is used which specifies the simulation scenarios with the respective input parameter settings. Table 1 shows an overview of the most important input parameters and the ranges of values that are used.

The focus of this evaluation is preferences with indifferences, both in the complete (Stable Matching with Ties, SMT) and incomplete (Stable Matching with Ties and Incompleteness, SMTI) preference setting. Depending on the scenario, preferences can be either complete or incomplete, and are either random or correlated. The numbers of participants per side, n_X and n_Y , determine the size of the problem instance. Usually, symmetric problems with $n_X = n_Y$ are considered in the literature. This evaluation, however, also considers unequally sized problem instances. The remaining parameters are concerned with the structure of the preferences. l determines the average expected number of ranked participants in the preference lists and is used during preference creation. For example, $l = 0.3$ defines that on average, 30% of participants of the other side are included in a participant's preference list. ψ specifies the maximum length of the ties in the preference lists. Furthermore, ξ defines if and to what degree the preferences are correlated. For a given value of ξ , the participants of the respective other side are split into two groups, and ξ determines what percentage of participants are in the first group. This grouping of participants is the same for all participants of the side, yet the subsequent randomization leads to randomized rankings within the two groups. For each scenario, the three evaluation metrics as described in Sect. 3.1 are recorded.

For the implementation of the GA, a population of 50 chromosomes, a crossover probability of 0.6, and a mutation probability of 0.2 per chromosome were used. For complete preferences, the GA uses DA, AMRO, and FE to create initial (stable) solutions. In case of incomplete preferences, two versions of the GA are considered. The baseline version uses the DA with randomized tie breaking to get the initial population. The mixed version uses the DA as well, yet one matching from each of the approximation algorithms is added to increase the diversity of the initial starting matchings (the total population size remains 50). In total, 1000 evolution rounds are calculated after which the fittest chromosome is used for evaluation.

The TA uses the thresholds $\{0.04 * n, 0.02 * n, 0.01 * n, 0\}$, where $n = \min(n_X, n_Y)$. For each threshold, 10,000 perturbations of the current solution are computed sequentially, and if the perturbation is within the threshold of the current solution, the perturbation is set as the new current solution. Perturbations in this case are the swapping of participant-ID's in a certain number of matched pairs.

6 Comparing Solution Properties for Complete Preferences with Indifferences

This section compares the properties of solutions calculated by the best existing mechanisms for certain objective functions to the properties of solutions found by the heuristics (GA, TA, or combination of both). Only strictly stable solutions are considered, i.e. the stability criterion is not mentioned separately. All simulations are run with random preferences and different lengths of indifference groups, as well as different number of participants.

If the preference profiles of participants are complete yet contain indifferences, the same algorithms as in the case of strict preferences can be applied by breaking the ties first. However, as mentioned before, applying the algorithms that are optimal in the case of strict preferences does not guarantee finding the best solution with respect to a certain property anymore. Specifically, finding an AMR- or fairness-optimal stable solution is NP-hard. As the considered algorithms always yield a stable solution of maximum size in this case, a common goal is to find a stable solution with good AMR or fairness characteristics. Without loss of generality, the subsequent analysis focuses on the AMR and fairness properties of stable matchings found by the different mechanisms, respectively, without taking other potential properties into consideration.

Considering GA and GATA heuristics, the subscript “*.DA*” in the subsequent evaluations indicates that they are initialized only with DA solutions, whereas “*.MIXED*” means that a randomly created mixture of DA, AMRO, and FE solutions are used. Both AMRO (AMR-optimal) and FE (fairness-equal) were developed for symmetric problem sizes, and some of their routines in calculating the solution do not easily extend to non-symmetric settings. Hence, asymmetric problem sizes are not considered in this evaluation.³

Unless stated otherwise, the statistical tests refer to non-parametric Wilcoxon signed-rank tests with Bonferroni p value adjustment in order to account for multiple comparisons. The test is used due to the simulation design (algorithms have the same preference lists as input, leading to a paired design), as well as non-normally distributed data. The following figures refer to the average values over the 100 independent repetitions. As DA, AMRO, and FE depend on a (random) tie breaking to run properly, the results visualize the average of 100 repetitions, i.e., 100 random tie breakings.

³ This again shows an advantage of having flexible heuristics to calculate matching solutions: they can easily be applied to non-symmetric settings as well.

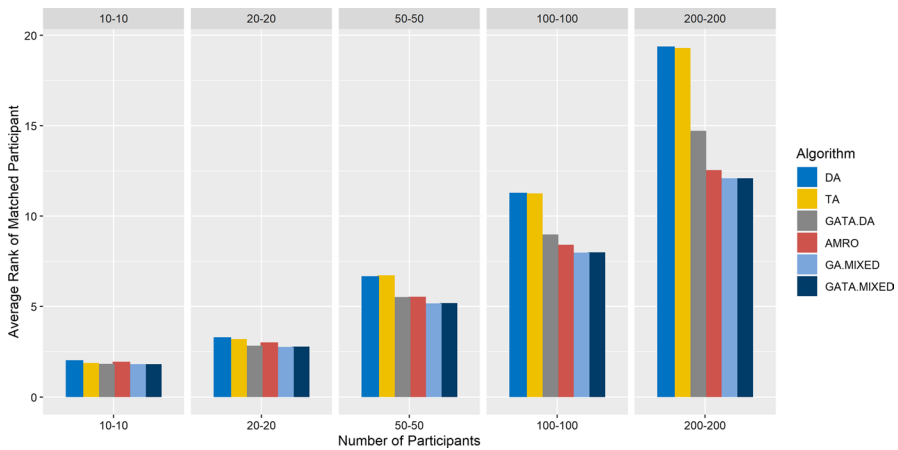


Fig. 1 Comparison of AMR properties for complete preferences

6.1 Optimizing Stability and Average Matched Rank

When focusing on the properties of stability and average matched rank, the aim is to find stable matchings with good average matched rank properties, i.e., where the average rank of the matched participants is as close to the respective most preferred option as possible. For complete preferences with indifferences, this is an NP-hard problem (and even hard to approximate), which means that finding a good solution is far from trivial (Halldórsson et al. 2003). Hence, heuristics such as GA and TA might be able to yield allocations with better AMR properties than applying the algorithms developed for complete preferences without indifferences. For GA, TA, and GATA, as the implicit checks for blocking pairs ensures that only stable matchings are generated. Hence, the objective function in this case is the minimization of the average matched rank of participants as defined in Eq. (2).

6.1.1 Uncorrelated Preferences

Figure 1 shows the results for the considered algorithms and problem sizes between 10×10 and 200×200 participants, averaged over different tie-lengths. First, the results show that the average matched rank becomes worse with an increasing number of participants (indicated by an increasing score). In the given uncorrelated case this is an interesting result, as it implies that even for independent preferences, stable solutions in larger systems tend to consist of pairs where participants are matched to lower ranked options.

Considering the properties of matchings from different algorithms, GA.MIXED and GATA.MIXED on average provide allocations with the best AMR properties. They are able to significantly decrease the average matched rank properties from allocations calculated by the DA and AMRO mechanisms (statistically significant at the level $p < 0.001$). This indicates that both GA and GATA are able to improve

upon AMRO matchings with respect to the average matched rank property, as the MIXED versions use AMRO solutions as part of their initial solution population. Additionally, the fairness properties of the GA and GATA matchings are, on average, slightly improved compared to the AMRO (average and best) matchings.

Considering the GATA matching properties with different initial solutions, the results show that GATA.DA is able to find matchings with significantly decreased average matched rank compared to DA matchings, and also yields decreased AMR compared to AMRO matchings for problem sizes up to 50×50 . For a larger number of participants, AMRO generally yields matchings with a lower (better) average matched rank, but the GATA.DA still finds matchings with improved AMR properties compared to DA matchings to a significant degree ($p < 0.001$). Both GA.MIXED and GATA.MIXED are able to improve upon their respective initial matchings as well. Two observations can be derived from this: first, the GA effectively improves the given starting matchings with respect to average matched rank, showing its usefulness for finding allocations with improved properties for a specific goal. Second, the initial performance of the starting solutions with respect to a given property such as average matched rank determine the improvement that can be achieved by the heuristics, which means that feeding the heuristics with promising initial matchings leads to an improvement with respect to the specified property.

Interestingly, while GA and GATA perform very well, TA does not seem to be an adequate heuristic for this scenario. In the given setting, TA starts with a DA matching and then tries to find an improved matching with respect to a specific property as specified earlier. Figure 1 shows that only for small problem instances the TA is able to significantly improve the average matched rank property of the starting allocation. For larger problem instances, TA finds matchings that are considerably worse compared to AMRO and GA/GATA. This indicates that starting with DA matchings is not promising for TAs as it might get stuck in local optima. In contrast, the GA with its ability to sample large search spaces is an adequate heuristic for complete preferences with indifferences. Due to this, the performance of GA.MIXED and GATA.MIXED are basically the same as TA is not able to find matchings that significantly improve its initial GA matching.

6.1.2 Effect of Preference Correlation

To study the effect of preference correlation, preferences with a correlation factor $\xi = 25$ are considered next. The factor $\xi = 25$ implies that the set of participant IDs is split into two sets of relative size 25% and 75%, and the 25% highest ranked participants in all the preference profiles (of each side) are drawn from the same set. This creates a set of highly desired and less desired participants, increasing the (relative) competition for the highly desired group.

Considering the average matched rank score of the matched participants, Fig. 2 compares the properties of matchings found by DA, AMRO, GA.MIXED, TA, GATA.DA, and GATA.MIXED. The results are qualitatively very similar to the case of uncorrelated preferences. On one hand, the results show that the average matched rank decreases compared to the uncorrelated case, which is not surprising as the correlation means that some participants are more likely to be matched with

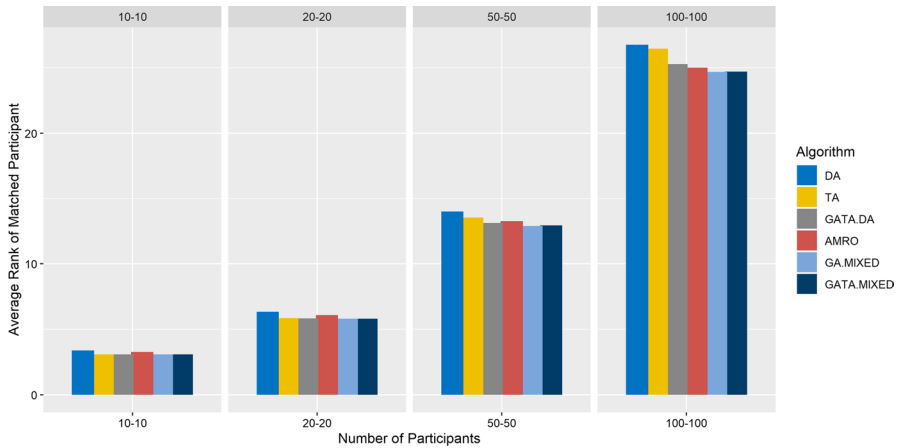


Fig. 2 Comparison of AMR performance for complete and correlated preferences

lower-ranking participants as there is an increased competition for higher-ranked participants. On the other hand, the relative ranking of the matchings found by different algorithms with respect to their average AMR scores is practically the same. GA.MIXED and GATA.MIXED on average find matchings with improved AMR properties than the other algorithms, and only AMRO solutions yield comparable results (for all considered scenarios, the signed-rank test reveals a statistical difference between the AMRO solutions and GA.MIXED/GATA.MIXED at $p < 0.001$, although the absolute difference is negligible).

The results above are also robust with respect to different values for maximum tie length, as a follow-up sensitivity analysis shows. In fact, GA.MIXED and GATA.MIXED matchings have particularly improved AMR properties compared to the AMRO mechanism in case of small tie lengths.

6.2 Optimizing Stability and Fairness

A second common goal for complete preferences is to find a stable and particularly fair matching. However, even for strict preferences this is an NP-hard problem. The approximation algorithm FE can be applied for complete preferences with indifferences by breaking ties, yet the fairness properties of the found solutions in this case is unclear. Hence, this part of the evaluation considers the performance of DA, FE, GA, TA, and GATA algorithms for finding stable and fair matchings. For GA, TA, and GATA, the objective of fairness maximization is used with the fairness definition from Eq. (3). In combination with internal validity checks for blocking pairs this again yields completely stable solutions for the given scenarios.

6.2.1 Uncorrelated Preferences

Figure 3 presents the fairness properties of solution found by the considered algorithms. As expected from theory, the DA calculates matchings with a high difference

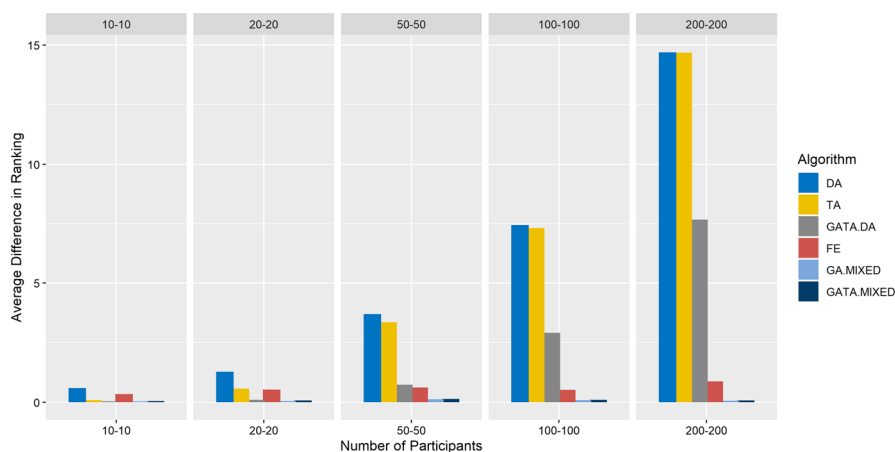


Fig. 3 Comparison of fairness performance for complete preferences

between the two sides as it yields the 'best' stable solution for one side and the 'worst' stable solution for the other side. Furthermore, similar to the case of AMR optimization, TA finds improvements (in this case: improvements on the fairness property) only for small problem instances. For larger instances the matchings found by TA have higher average differences in ranking than solutions from other (non-DA) algorithms.

Comparing the fairness properties of matchings found by DA, FE, GA, and GATA, the results indicate that GA.MIXED and GATA.MIXED yield matchings with significantly better fairness properties, and also (slightly) improve upon average FE solutions. This is confirmed by Wilcoxon signed-rank tests at the level $p < 0.001$. Over all considered scenarios, the signed-rank test indicates a significant difference between the best FE solution and the GA solution with mixed initial solutions, although the practical effect size is negligible. The performance of GATA.DA, initialized with DA solutions, significantly improves the fairness of the initial DA matchings (at $p < 0.001$). However, the overall fairness of matchings from GATA.DA is still worse than the FE solution (with the exception of very small problem sizes).

6.2.2 Effect of Preference Correlation

Whereas preference correlation can be expected to have a clear effect on the average matched rank score, the effects on fairness are less clear. Figure 4 shows the resulting fairness scores for preferences which are correlated with $\xi = 25$. The overall results are qualitatively similar to the uncorrelated scenario. GA.MIXED and GATA.MIXED, which use fairness optimization for stable matchings, consistently finds almost perfectly fair matchings for the studied scenarios. The solutions from GA.MIXED and GATA.MIXED have significantly better fairness properties than even the FE approach, confirmed by Wilcoxon signed-rank tests at the level $p = 0.01$. Interestingly, the TA performs slightly better for correlated preferences. In

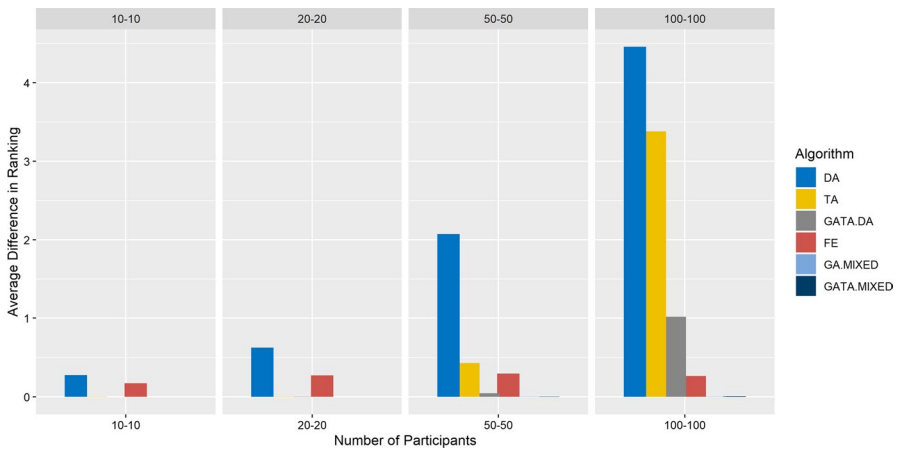


Fig. 4 Comparison of fairness performance for complete and correlated preferences

this case, it is able to find comparably fair matchings for problem sizes up to 50×50 participants.

Overall, considering their performance with respect to average matched rank and fairness, matchings found by GA.MIXED and GATA.MIXED seem to have better AMR/fairness properties compared to matchings found by other approaches in the case of complete preferences with indifferences.

7 Comparing Solution Properties for Incomplete Preferences With Indifferences

Building on the previous evaluation of complete preferences, the most general case of preference profiles includes both incompleteness and indifferences (SMTI scenario). In this case, a common goal of matching algorithms is to find a stable solution of maximum size, i.e., that matches as many participants as possible. As described in Sect. 4, there are several approximation algorithms and heuristics specifically developed for this case due to the underlying optimization problem being NP-hard. This part of the evaluation studies how matchings calculated by GA, TA, and GATA heuristics compare against matchings from the specialized algorithms in different scenarios. For statistical evaluation, Wilcoxon signed-rank tests with Bonferroni adjustments are again used due to the non-normality of the results.

7.1 Comparing Matching Properties

As introduced in Sect. 4.2, a commonly used property for evaluating matchings in the case of incomplete preferences with indifferences is the number of matched participants. The following figures represent the average values based on 100 independent repetitions for each scenario. Similar to the previous section, for GATA the suffix ".MIXED" indicates that initial solutions are based on DA matchings plus

an additional matching from applying Király, McDermid, and GSModified, respectively. If no suffix is provided, GA and GATA are initialized only with DA matchings. Considering the optimization function, GA, TA, and GATA use a weighted function that emphasizes the number of matched pairs as main goal, and also tries to optimize AMR as secondary goal. The stability of the matching is again checked internally while creating new matchings. More specifically, the goal is to minimize an objective function of the form

$$\min_{(x,y)} Z = \left(\text{Max}(n_x, n_y) \right)^2 * \left(\text{Min}(n_x, n_y) - \text{NumPairs} \right) + \text{AMR} \quad (4)$$

This ensures that the number of matched pairs is the dominant objective, yet still allows for the simultaneous optimization of AMR. For the evaluation and comparison with solutions found by the approximation algorithms, such an optimization function represents a pessimistic comparison as the AMR scores tend to increase with a larger number of matched pairs, making AMR and matched pairs substitute optimization goals.

The matched pairs property of the matchings can be measured in two ways in this case. First, the percentage of matched participants can be considered to rank different algorithms with respect to their ability to found favorable matchings. Depending on the preferences, however, it cannot be guaranteed that there is a stable solution where all participants are matched. This means that for specific preferences, the maximum achievable percentage of matched participants can be smaller than 100%. Hence, the second way of defining an optimal matching property is relative to the theoretically optimal solution as an indicator how close the algorithms are to the optimal outcome. As finding the optimal outcome is NP-hard, the best matching for the scenarios was calculated only for small problem instances up to 100 participants on each side. These optimal matchings and their number of matched pairs were then used as a baseline to compare the matchings of the approximation algorithms and heuristics.

Table 2 displays the results for the first matching property, the percentage of matched participants. Averaged over the different parameter values for tie length and length of preference lists (refer to Table 1), the results show that with an increasing number of participants the percentage of matched participants in the matchings increases as well. Especially for large problem sizes, the majority of participants are matched. However, a more detailed analysis in Sect. 7.2 shows that this seems to depend on the lengths of the preference lists. Comparing the matchings from different algorithms and problem sizes, considerable differences can be observed. DA yields matchings with the smallest number of matched participants on average, which is not surprising as it was not specifically developed for this scenario. Applying the WOSMA improvement cycle on the DA matching leads to a significantly increase in the number of matched pairs. Considering the approximation algorithms described earlier, matchings from the Király algorithm yield the largest number of matched pairs on average, while GSModified, McDermid, and Shift match slightly fewer participants. Considering the Shift algorithm, extended results show that it find matchings with a large number of matched pairs in case the maximum length of ties is 2, which is the core scenario for which it was developed. However, the Király

Table 2 Percentage of matched participants based on problem size

Size	DA (%)	GSM. (%)	McDer. (%)	WOSM. (%)	Shift (%)	TA (%)	GATA (%)	GA (%)	Kiraly (%)	GATA-MIX (%)
10 × 10	66.37	69.16	69.41	69.48	70.03	70.23	70.11	70.16	70.17	70.49
20 × 20	78.08	80.91	81.00	81.12	81.57	81.54	81.77	81.90	81.89	82.25
50 × 50	90.64	93.06	93.62	93.59	93.04	94.05	94.33	94.33	94.33	94.65
100 × 100	95.92	97.40	97.84	97.72	97.22	98.22	98.33	98.25	98.28	98.46
200 × 200	98.65	99.14	99.30	99.24	99.17	99.45	99.56	99.55	99.53	99.61
500 × 500	99.82	99.87	99.90	99.90	99.93	99.91	99.96	99.96	99.95	99.97

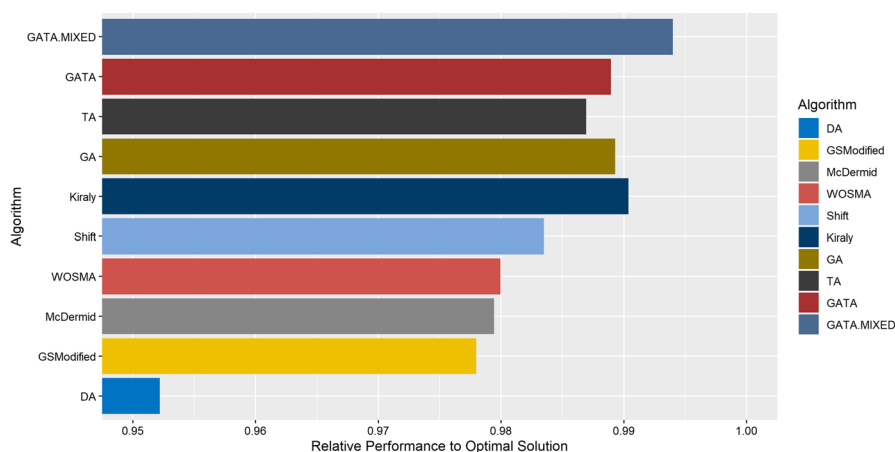


Fig. 5 Algorithm comparison relative to optimum, uncorrelated preferences, 10×10 to 100×100 participants

algorithm usually finds better matchings with respect to percentage of matched participants, and for larger tie lengths the performance of Shift decreases considerably.

Both the GA and TA perform very well, yielding matchings with only slightly smaller percentage of matched pairs compared to Király matchings. This indicates that in contrast to complete preferences settings, TA is a useful heuristic in SMTI scenarios. Additionally, the results for GATA and GATA.MIXED show that instantiating the GATA with matchings from approximation algorithms can significantly increase the percentage of matched pairs in GATA solutions. In particular, matchings found by GATA.MIXED on average yield improved results with respect to the percentage of matched participants compared to even the best approximation algorithm in the considered scenarios (Király). The statistical analysis with a Wilcoxon signed-rank test for a difference in percentage of matched participants reveals that there is no significant difference between GA, GATA, and Király matchings, and that GA, GATA, and Király matchings have a higher percentage of matched participants than TA, Shift, McDermid, GSModified, as well as WOSMA matchings ($p < 0.001$). Furthermore, matchings from GATA.MIXED significantly improve upon GA, GATA, and Király matchings for the given objective ($p < 0.001$).

Considering the relative properties of the matchings found by the considered algorithms with respect to the optimum solution calculated with an Integer Program formulation, Fig. 5 shows the relative ranking of average matchings found by the algorithms. The graph shows that for all considered scenarios the algorithms are within 95% of the optimal solution, i.e., are able to find relatively large matches which are close to the optimal solution.

Table 3 presents several statistics for the relative performance of the algorithms' solutions compared to the optimal solution. Interestingly, the heuristics not only perform well on average, they also find matchings with the optimal percentage of matched participants in more cases than the approximation algorithms. For example, GATA.MIXED successfully finds matchings with the optimal percentage of

Table 3 Matching properties relative to optimal solution, incomplete preferences, 10–100 users per side

Statistic	DA	GS-Mod.	Mc-Der.	WOSM.	Shift	TA	GATA	GA	Király	GATA-MIX
Mean	0.952	0.980	0.983	0.983	0.984	0.989	0.991	0.991	0.991	0.995
Median	0.965	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
S.D.	0.051	0.032	0.030	0.031	0.029	0.023	0.022	0.020	0.019	0.014
Min	0.500	0.778	0.667	0.667	0.750	0.750	0.667	0.750	0.857	0.857
Max	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Optimal (%)	0.228	0.619	0.660	0.669	0.684	0.731	0.763	0.759	0.754	0.813

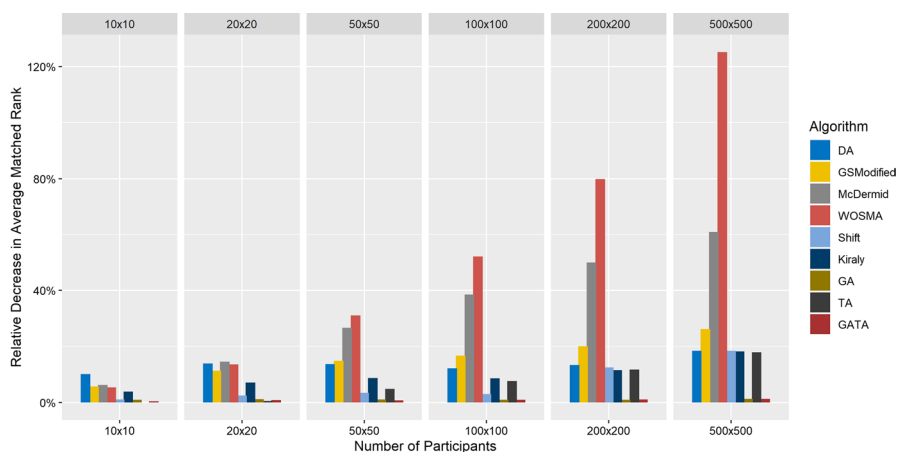


Fig. 6 Relative decrease in average matched rank (AMR) compared to GATA.MIXED

matched participants in 81.3% of the repetitions, whereas the other algorithms have success rates of about 61–75%. Furthermore, the standard deviation for the heuristics tends to be lower as well. As approximation algorithms provide a guaranteed quality bound, i.e., worst case performance for the size of the matching, Table 3 also shows that the worst-case performance is best for Király and GATA.MIXED.⁴

Overall, similar to the previous findings about the percentage of matched participants, the consideration of average and worst-case performance relative to the optimal solution shows that GATA.MIXED not only yields matchings with the highest percentage of matched participants on average, yet also the matchings with the best worst-case performance. This is particularly interesting as it indicates that for the considered scenarios, the practical quality bounds of GA, GATA, and especially GATA.MIXED are comparable (or even better) to the quality bounds of the approximation algorithms. However, as heuristics are not able to provide definite quality bounds, there might be scenarios where the worst-case performance is lower than that of the approximation algorithms.

In addition to comparing the number of matched pairs, it is also worthwhile to consider AMR and fairness properties of the matchings. Even though the approximation algorithms are not specifically developed for this combination of matching properties, the additional properties are relevant for the involved participants. As GATA.MIXED matchings have the best (lowest) average matched rank scores, Fig. 6 presents the average matched rank performance relative to GATA.MIXED. For example, a value of 50% indicates that the average matched rank score of matchings found by an algorithm was 50% larger than the AMR of average GATA.MIXED matchings. The results show that not only does GATA.MIXED yield matchings with a larger number of matched pairs, the corresponding matchings also have AMR

⁴ Note that the worst-case performance of 0.66 for McDermid is exactly its guaranteed $3/2$ approximation factor.

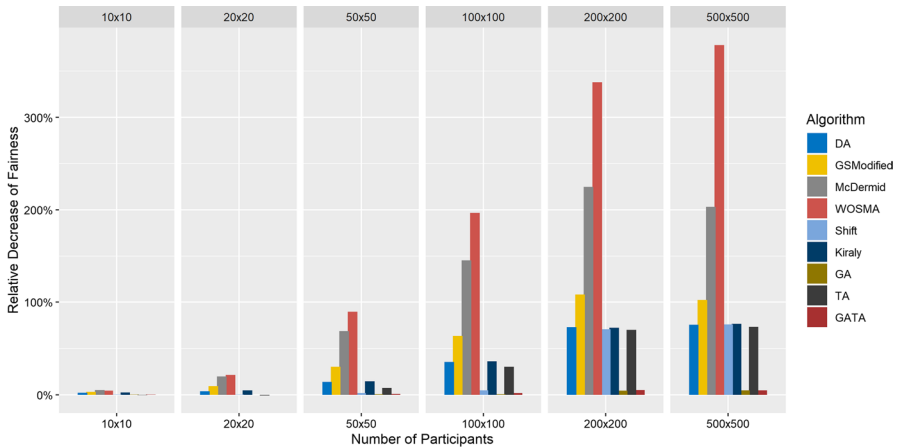


Fig. 7 Relative decrease in fairness compared to GATA.MIXED

properties that are considerably better than the AMR scores of matchings from approximation algorithms.⁵

The results for the fairness metric are similar with a slightly larger magnitude, as shown in Fig. 7. The relative improvements in AMR and fairness properties are particularly high for larger problem instances, which indicates that these matchings have considerably improved AMR/fairness properties. For example, compared to the best approximation algorithm Király, AMR improvements of up to 18% and fairness improvements of up to 76% can be achieved, which means that participants are on average matched to higher ranked partners, and that the matchings treat both sides more equally.

These results show that the heuristics not only find matchings with good properties for the standard combinations of goals, but also for multiple objectives such as finding a maximum size stable match with good AMR or fairness properties. This can be especially important in real-life settings (such as school choice, college admission, or mentor-mentee matching) where such average matched rank and fairness aspects are of importance, making the proposed heuristics particularly beneficial in these settings.

7.2 Effect of Preference Structures and Participant Distributions

To study the sensitivity of the results on different market parameters, this section evaluates the matching properties in case of different tie lengths, preference lengths, and correlation in preferences.

⁵ Interestingly, TA seems to find slightly lower AMR properties for small problem instances. The relative improvement, however, is small, and taking into account the AMR properties for larger problem instances, GATA.MIXED finds matchings with the best overall properties.

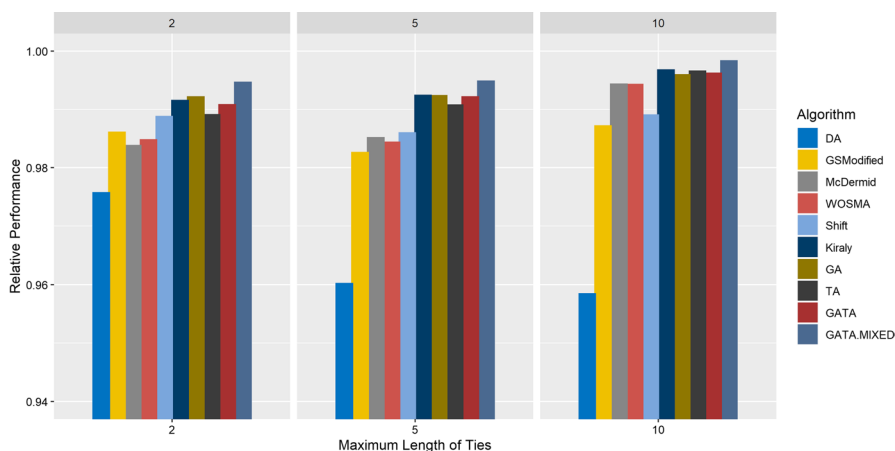


Fig. 8 Performance of matching algorithms, random preferences

7.2.1 Influence of Tie Lengths

For different maximum lengths of ties, Fig. 8 shows that, as before, GA and TA perform quite well. In most cases the matchings found by the heuristics match a larger number of participants than matchings from approximation algorithms, and only Király solutions match more participants on average. GATA.MIXED consistently finds matchings with the largest average percentage of matched participants. Overall, the relative differences between matchings become smaller with increasing tie lengths. These observations are confirmed by a statistical analysis applying Wilcoxon signed-rank tests. For small maximum tie lengths, $\psi = 2$ and $\psi = 5$, matchings found by GA and Király match significantly more participants than TA, and GATA.MIXED matchings significantly more than GA and Király ($p < 0.001$). For $\psi = 10$, the performance of the algorithms is more similar overall. TA is in fact slightly better than GA ($p = 0.001$) and yields similar matchings as Király (no statistically significant difference), and GATA.MIXED is still better than Király at the level $p = 0.001$.

7.2.2 Influence of Preference Lengths

Figure 9 shows the results for the percentage of matched participants and different average preference lengths, whereas Fig. 10 shows the performance relative to the theoretical optimum. For larger values of l , more participants will be deleted from each others' preference profiles, thereby shortening the profiles and increasing the probability that some participants might be unmatched in a solution. This is especially reflected in Fig. 9, which shows that the percentage of matched participants considerably decreases if participants have short preference lists. Per definition of parameter l , the preference length is measured proportionally to the number of participants, which means that for larger problem

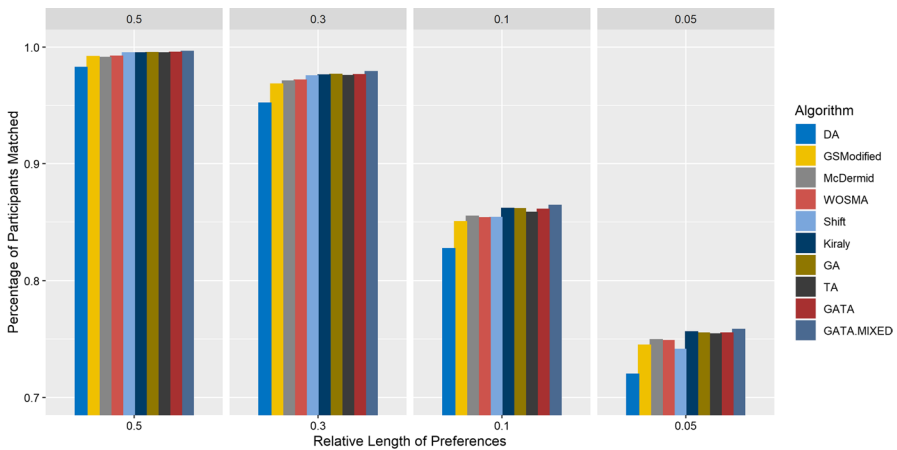


Fig. 9 Percentage matched participants for different preference lengths

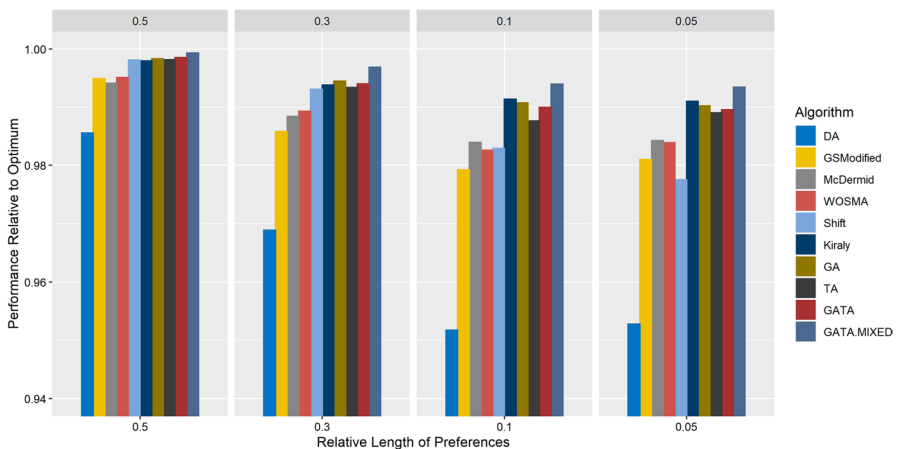


Fig. 10 Performance relative to optimum for different preference lengths

sizes, the absolute preference lengths increase for the same l . This can explain the findings shown earlier in Table 2, which showed that with increasing participant size the percentage of matched participants increases.

For smaller values of l significant differences between some of the matchings with respect to the percentage of matched participants can be found, yet they are relatively small. Especially for larger values of l , i.e., longer preference lists, it seems to be easier to find large stable matchings which makes the differences between algorithms marginal. However, in scenarios where the number of ranked participants is limited, the differences in matchings found by different algorithms are significant.

Table 4 Relative comparison to optimum for asymmetric scenarios

Size	DA (%)	GSM. (%)	McDer. (%)	WOSM. (%)	Shift (%)	Kiraly (%)	TA (%)	GA (%)	GATA (%)	GATA-MIX (%)
10 × 50	99.57	99.91	99.90	99.94	99.98	99.99	99.97	99.98	99.98	100.0
10 × 100	99.96	99.99	99.99	100.0	100.0	99.99	100.0	100.0	100.0	100.0
20 × 50	98.87	99.73	99.79	99.81	99.91	99.94	99.87	99.97	99.97	100.0
20 × 100	99.94	99.99	99.98	99.99	100.0	100.0	99.99	100.0	100.0	100.0
50 × 10	99.60	99.96	99.92	99.91	100.0	99.99	99.92	99.99	99.98	100.0
50 × 100	99.60	99.91	99.96	99.94	99.96	99.97	99.96	100.0	100.0	100.0

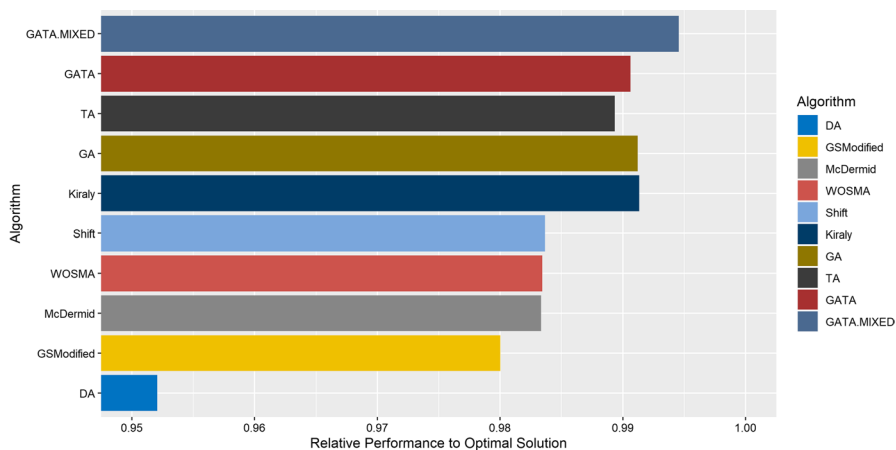


Fig. 11 Algorithm ranking with respect to relative performance to optimal solution, correlated preferences

7.2.3 Scenarios with Asymmetric Numbers of Participants

Table 4 shows a comparison of matchings from algorithms relative to the optimal solution for problems where the numbers of participants on both sides are unequal. Overall, the percentage of matched participants for matchings found by the considered algorithms is larger than in the case of symmetric problem instances. Furthermore, the more unequal the two sides with respect to number of participants, the more similar the matchings found by the different algorithms. This can be expected, as the maximum number of participants matched is bounded by the number of participants of the smaller market side, and the probability to match all such participants increases if (relatively) more participants of the larger market side are present in the market. As a result, all algorithms calculate a matching with the optimal percentage of matched participants in 99% of the cases.

Table 4 also shows that, while the increase in percentage of matched participants is comparably small, matchings found by GA, GATA, and GATA.MIXED heuristics have a slightly higher percentage of matched participants than approximation algorithms, which confirms the previous findings. In particular, GATA.MIXED is able to find an optimal matching (with respect to number of participants matched) in all the studied problem instances.

7.2.4 Correlated Preferences

For correlated preferences, Fig. 11 shows the relative performance ranking of the matchings by different algorithms averaged over all combinations of problem size, lengths of preference lists, and maximum lengths of ties. Similar to the scenario with uncorrelated preferences, the heuristics are able to find matchings with

Table 5 Matching properties relative to optimal solution, incomplete correlated preferences

Statistic	DA	GS-Mod.	Mc-Der.	WOSM.	Shift	TA	GATA	GA	Király	GATA-MIX
Mean	0.952	0.978	0.980	0.979	0.983	0.987	0.989	0.990	0.989	0.994
Median	0.962	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
S.D.	0.048	0.033	0.033	0.035	0.028	0.024	0.023	0.020	0.022	0.014
Min	0.500	0.750	0.667	0.500	0.750	0.818	0.818	0.857	0.800	0.857
Max	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Optimal (%)	0.205	0.588	0.618	0.618	0.683	0.690	0.730	0.745	0.721	0.799

good properties. As before, GATA.MIXED on average yields matchings with the largest percentage of matched participants of the considered algorithms.

Table 5 also shows that the heuristics have a low variance in the percentage of participants that are matched, and that its worst case performance is better than Király and the other approximation algorithms. Whereas the general worst-case performance seems to be similar to the case of uncorrelated preferences, the number of times where the algorithms are able to find matchings with the optimal number of matched participants is lower. Compared to the findings in Table 3, the percentage of scenarios where an optimal solution was found decreases from up to 86–80% in the best case.

8 Summary

For the general Two-Sided Matching problem with indifferences, finding average-matched-rank-optimal or fairness-optimal stable matchings (for complete preferences) or maximum-size stable matchings (for incomplete preferences) are NP-hard problems. While approximation algorithms provide lower quality bounds for properties of the calculated matchings, heuristics provide the flexibility to consider various combinations of properties and objective functions yet do not give performance guarantees. The aim of this article is to compare the properties of matchings found by heuristics in Two-Sided Matching to matchings from approximation algorithms with the best known performance guarantees.

Overall, for both complete and incomplete preferences with indifferences, matchings found by the studied heuristics have improved properties for the respective goal in most of the studied cases. For the scenario of complete preferences with indifferences, average matched rank and fairness properties are commonly considered. The evaluation showed that depending on the initial matchings, a Genetic Algorithm (potentially combined with a Threshold Accepting algorithm) improves the properties of the initial solutions and, on average, yields matchings with better properties than existing algorithms. In case of incomplete preferences, the heuristics (especially GATA with mixed initial solutions) are able to find solutions that match a significantly larger percentage of participants, on average, than even the best approximation algorithms. Furthermore, the matchings found by the heuristics also substantially improve upon additional properties such as average matched rank and fairness compared to matchings from approximation algorithms. This is a particularly important result as participants might value average matched rank and fairness aspects in addition to the standard set of optimization goals for which approximation algorithms exist (stability and number of matched pairs). A better average matched rank performance directly benefits the participants as they are, on average, matched to a higher ranked partner, and the solution can be fairer to both participating sides. This also shows that heuristics are able to use promising matchings and further improve them with respect to additional criteria, an aspect that has not often been considered in Two-Sided Matching so far.

While this article provides a systematic evaluation of matching properties found by heuristics for a wide area of scenarios, there are several directions for future

work. First, additional heuristics can be designed and evaluated to determine which types of heuristics perform best in a general Two-Sided Matching environment. Second, as indicated by the strong results of the Genetic Algorithm starting with mixed initial solutions, the combination of heuristics and approximation algorithms to find matchings with improved properties is an interesting endeavor. For example, additional analyses can be performed to find the best combination of algorithms for different goals. Third, using a Pareto-dominance sorting approach to compare the properties of the different matchings can shed additional light on the achievable trade-offs between several properties. For example, algorithm-specific trade-offs between average-matched-rank and number of matched participants can be studied in more detail. Finally, the aspect of incentive compatibility and the effects of preference manipulation in the context of using different solution algorithms is a worthwhile consideration. While preference manipulation has been studied mostly in the context of the DA algorithm, the effects on matching properties when other approximation algorithms or heuristics are applied is largely novel. As they can substantially increase properties of the resulting matching, the potential manipulation effects on participants and the overall system need to be studied in more detail.

Compliance with Ethical Standards

Conflict of interest The author declares that he has no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Abdulkadiroglu, A., & Sönmez, T. (2003). School choice: A mechanism design approach. *American Economic Review*, 93(3), 729–747.
- Abdulkadiroglu, A., & Sönmez, T. (2013). Matching markets: Theory and practice. *Advances in Economics and Econometrics*, 1, 3–47.
- Abdulkadiroglu, A., Pathak, P. A., & Roth, A. E. (2009). Strategy-proofness versus efficiency in matching with indifference: Redesigning the NYC high school match. *American Economic Review*, 99(5), 1954–1978.
- Abraham, D. J., Irving, R. W., & Manlove, D. F. (2007). Two algorithms for the student-project allocation problem. *Journal of Discrete Algorithms*, 5(1), 73–90.
- Aldershof, B., & Carducci, O. M. (1999). Stable marriage and genetic algorithms: A fertile union. *Journal of Heuristics*, 5(1), 29–46.
- Axtell, R. L., & Kimbrough, S. O. (2008). The high cost of stability in two-sided matching: How much social welfare should be sacrificed in the Pursuit of stability? In *Proceedings of the 2008 world congress on social simulation (WCSS-08)*.

- Boudreau, J. W. (2011). A note on the efficiency and fairness of decentralized matching. *Operations Research Letters*, 39(4), 231–233.
- Caton, S., Haas, C., Chard, K., Bubendorfer, K., & Rana, O. F. (2014). A social compute cloud: Allocating and sharing infrastructure resources via social networks. *IEEE Transactions on Services Computing*, 7(3), 359–372.
- Coles, P., & Shorrer, R. (2014). Optimal truncation in matching markets. *Games and Economic Behavior*, 87, 591–615.
- Diebold, F., & Bichler, M. (2017). Matching with indifferences: A comparison of algorithms in the context of course allocation. *European Journal of Operational Research*, 260(1), 268–282.
- de Haan, M., Gautier, P.A., Oosterbeek, H., & van der Klaauw, B. (2015). The performance of school assignment mechanisms in practice. In *CEPR discussion Paper No. DP10656*.
- Dueck, G., & Scheuer, T. (1990). Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, 90(1), 161–175.
- Echenique, F., Wilson, A. J., & Yariv, L. (2016). Clearinghouses for two-sided matching: An experimental study. *Quantitative Economics*, 7(2), 449–482.
- Ehlers, L. (2008). Truncation strategies in matching markets. *Mathematics of Operations Research*, 33(2), 327–335.
- Erdil, A., & Ergin, H. (2008). What's the matter with tie-breaking? Improving efficiency in school choice. *American Economic Review*, 98(3), 669–689.
- Erdil, A., & Ergin, H. (2017). Two-sided matching with indifferences. *Journal of Economic Theory*, 171, 268–292.
- Gale, D., & Shapley, L. S. (1962). College admissions and the stability of marriage. *American Mathematical Monthly*, 69(1), 9–15.
- Gelain, M., Pini, M. S., Rossi, F., Venable, K. B., & Walsh, T. (2013). Local search approaches in stable matching problems. *Algorithms*, 6(4), 591–617.
- Gent, I. P., & Prosser, P. (2002). An empirical study of the stable marriage problem with ties and incomplete lists. In *ECAI—15th European conference on artificial intelligence* (pp. 141–145).
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley Longman Publishing Co., Inc.
- Gusfield, D. (1987). Three fast algorithms for four problems in stable marriage. *SIAM Journal on Computing*, 16(1), 111–128.
- Gusfield, D., & Irving, R. W. (1989). *The stable marriage problem: Structure and algorithms*. Cambridge, MA: MIT Press.
- Haas, C. (2014). *Incentives and two-sided matching: Engineering coordination mechanisms for social clouds*. Karlsruhe: KIT Scientific Publishing.
- Haas, C., & Hall, M. (2019). Two-sided matching for mentor-mentee allocations-algorithms and manipulation strategies. *PLoS ONE*, 14(3), e0213323.
- Haas, C., Kimbrough, S. O., Caton, S., & Weinhardt, C. (2013). Preference-based resource allocation: using heuristics to solve two-sided matching problems with indifferences. In *Economics of Grids, Clouds, Systems, and Services. GECON 2013* (pp. 149–160), Springer.
- Haas, C., Hall, M., & Vlasnik, S. (2018). Finding optimal mentor-mentee matches: A case study in applied two-sided matching. *Heliyon*, 4(6), e00634.
- Halldórsson, M. M., Irving, R. W., Iwama, K., Manlove, D. F., Miyazaki, S., Morita, Y., et al. (2003). Approximability results for stable marriage problems with ties. *Theoretical Computer Science*, 306(1–3), 431–447.
- Halldórsson, M. M., Iwama, K., Miyazaki, S., & Yanagisawa, H. (2007). Improved approximation results for the stable marriage problem. *ACM Transactions on Algorithms (TALG)*, 3(3), 30.
- Holland, J. H. (1992). *Adaptation in natural and artificial systems*. Cambridge: MIT Press.
- Irving, R. W. (1994). Stable marriage and indifference. *Discrete Applied Mathematics*, 48(3), 261–272.
- Irving, R. W., & Manlove, D. F. (2008). Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems. *Journal of Combinatorial Optimization*, 16(3), 279–292.
- Irving, R. W., Leather, P., & Gusfield, D. (1987). An efficient algorithm for the optimal stable marriage. *Journal of the ACM*, 34(3), 532–543.
- Iwama, K., Miyazaki, S., Morita, Y., & Manlove, D. (1999). Stable marriage with incomplete lists and ties. In J. Wiedermann, P. Emde Boas, & M. Nielsen (Eds.), *Automata, languages and programming* (Vol. 1644, pp. 443–452). Lecture notes in computer science Berlin: Springer.
- Iwama, K., Miyazaki, S., & Yanagisawa, H. (2010). Approximation algorithms for the sex-equal stable marriage problem. *ACM Transactions on Algorithms (TALG)*, 7(1), 2.

- Iwama, K., Miyazaki, S., & Yanagisawa, H. (2014). A 25/17-approximation algorithm for the stable marriage problem with one-sided ties. *Algorithmica*, 68(3), 758–775.
- Kesten, O. (2012). On two kinds of manipulation for school choice problems. *Economic Theory*, 51(3), 677–693.
- Kimbrough, S. O., & Kuo, A. (2010). On heuristics for two-sided matching: Revisiting the stable marriage problem as a multiobjective problem. In *Proceedings of the 12th annual conference on genetic and evolutionary computation* (pp. 1283–1290). ACM.
- Király, Z. (2011). Better and simpler approximation algorithms for the stable marriage problem. *Algorithmica*, 60(1), 3–20.
- Klaus, B., & Klijn, F. (2006a). Median stable matching for college admissions. *International Journal of Game Theory*, 34(1), 1–11.
- Klaus, B., & Klijn, F. (2006b). Procedurally fair and stable matching. *Economic Theory*, 27, 431–447.
- Knuth, D. E. (1997). *Stable marriage and its relation to other combinatorial problems: An introduction to the mathematical analysis of algorithms* (Vol. 10)., CRM Proceedings & Lecture Notes, Centre de Recherches Mathématiques Université de Montréal Providence, RI: American Mathematical Society.
- Kwanashie, A., & Manlove, D. F. (2014). An integer programming approach to the hospitals/residents problem with ties. *Operations research proceedings 2013* (pp. 263–269). Cham: Springer.
- Manlove, D. F., Irving, R. W., Iwama, K., Miyazaki, S., & Morita, Y. (2002). Hard variants of stable marriage. *Theoretical Computer Science*, 276(1), 261–279.
- Mattei, N., & Walsh, T. (2013). PrefLib: A library for preferences. In *ADT 2013: International conference on algorithmic decision theory* (pp. 259–270). Berlin: Springer. <http://www.preflib.org>
- McDermid, E. (2009). A 3/2-approximation algorithm for general stable marriage. In *Automata, languages and programming* (pp. 689–700). Berlin: Springer.
- Nakamura, M., Onaga, K., Kyan, S., & Silva, M. (1995). Genetic algorithm for sex-fair stable marriage problem. In *1995 IEEE international symposium on circuits and systems, 1995. ISCAS '95* (Vol. 1, pp. 509–512).
- Pais, J. (2008). Random matching in the college admissions problem. *Economic Theory*, 35(1), 99–116.
- Paluch, K. (2014). Faster and simpler approximation of stable matchings. *Algorithms*, 7(2), 189–202.
- Pathak, P. A., & Sönmez, T. (2013). School admissions reform in Chicago and England: Comparing mechanisms by their vulnerability to manipulation. *American Economic Review*, 103(1), 80–106.
- Pini, M. S., Rossi, F., Venable, K. B., & Walsh, T. (2011). Manipulation complexity and gender neutrality in stable marriage procedures. *Autonomous Agents and Multi-Agent Systems*, 22(1), 183–199.
- Roth, A. E. (1982). The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4), 617–628.
- Roth, A. E. (2008). Deferred acceptance algorithms: History, theory, practice, and open questions. *International Journal of Game Theory*, 36(3), 537–569.
- Roth, A. E., & Rothblum, U. G. (1999). Truncation strategies in matching markets-in search of advice for participants. *Econometrica*, 67(1), 21–43.
- Roth, A. E., & Sotomayor, M. (1992). *Two-sided matching: A study in game-theoretic modeling and analysis*, (Vol. 18). Cambridge: Cambridge University Press.
- Vien, N. A., & Chung, T. C. (2006). Multiobjective fitness functions for stable marriage problem using genetic algorithm. In *International Joint Conference on SICE-ICASE, 2006* (pp. 5500–5503).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval, sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com