Rotating Proposer Mechanisms for Team Formation

Jian Lou[1], Martin Van der Linden[2], Pranav Batra[1], Chen Hajaj[1], Gregory Leo[2], Yevgeniy Vorobeychik[1], and Myrna Wooders[2]

[1] Department of Electrical Engineering and Computer Science, Vanderbilt University
[2] Department of Economics, Vanderbilt University

Abstract. We consider the problem of partitioning a collection of individuals, endowed with hedonic preferences, into teams (coalitions). First, we study this setting through the lens of a complete information non-cooperative sequential team formation game in which players iteratively recommend teams, which are either accepted or rejected by their prospective members. We show that in this game there always exists a subgame perfect equilibrium in which all team proposals are accepted. Next, we consider team formation as a mechanism design problem, where a center takes a profile of hedonic preferences (which are now private to the players) as input, and returns a partition of players into teams. We introduce a novel class of mechanisms which implement an "always-accept" equilibrium of the corresponding team formation game. We show that these mechanisms are individually rational (IR), and implement *iterative matching of soulmates (IMS)*, an important property in which mutually-preferred teams are always formed. Moreover, we exhibit a subclass of mechanisms, *rotating proposer mechanisms (RPM)*, which are also Pareto efficient. Remarkably, the resulting mechanisms are the first known team formation mechanisms to satisfy these three properties; indeed, this is the case even for the well-known *roommate* problem (teams of at most 2). While the mechanism is not stable or incentive compatible in general, we show that both these properties hold on an important restricted domain of preferences, and experimentally demonstrate more generally that few players can ever benefit from lying. A major shortcoming of RPM is implementation complexity. We address it by a combination of pruning strategies, and heuristics to compute an approximate subgame perfect equilibrium. Experimentally, we demonstrate that the resulting approximate RPM (which is still IR and implements IMS) remains approximately truthful, and significantly outperforms several known alternatives in terms of social welfare and fairness.

1 Introduction

Division of individuals into groups is a common task, important in a multitude of economic and social problems. Examples include dividing students into study groups or dorms, forming teams for a basketball game, or forming groups for carpooling. The issue of team, or coalition, formation in domains of hedonic preferences (players only care about the members of their own team) is commonly studied from the perspective of stability, where the focus is on characterizing or computing solutions of the game, such as the core [10, 4, 7]. In contrast, we consider team formation as a mechanism design problem in which a central authority is in charge of forming teams based on players' reported hedonic preferences. A challenging aspect of this mechanism design problem is that players may seek to benefit by misreporting their true preferences. It is further complicated if the mechanism is required to satisfy additional desiderata, such as individual rationality (players have incentives to participate), matching of soulmates (any collection of players who mostly prefer to be with one another are always matched), efficiency, and fairness. Indeed, in general, this combination of desiderata is impossible to achieve, even in two-sided matching problems [26], which is a subclass of the team formation mechanism design problem. Alcalde and Barberà [3] point out that without restrictions on the sets of admissible preferences, there is no matching mechanism that is Pareto efficient, individually rational, and incentive compatible. In general, it is also impossible to design team formation mechanisms that are both incentive compatible and match soulmates [19].

Two special cases of the team formation mechanism design problem have received considerable attention: two-sided matching markets [26], such as matching medical school residents with residency programs, and one-sided matching or assignment problems [1, 13], such as school choice and course allocation (the latter abstracted as combinatorial assignment). Well-known mechanisms for one- and two-sided matching, such as the deferred acceptance mechanism [26], possess many of the desired properties, but even generalizing to combinatorial assignment runs into numerous impossibility results [13]. Indeed, even in the roommate problem [16] where arbitrary teams of pairs can be constructed, few known positive results exist. In the general team formation problems, *random serial dictatorship (RSD)* is, to our knowledge, the only mechanism which is incentive compatible and ex post Pareto efficient [8, 29]. Although Wright and Vorobeychik [29] present several other mechanisms, these do not satisfy any of the mentioned desiderata, making RSD the only theoretically grounded mechanism known for general team formation.

We introduce a new class of mechanisms for general team formation problems, *always-accept mechanisms (AAMs)*. AAMs are constructed using subgame perfect equilibria of an *accept-reject game (ARG)* in which players propose teams in a predetermined order. Unlike RSD, prospective teammates may choose to either accept or reject the proposals in ARGs. We show that ARGs always have a subgame perfect equilibrium in which all proposals are accepted. These equilibria are then implemented in AAMs. We show that AAMs satisfy individual rationality and implement *iterated matching of soulmates* [19], where soulmate teams are matched in an iterative fashion, making these the first (to our knowledge) general class of team formation mechanisms to achieve both of these properties. In addition, we exhibit a subclass of AAMs, termed *Rotating Proposer Mechanism (RPM)*, which also satisfy Pareto efficiency. Moreover, while AAMs are not incentive compatible in general, they do possess this property for an important restricted set of hedonic preference domains in which iterated matching of soulmates matches all players to teams. In this restricted preference domain, truth telling in AAMs is a strong ex post equilibrium.

One significant challenge in implementing RPM in practice is the combinatorial complexity of backwards induction. We address this issue in two ways. First, we use the IMS as an efficient preprocessing and pruning procedure. We show experimentally that this significantly reduces the computational burden of performing backward induction.

Second, we develop a method to approximate RPM on the roommate problem which allows us to trade off computation time and quality of approximation of the subgame perfect equilibrium in RPM. Our experiments demonstrate that there is a natural tradeoff point which allows us to retain most of the positive properties of RPM at a significantly reduced computational overhead. To enable scalability on problems with larger teams, we propose *Heuristic Rotating Proposer Mechanism (HRPM)* which uses heuristics both to determine which teams are proposed and whether they are accepted.

In extensive experiments, we evaluate economic properties of RPM in both exact and approximate versions, compared to the RSD and a recent alternative, *one-player-one-pick (OPOP)* shown previously to be highly effective. We observe that in all instances, exact and approximate RPM is significantly more efficient (in terms of social welfare) and more equitable than RSD in the roommate problem. We also observe that HRPM significantly outperforms both RSD and OPOP on settings where teams' size can be at most 3 on both of these metrics in nearly all cases. Moreover, using an algorithm for finding an upper bound on untruthful players, we show that RPM and its approximate versions introduce few incentives to lie.

Related Work In recent years, team formation has been extensively investigated in Economics and AI literature. While some consider settings with cooperative agents [28, 27, 15, 20, 7], other research is focused on self-interested players aiming to maximize their utility [5, 24, 6, 12, 21]. In the latter literature, the focus tends to be on stability concepts, such as the core [10, 4, 7]. Several investigations consider transferable utility settings with an orthogonal notion of individual rationality to ours, and assume complete information [17, 18, 28]. Surprisingly, the literature considering team formation from a mechanism design perspective has been relatively limited. Several efforts consider this problem within a highly restricted set of teams (for example, single-lapping coalitions) [14, 23, 25]. Wright and Vorobeychik [29] consider the general team formation problem with cardinality being the only constraints on teams. They proposed several mechanisms for this problem; however, none are incentive compatible or Pareto efficient with the exception of random serial dictatorship, which is both.

2 Model and Preliminaries

We consider the standard model described in [9] of an environment populated a set of players $N = \{1, \ldots, n\}$ who need to be partitioned into teams. A team (coalition) $T \in 2^N$ is a set of players, and a partition (coalition structure) π is a collection of teams such that: 1) for any distinct $T, T' \in \pi$, $T \cap T' = \emptyset$, and 2) $\cup_{T\in\pi} T = N$. For a player i, let π_i be the team in the partition π containing i. In many tasks, teams have some feasibility constraints; for example, one could constrain teams to consist of at most k individuals. Generically, let $\mathcal{T}$ denote the set of *feasible* teams, which we assume to always include singleton teams, $\{i\}$. For a player i, we denote a subset $\mathcal{T}_i \subset \mathcal{T}$ of teams that include i by $\mathcal{T}_i$. Each player $i \in N$ has a strict hedonic preference ordering $\succeq_i$ over $\mathcal{T}_i$. A profile of preferences $\succeq$ (or *profile* for short) is a list of preferences for every $i \in N$. Given a profile $\succeq$, the list of preferences for all players except i is denoted by $\succeq_{-i}$. A *team formation mechanism* M maps every preference profile $\succeq$ to a partition π, i.e. $\pi = M(\succeq)$.

In designing a team formation mechanism, we consider the following general desiderata: *individual rationality*, *Pareto efficiency*, *incentive compatibility*, *matching soulmates*, and *fairness*.

Individual Rationality. A mechanism is *individually rational* if each player is always assigned to a team that it (weakly) prefers to be alone.[3] Formally, $\forall \succeq$ and $\forall\ i \in N$, $M(\succeq)_i \succeq_i \{i\}$.

Pareto Efficiency. A mechanism is (ex post) *Pareto efficient* if it is impossible to find a partition that makes any player better off without making another worse off. Formally, for all $\succeq$, there exists no π such that $\pi_i \succeq_i M(\succeq)_i$ for all $i \in N$ and $\pi_j \succ_j M(\succeq)_j$ for some $j \in N$.

Incentive Compatibility. A mechanism is *ex post incentive compatible* if no player has an incentive to misreport her preference for any realization of preference profiles. Formally, $\forall \succeq$, $\forall\ i \in N$, and $\forall \succeq_i'$, $M(\succeq)_i \succeq_i M(\succeq_i', \succeq_{-i})_i$.

Iterated Matching of Soulmates (IMS). We propose an additional criterion for team formation mechanisms introduced by Leo et al. [19]: *iterative matching of soulmates*, which captures the idea that players with reciprocal preferences (i.e., who prefer to be with one another) should be teamed up. Formally, a team T is a team of *soulmates* if $\forall i \in T$, $T \succeq_i T'$ for all $T' \in \mathcal{T}_i$. Iteratively applying this criterion we obtain *iterated matching of soulmates (IMS)*: in every iteration, we match all teams consisting of soulmates among players not matched in prior rounds. We formalize this criterion in the following definition.

Definition 1. *A mechanism M* implements IMS *if for every preference profile $\succeq$ the teams that form under the IMS form under M.*

Thus, the criterion is that the mechanism implements IMS. Informally, this criterion is crucial because any mechanism which does not match players who wish to be with one another would be ill perceived. A more formal motivation is that all teams matched by IMS are blocking coalitions [19], and players in these coalitions are likely to leave their assigned teams to be with one another. Moreover, implementing IMS has important consequences for incentive compatibility and core stability. We deal with these issues in greater detail below.

Fairness. Fairness aims to capture the disparity in utility achieved by the players. Commonly, ex post fairness is of primary concern. One natural measure of fairness is variance in utility. However, this measure does not do justice to mechanisms that strictly improve everyone's utility. We, therefore, consider two alternative notions of fairness described in Section 6.

Achieving the above desiderata together is impossible in general. As two-sided matching is a special case of our problem, the following theorem is a consequence of Proposition 1 in [3].

Theorem 1. *There exists no team formation mechanism that is Pareto efficient, individually rational, and incentive compatible.*

[3] With strict preferences, this means that the mechanism either assigns each player i a strictly preferred team to $\{i\}$, or the singleton team $\{i\}$.

For instance, random serial dictatorship (RSD) mechanism is incentive compatible and Pareto efficient, but not individually rational. Furthermore, incentive compatibility and matching of soulmates are also incompatible, as shown by Leo et al. [19].

Faced with these strong impossibility results, we adopt a pragmatic approach aimed at *partially* achieving the above desiderata. First, we attempt to satisfy them in important restricted preference domains. Second, we evaluate how well these are achieved empirically for synthetically generated distributions of preferences and real preference data.

3 A Team Formation Game

We begin by considering a natural sequential non-cooperative team formation game with complete information about hedonic preferences of all players, which will serve as the core component of the team formation mechanism below. We term such games *accept-reject games (ARGs)*, because they proceed through an exogenously specified order of players, with each player making a proposal of a team, and all prospective members having a chance to accept or reject this proposal.

Formally, an ARG is defined by a set of players N, a preference profile $\succeq$, a set of feasible teams $\mathcal{T}$, and an ordered list of players $O = (o_1, o_2, \ldots, o_m)$, in which each player $i \in N$ is included at least once. The game proceeds through a series of rounds. In each round the next player i in the order list O proposes to a team $T \in \mathcal{T}_i$, with the constraint that i cannot have proposed to T in any prior round. Given a proposal T made by i, all $j \in T \setminus i$ sequentially decide whether to accept or reject the proposal.[4] If any player rejects, the entire proposal is rejected, and we proceed with the next round. If all $j \in T \setminus i$ accept, the team T is added to the partition π, all players are removed from the game and from O, and the game proceeds to the next round, unless no players remain (in which case the game ends with a partition π). If after m rounds there are players remaining, they each become singleton teams, completing the partition. Algorithm 1 in Appendix describes the game procedure more precisely.

Example 1. Consider an ARG with four players $N = \{1,2,3,4\}$, and the order of proposers $O = (1, 2, 3, 4)$ in which the size of each team is at most two. Suppose that the profile is as follows:

$$\begin{array}{l} 1 : \{1,4\} \succeq_1 \{1,2\} \succeq_1 \{1,3\} \succeq_1 \{1\} \\ 2 : \{2,1\} \succeq_2 \{2,4\} \succeq_2 \{2,3\} \succeq_2 \{2\} \\ 3 : \{3,2\} \succeq_3 \{3,1\} \succeq_3 \{3,4\} \succeq_3 \{3\} \\ 4 : \{4,3\} \succeq_4 \{4,2\} \succeq_4 \{4,1\} \succeq_4 \{4\} \end{array}$$

The following is an example scenario:

1. Player 1 proposes to $\{1, 4\}$, and 4 rejects the proposal.
2. 2 proposes to $\{2, 1\}$ and 1 accepts the proposal. 1 and 2 are removed from the game.
3. 3 propose to $\{3, 4\}$ and 4 accepts the proposal. $3, 4$ are removed.

[4] The order of this sequence can be arbitrary; for example, it can follow the order of the players' first appearance in O.

The partition that results from this sequence is $\pi = \{\{1,2\},\{3,4\}\}$.

Even when preferences are strict, ARG admits multiple subgame perfect equilibria (SPE). However, we show that there is always some subgame perfect equilibrium in which every proposal is accepted.

Theorem 2. *Let $(N, \succeq, \mathcal{T}, O)$ be an ARG. There exists an SPE of this game in which all proposals are accepted.*

4 Always-Accept Mechanisms for Team Formation

The property that ARGs always have an equilibrium with all proposals accepted is the key ingredient in the novel class of mechanisms we now introduce which we term *always-accept mechanisms (AAMs)*. Always accept mechanisms are defined with respect to the tuple $(N, \succeq, \mathcal{T}, O)$ as follows:

1. In any iteration, a player i chosen according to order O proposes to her most preferred team $T \in \mathcal{T}_i$ to which it has not previously proposed and *in which all players would accept based on the equilibrium play of the corresponding ARG.*
2. All $j \in T \setminus i$ accept the proposal.
3. All players in T are removed (changing the set of feasible teams as a result), O is adjusted accordingly, and the mechanism proceeds to the next round.

The partition π can be computed for AAMs using backwards induction, which at the high level works as follows. A player who is making a proposal "tries" to propose to teams in order of its preferences. For each such hypothetical proposal, all team members determine whether they receive a better allocation in the subgame where they reject the proposal. If they do, they reject the proposal; otherwise, the proposal is accepted.

Next, we consider the economic properties of AAMs relating to our desiderata in Section 2.

Individual Rationality In AAMs, individual rationality comes form the fact that players could always reject the proposals and be singletons.

Theorem 3. *Every AAM mechanism is individually rational.*

While we view individual rationality as a basic criterion in mechanism design for team formation, surprisingly, previous team formation mechanisms have failed to satisfy it. For example, *random serial dictatorship (RSD)* [8, 29], in which players are randomly ordered and each proposer in this order forms their favorite team, are clearly not individually rational. Indeed, none of the four mechanisms discussed by [29] satisfy IR.

Matching Soulmates We next show that AAMs always satisfy another important criterion from Section 2: it implements IMS.

Theorem 4. *AAMs implement IMS.*

While implementing IMS is of independent interest, it also has important consequences for incentive compatibility. We deal with these below.

Incentive Compatibility Unlike RSD, AAMs are not, in general, incentive compatible, as the following example shows.

Example 2. Consider an ARG with three players $N = \{1, 2, 3\}$, the order of proposers $O =(1, 1, 1, 2, 2, 2, 3, 3, 3)$, and the size of each team is at most two. Suppose that the profile is as follows:

$$\begin{array}{l} 1 : \{1, 2\} \succeq_1 \{1, 3\} \succeq_1 \{1\} \\ 2 : \{2, 3\} \succeq_2 \{2, 1\} \succeq_2 \{2\} \\ 3 : \{3, 1\} \succeq_3 \{3, 2\} \succeq_3 \{3\} \end{array}$$

In AAM, 2 accepts 1's proposal because 2 knows that if she rejects, 1 will propose to 3 who will accept because 1 is 3's favorite teammate. But this means 3 is left alone under the true preferences. And the outcome is $\pi = \{\{1, 2\}, \{3\}\}$.

3 can then lie and pretend 2 is her favorite player: $3 : \{3, 2\} \succeq_3 \{3, 1\} \succeq_3 \{3\}$. Then both 2 and 3 will reject 1's proposal and 2 will propose to 3 and got accepted. Then the new outcome is $\pi' = \{\{1\}, \{2, 3\}\}$, and 3 get a teammate 2 rather than be a singleton.

Nevertheless, we now demonstrate that, by virtue of implementing IMS, truth telling in AAMs is incentive compatible for an important restricted class of preferences. In addition, we show experimentally below that few players have incentives to misreport preferences even for very general generative models of preferences. The formal notion of incentive compatibility we use is stronger than ex post incentive compatibility described in Section 2. We use R to denote the domain of *true* preferences, and D denotes the smallest cartesian domain of preferences which includes R. D_S denotes a restriction of the domain D to a subset of players S.

Definition 2. *A mechanism M is* strongly ex post incentive compatible (SEPIC) *on a preference domain R if for all preference profiles $\succeq \in R$ there does not exist a subset of players $S \subseteq N$ and a report $\succeq'_S \in D_S$ different from $\succeq_S$ such that $M(\succeq'_S, \succeq_{N \setminus S}) \succeq_i M(\succeq)$ for all $i \in S$.*

When the preference domain R is cartesian, SEPIC becomes equivalent to "group-strategyproof". The preference domain R of interest to us is the domain of all *IMS-complete* profiles.

Definition 3. *A profile $\succeq$ is* IMS-complete *if all players are matched to teams when IMS terminates.*

The following key result follows directly from [19], who show that any mechanism which implements IMS is SEPIC on the domain of all IMS-complete preference profiles.

Theorem 5. *AAMs are strongly ex post incentive compatible when restricted to a domain of all IMS-complete preference profiles.*

The importance of this result, as argued by Leo et al. [19], is that IMS-complete preferences are quite natural in the sense that they capture reciprocity commonly present in real preferences (see experimental investigation by Leo et al. [19]). Moreover, these generalize common ranking preferences and full reciprocal preferences (where all teams involve soulmates), as well as the *top-coalition* property considered in [9].

Pareto efficiency The final property we deal with is Pareto efficiency. To date, random serial dictatorship (RSD) [8] is the only Pareto efficient mechanism for team formation. In general, AAMs are not Pareto efficient (see Appendix 8.4). We now introduce a special class of these, termed *Rotating Proposer Mechanism (RPM)*, which is.

In RPM, the order O over players is such that each player i can make $|\mathcal{T}_i|$ offers before we move on to another player. Of course, by Theorem 2, only one offer is ever made by any player in our mechanism, but the fact that the player is able to continue making offers if it is rejected changes the equilibria. The crucial consequence is that when the order moves on to another proposer, i has necessarily been allocated to a team and removed from the remaining set of players (in the worst case, i ends by proposing to $\{i\}$ and accepting it). In RPM, a proposer i always proposes in the order $\succeq_i$ (i.e. i first proposes to i's most preferred team, then to i's second, and so on).

Example 3. Consider the profile in Example 1. In RPM, the order is $(1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4)$. Consider the following process: In the first round, 1 proposes to $\{1, 4\}$, and 4 rejects. In the second round, it is still 1's turn, 1 proposes to $\{1, 2\}$, and 2 accepts the proposal. 1 and 2 are removed from the game. In the third round, 3 proposes to 4 and 4 accept the proposal. 3 and 4 are removed from the game. The outcome of the game is $\pi = \{\{1, 2\}, \{3, 4\}\}$. In fact, it is also the subgame perfect equilibrium of the game and the outcome of RPM.

Theorem 6. *RPM is ex-post Pareto efficient.*

Consequently, RPM is IR, Pareto efficient, and implements IMS; it is the first known mechanism for general team formation problems to satisfy these three properties. In our implementation and experiments below, we therefore restrict attention to the RPM.

5 Implementing RPM

While we showed that RPM has important theoretical advantages, it is computationally challenging to implement. In particular, the size of the backward induction search tree is $O(2^{\sum_{i=1}^{n} |\mathcal{T}_i|})$. Even in roommate problem, in which the size of teams is at most two, computing SPE is $O(2^{n^2})$. We address this challenge in three ways: (1) preprocessing and pruning to reduce the search space, (2) approximation for the roommate problem, and (3) a general heuristic implementation.

5.1 Preprocessing and Pruning

One of the central properties of RPM is that it implements iterative matching of soulmates. In fact, it does so in every subgame in the backwards induction process. Now, observe that computing the subset of teams produced through IMS is $O(n^3)$ in general, and $O(n^2)$ for the roommate problem, and is typically much faster in practice. We therefore use it as a preprocessing step both initially (reducing the number of players we need to consider in backwards induction), and in each subgame of the backwards induction search tree (thereby pruning irrelevant subtrees).

5.2 Approximate RPM for the Roommate Problem

Using IMS for preprocessing and pruning does not sufficiently speed up RPM computation in large-scale problem instances. Next, we developed a parametric approximation of RPM which allows us to explicitly trade off computational time and approximation quality. We leverage the observation that the primary computational challenge of applying RPM to the roommate problem is determining whether a proposal is to be accepted or rejected. If we are to make this decision without exploring the full game subtree associated with it, considerable time can be saved. Our approach is to use a heuristic to evaluate the "likely" opportunity of getting a better teammate in later stages: if this heuristic value is very low, the offer is accepted; if it is very high, the offer is rejected; and we explore the full subgame in the balance of instances.

More precisely, consider an arbitrary offer from i to another player j. Given the subgame of the corresponding RPM, let $\mathcal{U}_j(i)$ denote the set of feasible teammates that j prefers to i, and let $\mathcal{U}_j(j)$ be the set of feasible teammates who j prefers to be alone. We can use these to heuristically compute the likelihood $R_j(i)$ that j can find a better teammate than the proposer i:

$$R_j(i) = \frac{|\mathcal{U}_j(i)|}{|\mathcal{U}_j(j)|} \cdot \frac{1}{|\mathcal{U}_j(i)|} \sum_{k \in \mathcal{U}_j(i)} \left(1 - \frac{|\mathcal{U}_k(j)|}{|\mathcal{U}_k(k)|}\right) = \frac{1}{|\mathcal{U}_j(j)|} \sum_{k \in \mathcal{U}_j(i)} \left(1 - \frac{|\mathcal{U}_k(j)|}{|\mathcal{U}_k(k)|}\right) \tag{1}$$

Intuitively, we first compute the proportion of feasible teammates that j prefers to i. Then, for each such teammate k, we find at the proportion of feasible teammates who are not more preferred by k than the receiver j. Our heuristic then uses an exogenously specified threshold, α, ($0 \leq \alpha \leq 0.5$) as follows. If $R_j(i) \leq \alpha$, player j accepts the proposal, while if $R_j(i) \geq 1 - \alpha$, the proposal is rejected. In the remaining cases, our heuristic proceeds with evaluating the subgame at the associated decision node. Consequently, when $\alpha = 0$, it is equivalent to the full backwards induction procedure, and computes the exact RPM. Note that for any α, this approximate RPM preserves IR, and we also maintain IMS by running it as a preprocessing step.

5.3 Heuristic Rotating Proposer Mechanism (HRPM)

Unlike the roommate problem, general team formation problems have another source of computational complexity: the need to iterate through the combinatorial set of potential teams to propose to. Moreover, evaluating acceptance and rejection becomes considerably more challenging. We therefore develop a more general heuristic which scales far better than the approaches above, but no longer has the exact RPM as a special case. We term the resulting approximate mechanism *Heuristic Rotating Proposer Mechanism (HRPM)*, and it assumes that the sole constraint on teams is their cardinality and that preferences can be represented by an additively separable utility function [9]. With the latter assumptions, we allow preferences over teams to be represented simply as preference orders over potential teammates, avoiding the combinatorial explosion in the size of the preference representation.

In HRPM, each proposer i attempts to add a single member to their team at a time in the order of preferences over players. If the potential teammate j accepts i's proposal, j is added to i's team, and i proposes to the next prospective teammate until either the team size constraint is reached, or no one else who i prefers to being alone is willing to join the team. Player j's decision to accept or reject i's proposal is based on calculating $R_j(l)$ for each member l of i's current team T using Equation 1, and then computing the average for the entire team, $R_j(T) = \frac{1}{|T|}\sum_{l \in T} R_j(l)$ (see Algorithm 2 in the Appendix for the fully precise description of HRPM). We then use an exogenously specified threshold $\beta \in [0, 1]$, where j accepts if $R_j(T) \leq \beta$ and rejects otherwise. The advantage of HRPM is that the team partition can be found in $O(\omega n^2)$, where ω is the maximum team size. The disadvantage, of course, is that it only heuristically implements RPM. Crucially, it does preserve IR, and IMS is implemented as a preprocessing step.

6 Experiment

Our evaluation considers two team formation settings: (1) the *roommate problem*, where teams are capped at 2, and (2) the *trio-roommate* problem, with teams of at most 3. We note that both of these problems are essentially open from a mechanism design perspective: in either case, RSD is the only known mechanism which is either Pareto efficient or incentive compatible even in a well-understood restricted setting. No mechanism is known for these problems which is both IR and Pareto efficient, or IR and implements IMS. We benchmark RPM and its approximate variants to RSD in the roommate problem, and additionally to the One-Player-One-Pick (OPOP) mechanism [29] in the trio-roommate setting (OPOP and RSD are equivalent in the roommate problem). OPOP first chooses a set of captains, and then captains choose a single teammate at a time, while non-captains choose a team to join following a heuristic evaluation function. In either case, such "proposals" are always accepted. OPOP was previously shown to outperform several others, including RSD, in terms of social welfare and fairness [29], but does not satisfy any of our desiderata. In all mechanisms, players are ordered randomly.

6.1 Data Sets

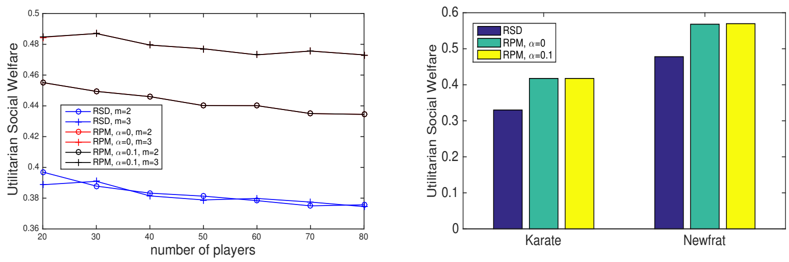
For evaluating our proposed mechanisms, we use both synthetic and real hedonic preference data. In both cases, preferences were generated based on a social network structure in which a player i is represented as a node and the total order over neighbors is then generated randomly. And non-neighbors represent undesirable teammates (i would prefer being alone to being teamed up with them). The networks used for our experiments were generated using the following models:

- **Scale-free network:** We adapt the Barabási-Albert model ([2]) to generate scale-free networks. For each (n, m), where n is the number of players, m denotes the density of the network, we generate $1,000$ instances of networks and profiles.
- **Karate-Club Network [30]:** This network represents an actual social network of friendships between 34 members of a karate club at a US university, where links correspond to neighbors. We generate 100 preference profiles based on the network.

Finally, we used a *Newfrat* dataset [22] which contains 15 matrices recording weekly sociometric preference rankings from 17 men attending the University of Michigan. In order to quantitatively evaluate both the exact and approximate variants of RPM, the ordinal preferences $\succeq_i$ have to be converted to cardinal ones $u_i(\cdot)$, upon which both mechanisms operate. For this purpose, we introduce a *scoring function* suggested by Bouveret and Lang [11] to measure a player's utility. The scoring function is a non-increasing function $g : [1..k] \to \mathbb{R}$ for some k and $[1..k]$ is the list of integers from 1 to k. To compute a player i's utility of player j we adopt *normalized Borda scoring function*, defined as $u_i(j) = g(r) = 2(k - r + 1)/k - 1$, where k is the number of i's neighbors, and $r \in [1..k]$ is the rank of j in i's preference list. Without loss of generality, for every player i we set the utility of being a singleton $u_i(i) = 0$. As mentioned above, in trio-roommate problem we assume that the preferences of players are additively separable [9], which means that a player i's utility of a team T is $u_i(T) = \sum_{j \in T} u_i(j)$.

6.2 Utilitarian Social Welfare

Ex post Pareto efficiency, satisfied by both RSD and RPM, is a very weak criterion. Conversion of ordinal to cardinal preferences allows us to consider empirically *utilitarian social welfare*, a much stronger criterion commonly used in mechanism design with cardinal preferences. We define social welfare as $\frac{1}{|N|} \sum_{i \in N} u_i(\pi_i)$, where π_i is the team that i was assigned to by the mechanism.

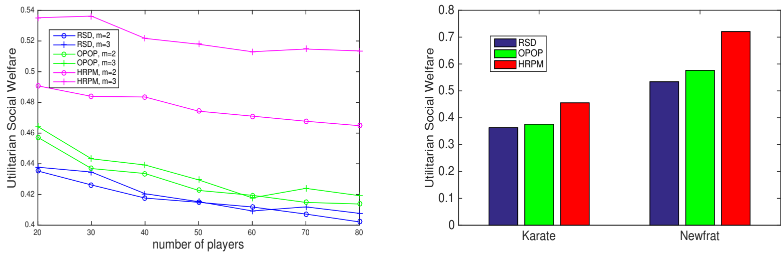


(a) scale-free networks (b) Karate and Newfrat

Fig. 1: Utilitarian social welfare for roommate problem

Figures 1a and 1b depict the average utilitarian social welfare for RSD and RPM in the roommate problem on scale-free networks, Karate club networks, and the Newfrat data. In all cases, RPM yields significantly higher social welfare than RSD, with $15\% - 20\%$ improvement in most cases. These results are statistically significant ($p < 0.01$). Furthermore, there is virtually no difference between exact and approximate RPM.

For the trio-roommate problem, we compare HRPM ($\beta = 0.6$) with RSD and OPOP on the same data sets. Figures 2a and 2b show that HRPM yields significantly higher social welfare than both RSD and OPOP in all instances, and HPRM performs even better when the network is comparatively dense ($m = 3$ in the scale-free network). All results are statistically significant ($p < 0.01$).

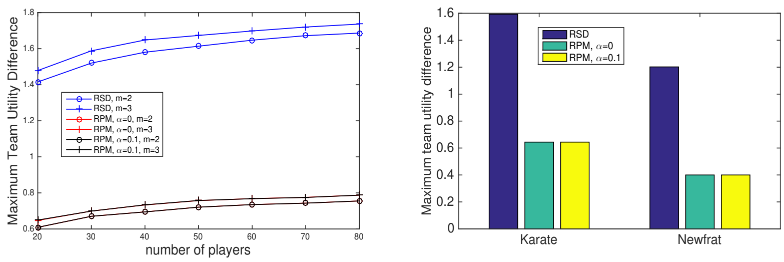


(a) scale-free networks (b) Karate and Newfrat

Fig. 2: Utilitarian social welfare for trio-roommate problem

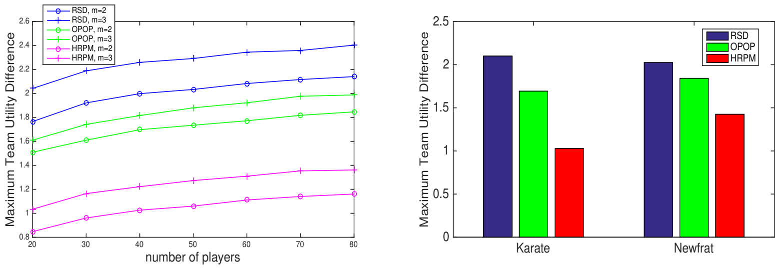
6.3 Fairness

A number of measures of fairness exist in prior literature. One common measure, envy-freeness, is too weak to use, especially for the roommates problem: every player who is not matched with his most preferred other will envy someone else. Indeed, because RPM matches soulmates—in contrast to RSD, which does not—it already guarantees the fewest number of players with envy. We consider two alternative measures, which aim to capture different and complementary aspects of fairness: maximum team utility difference, and the correlation between utility and rank in the random proposer order. Maximum team utility difference measures the difference in utility between teammates in each team T in a partition π, and takes the largest such difference over all teams. Formally, it computes $\max_{T\in\pi}(\max_{i\in T} u_i(T) - \min_{i\in T} u_i(T))$. Correlation between utility and rank considers each random ranking of players in O used for both RSD and RPM, along with corresponding utilities $u_i(\pi)$ of players for the partition π generated by the mechanism, and computes the correlation between these. It thereby captures the relative advantage that someone has by being earlier (or later) in the order to propose than others, and is a key cause of ex-post inequity in RSD.



(a) scale-free network (b) Karate and Newfrat

Fig. 3: Maximum team utility difference for the roommate problem



(a) scale-free networks (b) Karate and Newfrat

Fig. 4: Maximum Team Utility Difference for the trio-roommate problem

Here we result the outcomes of Maximum Team Utility, and those of Correlation can be see in Appendix. Our experiments on the roommate problem show that RPM is significantly more equitable than RSD on scale-free networks (Figures 3a), as well as on the Karate club network and Newfrat dataset (Figures 3b). The differences between exact and approximate RPM are negligible in most instances. In the trio-roommate problem, HRPM ($\beta = 0.6$) is much more equitable than both RSD and OPOP. These results are statistically significant ($p < 0.01$).

6.4 Incentive Compatibility

Although RPM is strongly incentive compatible on IMS-complete domains, we now explore its incentive properties empirically in more general settings. We focus on the roommate problem, because here we can compute an upper bound on the number of players with an incentive to lie (see Appendix for more details).

Table 1: Average Upper Bound of Untruthful Players for (Approximate) RPM

n	20	30	40	50	60	70	80
$m = 2, \alpha = 0$	0.015%	0.013%	0.013%	0.002%	0.008%	0.011%	0.010%
$m = 2, \alpha = 0.1$	0.015%	0.010%	0.015%	0.004%	0.022%	0.029%	0.036%
$m = 3, \alpha = 0$	0.105%	0.107%	0.072%	0.038%	0.037%	0.024%	0.023%
$m = 3, \alpha = 0.1$	0.115%	0.103%	0.085%	0.076%	0.065%	0.074%	0.093%

Table 1 presents the upper bound on the number of players with an incentive to lie, as a proportion of all players, on scale-free networks. We can observe that the upper bound is always below 0.2%, and is even lower when the networks are sparse ($m = 2$). On the Karate club data, we did not find any player with an incentive to lie in test cases when we apply (Approximate) RPM. On the Newfrat data, the upper bounds are less than 0.4% and 7% when we apply RPM without and with heuristics, respectively. More results concerning incentive compatibility can be seen in Appendix.

7 Discussion and Conclusions

We developed a novel class of mechanisms for general team formation problems. This class is based on team formation *games* in which players iteratively propose to teams which can be accepted or rejected by their prospective teammates. Showing that these games always possess a subgame perfect equilibrium in which all proposals are accepted, we designed a class of mechanisms which we term AAMs by implementing this equilibrium. We showed that AAMs are individually rational, and implement a general notion of conditional soulmates known as *iterated matching of soulmates (IMS)*, which is unlike all previously known mechanisms for general team formation problems. As a consequence of implementing IMS, AAMs possess a host of desirable theoretical properties on an important restricted class of IMS-complete preference domains. In addition, we exhibited a specific mechanism termed RPM in the AAM class which is also ex post Pareto efficient. In order to address the computational challenges in implementing RPM in practice, we introduce preprocessing and pruning, as well as approximate versions of RPM, one tailored to the roommate problem (with teams of at most two), and another for teams of arbitrary size. Our experiments show that even the approximate versions of RPM significantly outperform several alternative mechanisms for team formation in terms of social welfare and fairness, are do not introduce significant incentives to misreport preferences.

References

1. Abdulkadiroğlu, A., Sönmez, T.: School choice: A mechanism design approach. In: American Economic Review. vol. 93(3), pp. 729–747 (2003)
2. Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. Rev. Mod. Phys. 74, 47–97 (Jan 2002), `http://link.aps.org/doi/10.1103/RevModPhys.74.47`
3. Alcalde, J., Barberà, S.: Top dominance and the possibility of strategy-proof stable solutions to matching problems. Economic Theory 4(3), 417–435 (1994), `http://dx.doi.org/10.1007/BF01215380`
4. Alcalde, J., Revilla, P.: Researching with whom? stability and manipulation. Journal of Mathematical Economics 40(8), 869 – 887 (2004), `http://www.sciencedirect.com/science/article/pii/S0304406803001277`
5. Ambroszkiewicz, S., Matyja, O., Penczek, W.: Team formation by self-interested mobile agents. In: Australian Workshop on Distributed Artificial Intelligence. pp. 1–15. Springer (1998)
6. Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., Leonardi, S.: Online team formation in social networks. In: Proceedings of the 21st international conference on World Wide Web. pp. 839–848. ACM (2012)
7. Aziz, H., Brandt, F., Harrenstein, P.: Fractional hedonic games. In: International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). pp. 5–12. International Foundation for Autonomous Agents and Multiagent Systems (2014), `http://dl.acm.org/citation.cfm?id=2615731.2615736`
8. Bade, S.: Serial dictatorship: The unique optimal allocation rule when information is endogenous. Theoretical Economics 10, 385–410 (2015)
9. Banerjee, S., Konishi, H., Sönmez, T.: Core in a simple coalition formation game. Social Choice and Welfare 18(1), 135–153 (Jan 2001)

10. Bogomolnaia, A., Jackson, M.O.: The stability of hedonic coalition structures. Games and Economic Behavior 38(2), 201 – 230 (2002), http://www.sciencedirect.com/science/article/pii/S0899825601908772
11. Bouveret, S., Lang, J.: A general elicitation-free protocol for allocating indivisible goods. In: A general elicitation-free protocol for allocating indivisible goods. pp. 73–78 (2011), http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-024
12. Bredereck, R., Chen, J., Woeginger, G.J.: A characterization of the single-crossing domain. Social Choice and Welfare 41(4), 989–998 (2013), http://dx.doi.org/10.1007/s00355-012-0717-8
13. Budish, E.: The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. Journal of Political Economy 119(6), 1061–1103 (2011)
14. Cechlárová, K., Romero-Medina, A.: Stability in coalition formation games. International Journal of Game Theory 29(4), 487–494 (May 2001)
15. Gaston, M.E., desJardins, M.: Agent-organized networks for dynamic team formation. In: International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). pp. 230–237. ACM (2005)
16. Irving, R.W.: An efficient algorithm for the “stable roommates” problem. In: Journal of Algorithms. vol. 6(4), pp. 577–595 (1985)
17. Ketchpel, S.: Forming coalitions in the face of uncertain rewards. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). vol. 94, pp. 414–419. Citeseer (1994)
18. Klusch, M., Shehory, O.: A polynomial kernel-oriented coalition algorithm for rational information agents. In: AAAI. pp. 157–164 (1996)
19. Leo, G., Lou, J., Van der Linden, M., Vorobeychik, Y., Wooders, M.: Matching soulmates (February 2017), working paper, Available at SSRN: https://ssrn.com/abstract=2833553
20. Marcolino, L.S., Jiang, A.X., Tambe, M.: Multi-agent team formation: diversity beats strength? In: International Joint Conference on Artificial Intelligence (IJCAI) (2013)
21. Nahum, Y., Sarne, D., Das, S., Shehory, O.: Two-sided search with experts. Autonomous Agents and Multi-Agent Systems 29(3), 364–401 (2015)
22. Nordlie, P.: A longitudinal study of interpersonal attraction in a natural group setting. Ph.D. thesis, University of Michigan (1958)
23. Pápai, S.: Unique stability in simple coalition formation games. Games and Economic Behavior 48(2), 337–354 (Aug 2004)
24. Rathod, P., des Jardins, M.: Stable team formation among self-interested agents. In: AAAI Workshop on Forming and Maintaining Coalitions in Adaptive Multiagent Systems (2004)
25. Rodriguez-Alvarez, C.: Strategy-proof coalition formation. International Journal of Game Theory 38(3), 431–452 (2009)
26. Roth, A.E.: The economics of matching: Stability and efficiency. Mathematics of Operations Research 92, 617–628 (1982)
27. Sandholm, T., Sikka, S., Norden, S.: Algorithms for optimizing leveled commitment contracts. In: International Joint Conference on Artificial Intelligence (IJCAI). pp. 535–540 (1999), http://dl.acm.org/citation.cfm?id=1624218.1624295
28. Shehory, O., Kraus, S.: Methods for task allocation via agent coalition formation. Artificial Intelligence 101(1), 165 – 200 (1998), http://www.sciencedirect.com/science/article/pii/S0004370298000459
29. Wright, M., Vorobeychik, Y.: Mechanism design for team formation. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). pp. 1050–1056 (2015), http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9902
30. Zachary, W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research 33, 452–473 (1977)

8 Appendix

8.1 ARG Procedure

Algorithm 1 Construction of ARG

input: $(N, \succeq, \mathcal{T}, O)$
return: Team formation outcome π

$\pi = \emptyset$
while O is non-empty **do**
    $i \leftarrow$ the first player in O
    Player i proposes to a team $T \in \mathcal{T}_i(N)$
    All the players in T sequentially decide whether to accept i's proposal (if $T = \{i\}$, the proposal is automatically accepted)
    if All players in T accept player i's proposal **then**
        $\pi \leftarrow \pi \cup \{T\}$
        $N \leftarrow N \backslash T$
        for each $j \in T$ **do**
            $O \leftarrow O \backslash \{j\}$
    else
        Get new O by removing the first instance i in O
    for each player $i \in N$ **do**
        for each feasible team $T \in \mathcal{T}_i$ **do**
            if $T \not\subset N$ **then**
                $\mathcal{T}_i \leftarrow \mathcal{T}_i \backslash \{T\}$
while N is non-empty **do** ▷ add singletons into the outcome.
    pick an arbitrary instance i from N
    $\pi \leftarrow \pi \cup \{i\}$
    $N \leftarrow N \backslash \{i\}$
return π

8.2 Proof of Theorem 2

Theorem 2. *Let $(N, \succeq, \mathcal{T}, O)$ be an ARG. There exists an SPE of this game in which all proposals are accepted.*

Proof. For ease of notation, we will use (N, O) instead of $(N, \succeq, \mathcal{T}, O)$ to denote an arbitrary subgame of an ARG.

The proof is illustrated in Figure 5. Let s be an arbitrary SPE of (N, O) with the property that, for any two subgames γ, γ' of (N, O), the restriction of s to γ equals the restriction of s to γ'.[5] If s contains no rejections, we are done. Suppose that a proposal is rejected in s. Let $(N', O_{N'})$ be the *first* subgame of (N, O) in which some player

[5] Such SPE must exist and can be constructed by backward induction.

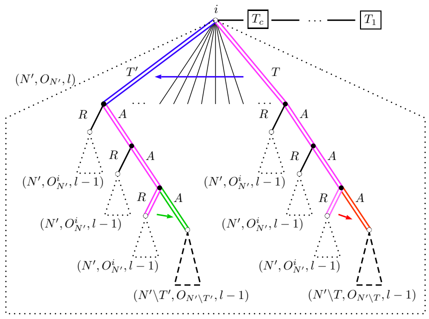


Fig. 5: Illustration of the proof of Theorem 2. Action A stands for "accept", and R stands for "reject".

i proposes to a team T and the proposal is rejected (where possibly, $(N', O_{N'}) = (N, O)$). We show how to construct another SPE s' from s in which player i proposes that is accepted. To construct s', we *only* have to change actions in the first round of $(N', O_{N'})$ where i proposes that is either rejected or accepted.

Let $O^i_{N'}$ be the list constructed from $O_{N'}$ by removing the first element of $O_{N'}$ (namely i). If i proposes to T that is rejected, i ends up being assigned to $(N', O^i_{N'})(s)_i$ (s is represented by the magenta double-lined strategy profile in Figure 5). There are two possible cases:

Case 1 : $(N', O^i_{N'})(s)_i = T$**.**
That is i proposes to T which is rejected, but i ends up in team T anyways. In this case it is easy to see that s' constructed from s by *only* changing the action of any agent who rejects i's proposal in the first stage of $(N', O_{N'})$ is also an SPE (change $\boxed{1}$ in Figure 5). Indeed, $(N', O_{N'})(s')_j = T = (N', O_{N'})(s)_j$ for all $j \in T$, and for any T_h that formed before $(N', O_{N'})$, we can get $(N', O_{N'})(s')_j = T_h = (N', O_{N'})(s)_j$ for all $j \in T_h$.[6]

Case 2 : $(N', O^i_{N'})(s)_i = T' \neq T$**.**
Then consider the strategy profile s' constructed from s by only changing i's proposal from T to T' (change $\boxed{2}$ in Figure 5). If T' is accepted under s, we are done, so assume there exists $j \in T'$ that rejects T' in $(N', O_{N'})$ under s. As illustrated in Figure 5 this again leads to i being assigned to team $(N', O^i_{N'})(s)_i = T'$ (because of the way s is chosen from the set of SPE at the beginning of the proof). That is $(N', O^i_{N'})(s')_j = (N', O^i_{N'})(s)_j$ for all $j \in T'$. But then again, it is not hard to see that we can construct yet another SPE s'' from s' by only changing to "accept" the action of any player in T' who rejects T' when proposed by i in the first stage of $(N', O_{N'})$ (change $\boxed{3}$ in Figure 5). Indeed, once again, $(N', O_{N'})(s'')_j = T' = (N', O_{N'})(s')_j$ for all $j \in T'$, and for any T_h that formed before $(N', O_{N'})$, $(N', O_{N'})(s'')_j = T_h = (N', O_{N'})(s')_j$ for all $j \in T_h$.

[6] And because we did not change any action elsewhere in the game, all the other subgames remain at an equilibrium.

Clearly, one can repeat the above argument starting with s'' as initial SPE and some subgame following $(N'\backslash T, O_{N'\backslash T'})$ (in Case 2 or $(N'\backslash T, O_{N'\backslash T})$ in Case 1) as the first subgame which starts with a team being rejected. Because the game is finite, repeating the argument sufficiently many time yields a strategy s^* in which all proposals are accepted.

8.3 Proof of Theorem 4

Before we prove Theorem 4, we firstly introduce the following Lemma,

Lemma 1. *Take any subgame $(N', \succeq' \mathcal{T}_{N'}, O_{N'})$ of an ARG $(N, \succeq, \mathcal{T}, O)$ which starts with some player $i \in N'$ proposing. If T is a team of soulmates, $i \in T$, and T is feasible for i, then i proposes to T in the SPE of $(N', \succeq', \mathcal{T}_{N'}, O_{N'})$ chosen by AAM.*

Proof. If i proposes to T, the players in T can only reject the proposal in equilibrium if T forms anyways later in the game. But then, by construction of AAM, the players in T accept the proposal, and i proposes to T.

Theorem 4. *AAMs implement IMS.*

Proof. The proof is by induction. Let S_r denote the set of soulmates formed in the r_{th} round of IMS. By Lemma 1, teams in S_1 form in any AAM. Now assume the teams in S_l form for every positive integer $l < r < r^*$. We show that teams in S_r form too.

Take any team $T \in S_r$. By assumption, some player $i \in T$ is given the ability to propose to a team. Let i be the first such player and let us focus on the first time i proposes in AAM. Because all the teams in S_l form for every positive integer $l < r$, if i proposes to a team including a player from a team that forms in S_l for some $l < r$, this proposal is rejected. Thus, in the SPE selected by AAM, i does not propose to any team containing a player from a team in S_l for some $l < r$. Formally, i only propose to teams composed of players in $N\backslash\bigcup_{\{T\in S_l|l<r\}}T$.

By assumption, T is the most preferred team for i among the players $N\backslash\bigcup_{\{T\in S_l|l<r\}} T$). Thus, if team T accepts i's proposal in the SPE selected by AAM, i must propose to T. In equilibrium, players in T refuse i's proposal only if they can be matched to a team they like better than T. But by assumption, any such team contains a player from $\bigcup_{\{T\in S_l|l<r\}} T$, and no such team can form under AAM. Thus, players in T accept i's proposal, which concludes the proof.

8.4 Example: ARG Subgame Perfect Equilibria are not Pareto Optimal

Example 4. Consider an ARG with 12 players $N = \{1, 2, \ldots, 12\}$, the order of proposers $O = (1, 2, \ldots, 12)$ in which the size of each team is at most two. Suppose that the profile is as follows (for easiness of notation, we only list the preferences of teammates rather than feasible teams):

1 :	4	6	8	9	3	5	11	2	10	12	7
2 :	9	4	8	7	1	6	5	10	3	11	12
3 :	11	4	2	9	7	6	8	12	5	10	1
4 :	9	7	8	3	10	5	12	1	11	2	6
5 :	1	6	11	2	4	10	8	7	9	12	3
6 :	2	5	8	4	3	7	10	12	11	9	1
7 :	5	8	2	3	6	4	11	10	12	1	9
8 :	6	9	7	11	3	4	10	1	5	12	2
9 :	10	8	1	12	11	2	5	4	7	3	6
10 :	3	5	8	6	4	7	2	11	1	12	9
11 :	3	10	2	6	7	12	4	9	8	5	1
12 :	10	2	4	5	11	7	9	6	8	3	1

We can use computer program to get the subgame perfect equilibrium of the ARG: $\{\{1,9\}, \{2,6\}, \{3,11\}, \{4,10\}, \{5,8\}, \{7,12\}\}$. In the outcome, $\{4,10\}, \{8,5\}$ is dominated by $\{4,8\}, \{5,10\}$, so we can find a Pareto improvement here.

8.5 Proof of Theorem 6

Theorem 6. *RPM is ex-post Pareto efficient.*

Proof. The proof is by contradiction. Suppose that the outcome π resulting from the mechanism is not Pareto optimal and π' is a Pareto-improvement over π. It is possible that some players are in the same teams for π and π'. We will focus on players that are in different teams in π and π'.

Of those players that are in different teams in π and π', there is a player i who is the first to propose among them in the mechanism. Let T_i be the team in π that includes i, and T_i' be the team in π' that includes i.

Suppose the Pareto-improvement π' includes i being matched with at least one partner not available when i proposed. Then there is a player who proposed before i who is on a different team under π and π', contradicting that i is the first such proposer. Thus, all the players in T_i and T_i' were available when i proposed. As i proposes in the order $\succeq_i$ in RPM, i must have proposed to team T_i'. This proposal must have been rejected, which implies that at least one person in the team T_i' prefers π to π', contradicting that π' is a Pareto-improvement.

Thus, there cannot be a Pareto-improvement, and the outcome of RPM is Pareto efficient.

8.6 Computing And Approximating RPM

Here we investigate the relationship between running time and approximation quality of our approaches to the roommate problem. Our simulations were performed on Mac OS 10.11 with a 2.6 GHz Intel Core i5 processor.

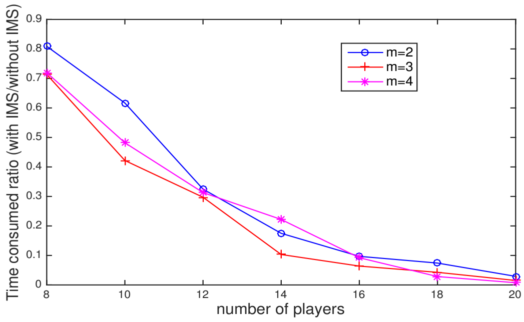
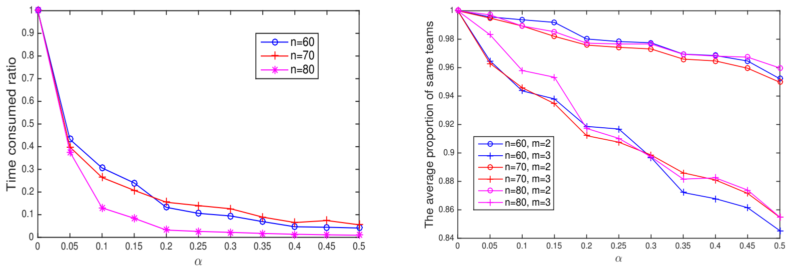


Fig. 6: Time consumed ratio (with IMS/without IMS) for RPM on scale-free networks

IMS Preprocessing First, we show the computational value of IMS in preprocessing and pruning using synthetic preference profiles based on the generative scale-free model.

Figure 6 shows the ratio of time consumed by RPM with IMS to that without IMS. In all cases, we see a clear trend that using IMS in preprocessing and pruning has increasing importance with increased problem size. The key takeaway is that *implementing IMS has both important economic and computational consequences*.

Approximating RPM The parameter α of our approximation method for RPM in the roommate problem allows us to directly evaluate the tradeoff between running time and quality of approximation: small α will lead to less aggressive use of the acceptance/rejection heuristic, with most evaluations involving actual subgame search, while large α yields an increasingly heuristic approach for computing RPM, with few subgames fully explored.



(a) Time consumed ratio (b) average proportion of same teams

Fig. 7: Time consumed and average proportion of same teams

Figure 7a depicts the fraction of time consumed by RPM with different values of α compared to exact RPM (when $\alpha = 0$) on scale-free networks ($m = 3$). Based on this figure, even a comparatively small value of α dramatically decreases computation time. Figure 7b compares similarity of the final team partition when using the heuristic compared to the exact RPM. Notice that even for high values of α, there is a significant overlap between the outcomes selected by RPM with and without the heuristic. $\alpha = 0.1$ appears to trade off approximation quality and running time particularly well: for comparatively sparse networks (i.e., $m = 2$) it yields over 99% overlap with exact RPM (this proportion is only slightly worse for denser networks), at a small fraction of the running time. Henceforth, we use $\alpha = 0.1$ when referring to the approximate RPM.

8.7 HRPM mechanism

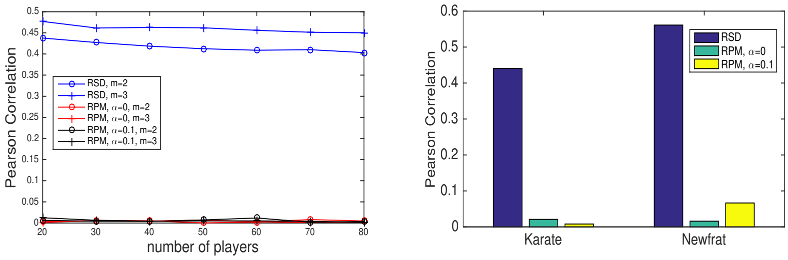
Algorithm 2 Heuristic Rotating Proposer Mechanism (HRPM)

input: $(N, \succeq, O), \omega, \beta$
return: Team formation outcome π

$\pi = \emptyset$
while O is non-empty **do**
    $i \leftarrow$ the first player in O
    $\pi_i \leftarrow \{i\}$
    while $|\pi_i| < \omega$ **do**
        if $\succeq_i$ is empty or the first player in $\succeq_i$ is i **then**
            $O \leftarrow O \backslash \{i\}$
            break
        Player i proposes to the first player j in $\succeq_i$
        for each $l \in \pi_i$ **do**
            Calculate $R_j(l)$ based on equation 1
        Calculate $R_j(\pi_i) = \frac{1}{|\pi_i|} \sum_{l \in \pi_i} R_j(l)$
        if $R_j(\pi_i) \leq \beta$ **then** ▷ player j accepts the proposal
            $\pi_i \leftarrow \pi_i \cup \{j\}$
            $O \leftarrow O \backslash \{j\}$
            $N \leftarrow N \backslash \{j\}$
        Delete j from $\succeq_l$ for each player $k \in N$
    $O \leftarrow O \backslash \{i\}$
    $N \leftarrow N \backslash \{i\}$
    Delete i from $\succeq_l$ for each player $k \in N$
while N is non-empty **do** ▷ add singletons into the outcome.
    pick an arbitrary instance i from N
    $\pi \leftarrow \pi \cup \{i\}$
    $N \leftarrow N \backslash \{i\}$
return π

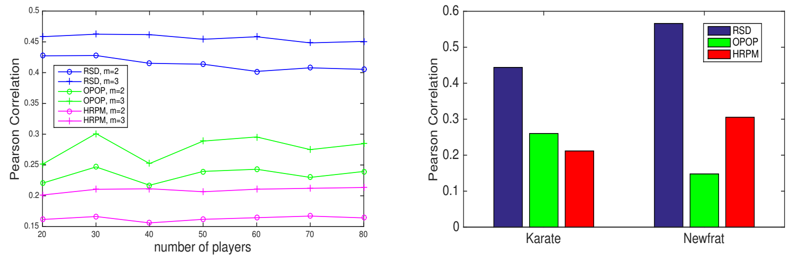
8.8 Additional Results of Fairness

Here we show the the Pearson correlation between player's ranking in the order and her utility. In roommate problem (Approximate) RPM is much more equitable than RSD in all test cases (Figure 8). In the trio-roommate problem, HRPM ($\beta = 0.6$) is much more equitable than both RSD and OPOP. except for Newfrat data, in which OPOP is better, as shown in Figures 9.



(a) scale-free network (b) Karate and Newfrat

Fig. 8: Pearson Correlation for the roommate problem



(a) scale-free networks (b) Karate and Newfrat

Fig. 9: Pearson Correlation for the trio-roommate problem

8.9 Incentive Compatibility

At the high level, Algorithm 3 considers all the players who have accepted or rejected a proposal, and checks whether reversing this decision improves their outcomes. The following theorem shows that this method indeed finds the upper bound of untruthful players.

Theorem 7. *Algorithm 3 returns an upper bound on the number of players who can gain by misreporting their preferences.*

Algorithm 3 Computing Upper Bound of Untruthful Players

input: $(N, \succeq, \mathcal{T}, O)$, teammate vector $teammate[]$ which results from RPM
return: number of potential untruthful players Sum

1: $Sum \leftarrow 0$
2: **while** $|O| \geq 2$ **do**
3: $\quad proposer \leftarrow$ the first player in O
4: $\quad receiver \leftarrow teammate[proposer]$
5: $\quad$ **for** player $i \in \mathcal{T}_{proposer}$ **do**
6: $\qquad$ **if** $i \succeq_{proposer} receiver$ and $proposer \succeq_i teammate[i]$ **then**
7: $\qquad\quad Sum \leftarrow Sum + 1$ $\triangleright$ i is potentially untruthful
8: $\quad$ **for** player $j \in \mathcal{T}_{receiver}$ **do**
9: $\qquad$ **if** $j \succeq_{receiver} proposer$ and $receiver \succeq_j teammate[j]$ **then**
10: $\qquad\quad Sum \leftarrow Sum + 1$ $\triangleright$ $receiver$ is potentially untruthful
11: $\quad$ remove $proposer$ and $receiver$ from N, O and $\mathcal{T}$
12: return Sum

Proof. We divide the players into *proposers* and *receivers*. Proposers are those who proposed in RPM and were thus teamed up (including singleton teams). Receivers accepted someone's offer.

There are 4 cases:

1. *A proposer i untruthfully reveals its preference and remains a proposer.* In RPM, a proposer proposes to other players in order of preference. When i proposes to j, all others more preferred by i must have already rejected. Consequently, i cannot improve the utility by lying.
2. *A receiver j untruthfully reveals its preference and is still a receiver.* In this case, if j has an incentive to lie, there has to be a proposer i' who prefers j to its teammate under RPM, while j must prefer i' to its teammate. Steps $4-7$ in Algorithm 3 count all such instances.
3. *A proposer i untruthfully reveals her preference and becomes a receiver.* In this case, if i has an incentive to untruthfully reveal her preference, there has to be a proposer i' who prefer i to their teammate under RPM, and who i also prefers to its teammate. Steps $4-7$ in Algorithm 3 count all such instances.
4. *A receiver j untruthfully reveals its preference and becomes a proposer.* In this case, if j has an incentive to misreport its preference, there must be a receiver j' who prefers j to its teammate, while j must prefer j' to its teammate. Steps $8-10$ in Algorithm 3 count all such instances.

In addition, we also computed the lower bound on the fraction of preference profiles where truth telling is a Nash equilibrium (Table 2). In Table 2 We find that without the heuristic, when $m = 2$ (sparse networks), RPM is incentive compatible in more than 99% of the profiles; and when $m = 3$ (the networks are comparatively dense), RPM is truthful at least 96% of the time.

Table 2: Lower Bound of Profiles Where Every Player Is Truthful for (Approximate) RPM

n	20	30	40	50	60	70	80
$m = 2$, $\alpha = 0$	99.7%	99.6%	99.5%	99.9%	99.6%	99.2%	99.2%
$m = 2$, $\alpha = 0.1$	99.7%	99.7%	99.4%	99.8%	98.8%	98.1%	97.2%
$m = 3$, $\alpha = 0$	97.9%	96.8%	97.1%	98.1%	97.8%	98.4%	98.3%
$m = 3$, $\alpha = 0.1$	97.8%	96.9%	96.8%	96.2%	96.3%	95.1%	92.9%

Table 3: Average Upper Bound of Untruthful Players for HRPM

n	20	30	40	50	60	70	80
$m = 2$, $\beta = 0.5$	1.44%	1.77%	1.71%	2.00%	2.09%	2.16%	2.06%
$m = 2$, $\beta = 0.6$	1.62%	1.83%	1.96%	2.09%	2.25%	2.11%	2.11%
$m = 3$, $\beta = 0.5$	2.99%	3.36%	3.76%	3.90%	4.18%	4.02%	4.33%
$m = 3$, $\beta = 0.6$	3.44%	3.69%	3.97%	3.98%	4.40%	4.24%	4.52%

Table 3 presents the upper bound on the number of untruthful players for HRPM (still for the roommate problem). Even with this heuristic, we can see that less than 5% of the players have any incentive to misreport preferences.