Proceedings of the 3rd Workshop on Exploring Beyond the Worst Case In Computational Social Choice (EXPLORE 2016)

Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016)

Singapore, May 10th, 2016

EXPLORE 2016 Website:
http://www.explore-2016.preflib.org/

AAMAS 2016 Website:
http://sis.smu.edu.sg/aamas2016

Preface

This volume contains the papers presented at EXPLORE-2016: The 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice held on May 8-9, 2016 in Singapore.

There were 11 paper accepted for presentation at the workshop and one invited talk from Tuomas Sandholm from Carnegie Mellon University.

This is the third installment of the EXPLORE workshop to be held at AAMAS. Our conference webpage at http://www.explore-2016.preflib.org/ has links to past iterations of the conference. More infomation about empirical testing in Social Choice and data can be found at www.PrefLib.org.

We thank Data61/NICTA for their continued support of EXPLORE and PrefLib. The conference was managed with EasyChair.

April 15, 2016
Sydney

Haris Aziz
Felix Brandt
David Manlove
Nicholas Mattei

Table of Contents

Program Committee

Haris Aziz	NICTA and UNSW
Peter Biro	Institute of Economics, Centre for Economic and Regional Studies, Hungarian Academy of Sciences
Felix Brandt	Technische Universität München
Markus Brill	Duke University
John P. Dickerson	Carnegie Mellon University
Edith Elkind	University of Oxford
Gabor Erdelyi	Universitaet Siegen
Piotr Faliszewski	AGH University of Science and Technology
Rupert Freeman	Duke University
Serge Gaspers	UNSW Australia and NICTA
Umberto Grandi	University of Padova
Jérôme Lang	LAMSADE
Kate Larson	University of Waterloo
Omer Lev	University of Toronto
David Manlove	University of Glasgow
Nicholas Mattei	NICTA and University of New South Wales
Reshef Meir	Harvard University
Nina Narodytska	Samsung Research America
Maria Silvia Pini	University of Padova
Mark Wilson	University of Auckland
Lirong Xia	RPI

Additional Reviewers

Reger, Christian

On the Parameterized Complexity of Manipulating Pairwise Voting Rules

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

Neeldhara Misra
Indian Institute of Technology Gandhinagar
India
mail@neeldhara.com

ABSTRACT

Pairwise voting rules are a generalization of the standard voting rules where instead of a ranked list, each voter provides a set of pairwise comparisons between the candidates and the voting rule picks a unique winner based on these preferences. In this paper, we study the parameterized complexity of manipulation of pairwise voting rules by a single manipulator when the votes are unweighted. The manipulator faces a graph orientation problem where the vertices correspond to the candidates and the edges correspond to the pairwise comparisons that the manipulator is allowed to make. We study the effect of various structural parameters associated with this graph on the computational complexity of the manipulation problem and provide a comprehensive classification of the complexity landscape. We also introduce a new parameter called diversity which is shown to have useful algorithmic implications.

General Terms

Algorithms, Economics, Theory

Keywords

Social Choice Theory, Voting, Manipulation, Pairwise Preferences, Parameterized Complexity

1. INTRODUCTION

One of the most well-studied questions in social choice theory [1] concerns the problem of manipulation of voting rules: given the votes of all the other voters, is it possible for a strategic voter (namely the *manipulator*) to make a preferred candidate win the election by casting a possibly non-truthful vote? Unfortunately, the celebrated Gibbard-Satterthwaite theorem [2, 3] states that strategic voting is unavoidable for any voting rule that is non-dictatorial and under which each of the three or more candidates has some chance of winning.

Inspired by the work of Bartholdi, Tovey and Trick [4], a large body of follow-up work has studied when and how the computational difficulty of finding a manipulative vote can be used as an effective workaround to this impossibility (see [5] for a survey on this topic). Much of this literature focuses on voting rules that aggregate preferences provided in the form of *complete rankings* over the entire set of candidates. This assumption, however, breaks down for large-scale settings like recommender systems that involve extremely large candidate sets (e.g. movies, products, webpages etc.). In such settings, it is much more practical to elicit *partial preferences* from the users in the form of top-k preferences [6, 7, 8], partial orders [9] etc. *Pairwise preferences* are the simplest form of partial preferences where each voter is only required to provide a set of pairwise comparisons between the candidates, without either having to compare all pairs of candidates (i.e. possibly incomplete) or provide a transitive vote (i.e. possibly cyclic).

Recent work [10] has studied the problem of manipulation of *pairwise voting rules* (i.e. voting rules that aggregate pairwise preferences) from both axiomatic and computational perspectives. It has been shown that while the impossibility of designing reasonable, non-manipulable voting rules extends to the much larger domain of pairwise preferences, computational complexity can once again provide a worst-case barrier against manipulation. The goal of our study is to develop a deeper understanding of the computational complexity results in [10] using the toolkit of *parameterized complexity analysis* [11, 12, 13, 14]. This involves a fine-grained analysis of the running time in terms of the various natural parameters associated with the problem, as opposed to a coarse dependence on the size of the input as in the classical setting [15].

Specifically, we follow the framework of [10] where the manipulator is presented with an undirected graph (called the *action space*), where each vertex corresponds to a candidate and the edges correspond to the pairs of candidates that the manipulator is allowed to compare. The task of the manipulator is to orient some or all of these edges (via votes of the form $A \succ B$, $B \succ A$ or *skip*) in order to make a favorite candidate win the election. We study how some of the natural structural parameters associated with this graph (like *vertex cover*, *feedback vertex set*, *maximum degree*, *treewidth* etc.) affect the computational complexity of the manipulation problem. Our results provide a comprehensive classification of the complexity landscape for all combinations of these parameters (see Table 1). An interesting feature of our study is the introduction of a parameter called *diversity* which, in conjunction with other structural parameters, explains a complete transition in the complexity of the manipulation problem from computational tractability (i.e. FPT, XP) to intractability (i.e. W-hardness, para-NP-hardness).

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

<table>
<tr><th></th><th>vc</th><th>pw</th><th>fvs</th><th>tw</th><th>$\emptyset$</th></tr>
<tr><td>d</td><td colspan="2">FPT</td><td colspan="2">W[1]-hard and XP</td><td rowspan="2">para-NP-complete</td></tr>
<tr><td>Δ</td><td colspan="4">FPT</td></tr>
<tr><td>$\emptyset$</td><td colspan="4">para-NP-complete</td><td></td></tr>
</table>

Table 1: *Parameterized complexity results for* pBorda-MANIPULATION *under the conditions specified by the corresponding combination of parameters. The notation $\emptyset$ is used to enable the consideration of singleton parameters. Merged cells indicate combined parameters (refer Section 2 for relevant definitions).*

Contributions

Our contributions are as listed below. We refer the reader to Table 1 for a summary of the results and to Section 2 for relevant definitions. Figure 2 shows the relationship among the various parameters considered in this study. All of our results focus on a specific pairwise voting rule called *pairwise Borda* (pBorda) which is defined later. The computational problem corresponding to the manipulation of pBorda rule is referred to as pBorda-MANIPULATION. Also note that all parameters considered by us in this study are defined with respect to the *action space* $\mathcal{A}$ of the manipulator.

1. We show that pBorda-MANIPULATION is efficiently solvable when $\mathcal{A}$ has bounded *treewidth*, subject to bounded *maximum degree* parameter. This extends the tractability result for $\mathcal{A}$ = tree graph shown in [10].
2. We also define a new parameter called *diversity* which, in combination with the parameter *vertex cover*, provides tractability even on instances where *maximum degree* can be unbounded (i.e. grow with the number of candidates).
3. Finally, we show *computational barriers* to extending the above tractability results to *any* other combination of the parameters listed in Table 1.

Organization of this paper.

We provide the relevant definitions and notation in Section 2 and describe our results and proof techniques in Section 3. We survey the related literature in Section 4 and conclude with some directions for future work in Section 5.

2. PRELIMINARIES

Our terminology and notation closely follow that of [10]. Let $[n] = \{1, 2, \ldots, n\}$ denote the set of *candidates* and $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$ denote the set of *voters* in an election.

Pairwise preferences and pairwise voting rules.

Let $\succ_u \subseteq [n] \times [n]$ denote the binary relation indicating the preferences of voter u, so that $i \succ_u j$ indicates that voter u prefers candidate i over candidate j. For each pair of candidates i, j and each voter u, we can have exactly one of $i \succ_u j$, $j \succ_u i$ or neither (i.e. voter u *skips* the comparison between i and j). We let $\mathcal{R}$ denote the set of all such anti-symmetric and irreflexive binary relations on $[n]$; and let $\Pi = (\succ_{u_1}, \succ_{u_2}, \cdots, \succ_{u_m}) \in \mathcal{R}^m$ denote the *pairwise preference profile* of the voters.

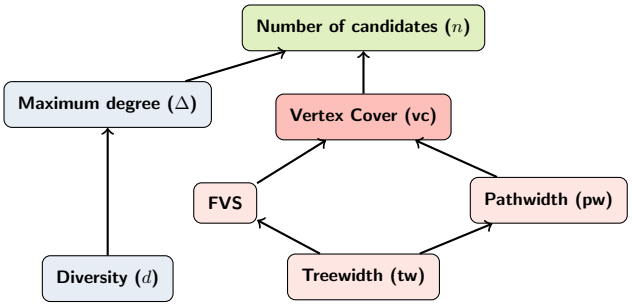


Figure 2: *A Hasse diagram depicting the relationship between the parameters. Each arrow is directed from a smaller parameter to a larger one. The implications for parameterized tractability (FPT, XP) propagate upwards along the figure in the direction of the arrows while intractability (W-hardness) propagates in the opposite direction.*

A *pairwise voting rule* r maps a pairwise preference profile $\Pi \in \cup_{k=1}^{\infty} \mathcal{R}^k$ to a unique candidate $r(\Pi) \in [n]$. Given a preference profile $\Pi \in \mathcal{R}^m$ and a pair of candidates i, j, let $m_{ij}(\Pi)$ denote the number of voters who strictly prefer candidate i over candidate j, i.e. $m_{ij}(\Pi) = \sum_{k=1}^{m} \mathbf{1}(i \succ_{u_k} j)$ where $\mathbf{1}(.)$ is the indicator function. A *score-based pairwise voting rule* is any pairwise voting rule r for which there exists a (natural) scoring function $\mathbf{s} : \cup_{k=1}^{\infty} \mathcal{R}^k \to \mathbb{R}^n$ such that $r(\Pi)$ is the highest-scoring candidate according to $\mathbf{s}(\Pi)$ under some fixed tie-breaking rule. That is, $r(\Pi) = T(\arg\max_i s_i(\Pi))$ for some tie-breaking rule $T : 2^{[n]} \setminus \{\emptyset\} \to [n]$ satisfying $T(S) \in S$ for all non-empty $S \subseteq [n]$. Some examples of score-based pairwise voting rules are as follows:

(i) *Pairwise Borda Rule* (pBorda) [16]: The pBorda score of candidate i under preference profile Π is given by[1]:

$$s_i^{\text{pBorda}}(\Pi) = \sum_{j=1}^{n} \frac{m_{ij}(\Pi)}{m_{ij}(\Pi) + m_{ji}(\Pi)}.$$

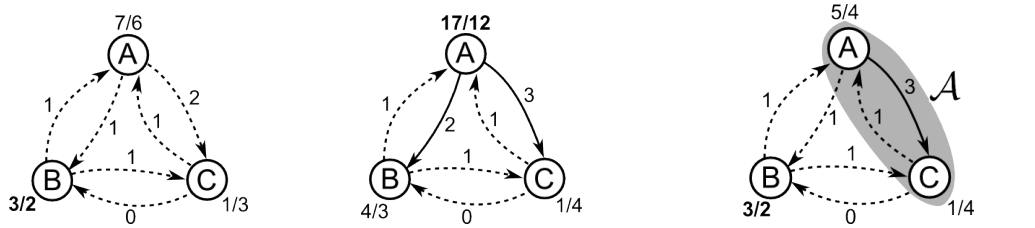
(ii) *Copeland$^\alpha$ Rule* [17]: The Copeland$^\alpha$score ($\alpha \in [0, 1]$) of candidate i under preference profile Π is given by:

$$s_i^{\text{Copeland}^\alpha}(\Pi) = \sum_{j=1}^{n} \mathbf{1}\big(m_{ij}(\Pi) > m_{ji}(\Pi)\big) + \alpha \cdot \mathbf{1}\big(m_{ij}(\Pi) = m_{ji}(\Pi)\big).$$

Manipulation of pairwise voting rules.

A pairwise voting rule r is said to be *manipulable* if there exists a pair of profiles $\Pi = (\succ_{u_1}, \ldots, \succ_{u_m}), \Pi' = (\succ_{u_1}, \ldots, \succ_{u_{m-1}}, \succ'_{u_m}) \in \mathcal{R}^m$ differing only in the preference of voter u_m such that $r(\Pi') \succ_{u_m} r(\Pi)$. That is, voter u_m (called *the manipulator*) strictly prefers the new outcome over the old one. The corresponding computational problem, referred to as r-MANIPULATION, is defined as follows:

[1] where we adopt the convention $0/0 = 0$.



(a) *Before manipulation* **(b)** *After manipulation* **(c)** *After restricted manipulation*

Figure 1: *An illustration of the election instance in Example 2.1. (a) Each vertex of the multigraph represents a candidate and each dashed edge represents the number of voters with that preference (e.g. two voters prefer $A \succ C$). (b) The pairwise comparisons made by the manipulator are represented by solid edges and the* pBorda *score of the winning candidate is indicated in boldface. (c) The restricted action space of the manipulator ($\mathcal{A} = \{(A, C)\}$) is shaded in grey.*

Definition 2.1. *r*-Manipulation

***Instance**: A tuple $\langle \Pi, i^*, \mathcal{A}, \texttt{pref-type} \rangle$ where $\Pi \in \mathcal{R}^{m-1}$ is the preference profile of the non-manipulators $(u_1, u_2, \ldots, u_{m-1})$, $i^* \in [n]$ is the distinguished candidate, $\mathcal{A} \subseteq \binom{[n]}{2}$ is the set of pairwise comparisons that the manipulator is allowed to make and* `pref-type` $\in$ {strict+acyclic, strict, acyclic, unrestricted} *is the preference constraint with respect to $\mathcal{A}$.*

***Question**: Does there exist a vote $\succ_{u_m}$ over $\mathcal{A}$ satisfying* `pref-type` *such that $r((\Pi, \succ_{u_m})) = i^*$?*

Here $\mathcal{A} \subseteq \binom{[n]}{2}$ denotes the *action space* of the manipulator i.e. the pairs of candidates that the manipulator is allowed to vote over. Alternately, no pair of candidates outside $\mathcal{A}$ can be compared by the manipulator. The parameter `pref-type` indicates whether the preferences of the manipulator over $\mathcal{A}$ are required to be *strict* (skipping comparisons is not allowed), *acyclic* (directed cycles of the form $1 \succ_u 2, 2 \succ_u 3, 3 \succ_u 1$ etc. are not allowed), *strict+acyclic* (both *strict* and *acyclic*) or *unrestricted* (no such restriction). The computational complexity of *r*-Manipulation was studied for various settings of the inputs $\mathcal{A}$ and `pref-type` in [10]. In this paper, however, we only focus on problems where `pref-type` = unrestricted and leave the study for other settings of `pref-type` as a direction for future work. The following example from [10] illustrates the role of the space $\mathcal{A}$ in the manipulation problem.

Example 2.1 (The role of action space $\mathcal{A}$). *Consider the election setting shown in Figure 1a, where the* pBorda *scores of the candidates A, B & C respectively are 7/6, 3/2 & 1/3 and B is the* pBorda *winner. Suppose we now add the manipulator u_4 to this election whose favorite candidate is A. Observe that if the manipulator casts the vote $\{(A \succ B), (A \succ C)\}$ (see Figure 1b), the new* pBorda *scores for A, B & C will be 17/12, 4/3 & 1/4 respectively and A becomes the winner. Thus, the answer to* pBorda-Manipulation *for this election instance is YES when $\mathcal{A} = \{(A, B), (A, C)\}$ or $\mathcal{A} = \{(A, B), (A, C), (B, C)\}$. If, however, the manipulator is allowed to compare only the candidates A and C (that is, $\mathcal{A} = \{(A, C)\}$), then despite voting in favor of A, the manipulator cannot make A win (Figure 1c). Therefore, the answer to* pBorda-Manipulation *is NO when $\mathcal{A} = \{(A, C)\}$.*

Excess scores.

The excess score of a candidate i is the amount by which the score of i exceeds the score of the distinguished candidate i^* in a given election. For instance, in Figure 1c, the excess pBorda scores of candidates B and C (with respect to distinguished candidate A) are $1/4$ and -1 respectively. Hence, *r*-Manipulation for a score-based voting rule r can be restated as finding a vote for the manipulator such that the final excess scores of all candidates are zero or less.

Vote configuration.

We will often use a shorthand of the form 1:3 for a pair of candidates (i, j) to denote that one voter votes $i \succ j$ while three other voters vote $j \succ i$. We will refer to 1:3 (or more generally $a : b$ for non-negative integers a, b) as the *vote configuration* between i and j.

Parameterized Complexity.

A *parameterized problem* is denoted by a pair $(Q, k) \subseteq \Sigma^* \times \mathbb{N}$. The first component Q is a classical language and the second component k is a number (called the *parameter*). Such a problem is called *fixed–parameter tractable* (FPT) if there exists an algorithm that decides it in time $\mathcal{O}(f(k)n^{\mathcal{O}(1)})$ on instances of size n.

Just as NP-hardness is used as evidence that a problem probably is not polynomial time solvable, there exists a hierarchy of complexity classes above FPT, and showing that a parameterized problem is hard for one of these classes is considered evidence that the problem is unlikely to be fixed-parameter tractable. The main classes in this hierarchy are

$$\text{FPT} \subseteq W[1] \subseteq W[2] \subseteq \cdots \subseteq W[P] \subseteq XP$$

where a parameterized problem belongs to the class XP if there exists an algorithm for it with running time bounded by $n^{g(k)}$ for some computable function g. We refer the reader to [11, 12, 13, 14] for further details.

A parameterized problem is said to be *para-NP-complete* if it is NP-complete even for constant values of the parameter. A classic example of a para-NP-complete problem is Graph Coloring parameterized by the number of colors [18] — recall that it is NP-complete to determine if a graph can be properly colored with three colors. Observe that a para-NP-complete problem does not belong to XP unless $P = NP$.

For any pair of parameterized problems A and B, we say that A is (uniformly many:1) *FPT-reducible* to B if there exist functions $f, g : \mathbb{N} \to \mathbb{N}$, a constant $\alpha \in \mathbb{N}$ and an algorithm Φ which transforms an instance (x, k) of A into an

instance $(x', g(k))$ of B in time $f(k) \cdot |x|^{\alpha}$ so that $(x, k) \in A$ if and only if $(x', g(k)) \in B$. A convenient way of showing that a problem is W[1]-hard is via an FPT reduction from a known W[1]-hard problem. Hence, in the above definition, if the problem A is known to be W[1]-hard in parameter k and there exists an FPT reduction from A to B, then B is W[1]-hard in the parameter $g(k)$.

Parameters used in this study.

Let $G = (V, E)$ denote a simple and undirected graph.

Maximum degree (Δ): The *maximum degree* of G is the maximum number of edges incident on any vertex of G.

Vertex Cover (**vc**): A set of vertices $V' \subseteq V$ is a *vertex cover* of G if for every edge $(u, v) \in E$, either $u \in V'$ or $v \in V'$ or both.

Feedback Vertex Set (**fvs**): A *feedback vertex set* of a graph is a set of vertices whose removal makes the graph acyclic.

Tree decomposition: A tree decomposition of a graph G is a tuple $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ where T is a tree and each node t of T is assigned a set of vertices $B_t \subseteq V$ (called a *bag*) such that the following hold: (i) for each vertex $v \in V$, there exists a node t such that $v \in B_t$ (alternately, $\cup_{t \in V(T)} B_t = V$); (ii) for each edge $(u, v) \in E$, there exists a node t such that $u \in B_t$ and $v \in B_t$, and (iii) for each vertex $v \in V$, the set of nodes $\{t \in V(T) : v \in B_t\}$ forms a connected subtree of T. Here $V(T)$ is the vertex set of the tree T. The *width* of a tree decomposition $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ equals $\max_{t \in V(T)} |B_t| - 1$, i.e. size of the largest bag minus one. The *treewidth* (**tw**) of a graph G is the minimum possible width of a tree decomposition of G. The notions of *pathwidth* (**pw**) and *path decomposition* are defined analogously in terms of paths.

Diversity (d): Given a preference profile Π of non-manipulators' votes, the *diversity* of the action space $\mathcal{A}$ is the maximum number of distinct score transfers that a candidate can witness due to a single pairwise comparison made by the manipulator. As an example, consider the election instance shown in Figure 1a and consider the candidate A in particular. Assuming that $\mathcal{A}$ = complete graph, the manipulator can make a pairwise comparison between any of the three pairs (A, B), (B, C) or (C, A). If the manipulator compares the pair (A, B), then the pBorda-score of candidate A can change by $+1/6$, 0 or $-1/6$ respectively, depending on whether the manipulator votes $A \succ B$, 'skip' or $B \succ A$. This can be concisely represented as a *score-transfer vector* $(+1/6, 0, -1/6)$. Similarly, the score transfer vector for candidate A for a comparison involving (A, C) or (B, C) is $(+1/12, 0, -1/6)$ or $(0, 0, 0)$ respectively. Since there are three different *kinds* of such vectors, the *diversity* for candidate A is three. The *diversity of an instance* is the maximum diversity witnessed by any candidate. Notice that for a given election, *diversity* can be $\Theta(n)$ under the pBorda rule while the same for Copeland$^{\alpha}$ is $\mathcal{O}(1)$ due to the limited types of score exchanges permitted under the definition of Copeland voting rule. For any pairwise voting rule where a pairwise comparison by the manipulator can only affect the scores of the two candidates involved (examples include pBorda and Copeland$^{\alpha}$), *diversity* is at most the *maximum degree* Δ.

Elimination problem in sports.

The sports elimination problem [19] asks whether a team i^* can still win a sports competition, given the current scores of the teams and the set of games to be played between them. Sports competitions are often scored according to a *scoring system*, which specifies how many points are awarded to the home and the away teams depending on the outcome of a game between them. For example, the well-known European football scoring system, denoted by $\mathrm{S} = [(3, 0), (1, 1), (0, 3)]$, awards 3 points for win, 1 point for draw and 0 for loss, regardless of the home-away distinction. Similarly, the system $\mathrm{S} = [(3, 0), (1, 2), (0, 3)]$ provides an extra point to an away team under a draw outcome. The computational problem corresponding to the above question, called S-Elimination, is defined as follows [20]:

Definition 2.2. S-Elimination

Instance: *A tuple* $\langle \mathbf{s}, i^*, \mathcal{G} \rangle$ *where* $\mathbf{s} = (s_1, s_2, \ldots, s_N)^T$ *is the vector of current scores of the* N *teams,* $i^* \in [N]$ *is a distinguished team and* $\mathcal{G} \subseteq \binom{[N]}{2}$ *is the set of remaining games between the teams.*

Question: *Does there exist an assignment of outcomes for the games in* $\mathcal{G}$ *such that* i^* *ends up with the (joint) highest total score among all teams under the scoring system* S*?*

Partition.

Definition 2.3. Partition

Instance: *A multiset* $A = \{a_1, a_2, \ldots, a_N\}$ *of* N *positive integers.*

Question: *Does there exist a partition of* A *into the sets* A_1 *&* A_2 *such that* $\sum_{a_i \in A_1} a_i = \sum_{a_j \in A_2} a_j = \frac{1}{2} \sum_{a_k \in A} a_k$*?*

Partition is a well-known NP-complete problem [18]. We assume without loss of generality that $a_1 \leq a_2 \leq \cdots \leq a_N$.

Capacitated Dominating Set.

Definition 2.4. Capacitated Dominating Set

Instance: *A triple* $\langle G, c, k \rangle$ *where* $G = (V, E)$ *is a graph,* $c : V \to \mathbb{N}$ *is a capacity function for the vertices of* G *and* k *is a positive integer.*

Question: *Does there exist a set of vertices* $V' \subseteq V$ *of size at most* k *in* G *such that each vertex* $v \in V \setminus V'$ *is adjacent to some vertex* $v' \in V'$ *and no vertex* $v' \in V'$ *is adjacent to more than* $c(v')$ *vertices in* $V' \setminus V$*?*

Capacitated Dominating Set was shown to be W[1]-hard when simultaneously parameterized by the *treewidth* and solution size k [21]. In fact, the problem remains W[1]-hard when simultaneously parameterized by the *pathwidth* and the size of the *feedback vertex set* of the graph G, even on instances with only constantly many distinct capacities[2].

[2]This can be shown by carrying out the reduction in [21] while starting from Multicolored Clique on regular graphs and observing that the parameters *pathwidth*, *treewidth* and *feedback vertex set* of the reduced Capacitated Dominating Set instance are all $\mathcal{O}(k^4)$ in size.

3. OUR RESULTS AND TECHNIQUES

Our classification result for the parameterized complexity of pBorda-Manipulation for any combination of the considered parameters is summarized by Theorem 3.1 and Table 1. We assume throughout that `pref-type`=unrestricted.

In the parameterized studies of computational problems that arise in the context of voting, a commonly used parameter is the *number of candidates* (n) [22, 23, 24, 25, 26, 27]. We observe that for any pairwise voting rule that is easy to evaluate, the problem of manipulation by a single manipulator is trivially FPT for this choice of parameter, because even a brute-force search over all possible votes of the manipulator will yield the desired running time (i.e. $\mathcal{O}(3^{n^2})$). The other natural choice of parameter is the *number of voters*. However, we know from [10] that pBorda-Manipulation is NP-complete *even with twelve non-manipulators*. Given the extreme behaviors on the two obvious choices of parameters, we turn to the action space of the manipulator $\mathcal{A}$ and try to understand how the problem complexity is influenced by parameters associated with the structure of $\mathcal{A}$.

We start by recalling the result in [10] which states that that pBorda-Manipulation is efficiently solvable when $\mathcal{A}$ is a tree/forest/graph with maximum degree two. Given this result, we follow the "distance from triviality" approach in parameterized analysis [28] and consider parameters that measure how *far* $\mathcal{A}$ is from the class of tractable instances i.e. degree of closeness to being a tree or a forest. This motivates the study of parameters like *treewidth* (**tw**), *feedback vertex set* (**fvs**) and *maximum degree* (Δ); and upper/lower bounds on these parameters like *pathwidth* (**pw**) and *vertex cover* (**vc**) (refer Section 2 for formal definitions). Interestingly, a similar set of parameters was recently used in the parameterized complexity analysis of the closely-related S-Elimination problem [29], and studying their influence on the complexity of pBorda-Manipulation allows us to compare the complexity landscapes of the two problems, as we will see.

Our first set of results shows that the manipulation problem is, somewhat surprisingly, *para-NP-complete* for (i) the *maximum degree* parameter, and (ii) *any combination of the parameters in* {**vc**, **fvs**, **pw**, **tw**}. This already establishes a contrast with S-Elimination which was shown to be in XP when parameterized by the *treewidth* of the graph formed by the set of remaining games [29].

On the other hand, pBorda-Manipulation is FPT when simultaneously parameterized by *maximum degree* and *any combination of the other parameters*. We ask if there is a *natural* parameter that is, in general, smaller than *maximum degree*, but that can still provide tractability when combined with some of the other structural parameters. We discover an answer in the form of a novel parameter called *diversity* (d), which is a measure of how many different types of score exchanges the manipulator encounters for any candidate. Unfortunately, it turns out that pBorda-Manipulation is NP-complete on graphs with constant *diversity*; in fact, it remains NP-complete even when the sum of *diversity* and *maximum degree* is bounded by a constant [10]. On the positive side, we show that *diversity*, when combined with *vertex cover*, leads us to an FPT algorithm, while we obtain XP algorithms by combining it with any of the other parameters in {**fvs**, **pw**, **tw**}. We do not expect to improve this XP result, as the problem remains W[1]-hard in those cases.

We now state our main result (Theorem 3.1) that summarizes the findings described above (see also Table 1).

Theorem 3.1. *Let $\mathcal{P} = \{$**vc**, **pw**, **fvs**, **tw**, $\Delta, d\}$ denote the set of parameters defined over the action space $\mathcal{A}$ of an instance of* pBorda-Manipulation. *Let $\mathcal{X}$ denote the set $\{$**vc**, **pw**, **fvs**, **tw**$\}$ and $\mathcal{Y}$ denote the set $\{d, \Delta\}$. Then*

1. *For any $\mathcal{Q} \subseteq \mathcal{X}$ or $\mathcal{Q} \subseteq \mathcal{Y}$,* pBorda-Manipulation *is* NP-complete *even when the sum of all parameters in $\mathcal{Q}$ is bounded by a constant.*
2. *For all $\mathcal{Q} \subseteq \mathcal{P}$,* pBorda-Manipulation *parameterized by $\mathcal{Q}$ is in XP if $\mathcal{Q}$ contains d along with any element of $\mathcal{X}$. Further, the problem is* FPT *if $\mathcal{Q}$ contains Δ along with any element of $\mathcal{X}$, or if it contains both d and* **vc**.
3. *In the remaining case when $\mathcal{Q} \subseteq \mathcal{P}$ does not contain either Δ or* **vc**, pBorda-Manipulation *is* W[1]-hard *parameterized by $\mathcal{Q}$, even on instances where d is bounded by a constant.*

We briefly summarize these results and their implications.

(i) We show that pBorda-Manipulation remains NP-complete even for instances where $\mathcal{A}$ has a *vertex cover* of size two (Theorem 3.2). Since a bound on the size of the *vertex cover* implies a bound on the size of the *feedback vertex set*, *pathwidth* and *treewidth*, we have NP-completeness of pBorda-Manipulation even when parameterized by all parameters in $\mathcal{X}$ combined. Together with the result from [10] showing NP-hardness of pBorda-Manipulation on instances of *maximum degree* $\Delta = 3$ (and therefore *diversity* $d \leq 3$), this implies statement 1 of Theorem 3.1.

(ii) We use dynamic programming over tree decompositions to show that pBorda-Manipulation is FPT when parameterized by *maximum degree* and *treewidth* (Theorem 3.4). Since all other parameters in $\mathcal{X}$ are larger than *treewidth*, this gives an FPT result when Δ is combined with any subset of parameters in $\mathcal{X}$. Similarly, we use Lenstra's result [30] on Integer Linear Programming being FPT in the number of variables to show that pBorda-Manipulation is FPT when simultaneously parameterized by the *vertex cover* and *diversity* of $\mathcal{A}$ (Theorem 3.5). These two results together imply statement 2 of Theorem 3.1.

(iii) Finally, we show that pBorda-Manipulation is W[1]-hard when simultaneously parameterized by the *feedback vertex set* and *pathwidth* of $\mathcal{A}$ via an FPT-reduction from Capacitated Dominating Set [21]. This proves statement 3 part of Theorem 3.1.

We now provide formal statements and proofs for the results stated above. Our first result shows that pBorda-Manipulation is para-NP-complete in *vertex cover*.

Theorem 3.2. pBorda-Manipulation *is* NP-complete *when $\mathcal{A}$ is a general graph with a vertex cover of size two and* `pref-type` *= unrestricted.*

Proof. The problem is clearly in NP. We show NP-hardness by reduction from Partition.

Construction of the reduced instance: Given an instance $A = \{a_1, a_2, \ldots, a_N\}$ of Partition, we construct

an instance $\langle \Pi, i^*, \mathcal{A}, \texttt{pref-type} \rangle$ of pBorda-Manipulation as follows: the *set of candidates* consists of (i) the *selector* candidates X and Y, (ii) a candidate i for each positive integer $a_i \in A$ (called the *integer* candidates), (iii) the *distinguished candidate* i^* and (iv) the *dummy candidates* $D_1, D_2, \ldots, D_{4N}$ (hence $n = 5N + 3$ candidates overall). The *action space* $\mathcal{A}$ is the complete bipartite graph between the *selectors* and the *integer* candidates. That is, $\mathcal{A} = \{\cup_{i \in [N]} \{(X, i) \cup (Y, i)\}\}$. The *set of voters* consists of $2Q$ non-manipulators (where $Q = (2a_N + 1) \cdot (2a_N + 2)$) and one manipulator. The *votes of the non-manipulators* between the *selectors* and the *integer* candidates are set up in order to ensure that the score transfers resulting from the manipulator's vote are a_i/Q (if the manipulator votes $X \succ i$ or $Y \succ i$) and $(2a_N + 1 - a_i)/Q$ (if the manipulator votes $i \succ X$ or $i \succ Y$). Specifically, for each $i \in [N]$, both the candidate pairs (i, X) and (i, Y) are in $a_i : (2a_N + 1 - a_i)$ configuration. The *votes involving dummy candidates* are set up as follows: for each $i \in [N]$, the pair (i^*, D_i) is in $(2a_N + 1 - a_i) : a_i$ configuration while the pair (i^*, D_{N+i}) is in $a_i : (2Q - a_i)$ configuration. For each $i \in [N]$, the pair (i, D_{2N+i}) is in $(2a_N + 1 - 3a_i) : 3a_i$ configuration while the pair (i, D_{3N+i}) is in $2a_i : (2Q - 2a_i)$ configuration. Finally, for each $i \in [N]$ and each $k \in [N] \setminus i$, the pair (i, D_{2N+k}) is in $(2a_N + 1 - a_k) : a_k$ configuration while the pair (i, D_{3N+k}) is in $a_k : (2Q - a_k)$ configuration.

It is easy to check that the excess score of each *integer* candidate i after this construction is $a_i/2Q$, while that of each *selector* is $\frac{-1}{2Q} \sum_{a_k \in A} a_k$. Also note that the *selector* vertices constitute a *vertex cover* of $\mathcal{A}$ of size two.

Equivalence of solutions: ($\Rightarrow$) Suppose there exists a partition of A into the sets A_1 and A_2. A valid manipulative vote can be constructed from this partition as follows: for each $i \in [N]$, the manipulator votes $X \succ i$ if $a_i \in A_1$ or $Y \succ i$ if $a_i \in A_2$ and skips all other comparisons. The final excess score of each *integer* candidate i is negative, since $\frac{a_i}{2Q} - \frac{a_i}{Q} < 0$. The final excess score for each *selector* equals 0 due to the partition property, making i^* the winner.

($\Leftarrow$) Suppose there exists a valid manipulative vote that makes i^* win. Then, without loss of generality, each *integer* candidate i must lose at least one of its two pairwise comparisons in $\mathcal{A}$ in order to get rid of its positive excess score (Observation 1). Similarly, no *integer* candidate i can win either of its pairwise comparisons against any of the *selectors* or otherwise it accumulates an excess that it cannot offload any further (Observation 2). Observation 1 implies that the combined pBorda score that gets transferred from the *integer* candidates to the *selectors* is at least $\sum_{a_k \in A} \frac{a_k}{Q}$. Observation 2 implies that no pBorda score gets transferred in the reverse direction. Since the *selectors* can together handle an influx of at most $\sum_{a_k \in A} \frac{a_k}{Q}$, each *integer* candidate i must lose to exactly one of the *selectors* while the other comparison is skipped. A partition can now be naturally inferred from such a vote. □

Remark 3.1. *An implication of Theorem 3.2 is a separation of the problems of* pBorda-Manipulation *and* S-Elimination *in terms of their computational complexity. As mentioned earlier,* S-Elimination *was shown to be in XP when parameterized by the treewidth of the graph formed by the set of remaining games [29] while* pBorda-Manipulation *is para-NP-complete in the same parameter. Hence,* pBorda-Manipulation *is necessarily harder than* S-Elimination *unless* $P = NP$.

Our next result establishes the W[1]-hardness of pBorda-Manipulation in terms of the size of the *feedback vertex set* and the *pathwidth* of $\mathcal{A}$, even on instances where the *diversity* of $\mathcal{A}$ is bounded by a constant.

Theorem 3.3. pBorda-Manipulation *is* W[1]-hard *when simultaneously parameterized by feedback vertex set and pathwidth of* $\mathcal{A}$ *when* $\mathcal{A}$ = *general graph with constant diversity and* `pref-type` = *unrestricted.*

Proof. We show an FPT reduction from Capacitated Dominating Set. Recall from Section 2 that Capacitated Dominating Set is W[1]-hard when simultaneously parameterized by the *feedback vertex set* and *pathwidth* of the input graph even on instances with only a constant number of distinct capacities.

Construction of the reduced instance: Given an instance $\langle G = (V, E), c, k \rangle$ of Capacitated Dominating Set, we construct an instance $\langle \Pi, i^*, \mathcal{A}, \texttt{pref-type} \rangle$ of pBorda-Manipulation as follows: the *set of candidates* consists of (i) the *source* X and the *sink* Y, (ii) a candidate v_i for each vertex in G (the *vertex* candidates), (iii) a candidate e_i for each edge in G (the *edge* candidates), (iv) the *distinguished* candidate i^* and (v) the *dummy* candidates $D_1, D_2, \ldots, D_\ell$ where $\ell = 7|V| + 2|E| + \Delta + 2 - 2k$ and Δ is the maximum degree of graph G. Hence, $n = 8|V| + 3|E| + \Delta - 2k + 5$. The *action space* $\mathcal{A}$ is the union of all unordered pairs of candidates connected by dashed edges in Figure 3. That is,

$$\mathcal{A} = \big\{ \{\cup_{i \in [|V|]} (X, v_i)\} \bigcup \{\cup_{j \in [|E|]} (e_j, Y)\} \bigcup \{\cup_{i \in [|V|], j \in [|E|]} (v_i, e_j) \text{ where } v_i \text{ is adjacent to } e_j \text{ in } G\} \big\}.$$

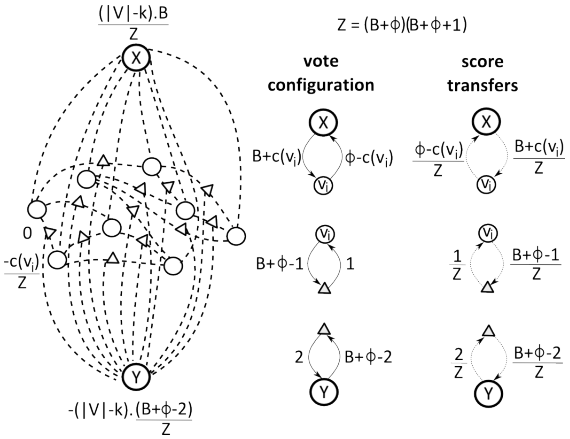


Figure 3: *This figure shows the reduced* pBorda-Manipulation *instance (excluding the dummy candidates and* i^**) constructed from the given* Capacitated Dominating Set *instance. The action space of the manipulator* $\mathcal{A}$ *is shown on the left via dashed lines along with the excess* pBorda *scores. The* vertex *candidates are indicated by circles in the middle layer and the* edge *candidates are shown as triangles. The right side shows the configuration of votes of the non-manipulators and the resulting scores transfers.*

The *set of voters* consists of Z non-manipulators (where $Z = (B + \phi) \cdot (B + \phi + 1)$, $B = |V| + |E| +$

$\sum_{v_i \in V(G)} c(v_i)$ and $\phi = \max_{i \in |V|} c(v_i)$) and one manipulator. The *votes corresponding to action space* $\mathcal{A}$ are set up as follows: for each $i \in [|V|]$, the candidate pair (X, v_i) is in $(B + c(v_i)) : (\phi - c(v_i))$ configuration. For each $i \in [|V|]$ and each $j \in [|E|]$ such that v_i is adjacent to e_j in G, the candidate pair (v_i, e_j) is in $(B + \phi - 1) : 1$ configuration. Finally, for each $j \in [|E|]$, the candidate pair (e_j, Y) is in $2 : (B + \phi - 2)$ configuration. We omit the description of the votes involving dummy candidates due to space limitations and remark that they are only used to calibrate the excess scores of candidates as shown in Figure 3. This finishes the construction of the election instance.

Note that the reduction is *efficient* since it uses $\mathcal{O}((|V| + |E|)^2)$ voters and $\mathcal{O}(|V| + |E|)$ candidates. Also note that the reduction is *parameter preserving*, in that if the *pathwidth* and size of the optimal *feedback vertex set* of G are w and t, then the same parameters for the action space $\mathcal{A}$ are $\mathcal{O}(w^2)$ and $(t + 2)$ respectively. Furthermore, the *diversity* of the reduced election instance is a constant since the original Capacitated Dominating Set instance only has $\mathcal{O}(1)$ distinct capacity values.

Equivalence of solutions: ($\Rightarrow$) Suppose $S \subseteq V(G)$ is a valid capacitated dominating set. Then a valid manipulative vote can be constructed as follows: first, the manipulator triggers score transfers from *source* X to *vertex* candidates in $V(G) \setminus S$ with a vote $v_i \succ X$ for all $v_i \in V(G) \setminus S$. This brings the excess score of X below zero and results in an excess of B/Z for each $v_i \in V(G) \setminus S$. Next, for each $v_i \in V(G) \setminus S$, the manipulator votes $e_j \succ v_i$ for exactly one *edge* candidate e_j that connects v_i to a candidate $v_i' \in S$ that v_i is *assigned to*. This results in a negative excess score for all *vertex* candidates in $V(G) \setminus S$ while each *edge* candidate e_j chosen above by the manipulator now acquires an excess score of $(B + \phi - 1)/Z$. Finally, for each such *edge* candidate e_j, the manipulator votes $Y \succ e_j$ and $v_i' \succ e_j$. It is easy to check that after this step no candidate in $\mathcal{A}$ has positive excess score, making i^* the pBorda winner.

($\Leftarrow$) Suppose there exists a valid manipulative vote that makes i^* win. Then, without loss of generality, X must lose against at least $(|V| - k)$ *vertex* candidates in order to offload its excess. Call this set S'. Hence, $|S'| \geq |V| - k$ and each candidate in S' acquires an excess of B/Z as a result. Next, observe that any candidate in S' can only offload its excess score to the *edge* candidates adjacent to it in G. As a result, each *edge* candidate affected in this manner (there must be at least $(|V| - k)$ such edge candidates overall) acquires an excess of $(B + \phi - 1)/Z$, that it must offload to the *sink* Y and the other *vertex* candidate adjacent to it. Hence, the *sink* Y suffers a total inflow of at least $(|V| - k) \cdot (B + \phi - 2)/Z$ from the affected *edge* candidates. By design, this is also the maximum inflow that the *sink* can handle without gaining positive excess. Therefore, the set S' must consist of exactly $(|V| - k)$ *vertex* candidates such that each candidate in S' is connected to a candidate in $V(G) \setminus S'$ via an *edge* candidate. Besides, no candidate in $V(G) \setminus S'$ can be adjacent to more than $c(v_i)$ candidates in S', or else there will be no means for this candidate to offload its own excess score. Therefore, the set $V(G) \setminus S'$ constitutes a capacitated dominating set of G. □

Remark 3.2. *Except for scoring systems of the form* $\mathrm{S} = \{(i, t - i) : 0 \leq i \leq t\}$ *for some* $t \in \mathbb{N}$, S-Elimination *was shown to be* W[1]*-hard in the parameters feedback vertex set and pathwidth via separate (although similar) proofs in [29]. By instantiating* S-Elimination *for* $\mathrm{S} = [(3, 0), (1, 2), (0, 3)]$ *and using the observation in [10] that* S-Elimination *becomes a special case of* pBorda-Manipulation *under such instantiation, one can alternately derive* W[1]*-hardness of* pBorda-Manipulation *in terms of the two parameters individually. By contrast, Theorem 3.3 provides a single proof for showing* W[1]*-hardness in the two parameters simultaneously.*

Our first algorithmic result shows that pBorda-Manipulation is FPT when simultaneously parameterized by the *treewidth* and *maximum degree* of $\mathcal{A}$. We show this by the standard dynamic programming procedure over a given tree decomposition [14] and omit the detailed proof due to space limitations.

Theorem 3.4. pBorda-Manipulation *is solvable in time* $\mathcal{O}\big(\Delta^{\mathcal{O}(dw^2)}(n \log m)^{\mathcal{O}(1)}\big)$, *where* Δ, w *and* d *denote the maximum degree, treewidth and diversity of* $\mathcal{A}$ *respectively.*

Note that since $d \leq \Delta$ for pBorda rule, the running time above is FPT in *maximum degree* Δ and the *treewidth* of $\mathcal{A}$. Further, since $\Delta \leq n$, the running time is also XP with respect to the *diversity* d and *treewidth* w of $\mathcal{A}$. We restate these observations as the following corollary.

Corollary 3.1. pBorda-Manipulation *is* FPT *when parameterized by the maximum degree and the treewidth of* $\mathcal{A}$*; and in* XP *when parameterized by the diversity and the treewidth of* $\mathcal{A}$.

Our next algorithmic result pertains to graphs of bounded *vertex cover* number and bounded *diversity*.

Theorem 3.5. pBorda-Manipulation *is solvable in time* $\mathcal{O}\big(f(k, d)(n \log m)^{\mathcal{O}(1)}\big)$, *where* k *and* d *denote the size of a vertex cover and diversity of* $\mathcal{A}$ *respectively and* f *is a computable function.*

Proof. The proof proceeds by partitioning the vertices of the independent set of $\mathcal{A}$ into equivalence classes based on their interactions with the vertex cover, and exploiting a size bound on the number of such equivalence classes in the subsequent ILP formulation.

Specifically, let $S \subseteq V(\mathcal{A})$ be a vertex cover of $\mathcal{A}$ of size k and let $I = V(\mathcal{A}) \setminus \mathcal{S}$ be the corresponding independent set. For any $T \subseteq S$, let $I_T \subseteq I$ denote the set of all vertices in I whose neighborhood within $\mathcal{A}$ is exactly the set T. Next, given $T = \{v_1, \ldots, v_t\} \subseteq S$ and a vector $E_T = \langle(\alpha_1, \beta_1), \ldots, (\alpha_t, \beta_t)\rangle$ consisting of pairs of non-negative integers (α_i, β_i), define the *equivalence class* $I_{T,E_T} \subseteq I_T$ such that for any pair of vertices $v_i \in T$ and $u \in I_{T,E_T}$, the candidate pair (v_i, u) is in $\alpha_i : \beta_i$ configuration with respect to the votes of non-manipulators. Note that since the diversity d of the instance is bounded, any vertex of the independent set must belong to exactly one of at most $2^k \cdot d^k$ equivalence classes.

Call a vertex $u \in I_{T,E_T}$ *safe* with respect to a vector $z \in \{-1, 0, 1\}^{|T|}$ if the excess score of u is zero or less for the following vote of the manipulator: for each $v_i \in T$, $u \succ v_i$ if $z(i) = +1$; 'skip' the comparison (u, v_i) if $z(i) = 0$ and $v_i \succ u$ if $z(i) = -1$. Note that fixing the manipulator's vote on all pairwise comparisons in $\mathcal{A}$ involving u fixes the pBorda score of u. Similarly, define the *safety-set* of a vertex $u \in I_{T,E_T}$ as the set of all vectors $z \in \{-1, 0, 1\}^{|T|}$ with

respect to which u is *safe*. We say that a vertex $u \in I_{T,E_T}$ *sees* a vote $z \in \{-1,0,1\}^{|T|}$ if z is the restriction of the manipulator's vote to the pairwise comparisons in $\mathcal{A}$ involving u.

Given an equivalence class I_{T,E_T} and the safety-set for each $u \in I_{T,E_T}$, define a *safe-subclass* as the set of all vertices in I_{T,E_T} with identical safety-sets. Denote the number of safe-subclasses in I_{T,E_T} by N_{T,E_T}. Thus, $N_{T,E_T} \leq 3^k$.

We now claim that any valid solution to pBorda-Manipulation can be transformed into another (possibly different) solution where all vertices inside a safe-subclass *see* the same vote vector. Indeed, fix a safe-subclass and let z' be the restriction of a valid vote $\succ$ as *seen* by the vertex with the highest excess score in that safe-subclass. An alternate vote can now be constructed as follows: in the original vote $\succ$, replace the vote vector currently *seen* by each vertex inside the given safe-subclass by z', while keeping the rest of the vote unchanged. It is easy to check that the excess score constraints for all vertices continue to remain satisfied in the new vote. Therefore, without loss of generality, all vertices inside a safe-subclass *see* the same vote vector in a valid vote of the manipulator.

Our algorithm takes as input an instance of pBorda-Manipulation, namely $\langle \Pi, i^*, \mathcal{A} \rangle$ and returns a YES/NO output indicating the existence of a valid manipulative vote (along with a valid vote, if one exists). The algorithm starts by *guessing* the manipulator's vote within the vertex cover (call this guess $\succ_{\mathcal{S}}$). There are at most $\binom{k}{2}$ such pairs, hence the total number of choices is at most $3^{O(k^2)}$. For each such guess, we obtain a new instance of pBorda-Manipulation, namely $\langle \Pi', i^*, \mathcal{A}' \rangle$ where Π' is a voting profile representing the original votes of the non-manipulators combined with the manipulator's vote $\succ_{\mathcal{S}}$ over the vertex cover and $\mathcal{A}'$ represents the restriction of the graph $\mathcal{A}$ to the bipartite subgraph $\mathcal{S} \times \mathcal{I}$. The algorithm now uses ILP to solve this new problem for each equivalence class in parallel, and checks if the combined vote constitutes a valid solution.

Formulating the ILP: We now describe the variables and constraints for the ILP.

Variables: For each subset $T \subseteq S$, each score vector $E_T = \langle (\alpha_1, \beta_1), \ldots, (\alpha_t, \beta_t) \rangle$, each $1 \leq p \leq 3^{|T|}$ and each $1 \leq q \leq N_{T,E_T}$, define a binary variable $Z_{T,E_T,p,q} \in \{0,1\}$. Here $Z_{T,E_T,p,q} = 1$ (respectively 0) indicates that given T, E_T and the induced equivalence class I_{T,E_T}, the safe-subclass indexed by q *sees* (respectively *does not see*) the vote vector indexed by p. Thus, there are at most $2^k \cdot d^k \cdot 3^{2k}$ variables overall. In other words, the number of variables depends only on the parameters d and k.

Constraints: Our ILP has three types of constraints:

(i) *Sanity constraints*:

(a) $Z_{T,E_T,p,q} \in \{0,1\}$ for all T, E_T, p and q.

(b) for every T, E_T and q, $\sum_p Z_{T,E_T,p,q} = 1$ (i.e. each safe-subclass *sees* exactly one vote vector).

(ii) *Excess score constraints for the vertex cover*: for each vertex $v_i \in S$

$$\begin{aligned}\sum_{T \in T_v} \sum_{E_T} \sum_q \sum_p Z_{T,E_T,p,q} \cdot \mathbf{1}(p_i = 0) \cdot \tfrac{\alpha_i}{\alpha_i+\beta_i} \cdot |q| \\ + Z_{T,E_T,p,q} \cdot \mathbf{1}(p_i = -1) \cdot \tfrac{\alpha_i+1}{\alpha_i+\beta_i+1} \cdot |q| \\ + Z_{T,E_T,p,q} \cdot \mathbf{1}(p_i = +1) \cdot \tfrac{\alpha_i}{\alpha_i+\beta_i+1} \cdot |q| \leq s^*\end{aligned}$$

where $T_v = \{T \subseteq S \mid v \in T\}$ and $|q|$ represents the cardinality of the safe-subclass q. The latter can be efficiently precomputed.

(iii) *Excess score constraints for the independent set*: for all T, E_T, p and q

$$Z_{T,E_T,p,q} \leq Z^{\text{safe}}_{T,E_T,p,q}$$

where $Z^{\text{safe}}_{T,E_T,p,q} \in \{0,1\}$ is a (precomputed) binary indicator specifying whether, given T, E_T and the induced equivalence class I_{T,E_T}, the vector p belongs to the safety-set of (any vertex in) the safe-subclass q.

The theorem now follows since ILP feasibility is FPT when parameterized by the number of variables [30], which, as remarked earlier, is a function of d and k alone. □

Remark 3.3. *The proof techniques used in our algorithmic results (Theorems 3.4 and 3.5) can be readily applied to* S-Elimination *to recover the corresponding results in [29].*

4. RELATED WORK

Parameterized complexity analysis has proven extremely useful in scrutinizing the computational behavior of a variety of problems in computational social choice, namely *winner-determination* [22, 31, 24, 32], *manipulation* [23, 33, 34, 26, 27, 35], *bribery* [36, 37, 38], *possible and necessary winner problems* [39, 40, 41], etc. We refer the reader to [42, 43] for detailed surveys on this topic.

Among the studies on the parameterized complexity of manipulation of standard voting rules, our work shares the spirit of [34, 35] where parameterization of the preference domain (in their case, in terms of closeness to single-peakedness) was used to show special-case tractability results. Specifically, [34] showed that unweighted Borda manipulation with two manipulators is efficiently solvable over the domain of single-peaked preferences, although the problem is known to be NP-complete over the unrestricted domain [44, 33].[3] This result was later generalized in [35] where the manipulation problems for Borda and Copeland$^\alpha$ rules by two manipulators were shown to be FPT in the parameter *single-peaked width* (which measures the distance of a preference profile from single-peakedness).

5. CONCLUDING REMARKS

We studied the problem of manipulation in the model of pairwise preferences and gave a complete classification of the parameterized complexity of manipulating the pairwise Borda rule in terms of various natural parameters relating to the action space. This involved the introduction of *diversity* as a parameter, which we demonstrated to be useful from an algorithmic perspective.

Our work opens up two very natural directions for future work. First, the parameterized complexity of pBorda-Manipulation for other settings of the parameter `pref-type` remains to be analyzed. Second, it would be interesting to compare the parameterized behavior of pBorda rule with that of other pairwise voting rules like Copeland$^\alpha$ [17] (the classical complexity landscape for this family of rules was described in [10]), PageRank [45], HodgeRank [46], Ranked Pairs, Schulze's rule [47] etc.

[3] Recall that the problem of unweighted Borda manipulation by a single manipulator was shown to be efficiently solvable over the domain of rankings in [4].

6. ACKNOWLEDGEMENTS

The authors thank the anonymous reviewers for their helpful comments. NM was supported by a DST INSPIRE Faculty grant [IFA12-ENG-31]. RV thanks Google and Microsoft Research for travel grants.

REFERENCES

[1] Kenneth J Arrow, Amartya Sen, and Kotaro Suzumura. *Handbook of Social Choice & Welfare*, volume 2. Elsevier, 2010.

[2] A. Gibbard. Manipulation of Voting Schemes: a General Result. *Econometrica: Journal of the Econometric Society*, pages 587–601, 1973.

[3] M.A. Satterthwaite. Strategy-proofness and Arrow's Conditions: Existence and Correspondence Theorems for Voting Procedures and Social Welfare Functions. *Journal of Economic Theory (JET)*, 10(2):187–217, 1975.

[4] John J. Bartholdi III, Craig A. Tovey, and Michael A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[5] Piotr Faliszewski and Ariel D. Procaccia. AI's war on manipulation: Are we winning? *AI MAGAZINE*, 2010.

[6] Nir Ailon. Aggregation of partial rankings, p-ratings and top-m lists. *Algorithmica*, 57(2):284–300, 2010.

[7] Nina Narodytska and Toby Walsh. The Computational Impact of Partial Votes on Strategic Voting. In *Proceedings of the 21st European Conference on Artificial Intel ligence (ECAI)*, pages 657–662, 2014.

[8] Vijay Menon and Kate Larson. Complexity of Manipulation in Elections with Top-truncated Ballots. *arXiv preprint arXiv:1505.05900*, 2015.

[9] Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Aggregating Partially Ordered Preferences. *Journal of Logic and Computation*, 19(3):475–502, June 2009.

[10] Rohit Vaish, Neeldhara Misra, Shivani Agarwal, and Avrim Blum. On the computational hardness of manipulating pairwise voting rules. In *Proceedings of the Fifteenth International Conference on Autonomous Agents and Multiagent Systems (to appear)*, 2016.

[11] R. G. Downey and M. R. Fellows. *Parameterized complexity*. Springer-Verlag, New York, 1999.

[12] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2006.

[13] R. Niedermeier. *Invitation to fixed-parameter algorithms*, volume 31 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2006.

[14] Marek Cygan, Fedor V Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[15] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[16] Arun Rajkumar and Shivani Agarwal. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *Proceedings of the 31st International Conference on Machine Learning*, pages 118–126, 2014.

[17] Piotr Faliszewski, Edith Hemaspaandra, and Henning Schnoor. Copeland voting: Ties matter. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 983–990. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[18] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[19] Benjamin L Schwartz. Possible winners in partially completed tournaments. *SIAM Review*, 8(3):302–308, 1966.

[20] Walter Kern and Daniël Paulusma. The Computational Complexity of the Elimination Problem in Generalized Sports Competitions. *Discrete Optimization*, 1(2):205–214, November 2004.

[21] Michael Dom, Daniel Lokshtanov, Saket Saurabh, and Yngve Villanger. Capacitated domination and covering: A parameterized perspective. In *Parameterized and Exact Computation*, pages 78–90. Springer, 2008.

[22] John J. Bartholdi III, Craig A. Tovey, and Michael A. Trick. Voting Schemes for which It Can Be Difficult to Tell Who Won the Election. *Social Choice and Welfare*, 6(2):157–165, 1989.

[23] Vincent Conitzer, Tuomas Sandholm, and Jérôme Lang. When are elections with few candidates hard to manipulate? *Journal of the ACM*, 54(3):14, 2007.

[24] Nadja Betzler, Michael R Fellows, Jiong Guo, Rolf Niedermeier, and Frances A Rosamond. Fixed-parameter algorithms for Kemeny rankings. *Theoretical Computer Science*, 410(45):4554–4570, 2009.

[25] Piotr Faliszewski, Edith Hemaspaandra, Lane A Hemaspaandra, and Jörg Rothe. Llull and Copeland voting computationally resist bribery and constructive control. *Journal of Artificial Intelligence Research*, pages 275–341, 2009.

[26] Yongjie Yang. Election Attacks with Few Candidates. In *ECAI*, pages 1131–1132, 2014.

[27] Lirong Xia. Fixed-parameter tractability of integer generalized scoring rules. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, pages 1599–1600. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[28] Jiong Guo, Falk Hüffner, and Rolf Niedermeier. A Structural View on Parameterizing Problems: Distance from Triviality. In *Parameterized and Exact Computation*, pages 162–173. Springer, 2004.

[29] Katarina Cechlárová, Eva Potpinková, and Ildikó Schlotter. Refining the complexity of the sports elimination problem. *Discrete Applied Mathematics*, 199:172 – 186, 2016.

[30] Hendrik W Lenstra Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, 1983.

[31] Nadja Betzler, Jiong Guo, and Rolf Niedermeier. Parameterized computational complexity of Dodgson and Young elections. *Information and Computation*, 208(2):165–177, 2010.

[32] Michael R Fellows, Bart MP Jansen, Daniel Lokshtanov, Frances A Rosamond, and Saket Saurabh. Determining the Winner of a Dodgson Election is Hard. In *FSTTCS*, pages 459–468, 2010.

[33] Nadja Betzler, Rolf Niedermeier, and Gerhard J Woeginger. Unweighted Coalitional Manipulation under the Borda Rule Is NP-Hard. In *IJCAI*, volume 11, pages 55–60. Citeseer, 2011.

[34] Yongjie Yang and Jiong Guo. Exact algorithms for weighted and unweighted Borda manipulation problems. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, pages 1327–1328. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[35] Yongjie Yang. Manipulation with bounded single-peaked width: A parameterized study. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 77–85. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[36] Britta Dorn and Ildikó Schlotter. Multivariate complexity analysis of swap bribery. *Algorithmica*, 64(1):126–151, 2012.

[37] Jianxin Wang, Min Yang, Jiong Guo, Qilong Feng, and Jianer Chen. Parameterized complexity of control and bribery for d-approval elections. In *Combinatorial Optimization and Applications*, pages 260–271. Springer, 2013.

[38] Robert Bredereck, Jiehua Chen, Piotr Faliszewski, André Nichterlein, and Rolf Niedermeier. Prices Matter for the Parameterized Complexity of Shift Bribery. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[39] Nadja Betzler, Susanne Hemmann, and Rolf Niedermeier. A Multivariate Complexity Analysis of Determining Possible Winners Given Incomplete Votes. In *IJCAI*, volume 9, pages 53–58, 2009.

[40] Lirong Xia and Vincent Conitzer. Determining Possible and Necessary Winners under Common Voting Rules Given Partial Orders. *Journal of Artificial Intelligence Research*, pages 25–67, 2011.

[41] Dorothea Baumeister, Piotr Faliszewski, Jérôme Lang, and Jörg Rothe. Campaigns for lazy voters: Truncated ballots. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 577–584. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[42] Claudia Lindner and Jörg Rothe. Fixed-parameter tractability and parameterized complexity, applied to problems from computational social choice. In *Mathematical Programming Glossary*, 2008.

[43] Nadja Betzler, Robert Bredereck, Jiehua Chen, and Rolf Niedermeier. Studies in Computational Aspects of Voting: a Parameterized Complexity Perspective. In *The Multivariate Algorithmic Revolution and Beyond*, pages 318–363. Springer-Verlag, 2012.

[44] Jessica Davies, George Katsirelos, Nina Narodytska, Toby Walsh, and Lirong Xia. Complexity of and algorithms for the manipulation of Borda, Nanson's and Baldwin's voting rules. *Artificial Intelligence*, 217:20–42, 2014.

[45] Sergey Brin and Lawrence Page. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer networks*, 56(18):3825–3833, 2012.

[46] Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming*, 127(1):203–244, March 2011.

[47] David C Parkes and Lirong Xia. A complexity-of-strategic-behavior comparison between Schulze's rule and ranked pairs. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI'12)*. American Association for Artificial Intelligence, 2012.

Misrepresentation in District Voting

Yoram Bachrach
Microsoft Research
Cambridge, United Kingdom
yobach@microsoft.com

Omer Lev
University of Toronto
Toronto, Canada
omerl@cs.toronto.edu

Yoad Lewenberg
The Hebrew University of
Jerusalem, Israel
yoadlew@cs.huji.ac.il

Yair Zick
Carnegie Mellon University,
USA
yairzick@cs.cmu.edu

ABSTRACT

Voting systems in which voters are partitioned to districts encourage accountability by providing voters an easily identifiable district representative, but can result in a selection of representatives not representative of the electorate's preferences. In some cases, a party may have a majority of the popular vote, but lose the elections due to districting effects.

We define the *Misrepresentation Ratio* which quantifies the deviation from proportional representation in a district-based election, and provide bounds for this ratio under various voting rules. We also examine probabilistic models for election outcomes, and provide an algorithm for approximating the expected Misrepresentation Ratio under a given probabilistic election model. Finally, we provide simulation results for several such probabilistic election models, showing the effects of the number of voters and candidates on the misrepresentation ratio.

1. INTRODUCTION

A voting system is a method by which voters choose between several alternatives and their opinions are aggregated, ultimately choosing a winner (or winners). Democratic countries, in principle, aim to have a representative outcome, by having a legislature roughly representative of the public's beliefs, and in some countries, by having the chief executive elected directly by the public.

However, many democracies use a *district based* system for the selection of their legislature (most prominently, the Westminster system and the US system). In district based schemes, voters are divided into geographically based districts[1], and each one selects a representative to the legislature. The selection mechanisms differ: Westminster and US systems use plurality, France uses plurality with runoff, while Australia uses STV. Typically, candidates in each constituency are members of political parties, and some systems have the majority party form the executive (others, such as the US, have a similar process for selecting the chief executive).

Single-member district elections provide voters a single and easily identifiable district representative, encouraging service and accountability. However, the proportion of seats in the legislature belonging to a party may be very different from the proportion of voters supporting that party in the overall population; this is known as the "referendum paradox" [33]. The disparity between the popular vote and the district vote has been a source of contention in US elections; by redistricting constituencies ("gerrymandering"), political parties have manipulated the elections [25, 10]; indeed, the US Voter Rights Act of 1965 includes several provisions that require change in congressional districts in several states to be approved by federal authorities [36].

Moreover, such a discrepancy is caused not only by gerrymandering, but is built into district-based mechanisms. In the US presidential election of 1876, the losing candidate, Samuel Tilden, got 6% more votes than the winner, Rutherford Hayes, an occurrence that happened twice since. The electoral college, through which the president is elected, displays the problem even more acutely; in the US 1992 presidential election a candidate that garnered 18.9% of the vote, Ross Perot, was not represented at all. In the UK system, in 1951 the Conservative party lost the popular vote to the Labour party while still winning a strict parliamentary majority; in general, while no party has received a majority of the popular vote in a British election since 1931, all but two elections resulted in a parliamentary majority for one of the parties. Similar scenarios have unfolded in Canada, Australia and elsewhere.

Such problems may also arise in any multi-level decision making process. So, if an organization or a sensor analysis system (e.g., an automated car), attempts to decide on its next step based on inputs from sub-systems employing their own decision making processes (e.g., each sensor family is a district, and sensors are "voting" between themselves), they may also encounter such a problem, as a small amount of signals may cause the system to reach a wrong outcome.

Consider the following example. Two political parties, A and B compete in seven districts of equal size, $D_1, D_2, \ldots, D_7$. Both parties run a candidate in every district, and the plurality voting rule is used to determine the winner. Now, suppose that in districts $D_1, \ldots, D_4$, 60% of the vote goes to the representative of party A; on the other hand, in districts D_5, D_6 and D_7 the representatives of party B take 100% of the vote. The final vote tally shows party A wins

[1]While we use the term "district", other terms include "electoral district" (US), "riding" (Canada), and "constituency" (UK).

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

the election, even though party B has nearly twice as many voters!

As the example above shows, a minority (those in favor of party A) may rule over the majority (those in favor of party B). We refer to the outcome in this case as a *misrepresentation of the voters' preferences*. District based electoral systems tend to be more "stable": they tend to result in a smaller number of candidates, and thus less fragmentation of the parliamentary body. However, misrepresentation is an inherent byproduct of electoral stability; indeed, stability comes at the heavy price of potentially overriding the preferences of most voters.

One way to quantify the degree to which a system skews the true desires of the voters — as captured by the total support for each party — is to examine the ratio between the number of those who voted for the most popular party in general, and the number of voters who voted for the winner. When more people voted for the losing party than for the winning party this ratio is larger than one. The higher this ratio is, the more pronounced the misrepresentation effect.

Our Contribution.

We examine the issue of misrepresentation due to district-based systems under several prominent voting rules. We first provide a metric for quantifying the misrepresentation effect, both for elections with two parties and for elections with more than two parties, which we call the Misrepresentation Ratio. We provide bounds on the Misrepresentation Ratio for several voting rules, depending on the numbers of voters, candidates and districts. Finally, we provide some simulation results regarding the misrepresentation ratio in certain scenarios, to examine its values under various settings.

2. RELATED WORK

Voting district representability has been the topic of much public debate and research for two centuries, since the advent of redistricting in United States ("gerrymandering") [2]

and in United Kingdom, redistricting due to an attempt to follow population changes in successive reform bills. Most of the research on these issues focused on historical [8, 7], sociological [29] and political science [10] issues. Particularly since the Voting Rights Act of 1965, much legal research has also dealt with district based system, though usually focusing on particular countries (mainly – though not solely – the US) [36, 25].

Political science research on such misrepresentation has mostly focused on trying to assess when it might occur, studying the history of elections or using statistic assumptions [38] and analysis on election data [16, 31, 32, 27, 28, 22]. This analysis, however, does not try to asses how bad the misrepresentation can get, and mostly focuses on plurality and its variants. A somewhat similar line of research tries to assess voter's misrepresentation as distance of parliamentary seat allocation compared to pure representational settings [19, 17, 20, 15], applying concepts such as the Banzhaf index and voting power. However, we deal with the different issue of quantifying the misrepresentation degree by examining the degree of support of the losing candidate.

The computational social choice community focused on manipulation by voters [40, 41, 35], and less on more institutional motivated manipulation. Some work has been done on control problems, in which a central authority may significantly influence results (e.g. [23] and the survey by [12]), including preliminary work on dividing voters into groups [9]. Optimal gerrymandering strategies have been studied in [34, 18]; however, these have mostly focused on 2 party scenarios, as in the US.

In establishing our bounds, we encounter the problem of finding the minimal score a candidate can win with, given a scoring rule; [39] studies a similar problem, but their results do not directly apply to our setting. Our problem is a special case of the bin packing problem with a constant number of types; this problem has been widely believed to be NP-hard, but recent advances [37, 26] have eventually established that it is in P [21].

3. PRELIMINARIES

We have a set of *voters*, V, and each voter $i \in V$ has a preference order (without ties) over the *candidate set* C, denoted $\prec_i$. For every $c, c' \in C$, we say that i *prefers* c *to* c' if $c' \prec_i c$. We denote the set of linear preferences over C as $\mathcal{L}(C)$.

A *voting rule* is a function $f : \mathcal{L}(C)^n \to C$, whose input is a finite list of linear preferences over C (a *profile*) of size n (most voting rules are well defined for any $n > 0$) and whose output is a candidate $c \in C$. Voting rules are often assumed to output a set of candidates, but since in our setting we are only interested in the winners of the election, the output of f is a single candidate (f may incorporate some fixed tie-breaking rule). Since the set of voters and their preferences is constant throughout the paper, we overload notation and define f over subsets of V: given a set $Q \subseteq V$, we let $f(Q)$ be the output of f over the preferences of the voters in Q.

A voting rule is *neutral* if the outcome is invariant under candidate name changes. More formally, a voting rule f is neutral if for any two candidates c and c', if every voter $i \in V$ now ranks c in the position of c' and vice versa, the outcome of f is unchanged if the winner was neither c or c', is c' if the winner was c, and is c if the winner was c'.

We say a voting rule f is *score-monotone* if it induces a score for every candidate $c \in C$ (i.e., each candidate ends up with some real number, and the one with the maximal one is chosen), and the following holds: given two preference profiles $R_1 = (\prec_i)_{i \in V}$ and $R_2 = (\prec'_i)_{i \in V}$, if for all $a, b \in C \setminus \{c\}$ we have $a \prec_i b \iff a \prec'_i b$, and under R_1, c is in a position that is no lower than its position in R_2, then c's score under R_1 is at least as high as its score in R_2. These properties hold for the common voting rules which will be used in this paper.

Scoring Rules: A scoring rule is defined by a vector of scores, $(\alpha_1, \ldots, \alpha_m)$, where $\alpha_1 \geq \cdots \geq \alpha_m = 0$, $\alpha_1 > 0$[3] and m is the number of candidates. Given a voter $i \in V$, let $rank_i(c)$ be the rank of candidate c in the preference order of i. Given a set of voters $Q \subseteq V$,

[2]Our analysis focused on how bad misrepresentation might be, rather that the issue of how restructuring districts may be used as a purposeful manipulation, designed to give an unfair advantage to a specific candidate, more in line with other forms of manipulation and control in elections [11, 13, 1, 3].

[3]Assuming $\alpha_m = 0$ is no loss of generality: if $\alpha_m > 0$ score vector can be normalized by setting $\alpha_j = \alpha_j - \alpha_m$.

let $score_c^{\alpha}(Q) = \sum_{i \in Q} \alpha_{rank_i(c)}$. Given a set of voters $Q \subseteq V$, the output of $Scoring_{\alpha}(Q)$ is a member of $\arg\max_{c \in C} score_c^{\alpha}(Q)$ (if there are ties, we break them according to some tie-breaking rule). Many scoring rules are widely used. For example, the *plurality* rule, where $\alpha_1 = 1$ and $\alpha_j = 0$ for all $j \geq 2$, the *veto* rule, where $\alpha_j = 1$ for all $j < m$, and $\alpha_m = 0$, both of which are specific examples of the family of k-approval scoring rules, in which $\alpha_1 = \ldots = \alpha_k = 1$ and $\alpha_{k+1} = \ldots = \alpha_m = 0$. Another common scoring rule is the *Borda* rule, where $\alpha_j = m - j$ for all $j \in \{1, \ldots, m\}$.

Copeland Rule: Given a set of voters $Q \subseteq V$, we say that a candidate c beats c' in a pairwise election under Q if a majority of the members of Q prefer c to c'. For each candidate $c \in C$, we let $score_c^{Cp}(Q)$ to be the number of candidates that c beats in a pairwise election, minus the number of candidates that beat c. The Copeland voting rule outputs $Copeland(Q) = \arg\max_{c \in C} score_c^{Cp}(Q)$.

We note that both scoring rules and Copeland have a natural notion of a candidate score, and were conceptualized as such in [6]. This will be instrumental in defining voting misrepresentation, as we do in the following section.

Definition 1. Given a voting rule f which induces a candidate score, we let $score_f(c, Q)$ to be the score of $c \in C$ under f, when the voter set is $Q \subseteq V$.

4. THE MISREPRESENTATION RATIO

We are interested in settings where the set of voters is partitioned into districts: these are z disjoint sets $D_1, \ldots, D_z$ whose union is V. The election winner under this model is determined by applying the voting rule f to $D_1, \ldots, D_z$; the candidate who wins the greatest number of districts is the winner (ties are broken using some tie-breaking rule).

Definition 2. Given a partition of voters into z districts $\mathcal{D} = \{D_1, \ldots, D_z\}$, let w be the winner of the election when voters are partitioned into districts as per $\mathcal{D}$. The *misrepresentation ratio* is the ratio between the maximum score of any candidate under f, and the score of w. Formally:

$$MR(V, \mathcal{D}, f) = \frac{\max_{c \in C} score_f(c, V)}{score_f(w, V)}.$$

Note that $MR(V, \mathcal{D}, f) \geq 1$; if $MR(V, \mathcal{D}, f) = 1$ then the winner of the district elections completely captures the popular vote (as measured by f). The higher $MR(V, \mathcal{D}, f)$, the less popular the winning candidate is in the eyes of the people; thus, district elections with a high misrepresentation ratio are ones where voters' preferences are not appropriately aggregated, due to the effects of district elections.

Remark 1. In this work we assume all districts have the same number of voters. Without any assumptions on the number of voters in each district, the worst case misrepresentation ratio (Definition 2) can be arbitrarily high: consider $2\ell + 1$ districts. Suppose $\ell + 1$ of the districts have only three voters, and ℓ of them have M voters, where M is a very large number. There are two candidates, A and B; candidate A wins all votes in the ℓ districts where there are M voters, and candidate B wins 2 of the 3 votes in the $\ell + 1$ districts holding three voters. Thus, the total number of votes for B is $\mathcal{O}(\ell)$, and the total number of votes for A is $\mathcal{O}(M\ell)$, resulting in an arbitrarily high misrepresentation. Our results can be extended by incorporating an additional parameter: $\max_{D, D' \in \mathcal{D}} \frac{|D|}{|D'|}$; however, in the interest of space and clarity, we assume that districts are of equal size.

5. BOUNDS ON THE MISREPRESENTATION RATIO

In what follows, we establish upper bounds on the *worst-case misrepresentation ratio*; we show that our bounds are tight, i.e. that there exist district elections where the bound holds.

Furthermore, we always refer to the district voting instance $\langle V, C, (\prec_i)_{i \in V}, \mathcal{D}, f \rangle$ as one that maximizes MR, where $|C| = m$, $|\mathcal{D}| = z$, for all $D \in \mathcal{D}$: $|D| = n$, and f is some neutral score-monotone rule. We assume that $w \in C$ is the winner of the district elections; that is, w won a plurality of the districts. We mark $d(c)$ for $c \in C$ as the number of districts won by candidate c. We assume that in the case of a tie for first place in a district, ties are broken against the district election winner; moreover, the district election winner must win a strict majority of the districts.

Intuitively, in order to establish our bound, we want to create the worst possible election. Such an election would have the candidate w win by as small a margin as possible, with some other candidate $p \neq w$ being as popular as possible while losing a majority of the districts.

Given a score inducing voting rule f, we let $L_n(f)$ be the maximal score that a candidate can get while still losing an n voter election, and $M_n(f)$ be the maximal score that a candidate can get and win an n voter election. For example, $L_n(Plurality) = \lceil \frac{n}{2} \rceil - 1$, and $M_n(Plurality) = n$. The following lemma offers some insights regarding the number of districts won by each candidate in an instance maximizing MR.

LEMMA 1. *If f is a score-monotone, neutral rule then there is a district voting instance maximizing MR such that:*

1. *if w does not win the election in a district D, then w is ranked last by all voters in D.*
2. *If the candidate p wins in D, then it is ranked first by all voters in D; if not and the winner is not w, then the score of p from D is $L_n(f)$.*
3. *$d(p) = d(w) - 1$ if $m > 2$ or z is odd, otherwise $d(p) = d(w) - 2$.*

PROOF. If w loses the election in a district D, let $i \in D$ be a voter who did not rank w last; we can simply swap the candidate that was ranked last by i with w in i's ranking. This will result in a voter profile where the score of w is weakly lower, and the score of every other candidate is weakly higher (by the score monotonicity and neutrality of f). Furthermore, if p wins the election in D, but is not ranked first by some voter $i \in D$, swapping p with the candidate ranked first by i (this is not w by the previous observation) will result in p having a weakly higher score. Whenever p loses the election in a district, its score from that district is lower than the score that it could have obtained by winning a district, but as we wish to maximize it

(and we do not care for the score of candidates other than w and p), p should score $L_n(f)$.

To show that $d(c) \leq d(p)$ for all $c \neq w$, suppose there is a candidate $c \neq w, p$ for which $d(c) > d(p)$. Changing districts who voted for c to vote for p will only improve p's score while not changing w as winner; thus, $d(c) \leq d(p)$ for all $c \neq p, w$.

For 2 candidates and an even number of districts, the difference between the number of districts must be at most 2; if it is more, we can take districts voting for w and change them to p, improving p's score and hurting w (while having w remain the election winner). For 2 candidates, if the number of districts is odd we do a similar process, reaching a difference of 1.

Now, suppose w has won q more districts than p, and $q > 2$. In this case, we can take some district where w won and make p the winner in that district. That would strictly improve p's score, reduce w's score and maintain w as the overall election winner. This establishes that $d(w) \leq d(p) + 2$. Now, we wish to show that if $m \geq 3$ then $d(w) = d(p)+1$. If $d(c) = 0$ for all $c \neq w, p$ then we are back in the two candidate case which we have previously covered, and if z is even, we can give one of w's district to c. If there is some district D won by some $c \neq w, p$ and $d(w) = d(p) + 2$ then we can make p win c's district, weakly improving its score while maintaining w's score at the same level, and keeping w as the district election winner. □

5.1 Scoring Rules

We begin our investigation by bounding *MR* when f is a scoring rule. We begin with a simple technical lemma.

LEMMA 2. *For every scoring rule $f = Scoring_\alpha$, where $\alpha = (\alpha_1, \ldots, \alpha_m)$, if $\alpha_1 > \alpha_2$, $L_n(f) = \alpha_1(\lceil \frac{n}{2} \rceil - 1) + \alpha_2(\lfloor \frac{n}{2} \rfloor + 1)$. If $\alpha_1 = \alpha_2$, denote α' as the maximal α_i such that $\alpha_i < \alpha_1$, then $L_n(f) = (n-1)\alpha_1 + \alpha'$.*

PROOF. We shall consider some candidate $c \in C$ which we make the winner, while giving candidate $p \in C$ the maximal score for a runner up. First, showing that there indeed can be a winner with a higher score, if $\alpha_1 > \alpha_2$, if $\lceil \frac{n}{2} \rceil - 1$ voters give α_1 to p and α_2 to c, while the rest of the voters give α_1 to c and α_2 to p, we reach the desired score for p while c is the winner. If $\alpha_1 = \alpha_2$, for $n-1$ voters giving α_1 or α_2 to both p and c, and the final voter giving α_1 points to c and α' points to p, we again reach the desired score while c is the winner.

Now, we shall show no higher score is possible. For the case $\alpha_1 = \alpha_2$ it is obvious: the only higher score possible is $n\alpha_1$, which is the maximal one, hence the winner's. For $\alpha_1 > \alpha_2$, suppose there is a better $L_n(f)$, which is described as $z_1\alpha_1 + z_2\alpha_2 + \ldots + z_m\alpha_m$. If $z_1 \leq \lceil \frac{n}{2} \rceil - 1$, then this sum is smaller than that suggested in the lemma. If $z_1 > \lceil \frac{n}{2} \rceil - 1$, this means the winner c has a score of at most $\alpha_1(\lceil \frac{n}{2} \rceil - 2) + \alpha_2(\lfloor \frac{n}{2} \rfloor + 1)$. This score needs to be higher than $z_1\alpha_1 + z_2\alpha_2 + \ldots + z_m\alpha_m$, but it is lower than the lemma's suggested $L_n(f)$, reaching a contradiction. □

COROLLARY 1. *If $f = Scoring_\alpha$ then $(z+1)L_n(f) \geq M_n(f)$ for any $z > 1$.*

PROOF. Given a vector $\alpha = (\alpha_1, \ldots, \alpha_m)$ we have that $M_n(f) = n\alpha_1$. Now, by Lemma 2, we know that

$$L_n(Scoring_\alpha) \geq \left(\left\lceil \frac{n}{2} \right\rceil - 1 \right) \alpha_1,$$

from which the claim trivially follows. □

The following lemma tells us the number of districts that must be won by the winning candidate in a district election that maximizes *MR*.

LEMMA 3. *Let $\langle V, C, (\prec_i)_{i \in V}, \mathcal{D}, f \rangle$ be a district voting instance which maximizes MR, where f is a scoring rule. z (number of districts) can be written as $\ell m + r$ ($\ell, r \in \mathbb{N} \cup \{0\}$ and $r < m$). There is an MR maximizing instance where the number of districts that w wins is at most $\ell + 2$, and every other candidate wins at least $\ell - 1$ districts.*

PROOF. Recall that $d(c)$ be the number of districts that each candidate $c \in C$ wins. In order to win the district election, w needs to win a plurality of the districts; that is, $d(w) \geq d(c) + 1$ for all $c \in C \setminus \{w\}$. Now, let us denote the score of w from winning a district as x_w; by Lemma 1 w gets no votes from all districts where it loses (the case where there are two candidates and an even number of districts can be handled similarly, and is omitted due to space constraints). Now, p gets a score of x_p from a district that w wins, a score of $M_n(f)$ from a district that it wins, and a score of $L_n(f)$ from a district that some other candidate wins. Putting it all together we have:

$$\frac{x_p d(w) + M_n(f) d(p) + L_n(f)\left(\sum_{c \in C \setminus \{w,p\}} d(c)\right)}{x_w d(w)} = \frac{x_p}{x_w} + \frac{M_n(f) d(p) + L_n(f)\left(\sum_{c \in C \setminus \{w,p\}} d(c)\right)}{x_w d(w)}.$$

Since $d(w) = d(p) + 1$, $\sum_{c \in C \setminus \{w,p\}} d(c) = z - 2d(w) + 1$, hence our equation is

$$\frac{x_p + M_n(f) - 2L_n(f)}{x_w} + \frac{(z+1)L_n(f) - M_n(f)}{x_w d(w)}. \quad (1)$$

We are interested in maximizing (1) as a function of $d(w)$; the derivative of (1) with respect to $d(w)$ is

$$\frac{(z+1)L_n(f) - M_n(f)}{x_w} \cdot \frac{1}{-d(w)^2}$$

which is positive if and only if $(z+1)L_n(f) < M_n(f)$. According to Corollary 1, $(z+1)L_n(f) > M_n(f)$ then (1) is maximized when $d(w)$ is as small as possible. If $r = 0$ then $d(w) = \ell + 1$ while every other candidate gets ℓ (if $d(w) \leq \ell$ more than one candidate will have such a score). If $r \geq 1$, then $d(w)$ must be more than $\ell + 1$, as more than one candidate must get that score, so $d(w) = \ell + 2$ and other candidates can get $\ell + 1$ districts or less. □

The following lemma discusses the score that w receives in districts that it wins.

LEMMA 4. *For every scoring rule f, if $n \gg z$, $z > 3$ (i.e., number of voters in each district is far larger than the number of districts), the score given to w in districts in which it wins must be the minimal possible score needed to win a district in an instance maximizing MR.*

PROOF. Note that this is not trivial – it is not obvious that we do not wish to increase w's score to allow p to receive a higher score as a 2nd place candidate.

For simplicity, we shall assume n is an odd number and $\alpha_1 > \alpha_2$. The proof is equivalent in the other cases. We can also assume $m > 2$, as otherwise this is already set by Lemma 1.

From Equation 1, we know that we seek to maximize

$$\frac{x_p + M_n(f) - 2L_n(f)}{x_w} + \frac{(z+1)L_n(f) - M_n(f)}{x_w d(w)}.$$

We know $M_n(f) = n\alpha_1$ and $L_n(f) = \alpha_1(\lceil \frac{n}{2} \rceil - 1) + \alpha_2(\lfloor \frac{n}{2} \rfloor + 1)$ from Lemma 2, hence we wish to maximize:

$$\begin{aligned} & \frac{x_p + n\alpha_1 - (n-1)\alpha_1 - (n+1)\alpha_2}{x_w} + \\ & \frac{(z+1)\big(\lceil \frac{n}{2} \rceil \alpha_1 + \lfloor \frac{n}{2} \rfloor \alpha_2\big) - n\alpha_1}{x_w d(w)} \\ = \quad & \frac{x_p + \alpha_1 - (n+1)\alpha_2}{x_w} + \frac{(z+1)\big(\lceil \frac{n}{2} \rceil \alpha_1 + \lfloor \frac{n}{2} \rfloor \alpha_2\big) - n\alpha_1}{x_w d(w)}. \end{aligned}$$

Now, whatever x_w is, we wish to maximize x_p as long as $x_w > x_p$. For a particular scoring rule f, $x_p = x_w - s_n(f)$ for some function s_n[4]. Therefore, looking at the derivative of our equation according to x_w, we get

$$\begin{aligned} & \frac{1}{x_w^2}\Bigg(s_n(f) - \alpha_1 + (n+1)\alpha_2 - \\ & \frac{(z+1)\big(\lceil \frac{n}{2} \rceil \alpha_1 + \lfloor \frac{n}{2} \rfloor \alpha_2\big) + n\alpha_1}{d(w)}\Bigg) \\ = \quad & \frac{1}{x_w^2}\Bigg(s_n(f) + \alpha_1 \frac{n - d(w) - (z+1)\lceil \frac{n}{2} \rceil}{d(w)} + \\ & \alpha_2 \frac{d(w)(n+1) - (z+1)\lfloor \frac{n}{2} \rfloor}{d(w)}\Bigg). \end{aligned}$$

As $n \gg z$ and $z > 3$, this is always negative as the α_1 and α_2 components overwhelms all others (since $s_n(f) < n\alpha_1$), hence x_w needs to be minimal in size. □

LEMMA 5. *For k-approval voting rules, the minimal score a winner can get in an election is $\lfloor \frac{nk}{m} \rfloor + 1$ if $m \lfloor \frac{nk}{m} \rfloor + 1 = nk$. Otherwise, it is $\lceil \frac{nk}{m} \rceil + 1$.*

PROOF. Since nk points are allocated by voters, a winner must have at least a score of $\lfloor \frac{nk}{m} \rfloor + 1$. If $m \lfloor \frac{nk}{m} \rfloor + 1 = nk$, then it is possible that w gets $\lfloor \frac{nk}{m} \rfloor + 1$ votes, while all other candidates get $\lfloor \frac{nk}{m} \rfloor$. If $m \lfloor \frac{nk}{m} \rfloor + 1 < nk$, there is no allocation of nk points such that only a single candidate gets $\lfloor \frac{nk}{m} \rfloor + 1$, hence the winner will need to get one more point, with the rest of the points to be allocated among the other candidates, with none of them getting more than $\lceil \frac{nk}{m} \rceil$ votes. □

Armed with Lemmata 1 – 5, we now proceed to analyze specific voting rules.

THEOREM 1. *Suppose that $n = qm + s$ and $z = \ell m + r$, where $q, s, \ell, r \in \mathbb{N} \cup \{0\}$ and $s, r < m$; then $MR(V, \mathcal{D}, Plurality)$ is at least*

$$1 + \frac{n - 2\lceil \frac{n}{2} \rceil + 1}{q+2} + \frac{(z+1)(\lceil \frac{n}{2} \rceil - 1) - n}{(\ell+2)(q+2)}$$

and at most

$$1 + \frac{n - 2\lceil \frac{n}{2} \rceil + 1}{q+1} + \frac{(z+1)(\lceil \frac{n}{2} \rceil - 1) - n}{(\ell+1)(q+1)}.$$

In particular, $MR(V, \mathcal{D}, Plurality)$ is in $\Theta(m^2)$.

[4]Obviously, $x_w - n\max_i(\alpha_i - \alpha_{i+1}) \le x_p$, as x_p can always be ranked below x_w. On the other hand, $x_p \le x_w - \min_i(\alpha_i - \alpha_{i-1})$.

PROOF. By Lemma 3 and Corollary 1, we know that w wins either $\ell+1$ or $\ell+2$ districts. Plugging in the values $L_n(Plurality) = \lceil \frac{n}{2} \rceil - 1$ and $M_n(Plurality) = n$ into (1), we have that $MR(V, \mathcal{D}, Plurality)$ equals

$$\frac{x_p + n - 2\lceil \frac{n}{2} \rceil + 2}{x_w} + \frac{(z+1)(\lceil \frac{n}{2} \rceil - 1) - n}{x_w d(w)}$$

(x_w and x_p denote the score of w and p (respectively) when w wins a district). We are left just with determining the value of x_p and x_w. For reasons similar to the ones detailed in Lemma 1, it holds that $x_p = x_w - 1$.

Due to Lemma 4, $MR(V, \mathcal{D}, Plurality)$ is maximized when x_w is minimal; We have that w receives $q+2$ votes if $s \ge 2$, and $q+1$ otherwise. Plugging this into (1) we obtain the desired result.

The second expression in the upper bound of Theorem 1, $\frac{n - 2\lceil \frac{n}{2} \rceil + 1}{q+1}$ can be upper bounded by $\frac{1}{q+1}$, which is at most $\frac{m}{n}$. Thus, if the number of voters dominates the number of candidates, this expression has little effect on MR. The second expression can be upper bounded as follows

$$\frac{(z+1)(\lceil \frac{n}{2} \rceil - 1) - n}{(\ell+1)(q+1)} \le \frac{zn}{(\ell+1)(q+1)} \le m^2$$

A similar lower bound of $\Omega(m^2)$ can be shown as well, which concludes the proof. □

Note that tightness is achieved as our constructed expressions were dependent on particular voting profiles (as described in Lemma 1), and hence carry on to these expressions.

As some parliamentary systems require not a plurality of districts to become a winner, but a majority, we also note the MR in these cases.

COROLLARY 2. *If the number of districts needed for a victory is above 50%, MR for plurality is $\Theta(m)$.*

PROOF. Suppose that $n = qm + s$ for $q, m \in \mathbb{N}$, $s < m$. As $d(w)$ in our expression is now $\lfloor \frac{n}{2} \rfloor + 1$, and thanks to Lemma 4 we know we should minimize x_w. Hence, w receives $q+2$ votes if $s \ge 2$, and $q+1$ otherwise. Thanks to Lemma 1 $d(p) = \lceil \frac{n}{2} \rceil - 1$, and its score in the other districts is $x_w - 1$. The formula turns out to be, for $s \ge 2$ (very similar figure for $s \le 1$):

$$\frac{q+1}{q+2} + \frac{n(\lceil \frac{n}{2} \rceil - 1)}{(q+2)(\lfloor \frac{n}{2} \rfloor + 1)} \overset{n \to \infty}{\approx} m.$$

□

THEOREM 2. *Suppose that the number of districts is expressed as $z = \ell m + r$, where $\ell, r \in \mathbb{N} \cup \{0\}$ and $r < m$; then $MR(V, \mathcal{D}, k\text{-approval})$ for $k > 1$ is at least*

$$1 + \frac{n(z + \ell + 2) - (z - \ell + 1)}{(\ell+2)(\lceil \frac{nk}{m} \rceil + 1)}$$

and at most

$$1 + \frac{n(z + \ell + 1) - (z - \ell)}{(\ell+1)(\lfloor \frac{nk}{m} \rfloor + 1)}.$$

PROOF. By Lemma 3 and Corollary 1, we know that w wins either $\ell+1$ or $\ell+2$ districts. Plugging in the values $L_n(k\text{-approval}) = n-1$ and $M_n(k\text{-approval}) = n$ into (1), we have that $MR(V, \mathcal{D}, k\text{-approval})$ equals $\frac{x_p + 2 - n}{x_w} + \frac{kn - (z+1)}{x_w d(w)}$

(x_w and x_p denote the score of w and p (respectively) when w wins a district).As before, for reasons similar to the ones detailed in Lemma 1, $x_p = x_w - 1$.

Due to Lemma 4, $MR(V, \mathcal{D}, k\text{-approval})$ is maximized when x_w is minimal; According to Lemma 5, $x_w = \left\lfloor \frac{nk}{m} \right\rfloor + 1$ if $m \left\lfloor \frac{nk}{m} \right\rfloor + 1 = nk$. Otherwise, it is $x_w = \left\lceil \frac{nk}{m} \right\rceil + 1$. Plugging these values into the equation, we receive the desired result. □

One of the main challenges in computing a closed form formula for *MR* for general scoring rules is that one must first decide what is the minimal score that w can obtain while winning a district for a given score vector α. This problem can be thought of as a *bin packing problem*: candidates can be thought of as bins, and the scores must be packed into them. It is only recently that a polynomial time algorithm has been proposed for bin packing problems with a constant number of types (also commonly referred to as the *one-dimensional cutting stock problem*) [21]. Thus, for general scoring rule we offer looser bounds on the number of votes needed to win:

Let $S_\alpha = \sum_{j=1}^{m} \alpha_j$; then the minimal number of votes needed to win a district is at most $\left\lceil \frac{nS_\alpha}{m} \right\rceil + \alpha_1$, and at least $\left\lceil \frac{nS_\alpha}{m} \right\rceil$: we allocate the scores as evenly as possible among the candidates, and break the tie in favor of the winner using at most α_1 points. Of course, in some cases this can be improved, but it depends on α, and on the divisibility of nS_α and m. The following theorem uses these loose bounds to bound *MR* for the Borda scoring rule.

THEOREM 3. *$MR(V, \mathcal{D}, Borda)$ is in $\Theta(m^2)$.*

PROOF. According to Lemma 2, $L_n(Borda) = (m-1)(\left\lfloor \frac{n}{2} \right\rfloor - 1) + (m-2)(\left\lfloor \frac{n}{2} \right\rfloor + 1)$, and $M_n(Borda) = n(m-1)$. We again write x_w and x_p to be the number of votes that are won by w and p (respectively) in a district where w wins. Thus, *MR* is

$$\frac{x_p + n(m-1) - 2(m-2)(n-1)}{x_w} + \frac{(z+1)(m-1)(m-2)n}{x_w d(w)}.$$

We can also show the score p receives in a district won by w is at least $x_w - (m-1)$, and at most $x_w - 1$. Furthermore, we observe that x_w is at least $\left\lceil \frac{\frac{(m-1)(m-2)}{2} n}{m} \right\rceil$, which is at least $\frac{(m-2)^2 n}{2m}$. Furthermore, x_w is upper bounded by $\left\lceil \frac{\frac{(m-1)(m-2)}{2} n}{m} \right\rceil + (m-1)$, which is at most $\frac{m(n+2)}{2}$. Thus, *MR* is upper bounded by

$$\frac{2m((m-2)n(m+1)(m-1) + (m-1))}{(m-2)^2 n}.$$

We note that the final expression has n in both the numerator and denominator and is thus $\mathcal{O}(m^2)$. We may similarly lower bound $MR(V, \mathcal{D}, Borda)$ by a similar value. □

5.2 Copeland

When using the Copeland voting rule, one can get an undefined value for *MR*, as a score of 0 is possible for the winner.

Example 1. Let us have two identical districts, each containing 21 voters with the preference $w \succ p \succ a$ and 20 voters who have $p \succ w \succ a$. A third district contains 41 voters with the preference $p \succ w \succ a$. w wins the 2 first districts, becoming the ultimate winner. But, looking globally, p's Copeland score is 2, while w's Copeland score is 0, making $MR(V, \mathcal{D}, Copeland)$ undefined.

The Copeland score can be additively adjusted by adding to each candidate's score a fixed amount that is larger than m. However, Copeland's performance remains bad, as is captured by Theorem 4.

THEOREM 4. *Under Copeland, the winner w may have the worst possible Copeland score, while another candidate has the best possible Copeland score.*

PROOF. The worst possible Copeland score is $-(m-1)$, while the best is $m-1$. Taking $z = \ell m + r$, $z > 4$ and $m > 2$, we take $\ell + 2$ districts, in each of them all voters rank $w \succ p \succ \ldots$. We now take $\ell + 1$ districts, in each of them all voters rank $p \succ \ldots \succ w$. Finally, the rest of the districts are divided between the $m-2$ other candidates, each candidate $c \in C \setminus \{p, w\}$ getting at most $\ell + 1$ districts, in which every voter ranks $c \succ p \succ \ldots \succ w$. Since $\ell + 2 < \frac{n}{2}$, w has lost to all candidates for most voters, and therefore w's Copeland score is $-(m-1)$. p, on the other hand, is preferred by most voters over any other candidate, leading its score to be $m-1$. □

Algorithm 1 Monte-Carlo *MR* Approximation

```
procedure EXPECTED-MR ( M , B, ε, δ)
    s = ⌈ B^2 · ln(2/δ) / (2 ε^2) ⌉
    T = 0
    for i = 1 to s do
        Sample an election outcome E from M
        w = arg max_{c∈C} score(c, V) // The winner
        s_m = max_{c∈C} score(c, V) // Maximal score
        R_i^samp = s_m / s_w // sampled MR
        T = T + R_i^samp
    return r̂ = T / s // average of sampled MRs
```

6. THE MISREPRESENTATION RATIO UNDER UNCERTAIN VOTES

In Section 5 we established bounds on the misrepresentation ratio by constructing pathological examples: settings where districting effects were so pronounced as to cause an extremely unpopular candidate to win the elections, despite the existence of a clearly better alternative. We now take the average-case, rather than the worst-case, approach, and ask how common are instances where misrepresentation is high.

We do this in the form of a probabilistic generative model, utilizing partial information to inform our assumptions on the general population. Any instantiation of the model is a voting domain for which we can compute the misrepresentation ratio. Thus, *MR* is a random variable and we evaluate the *expected MR*. A naive solution is to exhaustively search over the space of possible election outcomes; for each such outcome we can compute its probability of occurring under the generative model, and the *MR* value for that outcome; we can then sum the product of the two across all outcomes to get the expected *MR*. However, such an exhaustive search

is intractable, as the space of outcomes can be prohibitively large, especially when there are many candidates, voters and districts. We propose an alternative approximate solution, based on a Monte-Carlo algorithm. Our algorithm requires a bound on the *MR* for the examined domain, and such can be found for many cases (see previous section).

We assume the generative model is given in the form of a black-box, which outputs a sampled election outcome, consisting of the votes of every voter in every district. We further assume that the winner of the election can be computed in polynomial time[5]. Denote the generative model as M, and by $r = E_M(MR)$ the expectation of *MR* under the model M. Our proposed algorithm is "probably approximately correct": given two parameters, ε and δ, the algorithm returns an approximation $\hat{r}$ to r, such that with high probability $1-\delta$ the returned value $\hat{r}$ is very close to r, so that $|\hat{r} - r| \leq \varepsilon$. The running time of the algorithm depends on ε and δ; it is quadratic in $\frac{1}{\varepsilon}$ and logarithmic in $\frac{1}{\delta}$.

Our proposed algorithm is a Monte-Carlo algorithm, but it is only appropriate to voting rules where there is a known bound on the possible *MR* values. [6] The minimal *MR* value is 1 (as this is the ratio between the maximal score of any candidate and the score of a specific candidate, namely the winner). Given a bound H on the maximal *MR* in a domain, we refer to the *MR* value range as $B = H - 1$. The runtime of our algorithm is quadratic in B. The method is given in Algorithm 1, and we provide a proof for its correctness.

THEOREM 5. *The value returned by Expected-MR is an ε, δ approximation for the expected MR under M: with probability at least $1-\delta$ the returned value $\hat{r}$ is within a distance ε of $r = E_M(MR)$, i.e: $|\hat{r} - r| \leq \varepsilon$.*

PROOF. We note that the R_i^{samp} computed inside the loop is the *MR* in a specific instantiation of an election outcome E sampled from the generative model M (see Definition 2), so each R_i^{samp} is a random variable, whose expectation is $r = E_M(MR)$ (i.e. $E[R_i^{samp}] = r$). Our algorithm computes $T = \sum_{i=1}^{s} R_i^{samp}$, the sum of s i.i.d draws, each of which has a value of r in expectation, so $E[\hat{r}] = E[\frac{T}{s}] = r$. We use Hoeffding's inequality [24] to show that the number of samples s that we use achieves the desired accuracy ε and confidence δ. Hoeffding's inequality states that if $R_1, \ldots, R_n$ are independent random variables, where each R_i is bounded so that $R_i \in [a_i, b_i]$, and if $T = \sum_{i=1}^{s} R_i$, then $\Pr(|T - \mathrm{E}[T]| \geq s\varepsilon) \leq 2\exp\left(-\frac{2\,s^2\,\varepsilon^2}{\sum_{i=1}^{n}(b_i-a_i)^2}\right)$.

R_i^{samp} is the *MR* in an election outcome obtained under the generative model M, so R_i^{samp} is bounded in the range $[1, H]$ (i.e., by our assumption, the *MR* value range is $B = H-1$). Applying Hoeffding's bound and substituting $\frac{B^2 \cdot \ln\frac{2}{\delta}}{2\,\varepsilon^2}$ for s, we get $\Pr\left(\left|\frac{T}{s} - \frac{\mathrm{E}[T]}{r}\right| \geq \varepsilon\right) \leq \delta$ as required. □

7. SIMULATIONS

We now use our Algorithm Expected-*MR* to analyze the *MR* in several voting domains. We begin with a noisy version of the example domain described in the introduction.

[5]Not all voting rules admit a polynomial winner determination algorithm. As our algorithm samples election outcomes, its runtime in this case would not be polynomial.

[6]Our method is akin to Monte-Carlo methods used for analyzing voting under various forms of uncertainty [14, 2, 4, 5].

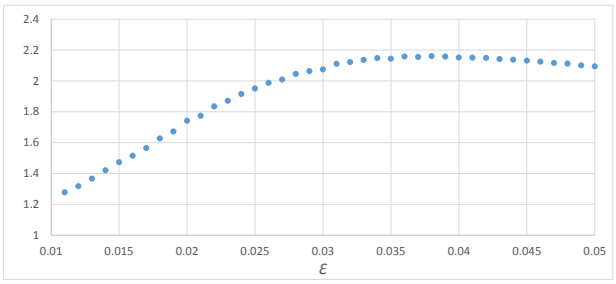


Figure 1: The effects of the noise on the *MR*

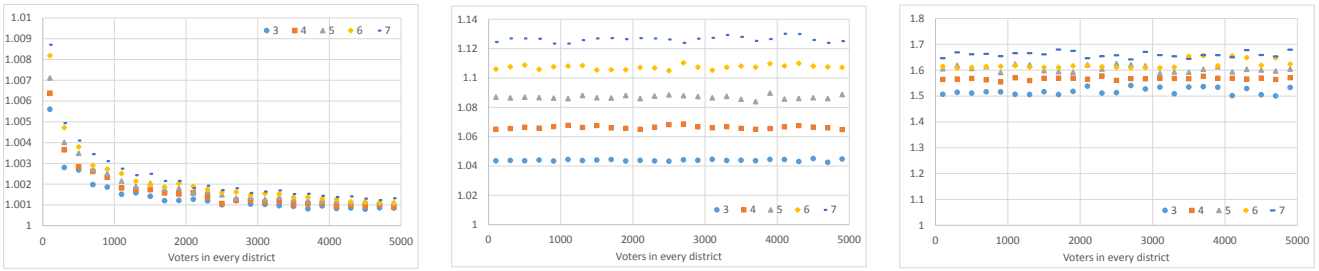
We fix $C = \{w, p\}$, and the number of districts to 11: 6 districts of type "A" and 5 of type "B", modeling heterogeneous and homogenous districts respectively. In type A districts, every voter v votes randomly with $\Pr[v \text{ votes for } w] = \frac{1}{2} + \varepsilon$, and $\Pr[v \text{ votes for } p] = \frac{1}{2} - \varepsilon$; in type B districts, $\Pr[v \text{ votes for } w] = \varepsilon$, $\Pr[v \text{ votes for } p] = 1 - \varepsilon$, for $\varepsilon \in \left(0, \frac{1}{2}\right)$. Figure 1 shows the averaged *MR* as a function of the noise ε (x-axis). Each point is average *MR* obtained for many elections sampled using this probabilistic model; *MR* first increases as the noise ε grows, until it reaches a sweet-spot from which it drops. This indicates that for some models the model noise may not have a monotone effect on the *MR*.

In our second experiment, we fix the number of districts to 15, and range the number of voters in every district from 100 to 5000. We examined *MR* of elections with $m \in \{3, 4, \ldots, 7\}$ candidates. Figure 2a shows the averaged *MR* where the preference of every voter in every district is uniformly drawn from the set of all $m!$ possible orders of candidates, under Borda scoring. The figure shows that increasing the number of voters tends to lower *MR*. This is not surprising, as all candidates will likely have nearly the same score. Next, we consider the Mallows model [30] for generating voter preferences, where we assume that there is a "ground truth" for every district, σ^* (representing the common ranking of candidates in that district) and dispersion parameter $\phi \in \left(\frac{1}{2}, 1\right]$. Under the Mallows model every voter compares every pair of candidates independently and ranks them correctly (according to σ^*) with probability ϕ. For every district, σ^* was drawn uniformly at random and $\phi \sim \mathcal{U}(\frac{1}{2}, 1)$. We used the plurality scoring rule (Figure 2b) and Copeland voting rule (Figure 2c). As predicted by our theoretical results, *MR* grows when there are more candidates. Under Copeland, a fixed amount of m was added each candidate's score so that *MR* would be positive.

Our simulation results indicate that voting misrepresentation may occur in several natural domains. Our second experiment is "fair" in the sense that there is no preferred candidate, and yet the *MR* values are quite high. Also, our theoretical results agree with experiments in some natural domains.

8. CONCLUSIONS

This work analyzes district-based elections. We demonstrate the representability issues that arise in such elections, and show tight bounds on misrepresentation. We further show that misrepresentation is a common occurrence under various natural voter distributions, and that its effect may



(a) *MR* where the preferences are uniformly drawn, under Borda scoring rule

(b) *MR* where the preferences are drawn by a Mallows model, under plurality

(c) *MR* where the preferences are drawn by a Mallows model, under Copeland voting rule

Figure 2: Simulations results

not diminish even when the number of voters is large.

District based elections tend to under-represent smaller parties; this is a long observed phenomenon (and, in some countries, a welcome stabilizing feature). However, we do not focus on smaller parties, but rather show that the preferences of large majorities may be completely unrepresented (the UK elections of 1951, where the Labour party with 48.8% support lost the elections, and the Conservative party with 44.3% not only beat it, but had a strong parliamentary majority, is just one example of these occurrences).

Research into the institutional bias in voting procedures, beyond control issues, is one which we think deserves more attention by the computational social choice community. Very few election systems in the world are proportional, and the effect this has on the expression of voters' views has mostly focused (in political science research) on how small minorities are hurt. As our analysis shows, large majorities may also be affected. Further research is needed with regards to other voting methods. Moreover, further and complementary concepts may be developed, indicating unfairness, lack of representation and other problems with various voting procedures (parliamentary entrance bounds, common in some countries, are an obvious candidate for such directions). In addition, while we have focused on an outcome of a single winner, a coalitional analysis of district settings may also be of interest.

Acknowledgements

This work was supported in part by NSERC grant 482671.

REFERENCES

[1] H. Aziz, Y. Bachrach, E. Elkind, and M. Paterson. False-name manipulations in weighted voting games. *Journal of Artificial Intelligence Research*, pages 57–93, 2011.

[2] Y. Bachrach, N. Betzler, and P. Faliszewski. Probabilistic possible winner determination. In *AAAI*, volume 10, pages 697–702, 2010.

[3] Y. Bachrach, E. Elkind, and P. Faliszewski. Coalitional voting manipulation: A game-theoretic perspective. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 49, 2011.

[4] Y. Bachrach, R. Meir, M. Feldman, and M. Tennenholtz. Solving cooperative reliability games. *arXiv preprint arXiv:1202.3700*, 2012.

[5] Y. Bachrach and N. Shah. Reliability weighted voting games. In *Algorithmic Game Theory*, pages 38–49. Springer, 2013.

[6] J. J. Bartholdi, C. A. Tovey, and M. A. Trick. The computational difficulty of manipulating an election. *SCW*, 6(3):227–241, 1989.

[7] D. Butler. The electoral process the redrawing of parliamentary boundaries in britain. *British Elections and Parties Yearbook*, 2(1), 1992.

[8] E. J. Engstrom. Stacking the states, stacking the house: The partisan consequences of congressional redistricting in 19th century. *APSR*, 100:419–427, 2006.

[9] G. Erdélyi, E. Hemaspaandra, and L. A. Hemaspaandra. More natural models of electoral control by partition. In *ADT*, pages 396–413, 2015.

[10] R. S. Erikson. Malapportionment, gerrymandering, & party fortunes in congressional elections. *APSR*, 66(4):1234–1245, 1972.

[11] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. The complexity of bribery in elections. In *AAAI*, volume 6, pages 641–646, 2006.

[12] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. Using complexity to protect elections. *CACM*, 53(11):74–82, 2010.

[13] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Llull and copeland voting computationally resist bribery and constructive control. *Journal of Artificial Intelligence Research*, pages 275–341, 2009.

[14] S. S. Fatima, M. Wooldridge, and N. R. Jennings. A linear approximation method for the shapley value. *Artificial Intelligence*, 172(14):1673–1699, 2008.

[15] M. Feix, D. Lepelley, V. Merlin, J. Rouet, and L. Vidu. Majority efficient representation of the citizens in a federal union. Technical report, Université de la Réunion, de Caen, d'Orleans, 2008.

[16] M. R. Feix, D. Lepelley, V. R. Merlin, and J. Rouet. The probability of conflicts in a U.S. presidential type election. *Economic Theory*, 23(2):227–257, 2004.

[17] D. S. Felsenthal and M. Machover. Minimizing the mean majority deficit: The 2nd square-root rule. *Math. Soc. Sci.*, 37(1):25–37, 1999.

[18] J. N. Friedman and R. T. Holden. Towards a theory of optimal partisan gerrymandering. Technical report,

Harvard University, 2005.

[19] M. Gallagher. Proportionality, disproportionality and electoral systems. *Electoral Studies*, 10(1):33–51, 1991.

[20] A. Gelman, J. N. Katz, and F. Tuerlinckx. The mathematics and statistics of voting power. *Statistical Science*, 17(4):420–435, 2002.

[21] M. Goemans and T. Rothvoß. Polynomiality for bin packing with a constant number of item types. In *SODA*, pages 830–839, 2014.

[22] B. Grofman, W. Koetzle, and T. Brunell. An integrated perspective on the three potential sources of partisan bias: Malapportionment, turnout differences, & the geographic distribution of party vote shares. *Electoral Studies*, 16(4):457–470, 1997.

[23] E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Anyone but him: The complexity of precluding an alternative. *AIJ*, 171(5–6):255–285, 2007.

[24] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[25] S. Issacharoff. Gerrymandering & political cartels. *Harvard Law Review*, 116(2):593–648, 2002.

[26] K. Jansen, S. Kratsch, D. Marx, and I. Schlotter. Bin packing with fixed number of bins revisited. *JCSC*, 79(1):39–49, 2013.

[27] R. Lahrach and V. Merlin. *Electoral Systems: Paradoxes, Assumptions, and Procedures*, chapter Which Voting Rule Minimizes the Probability of the Referendum Paradox? Lessons from French Data, pages 129–150. Springer, 2011.

[28] R. Lahrach and V. R. Merlin. Assessing the probability of the referendum paradox: The french local election case. Voting Power in Practice workshop, 2010.

[29] D. Lublin. *The Paradox of Representation: Racial Gerrymandering and Minority Interests in Congress*. Princeton University Press, 1997.

[30] C. Mallows. Non-null ranking models. I. *Biometrika*, pages 114–130, 1957.

[31] N. Miller. *Electoral Systems: Paradoxes, Assumptions & Procedures*, chapter Election Inversions by the U.S. Electoral College, pages 93–127. Springer, 2011.

[32] N. Miller. The house size effect and the referendum paradox in u.s. presidential elections. *Electoral Studies*, 35:265–271, September 2014.

[33] H. Nurmi. *Voting Paradoxes and How to Deal with Them*. Springer-Verlag, 1999.

[34] G. Owen and B. Grofman. Optimal partisan gerrymandering. *Political Geography Quarterly*, 7(1):5–22, January 1988.

[35] A. D. Procaccia, J. S. Rosenschein, and A. Zohar. Multi-winner elections: Complexity of manipulation, control & winner-determination. In *IJCAI*, pages 1476–1481, 2007.

[36] P. Schuck. The thickest thicket: Partisan gerrymandering and judicial regulation of politics. *Columbia Law Review*, 87(7):1325–1384, 1987.

[37] W. Stille. *Solution Techniques for specific Bin Packing Problems with Applications to Assembly Line Optimization*. PhD thesis, TU Darmstadt, June 2008.

[38] A. Tangian. Computational application of the mathematical theory of democracy to Arrow's impossibility theorem (how dictatorial are Arrow's dictators?). *SCW*, 35(1):129–161, 2010.

[39] L. Xia. Computing the margin of victory for various voting rules. In *EC*, pages 982–999, 2012.

[40] L. Xia, M. Zuckerman, A. D. Procaccia, V. Conitzer, and J. S. Rosenschein. Complexity of unweighted coalitional manipulation under some common voting rules. In *IJCAI*, pages 348–353, 2009.

[41] M. Zuckerman, A. D. Procaccia, and J. S. Rosenschein. Algorithms for the coalitional manipulation problem. In *SODA*, pages 277–286, 2008.

Small Representations of Big Kidney Exchange Graphs

John P. Dickerson, Aleksandr M. Kazachkov, Ariel Procaccia, Tuomas Sandholm
Carnegie Mellon University
{dickerson,arielpro,sandholm}@cs.cmu.edu, akazachk@cmu.edu

ABSTRACT

Kidney exchanges are organized markets where patients swap willing but incompatible donors. In the last decade, kidney exchanges grew from small and regional to large and national—and soon, international. This growth results in more lives saved, but exacerbates the empirical hardness of the $\mathcal{NP}$-complete problem of optimally matching patients to donors. State-of-the-art matching engines use integer programming techniques to clear fielded kidney exchanges, but these methods must be tailored to specific models and objective functions, and may fail to scale to larger exchanges. In this paper, we observe that if the kidney exchange compatibility graph can be encoded by a constant number of patient and donor attributes, the clearing problem is solvable in polynomial time. We give necessary and sufficient conditions for losslessly shrinking the representation of an arbitrary compatibility graph. Then, using real compatibility graphs from the UNOS nationwide kidney exchange, we show how many attributes are needed to encode real compatibility graphs. The experiments show that, indeed, small numbers of attributes suffice.

1. INTRODUCTION

There are over $100{,}000$ needy patients waiting for a kidney transplant in the United States, with similar, and increasing, demand worldwide.[1] Complementing potential cadaveric transplantation via the deceased donor waiting list, a recent innovation—kidney exchange [30, 32]—allows patients with willing *living* donors to participate in cyclic donor swaps or altruist-initiated donation chains to receive a life-saving organ. Kidney exchange now accounts for roughly 10% of living donation in the US, with that percentage increasing every year.

In reality, participating patients and donors are endowed with a set of attributes: blood type, tissue type, age, insurance, home transplant center, willingness to travel, and myriad other measurements of health, personal preference, and logistical constraint. While some of these features can, at a cost, be temporarily or permanently changed, the attributes determine the feasibility of a potential donation from each donor to each patient. As a concrete example, a donor with blood type AB can only give to a patient with that blood type.

A central aspect of kidney exchange is the *clearing problem*, that is, determining the "best" set of cyclic and chain-based swaps to perform in a given *compatibility graph*, which consists of all participating patients, donors, and their potential feasible transactions. For even simple (but realistic) models of kidney exchange, the clearing problem is $\mathcal{NP}$-hard [1, 6] and also extremely difficult to solve in practice [18, 3, 29].

In this paper, we tackle the complexity of the clearing problem via the introduction of a novel model for kidney exchange that explicitly takes into account all attributes of the participating patients and donors. Under the assumption that real kidney exchange graphs can be represented using just a constant number of attributes, we show that our model permits polynomial-time solutions to central $\mathcal{NP}$-hard problems in general kidney exchange. Inspired by classical results from intersection graph theory, we give conditions on the representation of arbitrary graphs in our model, and generalize to the case where participants are allowed to have a thresholded number of negative interactions between attributes. Noting that real-life kidney exchange graphs are *not* arbitrary, we show on actual data from the United Network for Organ Sharing (UNOS) US-wide kidney exchange that our model permits lossless representation of true graphs with far fewer attributes than the worst-case theoretical results require.

2. A NEW MODEL FOR KIDNEY EXCHANGE

In this section, we formalize our model of kidney exchange. We prove that under this model certain well-known $\mathcal{NP}$-hard problems in general kidney exchange are solvable in polynomial time. We also show that, given a compatibility graph, determining the best set of attributes to change (at some cost) is solvable in polynomial time.

2.1 Notation & Preliminaries

A kidney exchange can be represented by a directed *compatibility graph* $G = (V, E)$. Each patient-donor pair, or unpaired altruistic donor, forms a vertex $v \in V$, and a directed edge exists from one vertex to another if the donor at the former can give to the patient at the latter, i.e., are compatible [32, 33, 34].

In kidney exchange, patients and donors participate in *cycles* or *chains*. In a cycle, each participating vertex receives the kidney of the previous vertex. All transplants in a cycle must be performed simultaneously to ensure participation, and thus are limited to some small length in practice. This ensures that no donor backs out after her patient has received a kidney but before she has donated her kidney. Most fielded kidney exchanges—including UNOS—allow only 2- and 3-cycles. In a chain, a donor *without a paired patient* enters the pool, donating his kidney to a patient, whose paired donor donates his kidney to another patient, and so on [27, 35, 31].

[1] https://optn.transplant.hrsa.gov/converge/data/

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

Chains can be executed non-simultaneously[2] and thus chains can be longer (but typically not infinite) in length. Most exchanges—including UNOS—see great gains through the use of such "altruist-initiated" chains.

We consider a model that imposes additional structure on an arbitrary compatibility graph. For each vertex $v_i \in V$, associate with its constituent donor and patient attribute vectors $\mathbf{d}_i$ and $\mathbf{p}_i$, respectively. Here, the qth element d_i^q of $\mathbf{d}_i$ takes on one of a fixed number of types—for example, one of four blood types (O, A, B, AB), or one of a few hundred standard insurance plans. Then, for $v_i \neq v_j \in V$, we define a *compatibility function* $f(\mathbf{d}_i, \mathbf{p}_j)$, a boolean function that returns the compatibility of the donor of v_i with the patient of v_j.

Given V and associated attribute vectors, we can uniquely determine a compatibility graph $G = (V, E)$ such that $E = \{(v_i, v_j) : f(\mathbf{d}_i, \mathbf{p}_j) = 1 \ \forall v_i \neq v_j \in V\}$. We claim that this model accurately mimics reality, and we later support that claim with strong experimental results on real-world data. Furthermore, under this new model, certain complexity results central to kidney exchange change (for the better), as we discuss next.

2.2 The Clearing Problem is Easy (in Theory)

We now tackle the central computational challenge of kidney exchange: the clearing problem. Well-known to be $\mathcal{NP}$-hard [1, 6], a variety of custom clearing algorithms address adaptations of the clearing problem in practice.[3] We show that, in our model, the clearing problem itself is solvable in polynomial time.

Formally, we are interested in a polynomial-time algorithm that solves the L-CYCLE-COVER problem—that is, finding the largest disjoint packing of cycles of length at most L. For ease of exposition, in this section we use "cycles" to refer to both cycles and chains; indeed, it is easy to see that altruist donors are equivalent to standard patient-donor pairs with a patient who is compatible with all non-altruist vertices in the pool. Then, a chain is equivalent to a cycle with a "dummy" edge returning to the altruist. Also, again for ease of exposition, we assume the value of a chain of length L is equal to a cycle of length L, due to the donor at the end of the chain giving to a patient on the deceased donor waiting list.

Recall that we are working in a model where each vertex v_i belongs to one of a *fixed* number of types determined solely by its attribute vectors $\mathbf{d}_i$ and $\mathbf{p}_i$. Let Θ be the set of all possible types, and $\theta \in \Theta$ represent one such individual type. Then, with a slight abuse of notation, we can define a type compatibility function $f(\theta, \theta') = 1$ if and only if there is a directed edge between vertices of type θ and θ'. (Note that this captures chains and altruist donors as described above.)

A key observation of this section is that any additional edge structure that is imposed on the graph—such as a cycle cover—would be independent of the identity of *specific* vertices, rather, it would only depend on their types, as vertices of the same type have the exact same incoming and outgoing neighborhoods. For example, in any cycle cover, if v_i and v_j are two vertices of the same type, we can insert v_j in place of v_i, and v_i in place of v_j, and obtain a feasible cycle cover of the same size. This observation drives our theoretical algorithmic results.

In more detail, for a vector of types $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k) \in \Theta^k$, let us denote $f_C(\boldsymbol{\theta}) = 1$ if and only if $f(\theta_t, \theta_{t+1}) = 1$ for all $t < k$, and $f(\theta_k, \theta_1) = 1$. In other words, $f_C(\boldsymbol{\theta}) = 1$ if every k vertices $i_1, \ldots, i_k$ of types $\theta_1, \ldots, \theta_k$, respectively, are involved in a cycle in the graph. Furthermore, for $L \leq n = |V|$, denote

$$\mathcal{T}(L) = \{\boldsymbol{\theta} \in \Theta^k : k \leq L \text{ and } f_C(\boldsymbol{\theta}) = 1\}.$$

That is, $\mathcal{T}(L)$ contains all vectors of types that induce feasible cycles of length at most L.

Now consider the following algorithm for L-CYCLE-COVER in our model:

Algorithm 1 L-CYCLE-COVER

1. $\mathcal{C}^* \leftarrow \emptyset$
2. **for** every collection of numbers $\{m_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \mathcal{T}(L)}$ such that $\sum_{\boldsymbol{\theta} \in \mathcal{T}(L)} m_{\boldsymbol{\theta}} \leq n$
   - **if** there exists cycle cover $\mathcal{C}$ such that $\|\mathcal{C}\|_V > \|\mathcal{C}^*\|_V$ and for all $\boldsymbol{\theta} \in \mathcal{T}(L)$, $\mathcal{C}$ contains $m_{\boldsymbol{\theta}}$ cycles consisting of vertices of the types in $\boldsymbol{\theta}$ **then** $\mathcal{C}^* \leftarrow \mathcal{C}$
3. **return** $\mathcal{C}^*$

Here, $\|\mathcal{C}\|_V$ denotes the number of unique *vertices* matched in a cycle cover $\mathcal{C}$. We claim that, in our setting, Algorithm 1 is optimal and computationally efficient.

THEOREM 1. *Suppose that L and $|\Theta|$ are constants. Then Algorithm 1 is a polynomial-time algorithm for L-CYCLE-COVER.*

PROOF. We start by verifying that Algorithm 1 is indeed optimal. Consider the optimal cycle cover $\mathcal{C}^*$. For each $\boldsymbol{\theta} \in \mathcal{T}(L)$, let $m^*_{\boldsymbol{\theta}}$ be the number of cycles in $\mathcal{C}^*$ that are consistent with the types in $\boldsymbol{\theta}$. Clearly $\sum_{\boldsymbol{\theta} \in \mathcal{T}(L)} m^*_{\boldsymbol{\theta}} \leq n$, as there are only n vertices so there cannot be more than n cycles (in fact, $n/2$ is also a valid upper bound). Therefore, Algorithm 1 considers the collection of numbers $m^*_{\boldsymbol{\theta}}$ in Step 2. Because this collection of numbers does induce a valid cycle cover that is of the same size as $\mathcal{C}^*$, the algorithm would update its incumbent cycle cover if it was not already optimal.

We next analyze the running time of the algorithm. First, note that it is straightforward to check whether the numbers $\{m_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \mathcal{T}(L)}$ induce a valid cycle cover. Since $\mathcal{T}(L)$ consists only of valid cycles according to the compatibility function f_C, we just need to check that there are enough vertices of type θ to construct all the cycles that require them. This simply amounts to multiplying each $m_{\boldsymbol{\theta}}$ by the number of times type θ appears in $\boldsymbol{\theta}$, and verifying that the sum of these products over all $\boldsymbol{\theta}$ in $\mathcal{T}(L)$ is at most the number of vertices of type θ.

Second, we argue that there is only a polynomial number of possibilities to construct a collection of numbers $\{m_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \mathcal{T}(L)}$ such that $\sum_{\boldsymbol{\theta} \in \mathcal{T}(L)} m_{\boldsymbol{\theta}} \leq n$. Indeed, this number is at most $(n+1)^{|\mathcal{T}(L)|}$. Moreover, $|\mathcal{T}(L)| \leq L \cdot |\Theta|^L$. Because $|\Theta|$ and L are constants, $|\mathcal{T}(L)|$ is also a constant. The expression $(n+1)^{|\mathcal{T}(L)|}$ is therefore a polynomial in n. □

Even for constant L, the running time of Algorithm 1 is exponential in k. But this is to be expected. Indeed, any graph can trivially be represented using a set Θ of types of size n, where each vertex has a unique type, and a compatibility function f_C that assigns 1 to an ordered pair of types if the corresponding edge exists in G. Therefore, if the running time of Algorithm 1 were polynomial in n *and* k, we could solve the general L-CYCLE-COVER problem in polynomial time—and that problem is $\mathcal{NP}$-hard [1].

[2] To see why this is, take the case where a donor backs out of a chain after his paired patient received a kidney, but before his own donation. Unlike in the case of a broken cycle, no pair in the remaining tail of the planned chain is strictly worse off; that is, no donor was "used up" before her paired patient received a kidney.

[3] For an overview of practical approaches to solving the clearing problem, see a recent survey due to Mak-Hau [25].

2.3 Flipping Attributes is Also Easy (in Theory)

While patients and donors in a kidney exchange are endowed with an initial set of attributes, it may be possible in practice to—at a cost—change some number of those attributes to effect change in the final matching. For example, the human body naturally tries to reject, to varying degrees, a transplanted organ. Due to this, nearly all recipients of kidneys are placed on immunosuppressant drugs after transplantation occurs.[4] However, *preoperative* immunosuppression can also be performed to increase transplant opportunity—but at some cost to the patient's overall health.

With this in mind, we extend the model of Section 2.2 as follows. Associate with each pair of types $\theta, \theta' \in \Theta$ a cost function $c : \Theta \times \Theta \to \mathbb{R}$ representing the cost of changing a vertex of type θ to type θ'. Then, the L-FLIP-AND-CYCLE-COVER problem is to find a disjoint packing of cycles of length at most L that maximizes the size of the packing minus the sum of costs spent changing types. Building on Theorem 1, this problem is also solvable in polynomial time.

THEOREM 2. *Suppose that L and $|\Theta|$ are constants. Then L-FLIP-AND-CYCLE-COVER is solvable in polynomial-time.*

PROOF PROOF SKETCH. For any type $\theta_i \in \Theta$, there are n_i vertices. Then, for each of the $(|\Theta| - 1)$ choices of which type $\theta \neq \theta_i$ to switch to, choose how many vertices from θ_i will switch to a different type; there are $(n_i + 1)$ choices. Do this for all $|\Theta|$ types, resulting in $\prod_{\theta_i \in \Theta}[(n_i + 1)(|\Theta| - 1)]$ choices. Note that $\sum_{\theta_i \in \Theta}(n_i + 1) = n + |\Theta|$, meaning $\prod_{\theta_i \in \Theta}[(n_i + 1)(|\Theta| - 1)] \leq ((n + |\Theta|) \cdot (|\Theta| - 1)/|\Theta|)^{|\Theta|} \leq (n + |\Theta|)^{|\Theta|}$. Since $|\Theta|$ is a constant, this is polynomial in n; invoking an adaptation of the polynomial time Algorithm 1 that subtracts out $c(\theta, \theta')$ for every vertex that switches from θ to θ', for each of the polynomially-many choices, concludes the proof. □

3. A CONCRETE INSTANTIATION: THRESHOLDING

As motivated in Sections 1 and 2, compatibility in real kidney exchange graphs is determined by patient and donor attributes, such as blood or tissue type. In particular, if an attribute for a donor and patient is in conflict, they are deemed incompatible. Motivated by that reality, in this section, we associate with each patient and donor a bit vector of length k, and count incompatibilities based on any shared activated bits between a patient and potential donor.

As a concrete example, consider human blood types. At a high level, human blood contains A antigens, B antigens, both (type AB), or neither (type O). AB-type patients can receive from any donor, A-type (B-type) can receive from O-type and A-type (B-type) donors, and O-type patients can only receive from O-type donors. In our bit model, this is represented with $k = 2$, such that a donor's first (resp. second) bit is set if his blood holds A (resp. B) antigens. and a patient's first (resp. second) bit is set if she cannot receive from blood containing A (resp. B) antigens. Thus, the type space $\Theta = 2^{\{\texttt{has-A,has-B}\}} \times 2^{\{\texttt{no-A,no-B}\}}$; in general, $|\Theta| = 2^{2k}$.

Formally, unless otherwise stated, throughout this section G will refer to a directed graph with vertex set $V = [n] := \{1, \ldots, n\}$ and edge set E, and with each $i \in V$ associated with two k-bit vectors $\mathbf{d}_i, \mathbf{p}_i \in \{0, 1\}^k$. Let $Q_d(i) = \{q \in [k] : \mathbf{d}_{iq} = 1\}$ be the set of conflict bits for the donor associated with vertex $i \in V$, and similarly let $Q_p(i) = \{q \in [k] : \mathbf{p}_{iq} = 1\}$. For $i, j \in V$ such that $i \neq j$, the threshold feasibility function f^t_{thresh} is defined as

$$f^t_{\mathsf{thresh}}(\mathbf{d}_i, \mathbf{p}_j) = \begin{cases} 1 & \text{if } |Q_d(i) \cap Q_p(j)| \leq t, \\ 0 & \text{otherwise.} \end{cases}.$$

Note that $|Q_d(i) \cap Q_p(j)| \leq t$ if and only if $\langle \mathbf{d}_i, \mathbf{p}_j \rangle \leq t$.

Kidney exchange graphs constructed using threshold compatibility functions are closely related to complements of *intersection graphs* [26], which are graphs that have a set associated with each vertex and an edge between two vertices if and only if the sets intersect. Given a nonnegative integer t, the function f^t_{thresh} is related to *p-intersection graphs* [9, 13], in which an edge exists between two vertices if their corresponding sets intersect in at least $p \geq 1$ elements.

Our model is similar to that of *intersection digraphs* [36], or equivalently *bipartite intersection graphs* [19], both also considered in [28]. Both of these have mainly been studied under the assumption that the sets used to represent the graph have the "consecutive ones" property, i.e., each set is an interval from the set of integers. Our model is more general: we do not place such an assumption on the set of conflict bits. Moreover, most treatments of intersection digraphs consider loops on the vertices, whereas in the thresholding model we have defined, whether or not donor i and patient i are compatible is not considered. In addition, the directed and bipartite intersection graph literature has focused on the case that $t = 0$ (in our terminology). To the best of our knowledge, this paper is the first treatment *p-intersection digraphs*, and certainly their first real-world application.

3.1 Existence of Small Representations

It is natural to ask for what values of t and k can we select vertices with bit vectors $\mathbf{d}_i$ and $\mathbf{p}_i$ of length k such that f^t_{thresh} can create *any* graph of a specific size?

Formally, we say that G is *(k, t)-representable* (by feasibility function f^t_{thresh}) if, for all $i \in V$ there exist $\mathbf{d}_i, \mathbf{p}_i \in \{0, 1\}^k$ such that for all $j_1 \in V$, $j_2 \in V \setminus \{j_1\}$, $(j_1, j_2) \in E$ if and only if $f^t_{\mathsf{thresh}}(\mathbf{d}_{j_1}, \mathbf{p}_{j_2}) = 1$.

It is known [14] that any graph can be represented as an intersection graph with $k \leq n^2/4$. In contrast, the next theorem shows that, in our model, $k \leq n$ suffices to represent any graph. It is akin to a result on the *term rank* of the adjacency matrix of $\overline{G}$ [28, Theorem 6.6].

THEOREM 3. *Let $G = (V, E)$ be a digraph on n vertices. Let n_1 be the number of vertices with outgoing edges, Let n_2 be the number of vertices with incoming edges, and $n' = \min\{n_1 + 1, n_2 + 1, n\}$. Then G can be $(n', 0)$-represented.*

PROOF. We first show that the graph can be $(n_1+1, 0)$-represented. Assume without loss of generality that vertices $1, \ldots, n_1$ have outgoing edges. We show how to set $\mathbf{d}_i, \mathbf{p}_i \in \{0, 1\}^{n_1+1}$ for each vertex i in V. To set the donor attributes, for each $i \in [n_1]$, let $\mathbf{d}_i$ be e_i, the ith standard basis vector, i.e., the vector of length $n_1 + 1$ with a 1 in the ith coordinate and 0 everywhere else. For $i > n_1$, set $\mathbf{d}_i$ to be e_{n_1+1}. For the patient attributes of vertex $j \in [n]$, for each $i \in [n]$ such that $(i, j) \in E$, set $\mathbf{p}_{ji} = 0$, and set $\mathbf{p}_{ji} = 1$ otherwise. Note that if all the vertices have outgoing edges, then $n_1 = n$ unit vectors suffice. A similar approach works to $(\min\{n, n_2 + 1\}, 0)$-represent G, by using the n_2 unit vectors as the patient vectors of those vertices with incoming edges, and (if needed) one additional unit vector for any remaining vertices. In both of these cases, $\langle \mathbf{d}_i, \mathbf{p}_j \rangle = 0$ if and only if $(i, j) \in E$, which represents G by f^0_{thresh}. □

[4] https://www.kidney.org/atoz/content/immuno

Theorem 3 implies that any graph is $(n, 0)$-representable. The next theorem shows a matching lower bound. The same construction and bound also hold if loops are considered [36].

THEOREM 4. *For any $n \geq 3$, there exists a graph on n vertices that is not $(k, 0)$-representable for all $k < n$.*

PROOF. Define G to be the digraph on n vertices, $V = [n]$, with an edge from vertex i, for each $i \in V$, to every vertex except $i - 1$ (and itself), where vertex n is also referred to as vertex 0.

Assume that G is $(k, 0)$-representable, and consider vertex 1. Since $(1, n) \notin E$, and $(i, n) \in E$ for all $i \notin \{1, n\}$, there exists a conflict bit $q_1 \in Q_d(1) \cap Q_p(n)$ such that $q_1 \notin Q_p(V \setminus \{1, n\})$. More generally, there exists such a conflict bit q_i for all $i \in V$.

We claim that these conflict bits are all unique, which directly implies that $k \geq n$. Indeed, otherwise we can assume that $q_1 = q_i$ for some $i \neq 1$ (without loss of generality, as the graph is symmetric subject to cyclic permutations). But then $(1, i-1)$ and (i, n) do not appear as edges in G, which is not true for any $i \in V \setminus \{1\}$. □

More generally, it is easy to see that any graph that is $(k, 0)$-representable is also $(k+t, t)$-representable for any $t \geq 0$. Indeed, simply take the $(k, 0)$-representation of the graph, and append t ones to every vector. Together with Theorem 3, this shows that any graph is $(n + t, t)$-representable. However, the lower bound given by Theorem 4 does not extend to $t > 0$. We conjecture that for any n and t there exists a graph that can only be (k, t)-represented with $k = \Omega(n)$—this remains an open question.

3.2 Computational Issues

Given any real compatibility graph with n vertices, we know by Theorem 3 that we can $(k, 0)$-represent that graph for $k = n$. But, in practice, how large of a k do we actually need?

Various problems related to intersection graphs are $\mathcal{NP}$-complete for general graphs [23, 28], but we work in a setting with additional structure. And while we do not show that finding a (k, t)-representation is $\mathcal{NP}$-hard, we do show that a slightly harder problem, which we refer to as (k, t)-REPRESENTATION WITH IGNORED EDGES, is $\mathcal{NP}$-hard. Given an input of a directed graph $G = (V, E)$, a subset F of $\binom{V}{2}$, and integers $k \geq 1$ and $t \geq 0$, this problem asks whether there exist bit vectors $\mathbf{d}_i$ and $\mathbf{p}_i$ of length k for each $i \in V$ such that for any $(i, j) \in F$, we have $(i, j) \in E$ if and only if $\langle \mathbf{d}_i, \mathbf{p}_j \rangle \leq t$.

THEOREM 5. *The (k, t)-REPRESENTATION WITH IGNORED EDGES problem is $\mathcal{NP}$-complete.*

The theorem's nontrivial proof is omitted due to lack of space.[5] Here we give a proof sketch. One major idea is the construction of a *bit-grounding gadget* G_k—a subgraph where the bits are set uniquely (up to permutations) in any valid representation, and can be used to set the bits in other vertices. The gadget has $\binom{k}{2}$ vertices; we prove that there is a unique (up to permutations) $(k, 1)$-representation of G_k, where each donor vector has a unique pair of ones, and similarly for each patient vector. Figure 5 shows G_4.

Then, we prove $\mathcal{NP}$-hardness by reduction from 3SAT. In the constructed instance of our problem, we set the threshold to 1. The crux of the reduction is to add a vertex for each clause in the given 3SAT formula, where in the patient vector, the bit corresponding to each literal in the clause is set to 1. This can be done by connecting the vertex to the bit-grounding gadget. Moreover, there is a special vertex that does not have outgoing edges to any of the clause vertices. This means that it must have a 1 in a position that corresponds to one of the literals in each clause. A different part of the construction ensures that there is at most a single 1 in the two positions corresponding to a variable and its negation. Therefore, a valid assignment of the donor bits corresponds to a satisfying assignment for the 3SAT formula.

[5]The complete proof is given in Appendix A.

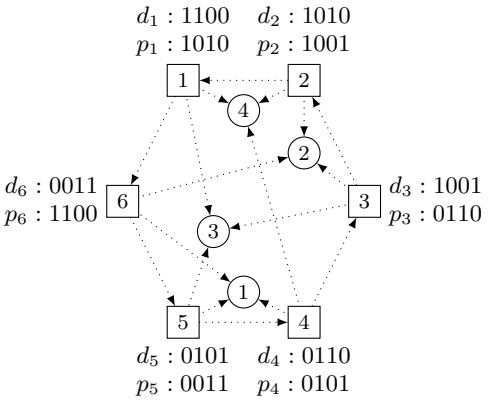


Figure 1: Gadget G_4 with a subset of *non*-edges shown; all edges between circle vertices (those in G_4^2) are also not in E.

4. COMPUTING SMALL REPRESENTATIONS OF REAL KIDNEY EXCHANGE COMPATIBILITY GRAPHS

In this section, we test our hypothesis that real compatibility graphs can be represented by a substantially smaller number of attributes than required by the worst-case theoretical results of Section 3. We begin by presenting general mathematical programming techniques to determine, given $k, t \in \mathbb{Z}$, whether a specific graph $G = (V, E)$ is (k, t)-representable. We then show on real and generated compatibility graphs from the UNOS US-wide kidney exchange that small k suffices for $(k, 0)$-representation, and conclude by exploring the allowance of greater thresholds t on match size. We find even small thresholds $t > 0$ result in substantial societal gain.[6]

4.1 Mathematical Programming Formulations

Implementation of f^t_{thresh} can be written succinctly as a quadratically-constrained discrete feasibility program (QCP) with $2k|V|$ binary decision variables, given as M1 below.

$$\begin{aligned} \langle \mathbf{d}_i, \mathbf{p}_j \rangle &\leq t \quad \forall (v_i, v_j) \in E \\ \langle \mathbf{d}_i, \mathbf{p}_j \rangle &\geq (t+1) \quad \forall (v_i, v_j) \notin E \\ \mathbf{d}_i, \mathbf{p}_i &\in \{0, 1\}^k \quad \forall v_i \in V \end{aligned} \tag{M1}$$

The constraint matrix for this program is not positive semi-definite, and thus the problem is not convex. Exploratory use of heuristic search via state-of-the-art integer nonlinear solvers [7] resulted in poor performance (in terms of runtime and solution quality) on even small graphs. With that in mind, and motivated by the presence of substantially more mature integer *linear* program (ILP) solvers, we linearize M1, presented as M2 below.

[6]All code for this section can be found at `https://github.com/JohnDickerson/KidneyExchange`.

$$
\begin{array}{lrl}
\min & \sum_{v_i \in V} \sum_{v_j \neq v_i \in V} \xi_{ij} & \\
\text{s.t.} & d_i^q \geq c_{ij}^q \wedge p_j^q \geq c_{ij}^q & \forall v_i \neq v_j \in V, q \in [k] \\
& d_i^q + p_j^q \leq 1 + c_{ij}^q & \forall v_i \neq v_j \in V, q \in [k] \\
& \sum_q c_{ij}^q \leq t + (k-t)\xi_{ij} & \forall (v_i, v_j) \in E \\
& \sum_q c_{ij}^q \geq (t+1)\xi_{ij} & \forall (v_i, v_j) \in E \\
& \sum_q c_{ij}^q \geq t + 1 - k\xi_{ij} & \forall (v_i, v_j) \notin E \\
& \sum_q c_{ij}^q \leq k - (k-t)\xi_{ij} & \forall (v_i, v_j) \notin E \\
& d_i^q, p_i^q \in \{0,1\} & \forall v_i \in V, q \in [k] \\
& c_{ij}^q, \xi_{ij} \in \{0,1\} & \forall v_i \neq v_j \in V, q \in [k]
\end{array}
\tag{M2}
$$

M2 generalizes M1; while M1 searches for a feasible solution to the (k,t)-representation problem, M2 searches for the "best" (possibly partially-incorrect) solution by minimizing the total number of edges that exist in the solution but not in the base graph G, or do not exist in the solution but do in G. This flexibility may be desirable in practice to strike a tradeoff between small k and accuracy of representation.

Interestingly, neither the fully general ILP nor its (smaller) instantiations for the special cases of feasibility and/or threshold $t = 0$ were solvable by a leading commercial ILP solver [22] within 12 hours for even small graphs, primarily due to the model's loose LP relaxation. Indeed, the model we are solving is inherently logical, which is known to cause such problems in traditional mathematical programming [21]. With that in mind, we note that the special case of $t = 0$ can be represented compactly as a satisfiability (SAT) problem in conjunctive normal form, given below as M3.

$$
\begin{array}{rl}
\bigwedge_{q \in [k]} (\neg d_i^q \vee \neg p_j^q) & \forall (v_i, v_j) \in E \\
(z_{ij}^1 \vee z_{ij}^2 \vee \ldots \vee z_{ij}^k) \wedge & \\
\bigwedge_{q \in [k]} \left[(\neg z_{ij}^q \vee d_i^q) \wedge (\neg z_{ij}^q \vee p_j^q) \right] & \forall (v_i, v_j) \notin E
\end{array}
\tag{M3}
$$

This formulation maintains two sets of clauses: the first set enforces no bit-wise conflicts for edges in the underlying graph, while the second set enforces at least one conflict via k auxiliary variables $z_{ij}^{\cdot}$ for each possible edge $(v_i, v_j) \notin E$. M3 was amenable to parallel SAT solving [5]. Next, we present results on real graphs using this formulation.

4.2 $(k,0)$-representations of Real Kidney Exchange Graphs

Can real kidney exchange graphs be represented by a small number of attributes? To answer that question, we begin by testing on real match run data from the first two years of the United Network for Organ Sharing (UNOS) kidney exchange, which now contains 143 transplant centers, that is, 60% of all transplant centers in the US. We translate each compatibility graph into a CNF-SAT formulation according to M3, and feed that into a SAT solver [5] with access to 16GB of RAM, 4 cores, and 60 minutes of wall time. (Timeouts are counted—conservatively against our paper's qualitative message—as negative answers.)

Figure 2 shows a classical phase transition from unsatisfiability to satisfiability as k increases as a fraction of graph size, as well as an associated substantial increase in computational intractability centered around that phase transition. This phenomenon is common to many central problems in artificial intelligence [8, 20, 38]. Indeed, we see that substantially fewer than $|V|$ attributes are required to represent real graphs; compare with the lower bound of Theorem 4.

Figure 3 explores the minimum k required to represent each graph as a function of $|V|$, compared against the theoretical upper bound of $k = |V|$. The shaded area represents those values of

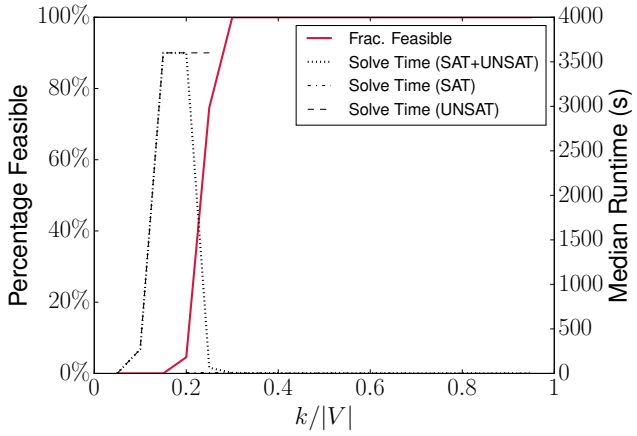


Figure 2: Classical hardness spike near the phase transition for $(k,0)$-representation on real UNOS compatibility graphs.

k where the SAT solver timed out; thus, the reported values of k are a conservative *upper* bound on the required minimum, which is still substantially lower than $|V|$.

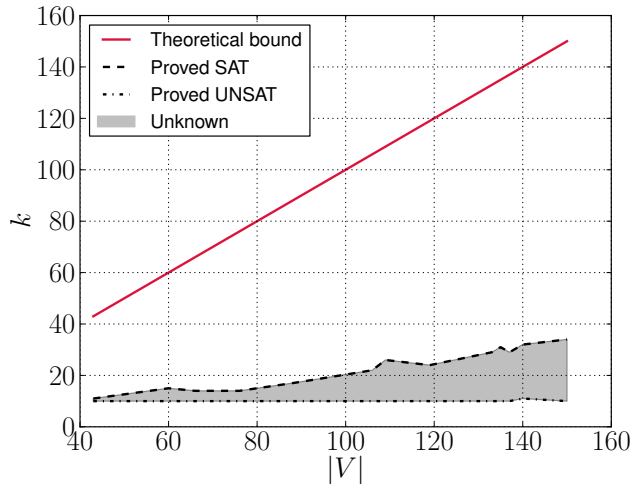


Figure 3: Comparison of number of bits (y-axis) required to $(k,0)$-represent real UNOS compatibility graphs of varying sizes (x-axis). The theoretical bound of $k = |V|$ is shown in red; it is substantially higher than the conservative upper bound of k solved by our SAT solver (upper dotted line).

4.3 Thresholding Effects on Matching Size

One motivation of this work is to provide a principled basis for optimally "flipping bits" of participants (via, e.g., immunosuppresion) in fielded kidney exchanges, in the hope that additional edges in the compatibility graph will result in gains in the final algorithmic matchings. We now explore this line of reasoning—that is, increasing the t in f_{thresh}^t instead of the k, which is now endogenous to the underlying model—on realistic generated UNOS graphs of varying sizes.

Figure 4 shows the effect on the percentage of patient-donor pairs matched by 2- and 3-cycles as a global threshold t is raised incrementally from $t = 0$ (the current status quo) to $t = 5$. Intuitively, larger compatibility graphs result in a higher fraction of pairs being matched; however, a complementary approach—making

the graph denser via even small increases in t—also results in tremendous efficiency gains of 3–4x (depending on $|V|$) over the baseline for $t = 1$, and quickly increasing to all pairs being matched by $t = 5$.

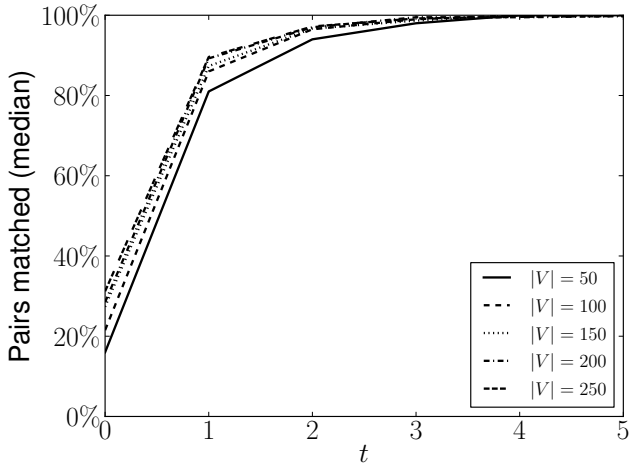


Figure 4: Pairs matched (%, y-axis) in generated UNOS graphs of varying sizes (lines), as t increases (x-axis).

We note that any optimal matching found after increasing a global threshold t could also be created by paying to change at most t bits per vertex in a graph; however, the practical *selection* of the minimum-sized set of at most t bits per vertex such that the size of the resulting optimal matching is equal to that found under the global threshold of t is a difficult two-stage problem and is left as future research. The large efficiency gains realized by moving from f^0_{thresh} to even f^1_{thresh} motivate this direction of research.

5. CONCLUSIONS & FUTURE RESEARCH

Motivated by the increasing size of real-world kidney exchanges, in this paper, we presented a compact approach to modeling kidney exchange compatibility graphs. Our approach is intimately connected to classical intersection graph theory, and can be viewed as the first exploration and practical application of p-intersection digraphs. We gave necessary and sufficient conditions for losslessly shrinking the representation of an arbitrary compatibility graph in this model. Real compatibility graphs, however, are not arbitrary, and are created from characteristics of the patients and donors; using real data from the UNOS US-wide kidney exchange, we showed that using only a small number of attributes suffices to represent real graphs. This observation is of potential practical importance; if real graphs can be represented by a constant number of attributes, then central $\mathcal{NP}$-hard problems in general kidney exchange are solvable in polynomial time.

This paper only addresses the representation of static compatibility graphs; in reality, exchanges are dynamic, with patients and donors arriving and departing over time [37]. Extending the proposed method to cover time-evolving graphs is of independent theoretical interest, but may also be useful in speeding up the (presently-intractable) dynamic clearing problem [4, 10, 2, 11, 17]. Better exact and approximate methods for computing (k, t)-representations of graphs would likely be a prerequisite for that line of research. Furthermore, adaptation of the theoretical results to alternate organ models like lung [15, 24], liver [16], and multi-organ [12] exchange would be of practical use.

Appendix A: Omitted Proofs

In this section, we provide the full proof of Theorem 5. Recall the (k, t)-REPRESENTATION WITH IGNORED EDGES: given an input of a directed graph $G = (V, E)$, a subset F of E, and integers $k \geq 1$ and $t \geq 0$, this problem asks whether there exist bit vectors $\mathbf{d}_i$ and $\mathbf{p}_i$ of length k for each $i \in V$ such that the $\{i, j\} \in F$ if and only if $\langle \mathbf{d}_i, \mathbf{p}_j \rangle \leq t$.

Consider the gadget G_k defined as follows on a graph on $\binom{k}{2} + k$ vertices. Let G^1_k be the graph defined in Theorem 4 on $\binom{k}{2}$ vertices, i.e., the complement of a directed cycle on this many vertices. Associate with each vertex $u \in G^1_k$ a unique element from $\binom{[k]}{2}$ (all subsets of $[k]$ of size 2). Let G^2_k be an independent set of k vertices. For each vertex $i \in G^2_k$, $i \in [k]$, add an incoming edge into i from $u \in G^2_k$ if and only if $i \in S_u$. Figure 5 shows G_4.

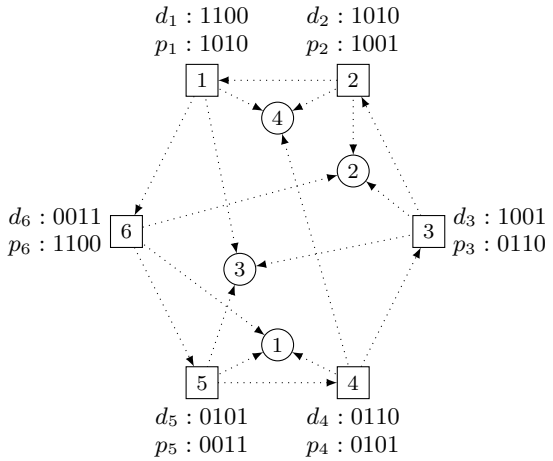


Figure 5: Gadget G_4 with a subset of *non*-edges shown; all edges between circle vertices (those in G^2_4) are also not in E.

Denote the *donor neighborhood* of $i \in V$ by $N_d(i) = \{j \in V : (i, j) \in E, i \neq j\}$, i.e., the set of patients compatible with donor i. Similarly, the *patient neighborhood* of $j \in V$ is $N_p(j) = \{i \in V : \{i, j\} \in E, i \neq j\}$.

LEMMA 1. *There is a unique (up to permutations) $(k, 1)$-representation of G_k.*

PROOF. First consider G^1_k. For all $u \in V(G^1_k)$, since $\{u, u - 1\} \notin E(G^1_k)$, and the compatibility function is f^1_{thresh}, there exist two distinct conflict bits q^u_1 and q^u_2 in $Q_d(u) \cap Q_p(u - 1)$. Moreover, for any u, v distinct, $\{q^u_1, q^u_2\} \neq \{q^v_1, q^v_2\}$. Otherwise, $\{q^u_1, q^u_2\} \subseteq Q_p(v - 1)$ and $\{q^v_1, q^v_2\} \subseteq Q_p(u - 1)$, but at least one of the edges $\{u, v - 1\}$ or $\{v, u - 1\}$ exists in G^1_k.

In addition, $|Q_d(u)| = 2$ for all $u \in V(G^1_k)$. Suppose not, and there exists a third distinct (from q^u_1 and q^u_2) conflict bit q^u_3 in $Q_d(u)$. As the number of vertices is $\binom{k}{2}$, there exists a vertex v_1 with $\{q^{v_1}_1, q^{v_1}_2\} = \{q^u_1, q^u_3\}$, and a (different) vertex v_2 with $\{q^{v_2}_1, q^{v_2}_2\} = \{q^u_2, q^u_3\}$. Then $\{u, v_1 - 1\}$ and $\{u, v_2 - 1\}$ are both not in $E(G^1_k)$. However, u has edges to all vertices except itself and $u - 1$, which is a contradiction, as u, v_1, and v_2 are all distinct. From this, it also follows that $|Q_p(u)| = 2$.

We have thus shown that every vertex $u \in G^1_k$ has exactly two bits set to one in its donor attribute vector, with a unique pair of bits per vertex, and $Q_d(u) = Q_p(u - 1)$. However, without more structure, it is not possible to tell in which donor vectors a particular conflict bit appears. The additional graph G^2_k allows us to identify this, up to permutations.

Since there are no outgoing edges from any of the vertices in G_k^2, and every pair of bits in $\binom{[k]}{2}$ appears in exactly one patient vector of a vertex in G_k^1, each donor vector in G_k^2 must be the all-ones vector of length k.

Consider vertex $i \in [k]$ in G_k^2. It has an incoming edge from each vertex $u \in V(G_k^1)$ such that $i \in S_u$ and it is missing the $\binom{k-1}{2}$ other possible incoming edges from G_k^1 (note that the labeling of the vertices, as well as the choices of the sets S_u, are made without any knowledge of the bit-vectors associated with the vertices). We next show that $\left|\cap_{u \in N_p(i)} Q_d(u)\right| = 1$. That this quantity is at most 1 is clear, as $Q_d(u)$ and $Q_d(v)$ intersect in at most one conflict bit for all $u, v \in V(G_k^1)$, $u \neq v$. If this quantity were 0, then for some $u, v \in N_p(i)$, $Q_d(u) \cap Q_d(v) = \emptyset$. But then at least two zeroes would appear in $Q_p(i)$, which is a contradiction as it implies that i would have more than k incoming edges. Thus, the patient vector $\mathbf{p}_i$ for $i \in V(G_k^2)$ has exactly one zero and ones elsewhere. Moreover, since $N_p(i) \neq N_p(j)$ for any distinct $i, j \in [k]$, it follows that $\mathbf{p}_i \neq \mathbf{p}_j$, so each patient vector is distinct and the position of its only zero is unique. □

LEMMA 2. *Consider a digraph G having G_k as a subgraph and an additional vertex $x \notin V(G_k)$. We use the compatibility function f^1_{thresh} and seek to find a $(k,1)$-representation for the induced subgraph $G[V(G_k) \cup \{x\}]$. Let $U \subseteq V(G_k^1)$ having that property that if $v \in V(G_k^1)$ with $Q_d(v) \subseteq \cup_{u \in U} Q_d(u)$, then $v \in U$. Let $U' = \{u \in V(G_k^1) : u + 1 \in U\}$. Let $Q = \cup_{u \in U} Q_d(u)$.*

If $N_p(x) = V(G_1^k) \setminus U$, then $Q_p(x) = Q$. If $N_d(x) = V(G_1^k) \setminus U'$, then $Q_d(x) = Q$.

PROOF. We use the fact that there are exactly two bits set to one in the donor and patient vectors of each vertex in G_k in any $(k,1)$-representation. For the first statement, since x has no edge from $u \in U$, $Q_p(x) \supseteq Q_d(u)$. Thus $Q_p(x) \supseteq Q$. Now let $v \in V(G_k^1) \setminus U$ and $q_v \in Q_d(v) \setminus Q$. If $q_v \in Q_p(x)$, then for each $q \in Q$, there exists a vertex w in G_k^1 with $Q_d(w) = \{q, q_v\}$, so that $\{w, x\}$ would also not be an edge of G, a contradiction. Hence, $Q_p(x) = Q$. The second statement follows analogously. □

THEOREM 5. *The (k,t)-REPRESENTATION WITH IGNORED EDGES problem is $\mathcal{NP}$-complete.*

PROOF. Consider a 3SAT formula on n variables and with m clauses. Set $k = 2n + 2$, and build the following graph on $2 + n + m + \binom{k}{2} + k$ vertices. The first two vertices are labeled v and u. Then there is a vertex v_i for each variable $i \in [n]$, a vertex c for each clause $c \in [m]$. Call the subgraph induced by these $2+n+m$ vertices G'. The last vertices come from the gadget G_k.

The vertices in G_k^2 ground the k bits used in each donor and patient vector.. We think of the k bits, in order, as corresponding to the n positive literals, then their n negations, followed by two "extra" bits. Then the index of literal x_i will be i, and the index of literal $\bar{x}_i$ will be $n + i$. For i and j distinct in $V(G_k^2)$, $|N_p(i) \cap N_p(j)| = 1$ within G_k. Denote this vertex of G_k^1 by $v(i,j)$, and without loss of generality we can assume that $Q_d(v(i,j)) = \{i, j\}$.

The edges among vertices in the induced subgraph G_k are already defined; we define (a subset) of the rest of the edges. Together, these comprise precisely the subset F of the edges and non-edges specified as an input the instance we are creating of (k,t)-REPRESENTATION WITH IGNORED EDGES.

Vertex v has no incoming edges, and the only outgoing edges from v to $V(G')$ are to every variable vertex v_i, $i \in [n]$. The rest of the vertices that are not in G_k have no outgoing edges at all, to either $V(G')$ or $V(G_k)$, and the only incoming edges are from vertices of G_k^1. Vertex u has an incoming edge from every vertex of G_k^1 except $v(2n+1, 2n+2)$. For each variable vertex v_i, $i \in [n]$, it has an incoming edge from every vertex in $V(G_k^1)$ except $v(i, n+i)$. For each clause $c \in [m]$, let $\{c_1, c_2, c_3\}$ be the indices of the three literals that appear in c. Let $C \subset V(G_k^1)$ be $\{v(c_1,c_2), v(c_1,c_3), v(c_2,c_3), v(c_1,k), v(c_2,k), v(c_3,k)\}$. Then the vertex corresponding to c has an incoming edge from every vertex in $V(G_k^1) \setminus C$.

Every vertex of $V(G')$ except for v will have a donor vector with every bit set to one because there are no outgoing edges to any vertex of G_k^1, and v will have an all-ones patient vector because it has no incoming edges from G_k^1. By Lemma 2, in any $(k,1)$-representation of G, vertex u will have $Q_p(u) = \{2n+1, 2n+2\}$. Variable vertex v_i, $i \in [n]$, will have $Q_p(v_i) = \{i, n+i\}$. Clause vertex $c \in [m]$ will have $Q_p(c) = \{c_1, c_2, c_3, 2n+2\}$.

Since the graph does not have an edge from v to u, $\{2n+1, 2n+2\} \subseteq Q_d(v)$ (these are the only two conflict bits in $Q_p(u)$ and the threshold is 1). Since the graph has an edge from v to each variable vertex v_i, $i \in [n]$, $Q_d(v)$ must contain at most one of the indices corresponding to the variable or its negation (there are no conflicts from the extra bits, which are set to 0 in the patient vector of v_i). Since the graph does not have an edge from v to any of the clause vertices, it has to have at least one conflict bit in a position corresponding to one of the three literals in the clause (the other conflict comes from the extra bit $2n+2$).

Thus, finding a suitable $(k,1)$-representation that satisfies the adjacencies of edges that appear in F would involve finding an appropriate set $Q_d(v)$, which we have shown corresponds to choosing at most one value for each x_i, as well as choosing at least one literal that appears in each clause. This is the same as the problem of finding a satisfying formula for the initial instance of 3SAT.

As an example, consider the 3SAT formula $x_1 \vee \bar{x}_2 \vee x_3$. Figure 6 shows the most relevant part of the graph used in the reduction. One possible $(k,1)$-representation may have $Q_d(v) = \{1, 7, 8\}$, indicating $x_1 = 1$ and the rest of the variables are arbitrary. Another example of a possible representation is $Q_d(v) = \{1, 3, 5, 7, 8\}$, meaning $x_1 = 1$, $x_2 = 0$ (index 5 appears), and $x_3 = 1$. □

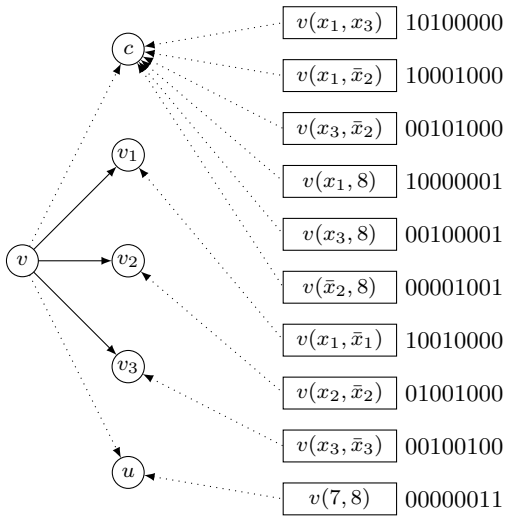


Figure 6: Example of 3SAT reduction to (k,t)-representation.

REFERENCES

[1] D. Abraham, A. Blum, and T. Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 295–304, 2007.

[2] R. Anderson. *Stochastic models and data driven simulations for healthcare operations*. PhD thesis, Massachusetts Institute of Technology, 2014.

[3] R. Anderson, I. Ashlagi, D. Gamarnik, and A. E. Roth. Finding long chains in kidney exchange using the traveling salesman problem. *Proceedings of the National Academy of Sciences*, 112(3):663–668, 2015.

[4] P. Awasthi and T. Sandholm. Online stochastic optimization in the large: Application to kidney exchange. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 405–411, 2009.

[5] A. Biere. Yet another local search solver and lingeling and friends entering the SAT Competition 2014. *SAT Competition*, 2014:2, 2014.

[6] P. Biró, D. F. Manlove, and R. Rizzi. Maximum weight cycle packing in directed graphs, with application to kidney exchange programs. *Discrete Mathematics, Algorithms and Applications*, 1(04):499–517, 2009.

[7] P. Bonami, L. T. Biegler, A. R. Conn, G. Cornuéjols, I. E. Grossmann, C. D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, et al. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, 5(2):186–204, 2008.

[8] P. Cheeseman, B. Kanefsky, and W. Taylor. Where the really hard problems are. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 331–337, Sydney, Australia, 1991.

[9] M. S. Chung and D. B. West. The p-intersection number of a complete bipartite graph and orthogonal double coverings of a clique. *Combinatorica*, 14(4):453–461, 1994.

[10] J. P. Dickerson, A. D. Procaccia, and T. Sandholm. Dynamic matching via weighted myopia with application to kidney exchange. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1340–1346, 2012.

[11] J. P. Dickerson and T. Sandholm. FutureMatch: Combining human value judgments and machine learning to match in dynamic environments. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 622–628, 2015.

[12] J. P. Dickerson and T. Sandholm. Multi-organ exchange: The whole is greater than the sum of its parts. *Journal of Artificial Intelligence Research*, 2016. To appear.

[13] N. Eaton, R. J. Gould, and V. Rödl. On p-intersection representations. *J. Graph Theory*, 21(4):377–392, 1996.

[14] P. Erdős, A. W. Goodman, and L. Pósa. The representation of a graph by set intersections. *Canad. J. Math.*, 18:106–112, 1966.

[15] H. Ergin, T. Sönmez, and M. U. Ünver. Lung exchange, 2014. Working paper.

[16] H. Ergin, T. Sönmez, and M. U. Ünver. Liver exchange, 2015. Working paper.

[17] K. Glorie, M. Carvalho, M. Constantino, P. Bouman, and A. Viana. Robust models for the kidney exchange problem, 2015. Working paper.

[18] K. M. Glorie, J. J. van de Klundert, and A. P. M. Wagelmans. Kidney exchange with long chains: An efficient pricing algorithm for clearing barter exchanges with branch-and-price. *Manufacturing & Service Operations Management (MSOM)*, 16(4):498–512, 2014.

[19] F. Harary, J. A. Kabell, and F. R. McMorris. Bipartite intersection graphs. *Comment. Math. Univ. Carolin.*, 23(4):739–745, 1982.

[20] T. Hogg, B. A. Huberman, and C. P. Williams. Phase transitions and the search problem. *Artificial Intelligence*, 81(1-2):1–15, 1996.

[21] J. N. Hooker. Logic, optimization and constraint programming. *INFORMS Journal of Computing*, 14:295–321, 2002.

[22] IBM ILOG Inc. CPLEX 12.6 User's Manual, 2015.

[23] L. T. Kou, L. J. Stockmeyer, and C. K. Wong. Covering edges by cliques with regard to keyword conflicts and intersection graphs. *Comm. ACM*, 21(2):135–139, 1978.

[24] S. Luo and P. Tang. Mechanism design and implementation for lung exchange. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 209–215, 2015.

[25] V. Mak-Hau. On the kidney exchange problem: cardinality constrained cycle and chain problems on directed graphs: a survey of integer programming approaches. *Journal of Combinatorial Optimization*, pages 1–25, 2015.

[26] T. A. McKee and F. R. McMorris. *Topics in intersection graph theory*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.

[27] R. Montgomery, S. Gentry, W. H. Marks, D. S. Warren, J. Hiller, J. Houp, A. A. Zachary, J. K. Melancon, W. R. Maley, H. Rabb, C. Simpkins, and D. L. Segev. Domino paired kidney donation: a strategy to make best use of live non-directed donation. *The Lancet*, 368(9533):419–421, 2006.

[28] J. Orlin. Contentment in graph theory: covering graphs with cliques. *Nederl. Akad. Wetensch. Proc. Ser. A*, 39(5):406–424, 1977.

[29] B. Plaut, J. P. Dickerson, and T. Sandholm. Fast optimal clearing of capped-chain barter exchanges. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2016.

[30] F. T. Rapaport. The case for a living emotionally related international kidney donor exchange registry. *Transplantation Proceedings*, 18:5–9, 1986.

[31] M. Rees, J. Kopke, R. Pelletier, D. Segev, M. Rutter, A. Fabrega, J. Rogers, O. Pankewycz, J. Hiller, A. Roth, T. Sandholm, U. Ünver, and R. Montgomery. A nonsimultaneous, extended, altruistic-donor chain. *New England Journal of Medicine*, 360(11):1096–1101, 2009.

[32] A. Roth, T. Sönmez, and U. Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119(2):457–488, 2004.

[33] A. Roth, T. Sönmez, and U. Ünver. A kidney exchange clearinghouse in New England. *American Economic Review*, 95(2):376–380, 2005.

[34] A. Roth, T. Sönmez, and U. Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125(2):151–188, 2005.

[35] A. Roth, T. Sönmez, U. Ünver, F. Delmonico, and S. L. Saidman. Utilizing list exchange and nondirected donation through 'chain' paired kidney donations. *American Journal of Transplantation*, 6:2694–2705, 2006.

[36] M. Sen, S. Das, A. B. Roy, and D. B. West. Interval digraphs: an analogue of interval graphs. *J. Graph Theory*, 13(2):189–202, 1989.

[37] U. Ünver. Dynamic kidney exchange. *Review of Economic Studies*, 77(1):372–414, 2010.

[38] T. Walsh. Where are the hard manipulation problems? *Journal of Artificial Intelligence Research*, 42:1–29, 2011.

Convergence and Quality of Iterative Voting Under Non-Scoring Rules

Aaron Koolyk
Hebrew University of Jerusalem
Jerusalem, Israel
aaron.koolyk@mail.huji.ac.il

Omer Lev
University of Toronto
Toronto, Canada
omerl@cs.toronto.edu

Jeffrey S. Rosenschein
Hebrew University of Jerusalem
Jerusalem, Israel
jeff@cs.huji.ac.il

ABSTRACT

Iterative voting is a social choice mechanism whereby voters are allowed to strategically change their stated preferences as the vote progresses, until an equilibrium is reached; at this point, no voter can make a beneficial strategic change. We study iterative voting for several common voting rules and show that, for these rules, such an equilibrium may never be reached. We also consider several variations of iterative voting (restrictions on the allowable changes to votes), and show that with these variations equilibrium may also not be reached. Finally, we do an empirical analysis of the quality of candidates elected through iterative voting.

1. INTRODUCTION

The topic of voting, that is, how to aggregate diverse individual preferences into a collective decision, is of great importance in many automated agent scenarios; it has thus been the topic of much research in multiagent systems. One innovative voting model that was recently proposed is that of *iterative voting* [12]. Whereas classic voting rules usually consist of a single round of ballot submission and announcement of the winner, in iterative voting there can be many such rounds. After each iteration, if any voter wishes to change their vote they may do so, and potentially a new winner replaces the previous one (when multiple such voters exist, an arbitrary voter is chosen). The process terminates when no voter wishes to change their vote. Iterative voting thus embraces the inevitable manipulability of voting shown in the Gibbard-Satterthwaite theorem [20, 6], and considers agents' uniform ability to vote strategically as a collective opportunity.

Besides being an intriguing method for reaching consensus, iterative voting has been proposed as a formal solution concept for voting games. Standard Nash equilibria are of limited usefulness in voting games where the group outcome is generally robust to changes in any single voter's action. The set of iterative voting equilibria, however, is a subset of Nash equilibria, and in particular those iterative voting equilibria reachable from the truthful profile could be considered a more natural (or meaningful) solution concept.

The most salient questions regarding iterative voting thus have two interpretations. Regarding iterative voting as a method for reaching an outcome, we ask whether the process terminates; if so, with what complexity; and does it arrive at "good" outcomes. Regarding iterative voting as a solution concept, we must explore the existence of solutions; the equilibria computation; and notions of price of stability/anarchy.

Most previous research on iterative voting has focused on plurality, with several extensions to other scoring rules. However, in this work we look into previously unexplored voting rules which are not scoring rules—Maximin, Copeland, Bucklin, STV, Second Order Copeland (SOC), and Ranked Pairs. In the process of investigating these voting rules, we design dynamics under which the iterative voting might evolve. While we show that convergence is not guaranteed, we proceed to analyze the outcomes of the iterative dynamic empirically, showing that cycles are not very common, and moreover, the outcomes are generally very good.

2. RELATED LITERATURE

There has been extensive research on solution concepts of voting games, and an overview of the research can be seen in Meir et al. [11]. Due to space constraints, we focus in this section on the iterative model which we extend.

Our model of iterative voting was initiated by Meir et al. [12], who showed that Plurality voting converges under a natural restricted best-response dynamic and linear ordered tie-breaking (a dynamic refined in [10]). Lev and Rosenschein [8] (and in parallel [19]) later showed that Veto, with a similarly natural restricted best-response dynamic, also converges. However, [8] and further work showed that no other scoring rules converge for best-response dynamics, as well as showing that allowing non-linear tie-breaking will result in Maximin not converging (though they showed no result on Maximin with linear tie-breaking).

Reijngoud and Endriss [18] added an epistemic element by varying the amount of information revealed at each stage, and also showed that any scoring rule converges under the k-pragmatism dynamic. Grandi et al. [7] showed that, for two additional restrictive dynamics, scoring rules (as well as Copeland and Maximin), converge, and Loreggia [9] added another very restrictive dynamic, showing that Copeland and Maximin converge under it. Obraztsova et al. [16] abstracted these ideas and put forth two theoretical properties which suffice to guarantee convergence. Not in connection to iterative voting, Obraztsova and Elkind [13] proposed several dynamics, of which we adopt, for example, the Kendall-Tau dynamic.

Research examining the properties of iterative voting oc-

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

curred in parallel. Meir et al. [11] showed that a generalization of iterative voting, where voters act under uncertainty, also converges for Plurality. [17, 14] considered questions of computational complexity related to iterative voting, with and without assumptions about "truth bias" and "lazy bias" on the part of voters.

Brânzei et al. [3] examined the *quality* of iterative voting, via the notion of the dynamic price of anarchy. They showed lower bounds for Plurality, Veto, and Borda, and a tight upper bound for Plurality. Additional work on the quality of iterative voting includes that of [11, 18, 7] (all mentioned in other contexts, above) who showed through simulations some improvements in the outcome of elections, in their various versions of iterative voting. However, the closest work in its pattern of simulations and quality measures is Thompson et al. [21], which analyzed truth-biased equilibria, without any assumption regarding their dynamics.

3. PRELIMINARIES

Our setting will be the standard voting model that includes a set of voters $V, |V| = n$, and a set of candidates $C, |C| = m$. Each voter i has a strict preference order $\succ_i$ over C, that is, a complete, reflexive, transitive, and anti-symmetric binary relation over C. Denote the set of all such preference orders as $\mathcal{P}(C)$. A profile

$$\vec{\succ} = (\succ_1, \succ_2, ..., \succ_n) \in \mathcal{P}(C)^n$$

is a vector of n preference orders, one for each voter. We denote by

$$\vec{\succ}_{-i} = (\succ_1, ..., \succ_{i-1}, \succ_{i+1}, \ldots, \succ_n) \in \mathcal{P}(C)^{n-1}$$

the profile of the voters excluding i and $(\vec{\succ}_{-i}, \succ_i) = \vec{\succ}$. We shall denote the truthful preferences of voters as $\vec{tr} = (\succ_1^{tr}, \ldots, \succ_n^{tr})$.

We model a collective decision through one of two functions. A *social welfare function* is a function $f : \mathcal{P}(C)^n \to \mathcal{P}(C)\backslash\{\emptyset\}$ and a *voting rule* is a function $F : \mathcal{P}(C)^n \to 2^{\mathcal{C}}\backslash\{\emptyset\}$. So, given a (not necessarily truthful) vector of preferences, a social welfare function chooses a preference order and a voting rule chooses a set of candidates. When a voting rule is irresolute, and we would like a unique winner, we use a *tie breaking rule*, a function $t : 2^{\mathcal{C}} \to \mathcal{C}$. A *linear-ordered tie breaking rule* is a rule that breaks ties according to a fixed linear order. It will be without loss of generality throughout this paper that the linear-ordered tie breaking rule we use is the *lexicographic tie breaking rule*, where ties are broken according to the lexicographic order of candidates' names.

3.1 Voting Rules

We shall investigate the following voting rules:

Maximin For each pair of candidates c_1, c_2 let $P(c_1, c_2) = |\{x \in V | c_1 \succ_x c_2\}|$. For each candidate c, let $sc(c) = \min_{c' \neq c \in C} P(c, c')$. The candidates with the maximum score, $\arg\max_{c \in C} sc(c)$, win.

Copeland Let $P(c_1, c_2) = |\{x \in V | c_1 \succ_x c_2\}|$, as above. For $\alpha \in [-1, 1]$, let $sc(c) = |\{c' | P(c, c') > n/2\}| - |\{c' | P(c, c') < n/2\}| + \alpha \cdot |\{c' | P(c, c') = n/2\}|$, and the candidates with the maximum score, $\arg\max_{c \in C} sc(c)$, win. (Generally $\alpha = 0$ is assumed).

Bucklin For each $c \in C$, let $sc(c) = \min_{k<m} |\{x \in V | \exists c_1, \ldots c_{m-k} \text{ s.t. } c \succ_x c_i\}| > n/2$. The winner is the candidate with the smallest score, $\arg\min_{c \in C} sc(c)$.

STV Under Single Transferable Voting (STV), the election proceeds in rounds. In each round, the candidate with the lowest plurality score is eliminated and any voter voting for them transfers their vote to their next ranked candidate. The last remaining candidate is the winner.

SOC Second Order Copeland (SOC) chooses winners as in Copeland, except that ties are broken according to the score of defeated candidates. If $sc(c)$ is the Copeland score of c, then Second Order Copeland chooses $c \in \arg\max_{c \in C} sc(c)$ s.t. $\sum_{c': P(c,c') > n/2} sc(c')$ is maximal.

Ranked Pairs (RP) Let $P(c_1, c_2) = |\{x \in V | c_1 \succ_x c_2\}|$, as above. Let

$$O = (P(c_{i_{1,1}}, c_{i_{1,2}}), P(c_{i_{2,1}}, c_{i_{2,2}}), ..., P(c_{i_{\binom{m}{2},1}}, c_{i_{\binom{m}{2},2}}))$$

be the sorted list of pairs of candidates' P-score such that

$$P(c_{i_{j,1}}, c_{i_{j,2}}) \geq P(c_{i_{j+1,1}}, c_{i_{j+1,2}}).$$

If $P(c_{i_{j,1}}, c_{i_{j,2}}) = P(c_{i_{j+1,1}}, c_{i_{j+1,2}})$, then

$$P(c_{i_{j,1}}, c_{i_{j,2}}) \succ_O P(c_{i_{j+1,1}}, c_{i_{j+1,2}})$$

iff $i_{j,1} < i_{j+1,1}$ or $i_{j,1} = i_{j+1,1}$ and $i_{j,2} < i_{j+1,2}$; i.e., break ties in order lexicographically (first candidate, second candidate). A ranking is constructed by the following algorithm. For j=0 to $\binom{m}{2}$ fix $c_{i_{j,1}} \succ c_{i_{j,2}}$ unless this contradicts a previous step (including by transitivity). The candidate at the top of the constructed ranking is selected as the winner.

An interesting property of which we will make use regards the *Condorcet winner*. A Condorcet winner is a candidate who is preferred to each other candidate by more than half of the voters; however, such a winner does not always exist. A voting rule is *Condorcet consistent* if whenever there is such a Condorcet winner, it is the election's outcome. Among the voting rules we discuss, Maximin, Copeland, SOC and Ranked Pairs are Condorcet consistent, while Bucklin and STV are not.

3.2 Dynamics

We will call a binary relation $\mathcal{D} \subset \mathcal{P}(C)^n \times \mathcal{P}(C)^n$ a *dynamic*. We call a (possibly finite) sequence of profiles $(\vec{\succ}_1, \vec{\succ}_2, ...) \in \mathcal{P}(C)^*$ a *profile sequence* and a (possibly finite) sequence of voters $(v_1, v_2, ...) \in V^*$ a *voter sequence*. A profile sequence $(\vec{\succ}_1, \vec{\succ}_2, ...)$ for which $\vec{\succ}_1$ are the truthful preferences, is called an *initially truthful* profile sequence.

We will say a profile sequence is *valid* for a dynamic $\mathcal{D}$ if $\forall i (\vec{\succ}_i, \vec{\succ}_{i+1}) \in \mathcal{D}$. We will mainly be concerned with dynamics for which all elements differ in a single preference, i.e.,

$$\forall [(\vec{\succ}^{(1)}, \vec{\succ}^{(2)}) \in \mathcal{D}] \;\; \exists i \in V \;\; \text{s.t. } \vec{\succ}^{(1)}_{-i} = \vec{\succ}^{(2)}_{-i}.$$

In such a case, a profile sequence induces a voter sequence $(v_1, v_2, ...)$ where v_i is the voter whose preference changed

at stage i. Likewise, a voter sequence defines a set of profile sequences by which it is induced. A voter sequence will be called *valid* if it is induced by some valid profile sequence.

The final element of a (finite) valid profile sequence $(\vec{\succ}_1, \vec{\succ}_2, ..., \vec{\succ}_k)$ will be called an *equilibrium* if there is no $\vec{\succ}_{k'}$ such that $(\vec{\succ}_k, \vec{\succ}_{k'}) \in \mathcal{D}$.

For a dynamic $\mathcal{D}$ and voting rule F with tie breaking rule t, let $\mathcal{I}(\mathcal{D}, F_t) = \{s | s \text{ is a valid profile sequence for } \mathcal{D}(F_t)\}$. We will say that *iterative-F converges* (or *always converges*) under $\mathcal{D}$ if every element of $\mathcal{I}(\mathcal{D}, F_t)$ is finite. Otherwise, we will say that iterative-F under $\mathcal{D}$ *cycles* (or *may cycle*) or does not converge (or may not converge). Notice that, as defined, the semantics of convergence are asynchronous. $\mathcal{I}(\mathcal{D}, F_t)$ converges if *every* element is finite, and is not limited to, say, a "fair schedule of play."

The dynamics we shall consider will be influenced by the truthful preferences, i.e., a dynamic in which a voter's vote changed must have increased the utility of that vote. Two main dynamics have been investigated (e.g., in Meir et al. [12]). An ordered pair of profiles is in the *better response* dynamic if the preferences of all voters but one are identical in the two profiles, and the voter whose preference changes prefers the outcome of the second profile to that of the first profile. In game-theoretic terms, any time a single player can make a better response to a given state, such a move is included in the dynamic. Formally, for two profiles $\vec{\succ}^{(1)}, \vec{\succ}^{(2)}$ and a voting rule F, $(\vec{\succ}^{(1)}, \vec{\succ}^{(2)}) \in$ BetterResponse iff:

$$\exists i \in V \text{ s.t. } \vec{\succ}^{(1)}_{-i} = \vec{\succ}^{(2)}_{-i} \text{ and } F(\vec{\succ}^{(2)}) \succ^{tr}_i F(\vec{\succ}^{(1)}).$$

Such an i is called the *manipulator*, $\succ^{(2)}_i$ is called the *new vote*, and $\succ^{(1)}_i$ is called the *old vote*. Notice that a stable state under this dynamic is a Nash equilibrium.

Similarly, an ordered pair of profiles is in the *best response* (BR) dynamic if the preferences of all voters but one are identical; the voter whose preference changes prefers the outcome of the second profile to that of the first profile (so it is contained in the better response dynamic); and of all possible changes to his preferences, the outcome under the second profile is preferred at least as much as the outcome under any other possible profile. Formally, $(\vec{\succ}^{(1)}, \vec{\succ}^{(2)}) \in$ BR iff:

$$\exists i \in V \text{ s.t. } \vec{\succ}^{(1)}_{-i} = \vec{\succ}^{(2)}_{-i} \text{ and } F(\vec{\succ}^{(2)}) \succ^{tr}_i F(\vec{\succ}^{(1)})$$

and

$$\forall \succ'' \in \mathcal{P}(C) \text{ s.t. } (\vec{\succ}^{(1)}_{-i}, \succ'') \neq \vec{\succ}^{(2)}, F(\vec{\succ}^{(2)}) \succeq^{tr}_i F(\vec{\succ}^{(1)}_{-i}, \succ'').$$

The above description clearly defines a game form. The set of voters is the set of players, the set of preferences is the set of strategies available to each player, and the voting rule determines the outcome of a strategy profile. Ordinal utilities are given by true preference orders. An equilibrium under Best Response (or Better Response) is a Nash equilibrium.

4. DYNAMICS

The study of best response dynamics is prolific, but in the iterative voting context, particular forms of best response have been utilized in the convergence proofs of both plurality [12] and veto [8]. For non-scoring rules, however, there is no immediately clear choice of best response form (indeed, in some cases, like STV, it is NP-complete to calculate what it is). We present here several dynamics that may serve as natural heuristics for a potential voter. There have been dynamics designed with the express purpose of ensuring convergence, as in k-pragmatism, M1, and M2 [18, 7]. However, we propose the following as possibly more natural correspondences to the strategic behavior of self-interested agents.

TOP: This dynamic assigns the candidate which the voter wishes to make a winner the top spot in the new preference order. In many of the voting rules we consider (and any weakly-monotone rule) this dynamic is a subset of the best-response dynamic (i.e., $TOP(\mathcal{P}(C)) \subset BR(\mathcal{P}(C))$), and, indeed, it generalizes the dynamic used in Meir et al. [12].

TB: This dynamic requires the new winner to be at the top of the new ballot, and the previous winner to be at the bottom. While in many scoring rules (e.g., plurality and veto) this is a subset of best response moves (and generalizes those used in Lev and Rosenschein [8]), this is not true in general, and particularly in the voting rules we study in this work.

KT: This dynamic restricts best response to those with minimum Kendall-Tau distance from the previous vote. That is, among all possible moves whose outcome will be the most preferred possible candidate, one with the minimal Kendall-Tau distance[1] from the current vote is chosen.

SWAP: This dynamic, inspired in part by notions from the literature on bribery (see, e.g., [5, 4]), is quite restrictive. It restricts manipulations to a single swap (called a 'shift' in the bribery literature) or even a single adjacent swap (i.e., changing to a vote within Kendall-Tau distance of one from the current vote; a 'swap' in the bribery nomenclature).

5. CONVERGENCE

In this section we consider the convergence of iterative voting for several voting rules. We distinguish between the first three, for which there exists a polynomial time algorithm for a single voter to compute a best response manipulation, and the last three for which such a computation is NP-Complete [2, 1, 22]. In reversal of the common situation in computational social choice, for iterative voting polynomial manipulation is actually quite felicitous.

A note on reading the examples: each column represents a profile of submitted ballots (beginning with the truthful one). The final row in the column indicates the winner of the profile. The i-th entry in a column represents voter i's submitted preferences, where, for example, ABC is to be read $A \succ_i B \succ_i C$. Arrows highlight the changed preference between two profiles at a given stage. The profile sequence formed by continual repetition of the indicated profiles thus forms an infinite element of $\mathcal{I}(\mathcal{D}, F_t)$ and proves non-convergence. Due to space constraints, we omit several proofs and examples.

5.1 Maximin

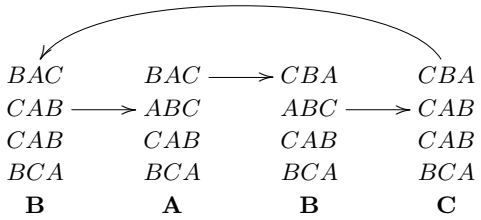
Similar to plurality and veto, Maximin changes gradually. The difference in score between the previous winner and the new one, when a single voter manipulates, can go up or down by at most one point. One might thus expect there to be an argument for convergence, similar to plurality/veto. But in fact, convergence with Maximin turns out to be elusive even

[1]For $a, b \in \mathcal{P}(C)$, the Kendall-Tau distance between them is defined as $dist(a, b) = |\{i, j\} \in V | (i \succ_a j \text{ and } j \succ_b i) \text{ or } (j \succ_a i \text{ and } i \succ_b j)\}|$.

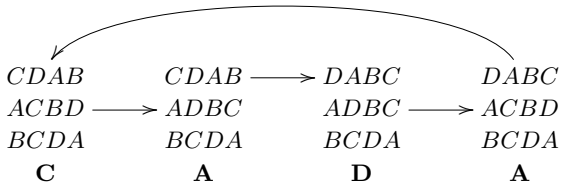
after major restrictions on the allowable moves.

THEOREM 1. *Maximin with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP.*

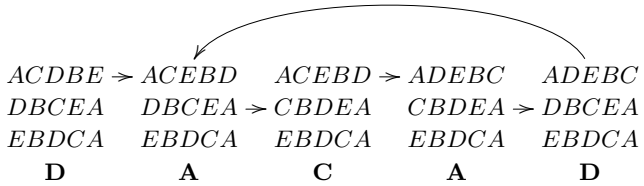
PROOF. For BR, the example is:



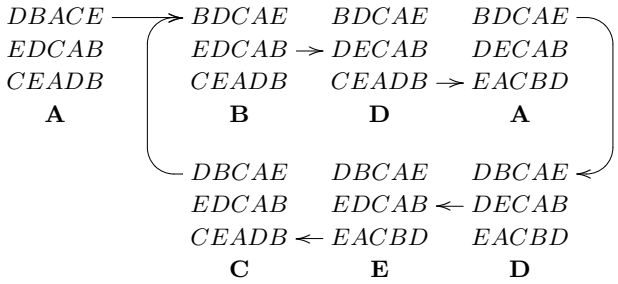
For TOP, the example is:



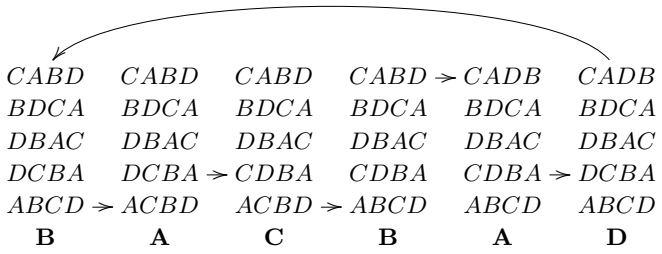
For TB, the example is:



For KT, the example is:



And, finally, for SWAP, the example is:



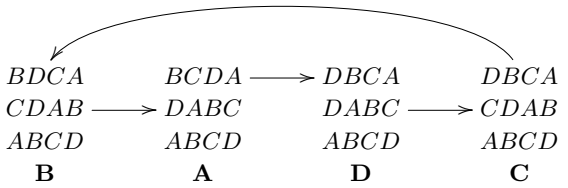
Although the changes to the winner's score are as gradual in Maximin as in plurality and veto, the exponential blowup in strategy space seems to make convergence harder. Whereas in plurality and veto, a voter's ballot reduces to a single candidate, in Maximin a ballot depends on the entire ranking. □

5.2 Copeland

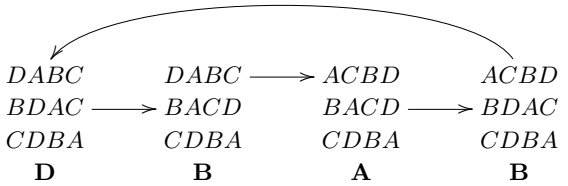
THEOREM 2. *Copeland with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP. This holds for Copeland$^\alpha$ for any α.*

PROOF. Since the number of voters in all our examples is odd, they hold for Copeland$^\alpha$ for any α.

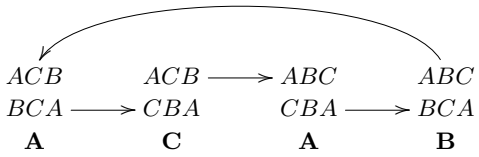
The example for BR:



The example for the TOP dynamic:

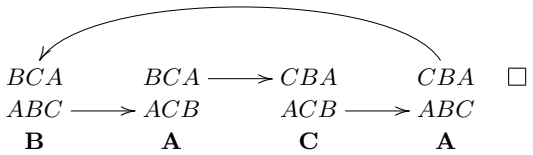


Using the TB dynamic and moving the desired winner to the top and the current undesired winner to the bottom does not suffice to avoid cycles:



The exact same example as TB also serves to show restricting best response by minimum Kendall-Tau distance does not suffice to avoid cycles.

Finally, restrictions to a single adjacent swap does not suffice:

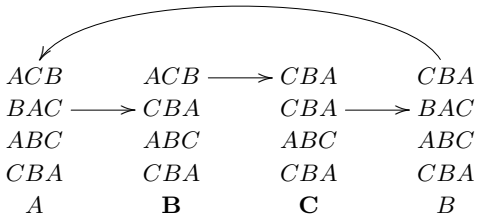


□

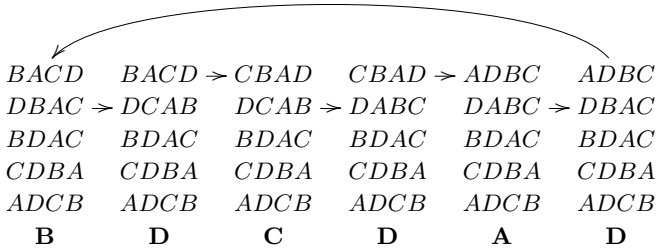
5.3 Bucklin

THEOREM 3. *Bucklin with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP.*

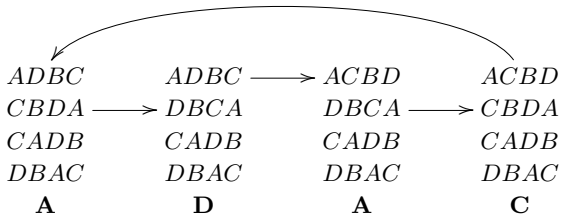
PROOF. The example for BR:



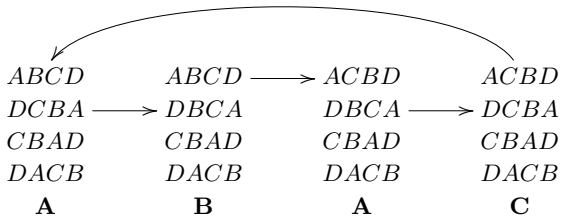
The example for TOP:



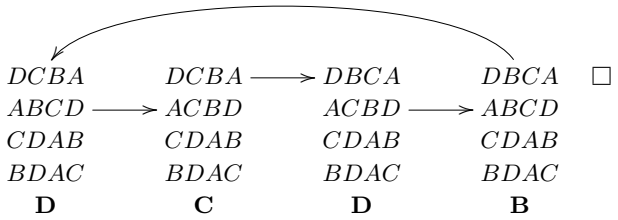
The example for the TB dynamic:



The example for the KT dynamic:



The example for the SWAP dynamic:

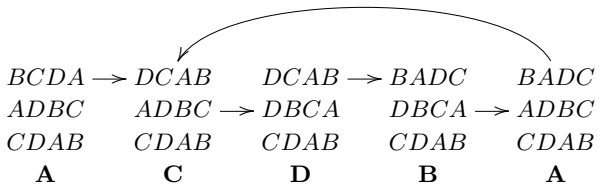


□

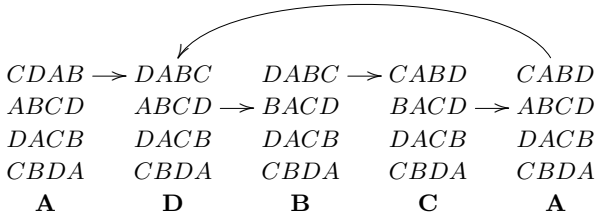
5.4 STV

THEOREM 4. *STV with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP.*

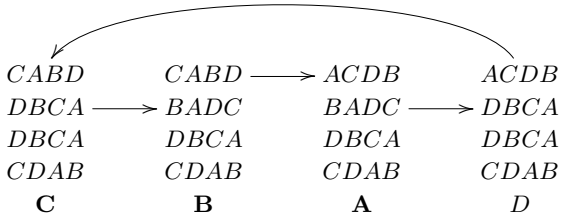
PROOF. The example for the BR dynamic:



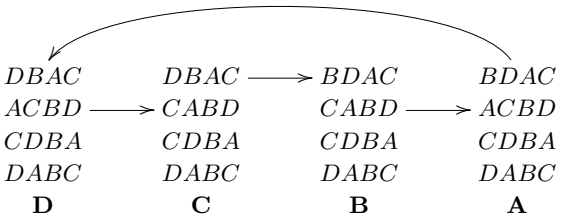
The example for the TOP dynamic:



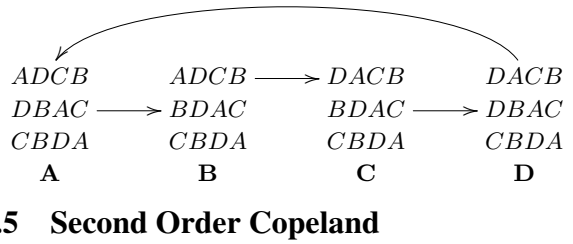
The example for the TB dynamic:



The example for the KT dynamic:



The example for the SWAP dynamic:

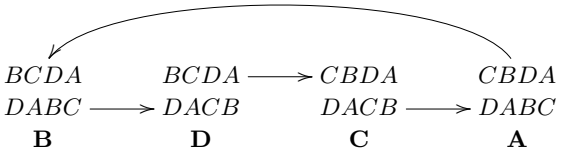


□

5.5 Second Order Copeland

THEOREM 5. *SOC with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP.*

PROOF. The example for BR is:



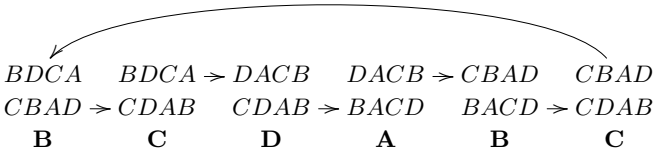
The examples for the other dynamics are the same as those for Copeland. □

5.6 Ranked Pairs

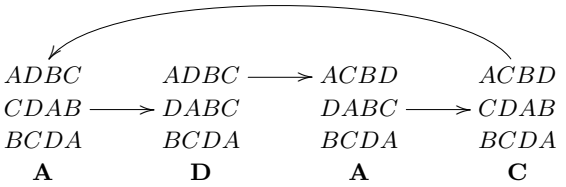
In Ranked Pairs, as in other voting rules which output a complete ranking, a stronger convergence property could be defined for the entire ranking, but convergence is elusive even for just the top element of the ranking (the winner of Ranked Pairs).

THEOREM 6. *Ranked pairs with linear order tie-breaking does not converge for the dynamics BR, TOP, TB, KT and SWAP.*

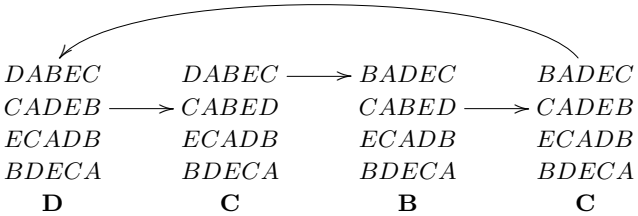
PROOF. The example for the BR dynamic:



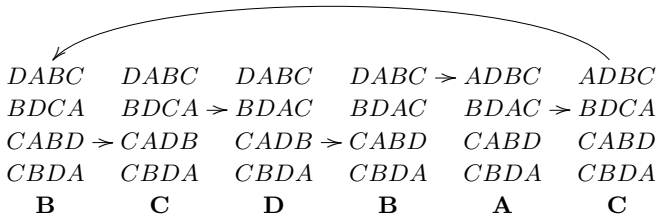
The example for the TOP dynamic:



The example for the TB dynamic:



The example for the KT dynamic:



The above example, is also for the SWAP dynamic, as all changes are of Kendall-Tau distance of one.

□

6. EMPIRICAL ANALYSIS OF OUTCOMES

We now consider the behavior of iterative voting, and the quality of its outcomes, through the results of empirical simulations. Assessing the quality of a voting method can be subtle. One general methodology is the *a posteriori* approach, to judge a rule by the quality of its outcome. Yet there is no definitive agreed-upon measure of quality of voting rules. Moreover, some voting rules have been designed with a particular measure of quality in mind, such as Maximin, ensuring the core number of supporters a candidate has, against any other one, is maximal.

Another compelling criterion is Condorcet efficiency. If a candidate is preferred to each other candidate by a majority of voters, there is reason to think it should be the winner. Thus it could be interesting to consider how often, for a given distribution of voter preferences, a rule chooses the Condorcet winner.

Furthermore, we wish to estimate social welfare. Social welfare's utility is limited in voting settings as we generally do not have the cardinal utility functions of our voters. However, as has been suggested in previous research, we can use the Borda score on the truthful preferences, in which the utility of each voter of an outcome is $m - i$, if the winner is candidate $c \in C$ which the voter ranks in place i.

As noted before, among the rules we considered in the previous section, Maximin, Copeland, Ranked Pairs, and Second Order Copeland are all Condorcet consistent. For these, therefore, we can only consider how much worse iterative voting can be than truthful voting (though, of course, we know voters do not actually always vote truthfully, so it is not as if iterative plurality is necessarily worse than a non-iterative model). For both STV and Bucklin—which are not Condorcet consistent—there is a possibility that iterative voting could have greater Condorcet efficiency than static voting. And for all rules, we can compare the Borda score of the truthful winner with the (truthful) Borda score of the equilibrium winner.

Our method is influenced by those of [21, 18, 7, 11]. Unlike [18, 7], however, which study iterative voting under restrictive dynamics (M1, M2, and k-pragmatism), we choose to analyze iterative voting under best-response dynamics. Unrestricted best-response is both computationally taxing as well as possibly cyclical. Nevertheless, as the most basic form of iterative voting, it seems to us to be of the greatest interest.

Our findings are the results of simulations of iterative voting for the six rules we have studied. Simulations were run for each rule twice, once with 10 voters and once with 25. Both runs had 4 candidates. For each set of parameters, 10,000 initial (truthful) profiles were sampled uniformly at random. Each profile evolved, for each voting rule, under best response dynamics and was run to completion (or detection of a cycle) 20 times. In keeping with the asynchronous conception of iterative voting, each of the 20 executions were developed at each step by uniformly sampling a voter with a potential move and uniformly sampling a move from all of that voter's possible best-response moves.

We begin with statistics regarding the behavior of best response in the iterative version of the various rules. Table 1 shows the average number of steps in each of the 200,000 paths considered per setting. Many of the 10,000 initial profiles were Nash equilibria, and so non-manipulable. Therefore, a significant fraction of the paths were of length zero. For convenience, we also include the average path length among non-zero-length paths. Finally, we show the maximum length among the 200,000 paths.

Voting rule	Avg. length	Corrected avg. length	Max length	Nash equilibrium share in truthful prefs
Maximin 10	2.29	8.66	143	73.5%
Maximin 25	4.97	25.19	325	80.24%
Copeland 10	6.42	19.42	269	66.55%
Copeland 25	5.51	24.69	504	77.65%
Bucklin 10	3.24	7.32	232	55.55%
Bucklin 25	4.67	11.66	301	59.94%
STV 10	0.94	4.46	97	78.84%
STV 25	1.65	8.00	182	79.32%
SOC 10	5.55	17.38	254	67.76%
SOC 25	5.66	26.00	306	78.18%
RP 10	2.13	8.55	131	75.03%
RP 25	4.18	22.83	376	81.7%

Table 1: Path Lengths

Unsurprisingly, as the number of voters grows, the probability of sampling a Nash equilibrium grows as there is a larger probability of the difference between the winner and the runner up being large enough so a single voter cannot change it (using similar reasoning, increasing the number of candidates would have increased the chances of strategic moves). On the other hand, the path lengths with more voters are longer than those with fewer voters. In the (rarer) case that an elections is close, more voters can participate in the strategic process. Copeland (and Second Order Copeland) tended to have longer paths and STV had especially short paths, but in general the NP-Complete rules did not have shorter paths than polynomial rules.

Before we begin analyzing the quality of the outcomes, we remark on an important point of relevance to the previous section.

For all of the rules, cycles occur quite rarely—the highest share of cycles was 0.57%, though most were well under 0.1%; see Figure 1. So although we have shown that all these rules *can* cycle, the frequency with which they do is very low. Copeland (and SOC), which exhibited the greatest (non-cyclic) path length, also tends to cycle more often than other rules, but it too cycles quite infrequently. STV, which has especially short paths, also cycles less frequently, but

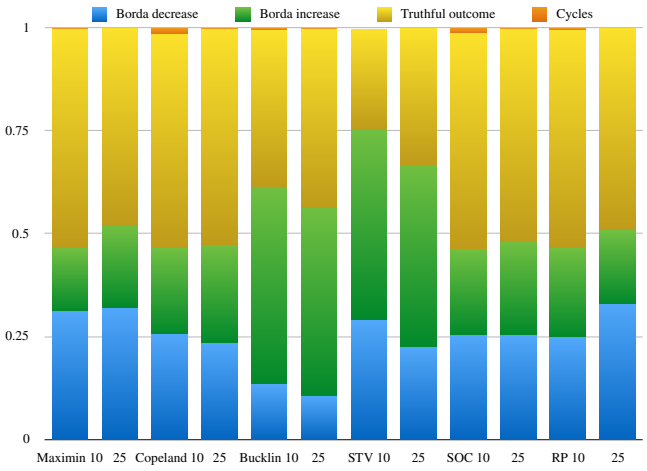


Figure 1: Out of all iterative processes (i.e., not including cases where the truthful preferences were a Nash equilibrium), the share of outcomes which decreased the Borda score, increased it, truthful outcome (of each voting rule), and the share of runs which ended in a cycle.

in general there does not appear to be a distinction in the number of cycles between NP-Complete rules and P rules.

For more voters, there are many fewer cycles, apparently because of the sparsity of cycles and uniform choice of paths.

The rarity of cycles suggest that perhaps iterative voting could be used even with these rules and best response dynamics. In the rare case of a cycle it could be detected and turned over to some cycle-breaking rule, e.g., either running the election again or deciding amongst the different outcomes in the cycle.

Turning to assess the observed outcomes of iterative voting, we first note that quite often iterative voting leads to the original outcome. Many such instances are the result of original profiles which are non-manipulable. But many are also the result of manipulations, whose equilibrium reverted to the original winner. Of the manipulable profiles (see Figure 1), the non-Condorcet consistent rules (Bucklin and STV) behave differently than the others—both of them have fewer than 50% of the outcomes truthful, but their ratio increases as voters grow, unlike the rest of the voting rules. Copeland exhibited the most consistency when increasing the number of voters.

Next we assess the change in Borda score (our proxy for social welfare) and Condorcet efficiency. As can be seen in Figure 1, once again Bucklin and STV behave significantly differently than other voting rules—for them, a significant number of outcomes increase the Borda score as compared to the truthful outcome. This is almost always not the case for the Condorcet consistent rules. However, in all Condorcet consistent rules except Maximin the share of outcomes which decreased the Borda score was close to the share of those that increased it. In all of them but Ranked Pairs this difference was decreased further when the number of voters increased. In all Condorcet consistent voting rules the average Borda score of the outcome was below that of the truthful outcome, but only slightly so—less than 1 point difference. Contrary to that, both Bucklin and STV's average Borda score was above their truthful one, and for Bucklin significantly so (above a 2 point difference).

With regard to Condorcet efficiency, we consider, for each of the 10,000 profiles, whether a Condorcet Winner existed in the original profile, whether it is selected by the voting rule, and whether it is was the outcome in a reached equilibrium. The latter is presented in Table 2 in terms of efficiency (out of 10,000) after aggregating equilibria over non-cycling paths.

Voting rule	# of Profiles with a Condorcet winner	Share of outcomes with Condorcet winner
Maximin 10	4764	0.47
Maximin 25	8413	0.82
Copeland 10	4764	0.47
Copeland 25	8413	0.81
Bucklin 10	3717	0.45
Bucklin 25	4461	0.58
STV 10	4610	0.47
STV 25	7795	0.83
SOC 10	4764	0.47
SOC 25	8413	0.81
RP 10	4764	0.47
RP 25	8413	0.82

Table 2: Comparing Condorcet Efficiency (of 10,000 profiles)

Of the 10,000 profiles sampled with n=10, there were 4764 for which a Condorcet Winner existed; among those with n=25, there were 8413 with a Condorcet Winner. As mentioned, Maximin, Copeland, Ranked Pairs, and Second Order Copeland are Condorcet-consistent, so efficiency has only one direction to move (downward). Yet it does so by very little, although slightly more when there are more voters.

Of the two rules that are not Condorcet consistent, Bucklin and STV improve their efficiency under iterative voting. These two rules also fared well under Borda criteria, suggesting that iterative Bucklin and iterative STV could be considered improvements on their static counterparts. Interestingly, Bucklin is also the most manipulable among the rules (it contained the fewest number of paths of size zero).

7. CONCLUSION AND DISCUSSION

In this work we have continued the exploration of iterative voting. We have done so in two dimensions. In the first, we expanded the set of dynamics to include some which reflect strategic behavior, but restrict best response in a, to a certain extent, natural way—whether by constraining the placement of affected candidates, or by prioritizing minor ballot changes. In the second dimension, we have ventured beyond scoring rules, and have shown that for Maximin, Copeland, Bucklin, STV, Second Order Copeland, and Ranked Pairs, iterative voting under best response dynamics does not always converge. Even after restricting the dynamics to allow voters only limited changes to their ballots they still do not always converge.

On the other hand, we have shown empirically that cycles seem to occur rather infrequently with all of these rules. Furthermore, we have shown that iterative voting, according to

certain common criteria, does not perform much worse, and sometimes does better, than non-iterative voting. Notably, in non-Condorcet consistent rules—Bucklin and STV—the winners tend to improve significantly through iterative voting with regard to both their Borda score and Condorcet efficiency.

Continuation of this line of work would include analysis of convergence conditions for more voting rules and additional dynamics, with an aim towards discovering convergence dynamics, or establishing broader impossibility results. The empirical aspect of this work would benefit from expanding the analysis, for example by analyzing more distributions than we had space to include here (e.g., the Mallows model). More generally, the study of iterative voting would be greatly enhanced by incorporating voter learning into the model, and endowing voters with a greater degree of strategic (non-myopic) capabilities (early work in this direction includes Obraztsova et al. [15]).

Acknowledgements

This research has been partly funded by Microsoft Research through its PhD Scholarship Program, and by Israel Science Foundation grant #1227/12. This work was also supported by NSERC grant 482671.

REFERENCES

[1] J. J. Bartholdi III and J. B. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8(4):341–354, 1991.

[2] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[3] S. Brânzei, I. Caragiannis, J. Morgenstern, and A. D. Procaccia. How bad is selfish voting? In *Proceedings of the 27th National Conference on Artificial Intelligence (AAAI)*, pages 138–144, Bellevue, Washington, July 2013.

[4] R. Bredereck, J. Chen, P. Faliszewski, A. Nichterlein, and R. Niedermeier. Prices matter for the parameterized complexity of shift bribery. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada.*, pages 1398–1404, July 2014.

[5] E. Elkind, P. Faliszewski, and A. M. Slinko. Swap bribery. In *Proceedings of the Second International Symposium on Algorithmic Game Theory SAGT*, pages 299–310, Paphos, Cyprus, October 2009.

[6] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41(4):587–602, 1973.

[7] U. Grandi, A. Loreggia, F. Rossi, K. B. Venable, and T. Walsh. Restricted manipulation in iterative voting: Condorcet efficiency and borda score. In *Proceedings of 3rd International Conference of Algorithmic Decision Theory (ADT)*, pages 181–192, Brussels, Belguim, November 2013.

[8] O. Lev and J. S. Rosenschein. Convergence of iterative voting. In *The Eleventh International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, pages 611–618, Valencia, Spain, June 2012.

[9] A. Loreggia. Iterative voting and multi-mode control in preference aggregation. Master's thesis, University of Padova, 2012. `http://www.math.unipd.it/~loreggia/pdf/tesi2012.pdf`.

[10] R. Meir. Plurality voting under uncertainty. In *Proceedings of the 29th Conference on Artificial Intelligence (AAAI)*, pages 2103–2109, Austin, Texas, January 2015.

[11] R. Meir, O. Lev, and J. S. Rosenschein. A local-dominance theory of voting equilibria. In *Proceedings of the 15th ACM conference on Economics and Computation (EC)*, pages 313–330, Palo Alto, Califronia, June 2014.

[12] R. Meir, M. Polukarov, J. S. Rosenschein, and N. Jennings. Convergence to equilibria of plurality voting. In *The Twenty-Fourth National Conference on Artificial Intelligence*, pages 823–828, Atlanta, Georgia, July 2010.

[13] S. Obraztsova and E. Elkind. Optimal manipulation of voting rules. In *Proceedings of the 26th National Conference on Artificial Intelligence (AAAI)*, pages 2141–2147, Toronto, Canada, July 2012.

[14] S. Obraztsova, O. Lev, V. Markakis, Z. Rabinovich, and J. S. Rosenschein. Beyond plurality: Truth-bias in binary scoring rules. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory (ADT)*, pages 451–468, Lexington, Kentucky, September 2015.

[15] S. Obraztsova, O. Lev, M. Polukarov, Z. Rabinovich, and J. S. Rosenschein. Farsighted voting dynamics. In *The First IJCAI Workshop on Algorithmic Game Theory, at the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, Buenos Aires, July 2015.

[16] S. Obraztsova, E. Markakis, M. Polukarov, Z. Rabinovich, and N. R. Jennings. On the convergence of iterative voting: How restrictive should restricted dynamics be? In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 993–999, 2015.

[17] Z. Rabinovich, S. Obraztsova, O. Lev, E. Markakis, and J. S. Rosenschein. Analysis of equilibria in iterative voting schemes. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, pages 1007–1013, Austin, Texas, January 2015.

[18] A. Reijngoud and U. Endriss. Voter response to iterated poll information. In *Proceedings of the 11th International Con- ference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, pages 635–644, Valencia, June 2012.

[19] R. Reyhani and M. C. Wilson. Best reply dynamics for scoring rules. In *The 20th European Conference on Artificial Intelligence (ECAI 2012)*, pages 138–144, Montpellier, France, August 2012.

[20] M. A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[21] D. R. M. Thompson, O. Lev, K. Leyton-Brown, and J. S. Rosenschein. Empirical aspects of plurality election equilibria. In *Proceedings of the 12th International Coference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 391–398, St.

Paul, Minnesota, May 2013.

[22] L. Xia, M. Zuckerman, A. D. Procaccia, V. Conitzer, and J. S. Rosenschein. Complexity of unweighted manipulation under some common voting rules. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pages 348–353, Pasadena, California, July 2009.

Preference elicitation in matching markets via interviews: A study of offline benchmarks

Baharak Rastegari
School of Computing Science
University of Glasgow
Glasgow, UK
baharak.rastegari@glasgow.ac.uk

Paul Goldberg
Dept. of Computer Science
University of Oxford
Oxford, UK
paul.goldberg@cs.ox.ac.uk

David Manlove
School of Computing Science
University of Glasgow
Glasgow, UK
david.manlove@glasgow.ac.uk

ABSTRACT

The *stable marriage problem* and its extensions have been extensively studied, with much of the work in the literature assuming that agents fully know their own preferences over alternatives. This assumption however is not always practical (especially in large markets) and agents usually need to go through some costly deliberation process in order to learn their preferences. In this paper we assume that such deliberations are carried out via *interviews*, where an interview involves a man and a woman, each of whom learns information about the other as a consequence. If everybody interviews everyone else, then clearly agents can fully learn their preferences. But interviews are costly, and we may wish to minimize their use. It is often the case, especially in practical settings, that due to correlation between agents' preferences, it is unnecessary for all potential interviews to be carried out in order to obtain a stable matching. Thus the problem is to find a good strategy for interviews to be carried out in order to minimize their use, whilst leading to a stable matching. One way to evaluate the performance of an interview strategy is to compare it against a naïve algorithm that conducts all interviews. We argue however that a more meaningful comparison would be against an optimal offline algorithm that has access to agents' preference orderings under complete information. We show that, unless P=NP, no offline algorithm can compute the optimal interview strategy in polynomial time. If we are additionally aiming for a particular stable matching (perhaps one with certain desirable properties), we provide restricted settings under which efficient optimal offline algorithms exist.

Categories and Subject Descriptors

Theory of Computation [**Algorithmic game theory and mechanism design**]: Algorithmic game theory

General Terms

Algorithms, Economics, Theory

Keywords

Two-sided matching, preferences, interviews

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

1. INTRODUCTION

Two-sided matching markets model many practical settings, such as corporate hiring and university admission [21, 16]. The classical *stable marriage problem* is perhaps the most widely studied matching problem in this class, where participants are partitioned into two disjoint sets – men and women – and each participant on one side of the market wishes to be matched to a candidate from the other side of the market, and has preferences over potential matches. A matching is called stable if no pair of participants would prefer to leave their assigned partners to pair with each other. Gale and Shapley's seminal paper [4] proposed a polynomial-time algorithm for finding a stable matching. The books by Knuth [13], Gusfield and Irving [6], Roth and Sotomayor [21], and Manlove [16] provide excellent introductions and surveys.

A key assumption in much of this literature is that all market participants know their full preference orderings. The classical Gale-Shapley (GS) algorithm [4] and its variants require participants' preferences as input. This assumption is reasonable in some settings. However, as markets grow large (e.g., in the hospitals-residents matching market [20, 8] or college admission market [4, 22]) it quickly becomes impractical for participants to assess their precise preference rankings. Instead, participants usually start out with some partial knowledge about their preferences and need to perform some deliberation in order to learn their precise preference ordering. In this paper we assume that deliberations are carried out via *interviews*, where an interview is a unit operation that involves one agent from each side of the market and is informative to both participants. For example, in the hospitals-residents problem (which models the entry-level labor market in which graduating medical students, or residents, are seeking to be assigned to hospital posts), hospitals are likely to be able to identify their "top-tier" residents, "second-tier" residents and so on, and in order to rank the residents in each tier they need to interview them. An interview between a hospital h and a resident r yields information about the qualities of each party to the other. Thus we initially assume that each agent's preference list is in general expressed in terms of a partial order, and after an agent has interviewed ℓ members of the opposite side of the market, he/she has discovered enough information to rank those elements in strict order.

In order to be able to use the GS algorithm to find a stable matching in this setting, a naïve solution is for each participant to conduct all potential interviews and fully learn their preferences. Interviews however are usually costly both

in terms of time, mental energy, and money, therefore we wish to minimize their usage. Indeed, the naïve approach may impose unnecessary deliberation. For example, in the hospitals-residents problem, one expects some degree of correlation across hospitals in the assessment of the most desirable residents, and likewise residents are expected to have correlated views (at least to some extent) on the desirability of hospitals. Therefore, it is expected that more desirable residents get matched to more desirable hospitals and so on. It is then not hard to see that it is wasted effort if a top-tier resident is to interview low-tier hospitals, or a second-tier resident is to interview top-tier hospitals. For a concrete example, consider a setting with four residents and four hospitals where each hospital can admit (at most) one resident. Assume that residents r_1 and r_2 are top-tier residents and r_3 and r_4 are second-tier residents. Likewise assume that h_1 and h_2 are top-tier hospitals and h_3 and h_4 are second-tier hospitals. The preference lists of all agents are correlated according to these hierarchies, although each agent's individual strict ranking (initially unknown) within these hierarchies may differ. It is not hard to verify that no matter what the true (initially unknown) preference orderings of the participants are, under a stable matching r_1 and r_2 each gets matched to either h_1 or h_2, and r_3 and r_4 each gets matched to either h_3 or h_4. Thus an interview between r_1 and either h_3 or h_4 is unnecessary, for example.

Unfortunately, we cannot always avoid unnecessary interviews. For example consider a setting with two residents and two hospitals, where initially agents have no information on their preference orderings and hence cannot compare the two alternatives. W.l.o.g. assume that h_1 interviews both residents and learns that it prefers r_1 to r_2. If r_1 additionally interviews h_2 and learns that he prefers h_1 to h_2, then a stable matching μ is found after 3 interviews, in which h_i is matched to r_i $(1 \leq i \leq 2)$. Now imagine that r_1 instead learns that he prefers h_2 to h_1. It is easy to verify that the identity of a stable matching is not yet revealed and hence more interviews are required. The only remaining interview is between h_2 and r_2 after which one can definitely identify a stable matching. If h_2 learns that it prefers r_2 to r_1 then μ is a stable matching. Assume that r_2 also prefers h_2 to h_1. In this case the interview between h_1 and r_1 is unnecessary as the three other interviews would have provided enough information – that h_2 and r_2 are each others' top choice – for μ to be identified as a stable matching. However, a priori we can not always rule the interview between h_1 and r_1 as unnecessary; for example when h_2's top choice is r_2 but r_2's top choice is h_1.

Any interviewing strategy leads to refinements of the partial orders contained in the original problem instance that represented uncertainty over the true preferences. A key aim could be to carry out sufficient interviews so as to arrive at an instance that admits a super-stable matching μ. Super-stability will be defined formally in the next section, but informally it ensures that μ will be stable regardless of how the remaining uncertainty is resolved. The original instance need not admit a super-stable matching (see [9] for an example) but we are guaranteed that a super-stable matching is always achievable (e.g., by conducting all possible interviews, we will arrive at a strictly-ordered instance, where super-stability and classical stability become equivalent, and the existence of a stable matching is assured [4]).

Thus our aim is to find a good strategy that conducts as few interviews as possible so as to obtain a refined instance that admits a super-stable matching. In general any such strategy will be an online algorithm, since the next interview to be carried out might depend on the results of previous ones.

This leads to the question of how to evaluate the performance of any given interview strategy. One could compare it against the naïve algorithm described above that conducts all interviews. We argue however, by analogy with online algorithms and their competitive ratio, that it makes more sense to compare it against an optimal "offline" algorithm. Here, the optimal offline algorithm has access to agents' preference orderings under full information and has to compute the optimal (i.e., minimum) number of interviews required in order to reach an information state under which it can identify a super-stable matching. In this paper we show that unless P=NP, no offline algorithm can compute an optimal interview strategy in polynomial time.

Some stable matchings have desirable properties, and we may be interested in refining the preferences further so as to obtain such matchings. For example, in the *man-optimal stable matching*, each man has the best partner that he could obtain in any stable matching, whilst the *woman-optimal stable matching* has a similar optimality property for the women. As described above, after a certain number of interviews we may reach an instance that admits a super-stable matching μ. But by carrying out more interviews, some men, for example, may end up with better partners than they had in μ. This would be the case if μ is not the man-optimal stable matching in the instance with the strict (true underlying) preferences.

If we wish to evaluate the performance of an online algorithm that aims for potential improvements in men's partners even after a super-stable matching has been identified, then a suitable offline benchmark for the competitive ratio would be the minimum number of interviews required to refine the original instance so as to make a *specified* matching super-stable. In this paper we show that, whilst this problem is NP-hard in general, there are restricted cases that are solvable in polynomial time.

Related work.

Until very recently, the problem of incremental preference elicitation has received little attention. Several works in the past few years however have addressed this problem from different angles [12, 14, 1, 3, 19, 7, 2]. Those closest in spirit to ours are [1, 19, 2].

In [19] the authors introduced a stable matching model in which participants start out with incomplete information about their preferences, in the form of partially ordered sets, and are able to refine their knowledge by performing interviews. They investigated the problem of minimizing the number of interviews required to find a matching that is stable w.r.t. the true underlying strict preference ordering and additionally is optimal for one side of the market. They presented several results among which are the following two: (i) finding a minimum certificate, that is a set of partial preferences that supports an optimal (for one side of the market) stable matching is NP-hard, and (ii) in a setting where participants on one side of the market have the same partially ordered preferences, an optimal interview policy can be found in polynomial time.

In [1], the authors studied a setting where deliberation is

in the form of pairwise comparison queries (that is, a query leads to strict order of preference being determined over two acceptable agents for a given agent). They proposed a method for finding approximately stable matchings, using minimax regret as a measure, while keeping the number of required comparisons relatively low. In [2] the authors combined the comparison query model of [1] with the interview model of [19] and introduced a unified model where both types of elicitation can take place. They provided an efficient (polynomial-time) scheme for generating queries and interviews, and examined the effectiveness of their scheme via empirical evaluation including comparison against the polynomial-time algorithm of [19] for the restricted setting in which participants on one side of the market have the same partially ordered preferences.

Our work is also related to the body of literature studying variants of stability defined in settings where agents' preferences may include ties. As discussed above, super-stable matchings are relevant in the context of incomplete preference information, because they are stable no matter which refinements represent the true (strict) preferences. Polynomial time algorithms have been proposed for finding a super-stable matching, or reporting that none exists, in various two-sided matching markets [9, 15, 10, 18].

In the next section we provide definitions of notation and terminology, leading to formal statements of the problems under consideration in this paper. A roadmap of the remaining sections is then given at the end of Section 2.

2. PRELIMINARY DEFINITIONS AND RESULTS

SMPI, SMTI, *and levels of stability.*

In an instance of the *Stable Marriage problem with Partially ordered preferences and Incomplete lists (*SMPI*)*, there are two sets of agents, namely a set of men $M = \{m_1, m_2, \ldots, m_n\}$, and a set of women $W = \{w_1, w_2, \ldots, w_n\}$. We assume without loss of generality that $|M| = |W|$ (we can easily reduce the case where the two sets are of different sizes to our setting). Let $[i]$ denote the set $\{1, 2, \ldots, i\}$. We use the term *agents* when making statements that apply to both men and women, and the term *candidates* to refer to agents on the opposite side of the market to that of an agent under consideration. Each agent a finds a subset of candidates acceptable – we refer to these as a's *acceptable* candidates. An agent a's preferences over his/her acceptable candidates need not be strict. That is, given two candidates, a might not be able to compare them against each other. We denote by p_{m_i} and p_{w_j} the partial orders that represent the preferences of m_i and w_j, respectively. We let $p_{M,W} = (p_{m_1}, \ldots, p_{m_n}, p_{w_1}, \ldots, p_{w_n})$ and call $p_{M,W}$ a *partial preference ordering profile.*

Let $I = (M, W, p_{M,W})$ be an instance of SMPI, let a be an agent and let c_1 and c_2 be two acceptable candidates for a in I. We say that *a strictly prefers* c_1 to c_2 if $(c_1, c_2) \in p_a$, and we say that *a cannot compare* c_1 and c_2 (or that a finds c_1 and c_2 *incomparable*) if $(c_1, c_2) \notin p_a$ and $(c_2, c_1) \notin p_a$. We sometimes use the graph-theoretic representation of p_a where candidates in p_a correspond to vertices and there is an arc from a candidate c_i to a candidate c_j if and only if $(c_i, c_j) \in p_a$.

An instance $I' = (M, W, p'_{M,W})$ of SMPI is a *refinement* of I if for each agent a, any strict total order that is a linear extension of p'_a is also a linear extension of p_a. We may also refer to $p'_{M,W}$ being a refinement of $p_{M,W}$ (or indeed I) using the same definition. Also we can define p'_a being a refinement of p_a for some specific agent a similarly.

Observation 1. *Given two instances I and I' of* SMPI*, I' is a refinement of I if and only if the following condition holds: for each agent a and every two candidates c_1 and c_2 acceptable to a, if $(c_1, c_2) \in p_a$ then $(c_1, c_2) \in p'_a$.*

A well studied special case of SMPI is the *Stable Marriage problem with Ties and Incomplete lists (*SMTI*)*, in which incomparability is transitive and is interpreted as indifference. In SMTI, each agent has a partition of acceptable candidates into *indifference classes* or *ties* such that he or she is indifferent between the candidates in the same indifference class, but has a strict preference ordering over the indifference classes. In an instance of SMTI, let C_t^a denote the t-th indifference class of agent a, where $t \in [n]$. We assume that $C_t^a = \emptyset$ implies $C_{t'}^a = \emptyset$ for all $t' > t$. The *Stable Marriage problem with Incomplete lists (*SMI*)* is the special case of SMTI in which each tie is of size one. Similarly the *Stable Marriage problem with Ties (*SMT*)* is the special case of SMTI in which each man finds each woman acceptable and vice versa.

Given an instance I of SMPI, a *matching* μ is a pairing of men and women such that each man is paired with at most one woman and vice versa, and no agent is matched to an unacceptable partner. If m and w are matched in μ then $\mu(m) = w$ and $\mu(w) = m$. We say that $\mu(a) = \varnothing$ if a is unmatched under μ. Different levels of stability can be defined in the context of SMPI [9, 15]. A *strong blocking pair* is an acceptable (man,woman) pair, each of whom is unmatched or strictly prefers the other to his/her partner. A *weakly stable matching* is a matching with no strong blocking pair. Every instance of SMPI admits a weakly stable matching [17]. An acceptable (man,woman) pair is a *weak blocking pair* if each member of the pair is either unmatched or strictly prefers the other to his/her partner or cannot compare the other with his/her partner, and one member of the pair is either unmatched or strictly prefers the other to his/her partner. A *strongly stable matching* is a matching with no weak blocking pair. Finally a *very weak blocking pair* is an acceptable (man,woman) pair, each of whom is unmatched or strictly prefers to other to his/her partner or cannot compare the other with his/her partner. A *super-stable matching* is a matching with no very weak blocking pair. It can be easily verified that a matching is super-stable if and only if it is weakly stable w.r.t. all strict total orders that are linear extensions of the given partial preference orderings [16, Lemma 3.2.4]. In instances of SMI, weak stability, strong stability and super-stability are all equivalent to classical stability.

Interviews to refine the partial orders.

In a given instance $I = (M, W, p_{M,W})$ of SMPI in this paper, we assume that $p_{M,W}$, the partial preference ordering profile, represents the agents' initial information state. That is, agents may not have enough information initially in order to rank their acceptable candidates in strict order. However in the problem instances that we will later define in this section, we will assume that each agent a has a strict preference ordering $\succ_a$ over his or her acceptable candidates. This rep-

resents the true (and strict) underlying preferences over a's acceptable candidates, although crucially, a may not (and in general will not) initially be aware of the entire ordering. We let $\succ_{M,W} = (\succ_{m_1}, \ldots, \succ_{m_n}, \succ_{w_1}, \ldots, \succ_{w_n})$ and call $\succ_{M,W}$ a *strict (true underlying) preference ordering profile*. The task of the agents is to learn enough information about their acceptable candidates in order to refine their preferences, in a manner consistent with $\succ_{M,W}$, to obtain an SMPI instance I' that admits a super-stable matching μ (thus μ will be stable with respect to $\succ_{M,W}$).

Following the model introduced in [19], we assume that instances can be refined through *interviews*. Each interview pairs one man m with one woman w. An interview is informative to both parties involved. Hence saying "m interviews w" is equivalent to saying "w interviews m". When agent a interviews ℓ candidates, this results in a new refined SMPI instance which is exactly the same as I except that a now has a strict preference ordering over the ℓ interviewed candidates.

Notice that if an agent interviews only one candidate, no refinement takes place. Note also that not all refinements of I can be reached by a set of interviews. For example, suppose that in I we have one man m_1 and three women w_1, w_2, and w_3. Suppose m_1 finds the three women acceptable and incomparable. Assume that in I' man m_1 prefers w_1 to both w_2 and w_3, and cannot compare w_2 and w_3. It is easy to see that I' is a refinement of I, but no set of interviews can reach I': for m_1 to learn that he prefers w_1 to the other two women he must interview all three women, but then he will have a strict preference ordering over the three of them.

We say that an SMPI instance I' is an *interview-compatible refinement* of an SMPI instance I if I' can be refined from I using interviews. We now show that interview-compatible refinements can be recognized easily.

PROPOSITION 2. *Let I and I' be two instances of* SMPI. *We can determine in $O(n^3)$ time whether I' is an interview-compatible refinement of I.*

PROOF. To verify whether I' is a refinement of I, it is sufficient to check whether the condition of Observation 1 holds. With a suitable data structure, we can do this in $O(n^3)$ time. For each agent a identify the edges present in p'_a that are not in p_a, and let $S(a)$ be the set of candidates in p'_a that form an endpoint of at least one such edge. For I' to be an interview-compatible refinement of I, it is necessary and sufficient that, for every a, $S(a)$ forms a complete subgraph in the undirected graph corresponding to p'_a. This can be tested in $O(n^3)$ time overall. □

Let I' be an SMPI instance that is an interview-compatible refinement of a given SMPI instance I. We define the *cost* of I' given I to be the minimum number of interviews required to refine I into I'. The following proposition shows how to compute this cost efficiently.

PROPOSITION 3. *Let I be an* SMPI *instance and let I' be an interview-compatible refinement of I. We can determine in $O(n^3)$ time the cost of I' given I.*

PROOF. We identify the set of interviews T that refines I into I' as follows. Initially $T = \emptyset$. For each agent a and every two candidates c_1 and c_2, if a cannot compare c_1 and c_2 under I, but prefers one to the other under I', a must have interviewed both c_1 and c_2. Add both of these interviews to T. Notice that we might have already accounted for one or both of these interviews. However since T is a set, no interview is going to be included in T more than once. With a suitable data structure, the aforementioned procedure terminates in $O(n^3)$ time overall, and once it does, $|T|$ denotes the cost of I'. □

$$\begin{array}{llll} r_1: & h_1\ h_2 & h_1: & r_2\ r_1 \\ r_2: & h_2\ h_1 & h_2: & r_2\ r_1 \end{array}$$

Figure 1: A stable marriage problem instance.

Problem definitions.

The motivating problem is as follows: given an instance $I = (M, W, p_{M,W})$ of SMPI, find an interview-compatible refinement I' of minimum cost such that I' admits a super-stable matching. Since the result of one interview might influence which interview/s to carry out next, any strategy for carrying out interviews should be regarded as an online algorithm.

In fact there may be no online algorithm that can guarantee to provide an optimal solution in all cases. To see this, let us return to the example of Section 1 involving two residents and two hospitals, and suppose that initially everyone finds the two agents on the other side of the market acceptable and incomparable. Let the true underlying preferences be given by Figure 1 (here, preference lists are ordered from left to right in decreasing order of preference). Any online interviewing strategy must start with one interview in the absence of any knowledge; w.l.o.g. suppose that the first interview involves r_1 and h_1. Then it may be verified that the algorithm is bound to use 3 more interviews before a super-stable matching can be found. (If one interview does not take place then two agents on opposite sides of the market cannot compare the two candidates in their preference list; it then follows that each of the two possible matchings would be blocked according to super-stability.) On the other hand the interview involving r_1 and h_1 was unnecessary and an optimal strategy uses only 3 interviews.

Towards computing bounds for the competitive ratio of an online algorithm, the offline scenario is of interest, and that is what we consider in what follows. In the offline case, the mechanism designer is given $\succ_{M,W}$, the strict (true underlying) preference ordering profile of the agents, and would like to compute an optimal interviewing schedule, i.e., an interview-compatible refinement I' of I, such that $\succ_{M,W}$ refines I'. This is reflected in the definition of the following problem, named MIN-ICR, which is an abbreviation for "Minimum-cost Interview Compatible Refinement problem".

DEFINITION 4. *An instance of* MIN-ICR *comprises a tuple $(I, \succ_{M,W})$, where I is an instance of* SMPI *and $\succ_{M,W}$ is a strict preference ordering profile that refines I. The problem is to find an interview-compatible refinement I' of I such that (i) $\succ_{M,W}$ refines I', (ii) I' admits a super-stable matching, and (iii) I' is of minimum cost amongst interview-compatible refinements that satisfy (i) and (ii).*

The decision version of MIN-ICR is defined as follows.

DEFINITION 5. *An instance of* ICR-DEC *comprises a tuple $(I, \succ_{M,W}, K)$, where I is an instance of* SMPI, *$\succ_{M,W}$*

is a strict preference ordering profile that refines I, and K is a non-negative integer. The problem is to decide whether there exists an interview-compatible refinement I' of I, with cost at most K, such that $\succ_{M,W}$ refines I' and I' admits a super-stable matching.

As discussed in Section 1, it is sometimes the case that we aim for a particular matching, stable under $\succ_{M,W}$, that has some desirable properties, for example the woman-optimal stable matching. The offline problem can then be viewed as a restricted variant of MIN-ICR where, in addition to I and $\succ_{M,W}$, we are also equipped with a matching μ. This is reflected in the definition of the following problem, named MIN-ICR-EXACT, which is an abbreviation for "Minimum-cost Interview Compatible Refinement problem with Exact matching".

DEFINITION 6. *An instance of* MIN-ICR-EXACT *comprises a tuple $(I, \succ_{M,W}, \mu)$, where I is an instance of* SMPI*, $\succ_{M,W}$ is a strict preference ordering profile that refines I, and μ is a matching that is weakly stable w.r.t. $\succ_{M,W}$. The problem is to find an interview-compatible refinement I' of I, such that (i) $\succ_{M,W}$ refines I', (ii) μ is super-stable in I', and (iii) I' is of minimum cost amongst interview-compatible refinements of I that satisfy (i) and (ii).*

The decision version of MIN-ICR-EXACT, called ICR-EXACT-DEC, is then defined analogously to the way that ICR-DEC was obtained from the definition of MIN-ICR.

The remainder of this paper is organized as follows. In Section 3 we first show that ICR-DEC is NP-complete even under quite restricted settings. The proof is by reduction from Vertex Cover. We also leverage the same proof to show that ICR-EXACT-DEC is also NP-complete. Then in Section 4 we provide a reverse reduction, from Vertex Cover to ICR-EXACT-DEC, and utilize it to show that MIN-ICR-EXACT is polynomial-time solvable for several restricted settings. Some concluding remarks are presented in Section 5.

3. NP-COMPLETENESS RESULTS

We show that ICR-DEC is NP-complete even if I is an instance of SMTI in which each indifference class is of size at most 3. Further, we prove that ICR-DEC is NP-complete even for SMT instances, and even if all men are indifferent between all women. We first provide a lemma that will come in handy in proving our claims; the proof is straightforward, and is omitted.

LEMMA 7. *Let $G = (V, E)$ be an undirected graph where for each vertex v, $deg(v) \leq 3$. We can direct the edges in E such that for each v, $deg^+(v) \leq 2$ and $deg^-(v) \leq 2$.*

Unlike many problems that are NP-complete, the membership of ICR-DEC in the class NP is not trivial. Hence, we provide a proof via the following lemma.

LEMMA 8. ICR-DEC *is in NP.*

PROOF. To prove this, it is sufficient to show that given SMPI instances I and I', a strict preference profile $\succ_{M,W}$ and an integer K, we can decide in polynomial time whether (i) I' is an interview-compatible refinement of I, (ii) I' has cost at most K, (iii) $\succ_{M,W}$ refines I', and (iv) I' admits a super-stable matching.

Both (i) and (ii) are established by Propositions 2 and 3 respectively. For (iii), it is straightforward to check in polynomial time whether $\succ_{M,W}$ refines I'.

Finally to establish (iv), we can use the polynomial time algorithm of [18], SUPER-SMP, to decide whether I' admits a super-stable matching or not. □

We show that ICR-DEC is NP-hard by reducing from the decision version of the Vertex Cover problem (VC). VC is defined as follows: given a graph $G = (V, E)$ and an integer K, decide whether G admits a vertex cover of size at most K. VC is NP-complete even if each vertex has degree at most 3 [5]; let VC-3 denote this restriction. We denote by MIN-VC the optimization version of VC, that is the problem of finding a minimum vertex cover in a given graph G.

THEOREM 9. ICR-DEC *is NP-complete even for* SMTI *instances in which each indifference class has size at most* 3.

PROOF. By Lemma 8, ICR-DEC belongs to NP. To show NP-hardness, we reduce from VC-3. Let the undirected simple graph $G = (V, E)$ be given such that $deg(v) \leq 3$, $\forall v \in V$. Let $V = \{v_1, \ldots, v_n\}$. Let $G' = (V, E')$ be a digraph where (i) $\forall (v_i, v_j) \in E'$, $(v_i, v_j) \in E$, (ii) $\forall (v_i, v_j) \in E$, either $(v_i, v_j) \in E'$ or $(v_j, v_i) \in E'$ (but not both), and (iii) $\forall v \in V$, $deg^+(v) \leq 2$ and $deg^-(v) \leq 2$. Note that by Lemma 7 such a graph G' exists. We create an instance $I = (M, W, p_{M,W})$ of SMTI as follows:

- For each vertex $v_i \in V$ there is a man $m_i \in M$ and a woman $w_i \in W$. That is $M = \{m_i | \forall v_i \in V\}$ and $W = \{w_i | \forall v_i \in V\}$.
- Each man m_i finds acceptable w_i and all women w_j such that $(v_i, v_j) \in E'$. Moreover, each man m_i is indifferent between all his acceptable women.
- Each woman w_i finds acceptable m_i and all men m_j such that $(v_j, v_i) \in E'$. Moreover, each woman w_i is indifferent between all her acceptable men.

Note that the total length of the men's preference lists is equal to $|E| + |V|$. Also note that as $deg^+(v) \leq 2$ and $deg^-(v) \leq 2$ for all $v \in V$, hence the indifference classes are of size at most 3. Let $\succ_{M,W}$ be a strict preference ordering under which m_i and w_i rank each other at the top of their preference lists. We prove that G has a vertex cover of size at most K if and only if there exists a refinement I' of I, of cost at most $K' = K + |E|$, such that $\succ_{M,W}$ refines I' and I' admits a super-stable matching. Notice that $\succ_{M,W}$ admits only one stable matching, that being $\mu = \{(m_i, w_i) | \forall i\}$. This implies that I' must admit exactly one super-stable matching, that being μ.

Proof of the only-if direction: Assume that G has a vertex cover C of size $k \leq K$. We show that there is a refinement I' of cost $k' = k + |E| \leq K'$ such that $\succ_{M,W}$ refines I' and μ is super-stable in I'. We create I' as follows. For each vertex $v_i \in C$ refine I such that both m_i and w_i now have strict preferences as in $\succ_{M,W}$ This refinement can of course be achieved by having both m_i and w_i interview all candidates in their lists; this includes an interview between m_i and w_i. Notice that since interviews are informative to both parties involved, partial refinements in the preference orderings of those persons whose corresponding vertices are not in C must have taken place as well. For example, consider a case in which $v_i, v_j \in C$, $v_k \notin C$, and $(v_i, v_k), (v_j, v_k) \in E'$.

Then both m_i and m_j interview w_k and hence w_k must now rank m_i and m_j in strict order of preferences as in $\succ_{M,W}$. An interview is either between same indexed agents, e.g., between m_i and w_i, or between agents of different indices, e.g., between m_i and w_j where $i \neq j$. We refer to an interview of the former type as a *same-index* interview and an interview of the latter type as a *different-index* interview. The total number of interviews performed by all agents is going to be k same-index interview plus a number of different-index interviews. The number of different-index interviews under any refinement can be at most $|E|$, and under our proposed refinement is exactly $|E|$ (since C is a vertex cover). Therefore the total number of interviews is exactly $k + |E|$. It remains to show that μ is a super-stable matching in I'. We call a $(man, woman)$ pair a *fixed pair* if they are matched in every stable matching of every strict order refinement of I'. We show that (m_i, w_i) is a fixed pair for all $i \leq n$, hence proving that μ is the only stable matching in every strict order refinement of I' and therefore definitely a super-stable matching in I'. Take any pair (m_i, w_i) such that $v_i \in C$. By our construction, m_i and w_i rank each other at top, so clearly (m_i, w_i) is a fixed pair. Now take any pair (m_j, w_j) such that $v_j \notin C$. Since v_j is not in the vertex cover, therefore it must be the case that all neighbors of v_j are in C. Thus, for any v_k that is a neighbor of v_j, it has been already established that (m_k, w_k) is a fixed pair. Therefore (m_j, w_j) is also a fixed pair. Moreover, neither m_j nor w_j can form a very weak blocking pair with a person they are not matched to.

Proof of the if direction: Assume that I has a refinement I' of cost $k' \leq K'$ such that $\succ_{M,W}$ refines I' and μ is super-stable in I'. We show that G admits a vertex cover of size at most $k' - |E| \leq K$. We first show that in order to arrive at I', every agent should have interviewed every candidate s/he finds acceptable and to whom s/he is not matched. Assume for a contradiction that this is not the case. Take a pair (m_i, w_j), acceptable to each other, who have not interviewed. Therefore, under I', m_i is indifferent between w_j and w_i (to whom he is matched in μ), and w_j is indifferent between m_i and m_j (to whom she is matched in μ). Hence (m_i, w_j) constitutes a very weak blocking pair in μ under I', a contradiction. We have established so far that every agent must have interviewed acceptable candidates to whom s/he is not matched, which means that each agent has interviewed all candidates in his/her list who have a different index from him/her. This amounts to the total of $|E|$ interviews. The only remaining interviews for which we have not yet accounted are those corresponding to matched pairs. Let C be a set of vertices such that vertex v_i is in C if and only if m_i and w_i have interviewed under I'. Notice that $|C| = k' - |E|$. Take any $v_j \notin C$. We show that all neighbors of v_j are in C, establishing that C is a vertex cover. Since $v_j \notin C$, it follows from the construction of C that m_j and w_j have not interviewed under I'. Assume for a contradiction that v_j has a neighbor, say v_k, who too is not in C. Therefore m_k and w_k have not interviewed under I' either. W.l.o.g. assume that $(v_j, v_k) \in E'$. (A similar argument applies if $(v_k, v_j) \in E'$.) Therefore m_j and w_k are acceptable to each other. Furthermore, since neither m_j nor w_k have interviewed their partners in μ, it is the case that m_j is indifferent between w_j (his partner in μ) and w_k, and w_k is indifferent between m_k (her partner in μ) and m_j. Therefore (m_j, w_k) constitutes a very weak blocking pair in μ under I', a contradiction. □

We next show that ICR-DEC is also NP-complete under a different restricted setting by making small alterations to the proof of Theorem 9.

COROLLARY 10. ICR-DEC *is NP-complete even for* SMT *instances and even if agents on one side of the market are indifferent between all the candidates.*

PROOF. W.l.o.g. assume that all men are indifferent between all women. Modify the reduction presented in the proof of Theorem 9 as follows.

- For each vertex $v_i \in V$ there is a man m_i in M and a woman w_i in W. That is $M = \{m_i | \forall v_i \in V\}$ and $W = \{w_i | \forall v_i \in V\}$.
- Each man m_i finds all women acceptable and is indifferent between them.
- Each woman w_i finds all men acceptable and has two indifference classes. In the top indifference class are m_i and all men m_j such that $(v_i, v_j) \in E$. In the second indifference class are all other men.

Note that the total length of the women's first indifference classes is equal to $2|E|+|V|$. Let $\succ_{M,W}$ be a strict preference ordering under which m_i and w_i rank each other at the top of their preference lists. We prove that G has a vertex cover of size at most K if and only if there exists a refinement I' of I, of cost at most $K' = K + 2|E|$, such that $\succ_{M,W}$ refines I' and I' admits a super-stable matching. Notice that $\succ_{M,W}$ admits only one stable matching, that being $\mu = \{(m_i, w_i) | \forall i\}$. This implies that I' must admit exactly one super-stable matching, that being μ. Modify the proof of Theorem 9 as follows.

In the only-if direction: For each vertex $v_i \in C$ refine I such that m_i has a strict preference ordering, as in $\succ_{m_i}$, over women in $\{w_i\} \cup \{w_j | (v_i, v_j) \in E\}$ and w_i has a strict preference ordering, as in $\succ_{w_i}$, over men in $\{m_i\} \cup \{m_j | (v_i, v_j) \in E\}$. Consequently, for all v_j adjacent to v_i, m_i prefers w_i to w_j and likewise w_i prefers m_i to m_j. This refinement can be achieved by having m_i interview w_i and all w_j such that $(v_i, v_j) \in E$, and additionally having w_i interview all m_j such that $(v_i, v_j) \in E$. The number of different-index interviews under any refinement can be at most $2|E|$, and under our proposed refinement is exactly $2|E|$ (since C is a vertex cover). So the total number of interviews is exactly $k + 2|E|$. It remains to show that μ is a super-stable matching in I'. Assume for a contradiction that there exists a very weak blocking pair (m_i, w_j).

- If $(v_i, v_j) \in E$, then v_i or v_j is in C. If $v_i \in C$ then m_i and w_i have interviewed and therefore $w_i \succ_{m_i} w_j$. If $v_j \in C$ then m_j and w_j have interviewed and therefore $m_j \succ_{w_j} m_i$. Both cases imply that (m_i, w_j) is not a very weak blocking pair, a contradiction.
- If $(v_i, v_j) \notin E$ then $m_j \succ_{w_j} m_i$, therefore (m_i, w_j) is not a very weak blocking pair, a contradiction.

In the if direction: We show that in order to arrive at I', every man m_i should have interviewed all women w_j such that $(v_i, v_j) \in E$, and likewise every woman w_i should have interviewed all men m_j such that $(v_i, v_j) \in E$. The proof is

similar to that presented in the proof of Theorem 9. Hence we can conclude that at least $2|E|$ different-index interviews must have taken place in the refinement. The rest of the proof is similar to that presented in the proof of Theorem 9, with the difference that $|C| \leq k' - 2|E|$. □

In the proof of Theorem 9, μ is the unique stable matching under $\succ_{M,W}$. Therefore, it follows from the proofs of Theorem 9 and Corollary 10 that ICR-Exact-Dec is also NP-complete for the restrictions stated in those results.

Corollary 11. ICR-Exact-Dec *is NP-complete even for* SMTI *instances, and even when each indifference class is of size at most* 3. ICR-Exact-Dec *is also NP-complete even for* SMTI *instances and even if agents on one side of the market are indifferent between all the candidates.*

We remark that Theorem 4.4 of [19] implies that ICR-Exact-Dec is NP-complete and, likewise, Corollary 11 implies Theorem 4.4 of [19]. However, Corollary 11 is stronger as it is stated for a more restricted setting.

4. POLYNOMIAL-TIME SOLVABLE VARIANTS

4.1 Preliminaries

In this section we explore the tractability of Min-ICR-Exact under various restricted settings. Recall that we have reduced from VC to ICR-Dec and ICR-Exact-Dec in order to show that these problems are NP-hard. Here we present a reverse reduction, from ICR-Exact-Dec to VC, that will come in handy in proving our claims.

Let an instance $(I, \succ_{M,W}, \mu)$ of ICR-Exact-Dec be given. As μ is weakly stable w.r.t. I, it admits no strong blocking pair. If μ is not super-stable w.r.t. I, then μ must admit some very weak blocking pairs. We refer to such blocking pairs as *potential blocking pairs*. We distinguish between potential blocking pairs by the degree of choice one has when attempting to resolve them.

Definition 12 (Potential Blocking Pair (PBP)). *Given an* ICR-Exact-Dec *instance $(I, \succ_{M,W}, \mu)$, a pair (m, w) is a* potential blocking pair (PBP) *if (m, w) is a very weak blocking pair under I. Each PBP (m, w) belongs to either of the following classes.*

- Potential Blocking Pair of Degree 1 (PBP-D1) *if either m or w strictly prefers the other to his or her current partner under $\succ_{M,W}$.*
- Potential Blocking Pair of Degree 2 (PBP-D2) *if both m and w strictly prefer their partners to each other under $\succ_{M,W}$.*

Let I' be an interview-compatible refinement of I. We say that a given potential blocking pair of I, (m, w), is *resolved* under I' if $(\mu(m), w) \in p'_m$ or $(\mu(w), m) \in p'_w$.

If (m, w) is a PBP-D2, then it must be that m and w cannot compare each other and their current partners under I. Thus in order to resolve (m, w) it is sufficient, and necessary, that m or w learn his/her true preference ordering over his/her partner and the other side.

Let (m, w) be a PBP-D1 and assume that m strictly prefers w to $\mu(m)$ (the argument is similar if $m \succ_w \mu(w)$). Therefore, w must find m and $\mu(w)$ incomparable under I, or (m, w) either blocks μ or is not a PBP, and $\mu(w) \succ_w m$, or (m, w) blocks μ. Furthermore, in order to resolve this PBP w has to learn that she prefers $\mu(w)$ to m.

In what follows we use PBP, PBP_1, and PBP_2 to refer to the set of potential blocking pairs, and those of degree 1 and degree 2 respectively.

Proposition 13. *Let $(I, \succ_{M,W}, \mu)$ be an instance of* ICR-Exact-Dec *and I' be an interview-compatible refinement of I. Then μ is super-stable under I' if and only if all PBPs in I are resolved under I'.*

It is easy to see that for a potential blocking pair (m, w) to be resolved, at least one of m or w needs to interview both the other side and his or her current partner and conclude that s/he prefers his or her current partner to the other side. The next proposition then immediately follows.

Proposition 14. *Let $(I, \succ_{M,W}, \mu)$ be an instance of* ICR-Exact-Dec *and I' be an interview-compatible refinement of I. Then μ is super-stable under I' only if, for all $(m, w) \in PBP$, m and w have interviewed under I'.*

For each agent $a \in M \cup W$ let $PBP_1(a)$ denote the set of candidates c such that either (a, c) or (c, a) is in PBP_1 and $a \succ_c \mu(c)$.

Lemma 15. *Let $(I, \succ_{M,W}, \mu)$ be an instance of* ICR-Exact-Dec *and I' be an interview-compatible refinement of I. Then μ is super-stable under I' only if a has interviewed $\mu(a)$ under I' for all agents a where $PBP_1(a) \neq \emptyset$.*

Proof. Assume for a contradiction that there exists an agent a where $PBP_1(a) \neq \emptyset$ and a has not interviewed $\mu(a)$. Therefore for every $c \in PBP_1(a)$ it is still the case that a cannot compare c and $\mu(a)$, and c prefers a to $\mu(c)$. Hence there exists at least one unresolved PBP under I'. □

4.2 Reduction from ICR-Exact-Dec to VC

Let $(I, \succ_{M,W}, \mu)$ be an instance of ICR-Exact-Dec. Let $M' = \{m | PBP_1(m) \neq \emptyset \vee PBP_1(\mu(m)) \neq \emptyset\}$. Let $G(I, \mu) = (V, E)$ be an undirected graph whose vertices V correspond to matched pairs $(m, \mu(m))$. Let $PBP'_2 = \{(m, w) | (m, w) \in PBP_2, m \notin M', \mu(w) \notin M'\}$. Let there be an edge between any two vertices $(m, \mu(m))$ and $(m', \mu(m'))$ if $(m, \mu(m')) \in PBP'_2$ or $(m', \mu(m)) \in PBP'_2$. Remove any vertex with degree zero. Note that for any remaining vertex $(m, \mu(m))$ it is the case that $m \notin M'$.

Theorem 16. *$G(I, \mu)$ has a vertex cover of size at most K if and only if there exists a refinement I' of I, of cost at most $K' = |PBP| + |M'| + K$, such that $\succ_{m,w}$ refines I' and μ is super-stable in I'.*

Proof. Assume that $G(I, \mu)$ has a vertex cover C of size at most K. Let I' be a refinement of I under which the following interviews have taken place.

1. Each pair $(m, w) \in PBP$ interview each other – a total of $|PBP|$ interviews.
2. Each $m \in M'$ interviews his partner $\mu(m)$ – a total of $|M'|$ interviews.
3. Each pair $(m, \mu(m)) \in C$ interview each other – a total of K interviews.

The total number of interviews is then equal to $|PBP| + |M'| + K$. As a result of the above interviews, each agent a learns his or her strict preference ordering over the interviewed candidates, as in $\succ_a$. (Recall that the interviews are informative to both sides.) It is then easy to see that all PBP-D1's are resolved. It is also straightforward to see that for a PBP-D2 (m, w), if either $m \in M'$ or $\mu(w) \in M'$, then (m, w) is resolved under I'. It remains to show that the remaining PBP-D2's, that is those in PBP'_2, are resolved as well. Let (m, w) be such a PBP-D2. By the construction of $G(I, \mu)$, V includes $(m, \mu(m))$ and $(\mu(w), w)$ and there is an edge between these two vertices. As C is a vertex cover, at least one of $(m, \mu(m))$ or $(\mu(w), w)$ belongs to C. If $(m, \mu(m)) \in C$ then, following the results of the interviews, m prefers $\mu(m)$ to w under I'. (A similar argument holds for w if $(\mu(w), w) \in C$.) Therefore (m, w) is resolved under I'.

Conversely, assume that I admits an interview-compatible refinement I' of size at most K' such that μ is super-stable in I'. We show that $G(I, \mu)$ admits a vertex cover of size at most $K' - (|PBP| + |M'|)$. Let C be a set of vertices $(m, \mu(m))$ in V where m and $\mu(m)$ have interviewed under I'. Note that as we have removed all vertices of degree zero from $G(I, \mu)$, hence all remaining vertices are adjacent to at least one edge corresponding to a member of PBP'_2. We show that C is a vertex cover and then prove an upper bound on the size of C.

C is a vertex cover: Let $((m, \mu(m)), (m', \mu(m')))$ be any edge in E. By the construction of $G(I, \mu)$, $(m, \mu(m'))$ or $(m', \mu(m))$ is in PBP'_2. Assume that $(m, \mu(m')) \in PBP'_2$. (The argument for the case where $(m', \mu(m)) \in PBP'_2$ is similar.) As $(m, \mu(m'))$ is resolved under I', either m prefers his partner to $\mu(m')$ under I', or $\mu(m')$ prefers her partner to m under I'. If the former, then m must have interviewed $\mu(m)$ and hence $(m, \mu(m)) \in C$, and if the latter then $\mu(m')$ must have interviewed m' and thus $(m', \mu(m')) \in C$. Thus C is a vertex cover.

C is of size at most $K' - (|PBP| + |M'|)$: We prove this by computing a lower bound on the number of interviews that do not correspond to a vertex in C. It follows Proposition 14 that all PBPs must have interviewed, hence a total of $|PBP|$ interviews. It also follows Lemma 15 that each agent a with $PBP_1(a) \neq \emptyset$ must have interviewed his/her partner. Looking at this from men's perspective, all men m must interview $\mu(m)$ if $PBP_1(m) \neq \emptyset$ or $PBP_1(\mu(m)) \neq \emptyset$ – hence a total of $|M'|$ interviews. Recall that $(m, \mu(m)) \notin V$ if $m \in M'$. Therefore none of the interviews we have accounted for so far, a total of $|PBP| + |M'|$ interviews, correspond to a vertex in C. □

Theorem 16 essentially tells us that an instance $(I, \succ_{M,W}, \mu)$ of Min-ICR-Exact is polynomial-time solvable if Min-VC is polynomial-time solvable in $G(I, \mu)$. Equipped with this knowledge, we provide three different restricted settings under which ICR-Exact-Dec, and hence Min-ICR-Exact, is solvable in polynomial time.

Theorem 17. Min-ICR-Exact *is solvable in polynomial time if one side has fully known strict preference ordering.*

Proof. Assume that women have strict preferences and the target matching is μ. Note that all PBPs must be of degree 1. Therefore $G(I, \mu)$ is an empty graph with vertex cover of size zero. It follows from Proposition 14 and Lemma 15 that Min-ICR has a solution of size $|PBP| + |M'|$. □

Theorem 18. Min-ICR-Exact *is solvable in polynomial time under the restriction of* SMTI *in which indifference classes are of size at most 2.*

Proof. We show that $G(I, \mu)$ is a collection of cycles and paths, and hence its minimum vertex cover can be computed in polynomial time. The size of a minimum vertex cover for any path or cycle of length ℓ is $\lceil \frac{\ell}{2} \rceil$.

Take any vertex $v_1 = (m, \mu(m))$ in V. Recall that if any vertex $v_2 = (m', \mu(m'))$ is a neighbor of v_1, then it must be that at least one of $(m, \mu(m'))$ or $(m', \mu(m))$ is in PBP'_2. Note that if $(m, \mu(m')) \in PBP'_2$, then under I man m is indifferent between $\mu(m)$ and $\mu(m')$. Since each indifference class is of size at most 2, at most one such neighbor exists. Likewise, if $(m', \mu(m)) \in PBP'_2$ then $\mu(m)$ is indifferent between m and m'. However, since each indifference class is of size at most 2, at most one such neighbor exist. Thus, each vertex has degree at most 2, hence $G(I, \mu)$ is a collection of cycles and paths. □

Theorem 19. Min-ICR-Exact *is solvable in polynomial time under the restriction of* SMTI *in which all men are endowed with the same indifference classes, as well as all women. That is $C_i^m = C_i^{m'}$ for all $m, m' \in M$ and all $i \in [n]$, and $C_i^w = C_i^{w'}$ for all $w, w' \in W$ and all $i \in [n]$.*

Proof. We show that $G(I, \mu)$ is a collection of complete graphs, and hence its minimum vertex cover can be computed in polynomial time, since the size of a minimum vertex cover for any complete graph K_ℓ is equal to $\ell - 1$. To prove that $G(I, \mu)$ is a collection of complete graphs, we show that for any three given vertices v_1, v_2 and v_3, if $(v_1, v_2) \in E$ and $(v_1, v_3) \in E$ then $(v_2, v_3) \in E$.

Take any three vertices $v_1 = (m, \mu(m))$, $v_2 = (m', \mu(m'))$, and $v_3 = (m'', \mu(m''))$. If $(v_1, v_2) \in E$ then, under I, all men are indifferent between $\mu(m)$ and $\mu(m')$, all women are indifferent between m and m', and $m, m' \notin M'$. If $(v_1, v_3) \in E$ then, under I, all men are indifferent between $\mu(m)$ and $\mu(m'')$, all women are indifferent between m and m'', and $m'' \notin M'$. Therefore, since I is an instance of SMTI, all men are indifferent between $\mu(m)$, $\mu(m')$ and $\mu(m'')$, and all women are indifferent between m, m', and m''. Hence $(m', \mu(m''))$ and $(m'', \mu(m'))$ are PBPs. If $(m', \mu(m''))$ is a PBP-D2 then, as $m', m'' \notin M'$, $(m', \mu(m'')) \in PBP'_2$ and therefore $(v_2, v_3) \in E$. Assume for a contradiction that $(m', \mu(m''))$ is a PBP-D1. Assume that $\mu(m'') \succ_{m'} \mu(m')$ (the argument is similar if $m' \succ_{\mu(m'')} m''$), implying that $PBP_1(\mu(m'')) \neq \emptyset$ and thus $m'' \in M'$, a contradiction. □

Theorem 18 is likely to be of more theoretical interest. For the setting of Theorem 17, we could envisage a hospitals-residents matching problem where residents are ranked uniformly (i.e., in a "master list" common to all hospitals [11]) according to some known objective value (e.g., which may be based on academic merit, as in the UK) and residents must use interviews in order to determine their true preferences over acceptable hospitals. For the setting of Theorem 19, consider a market with "tiered" preferences, where everybody agrees who/what belongs to each tier (again the membership of these tiers could relate to some objective values), but the precise ordering within these tiers could be subjective, and up to individuals to determine themselves. For

example, students may use national league tables for determining top tier universities, second tier universities and so on, but students' precise ranking over the universities in any given tier may vary.

If I is of one of the restricted forms for which MIN-ICR-EXACT is polynomial time solvable, then one straightforward approach to solving MIN-ICR is to enumerate all matchings that are stable under $\succ_{M,W}$ and then solve MIN-ICR-EXACT for each of them. This approach is practical if $\succ_{M,W}$ admits a polynomial number of stable matchings.

5. CONCLUSION AND FUTURE WORK

In this paper we have studied the complexity of the offline problem relating to computing an optimal interview strategy for a stable marriage market where initially participants have incomplete information, and the aim is to refine the instance using the minimum number of interviews in order to arrive at a super-stable matching. The main direction for future work is to investigate the online case, where the true underlying preferences are not known to the mechanism designer, with respect to measures such as the competitive ratio. Furthermore, an important question for which we do not know an answer yet is whether MIN-ICR is polynomial-time solvable under some restricted setting. Extending the known results on interviewing in stable marriage markets to many-to-one markets such as college admission is another important future direction. It is also interesting to study online algorithms in a setting where elicitation is taking place via comparison queries. In this paper we assume that the objective of the mechanism designer is to minimize the total number of interviews overall. One may however argue that such a strategy may require one or may agents to conduct most of the interviews while the others do none or very little. In the view of fairness and the practicality of such central interview-scheduling schemes, it is also of utmost importance to study settings in which a fair distribution of the interviews is also considered.

Acknowledgments. We thank Rob Irving and Piotr Krysta for the useful discussions and their valuable feedback. We also thank the anonymous reviewers for their insightful comments and suggestions. This work was supported by EPSRC grants EP/K01000X/1 and EP/K010042/1, and COST Action IC1205 on Computational Social Choice.

REFERENCES

[1] J. Drummond and C. Boutilier. Elicitation and approximately stable matching with partial preferences. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 97–105. AAAI Press, 2013.

[2] J. Drummond and C. Boutilier. Preference elicitation and interview minimization in stable matchings. In *Proceedings of the Twenty-Eight AAAI Conference on Artificial Intelligence*, AAAI '14, pages 645–653. AAAI Press, 2014.

[3] F. Echenique, S. Lee, M. Shum, and M. B. Yenmez. The revealed preference theory of stable and extremal stable matchings. *Econometrica*, 81(1):153–171, 2013.

[4] D. Gale and L. Shapley. College admissions and the stability of marriage. *American Mathematics Monthly*, 69:9–15, 1962.

[5] M. R. Garey and D. S. Johnson. The rectilinear steiner tree problem is NP-complete. *SIAM Journal on Applied Mathematics*, 32(4):826–834, 1977.

[6] D. Gusfield and R. W. Irving. *The stable marriage problem: structure and algorithms*. MIT Press, Cambridge, MA, USA, 1989.

[7] G. Haeringer and V. Iehle. Two-sided matching with one-sided information. *Working paper*, 2014.

[8] R. Irving and D. Manlove. Finding large stable matchings. *ACM Journal of Experimental Algorithmics*, 14(2), 2009.

[9] R. W. Irving. Stable marriage and indifference. *Discrete Applied Mathematics*, 48(3):261–272, 1994.

[10] R. W. Irving, D. F. Manlove, and S. Scott. The hospitals/residents problem with ties. In *SWAT*, volume 1851 of *Lecture Notes in Computer Science*, pages 259–271, 2000.

[11] R. W. Irving, D. F. Manlove, and S. Scott. The stable marriage problem with master preference lists. *Discrete Applied Mathematics*, 156(15):2959–2977, 2008.

[12] S. Kalyanaraman and C. Umans. The complexity of rationalizing matchings. In *Proceedings of ISAAC '08: the 19th International Symposium on Algorithms and Computation*, volume 5369 of *Lecture Notes in Computer Science*, pages 171–182. Springer, 2008.

[13] D. Knuth. *Mariages Stables*. Montréal: Les Presses de l'Université de Montréal, 1976.

[14] R. Lee and M. Schwarz. Interviewing in two-sided matching markets. *Working Paper*, 2012.

[15] D. Manlove. Stable marriage with ties and unacceptable partners. Technical Report TR-1999-29, University of Glasgow, Department of Computing Science, 1999.

[16] D. F. Manlove. *Algorithmics of matching under preferences*. World Scientific, 2013.

[17] D. F. Manlove, R. W. Irving, K. Iwama, S. Miyazaki, and Y. Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261 – 279, 2002.

[18] B. Rastegari, A. Condon, N. Immorlica, R. Irving, and K. Leyton-Brown. Reasoning about optimal stable matching under partial information. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 431–448. ACM, 2014.

[19] B. Rastegari, A. Condon, N. Immorlica, and K. Leyton-Brown. Two-sided matching with partial information. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 733–750. ACM, 2013.

[20] A. Roth. The national residency matching program as a labor market. *Journal of the American Medical Association*, 275(13):1054–1056, 1996.

[21] A. Roth and M. Sotomayor. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Cambridge University Press, 1990.

[22] A. E. Roth. The college admissions problem is not equivalent to the marriage problem. *Journal of Economic Theory*, 36:277–288, August 1985.

Social Welfare in One-Sided Matching Mechanisms

George Christodoulou
University of Liverpool
Ashton Street, Liverpool, U.K.
gchristo@liv.ac.uk

Aris Filos-Ratsikas
University of Oxford
Parks Road, Oxford, UK
aris.filos-ratsikas@cs.ox.ac.uk

Søren Kristoffer Stiil Frederiksen
Aarhus University
Abogade 34, Aarhus, DK
sorensf@gmail.com

Paul W. Goldberg
University of Oxford
Parks Road, Oxford, UK
paul.goldberg@cs.ox.ac.uk

Jie Zhang
University of Oxford
Parks Road, Oxford, UK
jie.zhang@cs.ox.ac.uk

Jinshan Zhang
University of Oxford
Ashton Street, Liverpool, UK
jinshan.zhang@liv.ac.uk

ABSTRACT

We study the Price of Anarchy of mechanisms for the well-known problem of one-sided matching, or house allocation, with respect to the social welfare objective. We consider both ordinal mechanisms, where agents submit preference lists over the items, and cardinal mechanisms, where agents may submit numerical values for the items being allocated. We present a general lower bound of $\Omega(\sqrt{n})$ on the Price of Anarchy, which applies to *all* mechanisms. We show that two well-known mechanisms, Probabilistic Serial, and Random Priority, achieve a matching upper bound. We extend our lower bound to the Price of Stability of a large class of mechanisms that satisfy a common proportionality property, and show stronger bounds on the Price of Anarchy of all *deterministic* mechanisms.

Categories and Subject Descriptors

I.12.11 [**Distributed Artificial Intelligence**]: Multiagent Systems; J.4 [**Computer Applications**]: Social and Behavioral Sciences - Economics

General Terms

Economics, Theory

Keywords

One-sided matching, probabilistic serial, truthfulness, price of anarchy, Nash equilibrium

1. INTRODUCTION

One-sided matching (also called the house allocation problem) is the fundamental problem of assigning items to agents, such that each agent receives exactly one item. It has numerous applications, such as assigning workers to shifts, students to courses or patients to doctor appointments. In this setting, agents are often asked to provide *ordinal preferences*, i.e. preference lists, or rankings of the items. We assume that underlying these ordinal preferences, agents have numerical values specifying how much they value each item [18]. In game-theoretic terms, these are the agents' von Neumann-Morgenstern utility functions [27] and the associated preferences are often referred to as *cardinal preferences*.

A *mechanism* is a function that maps agents' valuations to matchings. However, agents are rational strategic entities that might not always report their valuations truthfully; they may misreport their values if that results in a better matching (from their own perspective). Assuming the agents report their valuations strategically to maximize their utilities, it is of interest to study the *Nash equilibria* of the induced game, i.e. strategy profiles from which no agent wishes to unilaterally deviate.

A natural objective for the designer is to choose the matching that maximizes the *social welfare*, i.e. the sum of agents' valuations for the items they are matched with, which is the most prominent measure of aggregate utility in the literature. Given the strategic nature of the agents, we are interested in mechanisms that maximize the social welfare *in the equilibrium*. We use the standard measure of equilibrium inefficiency, the *Price of Anarchy* [22], that compares the maximum social welfare attainable in any matching with the *worst-case* social welfare that can be achieved in an equilibrium.

We evaluate the efficiency of a mechanism with respect to the Price of Anarchy of the induced game. We study both deterministic and randomized mechanisms: in the latter case the output is a probability mixture over matchings, instead of a single matching. We are interested in the class of *cardinal* mechanisms, which use cardinal preferences, and generalize the ordinal mechanisms.

Note that our setting involves no monetary transfers and generally falls under the umbrella of *approximate mechanism design without money* [24]. In general settings without money, one has to fix a canonical representation of the valuations. A common approach in the literature is to consider the *unit-sum* normalization, i.e. each agent has a total value of 1 for all the items. We obtain results for unit-sum valuations, and extend most of these to another common normalization, *unit-range*.

1.1 Our results

In Section 3 we bound the inefficiency of the two best-known mechanisms in the matching literature, *Probabilistic*

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), John Thangarajah, Karl Tuyls, Stacy Marsella, Catholijn Jonker (eds.), May 9–13, 2016, Singapore.*

Serial and *Random Priority*. In particular, for n agents and n items, the Price of Anarchy is $O(\sqrt{n})$. In Section 4 we complement this with a *matching* lower bound (i.e. $\Omega(\sqrt{n})$) that applies to *all* cardinal (randomized) mechanisms. As a result, we conclude that these two *ordinal* mechanisms (ones that compute matchings that only depend on preference orderings) are optimal. These results suggest that it does not help a welfare maximizer to ask agents to report more than the ordinal preferences.

We separately consider *deterministic* mechanisms and in Section 4 prove that their Price of Anarchy is $\Omega(n^2)$, even for cardinal mechanisms. This shows that randomization is necessary for non-trivial worst-case efficiency guarantees.

In Section 5, we extend our results to more general solutions concepts as well as the case of incomplete information. Finally, in Section 6, we prove that under a mild "proportionality" property, our lower bound of $\Omega(\sqrt{n})$ extends to the *Price of Stability*, a more optimistic measure of efficiency [3], which strengthens the negative results even further. Additionally, we discuss how our results extend to the other common normalization in the literature, *unit-range* [2, 15, 28].

1.2 Discussion and related work

The one-sided matching problem was introduced in [18] and has been studied extensively ever since (see [1] for a recent overview). Over the years, several different mechanisms have been proposed with various desirable properties related to truthfulness, fairness and economic efficiency with Probabilistic Serial [9, 7, 8, 1] and Random Priority [1, 9, 4, 23, 15, 2] being the two prominent examples.

As mentioned earlier, in settings without money, one needs to represent the valuations in some canonical way. A common approach is the *unit-sum* normalization, i.e. each agent has a total value of 1 for all the items. Intuitively, this normalization means that each agent has equal influence within the mechanism and her values can be interpreted as "scrip money" that she uses to acquire items. The unit-sum representation is standard for social welfare maximization in many settings without money including fair division, cake cutting and resource allocation [10, 11, 16, 15] among others. Moreover, without any normalization, non-trivial Price of Anarchy bounds cannot be achieved by any mechanism.

The objective of social welfare maximization for one-sided matching problems has been studied before in the literature, but mainly for truthful mechanisms [2, 15]. Our lower bounds are more general, since they apply to *all* mechanisms, not just truthful ones. In particular, our lower bound on the Price of Anarchy of all mechanisms generalizes the corresponding bound for truthful mechanisms in [15]. Note that Random Priority is truthful (truth-telling is a dominant strategy equilibrium) but it has other equilibria as well; we observe that the welfare guarantees of the mechanism hold for all equilibria, not just the truthtelling ones. Similar approaches have been made for truthful mechanisms like the second price auction in settings with money.

While given our general lower bound, proving a matching upper bound for Random Priority is enough to establish tightness, it is still important to know what the welfare guarantees of Probabilistic Serial are, given that it is arguably the most popular one-sided matching mechanism. The mechanism was introduced by [9] and since then, it has been in the center of attention of the matching literature, with related work on characterizations [17, 20], extensions [19], strategic aspects [21] and hardness of manipulation [6]. Somewhat surprisingly, the Nash equilibria of the mechanism were only recently studied. Aziz et al. [5] prove that the mechanism has pure Nash equilibria while Ekici and Kesten [14] study the *ordinal* equilibria of the mechanism and prove that the desirable properties of the mechanism are not necessarily satisfied for those profiles.

Another, somewhat different recent branch of study considers ordinal measures of efficiency instead of social welfare maximization, under the assumption that agents' preferences are only expressed through preference orderings over items. Bhalgat et al. [8] study the approximation ratio of matching mechanisms, when the objective is maximization of *ordinal social welfare*, a notion of efficiency that they define based solely on ordinal information. Other measures of efficiency for one-sided matchings were also studied in Krysta et al. [23], where the authors design truthful mechanisms to approximate the size of a maximum cardinally (or maximum agent weight) Pareto-optimal matching and in Chakrabarty and Swamy [12] where the authors consider the rank approximation as the measure of efficiency. While interesting, these measures of efficiency do not accurately encapsulate the socially desired outcome the way that social welfare does, especially since an underlying cardinal utility structure is part of the setting [9, 18, 27, 28]. Our results actually suggest that in order to achieve the optimal welfare guarantees, one does not even need to elicit this utility structure; agents can only be asked to report preference orderings, which is arguably more appealing.

Finally, we point out that our work is in a sense analogous to the literature that studies the Price of Anarchy in item-bidding auctions (e.g. see [13, 26] and references therein) for settings without money. Furthermore, the extension of our results to very general solution concepts (coarse correlated equilibria) and settings of incomplete information (Bayes-Nash equilibria) is somehow reminiscent of the *smoothness* framework [25] for games. While our results are not proven using the smoothness condition, our extension technique is similar in spirit.

2. PRELIMINARIES

Let $N = \{1, \ldots, n\}$ be a finite set of agents and $A = \{1, \ldots, n\}$ be a finite set of indivisible items. An *allocation* is a matching of agents to items, that is, an assignment of items to agents where each agent gets assigned exactly one item. We can view an allocation μ as a permutation vector $(\mu_1, \mu_2 \ldots, \mu_n)$ where μ_i is the unique item matched with agent i. Let O be the set of all allocations. Each agent i has a valuation function $u_i : A \to \mathbb{R}$ mapping items to real numbers. Valuation functions are considered to be well-defined modulo positive affine transformations, that is, for item $j : j \to \alpha u_i(j) + \beta$ is considered to be an alternative representation of the same valuation function u_i. Given this, we fix the canonical representation of u_i to be *unit-sum*, that is $\sum_j u_i(j) = 1$, with $u_i(j) \geq 0$ for all i, j. Equivalently, we can consider valuation functions as *valuation vectors* $u_i = (u_{i1}, u_{i2}, \ldots, u_{in})$ and let V be the set of all valuation vectors of an agent. Let $\mathbf{u} = (u_1, u_2, \ldots, u_n)$ denote a typical *valuation profile* and let V^n be the set of all valuation profiles with n agents.

We consider *strategic agents* who might have incentives to misreport their valuations. We define $\mathbf{s} = (s_1, s_2, \ldots, s_n)$ to

be a pure strategy profile, where s_i is the *reported* valuation vector of agent i. We will use $\mathbf{s_{-i}}$ to denote the strategy profile without the ith coordinate and hence $\mathbf{s} = (s_i, \mathbf{s_{-i}})$ is an alternative way to denote a strategy profile. A *direct revelation mechanism* without money is a function $M : V^n \to O$ mapping *reported* valuation profiles to matchings. For a randomized mechanism, we define M to be a random map $M : V^n \to O$. Let $M_i(\mathbf{s})$ denote the restriction of the outcome of the mechanism to the i'th coordinate, which is the item assigned to agent i by the mechanism. For randomized mechanisms, we let $p_{ij}^{M,\mathbf{s}} = \Pr[M_i(\mathbf{s}) = j]$ and $p_i^{M,\mathbf{s}} = (p_{i1}^{M,\mathbf{s}}, \ldots, p_{in}^{M,\mathbf{s}})$. When it is clear from the context, we drop one or both of the superscripts from the terms $p_{ij}^{M,\mathbf{s}}$. The utility of an agent from the outcome of a deterministic mechanism M on input strategy profile $\mathbf{s}$ is simply $u_i(M_i(\mathbf{s}))$. For randomized mechanisms, an agent's utility is $\mathbb{E}[u_i(M_i(\mathbf{s}))] = \sum_{j=1}^{n} p_{ij}^{M,\mathbf{s}} u_{ij}$.

A subclass of mechanisms that are of particular interest is that of *ordinal mechanisms*. Informally, ordinal mechanisms operate solely based on the *ordering* of items induced by the valuation functions and not the actual numerical values themselves, while cardinal mechanisms take those numerical values into account. Formally, a mechanism M is *ordinal* if for any strategy profiles $\mathbf{s}, \mathbf{s}'$ such that for all agents i and for all items j, ℓ, $s_{ij} < s_{i\ell} \Leftrightarrow s'_{ij} < s'_{i\ell}$, it holds that $M(\mathbf{s}) = M(\mathbf{s}')$. A mechanism for which the above does not necessarily hold is *cardinal*. Equivalently, the strategy space of ordinal mechanisms is the set of all permutations of n items instead of the space of valuation functions V^n. A strategy s_i of agent i is a *preference ordering* of items $(a_1, a_2, \ldots, a_n)$ where $a_\ell \succ a_k$ for $\ell < k$. We will write $j \succ_i j'$ to denote that agent i prefers item j to item j' according to her true valuation function and $j \succ_{s_i} j'$ to denote that she prefers item j to item j' according to her strategy s_i. When it is clear from the context, we abuse the notation slightly and let u_i denote the truthtelling strategy of agent i, even when the mechanism is ordinal. Note that agents can be indifferent between items and hence the preference order can be a weak ordering.

Two properties of interest are *anonymity* and *neutrality*. A mechanism is anonymous if the output is invariant under renamings of the agents and neutral if the output is invariant under relabeling of the objects.

An *equilibrium* is a strategy profile in which no agent has an incentive to deviate to a different strategy. First, we will focus on the concept of *pure Nash equilibrium*, formally

DEFINITION 1. *A strategy profile* $\mathbf{s}$ *is a* pure Nash equilibrium *if* $u_i(M_i(\mathbf{s})) \geq u_i(M_i(s'_i, s_{-i}))$ *for all agents* i*, and pure deviating strategies* s'_i*.*

In Section 5, we extend our results to more general equilibrium notions as well as the setting of incomplete information, where agents' values are drawn from known distributions. Let $S_{\mathbf{u}}^M$ denote the set of all pure Nash equilibria of mechanism M under truthful valuation profile $\mathbf{u}$. The measure of efficiency that we will use is the *pure Price of Anarchy*,

$$PoA(M) = \sup_{\mathbf{u} \in V^n} \frac{SW_{OPT}(\mathbf{u})}{\min_{\mathbf{s} \in S_{\mathbf{u}}^M} SW_M(\mathbf{u}, \mathbf{s})}$$

where $SW_M(\mathbf{u}, \mathbf{s}) = \sum_{i=1}^{n} \mathbb{E}[u_i(M_i(\mathbf{s}))]$ is the expected *social welfare* of mechanism M on strategy profile $\mathbf{s}$ under true valuation profile $\mathbf{u}$, and $SW_{OPT}(\mathbf{u}) = \max_{\mu \in O} \sum_{i=1}^{n} u_i(\mu_i)$ is the social welfare of the optimal matching. Let $OPT(\mathbf{u})$ be the optimal matching on profile $\mathbf{u}$ and let $OPT_i(\mathbf{u})$ be the restriction to the ith coordinate.

3. PRICE OF ANARCHY GUARANTEES

In this section, we prove the (pure) Price of Anarchy guarantees of Probabilistic Serial and Random Priority. Together with our lower bound in the next section, the results establish that both mechanisms are optimal among all mechanisms for the problem.

Probabilistic Serial

First, we consider *Probabilistic Serial*, which we abbreviate to PS. Informally, the mechanism is the following. Each item can be viewed as an infinitely divisible good that all agents can consume at unit speed during the unit time interval $[0, 1]$. Initially each agent consumes her most preferred item (or one of her most preferred items in case of ties) until the item is entirely consumed. Then, the agent moves on to consume the item on top of her preference list, among items that have not yet been entirely consumed. The mechanism terminates when all items have been entirely consumed. The fraction p_{ij} of item j consumed by agent i is then interpreted as the probability that agent i will be matched with item j under the mechanism.

We prove that the Price of Anarchy of PS is $O(\sqrt{n})$. Aziz et al. [5] proved that PS has pure Nash equilibria, so it makes sense to consider the pure Price of Anarchy; we will extend the result to the coarse correlated Price of Anarchy and the Bayesian Price of Anarchy in Section 5.

We start with the following two lemmas, which prove that in a pure Nash equilibrium of the mechanism an agent's utility cannot be much worse than what her utility would be if she were consuming the item she is matched with in the optimal allocation from the beginning of the mechanism until the item is entirely consumed. Let $t_j(\mathbf{s})$ be the time when item j is entirely consumed on profile $\mathbf{s}$ under $PS(\mathbf{s})$.

LEMMA 1. *Let* $\mathbf{s}$ *be any strategy profile and let* s_i^* *be any strategy such that* $j \succ_{s_i^*} \ell$ *for all items* $\ell \neq j$*, i.e. agent* i *places item* j *on top of her preference list. Then it holds that* $t_j(s_i^*, \mathbf{s_{-i}}) \geq \frac{1}{4} \cdot t_j(\mathbf{s})$*.*

PROOF. For ease of notation, let $\mathbf{s}^* = (s_i^*, \mathbf{s_{-i}})$. Obviously, if $j \succ_{s_i} \ell$ for all $\ell \neq j$ and since all other agents' reports are fixed, $t_j(\mathbf{s}^*) = t_j(\mathbf{s})$ and the statement of the lemma holds. Hence, we will assume that there exists some item $j' \neq j$ such that $j' \succ_{s_i} j$.

First, note that if agent i is the only one consuming item j for the duration of the mechanism, then $t_j(\mathbf{s}^*) = 1$ and we are done. Hence, assume that at least one other agent consumes item j at some point, and let τ be the time when the first agent besides agent i starts consuming item j in $\mathbf{s}^*$. Obviously, $t_j(\mathbf{s}^*) > \tau$, therefore if $\tau \geq \frac{1}{4} \cdot t_j(\mathbf{s})$ then $t_j(\mathbf{s}^*) \geq \frac{1}{4} \cdot t_j(\mathbf{s})$ and we are done. So assume that $\tau < \frac{1}{4} \cdot t_j(\mathbf{s})$. Next observe that in the interval $[\tau, t_j(\mathbf{s}^*)]$, agent i can consume at most half of what remains of item i because there exists at least one other agent consuming the item for the same duration. Overall, agent i's consumption is at most $\frac{1}{2} + \frac{1}{4} t_j(\mathbf{s})$ so at least $\frac{1}{2} - \frac{1}{4} t_j(\mathbf{s})$ of the item will be consumed by the rest of the agents.

Now consider all agents other than i in profile $\mathbf{s}$ and let α be the the amount of item j that they have consumed by time $t_j(\mathbf{s})$. Notice that the total consumption speed of

an item is non-decreasing in time which means in particular that for any $0 \leq \beta \leq 1$, agents other than i need at least $\beta t_j(\mathbf{s})$ time to consume $\alpha \cdot \beta$ in profile $\mathbf{s}$. Next, notice that since agent i starts consuming item j at time 0 in $\mathbf{s}^*$ and all other agents use the same strategies in $\mathbf{s}$ and $\mathbf{s}^*$, it holds that every agent $k \neq i$ starts consuming item j in $\mathbf{s}^*$ no sooner than she does in $\mathbf{s}$. This means that in profile $\mathbf{s}^*$, agents other than i will need more time to consume $\beta \cdot \alpha$; in particular they will need at least $\beta t_j(\mathbf{s})$ time, so $t_j(\mathbf{s}^*) \geq \beta t_j(\mathbf{s})$. However, from the previous paragraph we know that they will consume at least $\frac{1}{2} - \frac{1}{4}t_j(\mathbf{s})$, so letting $\beta = \frac{1}{\alpha}\left(\frac{1}{2} - \frac{1}{4}t_j(\mathbf{s})\right)$ we get

$$t_j(\mathbf{s}^*) \geq \beta t_j(\mathbf{s}) \geq t_j(\mathbf{s})\left(\frac{1}{2} - \frac{1}{4}\cdot t_j(\mathbf{s})\right)\frac{1}{\alpha}$$
$$\geq t_j(\mathbf{s})\left(\frac{1}{2} - \frac{1}{4}\cdot t_j(\mathbf{s})\right) \geq \frac{1}{4}\cdot t_j(\mathbf{s})$$

□

Now we can lower bound the utility of an agent at any pure Nash equilibrium.

LEMMA 2. *Let $\mathbf{u}$ be the profile of true agent valuations and let $\mathbf{s}$ be a pure Nash equilibrium. For any agent i and any item j it holds that the utility of agent i at $\mathbf{s}$ is at least $\frac{1}{4}\cdot t_j(\mathbf{s}) \cdot u_{ij}$.*

PROOF. Let $\mathbf{s}' = (s_i', \mathbf{s_{-i}})$ be the strategy profile obtained from $\mathbf{s}$ when agent i deviates to the strategy s_i' where s_i' is some strategy such that $j \succ_{s_i'} \ell$ for all items $\ell \neq j$. Since $\mathbf{s}$ is a pure Nash equilibrium, it holds that $u_i(PS_i(\mathbf{s})) \geq u_i(PS_i(\mathbf{s}')) \geq t_j(\mathbf{s}') \cdot u_{ij}$, where the last inequality holds since the utility of agent i is at least as much as the utility she obtains from the consumption of item j. By Lemma 1, it holds that $t_j(\mathbf{s}') \geq \frac{1}{4}\cdot t_j(\mathbf{s})$ and hence $u_i(PS_i(\mathbf{s})) \geq \frac{1}{4}\cdot t_j(\mathbf{s}) \cdot u_{ij}$. □

We can now prove the pure Price of Anarchy guarantee of the mechanism.

THEOREM 1. *The pure Price of Anarchy of Probabilistic Serial is $O(\sqrt{n})$.*

PROOF. Let $\mathbf{u}$ be any profile of true agents' valuations and let $\mathbf{s}$ be any pure Nash equilibrium. First, note that by reporting truthfully, every agent i can get an allocation that is at least as good as $(1/n, \ldots, 1/n)$, regardless of other agents' strategies. To see this, first consider time $t = 1/n$ and observe that during the interval $[0, 1/n]$, agent i is consuming her favorite item (say a_1) and hence $p_{ia_1} \geq 1/n$. Next, consider time $\tau = 2/n$ and observe that during the interval $[0, 2/n]$, agent i is consuming one or both of her two favorite items (a_1 and a_2) and hence $p_{ia_1} + p_{ia_2} \geq 2/n$. By a similar argument, for any k, it holds that $\sum_{j=1}^{n} p_{ia_j} \geq k/n$. This implies that regardless of other agents' strategies, agent i can achieve a utility of at least $\frac{1}{n}\sum_{j=1}^{n} u_{ij}$. Since $\mathbf{s}$ is a pure Nash equilibrium, it holds that $u_i(PS_i(\mathbf{s})) \geq (1/n)\sum_{j=1}^{n} u_{ij}$ as well. Summing over all agents, we get that $SW_{PS}(\mathbf{u}, \mathbf{s}) \geq (1/n)\sum_{i=1}^{n}\sum_{j=1}^{n} u_{ij} = 1$. If $SW_{OPT}(\mathbf{u}) \leq \sqrt{n}$ then we are done, so assume $SW_{OPT}(\mathbf{u}) > \sqrt{n}$.

Because PS is neutral we can assume $t_j(\mathbf{s}) \leq t_{j'}(\mathbf{s})$ for $j < j'$ without loss of generality. Observe that for all $j = 1, \ldots, n$, it holds that $t_j(\mathbf{s}) \geq j/n$. This is true because for any $t \in [0, 1]$, by time t, exactly tn mass of items must have been consumed by the agents. Since j is the jth item that is entirely consumed, by time $t_j(\mathbf{s})$, the mass of items that must have been consumed is at least j. By this, we get that $t_j(\mathbf{s}) \cdot n \geq j$, which implies $t_j(\mathbf{s}) \geq j/n$.

For each j let i_j be the agent that gets item j in the optimal allocation and for ease of notation, let w_{i_j} be her valuation for the item. Now by Lemma 2, it holds that

$$u_{i_j}(PS(\mathbf{s})) \geq \frac{1}{4}\frac{j}{n}w_{i_j} \quad \text{and} \quad SW_{PS}(\mathbf{u}, \mathbf{s}) \geq \frac{1}{4}\sum_{j=1}^{n}\frac{j}{n}w_{i_j}.$$

The Price of Anarchy is then at most

$$\frac{4\sum_{j=1}^{n} w_{i_j}}{\sum_{j=1}^{n} j \cdot w_{ij}/n}.$$

Consider the case when the above ratio is maximized and let k be an integer such that $k \leq \sum_{j=1}^{n} w_{i_j} \leq k+1$. Then it must be that $w_{i_j} = 1$ for $j = 1, \ldots, k$ and $w_{i_j} = 0$, for $k+2 \leq i_j \leq n$. Hence the maximum ratio is $(k+w_{i_{k+1}})/(aw_{i_{k+1}}+b)$, for some $a,\ b > 0$, which is monotone for $w_{i_{k+1}}$ in $[0, 1]$. Therefore, the maximum value of $(k + w_{i_{k+1}})/(aw_{i_{k+1}} + b)$ is achieved when either $w_{i_{k+1}} = 0$ or $w_{i_{k+1}} = 1$. As a result, the maximum value of the ratio is obtained when $\sum_{i=1^n} w_{i_{k+1}} = k$ for some k. By simple calculations, the Price of Anarchy should be at most:

$$\frac{4k}{\sum_{j=1}^{k}\frac{j}{n}} \leq \frac{4k}{\frac{k(k-1)}{2n}} = \frac{8n}{k-1},$$

so the Price of Anarchy is maximized when k is minimized. By the argument earlier, $k > \sqrt{n}$ and hence the ratio is $O(\sqrt{n})$. □

In Section 5, we extend Theorem 1 to broader solution concepts and the incomplete information setting.

Random Priority

We also consider another very well-known mechanism, Random Priority, often referred to as Random Serial Dictatorship. The mechanism first fixes an ordering of the agents uniformly at random and then according to that ordering, it sequentially matches them with their most preferred item that is still available. Filos-Ratsikas et al. [15] proved that the welfare in any truthtelling equilibrium is an $\Omega(1/\sqrt{n})$-fraction of the maximum social welfare. While Random Priority has other equilibria as well, to establish the Price of Anarchy bound, it suffices to observe that at least for distinct valuations, any strategy other than truthtelling does not affect the allocation and hence it does not affect the social welfare. Intuitively, since agents pick their most preferred items, any equilibrium strategy would place the most preferred available items on top of the preference list, while the ordering of the items that are not picked does not affect the allocation of other agents. For valuations that are not distinct, the argument can be adapted using small perturbations of the values, losing only a small fraction of welfare.

We first we prove the following lemma.

LEMMA 3. *If valuations are distinct, the social welfare is the same in all mixed Nash equilibria of Random Priority.*

PROOF. Let i be an agent, and let B be a subset of the items. Let $\mathbf{s}$ be a mixed Nash equilibrium with the property that with positive probability, i will be chosen to select an item at a point when B is the set of remaining items. In that

case (by distinctness of i's values), i's strategy should place agent i's favourite item in B on the top of the preference list among items in B. Suppose that for items j and j', there is no set of items B that may be offered to i with positive probability, in which either j or j' is optimal. Then i may rank them either way, i.e. can announce $j \succ_i j'$ or $j' \succ_i j$. However, that choice has no effect on the other agents, in particular it cannot affect their social welfare. □

Given the main theorem in [15], Lemma 3 implies the following.

COROLLARY 1. *If valuations are distinct, the Price of Anarchy of Random Priority is* $\Theta(\sqrt{n})$.

The same guarantee on the Price of Anarchy holds even when the true valuations of agents are not necessarily distinct.

THEOREM 2. *The Price of Anarchy of Random Priority is* $O(\sqrt{n})$.

PROOF. We know from [15] that the social welfare of Random Priority given truthful reports, is within $O(\sqrt{n})$ of the social optimum. The social welfare of a (mixed) Nash equilibrium $\mathbf{q}$ cannot be worse than the worst pure profile from $\mathbf{q}$ that occurs with positive probability, so let $\mathbf{s}$ be such a pure profile. We will say that agent i *misranks* items j and j' if $j \succ_i j'$, but $j' \succ_{s_i} j$.

If an agent misranks two items for which she has distinct values, it is because she has 0 probability in $\mathbf{s}$ to receive either item. So we can change $\mathbf{s}$ so that no items are misranked, without affecting the social welfare or the allocation. For items that the agent values equally (which are then not misranked) we can apply arbitrarily small perturbations to make them distinct. Profile $\mathbf{s}$ is thus consistent with rankings of items according to perturbed values and is truthful with respect to these values, which, being arbitrarily close to the true ones, have optimum social welfare arbitrarily close to the true optimal social welfare. □

Theorem 2 can be extended to solution concepts more general than the mixed Nash equilibrium. Again, the details are included in Section 5.

4. LOWER BOUNDS

In this section, we prove our main lower bound. Note that the result holds for any mechanism, including randomized and cardinal mechanisms. Since we are interested in mechanisms with good properties, it is natural to consider those mechanisms that have pure Nash equilibria.

THEOREM 3. *The pure Price of Anarchy of any mechanism is* $\Omega(\sqrt{n})$.

PROOF. Assume $n = k^2$ for some $k \in \mathbb{N}$. Let M be a mechanism and consider the following valuation profile $\mathbf{u}$. There are $\sqrt{n}$ sets of agents and let G_j denote the j-th set. For every $j \in \{1, \ldots, \sqrt{n}\}$ and every agent $i \in G_j$, it holds that $u_{ij} = 1/n + \alpha$ and $u_{ik} = 1/n - \alpha/(n-1)$, for $k \neq j$, where α is sufficiently small. Let $\mathbf{s}$ be a pure Nash equilibrium and for every set G_j, let $i_j = \arg\min_{i \in G_j} p_{ij}^{M,\mathbf{s}}$ (break ties arbitrarily). Observe that for all $j = 1, \ldots, \sqrt{n}$, it holds that $p_{i_j j}^{M,\mathbf{s}} \leq 1/\sqrt{n}$ and let $I = \{i_1, i_2, \ldots, i_{\sqrt{n}}\}$. Now consider the valuation profile $\mathbf{u}'$ where:

- For every agent $i \notin I$, $u'_i = u_i$.
- For every agent $i_j \in I$, let $u'_{i_j j} = 1$ and $u'_{i_j k} = 0$ for all $k \neq j$.

We claim that $\mathbf{s}$ is a pure Nash equilibrium under $\mathbf{u}'$ as well. For agents not in I, the valuations have not changed and hence they have no incentive to deviate. Assume now for contradiction that some agent $i \in I$ whose most preferred item is item j could deviate to some beneficial strategy s'_i. Since agent i only values item j, this would imply that $p_{ij}^{M,(s'_i,\mathbf{s}_{-\mathbf{i}})} > p_{ij}^{M,\mathbf{s}}$. However, since agent i values all items other than j equally under u_i and her most preferred item is item j, such a deviation would also be beneficial under profile $\mathbf{u}$, contradicting the fact that $\mathbf{s}$ is a pure Nash equilibrium.

Now consider the expected social welfare of M under valuation profile $\mathbf{u}'$ at the pure Nash equilibrium $\mathbf{s}$. For agents not in I and taking α to be less than $1/n^3$, the contribution to the social welfare is at most 1. For agents in I, the contribution to the welfare is then at most $(1/\sqrt{n})\sqrt{n} + 1$ and hence the expected social welfare of M is at most 3. As the optimal social welfare is at least $\sqrt{n}$, the bound follows. □

Interestingly, if we restrict our attention to *deterministic* mechanisms, then we can prove that only trivial pure Price of Anarchy guarantees are achievable.

THEOREM 4. *The pure Price of Anarchy of any deterministic mechanism is* $\Omega(n^2)$.

PROOF. Let M be a deterministic mechanism that always has a pure Nash equilibrium. Let $\mathbf{u}$ be a valuation profile such that for for all agents i and i', it holds that $u_i = u_{i'}$, $u_{i1} = 1/n + 1/n^3$ and $u_{ij} > u_{ik}$ for $j < k$. Let $\mathbf{s}$ be a pure Nash equilibrium for this profile and assume without loss of generality that $M_i(\mathbf{s}) = i$.

Now fix another true valuation profile $\mathbf{u}'$ such that $u'_1 = u_1$ and for agents $i = 2, \ldots, n$, $u'_{i,i-1} = 1 - \epsilon'_{i,i-1}$ and $u_{ij} = \epsilon'_{ij}$ for $j \neq i-1$, where $0 \leq \epsilon'_{ij} \leq 1/n^3$, $\sum_{j \neq i-1} \epsilon'_{ij} = \epsilon'_{i,i-1}$ and $\epsilon'_{ij} > \epsilon'_{ik}$ if $j < k$ when $j, k \neq i-1$. Intuitively, in profile $\mathbf{u}'$, each agent $i \in \{2, \ldots, n\}$ has valuation close to 1 for item $i-1$ and small valuations for all other items. Futhermore, she prefers items with smaller indices, except for item $i-1$.

We claim that $\mathbf{s}$ is a pure Nash equilibrium under true valuation profile $\mathbf{u}$ as well. Assume for contradiction that some agent i has a benefiting deviation, which matches her with an item that she prefers more than i. But then, since the set of items that she prefers more than i in both $\mathbf{u}$ and $\mathbf{u}'$ is $\{1, \ldots, i\}$, the same deviation would match her with a more preferred item under $\mathbf{u}$ as well, contradicting the fact that $\mathbf{s}$ is a pure Nash equilibrium. It holds that $SW_{OPT}(\mathbf{u}') \geq n-2$ whereas the social welfare of M is at most $2/n$ and the theorem follows. □

The mechanism that naively maximizes the sum of the reported valuations with no regard to incentives, when equipped with a lexicographic tie-breaking rule has pure Nash equilibria and also achieves the above ratio in the worst-case, which means that the bounds are tight.

5. GENERAL SOLUTION CONCEPTS

In the previous sections, we employed the pure Nash equilibrium as the solution concept for bounding the inefficiency

of mechanisms, mainly because of its simplicity. Here, we describe how to extend our results to broader well-known equilibrium concepts in the literature. Due to lack of space, we will only discuss the two most general solution concepts, the *coarse correlated equilibrium* for complete information and the *Bayes-Nash equilibrium* for incomplete information. Since other concepts (like the mixed-Nash equilibrium for instance) are special cases of those two, it suffices to use those for our extensions.

DEFINITION 2. *Given a mechanism M, let $\mathbf{q}$ be a distribution over strategies. Also, for any distribution Δ let Δ_{-i} denote the marginal distribution without the ith index. Then a strategy profile $\mathbf{q}$ is called a*

1. coarse correlated equilibrium *if*

$$\mathop{\mathbb{E}}_{\mathbf{s}\sim\mathbf{q}}[u_i(M_i(\mathbf{s}))] \geq \mathop{\mathbb{E}}_{\mathbf{s}\sim\mathbf{q}}[u_i(M_i((s_i', \mathbf{s_{-i}})))],$$

2. Bayes-Nash equilibrium *for a distribution Δ_u where each $(\Delta_u)_i$ is independent, if when $\mathbf{u}\sim\Delta_u$ then $\mathbf{q}(\mathbf{u}) = \times_i q_i(u_i)$ and for all u_i in the support of $(\Delta_u)_i$,*

$$\mathop{\mathbb{E}}_{\mathbf{u_{-i}},\mathbf{s}\sim\mathbf{q}(\mathbf{u})}[u_i(M_i(\mathbf{s}))] \geq \mathop{\mathbb{E}}_{\mathbf{u_{-i}},\mathbf{s_{-i}}\sim\mathbf{q_{-i}}(\mathbf{u_{-i}})}[u_i(M_i(s_i', \mathbf{s_{-i}}))]$$

where the given inequalities hold for all agents i, and (pure) deviating strategies s_i'. Also notice that for randomized mechanisms definitions are with respect to an expectation over the random choices of the mechanism.

The coarse correlated and the Bayesian Price of Anarchy are defined similarly to the pure Price of Anarchy.

Again, first we mention that we can obtain the extensions to Random Priority rather straightforwardly, based on the fact that even when using probability mixtures over strategies, an agent will always (in every realization) pick her most preferred item among the set of available items when she is chosen. In other words, any pure strategy in the support of the distribution will rank the most preferred available item first, and the ordering of the remaining items does not affect the distribution.

THEOREM 5. *The coarse correlated Price of Anarchy of Random Priority is $O(\sqrt{n})$. The Bayesian Price of Anarchy of Random Priority is $O(\sqrt{n})$.*

PROOF. For the correlated Price of Anarchy, the argument is very similar to the one used in the proof of Theorem 2. Again, if any strategy in the support of a correlated equilibrium $\mathbf{q}$ misranks two items j and j' for any agent i, it can only be because agent i has 0 probability of receiving those items, otherwise agent i would deviate to truthtelling, violating the equilibrium condition. The remaining steps are exactly the same as in the proof of Theorem 2.

For the incomplete information case, consider any Bayes-Nash equilibrium $\mathbf{q}(\mathbf{u})$ and let $\mathbf{u}$ be a any sampled valuation profile. The expected social welfare of the Random Priority can be written as $\mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{s\sim\mathbf{q}(\mathbf{u})}[u_i(\mathbf{s})]\right]$. Using the same argument as the one in the proof of Theorem 2, we can lower bound the quantity $\mathbb{E}_{s\sim\mathbf{q}(\mathbf{u})}[u_i(\mathbf{s})]$ by $\Omega\left(\frac{SW_{OPT}(\mathbf{u})}{\sqrt{n}}\right)$ and the bound follows. □

Next, we turn to Probabilistic Serial and prove the Price of Anarchy guarantees, with respect to coarse correlated equilibria and Bayes-Nash equilibria. Before we state our theorems however, we will briefly discuss the connection of those extensions with the *smoothness* framework of Roughgarden [25]. According to the definition in [25], a game is (λ,μ)-*smooth* if it satisfies the following condition

$$\sum_{i=1}^{n} u_i(s_i^*, \mathbf{s_{-i}}) \geq \lambda SW(\mathbf{s}^*) - \mu SW(\mathbf{s}), \tag{1}$$

where $\mathbf{s}^*$ is a pure strategy profile that corresponds to the optimal allocation and $\mathbf{s}$ is any pure strategy profile. It is not hard to see that a (λ,μ)-smooth game has a Price of Anarchy bounded by $(\mu+1)/\lambda$.

Since establishing that a game is smooth also implies a pure Price of Anarchy bound, an alternative way of attempting to prove Theorem 1 would be to try to show smoothness of the game induced by PS, for $\mu/\lambda = \sqrt{n}$. However, this seems to be a harder task than what we actually do, since in such a proof, one would have to argue about the utilities of agents and possibly reason about the relative preferences for other items, other than the item they are matched with in the optimal allocation. Our approach only needs to consider those items, and hence it seems to be simpler.

An added benefit to the smoothness framework is the existence of the *extension theorem* in [25], which states that for a (λ,μ)-smooth game, the Price of Anarchy guarantee extends to broader solution concepts verbatim, without any extra work. At first glance, one might think that proving smoothness for the game induced by PS might be worth the extra effort, since we would get the extensions "for free". A closer look at our proofs however shows that our approach is very similar to the proof of the extension theorem but using an alternative, simpler condition.

Specifically, the analysis in [25] uses Inequality 1 as a building block and substitutes the inequality into the expectations that naturally appear when considering randomized strategies. This can be done because the condition applies to all strategy profiles $\mathbf{s}$, when $\mathbf{s}^*$ is an optimal strategy profile. This is exactly what we do as well, but we use the inequality $t_j(s_i^*, \mathbf{s_{-i}}) \geq \frac{1}{4}\cdot t_j(\mathbf{s})$ instead, which is simpler but sufficient since it only applies to the game at hand. If $OPT_i(\mathbf{u}) = j$, which is what we use in the proof of Theorem 1, then $(s_i^*, \mathbf{s_{-i}})$ can be thought of as a profile where an agent deviates to her strategy in the optimal profile and hence the left-hand side of the inequality is analogous to the left-hand side of Inequality 1. In a sense, the inequality $t_j(s_i^*, \mathbf{s_{-i}}) \geq \frac{1}{4}\cdot t_j(\mathbf{s})$, can be viewed as a "smoothness equivalent" for the game induced by PS, which then allows us to extend the results to broader solution concepts.

First, we extend Theorem 1 to the case where the solution concept is the coarse correlated equilibrium.

THEOREM 6. *The coarse correlated Price of Anarchy of Probabilistic Serial is $O(\sqrt{n})$.*

PROOF. Let $\mathbf{u}$ be any valuation profile and let i be any agent. Furthermore, let $j = OPT_i(\mathbf{u})$ and let s_i' be the pure strategy that places item j on top of agent i's preference list. By Lemma 1, the inequality $t_j(s_i', \mathbf{s_{-i}}) \geq \frac{1}{4}t_j(\mathbf{s})$ holds for every strategy profile $\mathbf{s}$. In particular, it holds for any pure strategy profile $\mathbf{s}$ where s_i is in the support of the distribution of the mixed strategy q_i of agent i, for any coarse correlated equilibrium $\mathbf{q}$.This implies that

$$\begin{aligned}\mathop{\mathbb{E}}_{\mathbf{s}\sim\mathbf{q}}[u_i(PS_i(\mathbf{s}))] &\geq \mathop{\mathbb{E}}_{\mathbf{s}\sim\mathbf{q}}[u_i(PS_i(s_i', \mathbf{s_{-i}}))] \\ &\geq \mathop{\mathbb{E}}_{\mathbf{s}\sim\mathbf{q}}[u_{ij}t_j(s_i', \mathbf{s_{-i}}))] \geq \frac{1}{4}u_{ij}t_j(\mathbf{s}).\end{aligned}$$

where the last inequality holds by Lemma 1. Using this, we can use very similar arguments to the arguments of the proof of Theorem 1 and obtain the bound. □

For the incomplete information setting, when valuations are drawn from some publically known distributions, we can prove the same upper bound on the Bayesian Price of Anarchy of the mechanism.

THEOREM 7. *The Bayesian Price of Anarchy of Probabilistic Serial is $O(\sqrt{n})$.*

PROOF. The proof is again similar to the proof of Theorem 1. Let $\mathbf{u}$ be a valuation profile drawn from some distribution satisfying the unit-sum constraint. Let i be any agent and let $j_u = OPT_i(\mathbf{u})$, $i \in [n]$. Note that by a similar argument as the one used in the proof of Theorem 1, the expected social welfare of PS is at least 1 and hence we can assume that $\mathbb{E}_{\mathbf{u}}[SW_{OPT}(\mathbf{u})] \geq 2\sqrt{2n}+1$. Observe that in any Bayes-Nash equilibrium $\mathbf{q}(\mathbf{u})$ it holds that

$$\begin{aligned}
\mathop{\mathbb{E}}_{\substack{\mathbf{u}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}[u_i(\mathbf{s})] &= \mathop{\mathbb{E}}_{u_i}\left[\mathop{\mathbb{E}}_{\substack{\mathbf{u_{-i}}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}[u_i(\mathbf{s})]\right] \\
&\geq \mathop{\mathbb{E}}_{u_i}\left[\mathop{\mathbb{E}}_{\substack{\mathbf{u_{-i}}\\ \mathbf{s_{-i}}\sim\mathbf{q_{-i}}(\mathbf{u_{-i}})}}\left[u_i(s_i', \mathbf{s_{-i}})\right]\right] \\
&\geq \mathop{\mathbb{E}}_{u_i}\left[\mathop{\mathbb{E}}_{\substack{\mathbf{u_{-i}}\\ \mathbf{s_{-i}}\sim\mathbf{q_{-i}}(\mathbf{u_{-i}})}}\left[u_{ij_u}t_{j_u}(s_i', \mathbf{s_{-i}})\right]\right] \\
&\geq \mathop{\mathbb{E}}_{u_i}\left[\mathop{\mathbb{E}}_{\substack{\mathbf{u_{-i}}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}\left[\frac{1}{4}u_{ij_u}t_{j_u}(\mathbf{s})\right]\right] \\
&= \frac{1}{4}\mathop{\mathbb{E}}_{\substack{\mathbf{u}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}[u_{ij_u}t_{j_u}(\mathbf{s})]
\end{aligned}$$

where the last inequality holds by Lemma 1 since s_i' denotes the strategy that puts item j_u on top of agent i's preference list. Note that this can be a different strategy for every different $\mathbf{u}$ that we sample. For notational convenience, we use s_i' to denote every such strategy. The expected social welfare at the Bayes-Nash equilibrium is then at least

$$\begin{aligned}
\sum_{i=1}^{n}\mathop{\mathbb{E}}_{\mathbf{u},\mathbf{s}\sim\mathbf{q}(\mathbf{u})}[u_i(\mathbf{s})] &\geq \frac{1}{4}\sum_{i\in[n]}\mathop{\mathbb{E}}_{\substack{\mathbf{u}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}[u_{ij_u}t_{j_u}(\mathbf{s})] \\
&\geq \mathop{\mathbb{E}}_{\substack{\mathbf{u}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}\left[\sum_{i\in[n]}\frac{i}{4n}u_{ij_u}\right] \\
&\geq \mathop{\mathbb{E}}_{\substack{\mathbf{u}\\ \mathbf{s}\sim\mathbf{q}(\mathbf{u})}}\left[\frac{SW_{OPT}(\mathbf{u})(SW_{OPT}(\mathbf{u})-1)}{8n}\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{u}}\left[\frac{SW_{OPT}(\mathbf{u})(SW_{OPT}(\mathbf{u})-1)}{8n}\right] \\
&\geq \frac{\mathbb{E}_{\mathbf{u}}\left[(SW_{OPT}(\mathbf{u}))^2\right]-\mathbb{E}_{\mathbf{u}}\left[SW_{OPT}(\mathbf{u})\right]}{8n} \\
&\geq \frac{\mathbb{E}_{\mathbf{u}}[SW_{OPT}(\mathbf{u})]}{2\sqrt{2n}},
\end{aligned}$$

and the bound follows. □

6. EXTENSIONS

6.1 Price of Stability

Theorem 3 bounds the Price of Anarchy of all mechanisms. A more optimistic (and hence stronger when proving lower bounds) measure of efficiency is the *Price of Stability*, i.e. the worst-case ratio over all valuation profiles of the optimal social welfare over the welfare attained at the *best* equilibrium. We extend Theorem 3 to the Price of Stability of all mechanisms that satisfy a "proportionality" property.

Let $a_1 \succ_i a_2 \succ_i \cdots \succ_i a_n$ be the (possibly weak) preference ordering of agent i. A random assignment vector p_i for agent i *stochastically dominates* another random assignment vector q_i if $\sum_{j=1}^{k} p_{ia_j} \geq \sum_{j=1}^{k} q_{ia_j}$, for all $k = 1, 2, \cdots, n$. The notation that we will use for this relation is $p_i \succ_i^{sd} q_i$.

DEFINITION 3 (SAFE STRATEGY). *Let M be a mechanism. A strategy s_i is a* safe strategy *if for any strategy profile s_{-i} of the other players, it holds that $M_i(s_i, s_{-i}) \succ_i^{sd} \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$.*

We will say that a mechanism M has a safe strategy if every agent i has a safe strategy s_i in M. We now state our lower bound.

THEOREM 8. *The pure Price of Stability of any mechanism that has a safe strategy is $\Omega(\sqrt{n})$.*

PROOF. Let M be a mechanism and let $I = \{k+1, \ldots, n\}$ be a subset of agents. Let $\mathbf{u}$ be the following valuation profile.

- For all agents $i \in I$, let $u_{ij} = \frac{1}{k}$ for $j = 1, \cdots, k$ and $u_{ij} = 0$ otherwise.
- For all agents $i \notin I$, let $u_{ii} = 1$ and $u_{ij} = 0, j \neq i$.

Now let $\mathbf{s}$ be a pure Nash equilibrium on profile $\mathbf{u}$ and let s_i' be a safe strategy of agent i. The expected utility of each agent $i \in I$ in the pure Nash equilibrium $\mathbf{s}$ is

$$\begin{aligned}
\mathbb{E}[u_i(\mathbf{s})] &= \sum_{j\in[n]} p_{ij}(s_i, \mathbf{s_{-i}})v_{ij} \geq \sum_{j\in[n]} p_{ij}(s_i', \mathbf{s_{-i}})v_{ij} \\
&\geq \frac{1}{n}\sum_{j\in[n]} v_{ij} = \frac{1}{n},
\end{aligned}$$

due to the fact that $\mathbf{s}$ is pure Nash equilibrium and s_i' is a safe strategy of agent i. On the other hand, the utility of agent $i \in I$ can be calculated by $\mathbb{E}[u_i(\mathbf{s})] = \sum_{j\in[n]} p_{ij}(s_i, s_{-i})v_{ij} = (\sum_{j=1}^{k} p_{ij})/k$. Because $\mathbf{s}$ is a pure Nash equilibrium, it holds that $\mathbb{E}[u_i] \geq 1/n$, so we get that $\sum_{j=1}^{k} p_{ij} \geq k/n$ for all $i \in I$. As for the rest of the agents,

$$\sum_{i\in N\setminus I}\sum_{j=1}^{k} p_{ij} = k - \sum_{i\in I}\sum_{j=1}^{k} p_{ij} \leq k - (n-k)\frac{k}{n} = \frac{k^2}{n}.$$

This implies that the contribution to the social welfare from agents not in I is at most k^2/n and the expected social welfare of M will be at most $1 + (k^2/n)$. It holds that $SW_{OPT}(\mathbf{u}) \geq k$ and the bound follows by letting $k = \sqrt{n}$. □

Due to Theorem 8, in order to obtain an $\Omega(\sqrt{n})$ bound for a mechanism M, it suffices to prove that M has a safe strategy. In fact, most reasonable mechanisms, including Random Priority and Probabilistic Serial, as well as all ordinal *envy-free* mechanisms satisfy this property.

DEFINITION 4 (ENVY-FREENESS). *A mechanism M is (ex-ante)* envy-free *if for all agents i and r and all profiles $\mathbf{s}$, it holds that $\sum_{j=1}^{n} p_{ij}s_{ij} \geq \sum_{j=1}^{n} p_{rj}s_{rj}$. Furthermore, if M is ordinal, then this implies $p_i^{M,\mathbf{s}} \succ_{s_i}^{sd} p_r^{M,\mathbf{s}}$.*

Given the interpretation of a truth-telling safe strategy as a "proportionality" property, the next lemma is not surprising.

LEMMA 4. *Let M be an ordinal, envy-free mechanism. Then for any agent i, the truth-telling strategy u_i is a safe strategy.*

PROOF. Let $\mathbf{s} = (u_i, \mathbf{s_{-i}})$ be the strategy profile in which agent i is truth-telling and the rest of the agents are playing some strategies $\mathbf{s_{-i}}$. Since M is envy-free and ordinal, it holds that $\sum_{j=1}^{\ell} p_{ij}^{\mathbf{s}} \geq \sum_{j=1}^{\ell} p_{rj}^{\mathbf{s}}$ for all agents $r \in \{1, \ldots, n\}$ and all $\ell \in \{1, \ldots, n\}$. Summing up these inequalities for agents $r = 1, 2, \ldots, n$ we obtain

$$n\sum_{j=1}^{\ell} p_{ij}^{\mathbf{s}} \quad \geq \quad \sum_{j=1}^{\ell}\sum_{r=1}^{n} p_{rj}^{\mathbf{s}} = \ell,$$

which implies that $\sum_{j=1}^{\ell} p_{ij}^{\mathbf{s}} \geq \frac{\ell}{n}$, for all $i \in \{1, \ldots, n\}$, and for all $\ell \in \{1, \ldots, n\}$. □

Note that since Probabilistic Serial is ordinal and envy-free [9], by Lemma 4, it has a safe strategy and hence Theorem 8 applies. It is not hard to see that Random Priority has a safe strategy too.

LEMMA 5. *Random Priority has a safe strategy.*

PROOF. Since Random Priority first fixes an ordering of agents uniformly at random, every agent i has a probability of $1/n$ to be selected first to choose an item, a probability of $2/n$ to be selected first or second and so on. If the agent ranks her items truthfully, then for every $\ell = 1, \ldots, n$, it holds that $\sum_{i=1}^{\ell} p_{ij} \geq \ell/n$. □

In a sense, the safe strategy property is essential for the bound to hold; one can show that the *randomly dictatorial* mechanism, that matches a uniformly chosen agent with her most preferred item and the rest of the agents with items based solely on that agent's reports achieves a constant Price of Stability. On the other hand, the Price of Anarchy of the mechanism is $\Omega(n)$. It would be interesting to show whether Price of Anarchy guarantees imply Price of Stability lower bounds in general.

6.2 Unit-range representation

Our second extension is concerned with the other normalization that is also common in the literature [28, 2, 15], the unit-range representation, that is, $\max_j u_i(j) = 1$ and $\min_j u_i(j) = 0$. First, the Price of Anarchy guarantees from Section 3 extend directly to the unit-range case. For Random Priority, since the results in [15] hold for this normalization as well, we can apply the same techniques to prove the bounds. For Probabilistic Serial, first, observe that Lemma 2 holds independently of the representation. Secondly, in the proof of Theorem 1, it now holds that

$$SW_{PS}(\mathbf{u}, \mathbf{s}) \geq \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} u_{ij} \geq 1,$$

which is sufficient for bounding the Price of Anarchy when $SW_{OPT}(\mathbf{u}) \leq \sqrt{n}$. Finally, the arguments for the case when $SW_{OPT}(\mathbf{u}) \leq \sqrt{n}$ hold for both representations.

Next, we present a Price of Anarchy lower bound for deterministic mechanisms. First, we prove the following lemma about the structure of equilibria of deterministic mechanisms. Note that the lemma holds independently of the choice of representation.

LEMMA 6. *The set of pure Nash equilibria of any deterministic mechanism is the same for all valuation profiles that induce the same preference orderings of valuations.*

PROOF. Let $\mathbf{u}$ and $\mathbf{u}'$ be two different valuation profiles that induce the same preference ordering. Let $\mathbf{s}$ be a pure Nash equilibrium under true valuation profile $\mathbf{u}$ and assume for contradiction that it is not a pure Nash equilibrium under $\mathbf{u}'$. Then, there exists an agent i who by deviating from $\mathbf{s}$ is matched to a more preferred item according to u_i'. But that item would also be more preferred according to u_i and hence she would have an incentive to deviate from $\mathbf{s}$ under true valuation profile $\mathbf{u}$, contradicting the fact that $\mathbf{s}$ is a pure Nash equilibrium. □

Using Lemma 6, we can then prove the following theorem.

THEOREM 9. *The Price of Anarchy of any deterministic mechanism that always has pure Nash equilibria is $\Omega(n)$ for the unit-range representation.*

PROOF. Let M be a deterministic mechanism that always has a pure Nash equilbrium and let $\mathbf{u}$ be a valuation profile such that for all agents i and i', it holds that $u_i = u_{i'}$ and $u_{ij} > u_{ik}$, for all items $i < k$. Let $\mathbf{s}$ be a pure Nash equilibrium for this profile and assume without loss of generality that $M_i(\mathbf{s}) = i$. By Lemma 6, $\mathbf{s}$ is a pure Nash equilibrium for any profile $\mathbf{u}$ that induces the above ordering of valuations. In particular, it is a pure Nash equilibrium for a valuation profile satisfying

- For agents $i = 1, \ldots, \frac{n}{2}$, $u_{i1} = 1$ and $u_{ij} < \frac{1}{n^3}$, for $j > 1$.
- For agents $i = \frac{n}{2} + 1, \ldots, n$, $u_{ij} > 1 - \frac{1}{n^3}$ for $j = 1, \ldots, n/2$ and $u_{ij} < \frac{1}{n^3}$ for $j = \frac{n}{2} + 1, \ldots, n$.

It holds that $OPT(\mathbf{u}) \geq \frac{n}{2}$, whereas the social welfare of M is at most 2 and the theorem follows. □

Again, similarly to the corresponding bound in Section 4, the mechanism that naively maximizes the sum of the reported valuations has pure Nash equilibria and achieves the above bound.

More importantly, it is not clear whether the general lower bound on the Price of Anarchy of all mechanisms that we proved in Theorem 3 extends to the unit-range representation as well. In fact, we do not know of any bound for the unit-range case and proving one seems to be a quite complicated task. As a first step in that direction, the following theorem obtains a lower bound for ϵ-*approximate* (pure) Nash equilibria. A strategy profile is an ϵ-approximate pure Nash equilibrium if no agent can deviate to another strategy and improve her utility by more than ϵ. While the following result applies for any positive ϵ, it is weaker than a corresponding result for exact equilibria.

THEOREM 10. *Let M be a mechanism and let $\epsilon \in (0,1)$. The $\epsilon-$approximate Price of Anarchy of M is $\Omega(n^{1/4})$ for the unit-range representation.*

PROOF. Assume $n = k^2$, where $k \in \mathbb{N}$ will be the size of a subset I of "important" agents. We consider valuation profiles where, for some parameter $\delta \in (0,1)$,

- all agents have value 1 for item 1,
- there is a subset I of agents with $|I| = k$ for which any agent $i \in I$ has value δ^2 for any item $j \in \{2,\ldots,k+1\}$ and 0 for all other items,
- for agent $i \notin I$, i has value δ^3 for items $j \in \{2,\ldots,k+1\}$ and 0 for all other items.

Let $\mathbf{u}$ be such a valuation profile and let $\mathbf{s}$ be a Nash equilibrium. In the optimal allocation members of I receive items $\{2,\ldots,k+1\}$ and such an allocation has social welfare $k\delta^2 + 1$.

First, we claim that there are $k(1-2\delta)$ members of I whose payoffs in $\mathbf{s}$ are at most δ; call this set X. If that were false, then there would be more than $2k\delta$ members of I whose payoffs in $\mathbf{s}$ were more than δ. That would imply that the social welfare of $\mathbf{s}$ was more than $2k\delta^2$, which would contradict the optimal social welfare attainable, for large enough n (specifically, $n > 1/\delta^4$).

Next, we claim that there are at least $k(1-2\delta)$ non-members of I whose probability (in $\mathbf{s}$) to receive any item in $\{1,\ldots,k+1\}$ is at most $4(k+1)/n$; call this set Y. To see this, observe that there are at least $\frac{3}{4}n$ agents who all have probability $\leq 4/n$ to receive item 1. Furthermore, there are at least $3n/4$ agents who all have probability $\leq 4k/n$ to receive an item from the set $2,\ldots,k+1$. Hence there are at least $n/2$ agents whose probabilities to obtain these items satisfy both properties.

We now consider the operation of swapping the valuations of the agents in sets X and Y so that the members of I from X become non-members, and vice versa. We will argue that given that they were best-responding beforehand, they are δ-best-responding afterwards. Consequently $\mathbf{s}$ is an δ-NE of the modified set of agents. The optimum social welfare is unchanged by this operation since it only involves exchanging the payoff functions of pairs of agents. We show that the social welfare of $\mathbf{s}$ is some fraction of the optimal social welfare, that goes to 0 as n increases and δ decreases.

Let I' be the set of agents who, after the swap, have the higher utility of δ^2 for getting items from $\{2,\ldots,k+1\}$. That is, I' is the set of agents in Y, together with I minus the agents in X.

Following the above valuation swap, the agents in X are δ-best responding. To see this, note that these agents have had a reduction to their utilities for the outcome of receiving items from $\{2,\ldots,k+1\}$. This means that a profitable deviation for such agents should result in them being more likely to obtain item 1, in return for them being less likely to obtain an item from $\{2,\ldots,k+1\}$. However they cannot have probability more than δ to receive item 1, since that would contradict the property that their expected payoff was at most δ.

After the swap, the agents in Y are also δ-best responding. Again, these agents have had their utilities increased from δ^3 to δ^2 for the outcome of receiving an item from $\{2,\ldots,k+1\}$. Hence any profitable deviation for such an agent would involve a reduction in the probability to get item 1 in return for an increased probability to get an item from $\{2,\ldots,k+1\}$. However, since the payoff for any item from $\{2,\ldots,k+1\}$ is only δ^2, such a deviation pays less than δ.

Finally, observe that the social welfare of $\mathbf{s}$ under the new profile (after the swap) is at most $1 + 3k\delta^3$. To see this, note that (by an earlier argument and the definition of I') $k(1-2\delta)$ members of I' have probability at most $4(k+1)/n$ to receive any item from $\{1,\ldots,k+1\}$. To upper bound the expected social welfare, note that item 1 contributes 1 to the social welfare. Items in $\{2,\ldots,k+1\}$ contribute in total, δ^2 times the expected number of members of I' who get them, plus δ^3 times the expected number of non-members of I' who get them, which is at most $\delta^2 k 2\delta + \delta^3 k(1-2\delta)$ which is less than $3k\delta^3$.

Overall, the price of anarchy is at least $(k\delta^2+1)/3k\delta^3$, which is more than $1/\delta$. The statement of the theorem is obtained by choosing δ to be less than ϵ, n large enough for the arguments to hold for the chosen δ, i.e. $n > 1/\delta^4$. □

7. CONCLUSION AND FUTURE WORK

Our results are rather negative: we identify a non-constant lower bound on the Price of Anarchy for one-sided matching, and find a matching upper bound achieved by well-known ordinal mechanisms. However, such negative results are important to understand the challenge faced by a social-welfare maximizer: for example, we establish that it is not enough to elicit cardinal valuations, in order to obtain good social welfare. It may be that better welfare guarantees should use some assumption of truth-bias, or some assumption of additional structure in agents' preferences.

An interesting direction of research would be to identify conditions on the valuation space that allow for constant values of the Price of Anarchy or impose some distributional assumption on the inputs and quantify the average loss in welfare due to selfish behavior. For the general, worst-case setting, one question raised is whether one can obtain Price of Anarchy or Price of Stability bounds that match our upper bounds for the unit-range representation as well.

Acknowledgements

George Christodoulou, Paul W. Goldberg and Jinshan Zhang were supported by the EPSRC grant EP/K01000X/1. Paul W. Goldberg was supported by COST Action IC1205. Aris Filos-Ratsikas and Jie Zhang were supported by the ERC Advanced Grant 321171 (ALGAME). Aris Filos-Ratsikas and Søren K.S. Frederiksen acknowledge support from the Danish National Research Foundation and The National Science Foundation of China (under the grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation, within which part of this work was performed and support from the Center for Research in Foundations of Electronic Markets (CFEM), supported by the Danish Strategic Research Council. The authors would like to thank Piotr Krysta for useful discussion.

REFERENCES

[1] Atila Abdulkadiroğlu and Tayfun Sönmez. Matching Markets: Theory and Practice. *Advances in Economics and Econometrics (Tenth World Congress)*, pages 3–47, 2013.

[2] Marek Adamczyk, Piotr Sankowski, and Qiang Zhang. Efficiency of Truthful and Symmetric Mechanisms in One-sided Matching. In *Proceedings of the 7th Symposium of Algorithmic Game Theory*, pages 13–24. Springer, 2014.

[3] Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Eva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008.

[4] Haris Aziz, Felix Brandt, and Markus Brill. The computational complexity of random serial dictatorship. *Economics Letters*, 121(3):341–345, 2013.

[5] Haris Aziz, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Nina Narodytska, and Toby Walsh. Equilibria under the probabilistic serial rule. *arXiv preprint arXiv:1502.04888*, 2015.

[6] Haris Aziz, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Nina Narodytska, and Toby Walsh. Manipulating the probabilistic serial rule. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1451–1459. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[7] Haris Aziz, Serge Gaspers, Nick Mattei, Nina Narodytska, and Toby Walsh. Strategic aspects of the probabilistic serial rule for the allocation of goods. *arXiv preprint arXiv:1401.6523*, 2014.

[8] Anand Bhalgat, Deeparnab Chakrabarty, and Sanjeev Khanna. Social Welfare in One-Sided Matching Markets without Money. In *APPROX-RANDOM*, pages 87–98, 2011.

[9] Anna Bogomolnaia and Hervé Moulin. A New Solution to the Random Assignment Problem. *Journal of Economic Theory*, 100:295–328, 2001.

[10] Steven J Brams, Michal Feldman, John K Lai, Jamie Morgenstern, and Ariel D Procaccia. On maxsum fair cake divisions. In *AAAI*, 2012.

[11] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. *Theory of Computing Systems*, 50(4):589–610, 2012.

[12] Deeparnab Chakrabarty and Chaitanya Swamy. Welfare maximization and truthfulness in mechanism design with ordinal preferences. In *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science*, pages 105–120, 2014.

[13] George Christodoulou, Annamária Kovács, and Michael Schapira. Bayesian Combinatorial Auctions. In *ICALP '08: Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part I*. Springer-Verlag, July 2008.

[14] Ozgün Ekici and Onur Kesten. On the Ordinal Nash Equilibria of the Probabilistic Serial Mechanism. Technical report, working paper, Tepper School of Business, Carnegie Mellon University, 2010.

[15] Aris Filos-Ratsikas, Søren Kristoffer Stiil Frederiksen, and Jie Zhang. Social welfare in one-sided matchings: Random priority and beyond. In *7th International Symposium on Algorithmic Game Theory (SAGT)*, pages 1–12, 2014.

[16] Mingyu Guo and Vincent Conitzer. Strategy-proof allocation of multiple items between two agents without payments or priors. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1*, pages 881–888, 2010.

[17] Tadashi Hashimoto, Daisuke Hirata, Onur Kesten, Morimitsu Kurino, and M Utku Ünver. Two axiomatic approaches to the probabilistic serial mechanism. *Theoretical Economics*, 9(1):253–277, 2014.

[18] Aanund Hylland and Richard Zeckhauser. The Efficient Allocation of Individuals to Positions. *The Journal of Political Economy*, 87(2):293–314, 1979.

[19] Akshay-Kumar Katta and Jay Sethuraman. A solution to the random assignment problem on the full preference domain. *Journal of Economic theory*, 131(1):231–250, 2006.

[20] O Kesten. Probabilistic serial and top trading cycles from equal division for the random assignment problem. Technical report, mimeo, 2006.

[21] Fuhito Kojima and Mihai Manea. Incentives in the probabilistic serial mechanism. *Journal of Economic Theory*, 145(1):106–123, 2010.

[22] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *STACS 99*, pages 404–413. Springer, 1999.

[23] Piotr Krysta, David Manlove, Baharak Rastegari, and Jinshan Zhang. Size versus truthfulness in the House Allocation problem. In *Proceedings of the 15th ACM Conference on Economics and Computation*, pages 453–470. ACM, 2014.

[24] Ariel D Procaccia and Moshe Tennenholtz. Approximate mechanism design without money. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, pages 177–186. ACM, 2009.

[25] Tim Roughgarden. Intrinsic robustness of the price of anarchy. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 513–522. ACM, 2009.

[26] Vasilis Syrgkanis and Eva Tardos. Composable and Efficient Mechanisms. In *STOC '13: Proceedings of the 45th symposium on Theory of Computing*, November 2013.

[27] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior (60th Anniversary Commemorative Edition)*. Princeton university press, 2007.

[28] Lin Zhou. On a Conjecture by Gale about One-Sided Matching Problems. *Journal of Economic Theory*, 52:123–135, 1990.

Egalitarianism of Random Assignment Mechanisms

Haris Aziz
Data61 and UNSW
Sydney, Australia
haris.aziz@nicta.com.au

Aris Filos-Ratsikas
University of Oxford
Oxford, UK
ratsikas@cs.ox.ac.uk

Jiashu Chen
UNSW
Sydney, Australia
jiashu.chen@student.unsw.edu.au

Simon Mackenzie Nicholas Mattei
Data61 and UNSW
Sydney, Australia
{simon.mackenzie,nicholas.mattei}@nicta.com.au

ABSTRACT

We consider the egalitarian welfare aspects of random assignment mechanisms when agents have unrestricted cardinal utilities over the objects. We give bounds on how well different random assignment mechanisms approximate the optimal egalitarian value and investigate the effect that different well-known properties like ordinality, envy-freeness, and truthfulness have on the achievable egalitarian value. Finally, we conduct detailed experiments analyzing the tradeoffs between efficiency with envy-freeness or truthfulness using two prominent random assignment mechanisms — random serial dictatorship and the probabilistic serial mechanism — for different classes of utility functions and distributions.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity; I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems; J.4 [**Computer Applications**]: Social and Behavioral Sciences - Economics

General Terms

Economics, Theory and Algorithms

Keywords

Game theory (cooperative and non-cooperative), Social choice theory

1. INTRODUCTION

We explore the tradeoffs between fairness and efficiency for randomized mechanisms for the *assignment problem*. Specifically, we consider settings where n agents express preferences (cardinal or ordinal) over a set of m indivisible objects. The objective is to assign the objects to agents in a fair and mutually beneficial manner [5, 9, 23, 27]. This general setting has a number of important and significant applications including the assignment of tasks to cores in cloud computing, kidneys to patients in organ exchanges, runways to airplanes in transportation, and students to seats in schools. We consider the classic assumption that irrespective of whether agents are asked to report ordinal or cardinal preferences, they have an underlying utility structure, where each agent assigns real values or cardinal valuations to the different objects [9, 23, 39, 40].

A well-established criterion for fairness is the Rawlsian concept of maximizing the happiness of the least satisfied agent [36, 26, 27, 32]. Following the spirit of this idea, we quantify the fairness of an allocation in terms of its *egalitarian value*: the minimum ratio of the value of objects assigned to an agent to his total valuation for all the objects. The *optimal egalitarian value (OEV)* for a valuation profile of all agents is the best egalitarian value achievable over all assignments. The optimal egalitarian value is well-grounded for a number of reasons. If each agent has a total utility of one for the set of all objects (a standard assumption in the literature [12, 15, 18, 20]), it is equivalent to the maximum egalitarian welfare. The advantage of considering the optimal egalitarian value is that it does not change if agents scale their relative values for the objects. Furthermore, if the egalitarian value of each agent is $1/n$, then the well-known proportionality requirement [11] is satisfied.

The egalitarian value is not the only criterion for desirable allocation mechanisms. Allocation mechanisms may have other goals and requirements such as envy-freeness or truthfulness. Crucially, both these properties are incompatible with optimizing the egalitarian value except in very restricted domains [10]. Thus, it is natural to examine the *tradeoffs* between optimizing the egalitarian value and achieving other desirable properties. In some settings, such as kidney exchanges, the tradeoff between fairness and efficiency is of the utmost concern [17].

Evaluating these tradeoffs also motivates the study of how established mechanisms with other desiderata perform in terms of the egalitarian value. For a given mechanism J, we examine the approximation ratio $guar(J)$, which is the minimum ratio (among all valuation profiles) of the egalitarian value of an allocation returned by the mechanism to the optimal egalitarian value. Our work falls under the umbrella of *approximate mechanism design without money*, a framework set by Procaccia and Tennenholtz [34] for the study of how well mechanisms with certain properties approximate some objective function of the agents' inputs.

In this paper, we study randomized assignment mechanisms for which achieving ex ante fairness is easier com-

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

pared to deterministic mechanisms. Thus, to evaluate the performance of the mechanisms, we compare their egalitarian value with the optimal egalitarian value achieved by any randomized allocation. Note that computing the allocation with the optimal egalitarian value is an NP-hard problem when we restrict ourselves to deterministic allocations [16]. On the other hand, when we consider randomized allocations, the optimal egalitarian value can be computed in polynomial time via a linear program.

We give extra consideration to two randomized assignment mechanisms — *random serial dictatorship (RSD)* and *probabilistic serial (PS)*, which are probably the best-known and most-studied mechanisms in the random assignment literature [9, 22]. In RSD,[1] a permutation over the agents is selected uniformly at random and each agent in the permutation picks the most preferred m/n units of object that are not yet allocated [4, 9, 37]. In PS, each object is considered to have an infinitely divisible probability weight of one. To compute an allocation, agents simultaneously and with the same speed eat the probability weight of their most preferred object which has not been completely consumed. Once an object has been completely eaten by a subset of agents, each of these agents moves on to eat their next most preferred object that has not been completely eaten. The procedure terminates after all the objects have been eaten. The random allocation of an agent by PS is the amount of each object he has eaten [9, 25]. PS satisfies stochastic dominance (SD) envy-freeness (envy-freeness with respect to all cardinal utilities consistent with the ordinal preferences). We also define a mechanism which we refer to as *Optimal Egalitarian and Envy-Free Mechanism (OEEF)*, which maximizes the egalitarian value of an allocation under the constraint that the allocation is envy-free. Allocations under this mechanism can be computed in polynomial time via linear programming since envy-freeness can be captured by linear constraints.

1.1 Our contributions

We present novel theoretical and empirical results regarding fairness in randomized mechanisms. Our main theoretical contributions are as follows.

- For *any* SD envy-free mechanism J: $guar(J) = O(n^{-1})$.
- For *any* envy-free mechanism J: $guar(J) = \Omega(n^{-1})$ and $guar(J) = O(n^{-1/5})$.
- For *any* truthful-in-expectation mechanism J: $guar(J) = O(n^{-1/5})$.
- For *any* ordinal mechanism J: $guar(J) = O(n^{-1})$.

The first three results apply to mechanisms that may be cardinal mechanisms. As a result of our general bounds, we also get asymptotically tight bounds of $\Theta(n^{-1})$ for RSD and PS. As a result of our general bounds for envy-free mechanisms, we obtain bounds for well-known envy-free mechanisms such as *competitive equilibrium with equal incomes (CEEI)* [39] and the *pseudo-market* mechanism [23]. Since a random assignment of indivisible objects can also be interpreted as a fractional assignment of divisible objects, *our results apply as well to fair allocation of divisible objects.*

[1]The original definition of RSD is for n agents and n objects; the definition here is a straightforward adaptation for $n < m$.

The constructions that provide the upper bounds for the *guar* values can be considered as extreme examples that may not be common in real-life scenarios. In order to better understand how the mechanisms may perform in practice, we consider the approximation ratio achieved by RSD and PS. We also examine the effect of imposing the envy-freeness constraint. We generate ordinal profiles via a Mallows model for different levels of dispersion ϕ from a common reference ranking of objects, assigning cardinal utilities via the Borda and exponential scoring functions. Sweeping ϕ from 0, where all agents have the same preference, to 1.0, where all preference orders are equally likely (the Impartial Culture), allows us to make statements regarding situations where agent preferences are more or less correlated. We make the following observations.

- There is a negligible difference between the minimum and average achievable approximation ratios for PS and RSD under Borda utilities. While PS performs slightly better than RSD when agents have more extreme (exponential) utilities, both mechanisms perform strictly worse when agents' valuations are more similar, as they are under Borda utilities.
- When we require envy-freeness (as in OEEF) with exponential utilities, as ϕ increases towards 1.0 (i.e. Impartial Culture) the achievable approximation ratio first decreases slightly and then increases. Hence, as agents value more disparate objects highly, satisfying envy-freeness does not impose as stiff a penalty on the achievable approximation ratio.
- In our experiments, the requirement of envy-freeness as a constraint in itself (as in the OEEF mechanism) does not have a large impact on the OEV. However, since PS returns an SD envy-free (envy-free for all cardinal utilities consistent with the ordinal preferences) allocation, its achievable approximation ratio is strictly less than OEEF.

1.2 Related work

The assignment problem has been in the center of attention in recent years in both computer science and economics [12, 20, 21]. Often, in the classical assignment literature, agents are assumed to have an underlying cardinal utility preference structure, even if they are not asked to report it explicitly. On the other hand, there are many examples of well-known cardinal mechanisms, such the pseudo-market (PM) mechanism of Hylland and Zeckhauser [23] and the competitive equilibrium with equal incomes (CEEI) mechanism [39]. Both mechanisms return allocations that are envy-free in expectation. The two prominent ordinal mechanisms in the literature are the probabilistic serial mechanism (PS) [9, 13, 38] and random serial dictator (RSD), a folklore mechanism that pre-existed the formulation of the assignment problem in [23]. Later, Che and Kojima [13] proposed a variant of PS called *multi-unit eating probabilistic serial (MPS)* that was formalised and axiomatically studied by Aziz [2].

The egalitarian welfare has received considerable interest within the computer science literature, especially for allocation of discrete objects in a deterministic manner. The problem is also referred to as the *Santa Claus problem* in which the goal is to compute an assignment which maximizes the utility of the agent that gets the least utility [1, 6].

For deterministic settings, Demko and Hill [16] proved that the problem is NP-hard. On the other hand, for randomized/fractional allocations, the problem can be solved via a linear program.[2] Recently, another fairness constraint that has been considered is the maxmin fair share [11, 35]. The notion coincides with proportionality in the context of randomized/fractional allocations and hence is weaker than OEV.

Another popular objective is the maximization of the *utilitarian welfare*, i.e. the sum of agents' valuations for an assignment. Filos-Ratsikas et al. [18] proved that RSD guarantees $\Omega(n^{-1/2})$ of the total utilitarian welfare if the utilities are normalized to sum up to one for each agent, which is asymptotically optimal among all randomized truthful mechanisms. In a recent paper, Christodoulou et al. [14] proved similar results for the price of anarchy with respect to the utilitarian welfare of random assignment mechanisms, including RSD and PS. In this paper, we consider the effect on approximations of the egalitarian value from strategic aspects (truthful mechanisms), limited information (ordinal mechanisms), or additional fairness requirements (envy-free mechanisms). The egalitarian value does not require the agents' utilities to be normalized and does not require agents' utilities to be added.

Bhalgat et al. [7] determined the approximation ratio of RSD and PS when the objective is the maximization of a different notion, the ordinal social welfare, which is related to the "popularity" of an assignment [24]. Caragiannis et al. [12] examined the issue of how much efficiency loss fairness requirements like envy-freeness incur but crucially, their objective is maximization of utilitarian welfare.

2. PRELIMINARIES

An assignment problem is a triple (N, O, v) such that $N = \{1, \ldots, n\}$ is the set of agents, $O = \{o_1, \ldots, o_m\}$ is the set of indivisible objects, and $v = (v_1, \ldots, v_n)$ is the valuation profile which specifies for each agent $i \in N$ utility or valuation function v_i where v_{ij} or $v_i(o_j)$ denotes the value of agent i for object o_j. We will denote by V^n the set of all possible valuation profiles.

A fractional or random allocation p is a $(n \times m)$ matrix $[p(i)(j)]$ such that $p(i)(j) \in [0, 1]$ for all $i \in N$, and $o_j \in O$. We denote by $\mathcal{P}$ the set of all feasible allocations. The term $p(i)(j)$ which we will also write as $p(i)(o_j)$ represents the probability of object o_j being allocated to agent i. Each row $p(i) = (p(i)(1), \ldots, p(i)(m))$ represents the *allocation of agent i*. The set of columns correspond to the objects $o_1, \ldots, o_m$. We will denote by $\hat{p}$ the m dimensional vector where the j-th entry is $\sum_{i \in N} p(i)(j)$ and denotes the total probability that object j will be allocated to some agent. The utility of agent i from allocation p is $u_i(p(i)) = \sum_{j \in O}(p(i)(j))v_{ij}$. An allocation p is *proportional* if for all $i \in N$, $u_i(p(i)) \geq \frac{1}{n}u_i(O)$. An allocation p is *envy-free* if for all $i, j \in N$, $u_i(p(i)) \geq u_i(p(j))$. An allocation is *SD envy-free* if it is envy-free with respect to all cardinal utilities consistent with the ordinal preferences.[3]

[2]Even a lexicographic refinement of the OEV maximizing allocations (in which the value of the worst off agent, then the second worst off agent, and so on, are maximized) can be computed in polynomial time via a series of linear programs [19].

[3]SD envy-freeness also applies to cardinal mechanisms e.g., one that maximize total welfare subject to SD envy-freeness constraints.

We will consider randomized mechanisms that return a random allocation for each instance of an assignment problem. Note the connection between random assignments for indivisible objects and fractional assignments of divisible objects; a random assignment can be viewed as a fractional assignment when agents have additive utilities over the objects. In that sense, we can use well-known mechanisms for fractional assignments, like the CEEI mechanism, as randomized mechanisms for our setting.

We say that a mechanism is proportional if it always returns a proportional allocation. Similarly, a mechanism is envy-free if it always returns an envy-free allocation. A mechanism M is *truthful-in-expectation*, if for any agent $i \in N$, any valuation profile $v = (v_i, v_{-i})$ and any misreport v_i' of agent i it holds that $u_i(M(v_i, v_{-i})) \geq u_i(M(v_i', v_{-i}))$, i.e. no agent has any incentive to misreport her true valuation.

Two valuations v_i and v_i' are *ordinally equivalent* if they induce the same ranking over objects, formally $v_i(o_j) \geq v_i(o_k)$ iff $v_i'(o_j) \geq v_i'(o_k)$. A profile v is ordinally equivalent to profile v' if for each $i \in N$, v_i and v_i' are ordinally equivalent. A mechanism J is *ordinal* if for any two preference profiles v and v' that are ordinally equivalent, $J(v) = J(v')$, i.e., the allocations are the same for any pair of ordinally equivalent profiles. We now define the main efficiency measures that we will examine in the paper.

- The *egalitarian value (EV)* of an allocation p with respect to valuation profile v is $EV(p, v) = \inf\{\frac{u_i(p(i))}{u_i(O)} : i \in N\}$.
- For a given valuation profile, the *optimal egalitarian value (OEV)* is the maximum possible egalitarian value that can be achieved $OEV(v) = \sup_\lambda\{\exists p \in \mathcal{P} : EV(p, v) = \lambda\}$.
- For a given valuation profile v, an allocation p *achieves approximation ratio* $\frac{EV(p,v)}{OEV(v)}$.
- For a given mechanism J and valuation profile v, we will say that J *achieves approximation ratio* $aar(J, v)$ *for valuation* v where $aar(J, v)$ is defined as $aar(J, v) = \frac{EV(J(v),v)}{OEV(v)}$.
- An allocation rule J *guarantees an approximation ratio* of $guar(J)$ where $guar(J)$ is defined as $guar(J) = \inf_{v \in V^n}\{aar(J, v)\}$.

The guaranteed approximation ratio $guar(J)$ is the worst-case guarantee over all instances of the problem that we will be looking to maximize in our theoretical results.

3. THEORETICAL RESULTS

We first note that for a deterministic mechanism J, $guar(J) = 0$; in the worst case, if all the agents only value the same object then all agents get zero utility except the agent who gets the valued object. From now on, we will focus on randomized mechanisms. We start with the following lemma about proportional mechanisms.

LEMMA 1. *For any mechanism J that is proportional, $guar(J) \geq n^{-1}$.*

PROOF. If the allocation p is proportional then for each $i \in N$, $u_i(p(i)) \geq n^{-1}(u_i(O))$. Since $EV(p,v) \geq \inf_{i \in N}(u_i(p(i))/(u_i(O))$ and $(u_i(p(i))/(u_i(O)) \geq n^{-1}$ for all $i \in N$, we get that $EV(p,v) \geq n^{-1}$. Since $OEV(v) \leq 1$, $\frac{EV((p,v))}{OEV(v)} \geq n^{-1}$. □

Since both PS and RSD are proportional [3, 9], we obtain the following guarantee on their approximation ratio: $guar(PS) \geq n^{-1}$ and $guar(RSD) \geq n^{-1}$.
A mechanism satisfies the *favourite share* property if whenever all the agents have the same most preferred object then each agent is assigned to it with probability n^{-1}. We obtain the following theorem.

THEOREM 1. *For any mechanism J that satisfies the favourite share property, $guar(J) = O(n^{-1})$.*

PROOF. Consider the following valuation profile with $n = m$, where ϵ is an arbitrarily small positive value.

$$v_1(o_j) = \begin{cases} 1, & \text{if } j = 1 \\ 0, & \text{otherwise.} \end{cases}$$

For $i \in \{2, \ldots, n\}$,

$$v_i(o_j) = \begin{cases} 0.5 + \epsilon, & \text{if } j = 1 \\ 0.5 - \epsilon, & \text{if } j = i \\ 0, & \text{otherwise.} \end{cases}$$

Note that OEV is at least 0.5 that can be achieved by allocating most of o_1 to agent 1 and the rest uniformly to the other agents and the o_j completely to agent j for $j \in \{2, \ldots, n\}$. On the other hand, J gives $1/n$ of o_1 to each of the agents so that agent 1 gets utility $1/n$. Since ϵ can be arbitrarily small, it follows that $guar(J) = O(n^{-1})$. □

We remark here that Theorem 1 holds even if agents have strict preferences; the utilities can be perturbed slightly to reflect strict preferences. Since RSD and PS satisfy the favourite share property as well as proportionality, $guar(RSD) = \Theta(n^{-1})$ and $guar(PS) = \Theta(n^{-1})$.

THEOREM 2. *For any mechanism J that satisfies SD envy-freeness, $guar(J) = O(n^{-1})$.*

PROOF. SD envy-freeness implies the favourite share property. If an allocation does not satisfy the favourite share property, then the agent who gets less than $1/n$ of his most preferred object will be envious of another agent if he has extremely high utility for the object. □

In the following, we will prove an upper bound on the approximation ratio of the OEV for two classes of mechanisms: envy-free mechanisms and truthful mechanisms. First we prove the following lemma that states that when looking for upper bounds on the approximation ratio, it suffices to only consider anonymous mechanisms. Similar lemmas have been proven before in literature [18, 20].

LEMMA 2. *Let J be a mechanism with approximation ratio ρ. Then, there exists another mechanism J' which is anonymous and has approximation ratio at least ρ. Furthermore, if J is truthful or truthful-in-expectation, then J' is truthful-in-expectation and if J is envy-free, J' is envy-free.*

PROOF. Let J' be the mechanism that on input valuation profile v first applies a uniformly random permutation to the set of agents and then runs mechanism J on v. Obviously, J' is anonymous. Additionally, since v can be an input to J and the approximation ratio is calculated over all possible instances, the ratio of J cannot be better than the ratio of J'. Finally, since the permutation is independent of valuations, if J is truthful or truthful-in-expectation, J' is truthful-in-expectation. □

Now we state the following theorem, bounding the approximation ratio of any truthful-in-expectation mechanism. RSD and the uniform mechanism (that gives assignment probability of $1/n$ of each object to each agent) are strategyproof and ordinal mechanisms that both achieve a $\Theta(n^{-1})$ approximation of the OEV. We prove that for any truthful-in-expectation mechanism J, it holds that $guar(J) = O(n^{-1/5})$.

THEOREM 3. *For any truthful-in-expectation mechanism J, $guar(J) = O(n^{-1/5})$.*

PROOF. Let J be a truthful-in-expectation mechanism; by Lemma 2, we can assume without loss of generality that J is anonymous. Consider the following valuation profile v (summarized in Figure 1) with $n = n_1 + {n_1}^2 + n_1^{5/2}$ agents and ${n_1}^2 + 1$ objects, where ϵ will be defined later:

	$\mathbf{n_1^2+1}$ **Objects**	**1**	**2**	**3**	**4**	$\cdots$	$\mathbf{n_1^2+1}$	
n_1 agents, set A		1	0	0	0	$\cdots$	0	
		1	0	0	0	$\cdots$	0	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
		1	0	0	0	$\cdots$	0	
${n_1}^2$ agents, set B		$1-\epsilon$	ϵ	0	0	$\cdots$	0	
		$1-\epsilon$	0	ϵ	0	$\cdots$	0	
		$1-\epsilon$	0	0.	ϵ	$\cdots$	0	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
		$1-\epsilon$	0	0	0	$\cdots$	ϵ	
${n_1}^{\frac{5}{2}}$ agents, set C		0	1	0	0	$\cdots$	0	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\sqrt{n_1}$ agents, set C_1
		0	1	0	0	$\cdots$	0	
		0	0	1	0	$\cdots$	0	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\sqrt{n_1}$ agents, set C_2
		0	0	1	0	$\cdots$	0	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
		0	0	0	0	$\cdots$	1	
		$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\sqrt{n_1}$ agents, set $C_{n_1^2}$
		0	0	0	0	$\cdots$	1	

Figure 1: Valuation profile v.

- For every agent $i \in A = \{1, \ldots, n_1\}$, it holds that $v_i(1) = 1$ and $v_i(j) = 0$ for every object $j \neq 1$.
- For every agent $i \in B = \{n_1 + 1, {n_1}^2\}$, it holds that $v_i(1) = 1 - \epsilon$, $v_i(i - n_1 + 1) = \epsilon$ and $v_i(j) = 0$ for all objects $j \in O \backslash \{1, i - n_1 + 1\}$.
- For every $\ell = 1, \ldots, {n_1}^2$ and agent $i \in C_\ell = \{{n_1}^2 + (\ell - 1)\sqrt{n_1} + 1, {n_1}^2 + \ell\sqrt{n_1}\}$, it holds that $v_i(\ell + 1) = 1$ and $v_i(j) = 0$ for all objects $j \neq \ell + 1$.

In other words, the instance consists of n_1 agents that have value 1 for the first object (set A) and 0 for everything

else, ${n_1}^2$ agents that value object 1 at $1-\epsilon$ and another object at value ϵ (set B) and $n_1^{5/2}$ agents that value a single object at 1 (set $C = \cup_l C_l$), such that $\sqrt{n_1}$ agents value the object that some agent in the set B has value ϵ for.

Now if we let $\epsilon = 1/(n_1 - \sqrt{n_1})$, then the egalitarian value of the optimal allocation is at least $1/n_1$; an allocation with such a value is the following:

- Every agent $i \in A$ is allocated $1/n_1$ of object 1.
- Every agent $i \in B$ is allocated $(n_1 - \sqrt{n_1})/n_1)$ of the object they value at ϵ.
- Every agent $i \in C_l$ is allocated $1/n_1$ of object $l+1$.

Next consider a family of valuation profiles $\mathcal{V}$, consisting of profiles where all agents have the same valuations as in v, except one agent from B that has value 1 for the object that she had value ϵ in v and 0 for all other objects. Formally, for $\ell = 1, \ldots, {n_1}^2$, we define a profile $v^\ell \in \mathcal{V}$ as follows:

- For every agent $i \neq n_1 + \ell$, it holds that $v_i^\ell(j) = v_i(j)$ for all objects $j \in M$.
- For agent $n_1 + \ell$, it holds that $v_{n_1+\ell}^\ell(\ell+1) = 1$ and $v_{n_1+\ell}^\ell(j) = 0$, for all objects $j \neq \ell + 1$.

Consider now any ℓ and the corresponding valuation profile v^ℓ. Since J is anonymous and agents in $C_\ell \cup \{n_1 + \ell\}$ have identical valuations and since $|C_\ell| = \sqrt{n_1}$, the probability that agent $n_1+\ell$ is allocated object $\ell+1$ is at most $1/(\sqrt{n_1}+1)$ and her utility is hence at most $1/(\sqrt{n_1}+1)$. Now consider valuation profile v and consider the probability $p(n_1+\ell)(\ell+1)$ that agent $n_1+\ell$ is allocated object $\ell+1$. By truthfulness, and since $v_{n_1+\ell}$ could be a misreport from $v_{n_1+\ell}^\ell$, it must hold that $p(n_1+\ell)(\ell+1) \leq 1/(\sqrt{n_1}+1) < 1/\sqrt{n_1}$. This implies that the contribution to the expected utility of agent $n_1 + \ell$ from object $\ell + 1$ is at most $\epsilon/\sqrt{n_1}$, which is at most $1/(n_1\sqrt{n_1} - n_1)$.

Now consider the probability $p(n_1 + \ell)(1)$ that agent $n_1 + \ell$ is allocated object 1. From the arguments above, if $p(n_1 + \ell)(1) < 1/(n_1\sqrt{n_1} - n_1)$, then the expected utility of agent $n_1 + \ell$ is at most $2/(n_1\sqrt{n_1} - n_1)$ and the ratio is $O(1/\sqrt{n_1})$. Since $n = n_1 + {n_1}^2 + n_1^{5/2}$, that would mean that the theorem is proven. Hence, for J to achieve a better ratio than $O(n^{-1/5})$, it has to be the case that for every $\ell = 1, \ldots, {n_1}^2$, it holds that $p(n_1 + \ell)(1) \geq 1/(n_1\sqrt{n_1} - n_1)$. This is not possible however, since then $\sum_{\ell=1}^{{n_1}^2} p(n_1+\ell)(1) \geq n_1/(\sqrt{n_1} - 1) > 1$. This completes the proof. □

Note that for utilitarian welfare maximization, Filos-Ratsikas et al. [18] proved that an ordinal mechanism, RSD achieves the best approximation ratio among all truthful mechanisms. We conjecture that this is the case for the maximization of the egalitarian value as well, i.e. for any truthful mechanism J, $guar(J) = O(n^{-1})$.

We now turn our attention to envy-free mechanisms. For this class, we will prove an $O(n^{-1/5})$ upper bound as well; the proof actually uses the same valuation profile as the proof of Theorem 3.

THEOREM 4. *For any mechanism J that satisfies envy-freeness, $guar(J) = O(n^{-1/5})$.*

PROOF. Consider the valuation profile v used in the proof of Theorem 3 and again consider the probability $p(n_1+\ell)(\ell+1)$ that agent $n_1 + \ell$ is allocated object $\ell + 1$. Recall the definition of sets A, B and C from the proof of Theorem 3. By envy-freeness, it holds that $p(n_1 + \ell)(\ell + 1) \leq 1/(\sqrt{n} + 1) \leq 1/\sqrt{n_1}$ otherwise some agent $j \in C_\ell$ (who only values object $\ell + 1$) would be envious of agent $n_1 + \ell$.

The rest of the steps are the same as in the proof of Theorem 3. Again, consider the probability $p(n_1 + \ell)(1)$ that agent $n_1 + \ell$ is allocated object 1. Since $p(n_1 + \ell)(1) < 1/(n_1\sqrt{n_1} - n_1)$, if $p(n_1 + \ell)(1) < 1/(n_1\sqrt{n_1} - n_1)$ then for the same reasons mentioned in the last paragraph of the proof of Theorem 3, we are done. Hence, we can assume that for every $\ell = 1, \ldots, {n_1}^2$, it holds that $p(n_1 + \ell)(1) \geq 1/(n_1\sqrt{n_1} - n_1)$. This is not possible however, since then $\sum_{\ell=1}^{{n_1}^2} p(n_1 + \ell)(1) \geq n_1/(\sqrt{n_1} - 1) > 1$. □

From Theorem 4, we obtain the following corollary: $guar(CEEI) = O(n^{-1/5})$ and $guar(PM) = O(n^{-1/5})$.
It would be interesting to provide a better bound for Theorem 4 or show it is optimal, i.e. come up with an envy-free mechanism that actually achieves the ratio. Finally, we consider the OEV guarantees of ordinal mechanisms.

THEOREM 5. *For any mechanism J that is ordinal, $guar(J) = O(n^{-1})$.*

PROOF. Consider the setting with n agents and $n + 1$ objects $\{o^*, o_0, \ldots, o_{n-1}\}$. The preferences are as follows: each agent values o^* the most. Agent 1 has preference order $o^*, o_0, \ldots, o_{n-2}, o_{n-1}$. The preference of each agent $i \in N \setminus \{1\}$ over the objects $O \setminus \{o^*\}$ are obtained as follows: take agent $i-1$ preference order over $O \setminus \{o^*\}$ and move the most preferred object of $i - 1$ among $O \setminus \{o^*\}$ to the end of the preference order for agent i.

$$
\begin{array}{lllll}
1: & o^*, o_0, & , \ldots, o_{n-2}, & & o_{n-1} \\
2: & o^*, o_1, & , \ldots, o_{n-1}, & & o_0 \\
\vdots & & & & \\
i: & o^*, o_{i-1}, & , \ldots, o_{n-i+1}, & & o_{i-2}
\end{array}
$$

By Lemma 2, we can assume without loss of generality that J is anonymous. Furthermore, since J is ordinal, due to the preference profile, the mechanism cannot differentiate among the agents even though they may have different valuations over the objects. Assume that there is some agent that is allocated at most $1/n$ of the universally most preferred object o^*. In this case, consider the scenario where this agent has utility almost 1 for o^* and the other agents i have utility $0.5 + \epsilon$ for o^* and utility $0.5 - \epsilon$ for o_{i-1} where ϵ is an arbitrarily small positive value. In this case, the egalitarian value achieved is $1/n$ whereas the OEV is almost 0.5. Hence $guar(J) = O(1/n)$. □

Since MPS is an ordinal mechanism, it follows that $guar(MPS) = O(n^{-1})$.

4. EXPERIMENTAL RESULTS

The results in Section 3 give us worst case bounds on the *guaranteed approximation ratios ($guar(J)$)* for a number of prominent randomized mechanisms including RSD and PS. Hence, in this section we present experimental results which provide a perspective on what may happen "in practice." Since PS can be considered as the most efficient SD envy-free mechanism (in view of various characterizations [8, 38]),

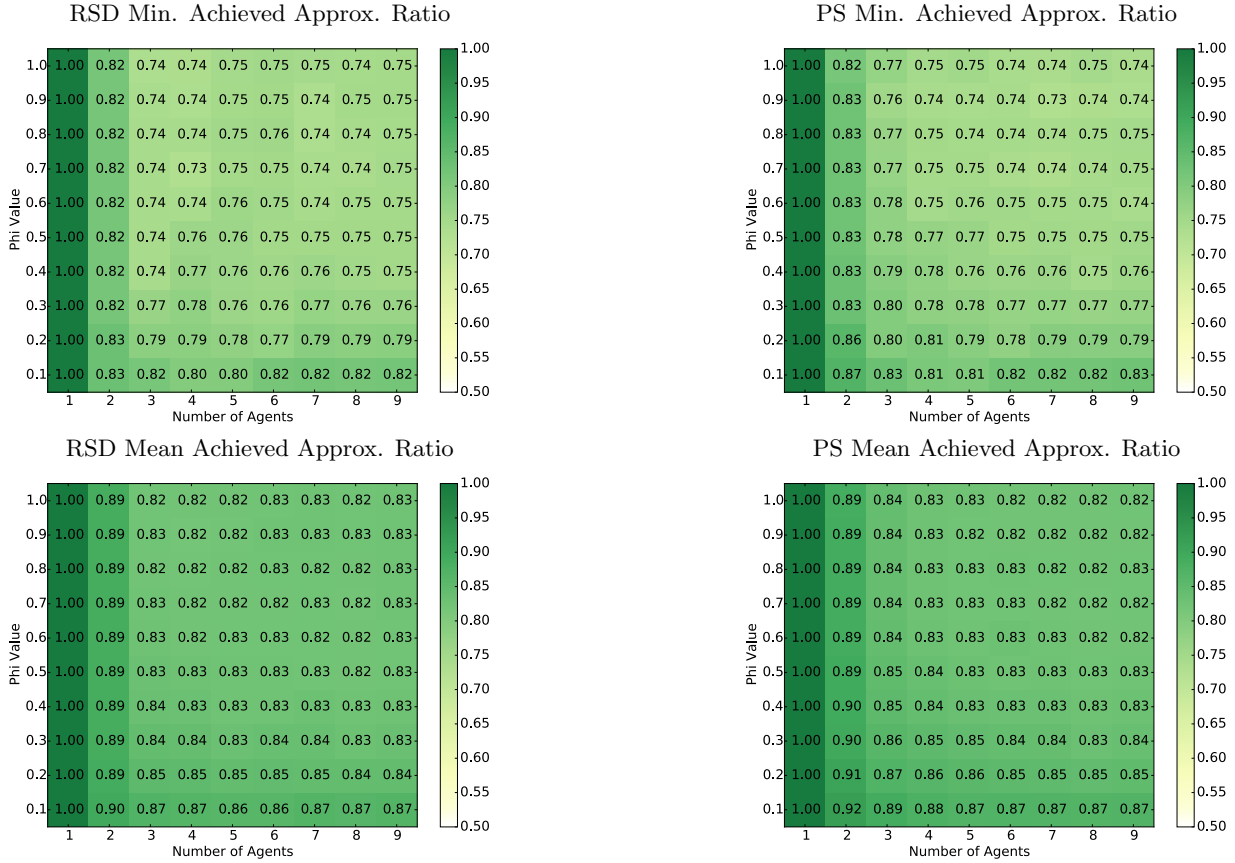


Figure 2: Minimum (top) and average (bottom) achieved approximation ratio for the RSD (left) and PS (right) mechanisms with Borda utilities. Observe that both mechanisms perform similarly and significantly better than the derived $guar(J)$. Both mechanisms are relatively invariant to the level of dispersion in the underlying valuation profiles. For each $n = \{1, \ldots, 9\}$ the graphs are aggregated over the complete range of objects (i.e., all $m \in \{2, \ldots, 9\}$). For example, the cell $(n = 4, \phi = 0.2)$ is the minimum (resp. average) achieved approximation ratio over all instances where m ranges over $\{1, \ldots, 9\}$.

the results for PS can also be viewed as the effect of enforcing SD envy-freeness. In order to test the quality of RSD and PS we need to generate both preferences and cardinal utilities for the agents. There are a number of generative statistical cultures that are commonly used to generate ordinal preferences over objects and the choice of model can have significant impact on the outcome of an experimental study (see e.g. [33]).

Since our focus is fairness, and fairness is often hard to achieve when agents have similar valuations, we employ the *Mallows model* [29] and use the generator from WWW.PREFLIB.ORG [31] in our study. Mallows models are often used in machine learning and preference handling as they allow us to easily control the correlation between the preferences of the agents; a common phenomenon in preference data [31, 30, 28]. A Mallows model has two parameters: (1) a **Reference Order (σ)**, the preference order at the center of the distribution, and (2) a **Dispersion Parameter (ϕ)**, the variance in the distribution which controls the level of similarity of the agent preference orders. When $\phi = 0$ all agents have the same ordinal preference; when $\phi = 1$ then the ordinal preferences are drawn uniformly at random from the space of all preference orders.

Formally, the probability of observing an ordering r is inversely proportional to the Kendall Tau distance between σ and r. This probability is weighted by ϕ, which allows us to control the shape of the distribution. For a given ordinal preference, we superimpose cardinal utilities for the agents using two well-established scoring functions: (1) **Borda Utilities**, each agent has valuation of $m - i$ for his i-th preferred object, and (2) **Exponential Utilities**, each agent has valuation of 2^{m-i} for his i-th preferred object.

In our experiments we generate 10,000 valuation profiles (instances) for each combination of parameters with the number of agents $n \in \{2, \ldots, 9\}$, number of objects $m \in \{2, \ldots, 9\}$, and dispersion parameter $\phi \in \{0.0, 0.1, \ldots, 1.0\}$. We draw σ i.i.d. for each instance.

4.1 Experiments: The Performance of RSD and PS

For each instance v generated, and each mechanism J, we examined the achieved approximation ratio, $aar(J, v) = \frac{EV(J(v),v)}{OEV(v)}$, of the RSD and PS mechanisms. Among all such values computed, we examined the minimum and average ratio achieved for a given set of parameters. The results of our experiments for Borda Utilities are shown in Figure 2 while

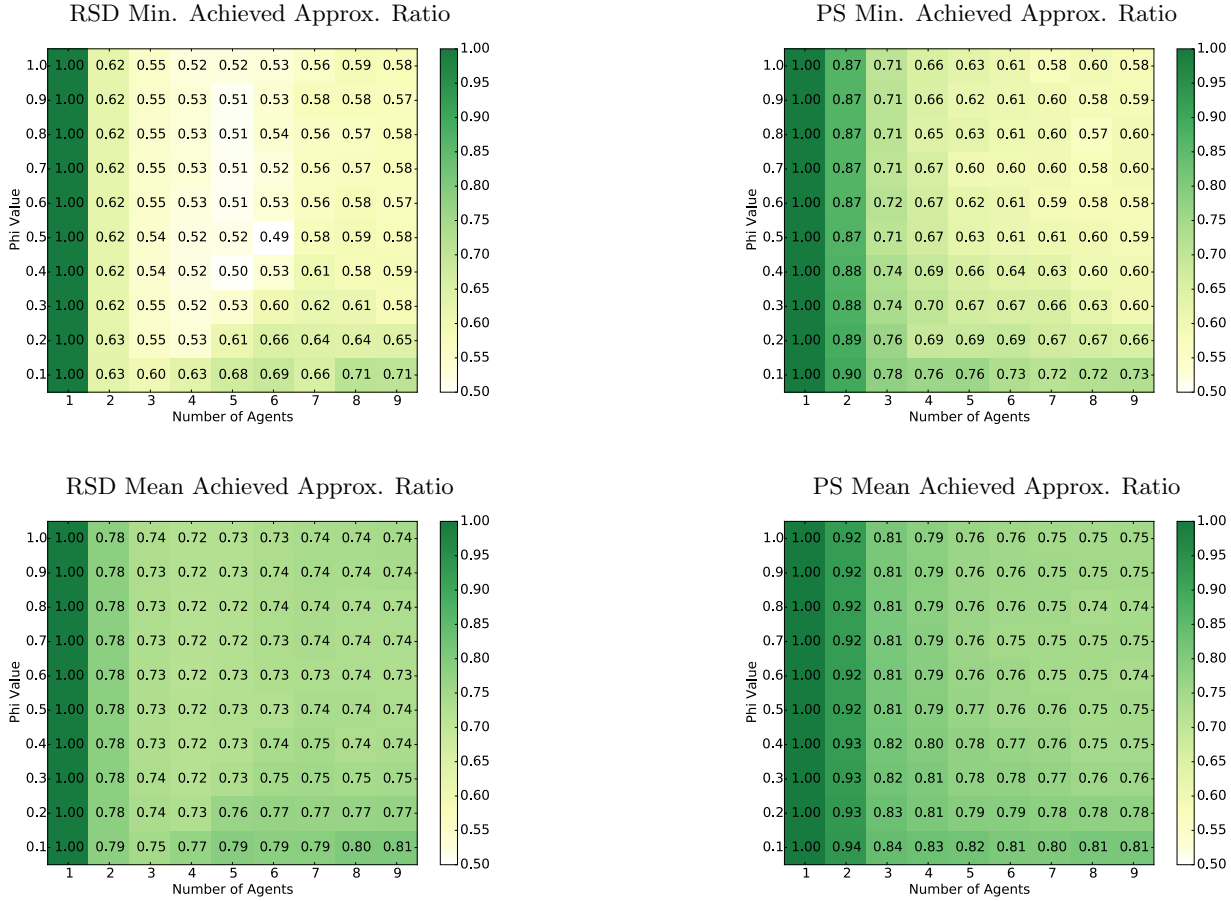


Figure 3: Minimum (top) and average (bottom) achieved approximation ratio for the RSD (left) and PS (right) mechanism with exponential utilities. For each $n = \{1, \ldots, 9\}$ these graphs are aggregated over the complete range of objects (i.e., all $m \in \{2, \ldots, 9\}$). For example, the cell $(n = 4, \phi = 0.2)$ is the minimum (resp. average) approximation ratio achieved over all instances where m ranges over $\{1, \ldots, 9\}$.

the results for exponential utilities are shown in Figure 3. All our figures are aggregations over all values of m for particular combinations of n and ϕ. Note that since $aar(J, v)$ is normalized over the total utility we can aggregate these terms as it is invariant to this scaling. This allows us to draw more general conclusions as we range over the number of agents and objects. Empirically we found that increasing the number of agents has a greater impact on the approximation performance of the mechanisms compared to an increase in the number of objects, hence the decision to aggregate the graphs in the manner chosen. This empirical result is in line with our theoretical results showing that the worst case approximation ratio is a function of n. The results for both mechanisms for Borda utilities strictly dominate the results in Figure 3. Hence, we observe that the achieved approximation ratio is better for Borda utilities than for exponential utilities.

When $\phi = 0.0$ (not shown in our graphs), the achieved approximation ratio is 1 for both PS and RSD. Both of these mechanisms return the uniform allocation, assigning probability of $1/n$ for each object to each agent, when all the agents have identical preferences. In general, sweeping the value of ϕ from completely correlated to completely uncorrelated preferences has little impact on the overall achievable approximation ratio, though for both models the achievable approximation ratio did strictly decrease as we increased ϕ. The impact of changing ϕ was strictly greater for the exponential utility model than it was for the Borda utility model, highlighting again that, as the difference between the valuations of the objects grows large, it becomes harder to achieve fair allocations.

There appears to be almost no difference between the minimum and average ratios for PS and RSD under Borda utilities. Furthermore, these ratios appear to be very high compared to our theoretical results. Finally, PS consistently performs slightly better than RSD for the minimum and average ratios under exponential utilities and on par with RSD for Borda utilities. This provides more empirical support to the argument that PS is superior to RSD in terms of fairness.

4.2 Experiments: The Effect of Envy-freeness

In order to evaluate the effect that envy-freeness has on the allocations we turn to the OEEF mechanism. To understand the worst case effects of adding envy-freeness as a hard constraint has on small instances we exhaustively tested the parameter space with agents $n \in \{2, \ldots, 6\}$, number of objects $m \in \{3, 4\}$ under Borda and exponential utilities. In this entire parameter space, the worse case achiev-

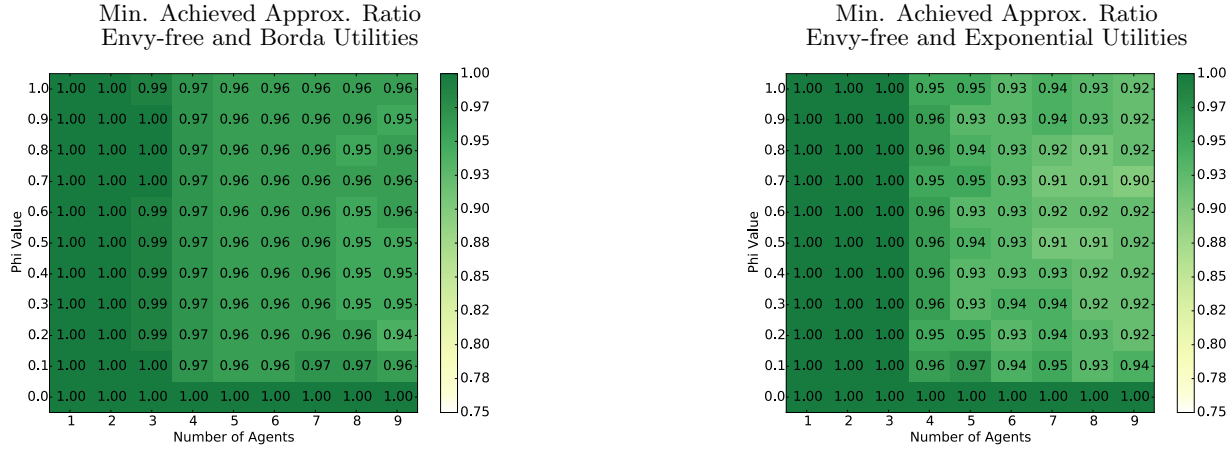


Figure 4: Minimum achieved approximation ratio when we enforce envy-freeness as a hard constraint. For each $n = \{1, \ldots, 9\}$ these graphs are aggregated over the complete range of objects (i.e., all $m \in \{2, \ldots, 9\}$). For example, the cell $(n = 4, \phi = 0.2)$ is the minimum (resp. average) approximation ratio achieved over all instances where m ranges over $\{1, \ldots, 9\}$.

able approximation ratio was 0.87, significantly higher than the theoretical worst case. This shows that for smaller instances and some standard utility models, the requirement of envy-freeness does not have a significant negative impact on the achievable approximation ratio. To get an understanding of the performance of OEEF in a larger parameter space we repeated the experiments from the previous section here, evaluating the performance of OEEF across a large parameter space with number of agents $n \in \{2, \ldots, 9\}$, number of objects $m \in \{2, \ldots, 9\}$, and dispersion parameter $\phi \in \{0.0, 0.1, \ldots, 1.0\}$. The results of these tests, again aggregated by m and ϕ, are shown in Figure 4.

When agents have exponential utilities, the achieved approximation ratio, much like in the last section, is strictly worse. Additionally, when we have exponential utilities, as ϕ increases, the approximation ratio for the envy-free mechanisms first decreases slightly and then increases for higher number of agents. Since $\phi = 1.0$ means that agents preferences are drawn uniformly at random, it is more likely that each agent has high valuation for different objects. Hence, as the preferences move from concentrated to dispersed, there seems to be an interesting transition from high to low and back to high in terms of the achievable approximation ratio. As in the previous subsection, we observe that when all agents have the same preferences, the uniform allocation is both envy-free and has maximal achieved approximation ratio. Hence, when $\phi = 0$, the ratio is 1.0 (not shown in Figure 4). We note that RSD performs much more poorly across the board compared to OEEF. The results in Figure 4 strictly dominate the results for PS. Hence, SD envy-freeness that is satisfied by PS has a significant impact on the achieved approximation ratio.

5. CONCLUSION

We present theoretical and experimental results concerning how well different randomized mechanisms approximate the optimal egalitarian value. It has been well-known that egalitarianism can be incompatible with envy-free or truthfulness. In this paper, we quantified how much egalitarianism is affected by such properties. In a recent paper, Christodoulou et al. [14] proved results for the utilitarian welfare of the Nash equilibria of assignment mechanisms. It will be interesting to adopt a similar approach with respect to the egalitarian value and study the price of anarchy of randomized mechanisms with respect to that objective. To conclude, we mention an open problem: what is the *best* OEV approximation guaranteed by truthful mechanisms?

REFERENCES

[1] A. Asadpour and A. Saberi. An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM Journal on Computing*, 39(7):2970–2989, 2010.

[2] H. Aziz. Random assignment with multi-unit demands. Technical Report 1401.7700, arXiv.org, 2014.

[3] H. Aziz and C. Ye. Cake cutting algorithms for piecewise constant and piecewise uniform valuations. In *Proc. of 10th WINE*, pages 1–14, 2014.

[4] H. Aziz, F. Brandt, and M. Brill. The computational complexity of random serial dictatorship. *Economics Letters*, 121(3):341–345, 2013.

[5] H. Aziz, S. Gaspers, S. Mackenzie, and T. Walsh. Fair assignment of indivisible objects under ordinal preferences. In *Proc. of 13th AAMAS Conference*, pages 1305–1312, 2014.

[6] I. Bezáková and V. Dani. Allocating indivisible goods. *SIGecom Exchanges*, 5(3):11–18, 2005.

[7] A. Bhalgat, D. Chakrabarty, and S. Khanna. Social welfare in one-sided matching markets without money. In *Proceedings of APPROX-RANDOM*, pages 87–98, 2011.

[8] A. Bogomolnaia and E. J. Heo. Probabilistic assignment of objects: Characterizing the serial rule. *Journal of Economic Theory*, 147:2072–2082, 2012.

[9] A. Bogomolnaia and H. Moulin. A new solution to the random assignment problem. *Journal of Economic Theory*, 100(2):295–328, 2001.

[10] A. Bogomolnaia and H. Moulin. Random matching under dichotomous preferences. *Econometrica*, 72(1):257–279, 2004.

[11] S. Bouveret and M. Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. In *Proc. of 13th AAMAS Conference*, pages 1321–1328. IFAAMAS, 2014.

[12] I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, and M. Kyropoulou. The efficiency of fair division. *Theory of Computing Systems*, 50(4):589–610, 2012.

[13] Y.-K. Che and F. Kojima. Asymptotic equivalence of probabilistic serial and random priority mechanisms. *Econometrica*, 78(5):1625–1672, 2010.

[14] G. Christodoulou, A. Filos-Ratsikas, S. K. S. Frederiksen, P. W. Goldberg, J. Zhang, and J. Zhang. Welfare ratios in one-sided matching mechanisms. Technical report, arXiv.org, 2015.

[15] R. Cole, V. Gkatzelis, and G. Goel. Positive results for mechanism design without money. In *Proc. of 12th AAMAS Conference*, pages 1165–1166. IFAAMAS, 2013.

[16] S. Demko and T. P. Hill. Equitable distribution of indivisible objects. *Mathematical Social Sciences*, 16:145–158, 1988.

[17] J. P. Dickerson, A. D. Procaccia, and T. Sandholm. Price of fairness in kidney exchange. In *Proceedings of the 13th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1013–1020, 2014.

[18] A. Filos-Ratsikas, S. K. S. Frederiksen, and J. Zhang. Social welfare in one-sided matchings: Random priority and beyond. In *Proc. of 7th International Symposium on Algorithmic Game Theory (SAGT)*, pages 1–12, 2014.

[19] N. Garg, T. Kavitha, A. Kumar, K. Mehlhorn, and J. Mestre. Assigning papers to referees. *Algorithmica*, 58:119–136, 2010.

[20] M. Guo and V. Conitzer. Strategy-proof allocation of multiple items without payments or priors. In *Proc. of 9th AAMAS Conference*, pages 881–888. IFAAMAS, 2010.

[21] L. Han, C. Su, L. Tang, and H. Zhang. On strategy-proof allocation without payments or priors. In *Proc. of 7th WINE*, LNCS, pages 182–193, 2011.

[22] H. Hosseini, K. Larson, and R. Cohen. Random serial dictatorship versus probabilistic serial rule: A tale of two random mechanisms. Technical Report 1503.01488, arXiv.org, 2015.

[23] A. Hylland and R. Zeckhauser. The efficient allocation of individuals to positions. *The Journal of Political Economy*, 87(2):293–314, 1979.

[24] T. Kavitha, J. Mestre, and M. Nasre. Popular mixed matchings. *Theoretical Computer Science*, 412(24): 2679–2690, 2011.

[25] F. Kojima. Random assignment of multiple indivisible objects. *Mathematical Social Sciences*, 57(1):134–142, 2009.

[26] D. Kurokawa, A. D. Procaccia, and N. Shah. Leximin allocations in the real world. Technical report, 2015.

[27] R. J. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proc. of 5th ACM-EC Conference*, pages 125–131. ACM Press, 2004.

[28] T. Lu and C. Boutilier. Learning Mallows models with pairwise preferences. In *Proc. of 28th ICML*, pages 145–152, 2011.

[29] C. L. Mallows. Non-null ranking models. *Biometrika*, 44(1/2):114–130, 1957.

[30] N. Mattei. Empirical evaluation of voting rules with strictly ordered preference data. In *Proc. of 2nd ADT*, pages 165–177. Springer-Verlag, 2011.

[31] N. Mattei and T. Walsh. PrefLib: A library for preference data. In *Proc. of 3rd ADT*, volume 8176 of *LNCS*, pages 259–270. Springer, 2013.

[32] H. Moulin. *Fair Division and Collective Welfare*. The MIT Press, 2003.

[33] A. Popova, M. Regenwetter, and N. Mattei. A behavioral perspective on social choice. *Annals of Mathematics and Artificial Intelligence*, 68(1–3):135–160, 2013.

[34] A. D. Procaccia and M. Tennenholtz. Approximate mechanism design without money. *ACM Transactions on Economics and Computation*, 1(4), 2013.

[35] A. D. Procaccia and J. Wang. Fair enough: Guaranteeing approximate maximin shares. In *Proc. of 15th ACM-EC Conference*, pages 675–692. ACM Press, 2014.

[36] J. Rawls. *A Theory of Justice*. Harvard University Press, 1971.

[37] L.-G. Svensson. Strategy-proof allocation of indivisible goods. *Social Choice and Welfare*, 16(4):557–567, 1999.

[38] M. U. Ünver, O. Kesten, M. Kurino, T. Hashimoto, and D. Hirata. Two axiomatic approaches to the probabilistic serial mechanism. *Theoretical Economics*, 9: 253–277, 2014.

[39] V. V. Vazirani. Combinatorial algorithms for market equilibria. In N. Nisan, T. Roughgarden, É. Tardos, and V. Vazirani, editors, *Algorithmic Game Theory*, chapter 5, pages 103–134. Cambridge University Press, 2007.

[40] L. Zhou. On a conjecture by Gale about one-sided matching problems. *Journal of Economic Theory*, 52 (1):123–135, 1990.

Fair Social Choice in Dynamic Settings

Rupert Freeman
Duke University
rupert@cs.duke.edu

Seyed Majid Zahedi
Duke University
zahedi@cs.duke.edu

Vincent Conitzer
Duke University
conitzer@cs.duke.edu

ABSTRACT

We study a dynamic social choice problem in which an alternative is chosen at each round according to the reported valuations of a set of agents. In the interests of obtaining a solution that is both efficient and fair, we aim to maximize the *Nash social welfare*, which is the product of all agents' utilities. We present three novel rules and discuss some of their properties. Two are greedy algorithms and the third attempts to explicitly learn the distribution over inputs, updating its decisions by solving a convex program at each round. We also take a more generally applicable algorithm from existing literature and apply it to our problem. Finally, we compare all four algorithms against the offline optimal solution in simulations.

1. INTRODUCTION

Fairness is a topic of rapidly increasing interest in social choice. On the one hand, there has been much recent interest in the fair allocation of resources—cake cutting [25] as well as other models [14, 22]. On the other hand, in voting, fairness considerations have received attention in selecting a committee of candidates, in the form of a focus on the voters being *represented* in the committee [10, 20, 8].

A classical approach to obtaining a fair outcome in a context where agents have utility functions is to maximize the *Nash social welfare* [21], which is the product of the agents' utilities. One attractive feature of using the Nash social welfare is scale invariance: if an agent doubles all her utilities (or, equivalently, changes the units in which she expresses her utilities), this does not change which outcomes maximize the objective.

In life, it is often difficult to make a completely fair decision in a single-shot context; often, every option will leave some agents unhappy. Fortunately, we can often address this over time—we will go to my most preferred restaurant today, and to yours next week. Achieving fairness over time is the topic of our paper. Ours is certainly not the first work to consider fairness or social choice in dynamic settings; see, for example, [24, 16, 6].

When we make multiple decisions over time, we could simply maximize the Nash welfare in each round separately. But it is easy to see that this can lead to dominated outcomes. For example, suppose there are two agents, and we can choose an alternative that gives one a reward of 3, and the other a reward of 0; or vice versa; or an alternative that gives each of them 1. Within a round, the last alternative maximizes Nash welfare; but if this scenario is repeated every round, then it would be better to alternate between the first two alternatives, so that each agent obtains 1.5 per round on average. Of course, *initially*, say in the first round, we may not realize we will have these options every round, and so we may choose the last alternative; but if we do have these options every round, we should *eventually* catch on to this pattern and start alternating. Ideally, we would maximize the long-term Nash welfare, that is, the product of the long-run utilities (which are the sums of each agent's rewards), rather than, for example, the sum of the products within the rounds. Of course, if there is uncertainty about the options that we will have in future rounds, we cannot expect to get the same Nash welfare that we could obtain with perfect foresight. For example, we may choose to make an agent happy this round, and then later realize that in typical rounds, this agent is very easy to make happy and we should have focused our efforts on an agent that is more difficult to make happy. While such scenarios are inevitable, we do want to adapt and learn over time and thereby approximate the ideal Nash welfare.

In this work, we do not focus primarily on strategic concerns (though we discuss this in more detail in Section 7). Of course it is fairly common to ignore strategic concerns in the social choice literature, but we do think this is an important topic for future work. On the other hand, there are also important contexts where strategic concerns do not come into play. For example, instead of considering a setting where there are multiple *agents* that have different utility functions, we can consider a setting where there are multiple *objectives* that each alternative contributes towards. For example, consider faculty hiring. Suppose the three objectives that we want our faculty hires to contribute to are research, teaching, and service; moreover, suppose that at the time of hiring we can predict well how much each candidate would contribute to each of these objectives, if hired. Then, it stands to reason that, one year, we may hire a top researcher that we do not expect to contribute much to our teaching or service objectives. But we would be loath to make such a decision *every* year; having hired a few top researchers who are not good at teaching or service, pressure will mount to address needs in the latter two. This fits well into our framework, if we simply treat each of the three objectives as an agent that is "happy" with an alternative to the extent to which it addresses the corresponding objective. In particular, note that the fact that objectives are measured in incomparable units – for example, we might measure research crudely by number of top-tier publications, and teaching crudely by course evalation scores – poses no problem to our methodology, since this methodology can anyway address agents measuring their utilities in different units. (Since we are not in a setting with a numeraire, there is no reason their utilities should

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

have similar units.) Thus, a reader who insists on game-theoretic modeling in the case of agents with utility functions may instead substitute this modified interpretation of addressing multiple objectives everywhere in our paper.

The rest of the paper is organized as follows. In Section 2 we introduce notation and preliminaries. In Section 3 we present two simple greedy algorithms for choosing alternatives, and provide intuitive interpretations of them. We make a computational distinction between them and provide an axiomatic justification for one of them. In Section 4 we present an algorithm which can be seen as an approximation to the optimal solution when T is infinite. In Section 5 we present an existing algorithm designed for a more general class of stochastic optimization problems with good regret guaranatees.

After presenting the algorithms, we evaluate them on simulated data in Section 6. Finally, in Section 7 we discuss some strategic considerations in the repeated setting, and in Section 8 we discuss specific applications of our methodology, including to voting.

Related work: Parkes and Procaccia [24] examine a similar problem by modeling agents' evolving preferences with Markov Decision Processes, with a reward function defined over states and actions (alternatives). However, their goal is to maximize the sum of (discounted) rewards and they do not explicitly consider fairness as an objective. Kash, Procaccia and Shah [16] examine a model of dynamic fair division where agents arrive at different points in time and must be allocated resources; however, they do not allow for the preferences of agents to change over time as we do. A recent paper by Aleksandrov et al. [6] considers an online fair division problem in a setting where items appear one at a time, and agents declare yes/no preferences over that item. In our setting, each round has many alternatives and we allow agents to express more general utilities. Our work is related to the literature on dynamic mechanism design (see, e.g., [23] for an overview), except that we do not consider monetary transfers. Guo, Conitzer and Reeves [15] consider a setting similar to ours, also without money, except that they are not explicitly interested in fairness, only welfare, and focus on incentive compatibility.

2. PRELIMINARIES

Consider a set of n agents and let A be a set of m possible alternatives.[1] At every round $t = 1, \ldots, T$, agents report their valuation for every alternative. In this paper we allow the valuations to be integers in the range 0 to K for some finite K (therefore we can achieve arbitrarily fine granularity by allowing K to be large). Thus the input at every round is a matrix $V_t \in \mathbb{Z}^{n\times m}_{\geq 0,\leq K}$. For every round t, a *Dynamic Social Choice Function (DSCF)* chooses a single alternative, corresponding to a column of V_t, which we denote by v_t. Importantly, the problem is *online*, so we may only use information up to time t in order to choose v_t.

The valuation of agent i for the alternative j at time t is $V_t(i, j)$, and at each round we can think of an agent's valuation vector, $V_t(i, \cdot)$, as their reported valuation for each alternative. Although the columns of V_t are formally indexed by alternatives, we will often refer to the vector $V(\cdot, j)$ simply as j when there is no risk of confusion. Thus the valuation of agent i for alternative v_t will be denoted by $v_t(i)$. We define a vector of *accrued rewards at round* t, u_t, where the accrued reward of agent i at round t is the sum of agent i's valuations for the chosen alternatives up to and including round t, $u_t(i) = \sum_{t'=1}^{t} v_{t'}(i)$. We will most often be interested in an agent's accrued reward *before* the start of round t, $u_{t-1}(i)$. The average utility of the agents over the first t rounds is given by $v_t^{\text{ave}} = \frac{1}{t}u_t$.

A DSCF is *anonymous* if applying permutation σ to the rows of V_t, for all t, does not change the chosen alternative v_t, for any t. A DSCF is *neutral* if applying permutation σ to the columns of V_t, for all t, results in choosing alternative $\sigma(v_t)$ for all t. For the rest of this paper we only consider anonymous, neutral DSCFs.

The DSCFs that we discuss are presented in a way that naturally allows ties between alternatives. We think of the mechanisms choosing a set of possible alternatives, and then choosing a single alternative from the set arbitrarily.

The *Nash social welfare (NSW)* of utility vector v, $NSW(v)$, is defined to be the product of the agents' utilities,

$$NSW(v) = \prod_{i=1}^{n} v(i). \tag{1}$$

The NSW is frequently used as an objective in the fair division literature as it strikes a balance between maximizing efficiency and fairness (for recent examples in the computer science literature, see [11, 13, 26]). One further nice property of NSW is that it is *scale-free*, meaning that the optimal choice of alternative is unchanged if some agent(s) report valuations on different scales from others. Our aim is to maximize the NSW of the average utility across all T rounds, $NSW(v_T^{ave})$. Some of our algorithms involve the use of convex programming, which requires a concave objective function to maximize. Unfortunately, NSW is not a concave function, however $\ln(NSW)$ is. Thus, we will interchangeably talk about maximizing

$$\ln(NSW(v_t^{\text{ave}})) = \ln\left(\prod_{i=1}^{n} v_t^{\text{ave}}(i)\right) = \sum_{i=1}^{n} \ln(v_t^{\text{ave}}(i)).$$

Since $\ln$ is an increasing function, the solution maximizing $\ln NSW(v)$ is the same as the solution maximizing $NSW(v)$.

The benchmark algorithm is the opfimal offline algorithm, where an offline instance of the problem is given by the set of matrices $\{V_t\}_{t\in\{1,\ldots,T\}}$. The offline problem can be solved via the following mixed integer convex program:

$$\text{Maximize} \sum_{i=1}^{n} \ln\left(\sum_{t=1}^{T}\sum_{j=1}^{m} x_{tj} V_t(i,j)\right) \tag{2}$$
$$\text{subject to} \sum_{j=1}^{m} x_{tj} = 1 \quad \forall t, \qquad x_{tj} \in \{0,1\} \quad \forall t, j$$

where x_{tj} is a binary variable denoting whether or not alternative j is chosen at time t. The constraint simply says that for each t, we must choose exactly one alternative. We denote the optimal value achieved by convex program 2 by OPT (thus the optimal Nash social welfare is e^{OPT}).

3. GREEDY ALGORITHMS

In this section we present two simple greedy algorithms. The first algorithm, GREEDY, simply chooses v_t to maximize $NSW(v_t^{ave})$. The other, LINEARGREEDY, is a linear version of GREEDY which assigns each agent a weight w_i equal to the inverse of their accrued utility at the start of each round and simply chooses $v_t = \text{argmax}_{v\in V_t}\, w \cdot v$. The advantage of these algorithms lies in their simplicity to understand and execute.

[1] For simplicity of presentation, we define the set of alternatives to be static. However, all of our algorithms and results hold if the set of alternatives, and even the number of alternatives, changes from round to round.

One challenge that these algorithms face is that in the early rounds it may not be possible to give all agents non-zero utility. Therefore it may be the case that $u(v_t^{ave}) = 0$ for all choices of v_t, even when one allocation is clearly better than all others. We address this by allocating some small 'hallucinated' utility to those agents with zero accrued reward (not necessarily the same to each agent), which is removed once the agent accrues some positive reward. The algorithms are shown as Algorithms 1 and 2. For Algorithm 1, let the parameter x be such that $0 < x < \frac{1}{2^n(K+1)^{n(n+1)}}$.

Algorithm 1 GREEDY

1: **for** $i = 1, \ldots, n$ **do**
2: Set $x \leq \epsilon_i \leq 2(K+1)^{n-1}x$
3: **end for**
4: Initialize $u_0 = (0, \ldots, 0)$
5: **for** $t = 1, \ldots, T$ **do**
6: Choose $v_t \in \text{argmax}_{v \in V_t} \prod_{i=1}^{n} (\max\{u_{t-1}(i) + v(i), \epsilon_i\})$
7: $u_t = u_{t-1} + v_t$
8: **end for**

Algorithm 2 LINEARGREEDY

1: **for** $i = 1, \ldots, n$ **do**
2: Set $0 < \delta_i < \frac{1}{nK}$
3: **end for**
4: Initialize $u_0 = (0, \ldots, 0)$
5: **for** $t = 1, \ldots, T$ **do**
6: Set $w_i = \frac{1}{\max\{\delta_i, u_{t-1}(i)\}}$ for all i
7: Choose $v_t \in \text{argmax}_{v \in V_t} w \cdot v$
8: $u_t = u_{t-1} + v_t$
9: **end for**

We first state a lemma which follows easily from the choice of ϵ_i.

LEMMA 3.1. *For all j, k such that $1 \leq k < k+j \leq n$, and all sets of agents I and I', of size $k+j$ and k respectively,*

$$(K+1)^{n-k-j} \prod_{i \in I} \epsilon_i < \prod_{i' \in I'} \epsilon_{i'}.$$

PROOF. Let $1 \leq k < k+j \leq n$. Let $I = \{i_1, \ldots, i_{k+j}\}$ and $I' = \{i'_1, \ldots, i'_k\}$. Recall that $0 < x < \frac{1}{2^n(K+1)^{n(n+1)}}$ and $x \leq \epsilon_i \leq 2(K+1)^{n-1}x$ for all i. Then

$$\begin{aligned}
(K+1)^{n-k-j} \prod_{i \in I} \epsilon_i &\leq (K+1)^{n-k-j}(2(K+1)^{n-1}x)^{k+j} \\
&< x^k (K+1)^{n-k-j}(2(K+1)^{n-1})^{k+j} \cdot \\
&\quad \left(\frac{1}{2^n(K+1)^{n(n+1)}}\right)^j \\
&\leq x^k \frac{2^n(K+1)^{(k+j+1)n-2k-2j}}{(2^n(K+1)^{n(n+1)})^j} \\
&\leq x^k \frac{2^n(K+1)^{(k+j+1)n-2k-2j}}{2^n(K+1)^{n(n+1)}} \\
&< x^k \frac{(K+1)^{n(k+j+1)}}{(K+1)^{n(n+1)}} \\
&\leq x^k \\
&\leq \prod_{i' \in I'} \epsilon_{i'}
\end{aligned}$$

□

We next state a simple interpretation of GREEDY. Let $NSW^+(v)$ be the product of all non-zero entries in v.

PROPOSITION 3.2. *At every round,* GREEDY *selects an alternative to maximize the number of agents with $u_t(i) > 0$. Subject to this condition, and holding fixed the set of agents with non-zero utility,* GREEDY *chooses an alternative which maximizes $NSW^+(v_t^{ave})$.*

PROOF. Consider the action of GREEDY at round t. Suppose first that all agents have $u_{t-1}(i) > 0$. Then the first condition in the proposition statement is vacuous; all choices maximize the number of agents with $u_t(i) > 0$. Since $\epsilon_i < 1$ for all i, the choice at Line 6 is precisely to maximize $NSW(v_t^{ave}) = NSW^+(v_t^{ave})$.

Now suppose that $u_{t-1}(i) = 0$ for some agent i, and consider two alternatives v' and v. Suppose that $|I' = \{i : u_{t-1}(i) + v'(i) = 0\}| = k$ and $|I = \{i : u_{t-1}(i) + v(i) = 0\}| = k + j > k$. To show that GREEDY maximizes the number of agents with $u_t(i) > 0$, it suffices to show that the product in Line 6 is larger when v' is chosen than when v is chosen.

We consider the case in which the produc tin Line 6 is greatest for v (compared to v'). In particular, all $n-k$ agents with $u_{t-1}(i)+v'(i) > 0$ have $u_{t-1}(i) + v'(i) = 1$, and all $n-k-j$ agents with $u_{t-1}(i) + v(i) > 0$ have $u_{t-1}(i) + v(i) = K+1$ (if $u_{t-1}(i) + v(i) > K+1$ then $u_{t-1}(i) + v'(i) \geq u_{t-1}(i) > 1$, since no single-round valuation is greater than K). Then the product on Line 6 for v' is $\prod_{i' \in I'} \epsilon_{i'}$, and for v is $(K+1)^{n-k-j} \prod_{i \in I} \epsilon_i$. By Lemma 3.1, the former is greater than the latter, and GREEDY chooses v', thus minimizing the number of agents with $u_t(i) = 0$.

Finally, suppose that GREEDY chooses v_t when there exists another alternative v which results in the same set of agents with $u_t(i) > 0$ (call this set of agents I), and $NSW^+(u_{t-1} + v_t) < NSW^+(u_{t-1} + v)$. Then, by definition of NSW^+,

$$\begin{aligned}
&\prod_{i \in I}(u_{t-1}(i) + v_t(i)) < \prod_{i \in I}(u_{t-1}(i) + v(i)) \\
\iff &\prod_{i \in I}(u_{t-1}(i) + v_t(i)) \prod_{i \notin I} \epsilon_i < \prod_{i \in I}(u_{t-1}(i) + v(i)) \prod_{i \notin I} \epsilon_i \\
\iff &\prod_{i=1}^{n} \max\{(u_{t-1}(i) + v_t(i)), \epsilon_i\} < \prod_{i=1}^{n} \max\{(u_{t-1}(i) + v(i)), \epsilon_i\},
\end{aligned}$$

which contradicts the choice of v_t. □

Indeed, we can show that every alternative not ruled out by Theorem 3.2 can be chosen by GREEDY, for some choice of $\{\epsilon_i\}$.

THEOREM 3.3. *Suppose alternative a maximizes the number of agents with $u_t(i) > 0$. Suppose further that for all j such that choosing j results in the same set of agents with non-zero accrued reward, $NSW^+(u_{t-1} + V_t(\cdot, j)) \leq NSW^+(u_{t-1} + V_t(\cdot, a))$. Then a is chosen by* GREEDY *for some choice of $\{\epsilon_i\}$.*

PROOF. Let $a \in V_t$ satisfy the conditions of the theorem statement. We exhibit a set of $\{\epsilon_i\}$ such that a is chosen by GREEDY. Let I be the set of agents with non-zero accrued reward after choosing a. Then let $\epsilon_i = x$ (the same x as in Line 2 of Algorithm 1) for all $i \in I$, and $\epsilon_{i'} = 2(K+1)^{n-1}x$ for all $i' \notin I$. Consider some alternative j that also maximizes the number of agents with non-zero accrued reward (since otherwise, by Proposition 3.2, it would certainly not be chosen by GREEDY), and denote by J the set of agents given non-zero accrued utility after choosing j.

Suppose that $J \neq I$. By assumption, $|J| = |I|$. Therefore $|I \backslash J| = |J \backslash I|$. Let us consider the contribution of all agents to the

product in Line 6 of Algorithm 1 for alternatives j and a. There are four types of agent to consider:

1. An agent $i \in I \backslash J$. These are agents for which $u_{t-1}(i) = 0$, with $V_t(i, a) \geq 1$ and $V_t(i, j) = 0$. Therefore each of these agents contributes a factor of at least 1 to the product for a and $\epsilon_i = x$ to the product for j.

2. An agent $i \in I \cap J$. These are agents for which the choice of either a or j both result in $u_t(i) > 0$. In the worst case for a (relative to j), $u_{t-1}(i) = 1$ with $V_t(i, a) = 0$ and $V_t(i, j) = K$. That is, each of these agents contributes a factor of at least 1 to the product for a and at most $K + 1$ to the product for j.

3. An agent $i \in J \backslash I$. These are agents for which $u_{t-1}(i) = 0$, with $V_t(i, a) = 0$ and $V_t(i, j) \geq 1$. Each of these agents contributes a factor of $\epsilon_i = 2(K+1)^{n-1}x$ to the product for a and at most K to the product for j.

4. An agent $i \notin I \cup J$. These are agents for which the choice of either a or j both result in $u_t(i) = 0$. Thus they contribute exactly the same factor to the product for both a and j.

Let us write down the product from Line 6 for a (not counting the agents of type 4 which make the same contribution to both). It is at least:

$$(2(K+1)^{n-1}x)^{|J\backslash I|} = (2(K+1)^{n-1}x)^{|I\backslash J|} \tag{3}$$

The product for j (not counting the agents of type 4 which make the same contribution to both) is at most:

$$\begin{aligned} x^{|I\backslash J|} \cdot (K+1)^{|I\cap J|} \cdot K^{|J\backslash I|} &\leq x^{|I\backslash J|} \cdot (K+1)^{|I\cap J|} \cdot (K+1)^{|J\backslash I|} \\ &= x^{|I\backslash J|} \cdot (K+1)^{|I\cap J|} \cdot (K+1)^{|I\backslash J|} \end{aligned} \tag{4}$$

Noting that $|I \cap J| + |I \backslash J| \leq n-1$ (since $|I \cap J| + 2|I \backslash J| \leq n$ and $I \backslash J$ is nonempty), it is clear that expression 3 is greater than expression 4. Therefore the product in Line 6 is higher for a than for j.

Finally, suppose that $J = I$. Then, by the condition of the theorem, $NSW^+(u_{t-1} + j) \leq NSW^+(u_{t-1} + a)$, which implies that choosing a results in a weakly higher product in Line 6 than choosing j.

Therefore a is chosen by GREEDY, since all other alternatives which maximize the number of agents with non-zero accrued reward after round t have been ruled out, given our particular choice of ϵ. □

Unlike GREEDY, LINEARGREEDY may leave some agents with zero utility even when it was possible to give positive utility to all agents.

EXAMPLE 3.4. *Let $n = 2$, $m = 3$, and suppose that $V_1 = \left(\begin{smallmatrix} 3 & 0 & 1 \\ 0 & 3 & 1 \end{smallmatrix}\right)$. The columns represent alternatives a_1, a_2, and a_3 respectively, and the rows represent agents i_1 and i_2 respectively.*

For any choice of $0 < \epsilon_1, \epsilon_2 << 1$, GREEDY *chooses a_3 since $3\epsilon_i < 1$. However,* LINEARGREEDY *assigns the agents weights w_1, w_2 and chooses $\text{argmax}_{v \in \{a_1, a_2, a_3\}} w \cdot v$. Since it must be the case that either $3w_1 > w_1 + w_2$ or that $3w_2 > w_1 + w_2$, it is not possible for a_3 to be chosen by* LINEARGREEDY *even though it is the only alternative which gives both agents positive utility.*

We can, however, provide a weaker guarantee for LINEARGREEDY.

PROPOSITION 3.5. LINEARGREEDY *always chooses an alternative v_t with $v_t(i) > 0$ for at least one agent i with $u_{t-1}(i) = 0$, if such an alternative exists.*

PROOF. Let i be an agent with $u_{t-1}(i) = 0$ and v be an alternative with $v(i) \geq 1$. Then the weight assigned to i at round t by LINEARGREEDY is $w_i = \frac{1}{\delta_i} \geq nK$, and the dot product in Line 7 of Algorithm 2 is at least nK. Suppose for contradiction that LINEARGREEDY chooses an alternative v' such that $v'(i) > 0$ only for agents with $u_{t-1}(i) > 0$. The weight of each such agent is $\frac{1}{u_{t-1}(i)} \leq \frac{1}{1} = 1$, and $v'(i) \leq K$, thus the dot product in Line 7 is at most $(n-1)K < nK$. Thus v' is not chosen by LINEARGREEDY. □

3.1 Computational Considerations

Clearly, when the number of allocations, m, is not too large, the outcome of both GREEDY and LINEARGREEDY can be computed efficiently. However, consider a setting in which every round is a combinatorial allocation problem, so the number of alternatives is exponential in the number of items being allocated. For instance, if every round is an allocation of food bank items [6] to different charities then we will have substitutes and complements which must be taken into account, thus charities have preferences over subsets of items, not just items themselves. In this setting, computing the chosen alternative under GREEDY is weakly NP-hard even in a very restricted case.

PROPOSITION 3.6. *Computing the chosen alternative v_t under* GREEDY *is weakly NP-hard, even when there are only two agents and each has additive valuations over the items.*

PROOF. Suppose there are two agents with the same valuation over the items in the first round. Then the allocation under GREEDY is to allocate each agent an equal share of the items according to their common valuation (or as close to equal as possible). This is exactly an instance of PARTITION, which is weakly NP-hard. □

Note that for the LINEARGREEDY algorithm, computing the chosen alternative is equivalent to the combinatorial auction winner determination (CAWD) problem, which has been studied extensively [27, 7, 18]. Thus, the outcome under LINEARGREEDY can be computed efficiently under exactly the same conditions as the CAWD problem.[2] Even in those cases where LINEARGREEDY can not be computed efficiently, we can use any existing algorithm for the CAWD problem.

3.2 Axiomatization of LINEARGREEDY

It is possible to simply consider LINEARGREEDY an approximation to GREEDY. However, in this section we provide an axiomatization of the LINEARGREEDY mechanism which provides some justification for seeing it as a worthwhile rule in and of itself, without needing to appeal to its approximation of GREEDY.

A DSCF is scale-free if it is not affected by a uniform (multiplicative) scaling of some agent's valuations. This property is desirable because it means we do not require any sort of agreement or synchronization as to the units of measurement used by the agents in their reporting.

DEFINITION 3.7. *Let $c > 0$. Say that a DSCF satisfies* scale-free-ness (SF) *if the chosen alternative at round t is still among the set of (possible) chosen alternatives if we replace $v(i)$ by $c \cdot v(i)$ for all $v \in V_t$ for every $t = 1, \ldots, T$.*

[2]For example, when agents have preferences over bundles of size at most 2, the problem is in P.

LEMMA 3.8. LINEARGREEDY *satisfies SF.*

PROOF. Suppose that agent i scales all their valuations by $c > 0$. We show by induction that there exists some choice of δ such that LINEARGREEDY still chooses the same alternative at each round as it did before the scaling.

Consider the first round, $t = 1$. Let δ' denote the vector of hallucinated utilities chosen before the scaling. When i scales their valuations by factor c, simply set $\delta_i = c \cdot \delta'_i$,[3] and $delta_j = delta'_j$ for all $j \neq i$. Thus the weight of i is scaled by $\frac{1}{c}$, so the value of $w \cdot v$ is unchanged for all $v \in V_t$, and the same alternative is chosen at round 1.

Now consider round $t > 1$, and suppose that the same alternatives are chosen for all rounds before t. In particular, the accrued utility of all agents $i' \neq i$ (or, in the case that i' has zero accrued utility, then the value of δ_i) is the same as before the scaling, so their weights have not changed. But the accrued utility of i has scaled by a factor of c, since i's valuation for every alternative at every round is scaled by c (and, in the case that i has zero accrued utility, we still have that $\delta_i = c \cdot \delta'_i$), so that the weight of i is scaled by $\frac{1}{c}$. Thus, again, $w \cdot v$ is unchanged for all $v \in V_t$ and the same alternative is chosen.

Finally we need to rule out the possibility of there being some setting of δ such that some new alternative, a, is chosen at round t as a result of the scaling that was not previously chosen. But if this were the case, then we can just scale the scaled instance by $\frac{1}{c}$ and return to the original instance where, by the above proof, there is some value of δ such that a is chosen at round t. □

A DSCF is separable into single-minded agents if the chosen alternative at a given round is unchanged by replacing an agent by several new agents with the same accrued utility, each of which has unit positive valuation for only one alternative.

DEFINITION 3.9. *Say that a DSCF is* separable into single-minded agents (SSMA) *if, at round t, the same allocation is chosen if we replace each agent with several new agents according to the following scheme: An agent with valuation vector $V_t(i,:)$ is, for each $j \in \{1, \ldots, m\}$, replaced by $V_t(i,j)$ new agents, each with valuation vector e_j. Each new agent has the same (possibly hallucinated) accrued utility as the original agent it replaces.*

LEMMA 3.10. LINEARGREEDY *satisfies SSMA.*

PROOF. Consider round t with valuation matrix V_t. LINEARGREEDY chooses

$$\operatorname*{argmax}_{j \in V_t} \sum_{i=1}^{n} \left(V_t(i,j) \frac{1}{u_{t-1}(i)} \right)$$

Now suppose that we replace the agents with new agents according to the definition of SSMA. For every agent i and alternative j, we now have $V_t(i,j)$ agents with accrued utility $u_{t-1}(i)$ and valuation vector e_j. The accrued utility, and therefore the weight of each of these agents in the LINEARGREEDY algorithm, is the same as the agent it replaced. Thus the value of the dot product on Line 7 for an alternative j is

$$\sum_{i=1}^{n} \left(V_t(i,j) \frac{1}{u_{t-1}(i)} \right),$$

and LINEARGREEDY chooses the same alternative in each case. □

[3]It is possible that δ_i is now greater than $\frac{1}{nK}$. If this is the case, we can simply divide δ by some small constant, bringing δ_i back into the allowed range and not changing the relative weights of the agents.

The plurality axiom says that if all agent valuation vectors are unit vectors, and we have no reason to distinguish between agents, then the allocation favored by the most agents should be chosen.

DEFINITION 3.11. *Say that an allocation satisfies* plurality (P) *if, when all agents have preferences of the form e_j, and all agents have the same (non-zero) accrued utility, then the chosen alternative is the one with non-zero valuation from the most agents.*

The axiom says nothing about the case when all agents have zero accrued utility. The idea of the axiom is that we should choose the alternative which provides the greatest utility gain, relative to what agents already have. However, in the case that agents have zero accrued utility, it is not possible to make accurate comparisons as to the relative benefit each agent receives.

OBSERVATION 3.12. LINEARGREEDY *satisfies plurality.*

We now show that any mechanism that achieves SF, SSMA, and P simultaneously must agree with LINEARGREEDY, provided that all accrued rewards are non-zero.

THEOREM 3.13. *Suppose that $u_{t-1}(i) > 0$ for all i. Denote by J_t the set of all alternatives that may be chosen by* LINEARGREEDY *at time t. If a DSCF satisfies SF, SSMA, and P then it must choose some alternative from J_t at time t.*

PROOF. We have already shown that LINEARGREEDY satisfies SF, SSMA, and P.

Suppose that all agents have $u_{t-1}(i) > 0$. Let M be a DSCF that satisfies all three axioms simultaneously. We show that M's choice of alternative is the same as one chosen by LINEARGREEDY.

Without loss of generality, let $u_{t-1}(i) = u$ for all agents i. We may assume this because, by SF, M would choose the same allocation at round t (and all previous rounds) even if the valuation vectors of some agent(s) were multiplied by a constant across all rounds. So, if $u_{t-1}(i) \neq u_{t-1}(j)$, we can transform the instance to one in which all agents have the same accrued reward by multiplying agent i's valuations by $\prod_{j \neq i} u_{t-1}(j)$ for all i. Then all agents have the same acccrued utility, $\prod_i u_{t-1}(i)$.

By SSMA, we can replace the agents with $\sum_{j=1}^{m} V_t(i,j)$ agents, such that $V_t(i,j)$ of them have valuation vector e_j for all $j \in \{1, \ldots, m\}$, all with accrued reward u. Then, by plurality, the chosen allocation is

$$\operatorname*{argmax}_{j \in V_t} \sum_{i=1}^{n} V_t(i,j). \tag{5}$$

But note that LINEARGREEDY assigns equal weight w_i to each agent since $u_{t-1}(i) = u_{t-1}(j)$ for all i, j. Thus LINEARGREEDY chooses precisely the alternatives which maximize Equation 5. □

4. DISTRIBUTIONAL UPDATE ALGORITHM

So far we have assumed nothing about the way that the input matrices are drawn. In this section, we will assume that there is some distribution, D, over $\mathbb{Z}^{n \times m}_{\geq 0, \leq K}$ from which matrices are drawn i.i.d at each round.[4]

Suppose first that we know D, and that $T = \infty$. Then the optimal solution is defined by a policy: when $V_t = v$, choose allocation j with probability x_{vj}. We simply need to choose values for $\{x_{vj}\}$

[4]In practice, this algorithm may be suitable when we believe the distribution of inputs to be somewhat stable over time.

in order to maximize $\mathbb{E}(NSW(v_t^{ave}))$, as $t \to \infty$. We can compute these variables by the following convex program:

$$\text{maximize} \sum_{i=1}^{n} \log\Big(\sum_{v \in \mathbb{Z}^{n\times m}_{\geq 0, \leq K}} \sum_{j=1}^{m} Pr(V_t = v) x_{vj} v(i,j) \Big) \quad (6)$$

$$\text{subject to} \sum_{j \in A} x_{vj} = 1 \quad \forall v \in \mathbb{Z}^{n\times m}_{\geq 0, \leq K}, \qquad x_{vj} \geq 0 \ \ \forall v, j$$

THEOREM 4.1. *The variables x_{vj} computed by convex program 6 define the optimal policy when the distribution D is known and $T = \infty$.*

PROOF SKETCH. Let $\{x_{vj}\}$ be the optimal solution to convex program 6. We show that $\{x_{vj}\}$ converges to the optimal solution for the offline problem in the case that $T \to \infty$. So consider an offline instance for some large T, large enough that every matrix v that occurs with non-zero probability in D appears a large number of times in the input. For every round that $V_t = v$, choose an alternative by sampling from the distribution x_{vj}. Denote the objective value achieved by S_D. Now consider the observed distribution in the finite instance, O. Consider solving convex program 6 for distribution O, giving variables x^O_{vj}, and denote the value of the resulting solution on the large instance, S_O. It can be shown that $S_O \to S_D$ as $O \to D$. By the law of large numbers, $\lim_{T\to\infty} O = D$, therefore $\lim_{T\to\infty} S_O = S_D$.

Now consider the optimal offline solution as defined by mixed integer program 2. Denote the value it achieves by S_{MIP}. Clearly $S_{MIP} \geq S_O$, since S_{MIP} is optimal. Next, consider the alternatives chosen in the offline solution and use them to define variables

$$x'_{vj} = \#\text{times}(V_t = v \text{ and } j \text{ is chosen}) / \#\text{times}(V_t = v). \quad (7)$$

Denote the value of the solution defined by the x'_{vj} variables by S'. $S' = S_{MIP}$ since sampling from the solution corresponding to S' gives the solution corresponding to S_{MIP}.

Lastly, we show that $\lim_{T\to\infty} S' \leq \lim_{T\to\infty} S_O$. Note that the variables x'_{vj} are a feasible solution to convex program 6. Therefore, as long as the instance is large enough that the probabilities x^O_{vj} can be well-sampled for every v that appears in the instance, S_O is the highest value that can be achieved. As $T \to \infty$, we can sample these variables arbitrarily well. Thus, $\lim_{T\to\infty} S' \leq \lim_{T\to\infty} S_O$.

So, in the limit as $t \to \infty$, we have the relations

$$S_{MIP} = S' \leq S_O = S_D \leq S_{MIP}. \quad (8)$$

Here the inequalities are forced to be equalities, otherwise we get $S_{MIP} < S_{MIP}$. In particular, $\lim_{T\to\infty} S_D = S_{MIP}$. □

Let us now relax the assumption that D is known to the algorithm. In this case, one approach would be to approximately learn the distribution by sampling, then compute the optimal policy according to the learned distribution, and act accordingly for the remaining rounds. We can even continue to update our belief on the distribution as often as we want, re-compute the variables $\{x_{vj}\}$, and choose according to them until we perform another update step. If $T = \infty$, we can learn the distribution arbitrarily well, and behave close to optimally in the long term.

The algorithm we present now uses the same heuristic even when T is finite. We begin with no knowledge of D, but update our belief with every new piece of information V_t, and use the inferred distribution to compute a policy $\{x_{vj}\}$.

Crucially, the update to p_s is done *before* v_t is actually chosen according to $x_{V_t j}$. Were this not the case, the algorithm would not be defined when valuation matrix s appears for the first time.

Algorithm 3 UPDATE

```
for t = 1, ..., T do
    for s ∈ {V_1, V_2, ..., V_t} do
        Let p_s = (number of times s has been realized)/t
    end for
    Solve Convex Program 6 using inferred probabilities p_s
    Randomly draw v_t according to x_{V_t j}
end for
```

EXAMPLE 4.2. *Let $n = m = 2$. Suppose that $V_1 = \left(\begin{smallmatrix}1 & 0\\ 0 & 1\end{smallmatrix}\right)$, where the columns represent alternatives a_1 and a_2 respectively. Then the algorithm updates its belief on D to be that V_1 appears with probability 1, in which case the optimal polcy is to choose a_1 and a_2 with probability 0.5 each. Suppose it randomly chooses a_1. Suppose that $V_2 = \left(\begin{smallmatrix}2 & 0\\ 0 & 1\end{smallmatrix}\right)$. Then the algorithm updates its belief on D to be $0.5V_1 + 0.5V_2$. Given this distribution, the optimal policy is to choose a_1 when V_2 is realized and a_2 when V_1 is realized. Thus, the algorithm chooses $v_2 = a_1$. Now suppose that $V_3 = V_2$. Then the updated belief on D is $\frac{1}{3}V_1 + \frac{2}{3}V_2$. The optimal policy now is to choose a_2 whenever V_1 is realized, and whenever V_2 is realized to draw randomly from $\frac{3}{4}a_1 + \frac{1}{4}a_2$. Thus the algorithm draws an allocation for v_3 from this distribution.*

Observe that, in Example 4.2, from the perspective of the algorithm at $t = 3$, a mistake was made at round 1 by choosing a_1 instead of a_2. As stated, this algorithm does nothing to take the mistake into account. However, one could imagine incorporating a more 'backwards-looking' approach into this algorithm. As a simple example we could, with probability p, simply use GREEDY at round t, which would act to partially compensate for past mistakes. In Example 4.2, GREEDY would choose a_2 to make up for agent 2 not accruing any utility from the first two rounds.

5. STOCHASTIC CONVEX PROGRAMMING APPROACH

A recent paper by Agrawal and Devanur [5] designed algorithms for a general class of problems that encompasses our framework. In their setting, the input is a concave function over a bounded domain R, a convex set $S \subseteq R$, and the goal is to choose a vector v_t at each round so that $f(v_T^{ave})$ is maximized, subject to $v_T^{ave} \in S$. For the setting in our paper, however, there is no constraint, since all input vectors are feasible. That is, $S = R$.

They provide an algorithm using tools from convex optimization which, in our setting, reduces to Algorithm 4. The algorithm assigns a vector of weights, ϕ, to the agents and minimizes the weighted sum of valuations at each round (weights can be negative). Every round, ϕ is updated by an online convex optimization update (the implementation we present uses the gradient descent algorithm to update ϕ).

The initialized variables ϕ and η can be set to any values satisfying the constraints. In our implementation, we set $\phi = -\mathbf{1}$ and $\eta = 0.5$ after some experimentation.

Agrawal and Devanur prove a regret bound on Algorithm 4 of $O\left(\sqrt{\frac{n\log(n)}{T}}\right)$. This is a bound on the *expected regret* when the input matrices appear in a random order. It is *not* a guarantee on any single instance. Therefore, while we would expect good performance from this algorithm on random instances, we may not necessarily expect low regret on instances where the agents' preferences change over time in a structured way. We explore this further in Section 6.

Algorithm 4 STOCHASTIC

Initialize $\phi \in \mathbb{R}^n$, $||\phi||_2 \leq \sqrt{n}$, $\eta > 0$.
for $t = 1, \ldots, T$ **do**
  Choose $v_t = \text{argmin}_{j \in V_t} j \cdot \phi_t$
  if $\phi_t(i) > \frac{-1}{K+1}$ **then**
    Set $\phi_{t+1}(i) = \phi_t(i) - \eta(v_t(i) - K - 1)$
  else if $\phi_t(i) < -1$ **then**
    Set $\phi_{t+1}(i) = \phi_t(i) - \eta(v_t(i) - 1)$
  else
    Set $\phi_{t+1}(i) = \phi_t(i) - \eta(v_t(i) - \frac{1}{\phi_t(i)})$
  end if
  if $||\phi_{t+1}||_2 > \sqrt{n}$ **then**
    Set $\phi_{t+1} = \frac{\sqrt{n}}{||\phi_{t+1}||_2}\phi_{t+1}$
  end if
end for

The regret guarantees provided by Algorithm 4 require that the concave function is Lipschitz continuous on the bounded domain R. Unfortunately, our function, $\log NSW(\cdot)$, is not Lipschitz continuous at 0, which we fix by shifting the agents' valuations to lie in the range $\{1, \ldots, K+1\}$. After solving, we then shift the valuations back before computing the value of the solution found. When K is large, this shift is not too significant. If K was small, we could shift the utilities by something less than 1. However, the Lipschitz constant increases as we allow inputs closer to 0, and this constant appears as a linear factor in the regret bound.

6. EXPERIMENTS

6.1 Simulated Data

We compare the four algorithms discussed in this paper – GREEDY, LINEARGREEDY, STOCHASTIC, and UPDATE– on input data randomly generated from a variety of distributions. As a benchmark we also compute the optimal offline solution for each input using MIP (2).

We consider three input models. The first, *uniform*, has each $V_t(i,j)$ chosen uniformly at random between 0 and 20. The second, *half-half*, draws V_t from a different distribution depending on t. For $t < \frac{T}{2}$, $V_t = \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right)$, where A, B, C, D are submatrices of size $\frac{n}{2} \times \frac{m}{2}$. Entries in A are integers in the range 0 to 25 drawn uniformly at random, entries in B and C are in the range 0 to 5, and entries in D are in the range 0 to 10. For $t \geq \frac{T}{2}$, submatrices A, B, C, D are drawn in the same way but $V_t = \left(\begin{smallmatrix} D & B \\ C & A \end{smallmatrix}\right)$. The third model, *alternating*, sets $V_t = \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right)$ for odd t and $V_t = \left(\begin{smallmatrix} D & B \\ C & A \end{smallmatrix}\right)$ for even t. In both of these latter models, it is almost always optimal to choose an alternative for which half of the valuations are being drawn from the high, 'A', distribution. The other agents can be compensated in a round where they draw from the 'A' distribution.

For every fixed value of n, m, T, and input model that we report, values are averaged over 15 random instances. When not explicitly varied, $n = 20$, $m = 10$, and $T = 40$.

Consider first the runtime comparisons in Figure 1. These simulations are performed on inputs drawn from the half-half model, varying values of n, m, and T seperately. Three of the algorithms, GREEDY, LINEARGREEDY, and STOCHASTIC, take virtually no time to run on the instances we consider. This is not surprising as each makes only a simple comparison between each of the m alternatives, followed by some very simple arithmetic operations to update weights and accrued utility. The UPDATE algorithm is the slowest by far on our simulations, even slower than the MIP for solving the offline problem (although we would expect that for large values of T, the MIP would become slower than UPDATE). We could speed it up by a constant factor of k by only updating the inferred distribution, and values of $x_{V_t j}$, every k rounds, and still expect reasonable results. All of our algorithms scale well with n and m. Runtime results for the other two input models are very similar and we do not present them here.

Turning to the value comparisons in Figure 2, we see that the input model used heavily influences the performance of the algorithms. In these graphs, OPT is normalized to 1, and for each input model we present results only for varying T. The results for varying m and n look very similar.

For the uniform input model, all algorithms perform well, achieving at least 75% of the optimal value. This model provides a relatively simple case for the algorithms; indeed, simply maximizing (additive) welfare at each round is the optimal solution in the limit as T grows. For the half-half distribution, STOCHASTIC is clearly better than all the other algorithms, achieving around 60-70% of OPT compared to less than 30% for the others. This is because the weight vector, ϕ_t, can change quite significantly in the space of just one round. Thus it is not over-burdened by rounds in the past and is able to quickly adapt when the input distribution changes. In fact, in all instances that we examined closely, STOCHASTIC chooses a 'good' alternative for the first $\frac{T}{2}$ rounds, followed by a 'bad' alternative for a single round, and then 'good' alternatives for all remaining rounds.

However, the tendency to take exactly one round to adapt to changing circumstances counts against STOCHASTIC in the alternating model, since by the time it adapts the input distribution has changed again. Here we see the other algorithms performing well, achieving very close to optimal performance, while STOCHASTIC achieves very close to 0. Indeed, this model is tailored to suit UPDATE, which quickly learns to choose an alternative with some 'A' valuations, and the greedy algorithms, since it enables us to alternate which agents get the high valuations, keeping everyone roughly equally well-off at the end of every round.

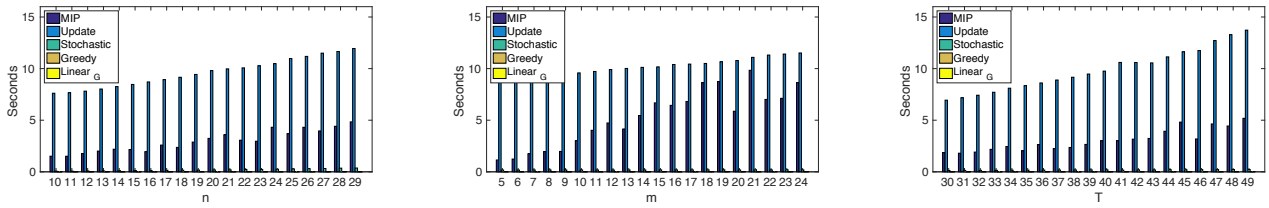
Table 1: Spark Workloads

App	Category	Dataset	Data Size
`Correlation`	Statistics	kdda2010 [28]	2.5G
`DecisionTree`	Classification	kdda2010	2.5G
`FP Growth`	Pattern Mining	Webdocs [19]	1.5G
`GradientBoostedTrees`	Classification	kddb2010 [28]	4.8G
`KMeans`	Clustering	uscensus1990 [3]	327M
`LinearRegression`	Classification	kddb2010	4.8G
`ALS`	Collaborative Filtering	movielens2015 [2]	325M
`NaiveBayesian`	Classification	kdda2010 [28]	2.5G
`SVM`	Classification	kdda2010	2.5G
`Pagerank`	Graph Processing	wdc2012 [4]	5.3G
`ConnectedComponents`	Graph Processing	wdc2012	5.3G
`TriangleCounting`	Graph Processing	wdc2012	5.3G

6.2 Real Data: Power Boost Allocation

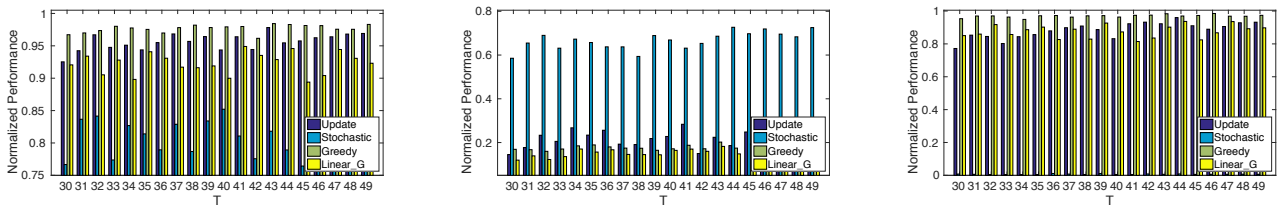
We ran the algorithms on real data gathered from a power boost allocation problem. In this problem, n computer applications are each allocated a base level of power, and compete for $m < n$ additional (indivisible) units of extra power (*power boosts*) at each of T rounds. For our instance, power boosts are allcoated using RAPL [1] technology and each application's performance is measured under base and high power limits, 30W and 130W, respectively. We evaluate Apache Spark [29] benchmarks. Table 1 lists the twelve Spark applications in our instance.

Each Spark application is defined by a fixed number of tasks. We profile tasks' completion time. We define an application's utility in a round as the number of tasks completed normalized by its total number of tasks.



(a) Running time, n varies (b) Running time, m varies (c) Running time, T varies

Figure 1: Simulation results showing the effect of varying number of agents, n, number of alternatives, m, and number of rounds, T, on the runtime of each algorithm.



(a) Value, uniform model (b) Value, half-half model (c) Value, alternating model

Figure 2: Simulation results showing the effect of the input model on the value achieved by the algorithms. For each model, n and m are held constant while T varies.

Since the length of the utility trace is shorter when profiled under boosted power, we use linear interpolation to extend the shorter trace. Thus, for each application a, we estimate the base power utility ($u^{\text{base}}_{a,t}$) and boosted power utility ($u^{\text{boost}}_{a,t}$) in each round.

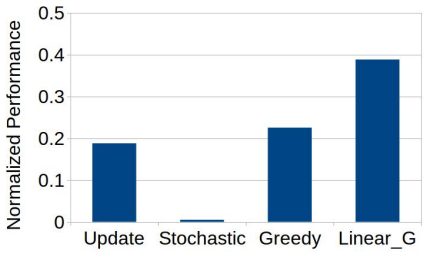


Figure 3: Nash Social Welfare achieved by the algorithms, as a fraction of OPT.

In our instance, there are two power boosts to be allocated. Therefore, at each round there are $\binom{12}{2}$ alternatives, one for each pair of applications. For an alternative j corresponding to power boosts for applications a and b, we have that $V_t(a,j) = u^{\text{boost}}_{a,t}$, $V_t(b,j) = u^{\text{boost}}_{b,t}$, and $V_t(c,j) = u^{\text{base}}_{c,t}$ for all other applications $c \neq a, b$. We have 120 rounds in the instance we tested.

The Nash Social Welfare achieved on this instance is shown in Figure 3, normalized against OPT. Most striking is the poor performance of STOCHASTIC. We hypothesize that this is due to some of the applications having long stretches of consecutive rounds where they achieve zero utility for all allocations, followed by short periods with positive reported utility. For these applications, the pattern of valuations looks most like the alternating distribution, where STOCHASTIC also performed poorly.

Also notable is the performance of LINEARGREEDY, which performs roughly twice as well as GREEDY and UPDATE. However, both of these consistently outperformed LINEARGREEDY for our simulated data. It is not yet clear to us whether there is a general structural property of this dataset which lends itself to LINEARGREEDY, or if its good performance was simply a chance occurrence.

Runtime results are similar to those presented in Section 6.1, with one exception. The time taken to solve the offline instance is 384 seconds, whereas UPDATE takes only 108 seconds. This provides evidence that, for large instances, the optimal MICP is prohibitively slow compared to our online algorithms. For comparison, the three other algorithms each ran in less than 0.2 seconds.

7. STRATEGIC ISSUES

In this section, we discuss strategic incentives of the agents that arise in the dynamic setting. We emphasize again that, while we have no formal results regarding incentives, and what we do know is mostly negative, we consider the multiple objective framework from Section 1 to be important and interesting in and of itself, and here there are no strategic concerns. Of course, we consider the strategic aspect to be very interesting also and believe it to be a fruitful area for generating research questions.

Without the upper bound on valuations, K, we might be worried about agents drastically overreporting their valuations. For instance, an agent could misreport some high valuation N in the first round and have their preferred alternative chosen. To compensate for this utility counting against them in the next round, they could report valuations on the order of N^2, then N^4, etc. Of course, with our upper bound in place, such a strategy would have to stop somewhere and could not be used to dominate every round (as it could if there was no bound).

Still, the setup and approaches described in this paper are highly vulnerable to strategic behaviour by the agents. The reason for this is that it is impossible to distinguish between an agent that is genuinely unhappy with choices made in previous rounds, and an agent simply pretending to be unhappy with previous alternatives. If we wish to compensate the former agent for their unhappiness, we must allow for the possibility of being gamed by the latter. This is essentially the free-rider problem well-known to economists (see,

e.g., [12] for a discussion).

There are many interesting and unresolved questions in this domain. While we can not hope for strategy-proofness in the most general setting, are there restricted settings in which we do regain (limited) strategy-proofness? For instance, can we limit the expressiveness allowed to the agents in exchange for strategy-proofness? Can we say anything about the case where alternatives are simply different allocations of private goods (so that no agent may benefit directly from 'free-riding' on another agent's high valuation)?

Even in the general setting, it may be true that agents never have an incentive to over-report their valuation in the case that $T = \infty$. Intuitively, if the algorithm believes the agent has more utility than they really do, it can only hurt the agent in future rounds. However, we have neither a proof nor a counter-example to this (imprecise) statement.

8. APPLICATIONS

8.1 Voting

Our setup can be directly applied to voting, where agents are voters who report a utility for each of the alternatives. If voters are required to report only an ordering over alternatives within each round, then we can simply infer utilities according to a chosen *scoring vector*. For example, we could set $V(i, j) = 1$ if alternative j is voter i's most preferred alternative and 0 otherwise (the *plurality* scoring vector), or $V(i, j) = m - k$ when j is ranked k-th in i's preference order (the *Borda* utility vector).

An interesting direction for future work is to investigate what social-choice theoretic properties are satisfied by the Nash social welfare in this repeated setting. One weak property is *unanimity*, which states that if all voters rank the same alternative at the top of their ordering, then that alternative should be chosen (in the dynamic setting we could require this on a round-by-round basis). Clearly all of the algorithms presented in this paper satsify this condition for all monotone scoring vectors. Some other fundamental axioms also extend naturally to the dynamic setting, for example *anonymity* and *neutrality* (which we define for the dynamic setting in Section 2).

For some axioms, however, it is not so clear how to extend to the dynamic setting. For instance, consider the *Condorcet criterion*, which states that any alternative which achieves a pairwise majority against all other alternatives should be chosen. This makes sense in the one-shot setting, but maybe less sense in the dynamic case. Suppose that there are two alternatives, A and B, and that 51% of voters prefer A to B in every round. Then the Condorcet criterion appears to say that we should choose A in every round, while fairness considerations dictate choosing B at least occasionally. It is not clear how we would extend the Condorcet criterion to the dynamic setting and, if we cannot, we may need novel axioms.

There is a natural link between repeated elections and the theory of multi-winner elections. In multi-winner elections, not only do we want to choose popular alternatives, but we also want to represent as many voters as possible, for which several rules have been designed [10, 20, 17]. Consider an election where the aim is to choose a committee of size $k < m$. This is exactly equivalent to setting $T = k$ and choosing a single distinct winner at each round, while also imposing the restriction that voters do not change their votes between rounds. Thus we can view multi-winner elections as a special case of repeated elections. It would be interesting to check whether any desiderata in the context of multi-winner elections extend naturally to the repeated setting.

8.2 Allocating Shared Resources

Consider a situation in which a group of agents take turns being allocated a shared resource for discrete units of time. Examples include allocating supercomputer time among members of a university department or assigning the use of a holiday home owned jointly by several people. In both cases, demand varies across time intervals and across agents. For instance, people who like skiing may want use of the holiday home in the winter, while those who like hiking may prefer a different season.

Another interesting aspect of these situations is that our notion of fairness may not be to treat all agents exactly equally. For instance, if people contributed unequally to the purchase of the holiday home, the group may decide that someone who paid twice as much as another person 'deserves' to get twice the benefit from the home. In the supercomputer example, we may wish to allocate time based on the amount of grant money contributed to the purchase of the machine (for example).

In these cases we may wish to generalize the Nash social welfare to the *Cobb-Douglas welfare*. The Cobb-Douglas welfare for utility vector v, $CD(v)$, is given by

$$CD(v) = \prod_{i=1}^{n} v(i)^{\alpha_i},$$

where $\sum_{i=1}^{n} \alpha_i \leq 1$. The case where all $\alpha_i = \frac{1}{n}$ is the special case of Nash social welfare, but setting other values of α_i allows us to prioritize some agents over others. It is illuminating to consider the simple case where all agents have a common unit of utility (say, dollars). In this case, the Nash social welfare is maximized when all agents receive exactly the same utility. If we generalize the coefficients, then the Cobb-Douglas welfare is maximized when the agents receive utility in exactly the ratio of their exponents α_i. So if agent i contributed twice as much money to the purchase of the holiday home as agent j, simply set $\alpha_i = 2\alpha_j$.

9. CONCLUSION

Election designers and social choice researchers often do not consider the fact that many elections are conducted as sequences of related elections. In this work, we have provided a framework to allow for the design and analysis of dynamic election protocols, and repeated decision making rules generally. We have presented four candidate online algorithms for solving these dynamic problems. Our simulations do not determine a clear winner, but instead suggest that the right choice of algorithm is highly dependent on the setting and the model of how agents' valuations change over time.

Our work is preliminary, and leaves a lot of scope for future research in addition to the specific directions already discussed. One direction would be to design a more precise model of voter preferences, possibly modeling changing preferences by an MDP as has been done in [9, 24]. We have also not considered modeling discounting of the agents' utilities. It would also be nice to have provable guarantees on the regret of the greedy algorithms.

Acknowledgments

We sincerely thank Songchun Fan for sharing the data used in Section 6.2. We are thankful for support from NSF under awards IIS-1527434, IIS-0953756, CCF-1101659, CCF-1149252 (CAREER), CCF-1337215 (XPS-CLCCA), SHF-1527610, and AF-1408784, ARO under grants W911NF-12-1-0550 and W911NF-11-1-0332, and a Guggenheim Fellowship. This work is also supported by STARnet, a Semiconductor Research Corporation program, sponsored by MARCO and DARPA. This work was done in part while Conitzer was visiting the Simons Institute for the Theory of Computing. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of these sponsors.

REFERENCES

[1] IntelÂő 64 and ia-32 architectures software developer's manual. https://www-ssl.intel.com/content/dam/www/public/us/en/documents/manuals/64-ia-32-architectures-software-developer-vol-3b-part-2-manual.pdf.

[2] Movielens. http://grouplens.org/datasets/movielens/.

[3] Us census data (1990) data set. https://archive.ics.uci.edu/ml/datasets/US+Census+Data+(1990).

[4] Web data commons: Hyperlink graphs. http://webdatacommons.org/hyperlinkgraph/index.html.

[5] S. Agrawal and N. R. Devanur. Fast algorithms for online stochastic convex programming. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1405–1424. SIAM, 2015.

[6] M. Aleksandrov, H. Aziz, S. Gaspers, and T. Walsh. Online fair division: analysing a food bank problem. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI-15)*, 2015.

[7] A. Andersson, M. Tenhunen, and F. Ygge. Integer programming for combinatorial auction winner determination. In *Proceedings of the Fourth International Conference on Multiagent Systems*, pages 39–46. IEEE, 2000.

[8] H. Aziz, M. Brill, V. Conitzer, E. Elkind, R. Freeman, and T. Walsh. Justified representation in approval-based committee voting. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 784–790, Austin, TX, USA, 2015.

[9] C. Boutilier and A. D. Procaccia. A dynamic rationalization of distance rationalizability. In *AAAI*, 2012.

[10] J. R. Chamberlin and P. N. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *American Political Science Review*, 77(03):718–733, 1983.

[11] R. Cole and V. Gkatzelis. Approximating the Nash social welfare with indivisible items. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 371–380. ACM, 2015.

[12] R. Cornes and T. Sandler. *The theory of externalities, public goods, and club goods*. Cambridge University Press, 1996.

[13] A. Darmann and J. Schauer. Maximizing Nash product social welfare in allocating indivisible goods. *European Journal of Operational Research*, 247(2):548–559, 2015.

[14] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, and I. Stoica. Dominant resource fairness: Fair allocation of multiple resource types. *Proceedings of the 8th USENIX conference on Networked systems design and implementation*, pages 24–24, 2011.

[15] M. Guo, V. Conitzer, and D. M. Reeves. Competitive repeated allocation without payments. In *Proceedings of the Fifth Workshop on Internet and Network Economics (WINE)*, pages 244–255, Rome, Italy, 2009.

[16] I. Kash, A. D. Procaccia, and N. Shah. No agent left behind: Dynamic fair division of multiple resources. *Journal of Artificial Intelligence Research*, pages 579–603, 2014.

[17] D. M. Kilgour. Approval balloting for multi-winner elections. In *Handbook on approval voting*, pages 105–124. Springer, 2010.

[18] D. Lehmann, R. Müller, and T. Sandholm. The winner determination problem. *Combinatorial auctions*, pages 297–318, 2006.

[19] C. Lucchese, S. Orlando, R. Perego, and F. Silvestri. WebDocs: a real-life huge transactional dataset. In *Workshop on Frequent Itemset Mining Implementations*, 2004.

[20] B. L. Monroe. Fully proportional representation. *American Political Science Review*, 89(04):925–940, 1995.

[21] J. F. Nash Jr. The bargaining problem. *Econometrica: Journal of the Econometric Society*, pages 155–162, 1950.

[22] D. Parkes, A. Procaccia, and N. Shah. Beyond dominant resource fairness: Extensions, limitations, and indivisibilities. In *ACM Conference on Electronic Commerce*, pages 808–825. ACM, 2012.

[23] D. C. Parkes, R. Cavallo, F. Constantin, and S. Singh. Dynamic incentive mechanisms. *AI Magazine*, 31(4):79–94, 2010.

[24] D. C. Parkes and A. D. Procaccia. Dynamic social choice with evolving preferences. AAAI, 2013.

[25] A. D. Procaccia. Cake cutting: not just child's play. *Communications of the ACM*, 56(7):78–87, 2013.

[26] S. Ramezani and U. Endriss. *Nash social welfare in multiagent resource allocation*. Springer, 2010.

[27] T. Sandholm. Algorithm for optimal winner determination in combinatorial auctions. *Artificial intelligence*, 135(1):1–54, 2002.

[28] J. Stamper, A. Niculescu-Mizil, S. Ritter, G. Gordon, and K. Koedinger. Algebra i 2006-2007. challenge data set from kdd cup 2010 educational data mining challenge. http://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp.

[29] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica. Spark: Cluster computing with working sets. In *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, volume 10, page 10, 2010.

Minimising the Rank Aggregation Error

Mathijs M. de Weerdt
Delft University of Technology, Netherlands
m.m.deweerdt@tudelft.nl

Enrico H. Gerding
University of Southampton, UK
eg@ecs.soton.ac.uk

Sebastian Stein
University of Southampton, UK
ss2@ecs.soton.ac.uk

ABSTRACT

Rank aggregation is the problem of generating an overall ranking from a set of individual votes. The aim in doing so is to produce a ranking which is as close as possible to the (unknown) correct ranking for a given distance measure such as the Kendall-tau distance. The challenge is that votes are often both noisy and incomplete. Existing work has largely focused on finding the most likely ranking for a particular noise model (such as Mallows'). Instead, here we focus on minimising the error, i.e., the expected distance between the aggregated ranking and the true underlying one. Specifically, we show that the two objectives result in different rankings, and that these differences become especially significant when many votes are missing. Furthermore, we show how to compute local improvements on existing rankings to reduce the expected error. Finally, we run extensive experiments on both synthetic and real data to compare different aggregation rules. In particular, a surprising result is that for votes generated according to the Mallows' model, Copeland often outperforms Kemeny optimal, despite the latter being the maximum likelihood estimator.

Categories and Subject Descriptors

I.2.11 [**Distributed AI**]: Multiagent systems

Keywords

Economic paradigms: Social Choice Theory

1. INTRODUCTION

Rank aggregation is the problem of producing a complete ranking from votes cast by individual agents, where the votes can be seen as noisy and incomplete estimates of a ranking that is an underlying ground truth. This perspective on voting dates back to Marquis de Condorcet [25], who said that voting may be regarded as a way of uncovering this ground truth. There are many practical examples[1] of rank aggregation, including websites that produce rankings of restaurants, books and movies based on crowdsourced contributions from their users, scientific communities that use votes from their members to select which project proposals to fund or which papers to accept [18, 5], or peer grading in massive online open courses [3]. Another prominent application is the use of rank aggregation to produce a meta search engine from the search results of individual search engines [13]. In these settings votes are not only noisy, but also incomplete since typically only a subset of the candidates (e.g., restaurants or websites) is ranked by any single individual.

To find a ranking which is close to the ground truth, most current work assumes a probabilistic noise model such as Mallows [22, 32], and then aims to maximise the likelihood of an aggregate ranking. In Mallows' model, a probability is assumed for ordering a pair of candidates correctly, and votes are produced by repeatedly ordering all pairs until this results in a consistent (acyclic) ranking. For this model, it has been shown that Kemeny's rule is the maximum likelihood estimator (MLE) [32]. Similarly, some other commonly used voting rules are MLEs for specific noise models [9, 8].

However, in most settings the aim should arguably not be to find a most likely explanation of the noisy observations, but to find a ranking that gives the best results when used in subsequent decision making. When votes are noisy and incomplete, many rankings may have a likelihood of similar magnitude, and there may even be multiple rankings with the maximum likelihood. In all these cases, the probability that a ranking with maximum likelihood is the true ranking is small. When an aggregated ranking is used, success does not depend on having found the true ranking exactly. Rather, it is important to construct and use a ranking for which the distance to the true ranking is as small as possible *in expectation*. This means that we should aim to minimise the expected distance (for a particular measure) of an aggregate ranking to the true ranking — which we term the *error* — instead of aiming for a ranking that maximises the likelihood. In contrast to MLEs, to date it is still unknown which commonly used voting rules perform best regarding this objective.

Against this background, in this paper we assume noise according to the Mallows' model, and for this model we make the following novel contributions. (i) We show that the MLE is not always minimising the error. (ii) We show that computing the error is #P-hard. (iii) We show that *local Kemenisation*, a computationally simple procedure for improving aggregated rankings in terms of their likelihood, also reduces the error, and (iv) through experiments on both synthetic and real data, we show how noise and incompleteness influence the performance of a large set of voting rules.

[1] `http://www.tripadvisor.com/Restaurants`, `http://www.goodreads.com/choiceawards`

This is an extended version of a two-page abstract accepted for the 15th International Conference on Autonomous Agents and Multiagent Systems.
Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. May 9th-10th, 2016. Singapore.

The paper is structured as follows. In Section 2 we introduce notation, the distance measure used, and the model for noise and incompleteness, and we show that Kemeny's rule is the MLE also for the model including incompleteness. In Section 3 we then formally define the objective of minimising the error and show how this is different from maximising the likelihood. We give the hardness proof of computing the error and show that local Kemenisation reduces the error. In Section 4 we first introduce the voting rules and their adaptations to settings with incomplete rankings. Subsequently, we evaluate these rules under varying levels of noise and incompleteness for both synthetic votes as well as on two ranking data sets from PrefLib [24]. Section 5 discusses related work and Section 6 concludes.

2. MODEL

The aim in this paper is to find rank aggregation rules which minimise the expected error, where the error is given by the Kendall-tau distance to the true ranking (as defined below). Formally, let $A = \{1, 2, \ldots, m\}$ denote a set of candidates or *alternatives*, where $m = |A|$ is the number of alternatives to be ranked. In addition, let $N = \{1, 2, \ldots, n\}$ denote the set of n agents or voters. Each agent has an incomplete ranking over the set of candidates. This is modeled as a complete order on a subset of the candidates.[2] We thus define a vote by agent k as a ranking, i.e., linear order, over a subset $A_k \subseteq A$ of the candidates, denoted by $\sigma_k : A_k \to \{1, 2, \ldots, |A_k|\}$. Here, $\sigma_k(i)$ defines the rank of candidate i (lower is better). We also use $i \succ_{\sigma_k} j$ to denote $\sigma_k(i) < \sigma_k(j)$, i.e., i is ranked higher than j according to agent k. Furthermore, $|\sigma_k| \leq m$ is the number of candidates voted for by agent k. Note that $\sigma_k(j)$ is undefined for any candidate $j \in A \setminus A_k$. In such a case we say the vote is *incomplete*. By inserting remaining alternatives $A \setminus A_k$ in an incomplete ranking, we can construct a potential underlying complete ranking of all alternatives. This is called a *completion* (or extension [17]). Furthermore, we sometimes use $D = \{\sigma_1, \ldots, \sigma_n\}$ (for observed data) to denote all votes.

If we have access to the underlying true ranking, we can measure the quality of a voting rule on a given profile of votes by the distance of the aggregated ranking to the true ranking. The most common distance metric, and the one we use in this paper, is the Kendall-tau distance [16]. In detail, the Kendall-tau distance K counts the pairs of alternatives that are differently ordered by σ than by τ.

$$K(\sigma, \tau) = |\{\{i, j\} \subseteq A : i \succ_\sigma j \text{ and } i \prec_\tau j\}| \quad (1)$$

Such a differently ordered pair i and j is called an *inversion*. The Kendall-tau distance can be found in $\mathcal{O}(m \ln m)$ using a folk algorithm variant of merge sort called "count inversions".

The rule selecting an aggregate ranking τ^* which minimises the Kendall-tau distances to all votes is called the Kemeny optimal aggregation rule, and is given by:

$$\tau^* = \arg\min_\tau \sum_{k \in N} K(\sigma_k, \tau) \quad (2)$$

A more convenient way to write this is:

$$\tau^* = \arg\min_\tau \sum_{\{i,j\} \subseteq A : i \succ_\tau j} n_d(i, j|D), \quad (3)$$

[2]This differs from [31] who consider any partial order.

where $n_d(i, j|D) = |\{k \in N : i \prec_{\sigma_k} j\}|$ is the number of voters who disagree with the order $i \succ j$. Likewise, $n_a(i, j|D) = |\{k \in N : i \succ_{\sigma_k} j\}|$ denotes the number of agreements. Note that, in case of complete rankings, we have that $n_a(i, j|D) + n_d(i, j|D) = n$. However, this is not necessarily the case when rankings are incomplete.

We now describe the model for noisy and missing observations/votes. We assume noise according to the well-known Mallows' model for a probability $p > 0.5$. In this model, the likelihood of a ranking τ given observed votes D is:

$$\mathcal{L}(\tau|D) = \frac{1}{Z_1} \prod_{\{i,j\} \subseteq A : i \succ_\tau j} p^{n_a(i,j|D)} (1-p)^{n_d(i,j|D)}, \quad (4)$$

where Z_1 is a (normalisation) constant. It has been shown that Kemeny optimal chooses the ranking with the highest likelihood [32].

We extend this model to incomplete rankings by introducing the probability of a vote missing, where this probability is given by q. We assume that this probability is independent of the position in the true ranking. Incorporating this probability, we can compute the likelihood of a ranking τ given observations D as follows:

$$\mathcal{L}'(\tau|D) = \mathcal{L}(\tau|D) \frac{1}{Z_2} \prod_{k \in N} (1-q)^{|\sigma_k|} q^{m - |\sigma_k|}. \quad (5)$$

This assumption underlying this model has sometimes been called the "missing at random assumption" [14].

It is easy to see that Kemeny optimal still maximises the likelihood, irrespective of the value of q.

THEOREM 1. *Kemeny optimal is the maximum-likelihood estimator for Equation 5.*

PROOF. The ranking which maximises the likelihood is also maximising the log-likelihood. This allows us to drop all constants, including the normalisations Z_1 and Z_2 and even the incompleteness probabilities. Therefore:

$$\arg\max_\tau \mathcal{L}'(\tau, D) = \arg\max_\tau \ln(\mathcal{L}'(\tau, D)) =$$

$$\arg\max_\tau \sum_{\{i,j\} \subseteq A : i \succ_\tau j} (n_a(i, j|D) \ln(p) + n_d(i, j|D) \ln(1-p))$$

Since $p > 0.5$ we know that $\ln(p) > \ln(1-p)$, and because the total sum of ranked pairs is constant, we conclude that this log-likelihood is maximised if the number of agreements is (i.e., the number of disagreements is minimised). □

3. MINIMISING THE ERROR

So far we have discussed maximising the likelihood. Instead, the goal in this paper is to minimise the *rank aggregation error*, which we define as the *expected Kendall-tau distance*. Formally this is given by:

$$\text{KT-error}(\tau, D) = \sum_{\tau' \in T} K(\tau, \tau') \cdot \mathcal{L}'(\tau'|D), \quad (6)$$

where T is the set of all possible rankings.

Below we first discuss several examples to show that the two aims can result in different rankings, and then discuss the computational hardness of minimising the error and a local search approach for finding incremental improvements.

3.1 Likelihood vs. Error

Minimising the expected Kendall-tau distance and maximising the likelihood result in different rankings. We start by showing that this is true for instances with candidates that do not occur in any vote. We call such a candidate *free*.

DEFINITION 1. *A candidate $a \in A$ is* free *when none of the votes D contain a.*

EXAMPLE 1. *Let three candidates a, b, c be given, and one agent with vote $a \succ_{\sigma_1} b$. Kemeny's rule is indifferent between the three possible aggregate rankings (without noise (i.e., $p = 1$), each has a likelihood of $\frac{1}{3}$). The expected Kendall-tau distances for each of these, however, differ:*

τ_i	*ranking*	$\mathcal{L}(\tau_i\|D)$	$K(\tau_i,\tau_j)$			*KT-error*(τ_i, D)
τ_1	$a \succ b \succ c$	$\frac{1}{3}$	*0*	*1*	*2*	*1*
τ_2	$a \succ c \succ b$	$\frac{1}{3}$	*1*	*0*	*1*	$\frac{2}{3}$
τ_3	$c \succ a \succ b$	$\frac{1}{3}$	*2*	*1*	*0*	*1*

Because the distance of $a \succ c \succ b$ to each of the other rankings is only 1, it has a lower expected error. Note that the free candidate here is in the middle of the ranking.

This example shows that minimising the error produces a single natural ranking with the free candidate in the middle, where the likelihood is the same for multiple rankings. Generally:

PROPOSITION 1. *On an instance D with free candidates:*

1. *Kemeny's rule is indifferent between the position of free candidates (i.e., each position is equally likely);*
2. *The KT-error is minimised when free candidates are positioned in the median of the ranking.*

PROOF. Let a ranking π of m candidates be given. Let T be the set of all rankings of length m, $\mathcal{L}(\tau|D)$ denote the likelihood of a ranking $\tau \in T$, and let π_i (or τ_i) denote the ranking π (or τ, respectively) where the free candidate is placed at position $i \in \{0, \ldots, m\}$.

For the first statement, since the free candidate does not appear in D, each position is equally likely, and therefore, for any τ_j the likelihood $\mathcal{L}(\tau_j|D) = c \cdot \mathcal{L}(\tau|D)$, with $c = \frac{1}{m+1}$. Therefore, with Theorem 1, Kemeny's rule places free candidates at every position with equal probability.

For the second statement, we show that inserting a free candidate in the middle of the ranking minimises the expected error. By definition of the KT-error and $\mathcal{L}(\tau_j|D) = c \cdot \mathcal{L}(\tau|D)$ from above, we have that

$$\text{KT-error}(\pi_i, D) = \sum_j \sum_{\tau \in T} K(\pi_i, \tau_j) \cdot c \cdot \mathcal{L}(\tau|D).$$

The distance (error) of π_i to τ_j is equal to the distance of π to τ except for the difference in the position of the free candidate, i.e., $|i - j|$. Consequently,

$$\begin{aligned}&\text{KT-error}(\pi_i, D)\\ &\quad = c \cdot \textstyle\sum_j \sum_{\tau \in T} (K(\pi, \tau) + |i - j|) \cdot \mathcal{L}(\tau|D)\\ &\quad = \text{KT-error}(\pi, D) + c \cdot \textstyle\sum_j |i - j| \cdot \sum_{\tau \in T} \mathcal{L}(\tau|D)\end{aligned}$$

This is minimal if and only if $\sum_j |i - j| = \sum_{j=0}^{i-1}(i - j) + \sum_{j=i+1}^{m}(j - i)$. This is minimal for $i = \frac{m+1}{2}$. With induction this also holds for a set of free candidates. □

Next, we show that the difference between the two objectives goes even beyond free candidates, and can result in different rankings even when the votes are complete.

EXAMPLE 2. *Let $p = 0.7$ and five complete votes be given: twice $a \succ b \succ c$, twice $c \succ a \succ b$ and once $b \succ c \succ a$.*

τ_k	*ranking*	$\sum n_a$	$\sum n_d$	$\mathcal{L}(\tau_k\|D)$	*KT-error*
τ_0	$a \succ b \succ c$	*9*	*6*	*0.361*	*1.123*
τ_1	$a \succ c \succ b$	*6*	*9*	*0.028*	*1.877*
τ_2	$b \succ a \succ c$	*8*	*7*	*0.155*	*1.035*
τ_3	$b \succ c \succ a$	*7*	*8*	*0.066*	*1.965*
τ_4	$c \succ a \succ b$	*9*	*6*	*0.361*	*1.123*
τ_5	$c \succ b \succ a$	*6*	*9*	*0.028*	*1.877*

Then the Kemeny rule selects $a \succ b \succ c$ or $c \succ a \succ b$, but the ranking that minimises the KT-error is $b \succ a \succ c$.

Also here, we see that the MLE is not minimising the expected error.

3.2 Hardness

Finding the ranking with the largest likelihood (i.e., Kemeny optimal) is NP-complete [15]. Towards establishing the computational complexity of finding an aggregate ranking with the minimum error, we can show that the problem of computing the error of a (single) aggregate ranking is #P-hard (even) when there is no noise in the data D. The proof uses a reduction from computing the number of linear extensions of a partial order, which is #P-complete [2].

In the proofs below, let T_D denote the set of extensions of the partial order defined by the votes of D and $x_D = |T_D|$ the number of such extensions. For our proof we need the following lemmas.

LEMMA 1. *$\mathcal{L}(\tau|D)$ is the same for all consistent rankings $\tau \in T_D$ if D is generated from a model without noise, and this is equal to $\frac{1}{x_D}$.*

This follows because incompleteness is determined independently from the position in the ranking.

If D does not imply a complete linear order, there is a pair a, b of unordered alternatives. We express the number of consistent extensions of D in terms of the numbers of extensions for both possible orders for a and b as follows.

LEMMA 2. *Given a ranking π and data D without noise on a set of alternatives A. Let $a, b \in A$ be given. Let D_{ab} denote $D \cup \{(a, b)\}$. Then*

$$\begin{aligned}KT\text{-}error(\pi, D) \cdot x_D = \;& KT\text{-}error(\pi, D_{ab}) \cdot x_{D_{ab}}\\ +\; & KT\text{-}error(\pi, D_{ba}) \cdot (x_D - x_{D_{ab}}).\end{aligned}$$

PROOF. With Lemma 1, $\mathcal{L}(\tau|D) = \frac{1}{x_D}$. Applying the definition of KT-error, we write

$$\begin{aligned}&\text{KT-error}(\pi, D) \cdot x_D = \textstyle\sum_{\tau \in T_D} K(\pi, \tau)\\ &\quad = \textstyle\sum_{\tau \in T_{D_{ab}}} K(\pi, \tau) + \sum_{\tau \in T_{D_{ba}}} K(\pi, \tau).\end{aligned}$$

Using the definition of KT-error and of D_{ab} and D_{ba} the result then follows. □

The idea of the proof below is now to repeatedly use this fact that the number of consistent extensions of D is equal to the sum of the number of consistent extensions of $D \cup \{(a, b)\}$ and of $D \cup \{(b, a)\}$. By repeatedly adding (yet) unordered pairs of a and b to D, we collect a polynomial number (at most m^2) of linear constraints on the numbers of consistent extensions of increasingly larger sets of votes, ultimately leading to a single consistent extension.

THEOREM 2. *Given data D generated from a model without noise, determining the expected error of an aggregate ranking π is #P-hard.*

PROOF. We show this by a (polynomial) reduction from the problem of computing the number of linear extensions of a partial order $\succ$. Let such a partial order $\succ$ over a set of candidates A be given. First, for each pair of candidates (a, b) with $a \succ b$ in the partial order, insert an incomplete vote with a before b in D. Let a ranking $\pi \in T_D$ be given (e.g., by taking a topological order). Initially, D has x_0 consistent extensions. Then execute the following algorithm.

1. Initialise i to 0 and $e_0 = \text{KT-error}(\pi, D)$.
2. For every pair (a, b), if adding $a \succ b$ does not create a cycle in the majority graph of D then
   (a) Increment i, set $D_{ab} = D \cup \{(a, b)\}$ and $D_{ba} = D \cup \{(b, a)\}$.
   (b) Let $e_i = \text{KT-error}(\pi, D_{ab})$.
   (c) Let C_i denote the constraint $e_{i-1} \cdot x_{i-1} = e_i \cdot x_i + \text{KT-error}(\pi, D_{ba}) \cdot (x_{i-1} - x_i)$.
   (d) Let $D = D_{ab}$.

Let k denote i in the last iteration. All thus found constraints C_i are valid because of Lemma 2. When the last pair of candidates has been added to D, at index $k \leq m^2$, there is only one consistent extension, hence $x_k = 1$. If we know a value x_i, we can compute x_{i-1} using the equality constraint C_i with both x_i and x_{i-1}. So with induction we can compute x_0 in k such steps. This gives us the total number of consistent extensions of the partial order in polynomial time. Since with this polynomial number of calls to KT-error we solved a #P-hard problem, we conclude that computing KT-error is also #P-hard. □

3.3 Local Search

Although computing the error for candidate rankings to find the minimal one does not seem to be feasible, we can improve a ranking by making local adjustments. In particular, given a ranking τ, it is easy to determine if, by swapping two adjacent candidates, we can improve the KT-error.

THEOREM 3. *Let τ_{ab} and τ_{ba} be two equal rankings except that two adjacent candidates, a and b, are swapped. That is, $a \succ_{\tau_{ab}} b$ and $b \succ_{\tau_{ba}} a$. Then: KT-error$(\tau_{ab}, D) <$ KT-error(τ_{ba}, D) iff $n_a(a, b|D) > n_a(b, a|D)$.*

PROOF. Let $T_{ab} \subset T$ denote the set of all rankings where $a \succ b$ (not necessarily adjacent ones) and $T_{ba} = T \backslash T_{ab}$ the set of all rankings where $b \succ a$. Then $T_{ab} \cup T_{ba}$ is a partition of all possible rankings and $|T_{ab}| = |T_{ba}|$. We can thus write the KT-error(τ, D) (using Equation 6) as:

$$\sum_{\tau' \in T_{ab}} K(\tau, \tau') \cdot \mathcal{L}(\tau'|D) + \sum_{\tau' \in T_{ba}} K(\tau, \tau') \cdot \mathcal{L}(\tau'|D).$$

For $\tau \in T_{ab}$, from the definition of the Kendall-tau distance, we know that $K(\tau_{ba}, \tau) = K(\tau_{ab}, \tau) + 1$, and for $\tau \in T_{ba}$, $K(\tau_{ba}, \tau) = K(\tau_{ab}, \tau) - 1$. Therefore the KT-error$(\tau_{ba}, D)$ is

$$\begin{aligned}
&= \sum_{\tau \in T_{ab}} (K(\tau_{ab}, \tau) + 1) \cdot \mathcal{L}(\tau|D) \qquad (7)\\
&\quad + \sum_{\tau \in T_{ba}} (K(\tau_{ab}, \tau) - 1) \cdot \mathcal{L}(\tau|D)\\
&= \sum_{\tau \in T_{ab}} K(\tau_{ab}, \tau) \cdot \mathcal{L}(\tau|D) + \sum_{\tau \in T_{ab}} \mathcal{L}(\tau|D)\\
&\quad + \sum_{\tau \in T_{ba}} K(\tau_{ab}, \tau) \cdot \mathcal{L}(\tau|D) - \sum_{\tau \in T_{ba}} \mathcal{L}(\tau|D)\\
&= \text{KT-error}(\tau_{ab}, D) + \sum_{\tau \in T_{ab}} \mathcal{L}(\tau|D) - \sum_{\tau \in T_{ba}} \mathcal{L}(\tau|D).
\end{aligned}$$

Note that $\sum_{\tau \in T_{ab}} \mathcal{L}(\tau|D) + \sum_{\tau \in T_{ba}} \mathcal{L}(\tau|D) = 1$, or, more generally, a constant (normalisation is not relevant here).

We can express the likelihood that a comes before b given the data as:

$$\sum_{\tau \in T_{ab}} \mathcal{L}(\tau|D) = \mathcal{L}(a \succ b|D) = \frac{1}{Z} p^{n_a(a,b|D)} (1-p)^{n_a(b,a|D)}.$$

Using Equation 7 we thus can write the difference between errors, KT-error(τ_{ba}, D) − KT-error(τ_{ab}, D), as:

$$\frac{1}{Z} \left(p^{n_a(a,b|D)} (1-p)^{n_a(b,a|D)} - p^{n_a(b,a|D)} (1-p)^{n_a(a,b|D)} \right)$$

Since $p > 0.5$, this difference is strictly positive (negative) iff $p^{n_a(a,b|D)} > p^{n_a(b,a|D)}$ (or $p^{n_a(a,b|D)} < p^{n_a(b,a|D)}$). □

It turns out that repeatedly applying this rule until a local optimum is found has been called local Kemenisation [13]. It has been shown that any ranking thus produced satisfies the generalised Condorcet criterion (i.e., if there is a partition of the candidates (A_1, A_2) such that for every $a \in A_1$ and $b \in A_2$ the majority prefers a to b, then every $a \in A_1$ must be ranked above every $b \in A_2$ [28]). The above proof adds that this is locally minimising the KT-error as well. Note that although the proof assumes Mallows' model, it seems intuitive for any noise model to swap adjacent candidates if one is ranked more often above the other, and that this is independent of the (often unknown) value for p in the model.

4. EXPERIMENTS

We now empirically evaluate a range of voting rules, to determine their performance in settings with incomplete rankings. To this end, we first discuss the experimental setup and data, followed by the voting rules and modifications to deal with incomplete rankings. Then we discuss the results.

4.1 Setup and Data

We consider two types of experiments: those generated using synthetic data, and those based on real data from the PrefLib library [24], specifically the Mechanical Turk Dots and Puzzle experiments by [23]. We focus on these datasets since they provide noisy (albeit complete) rankings and they also include an objective ground truth. In more detail, for the synthetic data, we use the repeated insertion method discussed in [10, 20] to generate noisy rankings according to the Mallows' model. Furthermore, to generate incomplete rankings, we independently remove each candidate from each agent with a probability q.

The real data consists of several datasets, each containing up to 800 agents ranking 4 candidates. In the Dots experiment, each voter was asked to rank 4 images according to the number of dots they contained, whereas in the Puzzle game the voters were asked to rank sliding puzzles according to how close they were to the solution (see [23] for details). The votes contain natural noise and are complete. To make the votes incomplete, we remove each candidate from each agent with probability q as before. In addition, we randomly select n agents without replacement (where we vary n).

We repeat each experiment 1000 times with resampled random values for candidates to determine the order in case of ties in the scoring rules, and we measure the average Kendall-tau distance (i.e., the number of inversions) of the aggregated ranking to the true ranking. Note that this is consistent with our objective of minimising the KT-error.

4.2 Voting Rules for Incomplete Votes

We consider the following common voting rules in the literature on rank aggregation. For each of the rules below, we also add a variant with local search, where we improve the rank produced by the corresponding rule by applying the algorithm described in Section 3.3 until it converges.

Borda.

According to the Borda rule every agent assigns $n-j$ points to the candidate ranked in position j, which is equal to the number of candidates it defeats. Candidates are then ranked according to the sum of points for each candidate, which is also called the *Borda count*. However, with incomplete votes, it is not clear how many points should be awarded to the candidates ranked by an agent and the ones missing. There are many variants (see, e.g., [1, 26, 13]) and we consider the following three: Pessimistic, where ranked candidates contribute a score of $(m-\sigma_k(i))$ and unranked ones contribute zero points; Optimistic, which is the same except that unranked candidates contribute $(m-|\sigma_k|)$; Scaled, where the score is proportional to the position within the ranked candidates, $m(|A_k|-\sigma_k(i))/|A_k|$, and missing candidates contribute zero. We choose these three since they vary widely in their performance, whereas other variants we tried performed similarly to one of these three.

Spearman's Footrule.

Spearman's footrule is another commonly-analysed voting rule, especially for the Mallows' noise model, since it is a 2-approximation of Kemeny optimal but computable in polynomial time. This rule minimises the sum of Spearman's distances of the complete ranking to the votes, where the distance between two rankings σ and τ is given by $S(\sigma,\tau)=\sum_{i\in A}|\sigma(i)-\tau(i)|$. For complete rankings, this is done by finding the minimal weighted matching of alternatives to their ranks in the aggregate ranking, where the weight w_{ij} of a candidate i in position j is given by $w_{ij}=\sum_k|\sigma_k(i)-j|$ (computable in $\mathcal{O}\left(m^3\right)$ using the Hungarian algorithm). Now, as with Borda, there are several ways to extend this rule to deal with incomplete votes. We choose *Scaled Footrule Optimal* (SFO) from [13] since it is simple and computationally tractable (unlike, e.g., using the induced distance [13]). In detail, to compute the distance for candidate i at position j, instead of using $\sigma_k(i)$ and j, both of these are scaled according to the total number of candidates. Specifically, the weight is replaced by $w_{ij}=\sum_k|\sigma_k(i)/|A_k|-j/m|$. This formulation represents the idea that the missing alternatives are equally spread in between the ranked alternatives.

Copeland.

The Copeland voting rule ranks individual candidates according to the number of wins in pair-wise contests minus the number of losses. This rule can be readily applied to incomplete settings by only counting pairs when both alternatives appear in an agent's ranking. Formally, let $P(i,j)=|\{k\in A: i,j\in A_k\wedge i\succ_{\sigma_k} j\}|$ denote the number of agents who prefer i to j. Then, candidates i's score is computed by:

$$|\{j\neq i: P(i,j)>P(j,i)\}|-|\{j\neq i: P(i,j)<P(j,i)\}|$$

Candidates then are ranked according to their score in descending order.

Kemeny optimal.

We implement the Kemeny optimal rule by a mixed integer optimisation problem on the weighted majority graph [7] for which we (uniform) randomly select one optimal solution.

Optimal.

Computing the KT-error exactly is hard (Theorem 2), but in practice we can still compute the Optimal for up to 6 candidates using a brute force approach. Specifically, the KT-error is computed for all possible rankings and then the one with the minimal KT-error is chosen. Note, however, that the definition of KT-error for Mallows' model depends on the noise probability p. For the synthetic experiments we simply use the p value that was used for generating the instances. For the experiments with the real data we compute the KT-error for a range of p values to establish the best one experimentally. As before, if there are multiple optimal solutions, we select one randomly.

4.3 Results

Figure 1 shows results using synthetic data with 6 candidates, a Mallows noise probability $p=\frac{2}{3}$ and a probability of candidates missing of $q=0.7$ for different values of the number of agents. The right figure shows the results after applying local Kemenisation to each of the voting rules. As expected, having more agents decreases the average distance to the true ranking for all rules. We also can observe that Kemeny is indeed not optimal, with on average around 0.5 inversions more than Optimal. Interestingly, Copeland performs significantly better than Kemeny, and at times on par with Optimal. Even more striking is the significant improvement of most rules by local Kemenisation, which can be observed by comparing the left to the right figure. We have similar results for other values for q, but show only $q=0.7$, because here the differences are most pronounced.

This can be seen in Figure 2, where we vary the probability q of missing candidates and show the average distance for all rules for a scenario with $p=\frac{2}{3}$ and 25 agents. Similar to the previous results, we see that Copeland consistently outperforms Kemeny, and that Kemeny is relatively far from optimal. Compared to the previous figure, we see here that differences between the rules are less pronounced for lower values of q. In particular, after applying local Kemenisation, none of the rules are statistically different up to $q=0.6$.

The results so far have considered data where the synthetic noise model is consistent with our objective. We now consider the real data, which uses natural noise generated through experiments rather than a particular model. To this end, Figures 3 and 4 show the results from the Dots dataset (number 1) and Puzzle (number 2) respectively, where we vary the number of agents and set the probability of missing a candidate to $q=0.7$. Surprisingly, trends for both datasets are very similar to the synthetic data: despite the fact that Optimal is not necessarily optimal with real data (since it assumes the Mallows' model), it significantly outperforms all other voting rules. Furthermore, Copeland outperforms Kemeny in most instances. Finally, again despite the fact that it assumes Mallows' model, local Kemenisation significantly improves most voting rules, except of course Kemeny and Optimal, which are already locally optimal, and Copeland for some instances (within the standard error). We can see the same trend for the other Dots and Puzzle instances (results not shown).

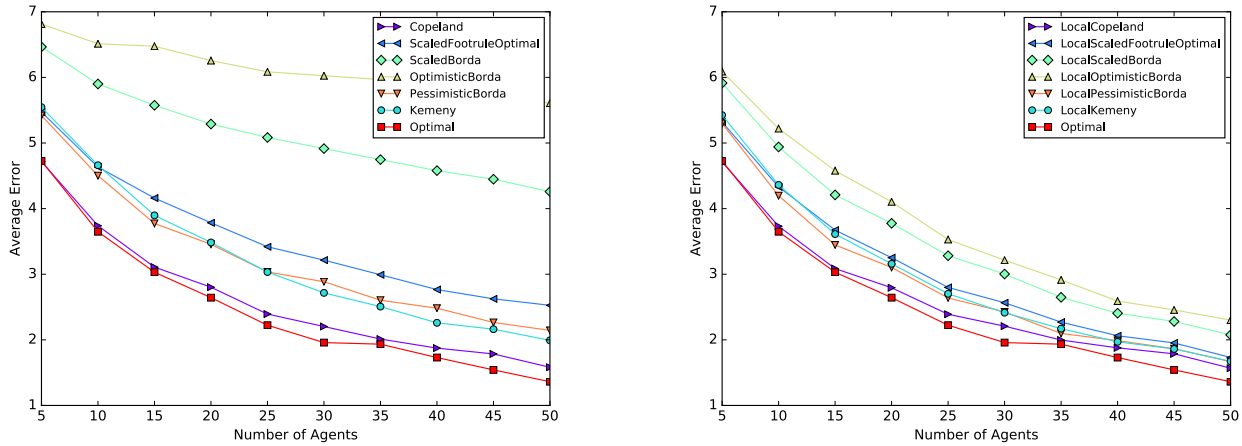


Figure 1: More agents decrease the average distance for all rules (6 candidates, $p = \frac{2}{3}$, and $q = 0.7$). Copeland performs better than Kemeny, and local Kemenisation (right) significantly improves most other rules.

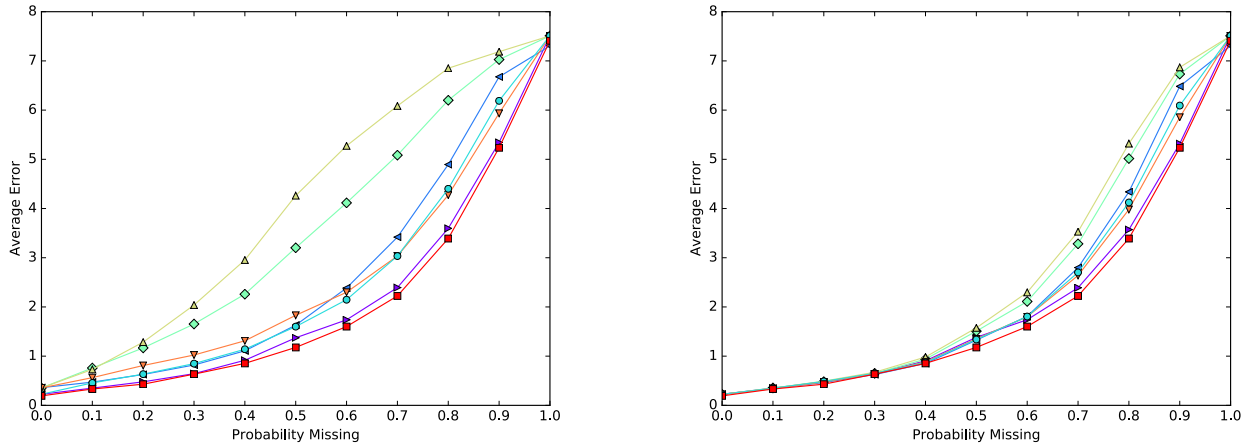


Figure 2: Missing candidates increase the average distance for all rules (6 candidates, $p = \frac{2}{3}$, and 25 agents).

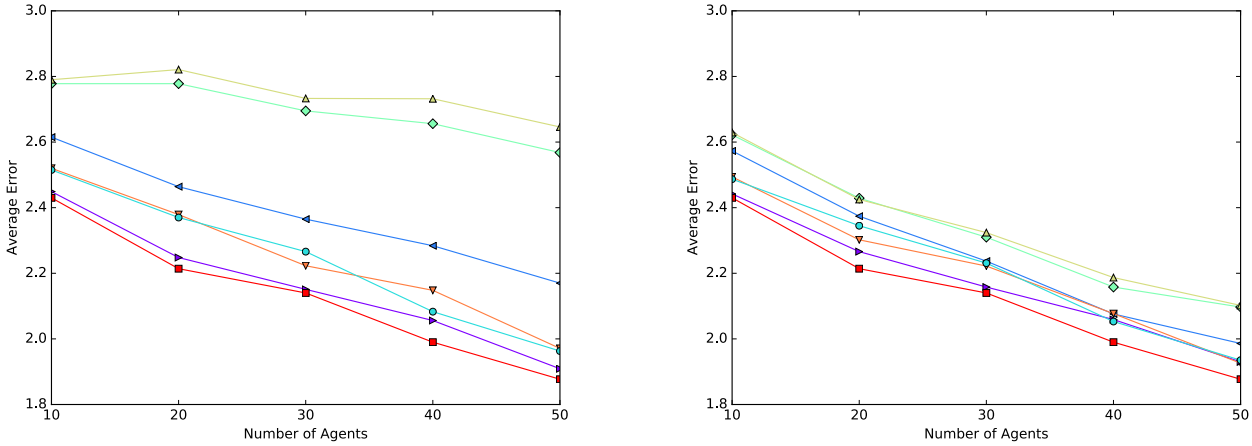


Figure 3: The relative performance of the rules on the Dots data set 1 with a probability of removing a candidate of 0.7 is similar to the synthetic data.

Table 1: The average distances for Optimal for different values of p are given on the Dots 1 problem instance for 10 agents (top) and 50 agents (bottom) and for probabilities 0.0–1.0 of missing candidates. Except for $p = 0.5$ these differences are statistically insignificant (standard errors of above 0.02).

p	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.5000	3.004	3.03	2.964	3.063	2.998	3.002	2.974	2.94	2.984	2.946	3.055
0.6225	1.615	1.679	1.815	1.965	2.054	2.192	2.315	2.446	2.691	2.815	3.055
0.7311	1.606	1.683	1.822	1.991	2.059	2.203	2.314	2.435	2.681	2.844	3.055
0.8176	1.605	1.685	1.82	1.989	2.057	2.207	2.335	2.439	2.705	2.81	3.055
0.8808	1.607	1.684	1.819	1.992	2.059	2.212	2.326	2.43	2.694	2.815	3.055
0.9241	1.611	1.69	1.824	1.997	2.061	2.208	2.346	2.447	2.68	2.813	3.055
0.9526	1.611	1.676	1.824	1.989	2.055	2.216	2.323	2.451	2.703	2.816	3.055
0.5000	2.946	2.936	3.006	3.083	3.023	2.978	3.016	2.986	3.069	3.125	2.964
0.6225	0.733	0.834	0.89	1.006	1.274	1.405	1.597	1.887	2.2	2.684	2.964
0.7311	0.731	0.833	0.898	1.02	1.285	1.429	1.627	1.877	2.185	2.67	2.964
0.8176	0.733	0.834	0.9	1.023	1.285	1.424	1.629	1.884	2.207	2.66	2.964
0.8808	0.734	0.834	0.9	1.023	1.284	1.427	1.628	1.89	2.189	2.7	2.964
0.9241	0.732	0.832	0.899	1.025	1.284	1.424	1.626	1.88	2.186	2.71	2.964
0.9526	0.731	0.834	0.901	1.024	1.284	1.429	1.627	1.885	2.197	2.653	2.964

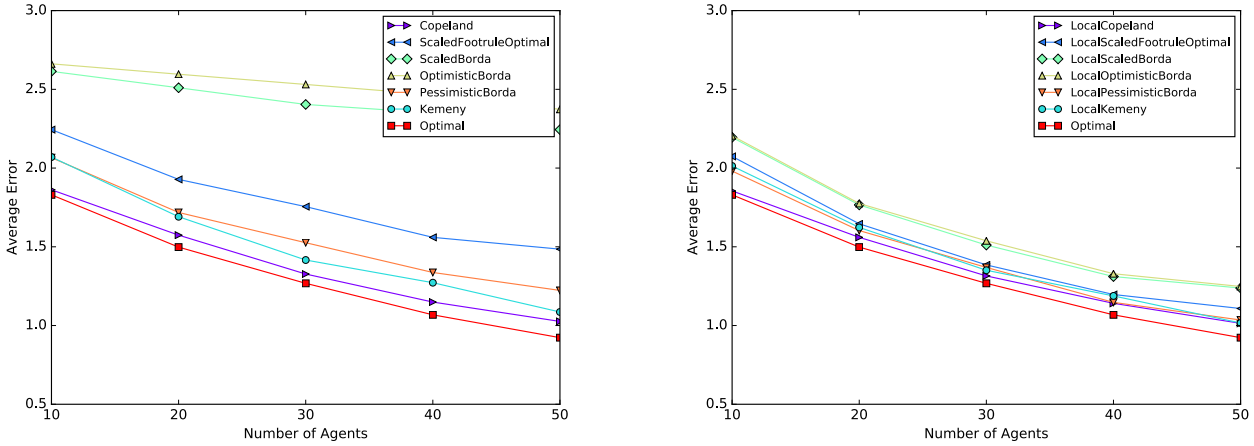


Figure 4: The performance of the rules on this Puzzle data set 2 with $q = 0.7$ shows the same trends as for Dots.

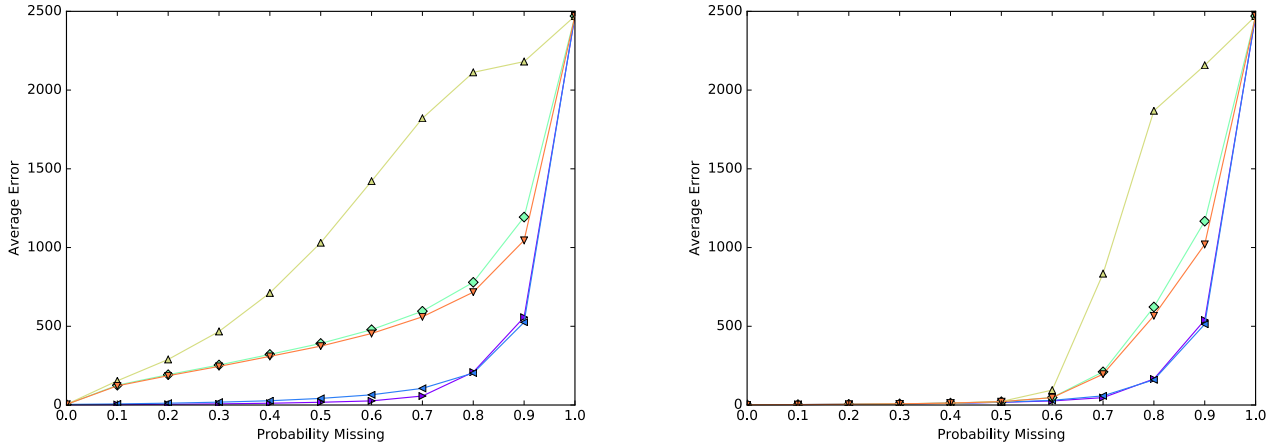


Figure 5: Local Kemenisation reduces the error drastically with 100 candidates, 40 agents and $p = \frac{2}{3}$.

We furthermore considered the effect of the unknown noise probability p on the performance of Optimal with real data. The results are shown in Table 1 for a range of q and p values, where the p values are computed using a generalised noise model [12] $p = \frac{e^\alpha}{1+e^\alpha}$ and $\alpha \in \{0, 0.5, \ldots, 3\}$. Interestingly, the results are not statistically significant for different values of p (except 0.5), suggesting that Optimal is robust with respect to the choice of p.

We also produced synthetic experiments for larger numbers of candidates (100) and more agents (up to 200). Although we were unable to compute Kemeny and Optimal rankings for these settings, the trends for the remaining voting rules were largely the same. A representative example in shown in Figure 5. As before, local Kemenisation significantly reduces the error.

5. RELATED WORK

The average Kendall-tau distance (among other measures) is often used in experiments with noisy and incomplete data to compare the effectiveness of a set of rules over a large set of problem instances, e.g., [11, 13]. However, this is typically done implicitly and without noticing that minimising this error is a significantly different objective from maximising the likelihood. The objective of minimising the error is first explicitly mentioned in a technical report [29, Section 5]. They say it is (more) difficult to optimise for this than maximising the likelihood (without proof) as "no closed form solution exists", which is in line with our hardness result. Later this statistical decision-theoretic viewpoint on social choice and the hardness proof have been formalised [27], but only considers complete votes.

A related argument against using maximum likelihood as the objective is that optimising for a single noise model may not be optimal in realistic settings, because the noise could take unpredictable forms [4, 23]. This has led to the design of a "modal" ranking rule that is robust against any "reasonable" noise model [4]. Following a similar argument, it is relevant to learn a mixture of (Mallows) noise models, e.g., through a Monte Carlo based approximation [21]. Such approaches can be seen as complementary to the direction we take in this paper. In fact, we argue that the ultimate objective should be to minimise the error based on a learned mixture of general noise models.

Specifically regarding incomplete votes, some work has considered machine learning techniques that view the missing ranking information as hidden variables, which are then inferred from other votes. For example, Cheng et al. [6] use the Expectation–Maximisation algorithm. Other experiments with real data show the performance of a number of "existing, standard, algorithms from machine learning" to infer the missing information [11]. For an overview of the workflow of designing social choice mechanisms using machine learning see the paper by Xia [30].

Much of the work in this area uses information retrieval as its main application domain (e.g., [13, 19]) and machine learning is used as an adaptive voting rule which learns how to rank the documents. An important challenge in this domain is scalability, especially for search engines, where the number of candidates (documents) can be as large as several billions. So far, this problem has mostly been approached as a machine learning classification task. However, other voting rules such as SFO, Borda, and a range of methods using a Markov decision model of the votes, have been evaluated on a web page data set [13]. Consistent with our findings, their results show that the rule said to be similar to Copeland (called MC_4) performs the best on this set. Importantly, however, the "ground truth" in such applications is not a full rank, but rather whether a document is relevant or not. As a result, the objective in those approaches is different (they are more concerned with measures such as recall and precision and other measures specifically relevant to information retrieval). Nevertheless, our results support the main conclusions from these papers in that the Copeland rule seems an appropriate choice for most levels of noise and missing candidates.

6. CONCLUSIONS

We have shown that voting rules which maximise the likelihood of a ranking do not necessarily minimise the rank aggregation error, i.e., the expected distance to the true ranking. Specifically, for rank aggregation with significant noise and missing votes, maximising the likelihood (i.e., using Kemeny's rule assuming Mallows' model) can result in a significantly higher error than computationally simpler methods such as Copeland. While the results are particularly pronounced with missing votes, we have shown that this discrepancy can occur even when votes are complete. Furthermore, we have shown that Optimal performs best in both synthetic and real data settings, even when we do not know the noise parameter exactly. In terms of theoretical results, for Mallows' model we have shown that computing this error is hard. Furthermore, we proved that an efficient procedure called local Kemenisation, which is known to improve the likelihood, also reduces the error, and that in fact this leads to a significant performance improvement for varying incompleteness and noise levels.

The next logical step is to design new voting rules with the objective of minimising the error in settings with incomplete and noisy observations. This would be particularly interesting for more general (mixtures of) noise models. These extensions also give rise to a number of questions regarding the complexity class of the problems of minimising the rank aggregation error. In particular, although we showed that computing the error is #P-hard, determining whether the complexity of finding the ranking with minimal error for the Mallows' model is also #P-hard is still an open problem. Other extensions include considering different incompleteness models (e.g., where the probability of missing depends on the position in the true ranking) and different distance measures (e.g., winner determination, top-k, or more general weighted measures). Additionally, it would be interesting to compare existing voting rules to approaches that apply machine learning methods, both through learning missing data [11], but also by directly applying classifiers as is common in the "learning to rank" information retrieval field [19].

Acknowledgments

This work has been partly supported by COST Action IC1205 on Computational Social Choice.

REFERENCES

[1] D. Baumeister, P. Faliszewski, J. Lang, and J. Rothe. Campaigns for lazy voters: Truncated ballots. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, pages 577–584, 2012.

[2] G. Brightwell and P. Winkler. Counting linear extensions is #P-complete. In *23rd ACM Symposium on Theory of Computation*, pages 175–181, 1991.

[3] I. Caragiannis, G. A. Krimpas, and A. A. Voudouris. Aggregating partial rankings with applications to peer grading in massive online open courses. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, pages 675–683, 2015.

[4] I. Caragiannis, A. D. Procaccia, and N. Shah. When do noisy votes reveal the truth? In *Proceedings of the fourteenth ACM conference on Electronic commerce*, pages 143–160. ACM, 2013.

[5] X. Chen, P. N. Bennett, K. Collins-Thompson, and E. Horvitz. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202. ACM, 2013.

[6] W. Cheng, J. Hühn, and E. Hüllermeier. Decision tree and instance-based learning for label ranking. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 161–168. ACM, 2009.

[7] V. Conitzer, A. Davenport, and J. Kalagnanam. Improved bounds for computing Kemeny rankings. In *AAAI*, volume 6, pages 620–626, 2006.

[8] V. Conitzer, M. Rognlie, and L. Xia. Preference Functions that Score Rankings and Maximum Likelihood Estimation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, volume 9, pages 109–115, 2009.

[9] V. Conitzer and T. Sandholm. Common voting rules as maximum likelihood estimators. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2005.

[10] J.-P. . P. Doignon, A. Pekeč, and M. Regenwetter. The repeated insertion model for rankings: Missing link between two subset choice models. *Psychometrika*, 69(1):33–54, 2004.

[11] J. A. Doucette, K. Larson, and R. Cohen. Conventional Machine Learning for Social Choice. In *Proceedings of the 29th AAAI Conference on Artificial Intellligence*, 2015.

[12] M. Drissi-Bakhkhat and M. Truchon. Maximum likelihood approach to vote aggregation with variable probabilities. *Social Choice and Welfare*, 23(2):161–185, 2004.

[13] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *10th Int. Conf. on World Wide Web*, pages 613–622. ACM, 2001.

[14] Z. Ghahramani and M. I. Jordan. Learning from incomplete data. Technical report, Massachusetts Institute of Technology, 1995.

[15] E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. *Theoretical Computer Science*, 349(3):382–391, 2005.

[16] M. G. Kendall. A new measure of rank correlation. *Biometrika*, pages 81–93, 1938.

[17] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proc. IJCAI-05 Multidisciplinary Workshop on Advances in Preference Handling*, volume 20, 2005.

[18] A. Kumar and M. Lease. Learning to rank from a noisy crowd. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1221–1222. ACM, 2011.

[19] T.-Y. . Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.

[20] T. Lu and C. Boutilier. Learning Mallows models with pairwise preferences. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 145–152, 2011.

[21] T. Lu and C. Boutilier. Effective sampling and learning for mallows models with pairwise-preference data. *The Journal of Machine Learning Research*, 15(1):3783–3829, 2014.

[22] C. L. Mallows. Non-null ranking models. I. *Biometrika*, pages 114–130, 1957.

[23] A. Mao, A. Procaccia, and Y. Chen. Better Human Computation Through Principled Voting. In *AAAI*, pages 1142–1148, 2013.

[24] N. Mattei and T. Walsh. PrefLib: A library of Preference Data. In *Algorithmic Decision Theory*, pages 259–270. Springer, 2013.

[25] M. Nicolas de Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. L'imprimerie royale, 1785.

[26] D. G. Saari. Mathematical structure of voting paradoxes: I. Pairwise votes. *Economic Theory*, 15(1):1–53, 2000.

[27] H. A. Soufiani, D. C. Parkes, and L. Xia. A statistical decision-theoretic framework for social choice. In *Advances in Neural Information Processing Systems*, pages 3185–3193, 2014.

[28] M. Truchon. An extension of the Condorcet criterion and Kemeny orders. Technical report, 1998.

[29] K. Tsukida and M. R. Gupta. How to analyze paired comparison data. Technical report, DTIC Document, 2011.

[30] L. Xia. Designing social choice mechanisms using machine learning. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems*, pages 471–474, 2013.

[31] L. Xia and V. Conitzer. A maximum likelihood approach towards aggregating partial orders. In *Proceedings of the 22th International Joint Conference on Artificial Intelligence*, pages 446–451, 2011.

[32] P. Young. Optimal voting rules. *The Journal of Economic Perspectives*, 9(1):51–64, 1995.

Preferences Single-Peaked on Nice Trees

Edith Elkind
University of Oxford
Oxford, UK
elkind@cs.ox.ac.uk

Dominik Peters
University of Oxford
Oxford, UK
dominik.peters@cs.ox.ac.uk

ABSTRACT

Preference profiles that are single-peaked on trees enjoy desirable properties: they admit a Condorcet winner (Demange 1982), and there are hard voting problems that become tractable on this domain (Yu et al. 2013). Trick (1989) proposed a polynomial-time algorithm that finds *some* tree with respect to which a given preference profile is single-peaked. However, some voting problems are only known to be easy for profiles that are single-peaked on "nice" trees, and Trick's algorithm provides no guarantees on the properties of the tree that it outputs. To overcome this issue, we build on the work of Trick and Yu et al. to develop a structural approach that enables us to compactly represent all trees with respect to which a given profile is single-peaked. We show how to use this representation to efficiently find the "best" tree for a given profile, according to a number of criteria; for other criteria, we obtain NP-hardness results. In particular, we show that it is NP-hard to decide whether an input profile is single-peaked with respect to a given tree. To demonstrate the applicability of our framework, we use it to identify a new class of profiles that admit an efficient algorithm for a popular variant of the Chamberlin–Courant (Chamberlin and Courant 1983) rule.

1. INTRODUCTION

Preference aggregation is a difficult task when voters' preferences may be arbitrary: one has to deal with voting paradoxes (Arrow 1951) and computationally hard problems (Brandt et al. 2013). This observation motivates the study of *domain restrictions*, i.e., special classes of voters' preferences that rule out paradoxical outcomes and/or allow one to circumvent computational hardness results. Perhaps the most-studied restricted domain is that of single-peaked preferences (Black 1948). This domain captures profiles where voters' preferences are determined by candidates' positions on a single issue, and has many desirable properties: for instance, single-peaked elections always have a Condorcet winner (a candidate that is preferred to every other candidate by a majority of voters), admit a non-manipulable voting rule (Moulin 1991), and allow for an efficient winner determination algorithm for a popular committee selection rule (Betzler et al. 2013).

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

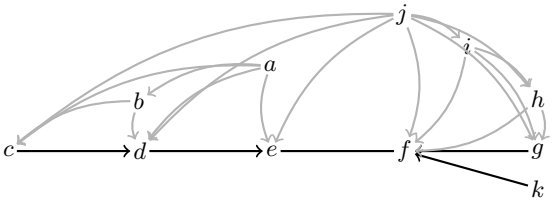


Figure 1: The 'attachment digraph' computed for the profile {*kfedghcijba, dcbeafghijk, gfhiedcbajk*} which is single-peaked on 336 different trees. All of them appear as subtrees of the above digraph, which can be computed in $O(|V| \cdot |C|^2)$ time. The black edges must appear in any tree; for each *free* vertex with gray outgoing arcs, we choose exactly one of them. The transitivity among the gray edges will be crucial to our approach.

Demange (1982) introduced a weaker domain restriction, namely, single-peakedness on a tree. Briefly, a profile is single-peaked on a tree if candidates can be mapped to the vertices of some tree so that the restriction of this profile to every path in this tree is single-peaked. This is a considerably broader domain than that of the single-peaked elections, which, nevertheless, retains some of the desirable properties of the latter: profiles that are single-peaked on a tree always have a Condorcet winner (Demange 1982), and there are committee selection problems that become easier when preferences are single-peaked on a tree (Yu et al. 2013). However, some of the algorithms for this domain require the input profile to be single-peaked on a "nice" tree, such as a tree with a small number of leaves or a star (Yu et al. 2013); indeed, positive results for single-peaked preferences can also be viewed in this light, as they require the preferences to be single-peaked on a specific tree, namely, a line (path).

Now, forcing Trick's algorithm to output a "nice" tree is not a trivial task: indeed, this algorithm may output a complex tree even when the input profile is single-peaked on a line. Fortunately, there are efficient algorithms for recognizing when a given profile is single-peaked on a line (Bartholdi and Trick 1986; Doignon and Falmagne 1994; Escoffier et al. 2008), and Yu et al. (2013) explain how to modify Trick's algorithm so that it outputs a tree with the minimum number of leaves and how to recognise when a profile is single-peaked on a star. However, prior to this work, no such algorithms were known for other types of "nice" trees, such as trees that have bounded diameter or a small number of internal nodes;

in fact, it was an open question whether, given a profile V and a tree T, one can check in polynomial time whether V is single-peaked on T.

In this paper, we propose a general framework for answering such questions, and use it to obtain polynomial-time algorithms for identifying "nice" trees when they exist, for several appealing notions of "niceness". Specifically, we define a digraph that encodes, in a compact fashion, all trees with respect to which a given profile is single-peaked, an example is shown in Figure 1. This digraph enables us to count and/or enumerate all such trees. Moreover, we show that it has many useful structural properties, which can be exploited to efficiently find trees that have, e.g., the minimum degree, diameter, or number of internal nodes among all trees with respect to which a given profile is single-peaked, or to decide if a given profile is single-peaked on some specific type of tree, such as a caterpillar or a subdivision of a star (see Section 2 for definitions). However, there are limits to what we can accomplish in this way: we show that it is NP-hard to decide whether a given profile is single-peaked on a regular tree. Moreover, given a profile and a tree, it is NP-hard to decide if this profile is single-peaked on this specific tree.

Knowing whether a profile is single-peaked on a "nice" tree enables us to develop stronger intuition about the structure of voters' preferences. However, our recognition algorithms also have more tangible benefits: for at least one type of "nice" trees that can be identified by our algorithm (namely, trees with few internal vertices), we develop a winner determination algorithm for a variant of the Chamberlin–Courant committee selection rule (Chamberlin and Courant 1983) that, under plausible complexity assumptions, is more efficient than any algorithm for this rule under general preferences. Importantly, our winner determination algorithm works directly with the underlying tree, i.e., it relies on having an efficient procedure for constructing a tree with few internal vertices. We expect that similar results can be obtained for other types of trees that we can recognise; we leave this question for future work.

2. PRELIMINARIES

For a finite set of *candidates* C, a *profile* V over C is a list of strict total orders over C; elements of V are called *votes*, or *preference orders*. We will view a vote $v \in V$ as a list of candidates, and use `Python`-like indices to refer to its entries: $v_{[1]}$, $v_{[2]}$, $v_{[-1]}$ are v's most, second-most, and least preferred candidates respectively, and $v_{[1:k]}$ is the set of v's k most preferred candidates. For readability, we write $c' \succ_v c$ to indicate that c' comes before c in the vote v. For a subset $C' \subseteq C$ of candidates, we denote by $v|_{C'}$ the preference order obtained from v by restricting it to $C' \times C'$.

A *tree* is a connected acyclic graph. A *leaf* of a tree is a vertex of degree 1. Vertices that are not leaves are *internal*. A *path* is a tree with exactly two leaves. The *diameter* of a tree T is the number of edges in a longest simple path in T. A *star* is a tree $K_{1,n}$ that has one internal vertex (the *center*) and n leaves. A *k-regular tree* is a tree in which every internal vertex has degree k. Note that paths are 2-regular (uniquely), and the star $K_{1,n}$ is n-regular. A *caterpillar* is a tree in which every vertex is within distance 1 of a central path; in other words, removing all leaves from a caterpillar results in a path. A *lobster* is a tree in which every vertex is within distance 2 of a central path. A *subdivision of a star* is a tree obtained from a star by replacing its edges by paths.

A *tree decomposition* of a graph G is given by a tree T and map $\beta : V(T) \to \mathcal{P}(V(G))$ that associates each vertex v of T with a *bag* $\beta(v) \subseteq V(G)$ of vertices of G, in such a way that for each $w \in V(G)$, the set $\beta^{-1}(w)$ is non-empty and connected in T, and for each edge $\{u, w\} \in E(G)$, there is a bag $\beta(v)$ with $\{u, w\} \subseteq \beta(v)$. The *width* of a tree decomposition (T, β) is given by $\max_{v \in V(T)} |\beta(v)| - 1$. A tree decomposition (T, β) is called a *path decomposition* if T is a path. The *path-width* of a graph G is the minimum width of a path decomposition of G. See Bodlaender (1994) for a survey of these notions.

A profile V over C is *single-peaked (on a line)* if there is a linear order $\sqsubset$ on C such that for every $v \in V$, we have $x \succ_v y$ whenever $v_{[1]} \sqsubset x \sqsubset y$ or $y \sqsubset x \sqsubset v_{[1]}$. V is *single-peaked on* T, where T is a tree with vertex set C, if V is single-peaked when restricted to the vertex set of every path in T. Equivalently, V is single-peaked on T if for every $v \in V$ and each $k = 1, \dots, |C|$, the set $v_{[1:k]}$ induces a subtree of T. Given a profile V, we denote the set of all trees T such that V is single-peaked on T by $\mathcal{T}(V)$; we say that trees in $\mathcal{T}(V)$ are *suitable* for V. A profile V over C is *single-peaked on a tree* if $\mathcal{T}(V) \neq \varnothing$. In particular, V is single-peaked on a line if and only if it is single-peaked on a tree T that is a path.

A *digraph* $D = (\mathcal{V}, A)$ is a directed graph with no self-loops or multiple arcs. An arc $(u, v) \in A$ points from its *tail* u to its *head* v. We will write uv for (u, v). An *acyclic* digraph (a *dag*) is a digraph with no directed cycles. For a vertex v, its *out-degree* (resp., *in-degree*) $d^+(v)$ (resp., $d^-(v)$) is the number of arcs whose tail (resp., head) is v. A *sink* is a vertex v with $d^+(v) = 0$, a *source* is a vertex with $d^-(v) = 0$. Every dag has at least one sink and one source.

Given a digraph $D = (\mathcal{V}, A)$, we can *forget about its orientation* to obtain an underlying graph $G = (\mathcal{V}, E)$ where $\{u, v\} \in E$ if and only if $uv \in A$ or $vu \in A$.

3. THE ATTACHMENT DIGRAPH

We now introduce the essential tool in our study of the set $\mathcal{T}(V)$. Consider a profile V over a candidate set C that is single-peaked on *some* tree: $\mathcal{T}(V) \neq \varnothing$. We associate to this profile its *attachment digraph* D with vertex set C: this digraph is obtained by running Algorithm 1, which builds on the ideas of Trick (1989) and Yu et al. (2013).

Algorithm 1 Build attachment digraph $D = (C, A)$ of V

```
D ← (C, A), A ← ∅          ▷ D is the empty digraph on C
C' ← C
while |C'| ⩾ 3 do
    L ← {(v|_{C'})_[-1] : v ∈ V}
    for each candidate c ∈ L do
        B_c = ⋂_{v∈V} B(v|_{C'}, c)
        if B_c = ∅ then
            return fail   ▷ V not single-peaked on any tree
        else
            add arcs cc' for each c' ∈ B_c to A
    C' ← C' \ L
return D
```

This algorithm runs in time $O(|V| \cdot |C|^2)$. It uses the operator $B(v, c)$ that takes as input a vote v and a candidate c, and returns a constraint on the candidates c' that c can be

attached to in suitable trees as a leaf. It is defined as

$$B(v,c) = \begin{cases} \{c' : c' \succ_v c\} & \text{if } v_{[1]} \neq c, \\ \{v_{[2]}\} & \text{if } v_{[1]} = c. \end{cases}$$

This operator can be computed in time $O(|C|)$. To see that the definition makes sense, suppose that c is a leaf. If it is not the top candidate in vote v, then it must be attached to some c' that v prefers to c, since every path from $v_{[1]}$ to c goes through the parent of c. On the other hand, if c is the top candidate in v, then c must be attached to the candidate c' ranked immediately below c by v, as otherwise the path from c to $v_{[2]}$ violates the single-peakedness condition.

We start with a few easy properties of attachment digraphs.

PROPOSITION 1. *Any attachment digraph (C, A) is acyclic and has at most two sinks. If it only has one sink t, then $d^-(t) \geqslant 2$.*

PROOF. We follow the proof of Theorem 6.1 in the work of Yu et. al. (2013). Suppose that the while loop is executed $f-1$ times, and denote the sets L found at each iteration by $L_1, \ldots, L_{f-1}$. Set $L_f := C \setminus (L_1 \cup \cdots \cup L_{f-1})$. Then $L_1, \ldots, L_f$ is a partition of C. Since for every $c \in C \setminus L_f$ we have $B_c \neq \varnothing$, at least one arc with tail c is added to A. Hence no $c \in C \setminus L_f$ can be a sink, so all sinks are in L_f. The condition of the while loop implies that $|L_f| \leqslant 2$, so there are at most two sinks. It also implies that $|L_{f-1} \cup L_f| \geqslant 3$, which gives $d^-(t) \geqslant 2$.

For acyclicity, note that since $B(v|_{C'}, c) \subseteq C'$, we have $B_c \subseteq C'$ at the stage when we compute B_c. Thus if $c \in L_i$ then all arcs with tail c point into $L_{i+1} \cup \cdots \cup L_f$. So (C, A) has a topological order given by a linearisation of the partial order induced by the partition $L_1, \ldots, L_f$ of C. □

Often, it will be convenient to use the *pointed attachment digraph* D^+ of a profile, which is obtained from D by inserting a single arc (arbitrarily directed) between the two sinks in case there are two sinks. Thus, D^+ always has exactly one sink.

We call a subset $F \subseteq A$ of the arcs of a digraph an *arc-function* if every vertex that is not a sink has exactly one outgoing arc in F. The reason we are interested in attachment digraphs and their arc functions is given in the following theorem, which follows from results of Trick (1989).

THEOREM 2. *For every profile V, the set $\mathcal{T}(V)$ is in bijection with the set of arc-functions for D^+, the pointed attachment digraph of V. Thus, every tree in $\mathcal{T}(V)$ appears as a subgraph of D^+, once we forget about its orientation.*

Indeed, given an arc-function F for the pointed attachment digraph, we can take the arcs in F, forget about their orientation, and obtain a suitable tree for V.

COROLLARY 3. *The number of suitable trees in $\mathcal{T}(V)$ is equal to the product of the out-degrees of the non-sink vertices of D^+. Hence we can compute $|\mathcal{T}(V)|$ in polynomial time.*

It turns out that attachment digraphs have a lot of structure beyond the results of Proposition 1. A key property, which will allow us to use essentially greedy algorithms, is what we call circumtransitivity.

DEFINITION 1. *A dag $D = (C, A)$ is* circumtransitive *if its vertices can be partitioned into a set $C^{\rightarrow}$ of* forced *vertices and a set $C^{\rightrightarrows} = C \setminus C^{\rightarrow}$ of* free *vertices so that*

1. *every forced vertex has out-degree at most one, and all vertices reachable from it are also forced, and*
2. *every free vertex has out-degree at least two, and whenever x and y are free vertices with $xy, yz \in A$, then $xz \in A$.*

Note that in particular the sinks of D are forced. A circumtransitive dag thus consists of an inner part (the *forced part*) that even after forgetting about orientations is acyclic, and an outer part that is transitively attached to the inner part. Further, since every free vertex starts a path to a sink, by transitivity it must have an arc to some forced vertex.

THEOREM 4. *Every attachment digraph (C, A) is circumtransitive.*

PROOF. We will argue that Definition 1 is satisfied by taking the partition

$$C^{\rightarrow} = \{c : d^+(c) \leqslant 1\}, \quad C^{\rightrightarrows} = \{c : d^+(c) \geqslant 2\}.$$

Forced: Let $x \in C^{\rightarrow}$ be a forced vertex. If $d^+(x) = 0$, there is nothing to prove, so assume that $d^+(x) = 1$, i.e., $xy \in A$ for some $y \in C$. We will show that $d^+(y) \in \{0, 1\}$.

If y is a sink, we are done, so suppose $yz \in A$ for some $z \in C$. Then $xz \notin A$ and so $z \notin B_x$. This means that $z \notin B(v|_{C'}, x)$ for some $v \in V$, where C' is the set of candidates remaining in the while-loop iteration in which x is attached. Note that $y \in C'$ and hence $z \in C'$ as well. Write $\bar{v} = v|_{C'}$. We consider two cases:

(i) $x \neq \bar{v}_{[1]}$. Then $xy \in A$ implies $y \succ_v x$. Consider now the iteration in which y is attached. If y is ranked first in v at that point, then by construction $|B_y| = 1$, so we are done. Otherwise $yz \in A$ implies $z \succ_v y$. But then by transitivity $z \succ_v x$, so $z \in B(\bar{v}, x)$, a contradiction.

(ii) $x = \bar{v}_{[1]}$. Then $xy \in A$ implies that y is ranked second in $\bar{v}$. Hence, in the iteration in which y is attached, it must be the top candidate in v, and therefore $|B_y| = 1$.

Free: Consider vertices $x, y, z \in C$ with $x, y \in C^{\rightrightarrows}$ and $xy, yz \in A$. At x's attaching time, no vote can begin in x since $d^+(x) > 1$. Thus since $xy \in A$ we have $y \succ_v x$ for all $v \in V$. Similarly, since $yz \in A$ and $d^+(y) > 1$ we have $z \succ_v y$ for all $v \in V$. Hence, by transitivity, $z \succ_v x$ for all $v \in V$, so $z \in B_x$ and $xz \in A$. □

It follows that every tree in $\mathcal{T}(V)$ contains the forced part of D^+ as a subtree (after forgetting orientations).

Finally, let us study the free vertices $C^{\rightrightarrows}$ more closely.

PROPOSITION 5. *Every free vertex of $D^+ = (C, A)$ has arcs to at least two forced vertices.*

PROOF. We have observed that every free vertex has an arc to at least one forced vertex. Assume for the sake of contradiction that there is a free vertex x that only has a single arc to a forced vertex; let this forced vertex be z. Take a topological ordering of D^+; among all free vertices y such that $yz \in A$ but $yz' \notin A$ for every $z' \in C^{\rightarrow} \setminus \{z\}$, let v be the one that minimises the distance to z in the topological order. Since v is free, it also has an arc to some other vertex w, which is free by the choice of v. Now, w has an arc to some forced vertex z'. By transitivity we have $vz' \in A$, and hence $z' = z$. As w appears between v and z in the topological order, this contradicts minimality of the distance between v and z. □

PROPOSITION 6. *For each free vertex $x \in C^{\rightrightarrows}$ of $D^+ = (C, A)$, the set $\{y \in C^{\rightarrow} : xy \in A\}$ induces a subtree in the undirected version of D^+.*

PROOF. Suppose A contains arcs xy and xz where $y, z \in C^{\rightarrow}$. Let P be the unique y–z path in the undirected version of D^+ that is contained in $C^{\rightarrow}$, and let C_P be its vertex set. We will argue that $C_P \subseteq B_x$. Fix a tree $T \in \mathcal{T}(V)$; note that P is a path in T. Pick a vote $v \in V$. Since $|B_x| > 1$, we have $y \succ_v x, z \succ_v x$. Take a top segment of v that includes y and z, but not x. This set induces a subtree of T, and hence contains C_P. Thus, $w \succ_v x$ for each $w \in C_P$. As this holds for each $v \in V$, x can be attached to any vertex of C_P. □

Taken together, Propositions 5 and 6 imply that each free vertex can be attached to two forced vertices that are adjacent to each other in D^+.

4. RECOGNITION ALGORITHMS

Suppose we are given a profile V with $\mathcal{T}(V) \neq \varnothing$ and wish to find trees in $\mathcal{T}(V)$ that satisfy additional desiderata. As argued above, this can be done by computing the attachment digraph and then choosing the arc-function appropriately. Next, we give examples to that effect; the list is not meant to be exhaustive and, in addition to new results, includes already known results for paths, stars, and trees with few leaves.

THEOREM 7. *Given a profile V that is single-peaked on a tree, we can find in polynomial time a suitable tree that among the trees in $\mathcal{T}(V)$ has a*

1. *minimum number of leaves,*
2. *minimum number of internal vertices,*
3. *minimum diameter,*
4. *minimum max-degree,*
5. *minimum path-width.*

Further, we can decide in polynomial time whether a given profile is single-peaked on a

6. *line,*
7. *star,*
8. *caterpillar,*
9. *lobster,*
10. *subdivision of a star.*

PROOF SKETCHES. Fix a profile V over a candidate set C, $|C| \geqslant 3$, and compute D, the attachment digraph of V, and D^+, the pointed attachment digraph.

1. This has already been done by Yu et al. (2013). Their algorithm can be phrased in the language of attachment digraphs as follows: first we find a maximum partial arc-function such that no vertex has two incoming arcs, and then we extend it to a full arc-function arbitrarily. The first step can be achieved by matching techniques (as explained by Yu et al.) or by matroid intersection of two partition matroids (Gabow and Tarjan 1984).

2. We only need to decide how to attach free vertices. By Propositions 5 and 6, each $x \in C^{\rightrightarrows}$ can be attached to two forced vertices that are adjacent to each other. Thus, if $|C^{\rightarrow}| > 2$, we can attach x to a forced vertex that is internal in the forced part. Then every leaf of the forced part remains a leaf, and every free vertex becomes a leaf, which is clearly optimal. If $|C^{\rightarrow}| = 2$ (note that $|C^{\rightarrow}| \geqslant 2$ by Proposition 5), we can pick $y \in C^{\rightarrow}$ and attach all free vertices to y.

3. Run the algorithm of part 2; this produces a tree of the same diameter as the forced part if $|C^{\rightarrow}| = 2$ or else of diameter one larger than the forced part, which must be minimal.

4. We show how to find a suitable tree with max-degree at most k if there is one, for each fixed k. By repeatedly calling this algorithm with $k = 2, 3, \ldots, |C| - 1$, we find a tree of minimum max-degree. The algorithm is similar to part 1: we check whether there is an arc-function for D such that no vertex has more than $k - 1$ incoming arcs in the arc-function. Such an arc-function, if it exists, can be found using flow techniques or by matroid intersection of two partition matroids.

5. The path-width of any suitable tree is at least the path-width of the forced part. For trees, we can find a path decomposition of minimum width in linear time (Scheffler 1990). Run this algorithm on the forced part. Then attach free vertices to forced vertices, prioritising forced vertices that appear in a bag that is not of maximum cardinality among the bags in the path decomposition. If such a bag is found, duplicate it and add the free vertex to one of the copies. If some free vertex v is only attachable to forced vertices that only appear in maximum-cardinality bags, we need to fix things up. Take two different adjacent forced vertices w and x to which v is attachable (such vertices are guaranteed to exist). Take the left-most bag in which w and x appear together and duplicate it. Suppose that w does not appear to the left of the left copy. Then replace w by v in that copy and attach v to x. This yields a path-decomposition of width equal to the path-width of the forced part, which is optimal.

6. The fastest algorithms for the line are given by Doignon and Falmagne (1994) and Escoffier et al. (2008); they do not rely on the pre-computation of the attachment digraph and are thus faster than anything we can offer. That said, running algorithms for parts 1 or 4 above will return a path if possible, and if not, return something as close to a path as possible (according to the various senses).

7. As observed by Demange (1982) and Yu et al. (2013), a profile is single-peaked on a star if and only if there exists a candidate that is ranked first or second in every vote. Hence the profiles single-peaked on a star form a regular language (under sensible encodings) and can be quickly recognised. In our framework we can look for a sink in D such that every non-sink vertex points to it.

8. A given profile can only be single-peaked on a caterpillar if the forced part of D^+ is a caterpillar. If so, use the algorithm from part 2 to make every free vertex a leaf. The result is still a caterpillar. The lobster case (9) is similar.

10. The forced part is either a path or a subdivision of a star $K_{1,s}$ with $s > 2$. In the first case, it suffices to check whether all free vertices attach to a single vertex of this path (which then becomes the center). In the second case, first attach as many free vertices to the center as possible. By Proposition 6, the remaining free vertices can be attached to at most one of the leaves of the forced subdivision of a star. Then for each leaf separately, find a longest path in the sub-dag of free vertices attachable to it, and check that the union of these longest paths contains all the remaining free

vertices. □

Applying these algorithms to the example in Figure 1, we see that suitable trees have between three and eight leaves, their diameter at least four, they can have max-degree three and path-width one, and include a caterpillar but not a subdivision of a star.

5. HARDNESS RESULTS

The algorithms in the proof of Theorem 7 enable us to answer a wide range of questions about the set $\mathcal{T}(V)$. The NP-hardness results in this section, however, show that it is likely that not every such question can be answered efficiently.

Consider the following computational problem.

SINGLE-PEAKED TREE LABELLING
Instance: Profile V over C, unlabelled tree T on $|C|$ vertices
Question: Is there a labelling of the vertices of T with candidates in C s.t. V is single-peaked on that labelled tree?

THEOREM 8. *Problem* SINGLE-PEAKED TREE LABELLING *is* NP*-complete even if the input trees are restricted to diameter at most four or to max-degree at most three.*

PROOF. The problem is in NP since for a given labelling we can easily check whether it makes the profile single-peaked on T.

For the hardness proof, we reduce from X3C. Given an X3C-instance with objects $x_1, \ldots, x_{3m}$ and sets $s_1, \ldots, s_n$ we construct a tree T by taking a star $K_{1,n}$ and attaching three fresh leaves to exactly m of the leaves of the star. Then T has diameter four. We construct a profile over the candidate set $\{\star, x_1, \ldots, x_{3m}, s_1, \ldots, s_n\}$, with one vote for each object and for each set. In the following, all indifferences can be resolved arbitrarily.

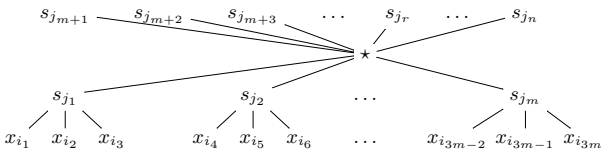
For each object x_i, vote to force x_i to be attached to $\star$ or to a set containing x_i:

$$\star, \{\text{sets containing } x_i\}, x_i, \{\text{other sets}\}, \{\text{other objects}\}.$$

For each set s_j, vote to force an edge from $\star$ to s_j:

$$\star, s_j, \{\text{sets other than } s_j\}, \{\text{objects}\}.$$

If there is a valid partition in the X3C-instance selecting precisely the sets $s_{j_1}, \ldots, s_{j_m}$, then we can label T as follows: The center of the star is $\star$. Its n neighbours are the s_js. We assign the s_{j_i}s to the neighbours of $\star$ of degree four. Then we can assign each object to its set s_{j_i} in the X3C-solution. By considering top-initial segments of the votes given above, we see that this makes all votes single-peaked on this tree.



If there is a labelling of T making all the votes single-peaked, then there must be an X3C-solution. To see this, first note that the vertex labelled $\star$ must have degree at least n because of the set votes. There is only one such vertex in T, namely the center, which is thus labelled $\star$. It has exactly n neighbours, which then must all be labelled by some s_j. This leaves us to decide which vertices are labelled by which objects. Since we have already labelled all neighbours of the center, the objects must be attached to s_js. Hence by the constraints of the object votes, the labelling induces an X3C-solution. □

By copying the center vertex and adding some peripheral vertices, we can adjust this reduction so that T's maximum degree is three. Notice that the problem is (trivially) fixed-parameter tractable with parameter $k = |C|$ by just trying all $k!$ possible labellings of the input tree.

We can use a similar reduction to prove a hardness result complementing the easiness results of Theorem 7 (we omit the proof). Recall that a tree is k-regular if every non-leaf vertex has degree k.

THEOREM 9. *Given a profile V, it is NP-complete to decide whether there exists a positive integer k such that V is single-peaked on a k-regular tree. The problem is also hard for each fixed $k \geqslant 4$.*

6. APPLICATION: COMMITTEE SELECTION

We will now demonstrate how to apply one of the recognition algorithms presented in the proof of Theorem 7 to obtain an algorithm for a committee selection problem that is known to be NP-hard for unrestricted preferences. Specifically, the computational problem we consider is winner determination under the Chamberlin–Courant rule with Borda misrepresentation function; our results extend to other misrepresentation functions that satisfy a mild condition. Our algorithm can be used for any profile that is single-peaked on a tree and is efficient for trees that have few internal vertices provided that the target committee size is not too large.

We start by defining the Chamberlin–Courant rule and providing a brief summary of complexity results for it, followed by the description of our algorithm and a proof of correctness.

Chamberlin and Courant (1983) propose a family of rules that take a candidate set C, a profile V over this set and a target committee size k as an input, and output a subset of candidates (committee) of size k. Given a candidate set C, $|C| = m$, every vector $\mathbf{s} = (s_1, \ldots, s_m)$ of non-negative integers with $0 = s_1 \leqslant \ldots \leqslant s_m$ defines a *positional misrepresentation function* $\mu_{\mathbf{s}} : V \times 2^C \to \mathbb{Z}$ as follows: $\mu_{\mathbf{s}}(v, C') = s_i$ if v ranks her most preferred candidate in C' in position i. The (utilitarian version of the) Chamberlin–Courant rule outputs some committee C' of size k that minimises the quantity $\sum_{v \in V} \mu_{\mathbf{s}}(v, C')$ (which we call the $\mathbf{s}$*-score* of C') over all size-k subsets of C. The misrepresentation function associated with the vector $\mathbf{s} = (0, 1, \ldots, m-1)$ is known as the *Borda misrepresentation function*.

Finding a winning committee for the Chamberlin–Courant rule is known to be NP-hard, even for the Borda misrepresentation function (Lu and Boutilier 2011); however, this problem can be solved in polynomial time for an arbitrary misrepresentation function if the input profile is single-peaked (Betzler et al. 2013) or, more broadly, has bounded single-peaked width (Cornaz et al. 2012), as well as for a large class of misrepresentation functions including the Borda misrepresentation function if the input profile is single-peaked on a star (Yu et al. 2013). Yu et al. also provide an algorithm for profiles that are single-peaked on a tree, which works for arbitrary misrepresentation functions; its running time is polynomial in $|V|$ and the quantities $|C|^\lambda$ and k^λ, where λ is the number of leaves of a suitable tree (they also explain how to find a suitable tree with the minimum number of leaves).

While the latter algorithm is useful for profiles that are single-peaked on a tree with few *leaves*, we will now present an

algorithm that is tailored for profiles that are single-peaked on trees with few *internal vertices*. It is inspired by Yu et al.'s algorithm for preferences that are single-peaked on a star.

THEOREM 10. *Given a candidate set C, $|C| = m$, a profile V over C, $|V| = n$, a tree $T \in \mathcal{T}(V)$ with η internal vertices such that V is single-peaked on T, and a target committee size $k \geqslant 1$, we can find a winning committee of size k for (C, V) under the Chamberlin–Courant rule with the Borda misrepresentation function in time* $\mathrm{poly}(n, m, (k+1)^\eta)$.

PROOF. Given a candidate $c \in C$, let $f(c)$ be the number of voters in V that rank c first, and let C° be the set of candidates that correspond to the internal vertices of T. For each candidate $c \in C^\circ$, let $\mathrm{ch}(c)$ denote the set of leaf candidates in $C \setminus C^\circ$ that are adjacent to c in T.

Our algorithm proceeds as follows. For each candidate $c \in C^\circ$ it guesses a pair $(b(c), \ell(c))$, where $b(c) \in \{0, 1\}$ and $0 \leqslant \ell(c) \leqslant k$: $b(c)$ indicates whether c itself is in the committee and $\ell(c)$ indicates how many candidates in $\mathrm{ch}(c)$ are in the committee. We require $\sum_{c \in C^\circ}(b(c) + \ell(c)) = k$. Next, it sets $C' = \{c \in C^\circ : b(c) = 1\}$, and then for each $c \in C^\circ$ it orders the candidates in $\mathrm{ch}(c)$ in non-increasing order of $f(c)$ (breaking ties according to a fixed ordering $\rhd$ over C), and adds the first $\ell(c)$ candidates in this order to C'.

Each guess corresponds to a committee of size k. Guessing can be implemented deterministically: consider all options for the collection $\{(b(c), \ell(c))\}_{c \in C^\circ}$ (there are at most $2^\eta \cdot (k+1)^\eta$ possibilities), compute the score of the resulting committee for each option, and output the best one.

It remains to argue that this algorithm finds a committee with the minimum Borda score. To see this, let $\mathcal{S}$ be the set of all size-k committees with the minimum Borda score, and pick a committee S^* from $\arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$, breaking ties according to $\rhd$ (note that this means that there is no set $S \in \arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$ such that $S^* \setminus S = \{c\}$, $S \setminus S^* = \{c'\}$ and $c' \rhd c$). For each $c \in C^\circ$, let $b^*(c) = 1$ if $c \in S^*$ and $b^*(c) = 0$ otherwise, and let $\ell^*(c) = |\mathrm{ch}(c) \cap S^*|$. Our algorithm will consider the collection $\{(b^*(c), \ell^*(c))\}_{c \in C^\circ}$ at some point, and output a committee S. We will now argue that $S = S^*$.

Indeed, we have $C^\circ \cap S = C^\circ \cap S^*$, so it remains to argue that $\mathrm{ch}(c) \cap S^* = \mathrm{ch}(c) \cap S$ for each $c \in C^\circ$. Suppose for the sake of contradiction that this is not the case, i.e., there exists a $c \in C^\circ$ and a pair of candidates $c', c'' \in \mathrm{ch}(c)$ with $c' \in S \setminus S^*$, $c'' \in S^* \setminus S$. If $c \in S^*$, consider the committee $S' = (S^* \setminus \{c''\}) \cup \{c'\}$. We claim that S' has the same Borda score as S^*. Indeed, the voters who do not rank c' or c'' first prefer c to either of these two candidates, so they are unaffected by the change, the misrepresentation of the $f(c'')$ voters who rank c'' first changes from 0 to 1, the misrepresentation of the $f(c')$ voters who rank c' first changes from 1 to 0, and we have $f(c') \geqslant f(c'')$ by construction of S. As we also have $c' \rhd c''$ by construction of S, this contradicts our choice of S^* from $\arg\max_{S' \in \mathcal{S}} |S' \cap C^\circ|$.

Now, suppose that $c \notin S^*$. Consider the committee $S' = (S^* \setminus \{c''\}) \cup \{c\}$. Again, we claim that S' has the same Borda score as S^*: we increase the misrepresentation of each of the $f(c'')$ voters who rank c'' first by 1 (as all of them rank c second), decrease the misrepresentation of each of the $f(c')$ voters who rank c' first by at least 1 (as all of them rank c second), and do not increase the misrepresentation of any other voter (as all of them prefer c to c''). Thus, the Borda score of S' does not exceed that of S^*, but $|S' \cap C^\circ| > |S^* \cap C^\circ|$, a contradiction with our choice of S^*. □

It is clear from our proof that Theorem 10 holds for every positional misrepresentation function whose score vector satisfies $s_1 = 0$, $s_2 = 1$, $s_3 \geqslant 2$. Observe also that our algorithm is in FPT with respect to the combined parameter (k, η); in contrast, for general preferences computing the Chamberlin–Courant winners is W[2]-hard with respect to k even under the Borda misrepresentation function (Betzler et al. 2013). Moreover, the algorithm of Yu et al. (2013) for trees with few leaves is in XP with respect to the number of leaves λ, but is not in FPT with respect to λ or even (k, λ).

7. CONCLUSIONS AND FUTURE WORK

We have designed polynomial-time algorithms for recognizing profiles that are single-peaked on special classes of trees, and demonstrated that such algorithms may be useful for efficient winner determination procedures under the Chamberlin–Courant rule. We believe that results similar to those of Section 6 can be obtained for other types of trees, and, more broadly, for other computational problems that are hard for general preferences, but easy for single-peaked preferences. Moreover, it seems plausible that such results can be extended to profiles that are "almost" single-peaked on a tree, for distance measures such as the ones proposed by Cornaz et al. (2012), Faliszewski et al. (2014) or Erdélyi et al. (2013). On the other hand, our analysis suggests new measures of closeness to single-peakedness (on the line) that are specialised to profiles that are single-peaked on a tree, namely, being single-peaked on a tree that is "almost" a line. Such measures may turn out to be easier to compute and exploit than the ones for arbitrary profiles, which tend to be computationally demanding (Erdélyi et al. 2013; Bredereck et al. 2013).

Conceptually, the notion of single-peakedness on a tree has been criticised for having less explanatory power than that of single-peakedness on a line. Indeed, profiles that are single-peaked on a star and ones that are single-peaked on a line have little in common, beyond the guaranteed existence of Condorcet winners, and arguably, a designation that lumps them together is of limited use. The tools developed in our work permit us to identify coherent subdomains of this broad domain, which, in turn, enables us to reason about structurally similar profiles (ones that are single-peaked on 'similar' trees) and their shared properties. We hope that this intuition will lead to new insights about real-life preference domains.

Acknowledgements

This work was supported by the European Research Council (ERC) under grant number 639945 (ACCORD). Dominik Peters is supported by EPSRC, and received additional support through COST Action IC1205 on Computational Social Choice.

References

K. Arrow. *Social Choice and Individual Values.* John Wiley and Sons, 1951.

J. Bartholdi, III and M. Trick. Stable matching with pref-

erences derived from a psychological model. *Operation Research Letters*, 5(4):165–169, 1986.

N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. *Journal of Artificial Intelligence Research*, 47(1):475–519, 2013.

D. Black. On the rationale of group decision-making. *The Journal of Political Economy*, pages 23–34, 1948.

H. L. Bodlaender. A tourist guide through treewidth. *Acta Cybernetica*, 11(1–2):1, 1994.

F. Brandt, V. Conitzer, and U. Endriss. Computational social choice. In G. Weiss, editor, *Multiagent Systems*, pages 213–283. MIT Press, 2013.

R. Bredereck, J. Chen, and G. Woeginger. Are there any nicely structured preference profiles nearby? In *IJCAI'13*, pages 62–68, Aug. 2013.

B. Chamberlin and P. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *American Political Science Review*, 77(3):718–733, 1983.

D. Cornaz, L. Galand, and O. Spanjaard. Bounded single-peaked width and proportional representation. In *ECAI'12*, pages 270–275, 2012.

G. Demange. Single-peaked orders on a tree. *Mathematical Social Sciences*, 3(4):389–396, 1982.

J.-P. Doignon and J.-C. Falmagne. A polynomial time algorithm for unidimensional unfolding representations. *Journal of Algorithms*, 16(2):218–233, 1994.

G. Erdélyi, M. Lackner, and A. Pfandler. The complexity of nearly single-peaked consistency. In *AAAI'13*, pages 283–289, July 2013.

B. Escoffier, J. Lang, and M. Öztürk. Single-peaked consistency and its complexity. In *ECAI'08*, volume 8, pages 366–370, 2008.

P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. The complexity of manipulative attacks in nearly single-peaked electorates. *AI Journal*, 207:69–99, 2014.

H. N. Gabow and R. E. Tarjan. Efficient algorithms for a family of matroid intersection problems. *Journal of Algorithms*, 5(1):80–131, 1984.

T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *IJCAI'11*, pages 280–286, 2011.

H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1991.

P. Scheffler. A linear algorithm for the pathwidth of trees. In *Topics in combinatorics and graph theory*, pages 613–620. Springer, 1990.

M. A. Trick. Recognizing single-peaked preferences on a tree. *Mathematical Social Sciences*, 17(3):329–334, 1989.

L. Yu, H. Chan, and E. Elkind. Multiwinner elections under preferences that are single-peaked on a tree. In *IJCAI'13*, pages 425–431, 2013.

An Empirical Comparison of One-Sided Matching Mechanisms

Hadi Hosseini, Kate Larson, and Robin Cohen
Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
{h5hossei,klarson,rcohen}@uwaterloo.ca

ABSTRACT

For one-sided matching problems, two widely studied mechanisms are the Random Serial Dictatorship (RSD) and the Probabilistic Serial Rule (PS). Both mechanisms require only that agents specify ordinal preferences and have a number of desirable economic and computational properties. However, the induced outcomes of the mechanisms are often incomparable and thus there are challenges when it comes to deciding which mechanism to adopt in practice. In this paper, working in the space of general ordinal preferences, we provide empirical results on the (in)comparability of RSD and PS and analyze their respective economic properties. We then instantiate utility functions for agents, consistent with the ordinal preferences, with the goal of gaining insights on the manipulability, efficiency, and envyfreeness of the mechanisms under different risk attitude models.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Economics, Theory, Experimentation

Keywords

Mechanism Design, Matching, Random Assignment, Probabilistic Serial, Random Serial Dictatorship

1. INTRODUCTION

The problem of assigning a number of indivisible objects to a set of agents, in the absence of monetary transfers, is fundamental in many multiagent resource allocation applications, and has been the center of attention amongst researchers at the interface of artificial intelligence, economics, and mechanism design. Assigning dormitory rooms or offices to students, students to public schools, college courses to students, organs and medical resources to patients, members to subcommittees, etc. are some of the myriad examples of one-sided matching problems [31, 5, 13, 25].

Appears at: 3rd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 15th International Conference on Autonomous Agents and Multiagent Systems. May 9th-10th, 2016. Singapore.

Two important (randomized) matching mechanisms that only elicit ordinal preferences from agents are *Random Serial Dictatorship* (RSD) [2] and *Probabilistic Serial Rule* (PS) [11]. Both mechanisms have important economic properties and are practical to implement. The RSD mechanism has strong truthful incentives but guarantees neither efficiency nor envyfreeness. PS satisfies efficiency and envyfreeness; however, it is susceptible to manipulation. Therefore, there are subtle points to be considered when deciding which mechanism to use. For example, given a particular preference profile, the mechanisms often produce random assignments which are simply incomparable and thus, without additional knowledge of the underlying utility models of the agents, it is difficult to determine which is the "better" outcome. Furthermore, properties like efficiency, truthfulness, and envyfreeness can depend on whether there is underlying structure in the preferences, and even in general preference models it is valuable to understand under what conditions a mechanism is likely to be efficient, truthful, or envyfree as this can guide designers choices.

We study the comparability of PS and RSD when there is only one copy of each object, and analyze the space of all preference profiles for different combinations of agents and objects. We show that despite the inefficiency of RSD, the fraction of random assignments at which PS stochastically dominates RSD vanishes, especially when the number of agents is less than or equal to the available objects. We then instantiate utility functions for agents to gain insights on the manipulability, social welfare, and envyfreeness of the two mechanisms under different risk attitudes.

Our main result is that under risk aversion, the social welfare of RSD is as good as PS but RSD does create envy among the agents (though the fraction of envious profiles and total envy are small). Moreover, when the number of agents and objects are equal, RSD assignments are less likely to be dominated by PS and overall RSD assignments create negligible envy among agents. We also show that PS is highly susceptible to manipulation in almost all combinations of agents and objects. The fraction of manipulable profiles and the gain from manipulation rapidly increases, particularly when agents become more risk averse.

2. MODEL

In this section, we describe the basic one-sided matching problem and introduce the two mechanisms we study in detail, Random Serial Dictatorship (RSD) [2] and Probabilistic Serial Rule (PS) [11]. We then introduce a number of properties and criteria used to evaluate these mechanisms.

2.1 One Sided Matching

A one-sided matching problem consists of a set of n agents, N, and a set of m indivisible objects, M.[1] Each agent $i \in N$ has a private strict preference ordering, $\succ_i$, over M where $a \succ_i b$ indicates that agent i prefers to receive object a over object b. We represent the preference ordering of agent i by the ordered list of objects $\succ_i = a \succ_i b \succ_i c$ or $\succ_i = (abc)$, for short. We let $\mathcal{P}$ denote the set of all complete and strict preference orderings over M. A *preference profile* $\succ \in \mathcal{P}^n$ specifies a preference ordering for each agent, and we use the standard notation $\succ_{-i} = (\succ_1, \ldots, \succ_{i-1}, \succ_{i+1}, \ldots, \succ_n)$ to denote preferences orderings of all agents except i and thus $\succ = (\succ_i, \succ_{-i})$.

The goal in a one-sided matching problem is to assign the objects in M to the agents in N according to preference profiles, under the constraint that no object can be assigned to more than one agent. If $m = n$ then this means that each agent will receive exactly one object, however if $m < n$ then some agents will receive no object and if $m > n$ then some agents may receive multiple objects. An assignment is represented as a matrix

$$A = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_n \end{pmatrix} = \begin{pmatrix} A_{1,1} & A_{1,2} & \ldots & A_{1,m} \\ A_{2,1} & A_{2,2} & \ldots & A_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \ldots & A_{n,m} \end{pmatrix}$$

where $A_{i,j} \in [0,1]$ is the probability that agent i is assigned object j. We let $\mathcal{A}$ denote the set of all *feasible* assignments where an assignment $A \in \mathcal{A}$ is *feasible* if and only if $\forall j \in M$, $\sum_{i \in N} A_{i,j} = 1$. If $A \in \mathcal{A}$ is such that $A_{i,j} \in \{0,1\}$ then we say that A is a *deterministic* assignment; otherwise, A is a *random* assignment. Every random assignment can be represented as a convex combination of deterministic assignments [37], and thus we view random assignments as a probability distribution over a set of deterministic assignments.

2.2 Matching Mechanisms

In general, a *matching mechanism*, $\mathcal{M}$, is a mapping from the set of preference profiles, $\mathcal{P}^n$ to the set of feasible assignments, $\mathcal{A}$. That is, $\mathcal{M} : \mathcal{P}^n \mapsto \mathcal{A}$. In this paper, we focus our attention on two widely studied mechanisms for one-side matching: Random Serial Dictatorship (RSD) [2] and Probabilistic Serial Rule (PS) [10].

RSD relies on the concept of priority orderings over agents. Such an ordering is an ordered list of agents where the first agent gets to select its most preferred object from the set of objects, the second agent then selects its most preferred object from the set of remaining objects and so on, until no objects remain.[2] Given a preference profile $\succ \in \mathcal{P}^n$, RSD returns an assignment $RSD(\succ) \in \mathcal{A}$ which is a uniform distribution over all deterministic assignments induced from all possible priority orderings over the set of agents. RSD has been widely adopted for fair and strategyproof assignment for the school choice problem, course assignment, house allocation, and room assignment [1, 35, 2, 3]

PS treats objects as a set of divisible goods of equal size and simulates a simultaneous eating algorithm. Each agent starts "eating" its most preferred object, all at the same rate. Once an object is gone (eaten away) then the agent starts eating its next preferred object among the remaining objects. This process terminates when all objects have been "eaten". Given a preference profile $\succ \in \mathcal{P}^n$, $PS(\succ) \in \mathcal{A}$ is a random assignment where $A_{i,j}$ is the probability (fraction) that object j is assigned to (or "eaten by") agent i.

2.3 General Properties

In this section, we define key properties for matching mechanisms. In particular, we formally define efficiency, strategyproofness and envyfreeness for (randomized) matching mechanisms under ordinal preferences. To evaluate the quality of a random assignment, we use first-order stochastic dominance [18, 11]. Given a random assignment A_i, the probability that agent i is assigned an object that is at least as good as object ℓ is defined as follows

$$w(\succ_i, \ell, A_i) = \sum_{j \in M : j \succeq_i \ell} A_{i,j} \tag{1}$$

We say an agent always prefers assignment A_i to B_i, if for each object ℓ the probability of assigning an object at least as good as ℓ under A_i is greater or equal that of B_i, and strictly greater for some object.

DEFINITION 1 (STOCHASTIC DOMINANCE). *Given a preference ordering $\succ_i$, random assignment A_i stochastically dominates (sd) assignment $B_i (\neq A_i)$ if*

$$\forall \ell \in M,\ w(\succ_i, \ell, A_i) \geq w(\succ_i, \ell, B_i) \tag{2}$$

A matching mechanism is *sd*-efficient if at all preference profiles $\succ \in \mathcal{P}^n$, for all agents $i \in N$, the induced assignment is not stochastically dominated by any other assignment.

DEFINITION 2 (*sd*-EFFICIENCY). *A random assignment is sd-efficient if for all agents, it is not stochastically dominated by any other random assignment.*

An important desirable property in matching mechanisms is strategyproofness, that is the mechanism is designed so that no agent has incentive to misreport its preferences.

DEFINITION 3 (*sd*-STRATEGYPROOFNESS). *Mechanism $\mathcal{M}$ is sd-strategyproof if at all preference profiles $\succ \in \mathcal{P}^n$, for all agents $i \in N$, and for any misreport $\succ'_i \in \mathcal{P}^n$, such that $A = \mathcal{M}(\succ)$ and $A' = \mathcal{M}(\succ'_i, \succ_{-i})$, we have:*

$$\forall \ell \in M,\ w(\succ_i, \ell, A_i) \geq w(\succ_i, \ell, A'_i) \tag{3}$$

Sd-strategyproofness is a strict requirement. It implies that under any utility model consistent with the preference orderings, no agent can improve her expected utility by misreporting. We say that a mechanism is *weakly sd-strategyproof* if the inequality in Equation 3 is strict for some $\ell \in M$, but does not hold for all objects. Clearly, *sd*-strategyproofness implies weak *sd*-strategyproofness but the converse does not hold.

An assignment is *manipulable* if it is not *sd*-strategyproof. If there exists some agent who strictly benefits from the

[1] This problem is sometimes called the assignment problem or house allocation problem in the literature.

[2] For $n < m$, RSD requires a careful method for picking sequence at each realized priority ordering based on an arbitrary serial dictatorship quota mechanism, which directly affects the efficiency and envy of the assignments [19, 12]. For simplicity, we use the variant of RSD based on a *quasi-dictatorial* mechanism [29] where the first agent selects its most preferred $(m-n+1)$ objects, and the rest of the agents choose one object each.

	$n \geq m$		$n < m$	
	PS	RSD	PS	RSD
sd-strategyproof	weak	✓	✗	✓
sd-efficiency	✓	✗	✓	✗
sd-envyfree	✓	weak	✓	weak

Table 1: Properties of PS and RSD.

manipulation, (i.e. the mechanism is not even weakly *sd*-strategyproof) then we say the assignment is *sd*-manipulable.

Finally, we are interested in whether mechanisms are fair and use the notion of envyfreeness to this end. An assignment is *sd*-envyfree if each agent strictly prefers her random allocation to any other agent's assignment.

Definition 4 (*sd*-Envyfreeness). *Given agent i's preference $\succ_i$, assignment A_i is sd-envyfree if for all agents $\forall k \neq i \in N$,*

$$\forall \ell \in M,\ w(\succ_i, \ell, A_i) \geq w(\succ_i, \ell, A_k) \tag{4}$$

We say an assignment is weakly *sd*-envyfree if the inequality in Equation 4 is strict for some $\ell \in M$, but does not hold for all objects. A matching mechanism satisfies *sd*-envyfreeness if at all preference profiles $\succ \in \mathcal{P}^n$, it induces *sd*-envyfree assignments for all agents.

2.4 Properties of RSD and PS

The theoretical properties of PS and RSD have been well studied in the economics literature [11], and we summarize the results in Table 1. Both mechanisms are ex post efficient, that is, their realized outcomes cannot be improved without making at least one agent worse off. PS has been shown to be both *sd*-envyfree and *sd*-efficient. However, it is not even weakly *sd*-strategyproof when $n < m$ [22] and is only weakly *sd*-strategyproof when $n \geq m$. On the other hand, RSD is always *sd*-strategyproof, but it is only weakly *sd*-envyfree and is not *sd-efficient*. Example 1 illustrates the inefficiency of RSD.

Example 1. *Suppose there are four agents $N = \{1, 2, 3, 4\}$ and four objects $M = \{a, b, c, d\}$. Consider the following preference profile $\succ = ((abcd), (abcd), (badc), (badc))$. Table 2 shows the outcomes for $PS(\succ)$ and $RSD(\succ)$. In this example, all agents strictly prefer the assignment induced by PS over the RSD assignment. Thus, RSD is inefficient at this preference profile.*

	a	b	c	d
A_1	1/2	0	1/2	0
A_2	1/2	0	1/2	0
A_3	0	1/2	0	1/2
A_4	0	1/2	0	1/2

(a) Assignment under $PS(\succ)$

	a	b	c	d
A_1	5/12	1/12	5/12	1/12
A_2	5/12	1/12	5/12	1/12
A_3	1/12	5/12	1/12	5/12
A_4	1/12	5/12	1/12	5/12

(b) Assignment under $RSD(\succ)$

Table 2: Example showing the inefficiency of RSD

3. INCOMPARABILITY OF RSD AND PS

We argue that the theoretical findings on RSD and PS do not necessarily provide enough guidance to a market designer trying to select the correct mechanism for a specific setting. For example, while we know that PS is *sd*-efficient and RSD is not, this does not mean that PS always outperforms RSD.

	a	b	c
A_1	1/2	0	1/2
A_2	1/2	1/4	1/4
A_3	0	3/4	1/4

(a) Assignment under $PS(\succ)$

	a	b	c
A_1	1/2	0	1/2
A_2	1/2	1/6	1/3
A_3	0	5/6	1/6

(b) Assignment under $RSD(\succ)$

Table 3: Incomparability of RSD and PS

Example 2. *Suppose there are three agents $N = \{1, 2, 3\}$ and three objects $M = \{a, b, c\}$. Consider the following preference profile $\succ = ((acb), (abc), (bac))$. Table 3 shows $PS(\succ)$ and $RSD(\succ)$. Neither assignment dominates the other since agent 1 is ambivalent between the two assignments while agent 2 prefers $PS(\succ)$ and agent 3 prefers $RSD(\succ)$.*

If we knew the utility functions of the agents, consistent with their ordinal preferences, then we might be able to use the notion of (utilitarian) social welfare to help determine the better assignment.[3] However, it is easy to construct different utility functions for the agents in Example 2 where both *RSD* and *PS* maximize social welfare.

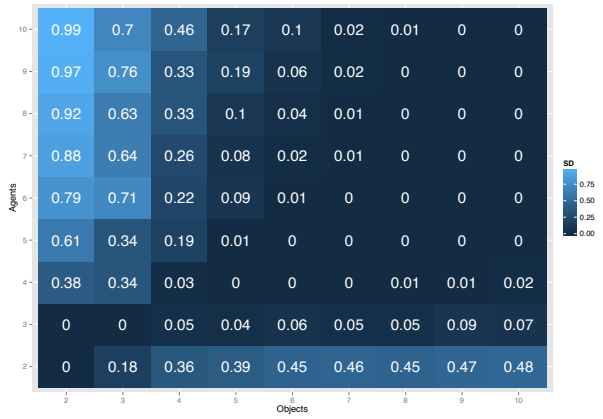
Similarly, the envy of RSD and the manipulability of PS both depend on the structure of preference profiles, and thus, a compelling question, that justifies the practical implications of deploying a matching mechanism, is to analyze the space of preference profiles to find the likelihood of inefficient, manipulable, or envious assignments under these mechanisms.

4. GENERAL PREFERENCES

The theoretical properties of PS and RSD only provide limited insight into their practical applications. In particular, when deciding which mechanism to use in different settings, the incomparability of PS and RSD leaves us with an ambiguous choice in terms of efficiency, manipulability, and envyfreeness. Thus, we examine the properties of RSD and PS in the space of all possible preference profiles as well as under lexicographic preferences. Lexicographic preferences are present in various applications and have been extensively studied in artificial intelligence and multiagent systems as a means of assessing allocations based on ordinal preferences [15, 32, 17]. Under lexicographic preferences, we denote the efficiency, strategyproofness, manipulablity, and envyfreeness with *ld-* (lexicographically dominate) prefix.

The number of all possible preference profiles is super exponential $(m!)^n$. For each combination of n agents and m objects we performed a brute force coverage of all possible preference profiles. Thus, for all subsequent figures each data point shows the fraction of all possible preference profiles. For the cases of $n = 10$ and $m \in \{9, 10\}$, we randomly generated 1,000 instances by sampling from a uniform preference profile distribution. For each preference profile, we ran both PS and RSD mechanisms and compared their outcomes in terms of the stochastic dominance relation. Note that not only is computing RSD probabilities #P-complete (and thus intractable) [6, 33], but checking the desire properties such as envyfreeness, efficiency, and manipulablity of random allocations is shown to be NP-hard for general settings [9, 8]. Thus, for larger settings even if we randomly

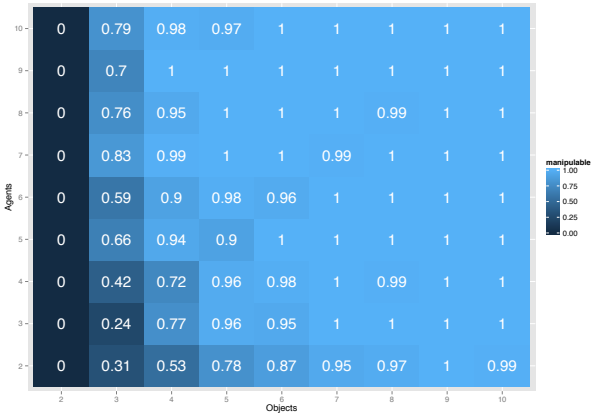
[3]Given utility functions for the agents, where $u_i(j)$ is the utility agent i derives from being assigned object j, the (utilitarian) social welfare of an assignment A is $\sum_i \sum_j A_{i,j} u_i(j)$.



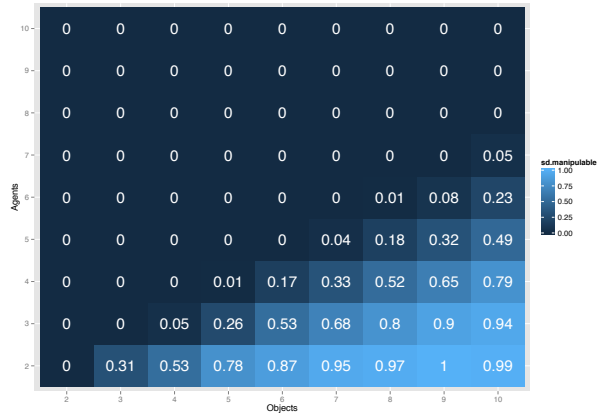
(a) The fraction that PS stochastically dominates RSD.

(b) The fraction that PS lexicographically dominates RSD.

Figure 1: The fraction of preference profiles under which PS dominates RSD.



(a) The fraction of manipulable preference profiles under PS.



(b) The fraction of *sd*-manipulable profiles under PS.

Figure 2: Heatmaps illustrating the manipulablity of PS.

sample preference profiles it is not easy to verify the aforementioned properties.

A preliminary look at our empirical results illustrates the following: when $m \leq 2, n \leq 3$, PS coincides exactly with RSD, which results in the best of the two mechanisms, i.e., both mechanisms are *sd*-efficient, *sd*-strategyproof, and *sd*-envyfree. Another interesting observation is that when $m = 2$, for all n PS is *sd*-strategyproof (although the PS assignments are not necessarily equivalent to assignments induced by RSD), RSD is *sd*-envyfree, and for most instances PS stochastically dominates RSD, particularly when $n \geq 4$.

4.1 Efficiency

Our first finding [4] is that the fraction of preference profiles at which RSD and PS induce equivalent random assignments goes to 0 when n grows. There are two conclusions that one can draw. First, this result confirms the theoretical results of Manea on asymptotic inefficiency of RSD [24], in that, in most instances, the assignments induced by RSD are not equivalent to the PS assignments. Second, this result suggests that the incomparability of outcomes is significant, that is, the social welfare of the random outcomes is highly dependent on the underlying utility models.

The fraction of preference profiles $\succ \in \mathcal{P}^n$ for which RSD is stochastically dominated by PS at $\succ$ converges to zero as $\frac{n}{m} \to 1$. Figure 1a shows that when n grows beyond $n > 5$, due to incomparability of RSD and PS with regard to the stochastic dominance relation, the RSD assignments are not stochastically dominated by *sd*-efficient assignments induced by PS.

We also see similar results when we restrict ourselves to lexicographic preferences (Figure 1b). The fraction of preference profiles $\succ \in \mathcal{P}^n$ for which RSD is lexicographically dominated by PS at $\succ$ converges to zero as $\frac{n}{m} \to 1$.

For lexicographic preferences, we also observe that the fraction of preference profiles for which PS assignments strictly dominate RSD-induced allocations goes to 1 when the number of agents and objects diverge. The fraction of preference profiles $\succ \in \mathcal{P}^n$ for which RSD is lexicographically dominated by PS at $\succ$ converges to 1 as $|n - m|$ grows.

[4] Periodically, we present results without a figure.

One immediate conclusion is that although RSD does not guarantee either *sd*-efficiency or *ld*-efficiency, in most settings when $\frac{n}{m} \to 1$ (and also $n \leq m$ for *sd*-efficiency), neither of the two mechanisms is preferred in terms of efficiency. Hence, one cannot simply rule out the RSD mechanism.

4.2 Manipulability of PS

One critical issue with deploying PS is that it does not provide incentives for honest reporting of preferences. Although for $n \geq m$ PS is weakly *sd*-strategyproof [11] and *ld*-strategyproof [34], when $n < m$ PS no longer satisfies these two properties.[5] The real concern is that, in the absence of strategyproofness, PS allocations are only efficient (or envyfree) with respect to the reported preferences, which in turn may not be truthful. Thus, we are interested in understanding the degree to which PS allocations are manipulable.

Figure 2 shows that the fraction of manipulable profiles goes to 1 as n or m grow. PS is almost 99% manipulable for $n > 5, m > 5$. Another interesting observation is that, for all $n < m$, the fraction of *sd*-manipulable preference profiles goes to 1 as $m - n$ grows (Figure 2b). These results imply that when agents are entitled to receive more than a single object, agents can strictly benefit from misreporting their preferences. The manipulability of PS under lexicographic preferences has a similar trend and the fraction of *ld*-manipulable preference profiles converges to 1 even more rapidly.

4.3 Envy in RSD

We measured the fraction of agents that are weakly *sd*-envious of at least one another agent when running RSD. Our results show that the percentage of agents that are weakly envious increases with the number of agents. Moreover, fixing any $n > 3$, the percentage of agents that are (weakly) envious grows with the number of objects; however, there is a sudden drop in the percentage of envious agents when there are equal number of agents and objects.

5. UTILITY MODELS

Given a utility model consistent with an agent's preference ordering, we can find the agent's expected utility for a random assignment. Let u_i denote agent i's Von Neumann-Morgenstern (VNM) utility model consistent with its preference ordering $\succ_i$. That is, $u_i(a) > u_i(b)$ if and only if $a \succ_i b$. Then, agent i's expected utility for random assignment A_i is $\mathbb{E}(u_i|A_i) = \sum_{j \in M} A_{i,j} u_i(j)$.

We say that agent i (strictly) prefers assignment A_i to B_i if and only if $\mathbb{E}(u_i|A_i) > \mathbb{E}(u_i|B_i)$. A mechanism is strategyproof if there exists no agent that can improve its expected utility by misreporting its preference ordering.

Definition 5 (Strategyproof). *Mechanism $\mathcal{M}$ is strategyproof if for all agents $i \in N$, and for any misreport $\succ_i' \in \mathcal{P}^n$, such that $A = \mathcal{M}(\succ)$ and $A' = \mathcal{M}(\succ_i', \succ_{-i})$, given a utility model u_i consistent with $\succ_i$, we have $\mathbb{E}(u_i|A_i) \geq \mathbb{E}(u_i|A_i')$.*

[5]A recent experimental study on the incentive properties of PS shows that human subjects are less likely to manipulate the mechanism when misreporting is a Nash equilibrium. However, subjects' tendency for misreporting is still significant even when it does not improve their allocations [20].

A matching mechanism is envyfree if for all preference profiles it prescribes an envyfree assignment.

Definition 6 (Envyfreeness). *Assignment A is envyfree if for all $i, k \in N$, given utility model u_i consistent with $\succ_i$, we have $\mathbb{E}(u_i|A_i) \geq \mathbb{E}(u_i|A_k)$.*

A random assignment A is *sd*-efficient if and only if there exists a profile of utility values consistent with $\succ$ such that A maximizes the social welfare ex ante [11, 26]. This existence result does not shed light on the social welfare when comparing two random assignments, since an assignment can be *sd*-efficient but may not have desirable ex ante social welfare. Given utility functions for the agents, the (utilitarian) *social welfare* of an assignment A is $\sum_i \mathbb{E}(u_i|A_i)$. Thus, given a profile of utilities we investigate the (ex ante) social welfare of the assignments under PS and RSD.

5.1 Instantiating Utility Functions

To deepen our understanding as to the performance of the two mechanisms, we investigate different utility models. In particular we look at the performance of the mechanisms when the agents are all risk neutral (i.e. have linear utility functions), when agents are risk seeking and when agents are risk averse.

Our first utility model is the well-studied linear utility model, and we use a variant based on the Borda rule from the social choice literature. Given an agent i's preference ordering $\succ_i$, we let $r(\succ_i, j)$ denote the rank of object j. For example, given preference ordering $a \succ_i b \succ_i c$ then $r(\succ_i, a) = 1$, $r(\succ_i, b) = 2$ and $r(\succ_i, c) = 3$. The utility function for agent i, given object j is $u_i(j) = m - r(\succ_i, j)$.

We use an *exponential* utility model to capture risk attitudes beyond risk-neutrality. An exponential utility has been shown to provide an appropriate translation for individuals' utility models and provides a constant risk aversion rate [4]. In particular,

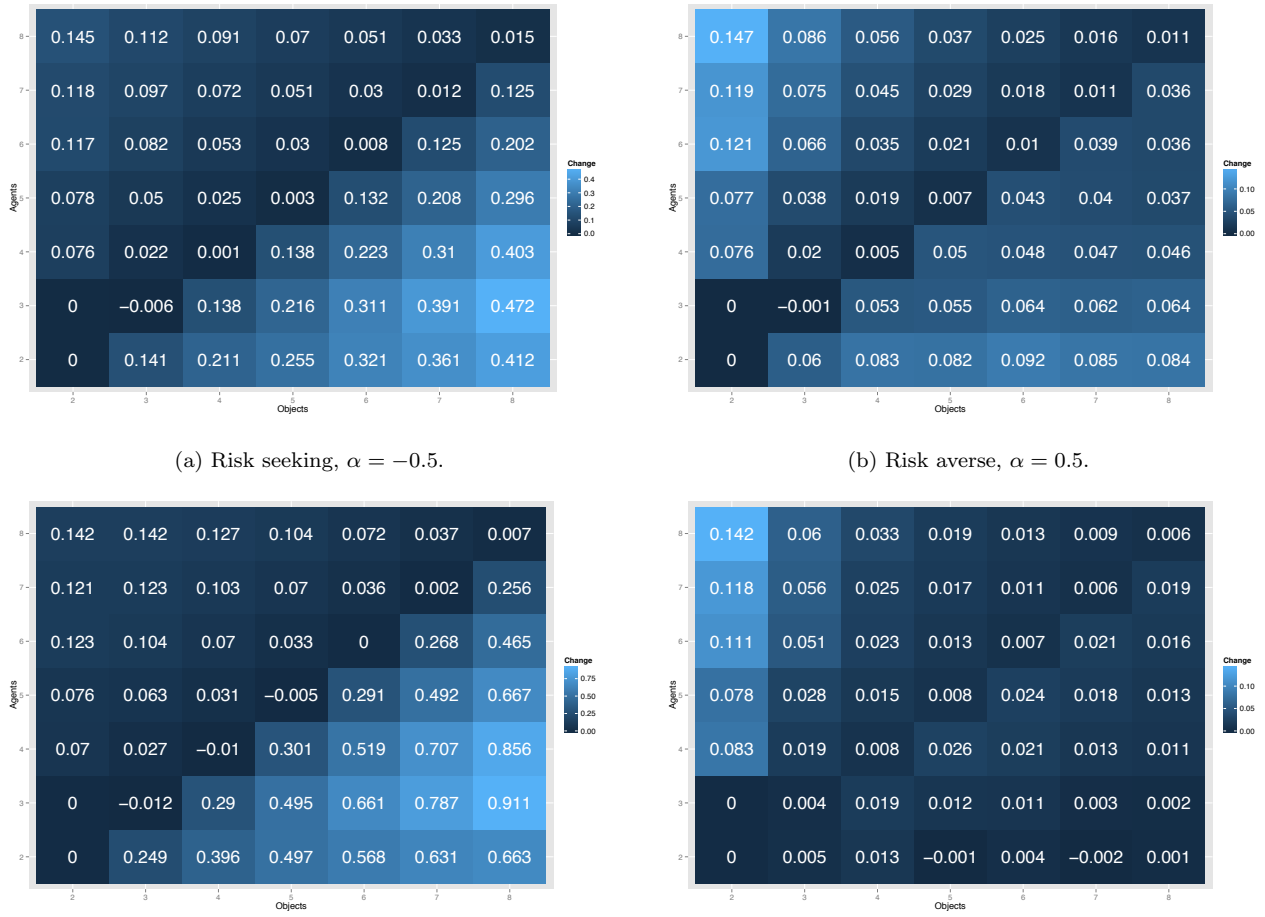
$$u_i(j) = \begin{cases} (1 - e^{-\alpha(m - r(\succ_i, j))})/\alpha, & \alpha \neq 0 \\ m - r(\succ_i, j), & \alpha = 0 \end{cases} \quad (5)$$

The parameter α represents the agent's risk attitude. If $\alpha > 0$ then the agent is risk averse, while if $\alpha < 0$ then the agent is risk seeking. When $\alpha = 0$ then the agent is risk neutral and we have a linear utility model. The value $|\alpha|$ represents the intensity of the attitude. That is, given two agents with $\alpha_1 < \alpha_2 < 0$, we say that agent 1 is more risk averse than agent 2. Similarly if $\alpha_1 > \alpha_2 > 0$ then agent 1 is more risk seeking than agent 2.

6. RESULTS

For our experiments, we vary three parameters: the number of agents n, the number of objects m, and the risk attitude factor α. Each data point in the graphs shows the average over all possible preference profiles. We study the same settings as in Section 4 when $n \geq m$ and $n < m$. For each utility function, we look at homogeneous populations of agents where agents have the same risk attitudes.

To compare the social welfare, we investigate the percentage change (or improvement) in social welfare of PS compared to RSD under various utility models. That is, $\frac{\sum_i \mathbb{E}(u_i|PS(\succ)) - \sum_i \mathbb{E}(u_i|RSD(\succ))}{\sum_i \mathbb{E}(u_i|RSD(\succ))}$. To measure the manipulability of PS, we are interested in answering two key questions: i) In what fraction of profiles PS is manipulable by



(a) Risk seeking, $\alpha = -0.5$.

(b) Risk averse, $\alpha = 0.5$.

(c) Risk seeking, $\alpha = -2$.

(d) Risk averse, $\alpha = 2$.

Figure 3: The percentage change in social welfare. The negative values show that RSD outperforms PS.

at least one agent? and *ii*) If manipulation is possible, what is the average percentage of maximum gain, that is $\max_i\{\frac{\mathbb{E}(u_i|PS(\succ_i',\succ_{-i}))-\mathbb{E}(u_i|PS(\succ))}{\mathbb{E}(u_i|PS(\succ))}\}$? To study the envy under the RSD mechanism, we consider two measures: *i*) the fraction of envious agents, and *ii*) the total envy felt by all agents.

6.1 Linear Utility Model

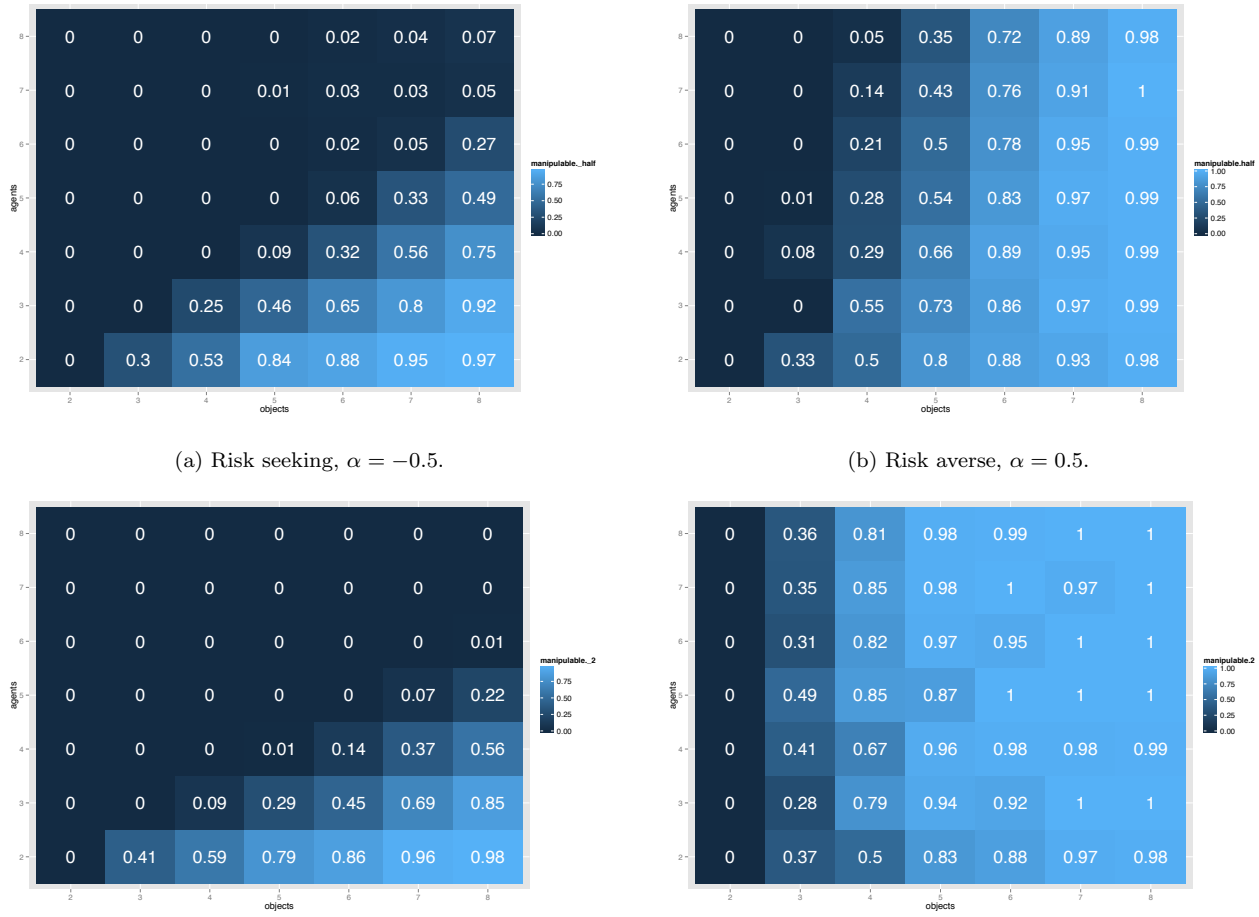
We first looked at how RSD and PS perform under the assumption that the utility models are linear. In most cases, the social welfare under PS increases compared to RSD; however, the percentage change from PS to RSD becomes smaller when $n = m$ (less than 0.015 overall improvement in all cases). Interestingly, under RSD the fraction of envious agents is approximately 0 when $n \geq m$. With regards to strategyproofness, PS is manipulable in most combinations of n and m and the fraction of manipulable profiles and the utility gain from manipulation increases as the number of objects compared to agents increases.

6.2 Risk Seeking

Figure 3 presents our results in terms of percentage change in social welfare. Positive numbers show the percentage of improvement in social welfare. Negative values represent those cases where RSD has increased social welfare compared to PS.

Social welfare: Fixing $\alpha < 0$, for $n \geq m$ when $\frac{n}{m}$ grows PS improves the social welfare compared to RSD in most cases and the percentage of improvement also increases. A similar trend holds when varying risk intensity α for fixed n and m where $n \neq m$. For $n < m$, when $\frac{m}{n}$ grows the fraction of profiles at which PS has higher social welfare compared to RSD rapidly increases and the percentage change is also noticeably larger, quickly getting close to 90% improvement (Fig. 3a and 3c). This social welfare gap between PS and RSD grows as the risk intensity $|\alpha|$ increases. Surprisingly, this trend changes for equal number of agents and objects $n = m$: the more risk-seeking agents are (larger $|\alpha|$), RSD becomes more desirable than PS, and in fact, RSD improves the social welfare in more instances.

Envy: For $n \geq m$, the fraction of envious agents under all profiles vanishes and RSD becomes envyfree. This is more evident when agents are more risk-seeking. Intuitively, these observations confirm the theoretical findings about the envyfreeness of RSD under lexicographic preferences [19] since one can consider lexicographic preferences as risk-seeking preferences where an object in a higher ranking is infinitely preferred to all objects that are ranked less



(a) Risk seeking, $\alpha = -0.5$.

(b) Risk averse, $\alpha = 0.5$.

(c) Risk seeking, $\alpha = -2$.

(d) Risk averse, $\alpha = 2$.

Figure 4: The fraction of manipulable instances under PS.

preferably. When $n < m$, our quasi-dictatorial extension of RSD creates some envy among the agents, but this envy also starts to fade out when the risk intensity $|\alpha|$ increases.

Manipulability: Figure 4 shows the manipulability of the PS assignments when agents are risk seeking. We see that the possibility of manipulation (and any gain) decreases as the risk intensity increases. When $n \geq m$ the fraction of manipulable profiles goes to 0 the more risk seeking agents become. However, when $n < m$ even though the the fraction of manipulable profiles (and manipulation gain) decreases, the fraction of manipulable profiles goes to 1 as $\frac{m}{n}$ grows.

6.3 Risk Aversion

Social welfare: Figures 3b and 3d show that fixing risk factor $\alpha > 0$, when $\frac{n}{m}$ grows PS assignments are superior to that of RSD in more instances and the percentage change in social welfare increases. Fixing risk factor $\alpha > 0$ and when $\frac{m}{n}$ grows, RSD is more likely to have the same social welfare as PS, and in fact in some instances the social welfare under RSD is better than the social welfare under PS. Fixing m and n, when the risk intensity α increases RSD is more likely to have the same social welfare as PS, that is, the welfare gap between PS and RSD closes when agents are more risk averse (α increases). This result is insightful and states that under risk aversion the random allocations prescribed by RSD are either as good as PS or in some cases even are superior to the allocations prescribed by PS. Figure 5 illustrates the percentage change in social welfare based on the difference between available objects and agents ($m-n$) for risk seeking, linear, and risk averse utilities with different risk intensities.

Envy: When $n \geq m$, the fraction of envious agents and total envy grows as $\frac{n}{m} \to 1$. Increasing the risk intensity ($|\alpha|$), the fraction of envious agents increases; however, the total envy among the agents remains considerably low. For $n < m$, the fraction of envious agents and total envy grows as risk intensity increases. Lastly, we noticed that in all instance where RSD creates envy among the agents, around 25% of agents bear more than 50% of envy. That is, few agents feel extremely envious while all other agents are either envyfree or only feel a minimal amount of envy.

Manipulability: Figures 4b and 4d illustrate the manipulability of the PS assignments when agents have risk averse preferences. The fraction of manipulable profiles rapidly goes to 1 as $\frac{m}{n}$ grows. Similarly, as agents become more risk averse (α increases) the fraction of manipulable profiles goes to 1 and the manipulation gain increases.

7. RELATED LITERATURE

Assignment problems with ordinal preferences have attracted interest from many researchers. Svensson showed that serial dictatorship is the only deterministic mechanism that is strategyproof, nonbossy, and neutral [36]. Random Serial Dictatorship (RSD) (uniform randomization over all serial dictatorship assignments) satisfies strategyproofness, proportionality, and ex post efficiency [2]. Bogomolnaia and Moulin noted the inefficiency of RSD from the ex ante perspective, and characterized the matching mechanisms based on first-order stochastic dominance [11]. They proposed the probabilistic serial mechanism as an efficient and envyfree mechanism with regards to ordinal preferences. While PS is not strategyproof, it satisfies weak strategyproofness for problems with equal number of agents and objects. However, PS is strictly manipulable (not weakly strategyproof) when there are more objects than agents [21]. Kojima and Manea, showed that in large assignment problems with sufficiently many copies of each object, truth-telling is a weakly dominant strategy in PS [22]. In fact PS and RSD mechanisms become equivalent [14], that is, the inefficiency of RSD and manipulability of PS vanishes when the number of copies of each object approaches infinity.

The practical implications of deploying RSD and PS have been the center of attention in many one-sided matching problems [1, 27]. In the school choice setting with multi-capacity alternatives, Pathak observed that many students obtained a more desirable random assignment through PS in public schools of New York City [30]; however, the efficiency difference was quite small. These equivalence results and their extensions to all random mechanisms [23], do not hold when the quantities of each object is limited to one.

Other interesting aspects of PS and RSD such as computational complexity and best-responses strategies have also been explored [16, 8, 7]. In this vein, Aziz et al. proved the existence of pure Nash equilibria, but showed that computing an equilibrium is NP-hard [7]. Nevertheless, Mennle et al. [28] showed that agents can easily find near-optimal strategies by simple local and greedy search. In the absence of truthful incentives, the outcome of PS is no longer guaranteed to be efficient or envyfree with respect to agents' true underlying preferences, and this inefficiency may result in outcomes that are worse than RSD, especially in 'small' markets [16].

8. DISCUSSION

We studied the space of general preferences and provided empirical results on the (in)comparability of RSD and PS. It is worth mentioning that at preference profiles where PS and RSD induce identical assignments, RSD is *sd*-efficient, *sd*-envyfree, and *sd*-strategyproof. However, PS is still highly manipulable. We investigated various utility models according to different risk attitudes. Our main results are:

- In terms of efficiency, the fraction of preference profiles $\succ \in \mathcal{P}^n$ for which PS stochastically (or lexicographically) dominates RSD converges to zero as $\frac{n}{m} \to 1$. When instantiating the preferences with actual utility functions, PS allocations are only slightly better than RSD allocations in terms of social welfare when varying n and m, particularly under risk averse utilities. In fact, in some cases RSD allocations are superior in terms of social welfare (see Figure 5).

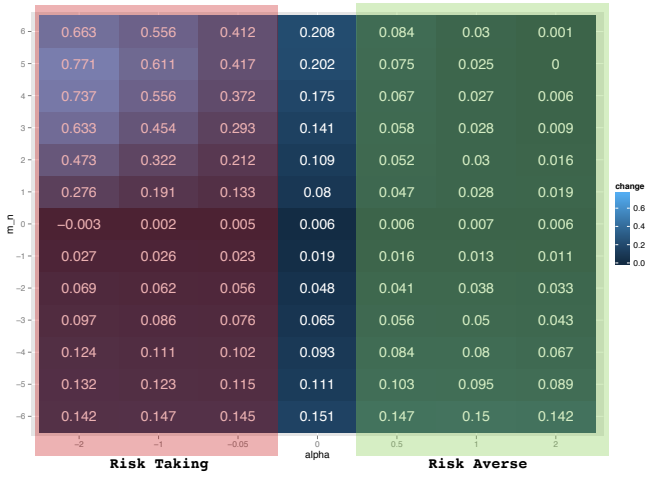


Figure 5: The percentage change in social welfare between RSD and PS for $\alpha \in (-2, -1, -0.5, 0, 0.5, 1, 2)$ and different combinations of $m - n$. Positive α indicates risk averse and negative α risk taking profiles. Linear utility is indicated by $\alpha = 0$. As agents become more risk averse the social welfare gap between RSD and PS closes.

- PS is almost 99% manipulable when $n \leq m$ and the fraction of *sd*- and *ld*- manipulable profiles rapidly goes to 1 as $\frac{m}{n}$ grows. When instantiating the preferences with utility functions, the manipulability of PS increases as agents become more risk averse. Moreover, an agent's utility gain from manipulation also grows when the risk intensity increases.

- For risk seeking utilities, when $n \geq m$ the fraction of envious agents under all profiles vanishes and RSD becomes envyfree. For risk averse utilities, the fraction of envious agents increases as agents become more risk averse. However, the total amount of envy just slightly grows, and surprisingly, only few agents feel extremely envious while all other agents are either envyfree or only feel a minimal amount of envy.

Our work in this paper can be used to help guide designers of multiagent systems who need to solve allocation problems. If a designer strongly requires *sd*-efficiency then the theoretical results of PS indicate that it is better than RSD. However, our results show that PS is highly prone to manipulation for various combinations of agents and objects. This manipulation and the possible gain from manipulation become more severe particularly when agents are risk averse, and designers need to take this into consideration. On the other hand, while RSD does not theoretically guarantee *sd*-efficiency, our results show that it tends to do quite well – sometimes even outperforming PS in terms of social welfare. RSD also has the added advantage of being *sd*-strategyproof and thus is not prone to the manipulation problems of PS.

An interesting future direction is to study egalitarian social welfare of the matching mechanisms in single and multi unit assignment problems as well as in the full preference domain. Another open direction is to provide a parametric analysis of the matching mechanisms according to the risk aversion factor.

REFERENCES

[1] A. Abdulkadiroglu, P. A. Pathak, and A. E. Roth. Strategy-proofness versus efficiency in matching with indifferences: redesigning the new york city high school match. Technical report, National Bureau of Economic Research, 2009.

[2] A. Abdulkadiroğlu and T. Sönmez. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998.

[3] A. Abdulkadiroğlu and T. Sönmez. House allocation with existing tenants. *Journal of Economic Theory*, 88(2):233–260, 1999.

[4] K. J. Arrow. Essays in the theory of risk-bearing. 1974.

[5] I. Ashlagi, F. Fischer, I. A. Kash, and A. D. Procaccia. Mix and match: A strategyproof mechanism for multi-hospital kidney exchange. *Games and Economic Behavior*, 2013.

[6] H. Aziz, F. Brandt, and M. Brill. The computational complexity of random serial dictatorship. *Economics Letters*, 121(3):341–345, 2013.

[7] H. Aziz, S. Gaspers, S. Mackenzie, N. Mattei, N. Narodytska, and T. Walsh. Equilibria under the probabilistic serial rule. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 1105–1112. AAAI Press, 2015.

[8] H. Aziz, S. Gaspers, S. Mackenzie, N. Mattei, N. Narodytska, and T. Walsh. Manipulating the probabilistic serial rule. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, pages 1451–1459, 2015.

[9] H. Aziz, S. Gaspers, S. Mackenzie, and T. Walsh. Fair assignment of indivisible objects under ordinal preferences. *Artificial Intelligence*, 227:71–92, 2015.

[10] A. Bogomolnaia and E. J. Heo. Probabilistic assignment of objects: Characterizing the serial rule. *Journal of Economic Theory*, 147(5):2072–2082, 2012.

[11] A. Bogomolnaia and H. Moulin. A new solution to the random assignment problem. *Journal of Economic Theory*, 100(2):295–328, 2001.

[12] S. Bouveret and J. Lang. Manipulating picking sequences. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, Prague, Czech Republic, August 2014. IOS Press.

[13] E. Budish and E. Cantillon. The multi-unit assignment problem: Theory and evidence from course allocation at harvard. *American Economic Review*, 102(5):2237–71, 2012.

[14] Y.-K. Che and F. Kojima. Asymptotic equivalence of probabilistic serial and random priority mechanisms. *Econometrica*, 78(5):1625–1672, 2010.

[15] C. Domshlak, E. Hüllermeier, S. Kaci, and H. Prade. Preferences in AI: An overview. *Artificial Intelligence*, 175(7):1037–1052, 2011.

[16] Ö. Ekici and O. Kesten. An equilibrium analysis of the probabilistic serial mechanism. *International Journal of Game Theory*, pages 1–20, 2015.

[17] P. C. Fishburn. Lexicographic orders, utilities and decision rules: A survey. *Management Science*, pages 1442–1471, 1974.

[18] J. Hadar and W. R. Russell. Rules for ordering uncertain prospects. *The American Economic Review*, pages 25–34, 1969.

[19] H. Hosseini and K. Larson. Strategyproof quota mechanisms for multiple assignment problems. *arXiv preprint arXiv:1507.07064*, 2015.

[20] D. Hugh-Jones, M. Kurino, and C. Vanberg. An experimental study on the incentives of the probabilistic serial mechanism. Technical report, Discussion Paper, Social Science Research Center Berlin (WZB), Research Area Markets and Politics, Research Unit Market Behavior, 2013.

[21] F. Kojima. Random assignment of multiple indivisible objects. *Mathematical Social Sciences*, 57(1):134–142, 2009.

[22] F. Kojima and M. Manea. Incentives in the probabilistic serial mechanism. *Journal of Economic Theory*, 145(1):106–123, 2010.

[23] Q. Liu and M. Pycia. Ordinal efficiency, fairness, and incentives in large markets. *Unpublished mimeo*, 2013.

[24] M. Manea. Asymptotic ordinal inefficiency of random serial dictatorship. *Theoretical Economics*, 4(2):165–197, 2009.

[25] D. Manlove. *Algorithmics of matching under preferences*. World Scientific Publishing, 2013.

[26] A. McLennan. Ordinal efficiency and the polyhedral separating hyperplane theorem. *Journal of Economic Theory*, 105(2):435–449, 2002.

[27] T. Mennle and S. Seuken. Hybrid mechanisms: Trading off strategyproofness and efficiency in one-sided matching. Technical report, Working Paper, 2014.

[28] T. Mennle, M. Weiss, B. Philipp, and S. Seuken. The power of local manipulation strategies in assignment mechanisms. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 82–89. AAAI Press, 2015.

[29] S. Pápai. Strategyproof multiple assignment using quotas. *Review of Economic Design*, 5(1):91–105, 2000.

[30] P. A. Pathak. Lotteries in student assignment. *Unpublished mimeo, Harvard University*, 2006.

[31] A. E. Roth, T. Sönmez, and M. U. Ünver. Kidney exchange. *The Quarterly Journal of Economics*, 119(2):457–488, 2004.

[32] D. Saban and J. Sethuraman. A note on object allocation under lexicographic preferences. *Journal of Mathematical Economics*, 2013.

[33] D. Saban and J. Sethuraman. The complexity of computing the random priority allocation matrix. *Mathematics of Operations Research*, 40(4):1005–1014, 2015.

[34] L. J. Schulman and V. V. Vazirani. Allocation of divisible goods under lexicographic preferences. *arXiv preprint arXiv:1206.4366*, 2012.

[35] T. Sönmez and M. U. Ünver. Course bidding at business schools. *International Economic Review*, 51(1):99–123, 2010.

[36] L.-G. Svensson. Strategy-proof allocation of indivisible goods. *Social Choice and Welfare*, 16(4):557–567, 1999.

[37] J. Von Neumann. A certain zero-sum two-person game equivalent to the optimal assignment problem. *Contributions to the Theory of Games*, 2:5–12, 1953.