Strategyproof Peer Selection: Mechanisms, Analyses, and Experiments

Haris Aziz
Data61 and UNSW
Sydney, Australia
haris.aziz@nicta.com.au

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

Nicholas Mattei
Data61 and UNSW
Sydney, Australia
nicholas.mattei@nicta.com.au

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

Toby Walsh
Data61 and UNSW
Sydney, Australia
toby.walsh@nicta.com.au

Abstract

We study an important crowdsourcing setting where agents evaluate one another and, based on these evaluations, a subset of agents are selected. This setting is ubiquitous when peer review is used for distributing awards in a team, allocating funding to scientists, and selecting publications for conferences. The fundamental challenge when applying crowdsourcing in these settings is that agents may misreport their reviews of others to increase their chances of being selected. We propose a new strategyproof (impartial) mechanism called Dollar Partition that satisfies desirable axiomatic properties. We then show, using a detailed experiment with parameter values derived from target real world domains, that our mechanism performs better on average, and in the worst case, than other strategyproof mechanisms in the literature.

1 Introduction

The problem arising from using peer review to select contestants has been well known for millennia: people might report untruthful valuations of others in order to improve their own chances of selection. The problem has been referred to as the *peer selection problem*. Various techniques have been attempted to solve it, most of which focused on reducing the influence of other participants on the selection of winners, for example, by using lotteries to either replace selection or to be a heavy part of the process (Mowbray and Gollmann 2007), or by using disinterested panels to prevent the participants of the contest from any influence at all.

Despite these measures, various settings in which each participant is both a candidate and a voter persist, for example, in the academic review process, various funding settings (the National Science Foundation (NSF) in the US is seeking to apply it to its funding allocation (Hazelrigg 2013)), and award decisions. Moreover, as the use of crowdsourcing in the online world is growing—from grading tasks in MOOCs to evaluating code in TopCoder (Lakhani, Garvin, and Lonstein 2010)—the need for desirable peer selection mechanisms becomes more and more important. All these settings can be captured by a simple model: agents rate or rank one another (or some subset of their peers), and based on these reports a specified number of agents are selected. This problem has received much recent attention (Alon et al. 2011; Fischer and Klimm 2014; Holzman and Moulin 2013; Kurokawa et al. 2015; Roos, Rothe, and Scheuermann 2011). Since peer review can be costly in terms of time and effort, each agent may be asked to review only a subset of other agents (Kurokawa et al. 2015; Merrifield and Saari 2009).

The main challenge in the peer selection problem is to propose strategyproof (also called impartial) mechanisms in which agents cannot increase their chances of being selected by misreporting.[1] Natural approaches to solve the peer selection problem, such the application of a voting rule, are *not* strategyproof (also called manipulable). A different approach to the selection problem is to use mechanisms similar to the Page Rank algorithm that use Markov chains to compute a ranking of the agents (Walsh 2014). Unfortunately, these approaches are also manipulable. If a mechanism is manipulable, the normative properties of the mechanism (that hold under the assumption that agents do not misreport) may not hold in general. Hence, we primarily focus on strategyproof mechanisms, although we also consider some natural manipulable mechanisms as well to provide a "best case" comparison.

Strategyproofness can be trivially satisfied by randomly selecting k agents or selecting the best k agents according to one particular agent (as do dictatorship mechanisms in social choice), but as we wish to select "quality" winners, we want to incorporate the views of all the agents. An established way to achieve strategyproofness is to partition the agents into a set of clusters and then select a specified number of agents from each cluster based on the reports of agents *outside* the cluster (Alon et al. 2011). We combine this partitioning principle with ideas from a fair division rule to divide a dollar (Dollar for short) (de Clippel, Moulin, and Tideman 2008), to create a novel mechanism called **Dollar Partition**.

Contributions: We provide a comprehensive survey and study of existing peer selection mechanisms. We show that iteratively using a strategyproof mechanism for $k = 1$ to select more than one agent is *not* strategyproof. We conduct a detailed experimental comparison with previously introduced strategyproof mechanisms with regard to their ability to recover the "ground truth". This is the first experimental

[1] A strict incentive to report truthfully is not possible in strategyproof mechanisms, so strictly better mechanisms are impossible.

comparison of the performance of strategyproof peer selection mechanisms.

Our main contribution is a novel peer selection mechanism (Dollar Partition) that satisfies desirable axiomatic properties including strategyproofness and natural monotonicity properties. We prove that although Dollar Partition relies on the Dollar mechanism to share a bonus (de Clippel, Moulin, and Tideman 2008), three natural peer selection mechanisms inspired from Dollar are manipulable. Dollar Partition has a number of advantages over previously introduced strategyproof mechanisms including the Partition mechanism and the Credible Subset mechanism (Kurokawa et al. 2015). Dollar Partition returns better quality outcomes ex post than Partition if one cluster has most of the top-ranked candidates. In contrast to Credible Subset, Dollar Partition never returns an empty set.

Experimentally, Dollar Partition selects more agents from a higher grade more often, selects more agents from higher grades in the worst case, and does so more consistently, than any other strategyproof mechanism in the literature. In the worst case we found that Dollar Partition provides a $\geq 17\%$ improvement over the agents selected by Partition. Additionally, as the proportion of reviews per agent increases, Dollar Partition performs increasingly better than Partition, Credible Subset, and the other Dollar adaptations.

2 Related Work

The criticism that prominent peer selection mechanisms such as ones under consideration by American and European funding bodies (Merrifield and Saari 2009; Hazelrigg 2013) are *not* strategyproof (Naghizadeh and Liu 2013) has underscored the need to devise mechanisms with better incentive properties. The literature most directly relevant to this article is a series of papers on strategyproof (impartial) selection initiated by Holzman and Moulin (2013) and Alon et al. (2011). We overview these mechanisms in Section 3. Most of the work on strategyproof peer selection focuses on the setting in which agents simply approve (nominate) a subset of agents (Alon et al. 2011; Bousquet, Norin, and Vetta 2014; Fischer and Klimm 2014; Holzman and Moulin 2013) with the latter three restricting attention to the setting in which exactly one agent is selected ($k = 1$). Mackenzie (2015) and Berga and Gjorgjiev (2014) focused on axiomatic aspects of peer selection. Kurokawa et al. (2015) presented an interesting strategyproof mechanism (Credible Subset) that performs well when each agent reviews a few other agents.

The peer selection problem is also related to peer-based grading/marking (Alfaro and Shavlovsky 2014; Joachims and Raman 2015; Kulkarni et al. 2013; Piech et al. 2013; Robinson 2001; Walsh 2014; Wright, Thornton, and Leyton-Brown 2015) especially when students are graded based on percentile scores. For peer grading, mechanisms have been proposed that make a student's grade slightly dependent on the student's grading accuracy (see e.g., Walsh (2014) and Merrifield and Saari (2009)). However such mechanisms are not strategyproof since one may alter one's reviews to obtain a better personal grade.

3 Setup and Survey of Existing Mechanisms

We have a set N of agents $\{1, \ldots, n\}$. Each agent reports a valuation over the other agents. These messages could be cardinal valuations $v_i(j)$ for agent i's valuations of agent j, or they could be a weak order reported by agent i of agents in $N \setminus \{i\}$, which may be transformed to cardinal valuations using some predetermined scoring rule. An agent, depending on the setting, evaluates some $0 \leq m \leq n-1$ of the other agents. Based on these messages, around k agents are selected. Some mechanisms such as Credible Subset and Dollar Partition may not always return a size of exactly k even if the *target* size is k.

The general idea of the partitioning based mechanisms is to divide the agents into a set of clusters $\mathcal{C} = \{C_1, \ldots C_\ell\}$. This clustering can be done using a random process or by a predetermined order, without adding randomness to the process. We will often abuse notation and refer to the value that one cluster has for another cluster $v_{C_i}(C_j)$; the valuation of all agents in C_i for the agents in C_j: $\sum_{r \in C_i, j \in C_j} v_r(j)$.

3.1 Mechanisms

We outline some prominent peer selection mechanisms.

- **Vanilla**: select the k agents with highest total value based on their reviews by other agents. Vanilla is not strategyproof; unselected agents have an incentive to lower the reported valuations of selected agents.
- **Partition**: divide the agents into ℓ clusters. and select a preset number of agents from each cluster, typically k/ℓ, according to the valuations of the agents *not* in that cluster. This class of mechanisms is a straightforward generalization of the Partition mechanism studied by Alon et al. (2011) (and in an early version of Kurokawa et al. (2015)) and is strategyproof.
- **Credible Subset** (Kurokawa et al. 2015): let T be the set of agents who have the top k scores. Let P be the set of agents who do not have the top k scores but will make it to the top k if they do not contribute any score to all other agents. With probability $(k+|P|)/(k+m)$, Credible Subset selects a set of k agents uniformly at random from $T \cup P$, and with probability $1 - (k + |P|)/(k + m)$, it selects no one. The mechanism is strategyproof.

Some other mechanisms are tailor-made for $k = 1$ and for the case where agents only approve a subset of agents: Partition by Holzman and Moulin (2013); Permutation by Fischer and Klimm (2014); and Slicing by Bousquet, Norin, and Vetta (2014). We take inspiration from the Dollar mechanism designed to split a divisible resource (de Clippel, Moulin, and Tideman 2008).

- **Dollar**: Each agent i has a value $v_i(j)$ of his estimation of how much j should get. We assume these values are normalized so that $\sum_{j \in N \setminus \{i\}} v_i(j) = 1/n$. Then the *Dollar share* of each agent i is $x_i = \sum_{j \in N \setminus \{i\}} v_j(i)$.

3.2 Properties of Mechanisms

We consider some basic axioms of peer selection mechanisms: (i) *Non-imposition*: for any target set W, there is

a valuation profile and a randomization seed that achieves W; (ii) *Strategyproofness (Impartiality)*: agents cannot affect their own selection; (iii) *Monotonicity*: if an agent i is selected, then if in a modified setting in which that agent is reinforced (score or ordinal ranking of i is improved with respect to other agents) by some other agents and no other agent's value/ranking improved, then i will remain selected; (iv) *Committee Monotonicity*: if W is the outcome when the target set size is k, then the agents in W are still selected if the target set size is $k+1$. (The final two properties are in expectation when algorithms involve randomization.)

4 Dollar Partition

Dollar Partition is formally described as Algorithm 1. The algorithm works as follows. First, agents are partitioned such that the difference between the sizes of any two clusters is at most one. Each agent $i \in N$ assigns a value $v_i(j)$ to each agent j that is in a cluster other than i's cluster and j is among the m agents that i reviews. Agent i may directly give a cardinal value to the agents he reviews or the cardinal value may be obtained by a scoring function that converts the ordinal ranking given by i to cardinal values. In either case, the values that i gives are normalized so that agent i assigns a total value of 1 to agents outside his own cluster. Based on the values from agents outside the cluster, each cluster C_j gets a normalized weight of x_j, its assigned share. Based on each share x_i, each cluster gets a quota $s_i = x_i \cdot k$ which may be a non-integer number.[2] If all s_i's are integers that are at most the size of the corresponding cluster, then each s_i is the quota of cluster C_i, i.e., exactly s_i agents are selected from cluster C_i. If not all s_i are integers, we take the ceiling of each s_i and use this as the quota for C_i. Note that the size of the winning set may be larger than k: $|W| \leq k + \ell - 1$.[3] As long as ℓ is a small integer and k is reasonably large then $|W| \approx k$.[4]

Note that if the partitioning into clusters and the review allocation can be done in a deterministic manner (e.g., lexicographic ordering), Dollar Partition is deterministic. We first prove that Dollar Partition is strategyproof.

Theorem 1 *Dollar Partition is strategyproof.*

Proof: Suppose agent i is in cluster C_j of the generated partition. Agent i will be selected in W if and only if its rank according to the v scores given by agents outside of C_j is at least t_j. Therefore agent i can either manipulate by increasing t_j or by increasing its score relative to other agents in C_j given by agents outside C_j. Since agent i cannot affect the latter, the only way it can manipulate is by increasing t_j. We argue that agent i cannot change t_j by changing his valuation v_i for agents outside the cluster. Note that i contributes a probability weight of $1/n$ to agents outside C_j and zero probability weight to agents in C_j. Hence it cannot affect the value x_j of cluster C_j. As s_j is derived from x_j, agent i cannot affect the number $t_j = \min(\text{Ceiling}(s_j), |C_j|)$. □

Input: (N, v, k); the algorithm is parametrized by number of clusters $2 \leq \ell \leq n$ and m, the number of reviews per agent.
Output: $W \subset N$ such that $|W| \leq k + \ell - 1$

1. Initialize $W \longleftarrow \emptyset$
2. Generate a partition $\{C_1, \ldots, C_\ell\}$ of N in which the difference between the sizes of any two clusters is at most 1. Let $C(i)$ be the cluster of agent i.
3. Each $i \in N$ reviews m agents outside $C(i)$. Ensure $v_i(j) = 0$ for $j \in C(i)$ and that $\sum_{j \notin C(i)} v_i(j) = 1$ by setting the valuation of agent i for agents in its cluster to 0 and normalizing.
4. x_i value of a cluster C_i is defined as follows:
$$x_i \longleftarrow \frac{1}{n} \times \sum_{j \in C_i, j' \notin C_i} v_{j'}(j).$$
5. Using the x_i values, we compute the number of agents t_i to be chosen from each cluster C_i. We first compute each s_i.
$$s_i \longleftarrow x_i * k \text{ for each } i \in \{1, \ldots, \ell\}.$$
6. **for** each $i \in \{1, \ldots, \ell\}$ **do**
7. $\quad t_i \longleftarrow \min(\text{Ceiling}(s_i), |C_i|)$.
8. For each $i \in C(i)$, the score of agent i is $\sum_{i' \notin C(i)} v_{i'}(i)$.
9. Select t_j agents with the highest scores from each cluster C_j and place them in set W.
10. **return** W

Algorithm 1: Dollar Partition

Dollar Partition easily satisfies non-imposition; we show it satisfies other key monotonicity properties.

Theorem 2 *Dollar Partition is monotonic.*

Proof: Let us compare the valuation profile v when i is not reinforced and v' when i is reinforced. The relative ranking of i is at least as high when i is reinforced. Since any decrease in valuation that an agent j in $C(i)$ receives translates into the same increase in the valuation received by agent i, hence the total valuation that $C(i)$ receives does not decrease and hence the number of agents to be selected from $C(i)$ is at least as high as before. □

Theorem 3 *Dollar Partition is committee monotonic.*

Proof: The only difference between running the algorithm for different target k values is when calculating the share vector $\vec{s}$. However, if agent i in cluster C_j was selected, that means its ranking in the cluster C_j was above $t_j = \min(\text{Ceiling}(s_i), |C_j|)$. When k increases, s_i will only increase (as x_i remains the same), and hence so will t_j, ensuring that i will be selected again. □

Although Dollar Partition draws inspiration from Dollar and Partition, it has key differences from these approaches.

[2] After AAAI-2016 publication, we recognized our (required) implicit assumption that $k \leq n/\ell$.

[3] In practical settings this may not be a problem, since often a couple of more winners can be accommodated (e.g., shortlists etc. in case of agents declining awards).

[4] One would think that if each cluster does not get an integer share, then one can consider various apportionment rules, which have been suggested in the literature in the past several centuries, in particular in political science (Young 1994). However, applying these rules violates the strategyproofness of the overall mechanism.

Comparison with other Dollar Based Mechanisms: Although Dollar Partition is partly based on the Dollar mechanism for dividing a bonus, it is more desirable than other peer selection mechanisms based on the Dollar framework:

- **Dollar Raffle** computes the relative fractions of how much of a dollar each agent should get via the Dollar mechanism of de Clippel, Moulin, and Tideman (2008). Using these shares as probabilities, do the following k times: randomly select an agent according to its dollar share probabilities until k different agents are selected.
- **Dollar Partition Raffle** takes the Dollar shares of the *clusters* in Dollar Raffle and uses these shares to define a probability distribution over the clusters. Until k different agents are selected, a cluster is drawn with respect to the Dollar probabilities over the clusters and the next best agent (based on reviews of agents outside the cluster) is selected. If all agents in the clusters have been selected, then no agent is selected.
- **Top Dollar** selects agents with maximum Dollar shares.[5]

Dollar Raffle relies too much on randomization and gives even the worst agents non-zero probability of being selected. Dollar Raffle is strategyproof for $k = 1$ but not for $k > 1$.

Theorem 4 *Dollar Raffle is not strategyproof for $k > 1$.*

The argument for the proof is as follows. The mechanism iterates until it chooses k different agents, which is equivalent to eliminating each selected agent and re-normalizing the dollar partitions, as once some agent is selected we ignore its repeated selection. This re-normalization prevents the mechanism from being strategyproof, as now the probabilities of others matter for each agent. For example, an agent will prefer to contribute to a very strong agent (which, once eliminated, will make our agent's probability increase significantly). The proof of this theorem carries on to the various mechanisms presented for $k = 1$ (e.g., (Fischer and Klimm 2014)): simply running the algorithm several times destroys their strategyproofness. This is true even for mechanisms that are strategyproof for $k = 1$, as long as any agent has the power to influence the outcome (i.e., not purely random, a dictatorship, or a combination of both).

Although Dollar Partition Raffle relies less on the "luck of the draw", it still has non-zero probability of selecting the worst k agents if the same cluster (consisting of the worst agents) is selected repeatedly. Dollar Partition Raffle is equivalent to Dollar Raffle if $\ell = n$ and hence is not strategyproof. Note that Dollar Partition Raffle is only strategyproof for $k < \min_{j\in\{1,\ldots,\ell\}}(|C_j|)$, otherwise this mechanism encounters the same problem as Dollar Raffle. Finally, **Top Dollar** requires no randomization but it is highly manipulable as an agent who is not selected may get selected by giving lower scores to the agents who are selected.

Comparison with the Partition Mechanism: Dollar Partition seems similar to the Partition mechanism but whereas Partition is too quick to preset the number of agents to be selected from each cluster, Dollar Partition *relies on the peer reviews* to decide the number of agents to be selected from each cluster. This difference allows Dollar Partition to have more consistent performance, no matter the clustering. Hence, in contrast to Dollar Partition, the rigidity of Partition means that it may not choose a large proportion of the best agents even if agents have unanimous valuations.

Example 5 *Consider the setting in which $N = \{1, \ldots, 18\}$, $k = 6$, and $\ell = 3$. Let the clusters be $C_1 = \{1, \ldots, 6\}$, $C_2 = \{7, \ldots, 12\}$, $C_3 = \{13, \ldots, 18\}$. C_1 puts all its weight on C_2, equally dividing its points between $7, 8, \ldots, 12$, with a slight edge to 7 and 8, C_2 and C_3 put all the weight on C_1, dividing their points between $1, 2, 3$ and 4. Now Partition will choose $1, 2, 7, 8, 13, 14$ where everyone thinks that $1, 2, 3, 4, 7, 8$ are the best. Dollar Partition will select exactly that set. Moreover, if we increase the number of clusters, the disparity between Dollar Partition and Partition only grows.*

The issue with Partition is that in contrast to Dollar Partition it performs poorly ex post[6] if the clusters are lopsided with some cluster containing all good agents and other clusters containing low value agents. One natural fix is that we deliberately choose a balanced partition where the weight of a cluster is based on the ratings of vertices outside the cluster and we want to minimize the margin of the cluster weights. However for this and various notions of balanced partitions, computing the most balanced partition is NP-hard. What is even more problematic is that if we choose a balanced partition, the resulting mechanism is not strategyproof.

We point out that there are instances where Partition may perform better than Dollar Partition even if the rankings of the agents are unanimous. Consider a case where a highly preferred agent is in the same group as the lowest preferred agents, whereas other groups only contain medium preferred agents. In that case the weight of the cluster with the highest preferred agent will be so high that lowest ranked agents will also be selected (this does not work for Borda utilities). The normalization of scores entailed in Dollar Partition causes a certain loss of information and granularity when compared to the other mechanisms. However, even in the example in the remark above, we notice that Dollar Partition will ensure that when agents have highly correlated or unanimous preferences, the agent(s) that are unanimously on the top will be selected, even if some low-ranked agents are also selected.

5 Simulation Experiments

Using Python and extending code from PREFLIB (Mattei and Walsh 2013) we have implemented the Dollar Partition, Credible Subset, Partition, Dollar Raffle, Dollar Partition Raffle, and Vanilla peer selection mechanisms. All the code developed for this project is implemented as an easily installable Python package available on GitHub free and open-source under the BSD license. We present results on the first systematic empirical study of strategyproof selection mechanisms. As in all experiments based on simulations there are

[5] When agents' valuations are normalized, Vanilla is equivalent to Top Dollar.

[6] For high stake outcomes, we want a mechanism that performs well on average and never returns an especially bad outcome.

a plethora of decisions that must be made. While the choice of parameter and model values can have significant impacts on the outcomes of these studies (see e.g., (Popova, Regenwetter, and Mattei 2013)) we have chosen an experimental setting that we feel is both well-motivated and closely models real-world settings.

Experimental Setup: Given n agents divided into l clusters with each agent performing m reviews we want to select k agents. We first generate the scoring matrix (profile) via a two-step process using a *Mallows Model* to generate the underlying ordinal evaluation (Mallows 1957). Mallows models are parameterized by a *reference order (σ)* and a *dispersion parameter (ϕ)*. Intuitively, we can think of ϕ as the probability of committing ranking error by swapping neighboring elements according to σ (Lu and Boutilier 2011). Mallows models are used when each agent is assumed to have the same reference ranking subject to noise.

Each of the algorithms takes as input a (sparse) $n \times n$ score matrix. In most settings where peer selection is used there is a set of scores that can be given by a reviewer. This creates a set of equivalence classes of proposals that are assigned the same overall score. For example, when reviewing papers for a conference, a reviewer may assign the highest score to only a very small percentage of papers if he were to see all of the papers. We suppose that agents are able to express these equivalence classes by assigning a set number of *grades*, G. To generate our input we define two functions F and D common to all agents (this is generalizable in our testing framework) that describe the scoring and distribution, respectively. For example, using Borda scoring where $F : [4, 3, 2, 1, 0]$ and a distribution function $D : [0.2, 0.2, 0.2, 0.2, 0.2]$; all agents in the top 20% of agents receive a score of 4, the next 20% a score of 3, and so on. The functions D and F are passed as parameters to the profile generator, allowing flexibility in testing. Formally, first we generate a complete, strict relation for agent i. Given a probability density function (PDF) D for each grade $g \in G, D(g) \to R^+$ where $\sum_{g \in G} D(g) = 1.0$ and a scoring function F for each grade $g \in G, F(g) \to Z^+$.

Each agent reviews m of the n proposals and is also reviewed by m other agents. Since we are dealing with clusters, we additionally have the constraint that each agent reviews m agents *outside* his cluster. We refer to review assignments satisfying these constraints as balanced m-regular assignments. We convert a complete $n \times n$ score matrix into a sparse score matrix by drawing a balanced m-regular assignment with respect to a given clustering. In order to maximize inter-cluster comparison, we would also like that the m agents that agent i is to review are reasonably balanced among the clusters (not including i's cluster) so that each agent in each cluster C_i reviews in total $\frac{|C_i| \cdot m}{\ell - 1}$ agents from each other cluster. We generate this assignment randomly and as close to balanced as possible. Given the balanced m-regular assignment for agent i, we remove all other candidates from i's complete score vector. Hence we are left with a sparse, m-regular score matrix which respects a clustering of the agents into ℓ clusters. The resulting score matrix resembles what a conference organizer or NSF program manager sees: a sparse and noisy observation of the ground truth filtered through equivalence classes

Results for an NSF-Like Program: Using numbers from the NSF[7] we settled on a set of realistic parameters that one may see in the real world. The "Mechanism Design" pilot, which used the mechanism proposed by Merrifield and Saari (2009) had 131 proposals, with each submitter reviewing 7 other proposals. The acceptance numbers are not broken out from the global acceptance rate for the program. Consequently we assume an $\approx$20% acceptance rate, the same as NSF as a whole and also similar to other conference and funding acceptance rates.

We use a "normal" distribution giving $|D| = [4, 7, 15, 20, 39, 20, 15, 7, 3]$ and a Borda scoring function that one would expect to find in most conference reviewing $F = [8, 7, 6, 5, 4, 3, 2, 1, 0]$ corresponding to the grades $G = [A+, A, B+, B, C+, C, D+, D, F]$. Without loss of generality we assume that the ground truth ordering σ is in agent order, i.e., $1, \ldots, 130$. The ground truth ordering σ gives us an indication of which agents are objectively better than the others. However, this ground truth is filtered not only through the noise of the individual agents (ϕ) but also by the inexactness of the m-regular assignment. Given D and k we can establish how many of the selections *should* come from each grade. In a competitive setting, we want to select those agents at the top of the ground truth ordering.

Figure 1 shows the performance of the six mechanisms discussed on two different metrics as we vary the number of reviews received. We fixed $\phi = 0.1$ for this testing as setting $\phi \in \{0.0, 0.1, 0.25, 0.4\}$ had no significant effect. The graphs show the mean cumulative proportion of the agents in each grade that are selected by each of the mechanisms over 1000 samples. For instance, the 1.0 score received by Vanilla for both A+ and A+|A for all settings of m mean that Vanilla always selects the 11 highest scoring agents in the ground truth ranking (σ). We use cumulative selection with respect to the ground truth ordering. This partial sum is well defined for each set of grades and clearly shows where a particular mechanism is over- or under-performing. Each mechanism was allowed to select a number of proposals equal to the number of agents returned by Dollar Partition per iteration, hence the average cumulative selection is > 1.0. Whilst Vanilla is the best in our experiment, strictly dominating all other mechanisms, it is the only non-strategyproof mechanism. In practice, agents may not report truthfully with Vanilla and so it can perform much worse. The other generalizations of Dollar are strictly dominated by Dollar Partition; our more nuanced mechanism yields a better selection.

Comparing Dollar Partition and Partition ($m = \{10, 15\}$), both mechanisms select all of the A+ grade agents on every iteration. Partition selects only $9/11$, in the worst case, of the A+|A, while Dollar Partition selects $10/11$, an 11% improvement. Considering the A+|A|B+ agents, Partition only selects $17/26$, while Dollar Partition selects $20/26$, a $\geq 17\%$ performance increase. Neither mechanism ever selects an agent with rank lower than C+; even in the worst case, both perform better than every other strategyproof mechanism in

[7] http://www.nsf.gov/nsb/publications/2015/nsb201514.pdf

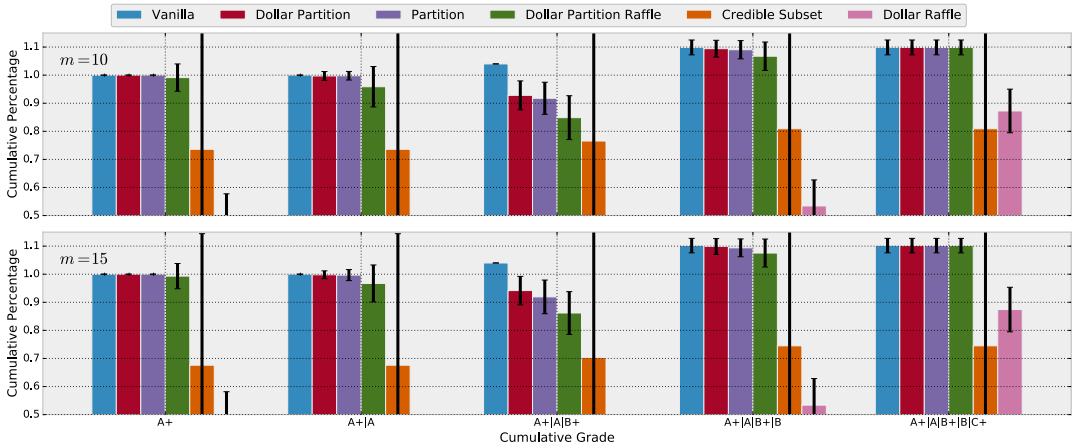

Figure 1: Mean cumulative percentage of each grade of agent selected by the six peer selection algorithms presented in this paper on 1000 random iterations selecting $k = 25$ agents from a population of $n = 130$ agents providing $m = 10$ (top) and $m = 15$ (bottom) reviews divided into $l = 5$ clusters with a Mallows dispersion $\phi = 0.1$. To enable comparisons, every mechanism selects $|W|$ equal to that of Dollar Partition; hence the ≥ 1.0 averages as $k = 25$ is the denominator. Error bars represent one standard deviation from the mean. Dollar Partition selects more agents from a higher grade more often, selects more agents from a higher grade in the worst case, and does so more consistently, than any other strategyproof mechanism. To highlight Partition and Dollar Partition we have cropped results where they are the same (cutting off Dollar Raffle).

our study.[8] Standard deviation is also higher for Partition for all these cases, indicating Partition is much more likely to make mistakes and select agents from a lower grade over agents in a higher grade. Dollar Partition performs better than Partition in the worst case, and performs better on average. In a low information setting (i.e., $m \leq 5$), Partition does perform slightly better on average than Dollar Partition. However, Dollar Partition shows a lower variance and better worst case performance across all settings to m, demonstrating its robustness to lopsided clusterings.

General Results: We explored a realistic part of the large parameter space to investigate the mechanisms. The practical upshot, after running hundreds of thousands of instances, is that there are numerous tradeoffs that system designers must consider, critically depending on their target domain. In general, varying other parameters, such as k, ℓ, D and F did not change the ranking of mechanisms shown here. However, increasing the number of clusters improved Dollar Partition's performance in comparison to Partition's, which may stem from the increased chance that Partition will select the bottom candidates of a given cluster instead of better ranked candidates in a different cluster. Accordingly, as it generally selects the top candidates, Partition's performance improves when scoring rules are exponential in comparison to less extreme scoring rules, such as Borda.

Dollar Partition is much better when there is sufficient information, in terms of the number of reviews and the granularity of the grades, to have a chance of recovering the ground truth ordering. Settings like conferences with $n = 2000$ papers and $m = 5$ reviews split into 5–8 grades often have no clear cutoff between accept and reject; the grades contain too many items. In these cases all the mechanisms perform poorly, as selecting a set of winners is akin to randomly selecting agents from the set of possible winners. See, e.g., the NIPS experiment[9] and the recent paper on the limits of noisy rank aggregation using data from the KDD conference (Joachims and Raman 2015). As the ratio of m to n grows, and the granularity of the grades increases, it becomes possible to recover the ground truth ranking, and Dollar Partition outperforms the other mechanisms.

6 Conclusion

We introduce a novel peer selection mechanism—Dollar Partition. Overall, Dollar Partition's flexibility in setting the number of agents to be selected from each cluster addresses the worst-case instances where partitions may be lopsided, allowing Dollar Partition to reach higher quality, more consistent results than existing mechanisms. Combined with the ability to always return a winning set, it is an improvement over current mechanisms.

Among strategyproof mechanisms, Partition and Dollar Partition may have a certain 'psychological' advantage: they may incentivize agents to report truthfully because an agent's contribution in selecting other agents (with whom he is not competing) is more direct. Moreover, partitioning into groups helps deal with conflict of interest cases, when there is fear of collusion among several agents; putting them in the same cluster prevents them from influencing one another's chance of success. Peer selection is a fundamental

[8] It is hard to directly compare results for Credible Subset due to the large probability of returning an empty set. This problem is not easily overcome; removing the ability to return an empty set means Credible Subset is no longer strategyproof. When Credible Subset does return a set, it slightly outperforms other mechanisms.

[9] http://blog.mrtz.org/2014/12/15/the-nips-experiment.html

problem that has received less attention than voting rules. We envisage the need to develop robust solutions with good incentive properties, as these are widely applicable in large-scale, crowdsourcing settings.

Acknowledgments

Data61 (formerly known as NICTA) is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program. This research has also been partly funded by Microsoft Research through its PhD Scholarship Program, and by Israel Science Foundation grant #1227/12. This work has also been partly supported by COST Action IC1205 on Computational Social Choice.

References

Alfaro, L. D., and Shavlovsky, M. 2014. CrowdGrader: Crowdsourcing the Evaluation of Homework Assignments. In *Proceedings of the ACM Technical Symposium on Computer Science Education (ACM-SIGCSE)*, 415–420.

Alon, N.; Fischer, F.; Procaccia, A. D.; and Tennenholtz, M. 2011. Sum of us: Strategyproof selection from the selectors. In *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK)*, 101–110.

Berga, D., and Gjorgjiev, R. 2014. Impartial social rankings. Working paper.

Bousquet, N.; Norin, S.; and Vetta, A. 2014. A near-optimal mechanism for impartial selection. In *Proceedings of the 10th International Workshop on Internet and Network Economics (WINE)*, Lecture Notes in Computer Science (LNCS).

de Clippel, G.; Moulin, H.; and Tideman, N. 2008. Impartial division of a dollar. *Journal of Economic Theory* 139:176–191.

Fischer, F., and Klimm, M. 2014. Optimal impartial selection. In *Proceedings of the 15th ACM Conference on Economics and Computation (ACM-EC)*, 803–820. ACM Press.

Hazelrigg, G. A. 2013. Dear Colleague Letter: Information to Principal Investigators (PIs) Planning to Submit Proposals to the Sensors and Sensing Systems (SSS) Program October 1, 2013, Deadline. NSF Website, http://www.nsf.gov/pubs/2013/nsf13096/nsf13096.jsp.

Holzman, R., and Moulin, H. 2013. Impartial nominations for a prize. *Econometrica* 81(1):173–196.

Joachims, T., and Raman, K. 2015. Bayesian ordinal aggregation of peer assessments: A case study on KDD 2015. Technical report, Cornell University.

Kulkarni, C.; Wei, K.; Le, H.; and Chia, K. D. 2013. Peer and self assessment in massive online classes. *ACM Transactions on Computer Human Interaction (TOCHI)* 20(6):1–31.

Kurokawa, D.; Lev, O.; Morgenstern, J.; and Procaccia, A. D. 2015. Impartial peer review. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press.

Lakhani, K. R.; Garvin, D. A.; and Lonstein, E. 2010. Topcoder (a): Developing software through crowdsourcing. *Harvard Business Review*.

Lu, T., and Boutilier, C. 2011. Learning Mallows models with pairwise preferences. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 145–152.

Mackenzie, A. 2015. Symmetry and impartial lotteries. *Games and Economic Behavior* 94(1):15–28.

Mallows, C. 1957. Non-null ranking models. *Biometrika* 44(1):114–130.

Mattei, N., and Walsh, T. 2013. Preflib: A library for preferences. HTTP://WWW.PREFLIB.ORG. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT)*, 259–270.

Merrifield, M. R., and Saari, D. G. 2009. Telescope time without tears: a distributed approach to peer review. *Astronomy & Geophysics* 50(4):4–16.

Mowbray, M., and Gollmann, D. 2007. Electing the Doge of Venice: Analysis of a 13th century protocol. In *Proceedings of the IEEE Symposium on Computer Security Foundations*, 295–310.

Naghizadeh, P., and Liu, M. 2013. Incentives, quality, and risks: A look into the NSF proposal review pilot. *arXiv preprint arXiv:1307.6528* 1–10.

Piech, C.; Huang, J.; Chen, Z.; Do, C.; Ng, A.; and Koller, D. 2013. Tuned models of peer assessment in MOOCs. In *Proceedings of the 6th International Conference on Educational Data Mining (EDM)*, 153–160.

Popova, A.; Regenwetter, M.; and Mattei, N. 2013. A behavioral perspective on social choice. *Annals of Mathematics and Artificial Intelligence* 68(1–3):135–160.

Robinson, R. 2001. Calibrated peer review an application to increase student reading and writing skills. *The American Biology Teacher* 63(7):474–476.

Roos, M.; Rothe, J.; and Scheuermann, B. 2011. How to calibrate the scores of biased reviewers by quadratic programming. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*, 255–260.

Walsh, T. 2014. The PeerRank method for peer assessment. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*, 909–914.

Wright, J.; Thornton, C.; and Leyton-Brown, K. 2015. Mechanical TA: Partially automated high-stakes peer grading. In *Proceedings of the ACM Technical Symposium on Computer Science Education (ACM-SIGCSE)*.

Young, H. P. 1994. *Equity: in Theory and Practice*. Princeton University Press.