Peer Selection with Noisy Assessments

Omer Lev
Ben-Gurion University, Israel
agent007@mi6.gov.uk

Nicholas Mattei
Tulane University, USA
nsmattei@tulane.edu

Paolo Turrini
University of Warwick, UK
p.turrini@warwick.ac.uk

Stanislav Zhydkov
University of Warwick, UK
s.zhydkov@warwick.ac.uk

ABSTRACT

In the peer selection problem a group of agents must select a subset of themselves as winners for, e.g., peer-reviewed grants or prizes. Here, we take a Condorcet view of this aggregation problem, i.e., that there is a ground-truth ordering over the agents and we wish to select the best set of agents, subject to the noisy assessments of the peers. Given this model, some agents may be unreliable, while others might be self-interested, attempting to influence the outcome in their favour. In this paper we extend PeerNomination, the most accurate peer reviewing algorithm to date, into WEIGHTEDPEERNOMINATION, which is able to handle noisy and inaccurate agents. To do this, we explicitly formulate assessors' reliability weights in a way that does not violate strategyproofness, and use this information to reweigh their scores. We show analytically that a weighting scheme can improve the overall accuracy of the selection significantly. Finally, we implement several instances of reweighing methods and show empirically that our methods are robust in the face of noisy assessments.

1 INTRODUCTION

The results of the 2016 NeurIPS experiment [20] and other studies of bias in evaluative processes [21, 24] have brought to the fore the extent of noisy assessments in conference reviewing. When agents evaluate each other to select a subset of themselves – the peer selection problem – various factors can come into play that hinder accurate assessment, including time pressure and strategic behaviour. Finding high quality mechanisms for peer review is a critical step in helping to ease reviewing in a number of areas, including large conferences [3], grant reviewing [17], and online courses [22].

Recently, researchers in algorithmic game theory and computational social choice have devised accurate and strategyproof algorithms for peer selection, such as Partition [1], Credible Subset [10], Exact Dollar Partition (EDP) [3] and most recently PeerNomination [15], each improving upon the state of the art. These algorithms take a Condorcet view on this aggregation problem, i.e., that there is a ground-truth over the agents, and we wish to select as many of the top ranked agents as possible [26]. However, none of the existing algorithms seek to alleviate the problem of noisy inputs in a unified, strategyproof mechanism. When earlier work did engage with noisy reports, it was limited to empirical testing with relatively low noise, e.g., a Mallows model with $\varphi = 0.5$ (see Aziz et al. [3] or Mattei et al. [15]), which yields fairly minor changes in agents' reports. We are concerned with algorithms that can handle a significant level of noise, while maintaining strategyproofness and high quality of selection, an important missing aspect in the peer reviewing literature.

Ideally, we want an algorithm that is capable of identifying inaccurate reviewers and reducing their influence on selection, using their own assessments as a guide. We could, for example, try and downgrade those reviewers that differ too much from others. However, there are two problems with this approach: first, the noise may be such that it is difficult to establish what the consensus actually is; and second, that this meta-level reweighing can be exploited strategically. Simple reweighing is not strategyproof. Consider an agent a that is harshly reviewing agent b, with both a and b reviewing a third agent c. Agent b could benefit by reviewing agent c in a way that would present agent a as one whom others disagree with, potentially lowering the impact of the report of agent a on agent b if weights are computed based on correlations to the evaluations of others, as done by Merrifield and Saari [17]. On the other hand, if a mechanism is able to identify agent b as a source of noise, it can increase the overall quality of the selection. While one can reweigh agents without maintaining strategyproofness [22, 25], we wish to achieve increased selection quality *and* strategyproofness.

Contribution. We extend PeerNomination, the most accurate peer reviewing algorithm to date, into WEIGHTEDPEERNOMINATION, to handle noisy and inaccurate agents. To do so we explicitly formulate assessors' reliability weights in a way that does not violate strategyproofness, and use this information to reweigh their scores. WEIGHTEDPEERNOMINATION is able to handle significant levels of noise, even when reviewers act adversarially. We show analytically that a weighting scheme can improve the overall accuracy of the selection significantly. We then implement several instances of reweighing protocols and show empirically that our novel methods

are able to significantly improve the quality of peer selection over PeerNomination, under a variety of noise parameters.

2 RELATED LITERATURE

Using the evaluations of peers to rank and select winners is a problem of broad interest beyond CS and AI, including numerous practical domains, e.g., conference, journal, and grant reviewing; large scale course grading, and group decision making. Brought to the fore by Merrifield and Saari [17] to allocate telescope time, the problem is deeply rooted in economics, from the work of de Clippel et al. [7] on "dollar partition", extended by Aziz et al. [3] to the Dollar Raffle and Dollar Partition methods. Other notable algorithms include the Credible Subset method [10], where the protocol examines the possibility of manipulations and accounts for it. Despite strategyproofness, the system was shown to yield significant number of cases where no proposal was funded in the end [2].

Two more prominent recent algorithms are Exact Dollar Partition (EDP) [3] which provides exactness at the cost of some randomness, while remaining strategyproof, and improving on earlier algorithms [3]. The second is PeerNomination [15], which improves upon EDP, while requiring reviewers to submit only approval based rankings, though at the cost of some exactness (details in Section 3).

Other developments in the multi-agent systems communities include voting rules to aggregate ranks, e.g., k-Partite [9], the Committee Rule [9], and Divide-and-Rank [27] algorithms. Others focus on proving bounds on the quality of a given rank aggregation scheme under noisy and partial observations [5]. Yet other methods are approval-based but focus on single agent selection: Permutation [8] and Slicing [4].

A key application area for peer evaluation mechanisms is education, where the problems of reviewer reliability and bias have been extensively studied [18]. We are motivated by evidence from fielded peer evaluation mechanisms showing that students are often unwilling to strictly rank assignments [6] and would rather rely on scores or pass/fail marks (approvals). Within the conference and journal reviewing ecosystem there is also growing interest in detecting strategic behaviour on the part of the reviewers [16, 21] as well as de-biasing and calibrating differences in the scores of reviewers [11, 23]. We go beyond calibration and de-biasing, identifying suboptimal behaviour in agents' populations and looking at the effect of rescaling on the system as a whole.

Outside peer selection, there is extensive work in the machine learning, information retrieval, and preference learning communities on the *learning to rank* problem: inferring the most likely ranking from possibly noisy observations [12]. These works include learning noise models, e.g., the parameters of a Mallows model, for use in inferring latent preferences of agents [12, 26]. This is of great practical interest in information retrieval, where one wishes to rank, e.g., web-pages based off user clicks [19] and in combining labellings from multiple sources for the construction of datasets [25]. However, all of these systems do not concern themselves with strategyproofness, a key focus of our study.

3 PRELIMINARIES

In our setup, a set of agents $N = \{1, 2, ..., n\}$ aim to select a subset of themselves of size k. We assume that, if agents were to assess each other accurately, they would report the same ranking. We refer to this ranking as the *ground truth*, a standard assumption in Condorcet views of voting [26]. In practical applications, it is often not feasible for agents to review all others, therefore we assume, for simplicity, that each agent reviews m agents and is reviewed by m of them. We represent such an m-regular graph via an assignment function $A : N \rightarrow 2^N$ and denote i's review pool, the agents reviewed by i, as $A(i)$. Let $A^{-1}(i)$ denote the set of agents that review agent i. In real-world settings, m is typically a small constant w.r.t. n.

Since we assume the ground truth to be a linear order, we also assume the belief of each reviewer over their review pool to be a strict ranking. Formally, we represent the underlying ranking of reviewer i as an injective function $\sigma_i : A(i) \rightarrow \{1, ..., m\}$, where $\sigma_i(j)$ is the rank given by reviewer i to agent j. The collection of these reviews is called a review *profile* and is denoted by σ. However, as in Mattei et al. [15], we take a less demanding approach and require reviewers to only report *approvals* to the mechanism, rather than a strict ranking. Formally, we instantiate each underlying belief σ_i by $\sigma_i^{\text{app}} : A(i) \rightarrow \{0, 1, \alpha\}$, where 1 represents approval and $\alpha \in [0, 1)$ is a constant representing "partial" approval. Similarly, $\sigma_i^{\text{app}}(j)$ is the approval score given by reviewer i to reviewer j.

PeerNomination. In PeerNomination, each agent, independently and concurrently, *nominates for certain* $\lfloor \frac{k}{n}m \rfloor$ of their reviewees, i.e., setting $\sigma_i^{\text{app}}(j) = 1$. In addition, they *probabilistically nominate* exactly one other agent, with probability $\frac{k}{n}m - \lfloor \frac{k}{n}m \rfloor$, i.e., setting $\sigma_i^{\text{app}}(j) = \frac{k}{n}m - \lfloor \frac{k}{n}m \rfloor$. In other words, probabilistic nominations become certain nominations with probability $\frac{k}{n}m - \lfloor \frac{k}{n}m \rfloor$. These partial nominations are used to maintain the size requirements as discussed by Mattei et al. [15]. An agent j is then selected iff they are nominated by the majority of their reviewers. Since each agent is considered independently for selection, the algorithm is not guaranteed to return exactly k agents, though it closely approximates k [15].

3.1 Noise Model

To model the inaccuracies in reviewers' assessments, we assume that each agent receives a noisy observation of the ground truth according to a Mallows model [14]. Mallows models have been widely used to compare the performance of peer selection algorithms empirically [3, 15], but so far only studied for very mild levels of noise, e.g., $\varphi = 0.5$.

The Mallows model is parameterised by a dispersion parameter $\varphi \in [0, 1]$ and a reference linear ranking R. Given R and φ, the model induces a probability distribution over all permutations of R such that the probability of the linear order R' is $\pi(R') \propto \varphi^{KT(R,R')}$, where $KT(R, R')$ is the Kendall-τ distance between R and R'. The Kendall-τ distance counts the number of pairwise disagreements between two rankings. In other words, the probability to find an additional pairwise disagreement from the reference ranking decreases exponentially. Note that, as we vary the dispersion parameter φ from 0 to 1, the probability distribution over all linear rankings moves from being concentrated on R to being uniform overall possible rankings.
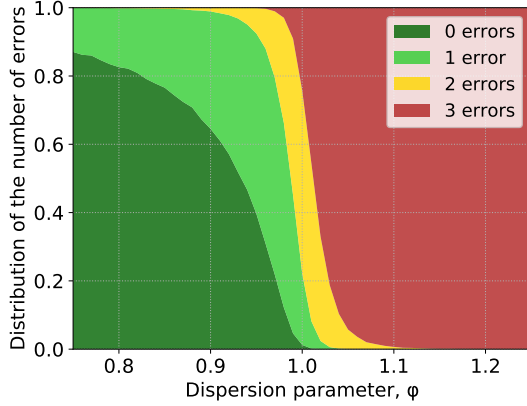
Figure 1: Nomination errors committed by agents as a function of φ. Each vertical slice shows the the distribution of the number of errors among the 200 agents in a simulation where each agent had to nominate 3 individuals out of 9.

In our simulations (Section 5), we take the ground truth as the reference ranking and sample a noisy ranking for each agent using the φ specified. An important feature of Mallows model is that it can be sampled efficiently [13, 26], which allows us to generate a unique reviewer profile for each experiment.

In addition, we test our weighting schemes in settings where some reviewers are not just random, but are actively adversarial. We thus extend the range of the dispersion parameter φ to $[0, 2]$, where $\varphi' \in (1, 2]$ means that the ranking is sampled using the *inverse* of the ground truth as the reference ranking and $\varphi = 2 - \varphi'$. Thus, the distribution moves smoothly from being concentrated at the ground truth to the inverse ground truth while still being uniform around 1.

It is important to note that the Mallows model does not produce errors in reviews proportionally to φ. Suppose that agents have to nominate top 3 out of 9 individuals and their beliefs are given by the Mallows model. With $\varphi = 0.5$ only a small number of agents are even going to commit 1 error and we need to increase φ to around 0.95 to have any meaningful probability of getting 2/3 nominations wrong. Only under our adversarial extension we start seeing agents that get all 3 nominations wrong. Figure 1 illustrates this relationship of errors and φ.

3.2 Properties

Our mechanism, like most, maintains the properties of *anonymity*, permuting agents makes no difference; *non-imposition*, any set of k accepted papers is a possible output; and *monotonicity*, if agent j was selected, and some agent i increased their score for it, agent j will still be selected.

We focus on *strategyproofness*, i.e., no agent is better off by reporting a non-truthful review. We do so by showing no agent's reviews have any influence over their own selection, and hence there is no incentive to misreport; and on ensuring the algorithm's quality. We measure the quality by two key parameters: *recall*, the

---

Algorithm 1 WEIGHTEDPEERNOMINATION CHOICE

Input: Assignment A, review profile σ, target quota k, slack parameter ε, reviewer weights $\{w_1, ..., w_n\}$
Output: Accepting set S
  Set $nomQuota := \frac{k}{n}m + \varepsilon$
  for all j in $\mathcal{N}$ **do**
    Initialise $nomCount := 0$
    for all $i \in A^{-1}(j)$ **do**
      if $\sigma_i(j) \leq \lfloor nomQuota \rfloor$ **then**
        increment $nomCount$ by w_i
      else if $\sigma_i(j) = \lfloor nomQuota \rfloor + 1$ **then**
        increment $nomCount$ by w_i with probability $nomQuota - \lfloor nomQuota \rfloor$
      end if
    end for
    if $nomCount \geq (\sum_{i \in A^{-1}(j)} w_i)/2$ **then**
      $S \leftarrow j$
    end if
  end for
  return S

---

proportion of true positives in the selected set[1]; and *size*, i.e., the number of selected agents.

4 WEIGHTED PEERNOMINATION

WEIGHTEDPEERNOMINATION extends PeerNomination [15] by adjusting a reviewer's weight as a function of the overall quality of their reviews. This design choice is based on the intuition that in many cases, agents' reviews are not independent when it comes to quality but, rather, correlated. PeerNomination is a special case of WEIGHTEDPEERNOMINATION with weights $(1, 1, \ldots, 1)$.

The new algorithm can be separated into three parts:

Assignment. Determining which agents review each agent (Algorithm 2).
Weights. Applying a reweighing protocol to adjust those reviews (Section 4.2).
Choice. Choosing, based on reweighed reviews, which agents to select (Algorithm 1).

As illustrated in Figure 2 (Left), the weighting scheme introduces the possibility of breaking strategyproofness: if agent a and agent b are both reviewing agent c, and agent a is also reviewing agent b, agent b may impact the weight given to agent a with their review to agent c, thus influencing agent a's role in determining if agent b himself is selected. Therefore, we complement the modified selection algorithm (Algorithm 1), with a method to assign agents to reviews that does not break strategyproofness, based on the Euler cycle (Figure 2 (Right)). This algorithm for assignment is presented as Algorithm 2. We separate the two algorithms as they do not depend on each other. Any assignment algorithm that does not create a case in which an agent can both review-with and be reviewed-by the same agent can run Algorithm 1 and maintain strategyproofness.

Weighting schemes may introduce various strategyproofness violations (as in Walsh [22]), so here we constrain them to be operators of the form $w_i : \mathbb{R}^m \times \mathbb{R}^{(m-1) \times m} \to \mathbb{R}$ for each agent i.

---

[1] Accuracy, which is often paired with recall, is less relevant in our setting as we have a fixed and often small proportion of positives (which is equal to k/n) in the population.
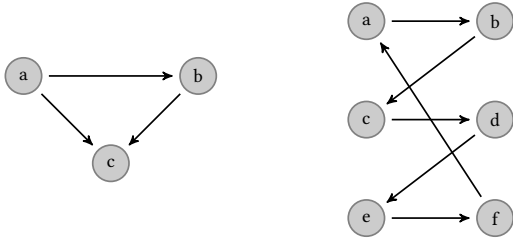
Figure 2: Algorithm 2 avoids non strategyproof assignments (Left), outputting strategyproof instances (Right).

These functions, which in our case will be identical for each agent, associate to each agent i's reviews the values of the other $m - 1$ reviews on those same reviewees. Under this condition we can state:

THEOREM 1. *WEIGHTEDPEERNOMINATION is strategyproof.*

PROOF. As was also shown for PeerNomination [15], for any exogenous set of weights, Algorithm 1, is strategyproof – no agent can influence their own chance of being selected. To show that it is not possible for an agent to influence the weights given to their own reviewers, observe that, for each agent i, w_i receives only the values of reviews on the same agents that agent i reviews. Thus, it is not possible for an agent j, who does not review a paper also reviewed by agent i, to change agent i's weight. Consider now an agent j that is reviewing a paper also reviewed by agent i. Agent j has a reason to deviate form truthful reporting if changing agent i's weight will benefit the chance that agent j is selected. Since the only use of agent i's weight in the outcome is on agents it reviews, agent j would benefit from altering agent i's weight only if agent i was reviewing agent j themselves.

Such an occurrence is prevented by Algorithm 2. Thanks to the initial creation of a bi-partite graph, agent i would review agent j only if $i \in X$ and $j \in Y$ (or vice versa). But then there can be no third agent both of them review: agent i only reviews agents in Y, while agent j only reviews agents in X (see Figure 2 (Right)). Thus, the only case where agent j would find it useful to not be truthful cannot happen. □

4.1 A Simple Theoretical Model

The constructive use of Algorithm 1 depends on the ability of identifying accurate and inaccurate reviewers, and using this identification to weigh their reviews. Clearly, lacking the knowledge of the ground truth means any identification of accurate/inaccurate agents has to depend on comparing agents to the reviews of the other agents. If all agents were accurate no reweighing of agents would be needed, but as the proportion of accurate agents drops the problem becomes more difficult. Still, when a large majority of them are accurate, the correct opinion is usually the majority, as inaccurate agents will give random rankings. However, if the number of accurate agents is very low, or other agents are actively malicious, identification becomes impossible, as finding a metric to evaluate the agents against becomes infeasible.

Algorithm 2 EULER-BASED ASSIGNMENT

Input: Set of n agents, review number $m \leq n/4$.
Output: Anti-transitive m-regular assignment A

Initialise $G = (V, E)$ with $V := [n]$, $E := \emptyset$
Partition V into X and Y such that $|X| = |Y| = n/2$
\\Make a $2m$-regular bipartite graph G(V,E)
for all x in X **do**
    for all i in $1, ..., 2m$ **do**
        $y^* \leftarrow \arg\min_y \{\deg(y) \mid y \in Y\}$
        $E \leftarrow E \cup \{\{x, y^*\}\}$
    end for
end for
\\Use Hierholzer's algorithm to find an Euler cycle
\\Euler cycle exists since every node has even degree
Set euler_cycle = *Hierholzers_algorithm*(G)
Set $A := ([n], E_A)$, $E_A := \emptyset$
for all (v_i, v_{i+1}) in euler_cycle **do**
    $E_A \leftarrow E_A \cup \{(v_i, v_{i+1})\}$
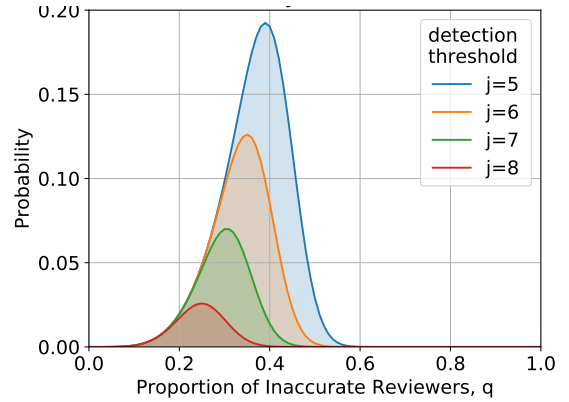**end for**
return A

Figure 3: The probability of identifying an inaccurate agent, when $m = 9$, and the threshold for identification is j.

To provide some intuition to the conceptual underpinnings of our algorithm we now present a simplified setting, and show how our algorithm – even with a very simple, conservative, weighting scheme – is still able to improve over PeerNomination. We start with an m-regular assignment, where each agent has one of two types: $\mathcal{A}$, meaning the agent is an accurate reviewer; or $\overline{\mathcal{A}}$, meaning the agent is inaccurate. Recall that in PeerNomination an agent is selected if a majority of their reviewers approve. We show that a very simple dynamic weighting scheme, only relying on knowing how many times an agent has been in a minority, has a good chance of flipping a decision made by $\overline{\mathcal{A}}$ agents to one made by $\mathcal{A}$ agents, improving on PeerNomination.

Let an $\overline{\mathcal{A}}$-agent be *identified as inaccurate* if they held the minority opinion in at least j reviews. We want to find the probability of the following event: (1) An agent's reviewers have an $\overline{\mathcal{A}}$-majority and (2) Enough of the $\overline{\mathcal{A}}$-agents of that majority are identified as inaccurate.
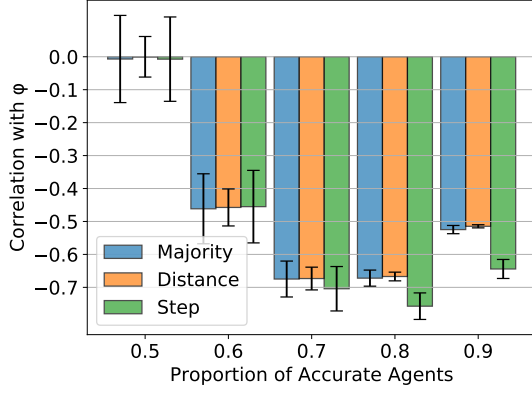
(a) Agents produced from Mallows distribution with $\varphi = 0.5$ for the share shown, the rest with $\varphi = 1$.
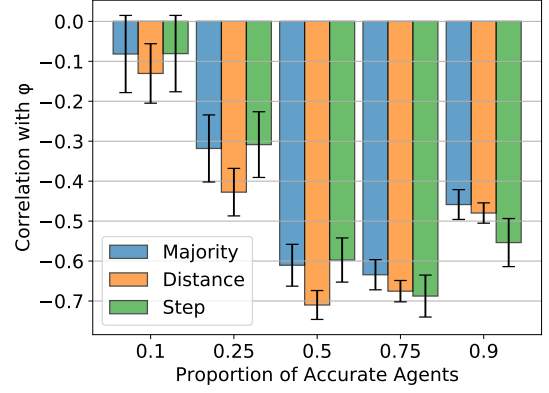
(b) Agents produced from Mallows distribution with $\varphi = 0.8$ for the share shown, the rest with $\varphi = 1.2$.

Figure 4: Spearman correlation of the weights from the different weighting schemes with the underlying φ of each agent. The bars represent the mean and the standard deviation over 1000 simulations of the weights.

Given the noise model, let q be the probability for an agent to be of type $\overline{\mathcal{A}}$. Then the probability that an agent is reviewed by a majority of $\overline{\mathcal{A}}$ agents and that majority is of size k is:

$$q_{\overline{\mathcal{A}},k} := \mathbb{P}[\overline{\mathcal{A}}\text{-majority of size } k] = \binom{m}{i} q^i (1-q)^{m-i}$$

where $i = \lfloor m/2 \rfloor + k$. In such a case we would like to identify at least k of the $\overline{\mathcal{A}}$ agents in order to nullify their votes.

We also find the probability that $\overline{\mathcal{A}}$-agents have a majority of any size:

$$q_{\overline{\mathcal{A}}} := \mathbb{P}[\overline{\mathcal{A}}\text{-majority}] = \sum_{k=1}^{\lceil m/2 \rceil} q_{\overline{\mathcal{A}},k} = \sum_{i=0}^{\lfloor m/2 \rfloor} \binom{m}{i}(1-q)^i q^{m-i}$$

Our simple weighing algorithm identifies a $\overline{\mathcal{A}}$-agent if they are in minority for at least j of their other reviewed papers. The probability of this event, q_{det}, is given by the cumulative binomial probability, keeping in mind that the probability of $\mathcal{A}$-majority in a set of reviews on an agent is conditioned on the fact that they contain at least one $\overline{\mathcal{A}}$-agent:

$$q_{\mathcal{A}} := \mathbb{P}[\mathcal{A}\text{-majority} \mid \text{there is at least one } \overline{\mathcal{A}}]$$

$$= \sum_{i=0}^{\lfloor m/2 \rfloor} \binom{m-1}{i}(1-q)^i q^{m-1-i}$$

$$\Rightarrow q_{\text{det}} = \sum_{i=j}^{m-1} \binom{m-1}{i}(q'_{\mathcal{A}})^i (1-q'_{\mathcal{A}})^{m-1-i}$$

Notice that $\overline{\mathcal{A}}$ may be not just inaccurate but adversarial, in which case we could flip their review and only need to do so for k of them. However, we take the safer approach here, which means we need to detect at least $2k$ $\overline{\mathcal{A}}$-agents to correct the decision. The probability of this correcting event is given by the following expression:

$$\mathbb{P}[\text{correction event}] =$$

$$\sum_{k=1}^{\lceil m/2 \rceil} q_{\overline{\mathcal{A}},k} \cdot \left(\sum_{i=2k}^{\lfloor m/2 \rfloor + k} \binom{\lfloor m/2 \rfloor + k}{i} q_{\text{det}}^i (1-q_{\text{det}})^{\lfloor m/2 \rfloor + k - i} \right)$$

This produces the desired result, empirically shown in Figure 3. As can be seen, for a wide variety of q and j, even our very conservative weighting scheme produces a nice probability of improving some reviews. As we shall see with the weighting schemes below, examined by our simulations, even better results can be achieved.

It should be noted that one could produce analogous probabilities of the weighting scheme incorrectly identifying $\mathcal{A}$-agents as $\overline{\mathcal{A}}$-agents. However, for a large enough j (say, $j \geq m/2$), and a majority of $\mathcal{A}$ agents (i.e., $q < n/2$), this number will always be smaller, i.e., the benefit from the reweighing will be positive.

4.2 Weighting Schemes

We present three main weighting schemes to evaluate the reliability of the reviewers, each based on the approval vote supplied to the algorithm, maintaining strategyproofness.

Distance. Distance directly computes the distance between the agent's review and those of other reviewers by simply averaging the individual differences between the reviewers. Formally, define the average distance of reviewer i to other reviewers as $d_i = \frac{1}{m^2} \sum_{j \in A(i)} \sum_{l \in A^{-1}(j)} |\sigma_i^{\text{app}}(j) - \sigma_l^{\text{app}}(j)|$. Then the distance weight is $w_i^{\text{dist}} = (1 - d_i)^\gamma$, where γ is an "aggression" parameter that exaggerates for better discrimination between the agents.

Majority Errors. Let an approval of a reviewee be an "error" by a reviewer if it is a minority opinion, summing fractional nominations. Let

$$\text{maj}_j^\sigma = \begin{cases} 1, & \text{if } \sum_{i \in A^{-1}(j)} \sigma_i^{\text{app}}(j) \geq m/2 \\ 0, & \text{otherwise} \end{cases}$$

be the majority for agent j in profile σ. Then define the number of errors of reviewer i to be $\text{err}_i^\sigma = \sum_{j \in A(i)} \mathbb{1}_{\sigma_i^{\text{app}}(j) \neq \text{maj}_j^\sigma}$. Lastly, the weight is defined as $w_i^{\text{majerr}} = 1 - \delta(\text{err}_i^\sigma / m)$, where δ is an aggression parameter.

Step. Step applies a step function to the error rate err_i^σ / m defined above. We choose two thresholds, t_1 and t_2 such that if the error rate reaches t_1, we reduce the weight of the reviewer to 0.5; if the error

(a) Population's $\varphi \in \{0.5, 1\}$.

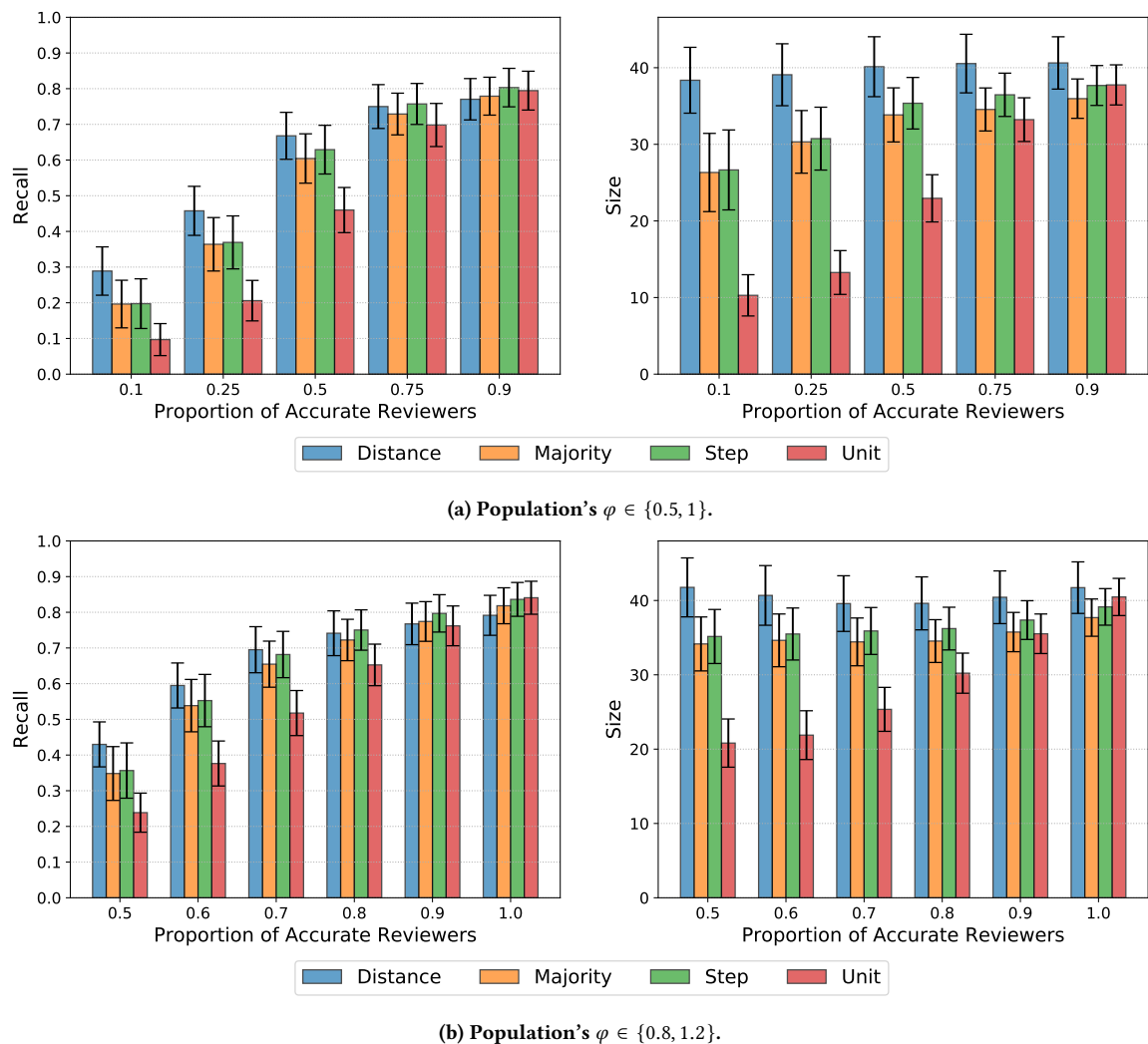(b) Population's $\varphi \in \{0.8, 1.2\}$.

Figure 5: Size and recall with $n = 200, m = 7, k = 40$ with noisy reviewers (a) and adversarial reviewers (b).

rate reaches t_2, we reduce the weight to 0. Additionally, we scale each threshold by the nomination quota as it plays a bigger role in error detection than just the size of the review pool, m. Formally,

$$w_i = \begin{cases} 1, & \text{if } \text{err}_i^\sigma / \frac{n}{k} < t_1 \\ 0.5, & \text{if } t_1 \le \text{err}_i^\sigma / \frac{n}{k} < t_2 \\ 0, & \text{otherwise.} \end{cases}$$

5 SETUP AND PERFORMANCE RESULTS

We test WEIGHTEDPEERNOMINATION, using our weighting schemes, against PeerNomination, following the testing framework used in Mattei et al. [15] and Aziz et al. [3]. However, our setup differs in a few significant ways. In contrast to the previous experiments, we do not include any partition-based mechanisms, thus allowing us to use the assignment-generating procedure given in Algorithm 2 without worrying about forming a given number of clusters (the procedure does generate a 2-clustering).

To simulate the presence of noise, we use Mallows model, as described in Section 3.1. We assume that the population consists of accurate and inaccurate reviewers, which we represent by using two distinct values of the dispersion parameter φ. In Section 5.1 we assume that the accurate reviewers have $\varphi = 0.5$ and the inaccurate reviewers have $\varphi = 1$, i.e., their reviews are random. We vary the proportion of accurate reviewers between 0.1 and 0.9. Also recall that in Section 3.1 we extended the definition of the dispersion parameter to range between 0 and 2. In Section 5.2, we set $\varphi = 0.8$ for the accurate reviewers and $\varphi = 1.2$ for the adversarial reviewers, i.e., most of their reviews will contradict the ground truth. We vary the proportion of accurate reviewers between 0.5 and 1. Note that as the proportion of accurate reviewers drops to 0.5 and lower, it becomes impossible to recover the ground truth.

Figure 4 demonstrates that our metrics strongly correlate with the underlying φ in both setups, demonstrating their effectiveness in singling out accurate reviewers.

5.1 Inaccurate Reviewers

Figure 5a shows that a high (90%) proportion of accurate reviewers results in barely any improvement over Unit (original PeerNomination). This indicates that the weighting schemes do not overfit in the search for non-existing noise. However, as the proportion of random reviewers rises, all weighting schemes outperform Unit.

We also see that the weighting schemes are indeed much better at keeping the output size close to the desired k with Distance keeping the output size consistent across all levels of noise. In addition, the much greater recall of Distance and other weighting schemes indicates the additional selected agents are usually the deserving ones. Between the weighting schemes, Distance manages to gain more and more advantage as the noise levels increase as it is the most fine-grained and aggressive one. This allows it to both identify the inaccurate reviewers and maintain consistent output size.

There are similar patterns for other settings of parameters. As k increases, the performance of all algorithms improves as not only can they be less selective, but the reviewers provide more data. The advantage over Unit, however, decreases as PeerNomination is capable of taking advantage of this as well. Increasing m, on the other hand, primarily benefits the weighting schemes as they can calculate the weights more accurately, increasing their advantage over Unit.

5.2 Adversarial Reviewers

We see a similar story with the adversarial model. Again, at low noise levels, PeerNomination, as expected, performs well while the weighting schemes' slight drop in performance can be attributed to overfitting. However, as the proportion of adversarial reviewers rises, we can observe a notable increase in performance for the weighting schemes. Even when half of the population is adversarial, Distance impressively achieves recall of over 40% as compared to the theoretical max of 50%.

6 DISCUSSION

In this paper we propose a novel strategyproof peer selection algorithm – WEIGHTEDPEERNOMINATION, which weighs reviewers based on their perceived accuracy. The basis for this reweighing is the observation that in most cases, one's reviews are correlated in quality. We develop several weighting methods, showing that even straightforward ones can reach high quality outcomes, with high level of noise. Our algorithm is constructed in a modular way, allowing for a variety of weighting and evaluation methods to be developed for particular settings or noise models. This modularity allows for multiple directions of future development, including developing additional weighting; matching weighting schemes to noise models; and optimising them using various techniques.

REFERENCES

[1] Noga Alon, Felix Fischer, Ariel Procaccia, and Moshe Tennenholtz. 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.

[2] Haris Aziz, Omer Lev, Nicholas Mattei, Jeffrey S. Rosenschein, and Toby Walsh. 2016. Strategyproof Peer Selection: Mechanisms, Analyses, and Experiments.. In *AAAI*, Dale Schuurmans and Michael P. Wellman (Eds.). AAAI Press, 397–403.

[3] Haris Aziz, Omer Lev, Nicholas Mattei, Jeffrey S. Rosenschein, and Toby Walsh. 2019. Strategyproof peer selection using randomization, partitioning, and apportionment. *Artificial Intelligence* 275 (2019), 295–309. https://doi.org/10.1016/j.artint.2019.06.004

[4] N. Bousquet, S. Norin, and A. Vetta. 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))*. 133–146.

[5] Ioannis Caragiannis, George A. Krimpas, and Alexandros A. Voudouris. 2015. Aggregating Partial Rankings with Applications to Peer Grading in Massive Online Open Courses. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS*. ACM, 675–683.

[6] Luca de Alfaro and Michael Shavlovsky. 2014. CrowdGrader: a tool for crowdsourcing the evaluation of homework assignments. In *The 45th ACM Technical Symposium on Computer Science Education, SIGCSE*. ACM, 415–420.

[7] Geoffroy de Clippel, Hervé Moulin, and Nicolaus Tideman. 2008. Impartial division of a dollar. *Journal of Economic Theory* 139 (2008), 176–191.

[8] Felix Fischer and Max Klimm. 2014. Optimal Impartial Selection. In *Proceedings of the 15th ACM Conference on Economics and Computation (ACM-EC)*. 803–820.

[9] Anson Kahng, Yasmine Kotturi, Chinmay Kulkarni, David Kurokawa, and Ariel Procaccia. 2018. Ranking wily people who rank each other. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*.

[10] David Kurokawa, Omer Lev, Jamie Morgenstern, and Ariel D. Procaccia. 2015. Impartial Peer Review. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15)*. AAAI Press, 582–588. http://dl.acm.org/citation.cfm?id=2832249.2832330

[11] Jing Wu Lian, Nicholas Mattei, Renee Noble, and Toby Walsh. 2018. The Conference Paper Assignment Problem: Using Order Weighted Averages to Assign Indivisible Goods. In *Proceedings of the32ndAAAI Conference on Artificial Intelligence (AAAI)*. 1138–1145.

[12] Tie-Yan Liu. 2011. *Learning to rank for information retrieval*. Springer Science & Business Media.

[13] Tyler Lu and Craig Boutilier. 2011. Learning Mallows models with pairwise preferences. In *Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML)*. 145–152.

[14] Colin Lingwood Mallows. 1957. Non-null ranking models. I. *Biometrika* 44, 1-2 (June 1957), 114–130.

[15] Nicholas Mattei, Paolo Turrini, and Stanislav Zhydkov. 2020. PeerNomination: Relaxing Exactness for Increased Accuracy in Peer Selection. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*. –.

[16] R. Meir, J. Lang, J. Lesca, N. Kaminski, and N. Mattei. 2021. A Market-Inspired Bidding Scheme for Peer Review Paper Assignment. In *Proceedings of the35thAAAI Conference on Artificial Intelligence (AAAI)*.

[17] Michael Merrifield and Donald Saari. 2009. Telescope time without tears: a distributed approach to peer review. *Astronomy and Geophysics* 50, 4 (2009), 4.16–4.20. https://doi.org/10.1111/j.1468-4004.2009.50416.x

[18] Chris Piech, Jonathan Huang, Zhenghao Chen, Chuong B. Do, Andrew Y. Ng, and Daphne Koller. 2013. Tuned Models of Peer Assessment in MOOCs. In *Proceedings of the 6th International Conference on Educational Data Mining (EDM)*. 153–160.

[19] Tobias Schnabel, Adith Swaminathan, Peter I Frazier, and Thorsten Joachims. 2016. Unbiased comparative evaluation of ranking functions. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*. 109–118.

[20] Nihar B Shah, Behzad Tabibian, Krikamol Muandet, Isabelle Guyon, and Ulrike Von Luxburg. 2018. Design and analysis of the NIPS 2016 review process. *The Journal of Machine Learning Research* 19, 1 (2018), 1913–1946.

[21] Ivan Stelmakh, Nihar B Shah, and Aarti Singh. 2020. Catch me if I can: Detecting strategic behaviour in peer assessment. *arXiv preprint arXiv:2010.04041* (2020).

[22] Toby Walsh. 2014. The PeerRank Method for Peer Assessment. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*. Prague, Czech Republic, 909–914.

[23] Jingyan Wang and Nihar B Shah. 2019. Your 2 is my 1, your 3 is my 9: Handling arbitrary miscalibrations in ratings. In *Proceedings of the18thInternational Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*. 864–872.

[24] Jingyan Wang, Ivan Stelmakh, Yuting Wei, and Nihar B Shah. 2020. Debiasing Evaluations That are Biased by Evaluations. *arXiv preprint arXiv:2012.00714* (2020).

[25] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. In *Advances in Neural Information Processing Systems*. 2035–2043.

[26] Lirong Xia. 2019. *Learning and Decision-Making from Rank Data*. Morgan and Claypool.

[27] Yichong Xu, Han Zhao, Xiaofei Shi, and Nihar B. Shah. 2019. On Strategyproof Conference Peer Review. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. Macau, 616–622.