A SUPER* Algorithm to Optimize Paper Bidding in Peer Review

Tanner Fiez
University of Washington
fiezt@uw.edu

Nihar B. Shah
Carnegie Mellon University
nihars@cs.cmu.edu

Lillian Ratliff
University of Washington
ratliffl@uw.edu

Abstract

A number of applications involve sequential arrival of users, and require showing each user an ordering of items. A prime example is the bidding process in conference peer review where reviewers enter the system sequentially, each reviewer needs to be shown the list of submitted papers, and the reviewer then "bids" to review some papers. The order of the papers shown has a significant impact on the bids due to primacy effects. In deciding on the ordering of the list of papers to show, there are two competing goals: (i) obtaining sufficiently many bids for each paper, and (ii) satisfying reviewers by showing them relevant items. In this paper, we develop a framework to study this problem in a principled manner. We present an algorithm called SUPER*, inspired by the A* algorithm, for this goal. Theoretically, we show a local optimality guarantee of our algorithm and prove that popular baselines are considerably suboptimal. Moreover, under a community model for the similarities, we prove that SUPER* is near-optimal whereas the popular baselines are considerably suboptimal. In experiments on real data from ICLR 2018 and synthetic data, we find that SUPER* considerably outperforms baselines deployed in existing systems, consistently reducing the number of papers with fewer than requisite bids by 50-75% or more, and is also robust to various real world complexities.

1 INTRODUCTION

It is well-known that peer review is an essential process for ensuring the quality and scientific value of research (Black et al., 1998; Thurner and Hanel, 2011). A fundamental challenge in peer review is matching or assigning papers to qualified and willing reviewers. Bidding has emerged as an important mechanism for aiding in and improving the peer review process under the guise that active engagement of the reviewer leads to assignments more aligned with their preferences

In typical peer review process, when the bidding process opens, reviewers enter the system in an arbitrary sequential order. Upon entering, a list of papers is shown to them and they are asked to place bids on papers they would prefer to review. Following the bidding process, bids can be incorporated into the reviewer-paper assignment mechanism. It is known that the order of papers presented to reviewers in the bidding stage can greatly impact the number of bids that a paper receives (Cabanac and Preuss, 2013). From the perspective of the platform, there are two competing goals: (i) ensure that each paper has a sufficient number of bids, and (ii) ensure individual reviewer satisfaction by showing relevant papers.

With regard to goal (i), the platform aims to select a display order for each reviewer such that at the end of the bidding process, each paper has at least a certain number of bids. The main objective of ensuring a minimum number of bids on each paper is to improve review quality for all papers (Shah et al., 2018b). The well-documented primacy effect (Murphy et al., 2006) suggests that papers shown on top of the ordering are the ones on which reviewers are more likely to bid. Consequently, this objective suggests that papers with few bids should be placed higher in the list (Cabanac and Preuss, 2013).

With regard to goal (ii), the platform aims to display 'well-matched' papers to each reviewer. That is, the set of papers to be displayed is composed of papers on which the reviewer is most likely to bid. It is generally assumed that reviewers are more likely to place bids on papers they are qualified to review (Rodriguez et al., 2007). Furthermore, reviewers that place positive bids

Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR volume 124, 2020.

on papers are more likely to give a review with high confidence and voice sharp opinions of acceptance or rejection that help guide final decisions on papers (Cabanac and Preuss, 2013). Failing to display relevant papers to reviewers can result in several unintended negative consequences. If irrelevant papers are shown early in the order to a reviewer, it may cause the reviewer to opt-out and disengage with the system even if further down the list there was an option that they would have happily bid on. Similarly, a poorly selected ordering may result in significantly fewer bids from a reviewer.

In peer review, it is recognized that actively engaging reviewers in the paper assignment process via bidding can greatly improve the review process. If administered inadequately, bidding can in fact have a significant negative impact on the quality of the review process (Rodriguez et al., 2007). A study on the 2016 Neural Information Processing Systems (NeurIPS) conference revealed the distribution of bids arising from a typical bidding process leaves significant challenges to match papers with reviewers (Shah et al., 2018b). It was observed that a considerable number of reviewers do not place a sufficient number of bids and papers commonly fail to obtain as many bids as the number of reviewers needed. This phenomenon is detailed in Shah et al. (2018b) among the 3,200 reviewers and 2,400 papers. The inability to elicit meaningful bidding information in NeurIPS is far from an aberration. In a study of the 2005 Joint Conference on Digital Libraries, 146 out of the 264 submissions did not obtain any bids (Rodriguez et al., 2007).

Despite the importance of the bidding process in peer review, there is not yet much fundamental research on the problem of optimizing the display order during the bidding process, and much less so in consideration of the two objectives identified in this paper. In practice, the display order is typically determined via heuristics such as a fixed ordering (e.g., order of submission), or in decreasing order of the relevance of the papers to that reviewer, or in increasing order of the number of bids received by the paper until then.

A key reason that bidding can fail is that papers are suboptimally displayed to the reviewers. Consider a paper that is not an ideal match for any reviewer in the system. If papers are ranked for display simply by how well-matched they are to reviewers, this particular paper may be shown far down in the list to each reviewer and hence, not receive many, if any, bids. The risk of this scenario is elevated for interdisciplinary research, which is know to face significant impediments as a consequence of the lack of ideally matched peers (Porter and Rossini, 1985).

On the other hand, if papers are inversely ranked by the number of bids they have obtained, then papers with fewer bids are more likely to be shown higher on the list regardless of how well-matched they are to a reviewer. This display order may cause reviewer dissatisfaction, which in the worst case could result in zero bids. Similarly, ordering heuristics that are based on a fixed baseline may lead to bias in the review process. In a study of 42 peer-reviewed Computer Science conferences, it was observed that under a fixed ordering (based on the submission time), the number of bids on papers is heavily influenced by the order of paper submission times (Cabanac and Preuss, 2013). It was concluded that the later the paper is submitted, the fewer bids it will receive.

Given the flaws of existing peer review bidding systems, we study the important problem of selecting the ordering of papers to display to each arriving reviewer.

1.1 CONTRIBUTIONS

The key contributions of this paper are now summarized.

Problem identification and formulation (Section 2). The bidding process is highly consequential, yet one of the most understudied components of the conference peer-review process. We identify a key source of unfairness and inefficiency in the bidding process, and develop principled methods to address it. A key challenge is suitably formalizing this problem in the peer review bidding process, for which to the best of our knowledge there are no prior formulations. We formulate an objective function that captures the competing goals of the platform while reflecting the underlying decision-making process of reviewers. The general framework developed in this paper to analyze the problem is an important step toward future improvements on bidding systems.

Algorithm design (Section 3). We present a sequential decision-making algorithm called $\texttt{SUPER}^*$ to address this problem. The algorithm takes as input the "similarities" between each reviewer-paper pair and the bids made by all past reviewers, and outputs the ordering of papers to show to any current reviewer.

Theoretical results (Section 4). We show two sets of theoretical results. We first consider a notion of 'local' performance: the performance with respect to a single reviewer. We prove that $\texttt{SUPER}^*$ is locally optimal whereas all popular baselines are considerably suboptimal. Our second set of theoretical results are based on a community model, where we theoretically show that $\texttt{SUPER}^*$ is near-optimal (globally) whereas all popular baselines are considerably suboptimal.

Experiments on real and synthetic data (Section 5). We run extensive experiments using similarity scores from ICLR 2018 and on synthetic data. The experi-

ments reveal that the SUPER* algorithm outperforms all popular baselines. For instance, it consistently reduces the number of papers with fewer than requisite bids by 50-75% while maintaining individual reviewer satisfaction. In addition, we observe that SUPER* is robust to model mismatches and complexities of the real-world review process. The code for the algorithm is available at github.com/fiezt/Peer-Review-Bidding.

1.2 RELATED WORK

The paper ordering problem for the bidding process in peer review bears a strong resemblance to the learning to rank problem (Singh and Joachims, 2019; Yadav et al., 2019; Aslanyan and Porwal, 2019). The objective of finding a ranking most suitable for an arriving reviewer during the bidding process is analogous to learning to rank methods that consider the utility of rankings for users along with the impact on the items being ranked (Singh and Joachims, 2019; Yadav et al., 2019). Moreover, the bidding model considered in this work is motivated from that which is commonly adopted in learning to rank models (Aslanyan and Porwal, 2019). The problem is also abstractly similar to online recommendation systems similarly facing competing objectives (Rodriguez et al., 2012; Agarwal et al., 2011; Jambor and Wang, 2010), where often the multi-objective problem is converted to a constrained optimization problem.

The objective in the peer review problem as formulated in this paper presents unique challenges not addressed in the aforementioned works on learning to rank and recommendation systems. Notably, it is not separable between the reviewers since it depends nonlinearly on the number of bids on each paper after each reviewer has arrived and placed bids on the papers. It is worth mentioning that in bandits with knapsacks (Badanidiyuru et al., 2018), the observation model can include both immediate user reward feedback and itemwise feedback that couples decisions such as in our model. However, there is not a way to encode the goal of ensuring a minimum amount of consumption on each of the items in the constraints, so our work cannot fit into the aforementioned framework.

Our work also contributes to a growing literature on improving the peer review process including reviewer assignment (Charlin and Zemel, 2013; Stelmakh et al., 2019b; Kobren et al., 2019), biases (Tomkins et al., 2017; Stelmakh et al., 2019a), subjectivity (Noothigattu et al., 2018), miscalibration (Wang and Shah, 2019), strategic behavior (Xu et al., 2019), among others (Lawrence and Cortes, 2014; Shah et al., 2018a; Jecmen et al., 2020; Ding et al., 2020; Stelmakh et al., 2020a,b). This paper addresses the bidding process in conference peer review, which has largely been unexplored in past literature.

2 PROBLEM FORMULATION

Consider $d \geq 2$ papers and $n \geq 2$ reviewers indexed as $\{1, \ldots, d\}$ and $\{1, \ldots, n\}$ respectively.[1] For each reviewer-paper pair, we have access to a *similarity score* that captures the similarity between the reviewer and the paper. We use $S_{i,j} \in [0, 1]$ to denote the given similarity between any reviewer $i \in [n]$ and paper $j \in [d]$. A higher similarity score indicates a greater relevance of the paper to that reviewer. There are several systems in use today that compute similarities (Charlin and Zemel, 2013), and in our work, we treat them as being given.

In the bidding period, reviewers sequentially arrive into the system and place bids on the papers. In our work, for any reviewer and paper, we only consider the existence of a bid or not, and do not consider the possibility of multiple bidding options. Moreover, we assume for simplicity that all n reviewers arrive exactly once, and that a reviewer arrives after the previous reviewer has completed their bidding. The problem is to determine the ordering of papers to show each reviewer on arrival in the interest of influencing the papers they decide to bid on while ensuring individual satisfaction. When deciding the paper ordering for any reviewer, the bids made by all reviewers who arrived in the past along with the paper orderings presented to them are known, but the bids made by the current or future reviewers are unknown. Let Π_d denote the set of all possible $d!$ permutations of the d papers. In what follows, for any reviewer $i \in [n]$, we let $\pi_i \in \Pi_d$ denote the ordering (permutation) of the papers shown to reviewer i. We also use the notation $\pi_i(j)$ to denote the position of paper $j \in [d]$ in the ordering π_i.

Gain function (objective). Any algorithm to determine the ordering of papers must trade-off between two competing objectives: ensuring each paper receives a sufficient number of bids and ensuring each reviewer gets to see relevant papers early in the ordering. A combination of the objectives comprise our "gain function," which is the objective we aim to optimize. We begin by discussing each objective component separately.

Paper-side gain: The paper-side gain is associated with a given function $\gamma_p : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$. At the end of the entire bidding process, the paper-side gain $\mathcal{G}_p$ is

$$\mathcal{G}_p = \sum\nolimits_{j \in [d]} \gamma_p(g_j),$$

where g_j is the number of bids received by paper $j \in [d]$. We assume the function γ_p is non-decreasing and concave. The non-decreasing property represents an improved gain if there are more bids, and the concavity

[1] Henceforth, for any positive integer κ, we will use the standard shorthand $[\kappa]$ to denote the set $\{1, \ldots, \kappa\}$.

property captures diminishing returns.[2] An example of a choice for the paper-side gain is the square-root function $\gamma_p(x) = \sqrt{x}$. This function is increasing, smooth, and captures the diminishing returns property. The reader may keep this function in mind as a running example for concreteness. A second example is $\gamma_p(x) = \min\{x, r\}$ for a given parameter $r \geq 1$, which emphasizes having at least r bids per paper.

Reviewer-side gain: This objective captures the desideratum that the reviewers should be shown papers with high relevance early in the paper ordering. The reviewer-side gain is associated with some predetermined function $\gamma_r : [d] \times [0, 1] \to \mathbb{R}_{\geq 0}$. Given this function, the reviewer-side gain $\mathcal{G}_r$ is defined as:

$$\mathcal{G}_r = \sum\nolimits_{i \in [n]} \sum\nolimits_{j \in [d]} \gamma_r(\pi_i(j), S_{i,j}).$$

The function γ_r is assumed to be non-increasing in the position (its first argument) and non-decreasing in the similarity (its second argument). One example choice of this function, which the reader may choose to keep in mind as a running example, is the Discounted Cumulative Gain or DCG used commonly in data mining (Järvelin and Kekäläinen, 2000). After taking the "relevance" parameter in DCG to be the similarity $S_{i,j}$, the function we consider is given by

$$\gamma_r(\pi_i(j), S_{i,j}) = (2^{S_{i,j}} - 1)/(\log_2(\pi_i(j) + 1)). \quad (1)$$

Overall gain function: Finally, we assume there is a trade-off parameter $\lambda \geq 0$, chosen by the program chairs, which trades off between these two objectives so that the overall gain function is given by

$$\mathcal{G} = \mathcal{G}_p + \lambda \mathcal{G}_r. \quad (2)$$

The goal is to determine the orderings of papers to show each reviewer to maximize the expected overall gain, $\mathbb{E}[\mathcal{G}]$, where the expectation is taken over the randomness in the bids made by the reviewers (see reviewer bidding model below) and any randomness in the algorithm.

Reviewer bidding model. An important aspect of any system that displays a list to users is the presence of primacy effects. In the context of our problem, the primacy effect means a reviewer is more likely to bid on a paper shown at the top of the list rather than later (Murphy et al., 2006). A second aspect of bidding is that a reviewer is more likely to bid on papers with greater similarity, although the reviewer may not bid on exactly the papers with the highest similarity since the similarities are noisy representations of their reviewing interests.

Thus in order to model reviewer bidding, we revert to literature on position-based click models that have a nearly identical setting (where clicks are analogous to our bids). We model the bidding via a given function $f : [d] \times [0, 1] \to [0, 1]$, where $f(\pi_i(j), S_{i,j})$ is non-increasing in the position that a paper is shown (the first argument) and non-decreasing in the similarity score (the second argument). Any reviewer $i \in [n]$ bids on paper $j \in [d]$ independently with probability $p_{i,j} = f(\pi_i(j), S_{i,j})$. As a running example throughout the paper, note that in position-based click models, the click probability decomposes into a product of relevance and position bias (Chuklin et al., 2015). Moreover, the literature considers the click probability to decay logarithmically as a function of the position (Aslanyan and Porwal, 2019). The translation of these models into our setting gives rise to the example bidding function

$$f(\pi_i(j), S_{i,j}) = S_{i,j}/\log_2(\pi_i(j) + 1). \quad (3)$$

Baselines. We consider the following three methods of ordering papers as baselines.

Random baseline (`RAND`): A commonplace practice (Cabanac and Preuss, 2013) is to show papers to reviewers in some fixed order, such as in order of submission of the papers. As a baseline, we consider a better variant of this practice, in which each reviewer is shown an independently and randomly selected paper ordering.

Similarity baseline (`SIM`): A second common practice, followed in several conference management systems today, is to order the papers according to their similarities. In other words, any reviewer $i \in [n]$ is shown the papers in order of the values in $\{S_{i,j}\}_{j \in [d]}$ (where the paper with maximum similarity is shown at the top, and so on). Any ties are broken by showing papers with fewer bids higher, and further ties are broken uniformly at random.

Bid baseline (`BID`): A third baseline shows papers to greedily optimize the minimum bid count. Each reviewer is shown papers in increasing order of the number of bids received so far from reviewers who arrived previously. Any ties are broken in favor of the paper with a higher similarity, and then uniformly at random.

3 ALGORITHM

The key challenge in designing a suitable algorithm for the problem at hand stems from the fact that the paper-side gain is coupled (non-separable) across the orderings of papers presented to all reviewers so the impact of each individual paper ordering cannot be fully realized un-

[2] Our algorithm easily adapts to paper-side gains that may also be a function of the similarity scores of the reviewers who bid; for example, a higher gain for bids from expert reviewers. We omit this detail for sake of brevity.

til the entire bidding process is complete. Conversely, the reviewer-side gain is decoupled (separable) across reviewers. This means the reviewer-side gain that can be obtained from any given reviewer is independent of the ordering of papers presented to any other reviewer. Thus, an algorithm for this problem is required to make local decisions, where the effect of the decision on the global gain (or cost) is only partially known. This perspective is reminiscent of the classical A* algorithm (Hart et al., 1968), and using A* as an inspiration, we now present an algorithm which we call SUPER* for our problem[3].

The A* algorithm operates with a goal of finding the minimum cost path between a pair of vertices in a cost-weighted graph. For any node in consideration, it considers two functions: a function which captures the cost so far and a second function—called the "heuristic"—which captures some estimate of the cost from the current node to the destination. The A* algorithm then finds a path based on these two functions. Before moving to a description of SUPER*, we discuss such a heuristic in the context of the problem at hand.

3.1 HEURISTIC FOR FUTURE BIDS

In a manner analogous to the A* algorithm, at any point in time SUPER* keeps track of the gains so far and also takes as input a heuristic that captures the "unseen" events. The heuristic in A* provides, for every vertex in the given graph, an estimate of the cost incurred in the future. Analogously, the heuristic in SUPER* provides, for every arrival of a reviewer, an estimate of the number of bids each paper will receive in the future. Formally, let us index the reviewers as $i \in [n]$ in the order of arrival (note that this order is unknown a priori). The heuristic comprises a collection of vectors $\{h_1, \ldots, h_n\}$, where each $h_i \in [0, n-i]^d$ represents an estimate of the number of bids each of the d papers will receive from all future reviewers $\{i+1, \ldots, n\}$. The vector h_i is provided to the SUPER* algorithm on arrival of the i^{th} reviewer. Two examples of heuristic functions that we consider in the subsequent narrative are described as follows.

- *Zero heuristic:* $h_i = 0$ for every $i \in [n]$.
- *Mean heuristic:* This function computes the expected number of bids each paper will receive if the permutations shown to all future reviewers are chosen independently and uniformly at random. Formally: $h_{i,j} = \frac{1}{d}\sum_{i'=i+1}^{n}\sum_{j' \in [d]} f(j', S_{i',j}) \; \forall \, i \in [n-1], j \in [d]$.

We set $h_n = 0$ for any heuristic, implying no bids are placed after the last reviewer. This is analogous to setting the heuristic value to zero for the target vertex in A*.

[3]The name SUPER* stands for SUperior PERmutations and also indicates the inspiration from A*.

3.2 INTUITION BEHIND THE ALGORITHM

We first provide some intuition about the SUPER* algorithm, and subsequently present a formal description. Since a primary impediment to designing an algorithm is the inability to fully realize the impact of a paper ordering on the paper-side gain until the end of the bidding process, we begin by considering the scenario where $(n-1)$ reviewers have already departed, and the problem is to determine the ordering of papers to show the final reviewer. In this scenario, we have access to the bids of all $(n-1)$ reviewers that have already arrived and the orderings of papers presented to them. We use the notation $g_{n-1,j} \in \{0, \ldots, n-1\}$ to denote the number of bids received by any paper $j \in [d]$ at the time of arrival of the last reviewer. The values $\{g_{n-1,1}, \ldots, g_{n-1,d}\}$ are thus known at the time when the final reviewer arrives. As a result, we can formulate an optimization problem for the final reviewer n to maximize the expected gain from (2) in the following manner. For every $j \in [d]$, let $\mathcal{B}_{n,j}$ denote a Bernoulli random variable with mean $p_{i,j} = f(\pi_n(j), S_{n,j})$, independent of all else. The random variable $\mathcal{B}_{n,j}$ represents the bid of the final reviewer on paper $j \in [d]$. The optimization problem is then

$$\begin{aligned}\max_{\pi_n \in \Pi_d} \sum\nolimits_{j \in [d]} \mathbb{E}[\gamma_p(g_{n-1,j} + \mathcal{B}_{n,j})] \\ + \lambda \sum\nolimits_{j \in [d]} \gamma_r(\pi_n(j), S_{n,j}),\end{aligned} \tag{4}$$

where the expectation is taken over the distribution of the random variables $\mathcal{B}_{n,1}, \ldots, \mathcal{B}_{n,d}$.

Observe that the constraint set for the optimization problem in (4) is the set Π_d of all permutations. This set is, in general, not very well behaved (Ailon et al., 2008; Shah et al., 2016), which makes even this one-step optimization a challenge. As we discuss later in the formal algorithm description along with Theorem 1 and its proof, SUPER* for the final reviewer optimally solves (4) and it is computationally efficient manner. The aforementioned subproblem forms the starting point for the SUPER* algorithm. Now that we know to handle a single (last) reviewer in an optimal fashion, we now describe the SUPER* algorithm for a general reviewer, say, $i \in [n]$. When reviewer i arrives, we have access to the number of bids made by all past reviewers on any paper $j \in [d]$, which we denote by $g_{i-1,j} \in \{0, \ldots, i-1\}$.

We now recall the A* algorithm: for any vertex, A* considers the cost "g" so far and a heuristic estimate "h" of the subsequent cost. Then, considering the cost of any vertex as "$g+h$", the A* algorithm takes the one-step optimal action given by selecting the neighboring vertex with the smallest value of "$g+h$". In an analogous fashion, SUPER* considers the number of bids so far (g_{i-1}) and takes as input a heuristic (h_i) for the number of bids

Algorithm 1: SUPER*

Input: $\gamma_p : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, paper-side gain
$\gamma_r : [d] \times [0,1] \to \mathbb{R}_{\geq 0}$, reviewer-side gain
$f : [d] \times [0,1] \to [0,1]$, bidding model
$\lambda \geq 0$, trade-off parameter
$S \in [0,1]^{n \times d}$, similarity matrix.

Algorithm:

1. Initialize bids on each paper to zero: $g_0 \leftarrow 0_d$
2. For each reviewer arrival $i \in [n]$
   (a) Compute or input heuristic $h_i \in [0, n-i]^d$
   (b) $\pi_i \leftarrow$ FindPaperOrder
   (c) Present papers in the order π_i
   (d) Observe vector of bids $b_i \in \{0,1\}^d$
   (e) Update paper bid counts: $g_i = g_{i-1} + b_i$

in the future. Then, considering the number of bids from all other reviewers as "$g_{i-1} + h_i$", SUPER* takes the action which is the one-step optimal action. In other words, SUPER* solves for each paper ordering using:

$$\max_{\pi_i \in \Pi_d} \sum_{j \in [d]} \mathbb{E}[\gamma_p(g_{i-1,j} + h_{i,j} + \mathcal{B}_{i,j})] + \lambda \sum_{j \in [d]} \gamma_r(\pi_i(j), S_{i,j}) \tag{5}$$

where $\mathcal{B}_{i,j}$ is a Bernoulli random variable with mean $p_{i,j} = f(\pi_i(j), S_{i,j})$ and independent of all else. As for the final reviewer, SUPER* is efficient for any reviewer.

3.3 FORMAL ALGORITHM DESCRIPTION

The SUPER* algorithm is presented in Algorithm 1. To determine a paper ordering to show any reviewer, SUPER* calls a procedure to efficiently solve (5). We give a general method in Algorithm 2 and a faster method in Algorithm 3 that is applicable for a special class of reviewer-side gain and bidding functions.

General version. In the general version of SUPER*, Algorithm 2 is called to return a paper ordering that is a solution to (5) each time a reviewer arrives. In the proof of Theorem 1, we show that the optimization problem over the set of permutations given in (4) to find the optimal paper ordering for the final reviewer can be reformulated as an integer linear programming problem with a totally unimodular constraint set. The totally unimodular property of the constraint set guarantees that the solution of a relaxed linear program is in fact the integer optimal solution. The application of this reduction from an optimization problem over permutations to a linear programming problem for any given reviewer forms the technique given in Algorithm 2 to efficiently obtain a solution to (5). Finally, the per-reviewer time complexity of the general version of SUPER* given the evaluations of the heuristic is $\mathcal{O}(d^3)$ (see Proposition 1 in Appendix B.1).

Algorithm 2: FindPaperOrder

1. Compute weight matrix $w \in \mathbb{R}^{d \times d}$ such that

$$w_{j,k} = \lambda \gamma_r(k, S_{i,j}) + f(k, S_{i,j})(\gamma_p(g_{i-1,j} + h_{i,j} + 1) - \gamma_p(g_{i-1,j} + h_{i,j}))$$

2. Solve linear program to obtain $x^* \in \mathbb{R}^{d \times d}$ with ties broken arbitrarily between maximizing solutions:

$$x^* \in \arg\max_{x \in [0,1]^{d \times d}} \sum_{j \in [d]} \sum_{k \in [d]} w_{j,k} x_{j,k}$$
$$\text{s.t.} \sum_{k \in [d]} x_{j,k} = 1 \; \forall \, j \in [d], \; \sum_{j \in [d]} x_{j,k} = 1 \; \forall \, k \in [d]$$

3. $\pi_i(j) = k$ such that $x^*_{j,k} = 1$ for each $j \in [d]$

Output: π_i

Algorithm 3: FindPaperOrderEfficient

1. Compute weights for each $j \in [d]$:

$$\alpha_{i,j} = \lambda \gamma_r^S(S_{i,j}) + f^S(S_{i,j})(\gamma_p(g_{i-1,j} + h_{i,j} + 1) - \gamma_p(g_{i-1,j} + h_{i,j}))$$

2. $\pi_i = \sigma(\alpha_i)$, where $\sigma : \mathbb{R}^d \to [d]^d$ returns the rank from maximum to minimum of each input in place and breaks ties arbitrarily.

Output: π_i

Faster specialized version. Given a bidding function that can be decomposed into $f(\pi_i(j), S_{i,j}) = f^S(S_{i,j}) f^\pi(\pi_i(j))$ where $f^S : [0,1] \to [0,1]$ is non-decreasing and $f^\pi : [d] \to [0,1]$ is non-increasing, along with a reviewer-side gain function that can be decomposed as $\gamma_r(\pi_i(j), S_{i,j}) = \gamma_r^S(S_{i,j}) f^\pi(\pi_i(j))$ where $\gamma_r^S : [0,1] \to \mathbb{R}_{\geq 0}$ is non-decreasing, SUPER* calls Algorithm 3 to return a paper ordering that is a solution to (5) each time a reviewer arrives. In the proof of Proposition 1 in Appendix B.1, we show for this model class that the problem from (4) to find the optimal paper ordering for the final reviewer after evaluating the expectation can be reformulated as

$$\max_{\pi_n \in \Pi_d} \sum_{j \in [d]} \alpha_{n,j} f^\pi(\pi_n(j)) \tag{6}$$

for some non-negative weights $\{\alpha_{n,j}\}_{j \in [d]}$. The problem in (6) admits a simple solution: f^π is non-increasing on the domain, so the objective is maximized by presenting

papers in decreasing order of the weights $\{\alpha_{n,j}\}_{j\in[d]}$. Obtaining this solution only requires sorting the weights, which has a time complexity of $\mathcal{O}(d\log(d))$. The application of this problem reformulation for the given model class and any reviewer forms the technique given in Algorithm 3 to obtain a solution to (5).

Before moving on to present our theoretical results, we comment on the relevance of this model class. Importantly, the DCG reviewer-side gain function and bidding model $f(S_{i,j}, \pi_i(j)) = S_{i,j}/\log_2(\pi_i(j)+1)$, which we have mentioned as running examples that can be kept in mind, satisfy the decomposition for which SUPER* is computationally efficient. This choice of functions is standard in the past literature on ranking models and click behavior (Järvelin and Kekäläinen, 2000; Aslanyan and Porwal, 2019), meaning that the time complexity result for this model class is quite relevant.

4 Theoretical Results

We now present the main theoretical results of this paper. Complete proofs of all results are in Appendix A.

4.1 LOCAL OPTIMALITY

The property of local optimality, as the name suggests, means that the algorithm is optimal with respect to the reviewer under consideration. Achieving even a good local performance in a computationally efficient manner is challenging due to the optimization over permutations in (4). The following results show that SUPER*, which is computationally efficient, is locally optimal.

The result is first presented in terms of the final reviewer for simplicity and extended to a general reviewer subsequently. In the following theorem, since we consider only the final reviewer, note that the heuristic for SUPER* is irrelevant because the heuristic value for the final reviewer is always set to zero.

Theorem 1. *Given any history of paper orderings and bids from reviewers that arrived previously, the paper ordering given by SUPER* to the final reviewer maximizes the expected gain conditioned on the history.*

In other words, the expected amount by which the gain is increased from the final reviewer is maximized. To generalize the previous result to a local optimality result for any reviewer, let the immediate gain from a reviewer be defined as the difference between the gain after and before the reviewer arrived.

Corollary 1. *Given any history of paper orderings and bids from reviewers that arrived previously, the paper ordering given to any reviewer by SUPER* with zero heuristic maximizes the expected immediate gain from that reviewer conditioned on the history.*

The property of local optimality also implies optimality of SUPER* (with any heuristic) when the paper-side gain function is linear. We refer the reader to Appendix B.2 for more details. We now show that an analogous local optimality statement cannot be made regarding the other baseline methods. In fact, in contrast to SUPER*, all the popular baselines are considerably suboptimal.

Theorem 2. *Consider a model with the paper-side gain function $\gamma_p(g_j) = \sqrt{g_j}$, reviewer-side gain function $\gamma_r(\pi_i(j), S_{i,j}) = (2^{S_{i,j}}-1)/\log_2(\pi_i(j)+1)$, and bidding function $f(\pi_i(j), S_{i,j}) = S_{i,j}/\log_2(\pi_i(j)+1)$. There exists a constant $c > 0$ such that for every $d \geq 2$ and $\lambda \geq 0$, in the worst case for the final reviewer: SIM, BID, and RAND are suboptimal by additive factors of at least $cd/\log_2^2(d)$, $cd\max\{1,\lambda\}/\log_2^2(d)$, and $cd\max\{1,\lambda\}/\log_2^2(d)$, respectively.*

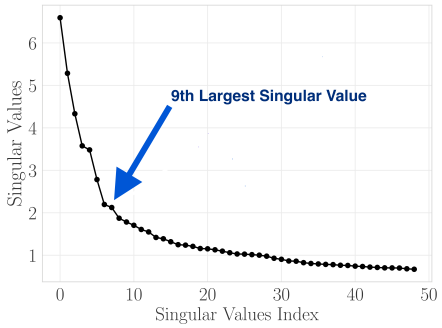
Theorems 1 and 2 in tandem show that SUPER* not only is locally optimal but can outperform currently popular algorithms by a wide margin.

4.2 GLOBAL OPTIMALITY UNDER A COMMUNITY MODEL

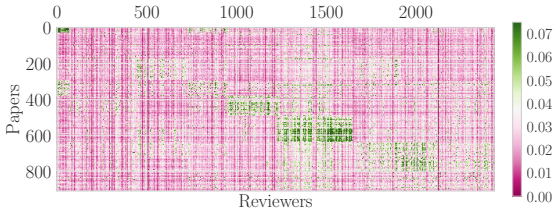
We now transition to consider the global performance of the algorithms. Given our focus on the application of peer review, we are motivated to give guarantees on the performance of SUPER* for similarity matrix classes that would be encountered in a real conference.

A common characteristic of networks is community structure (Newman and Girvan, 2004), where nodes can be grouped into clusters and links between groups are not as common. Pertinent to this work, empirical investigations have revealed community structures in scientific collaboration networks (Newman, 2001). Given this close connection, and the fact that scientific research is highly specialized, it is intuitive that communities exist in major conferences pertaining to different subtopics.

We explore the possible existence of such structure in the ICLR 2018 similarity matrix that was reconstructed by Xu et al. (2019) and is of size $n = 2435$ and $d = 935$. To begin, we investigate the spectral properties of the similarity matrix from ICLR 2018, and in particular, whether it is low rank. We plot the singular values of the similarity matrix in Figure 1a, where the (heuristic) elbow method suggests a low rank (≈ 10). In Figure 1b we plot the entries of the similarity matrix after permuting its rows and columns according to the spectral co-clustering algorithm (Dhillon, 2001). The result suggests the ICLR 2018 similarity matrix exhibits some characteristics of a



(a)



(b)

Figure 1: (a) The 50 singular values (excluding the maximum singular value) of the ICLR 2018 similarity matrix, which shows low-rank structure. (b) Similarity scores of the permuted ICLR matrix as a heatmap indicating the block diagonal structure.

noisy block diagonal structure.

We now perform an associated theoretical analysis of the algorithms under such community structures of the similarity matrix. We begin by proposing a simple model which we call the 'noiseless community model'.

Noiseless community model. Informally, the noiseless community model we study is a set of similarity matrices that up to a permutation of rows and columns belong to a subclass of block diagonal matrices. Formally, let $\mathbf{0}_{q\times q}$ and $\mathbf{1}_{q\times q}$ denote $q \times q$ matrices of all zeros and all ones respectively. Define an $mq \times mq$ block matrix B as:

$$B = \begin{bmatrix} \mathbf{1}_{q\times q} & \mathbf{0}_{q\times q} & \cdots & \mathbf{0}_{q\times q} \\ \mathbf{0}_{q\times q} & \mathbf{1}_{q\times q} & \cdots & \mathbf{0}_{q\times q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{q\times q} & \mathbf{0}_{q\times q} & \cdots & \mathbf{1}_{q\times q} \end{bmatrix}.$$

Finally, denote by $\mathcal{P}_{mq\times mq}$ the set of all $mq \times mq$ permutation matrices. The noiseless community model is defined as the following set of similarity matrices for $m \geq 2$ and $q \geq 2$:

$$\{S = P(sB)\widetilde{P} : s \in [0.01, 1],\ P, \widetilde{P} \in \mathcal{P}_{mq\times mq}\}. \quad (7)$$

The number of reviewers is given by $n = mq$ and the number of papers by $d = mq$. In words, this is the set of all similarity matrices obtained via a permutation of the rows and columns of the block matrix B.

We begin by showing that under the noiseless community formulation, both SUPER* and SIM are optimal, whereas BID and RAND fare poorly.

Theorem 3. *Consider a model with the paper-side gain function $\gamma_p(g_j) = \sqrt{g_j}$, reviewer-side gain function $\gamma_r(\pi_i(j), S_{i,j}) = (2^{S_{i,j}} - 1)/\log_2(\pi_i(j) + 1)$, and $f(\pi_i(j), S_{i,j}) = \mathbb{1}\{\pi_i(j) = 1\}\mathbb{1}\{S_{i,j} > s/2\}$ as the bidding function. Then, under the noiseless community model from (7), for all $m \geq 2$, $q \geq 2$ and $\lambda \geq 0$: SUPER* with zero heuristic and SIM are optimal. In contrast, there exists a constant $c > 0$ such that for all $m \geq 2$, $q \geq 2$ and $\lambda \geq 0$: BID and RAND are suboptimal by additive factors of at least $c\lambda mq/\log_2^2(mq)$ and cmq, respectively.*

Although SIM is optimal in the noiseless community model, this optimality is quite brittle. As we show below, even an infinitesimally small amount of noise makes SIM considerably suboptimal. In contrast, SUPER* is robust enough and suffers by only a small amount.

Noisy community model. We first define a 'noisy community model'. Under this model, we assume that the similarity matrix is generated by first selecting any similarity matrix S' from the noiseless community model defined in (7), and then adding noise to its entries as:

$$S_{i,j} = \begin{cases} s - \nu_{i,j} & \text{if } S'_{i,j} = s \\ \nu_{i,j} & \text{if } S'_{i,j} = 0, \end{cases} \quad (8)$$

where $\nu_{i,j}$ is drawn independently and uniformly from $(0, \xi)$ for each reviewer-paper pair, for some small value ξ to be defined subsequently. The next result shows that even under an arbitrarily small perturbation ξ from a noiseless community model, the baselines become significantly suboptimal. In contrast, SUPER* is robust to the noise and is near-optimal.

Theorem 4. *Consider the gain and bidding functions from Theorem 3 and the noisy community model given in (8) with any noise bound satisfying $\xi \leq (1 + \lambda)^{-1}e^{-emq}$. Then, for every $m \geq 2$, $q \geq 2$, and $\lambda \geq 0$: SUPER* with zero heuristic is within at least an additive factor of 0.0001 of the optimal. In contrast, there exists a constant $c > 0$ such that for all $m \geq 2$, $q \geq 2$ and $\lambda \geq 0$, with respect to SUPER* with zero heuristic: SIM, BID, and RAND are suboptimal by additive factors of at least cmq, $c\lambda mq/\log_2^2(mq)$, and cmq, respectively.*

This result thus establishes the global optimality of SUPER* for the community model, while in contrast all popular baselines are considerably suboptimal.

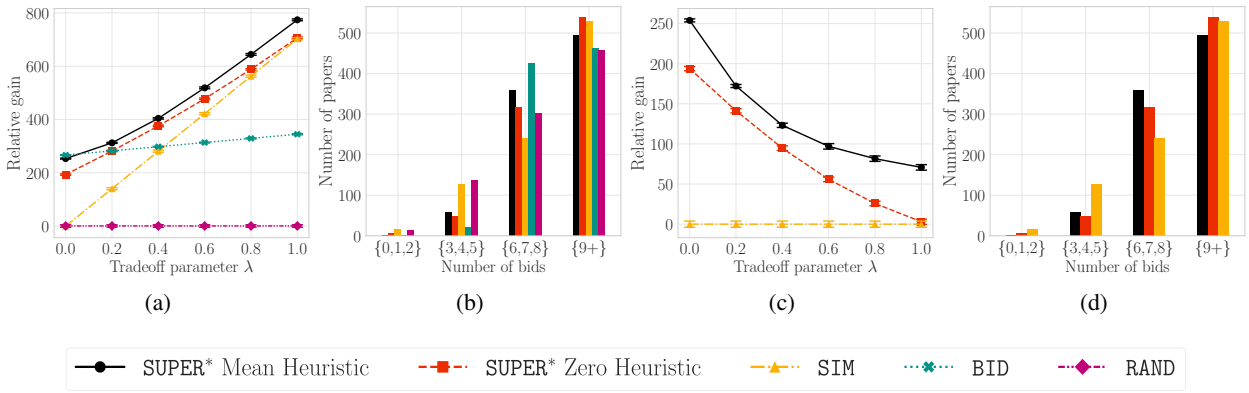


Figure 2: ICLR 2018 similarity matrix experiment.

5 EXPERIMENTAL RESULTS

We now empirically evaluate SUPER* (with zero and mean heuristics) and compare it with the baselines SIM, BID, and RAND (discussed earlier in Section 2). The experimentation methodology is as follows for any chosen set of model parameters. Given a fixed similarity matrix, we shuffle the rows of the matrix to randomize the sequence of reviewer arrivals and simulate each of the algorithms. Then, for each algorithm, we record the gain along with the number of papers that end up with bid counts in the intervals $\{0,1,2\}$, $\{3,4,5\}$, $\{6,7,8\}$, and $\{9+\}$. We repeat this process 20 times for a given similarity matrix. To evaluate performance, we show the means of the relative gains (additive gains relative to the gain of a baseline) across the runs and include error bars representing the standard error of the mean. Moreover, we present the mean number of papers across the repeated simulations that finish with bid counts in each of the previously given bid count intervals. The code and data to reproduce each of the experiments is available at `github.com/fiezt/Peer-Review-Bidding`.

We perform evaluations on the ICLR 2018 similarity matrix. The model we consider has a paper-side gain function $\gamma_p(g_j) = \min\{g_j, 6\}$, reviewer-side gain function $\gamma_r(\pi_i(j), S_{i,j}) = (2^{S_{i,j}} - 1)/\log_2(\pi_i(j) + 1)$, and bidding function $f(\pi_i(j), S_{i,j}) = S_{i,j}/\log_2(\pi_i(j) + 1)$. We remark that the paper-side gain function is a natural choice given that conferences often assign three reviewers to each paper and as such they may seek twice the number of bids per paper. Moreover, recall that for this pair of reviewer-side gain and bidding functions, the efficient routine in Algorithm 3 can be called in place of Algorithm 2 in SUPER* to retrieve a paper ordering, which is what we implement.

The results of the experiment are presented in Figure 2. In Figures 2a–2b we compare SUPER* to each baseline and in Figures 2c–2d we zoom in and only show the results for SUPER* and SIM. In terms of the gain results shown in Figures 2a and 2c, each version of SUPER* outperforms the baseline algorithms, while BID outperforms SIM when minimal weight λ is given to the reviewer-side gain and vice versa when a significant amount of weight λ is given to the reviewer-side gain. In Figures 2b and 2d, the distribution of the bid counts obtained for the algorithms are shown with $\lambda = 0.8$, which was chosen since this parameter choice gave nearly equal paper-side and weighted reviewer-side gain for RAND.

While BID has a similar number of papers with fewer than the minimum number of desired bids as each version of SUPER*, the gain demonstrates why it is not a generally adopted method. As a result of showing papers of limited relevance early in the paper orderings to elicit bids on papers with few bids, the overall gain is significantly smaller than SIM and SUPER* since the reviewer-side gain is worse. The distributions of the bid counts on the papers illustrate that SUPER* reduces the number of papers that end with fewer than the desired minimum number of bids by 60% compared to SIM and RAND.

In Appendix C, we present experiments on the ICLR 2018 similarity matrix with variations of the model parameters and under real-world complexities such as bid probability mismatch, reviewers failing to arrive, and reviewers arriving simultaneously. Moreover, we simulate several synthetic similarity score structures. We observe that SUPER* consistently outperforms the baselines in terms of the gain and reduces the number of papers with fewer than requisite bids by 50-75% or more.

Acknowledgements. This work was supported by NSF grants CIF 1763734, CAREER: CIF 1942124, CRII: CIF 1755656, and a DoD NDSEG fellowship.

References

D. Agarwal, B. Chen, P. Elango, and X. Wang. Click shaping to optimize multiple objectives. In *KDD*, 2011.

N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: ranking and clustering. *JACM*, 2008.

G. Aslanyan and U. Porwal. Position bias estimation for unbiased learning-to-rank in ecommerce search. In *ACM Symposium on SPIRE*. Springer, 2019.

A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. *JACM*, 2018.

N. Black, S. Van Rooyen, F. Godlee, R. Smith, and S. Evans. What makes a good reviewer and a good review for a general medical journal? *Jrnl. AMA*, 1998.

G. Cabanac and T. Preuss. Capitalizing on order effects in the bids of peer-reviewed conferences to secure reviews by expert referees. *American Society for Information Science and Technology*, 2013.

L. Charlin and R. Zemel. The Toronto paper matching system: an automated paper-reviewer assignment system. In *ICML Workshop on Peer Reviewing and Publishing Models*, 2013.

A. Chuklin, I. Markov, and M. d. Rijke. Click models for web search. *Syn. Lec. ICRS*, 2015.

I. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *KDD*, 2001.

W. Ding, N. B., and W. Wang. On the privacy-utility tradeoff in peer-review data analysis. *arXiv*, 2020.

P. Hart, N. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Systems Science and Cybernetics*, 1968.

T. Jambor and J. Wang. Optimizing multiple objectives in collaborative filtering. In *RecSys*, 2010.

K. Järvelin and J. Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In *SIGIR*, 2000.

S. Jecmen, H. Zhang, R. Liu, N. B. Shah, V. Conitzer, and F. Fang. Mitigating manipulation in peer review via randomized reviewer assignments. *arXiv*, 2020.

A. Kobren, B. Saha, and A. McCallum. Paper matching with local fairness constraints. In *SIGKDD*, 2019.

N. Lawrence and C. Cortes. The NIPS experiment, 2014.

J. Murphy, C. Hofacker, and R. Mizerski. Primacy and recency effects on clicking behavior. *Journal of Computer-Mediated Communication*, 2006.

M. E. Newman. The structure of scientific collaboration networks. *National academy of sciences*, 2001.

M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phy. review*, 2004.

R. Noothigattu, N. Shah, and A. Procaccia. Loss functions, axioms, and peer review. *arxiv:1808.09057*, 2018.

A. L. Porter and F. A. Rossini. Peer review of interdisciplinary research proposals. *Science, technology, & human values*, 1985.

M. Rodriguez, J. Bollen, and H. Van de Sompel. Mapping the bid behavior of conference referees. *Journal of Informetrics*, 2007.

M. Rodriguez, C. Posse, and E. Zhang. Multiple objective optimization in recommender systems. In *RecSys*, 2012.

N. Shah, S. Balakrishnan, A. Guntuboyina, and M. Wainwright. Stochastically transitive models for pairwise comparisons: Statistical and computational issues. In *ICML*, 2016.

N. B. Shah, B. Tabibian, K. Muandet, I. Guyon, and U. Von Luxburg. Design and analysis of the NIPS 2016 review process. *JMLR*, 2018a.

N. B. Shah, B. Tabibian, K. Muandet, I. Guyon, and U. Von Luxburg. Design and analysis of the NIPS 2016 review process. *JMLR*, 2018b.

A. Singh and T. Joachims. Policy learning for fairness in ranking. In *NeurIPS*, 2019.

I. Stelmakh, N. Shah, and A. Singh. On testing for biases in peer review. In *NeurIPS*, 2019a.

I. Stelmakh, N. Shah, and A. Singh. PeerReview4All: Fair and accurate reviewer assignment in peer review. In *ALT*, 2019b.

I. Stelmakh, N. Shah, and A. Singh. Catch me if i can: Detecting strategic behaviour in peer assessment. *arXiv*, 2020a.

I. Stelmakh, N. Shah, A. Singh, and H. Daumé III. Prior and prejudice: The bias against resubmissions in conference peer review. *arXiv*, 2020b.

S. Thurner and R. Hanel. Peer-review in a world with rational scientists: Toward selection of the average. *The European Physical Journal B*, 2011.

A. Tomkins, M. Zhang, and W. D. Heavlin. Reviewer bias in single-versus double-blind peer review. *PNAS*, 2017.

J. Wang and N. B. Shah. Your 2 is my 1, your 3 is my 9: Handling arbitrary miscalibrations in ratings. In *AAMAS*, 2019.

Y. Xu, H. Zhao, X. Shi, and N. Shah. On strategyproof conference review. In *IJCAI*, 2019.

H. Yadav, Z. Du, and T. Joachims. Fair learning-to-rank from implicit feedback. *arXiv:1911.08054*, 2019.