Incentives in Choosing Academic Research Projects

Project report for CS 269I: Incentives in Computer Science

Yair Carmon (`yairc`) Ingerid Fosli (`ifosli`) Nate Gruver (`ngruver`)

December 8, 2018

Abstract

We explore modifications to a stylized game-theoretic model of academic research, proposed by Kleinberg and Oren [9]. Our primary contribution is a utility penalty term that reflects researchers' aversion to the risk of being scooped; surprisingly, this "scoop penalty" improves social welfare. Theoretically, we prove that for identical researchers and pure Nash equilibria, the penalty improves the price of anarchy in worst-case instances from 2 to below 1.5. Empirically, we demonstrate this penalty also significantly improves social welfare for randomly sampled problem instances, when policies are learned via no-regret dynamics, even for non-identical researchers. In addition, we formulate and experiment with the following model modifications: credit replication, risk constraint and a reward-variance penalty.

1 Introduction

1.1 Problem description

The ideal academic research project is (a) important if successful, (b) likely to succeed and (c) unlikely to get scooped. These three desiderata are necessarily conflicting—e.g. a project that excels in (a) and (b) will inevitably attract competition and do poorly in (c)—and researchers face difficult strategic decisions when choosing what to work on. Do researchers acting selfishly optimize the social welfare of academic pursuits? To explore this, we consider a basic game-theoretic model and examine the resulting behavior.

1.2 Prior work

The problem of resource allocation for the production of knowledge has been studied in the context of game theory since at least the 1970s, when Kenneth Arrow published his economic model of invention [5]. Since then, much work has been done on modeling incentives in choosing research projects. In the literature, many works focus on the collaborative aspect of research and its game-theoretic modeling [10, 7, 3, 2, 1]. Others use graph-theoretic analysis of academic research to inform research selection strategies [14, 8]. The starting point of our project is the following simplified model put forth by Kleinberg and Oren [9].

1.3 Model: the Project Game

Consider a community of n researchers, each choosing to work on a single project from a set of m different projects. Each project j has importance w_j, which should be thought of as the project's contribution to society, if successful. A researcher i working on project j succeeds with probability $p_{i,j}$, independently of any other researcher working on this project. For any allocation $x : [n] \to [m]$ of researchers to projects, the social welfare is the expected importance of the successful projects,

$$W(x) = \sum_{j=1}^{m} w_j \mathbb{P}(\text{some researcher succeeds on } j) = \sum_{j=1}^{m} w_j \left(1 - \prod_{i \in X_j} (1 - p_{i,j})\right),$$

where here and in the sequel X_j denotes the set of researchers allocated to project j.

In the model, the incentive for researchers to succeed on projects is *credit*, which is distributed according to a simple rule: if project j is successful, w_j credit is divided evenly among the researchers that succeeded in the project. Specifically each successful researcher to succeed in project j receives w_j/K_j credit, where K_j is the number of researchers who succeeded in project j. A researcher's utility is the expected amount of credit she receives:

$$u_i(x) = \mathbb{E}[\text{credit for project } x_i] = w_{x_i} p_{i,x_i} \mathbb{E}\left[\frac{1}{1 + K_j(x_{\setminus i})}\right], \tag{1}$$

where $K_j(x_{\setminus i})$ is the (random) number of researchers other than i that are successful on project j under allocation a. We refer to this credit-splitting rule as *equal split*, and note that it is a special case of the Shapley utility [11].

Nash equilibria (NE) in the game defined by this utility function do not always maximize the social welfare, and a maximizing allocation is in general NP-hard to compute, as shown by [9]. However, they also show that the price of anarchy in this game is at most 2, and provide somewhat sharper bounds when $p_{i,j}$ is the same for every i. The primary focus of their paper is modifications of the credit allocation rule that make NE socially optimal, whose computation require centralized processing. In our project we instead propose completely decentralized modifications to the utility function, intended to make it more realistic.

1.4 Our contributions and paper outline

In Section 2 we introduce the "scoop penalty": a penalty term added to the utility, which is motivated by the observation that researchers suffer professional and emotional costs—not captured by the basic model—when somebody else succeeds at the project they were working on. In Section 2.1, we analyze this penalty in the simpler case of researchers with identical success probability for any given project. We show that, with a simple choice of parameters, the scoop penalty guarantees price of anarchy smaller than 1.5, for any number of researchers, w.r.t. pure Nash equilibria. In contrast, without the penalty, there exist instances with n identical researchers and a pure Nash equilibrium that is suboptimal by a factor of $2 - \frac{1}{n}$ [9]. We consider the case of non-identical researcher in Section 2.2, and provide robust price of anarchy bounds for the scoop penalty. Here, we could not prove beneficial effect of the penalty, but we can show that, with the parameters determined in Section 2.1, the robust PoA with different players is at most 2.5. Is Section 2.3, we provide additional interpretation and discussion of the scoop penalty.

In Section 3, we propose some additional model modification, motivated by a desire for greater realism: different credit splitting rules (Section 3.1) and failure aversion penalties (Section 3.2), based either on per-researchers risk tolerance threshold, or variance-based utility penalty terms. Unlike the scoop penalty, we show that each of these modifications can bring about catastrophically bad social welfare. We discuss the implications of these conclusion in Section 3.3.

Finally, in Section 4 we report on our empirical study. The study consists of generating random game instances with $n = 50$ researchers working on $m = 100$ projects, and solving them using the different utility functions proposed and three different no-regret learning algorithms. We first describe our experimental protocol, including instance generation and optimization methods (Sections 4.1–4.4). Next, we present and discuss our experimental findings (Section 4.5). The most notable finding is that, for identical researchers, the scoop penalty improves the ensemble-averaged suboptimality by over 10%, achieving nearly optimal allocations, and, for different researchers, the scoop penalty also provides significantly better results.

2 The Scoop Penalty

Let $u_i(x)$ denote some base utility function, representing the utility of researcher i from strategy x. We propose to augment the base utility function with a "scoop penalty": a punishment player i receives when another player succeed in her chosen project. The penalty aims to reflect repercussions of being scooped that are otherwise unaccounted for in the model, e.g. loss of partial research progress, and the emotional toll of efforts gone to waste.

Such penalty should be proportional to the probability that another player succeeds in the project. It also makes sense to make the penalty proportional to the the importance of the project, since the pain of getting scooped certainly increases the more the scooped result is celebrated. Moreover, making the penalty proportional to the project importance ensures that the penalty and the base utility function have matching scales. Thus, if strategy x has player i choose project j (i.e. $x_i = j$), we would like to penalize her by a quantity proportional to

$$w_j \mathbb{P}\left(\text{another succeeds in } j\right) = \text{social welfare from project } j \text{ with } i \text{ removed} \coloneqq W_j\left(x_{\setminus i}\right).$$

The proposed modified utility is therefore

$$\tilde{u}_i(x) = u_i(x) - \rho_{i,x_i} W_{x_i}\left(x_{\setminus i}\right) \tag{2}$$

where the constants $\rho_{i,j}$ represent how much player i is averse to getting scooped on project j. One natural setting is $\rho_{i,j} = \rho\left(1 - p_{i,j}\right)$, which gives the penalty the interpretation "in the event that player i fails at project j and someone succeeds in the project, he pays a fine ρw_j" (and no fine otherwise). For general $\rho_{i,j}$, the penalty can be interpreted as "in the event player i chooses project j and anyone else succeeds, he pays a fine $\rho_{i,j} w_j$" (regardless of whether player i succeeded as well). Under this interpretation the penalty is perhaps more accurately described as an "envy" penalty.

2.1 Guarantees for identical players

We begin with the special case where all the players are identical; $p_{i,j} = p_j$ and $\rho_{i,j} = \rho_j$ for every i, j; we refer to this case as the *Project Game with Identical Players*. For a strategy vector $x = [x_1, \dots x_m]$ we denote the number of players assigned to every project by the corresponding upper case letter $X = [X_1, \dots, X_m]$, i.e. $X_j = |\{i \mid x_i = j\}|$. The social welfare from project j when strategy x is used is given by

$$W_j(x) = w_j\left(1 - (1 - p_j)^{X_j}\right) \coloneqq W_j[X_j].$$

The base utility under consideration will be the equal split utility (1), which in this case reduces to splitting the *expected* social welfare evenly between all *participants*: $u_i(x) = W_{x_i}[X_{x_i}]/X_{x_i}$. The scoop-penalized utility takes the form

$$\tilde{u}_i(x) = \frac{W_j[X_j]}{X_j} - \rho_j W_j[X_j - 1] \coloneqq \tilde{U}_j[X_j] \text{ where } j = x_i. \tag{3}$$

Note that the utility is decreasing with X_j; therefore this is a congestion game and hence a potential game [12] and has a pure Nash equilibrium. Our first result is a bound on the social welfare of pure Nash equilibria, whose proof follows closely that of [11, Theorem 2].

Theorem 1. *Let x, y be two strategies for the Project Game with Identical Players. If x is a Nash equilibrium for utility (3), then*

$$W(x) \geq \frac{W(y)}{1+\gamma(x,y)},$$

where

$$\gamma(x,y) \coloneqq \max_{j\in[m]} \left\{ \max\left\{ \frac{W[Y_j]}{W[X_j]} - \frac{Y_j}{X_j} - \rho_j (X_j - Y_j) \frac{W[X_j - 1]}{W[X_j]}, \rho_j (Y_j - X_j) \right\} \right\}. \tag{4}$$

We prove Theorem 1 in appendix A.1.

To obtain a price of anarchy bound for the class of pure Nash equilibria, we simply optimize $\gamma(x,y)$ over x and y. For compactness we introduce the standard notation $a \vee b \coloneqq \max\{a,b\}$ and $a \wedge b \coloneqq \min\{a,b\}$.

Corollary 2. *In the Project Game with Identical Players and utility (3), suppose allocation y satisfies $Y_j \leq \bar{n}$ for every j. Any pure Nash equilibrium has utility a least*

$$\frac{W(y)}{1+\max_{j\in[m]} \gamma_j(\bar{n})} \text{ where } \gamma_j(\bar{n}) \coloneqq \max_{\substack{a,b \text{ s.t.} \\ 2\leq b\leq \frac{1}{\rho_j}\wedge n \\ 1\leq a\leq (b-1)\wedge \bar{n}}} \left\{ \frac{W_j[a]}{W_j[b]} - \frac{a}{b} - \rho_j (b-a) \frac{W_j[b-1]}{W_j[b]} \right\} \vee \rho_j \bar{n}. \tag{5}$$

Consequently, if an optimal allocation $y^\star$ satisfies $Y_j^\star \leq \bar{n}$ for every j, the price of anarchy is at most $1 + \max_{j\in[m]} \gamma_j(\bar{n})$.

Proof. To derive the bound we simply maximize $\gamma(x,y)$ in (4) subject to $Y_j \leq \bar{n}$ for any j . The only piece that requires explanation is the upper bound $b \leq \frac{1}{\rho_j}$ in the definition of $\gamma_j(\bar{n})$. To see why we may limit b so, fix some j and suppose $b \geq \frac{1}{\rho_j}$. We have

$$\frac{W_j[a]}{W_j[b]} - \frac{a}{b} - \rho_j (b-a) \frac{W_j[b-1]}{W_j[b]} \leq \frac{W_j[a]}{W_j[b]} - \frac{a}{b} - \left(1-\frac{a}{b}\right) \frac{W_j[b-1]}{W_j[b]} \leq \frac{W_j[a] - W_j[b-1]}{W_j[b]} \leq 0,$$

where the last inequality is due to $a \leq b-1$. Therefore there is no point in considering $b \geq 1/\rho_j$. This is not surprising, given that the penalized utility $\tilde{U}_j[k]$ is negative for $k \geq 1 + \frac{1}{\rho_j}$; the scoop penalty in effects implements a hard cutoff on the number of participants in any project. □

Corollary 2 tells us that the only downside of of having $\rho \neq 0$ comes from the term $\rho\bar{n}$, that depends on the a-priori bound $\bar{n}$; if we only know that $\bar{n} \leq n$, it would seem that we must set $\rho \leq 1/n$ to achieve any meaningful PoA guarantee. However, observe that in order to get a $1-\epsilon$ fraction of the maximum utility available from project j, it is enough to assign roughly $\log\epsilon / \log(1-p_j)$ players to that project. Thus, we can get very close to the optimal allocation with only a few players assigned to each project, effectively creating our own $\bar{n}$. The only projects that may require many players are those with low success probability p_j. However, for a project j with $p_j \ll 1$ the social welfare W_j is close to linear, and $\frac{W_j[a]}{W_j[b]} - \frac{a}{b} - \rho_j (b-a) \frac{W_j[b-1]}{W_j[b]}$ will be small even for small values of ρ. We formalize these observation in the following

Corollary 3. *In the Project Game with Identical Players with utility (3), the price of anarchy w.r.t. pure equilibria is at most*

$$\inf_{\epsilon\in(0,1)} \frac{1}{1-\epsilon} \cdot \left[1 + \max_{j\in[m]} \gamma_j \left(\left\lceil \frac{\log\epsilon}{\log(1-p_j)} \right\rceil \right) \right] \tag{6}$$

with $\gamma_j(\cdot)$ defined in (5).

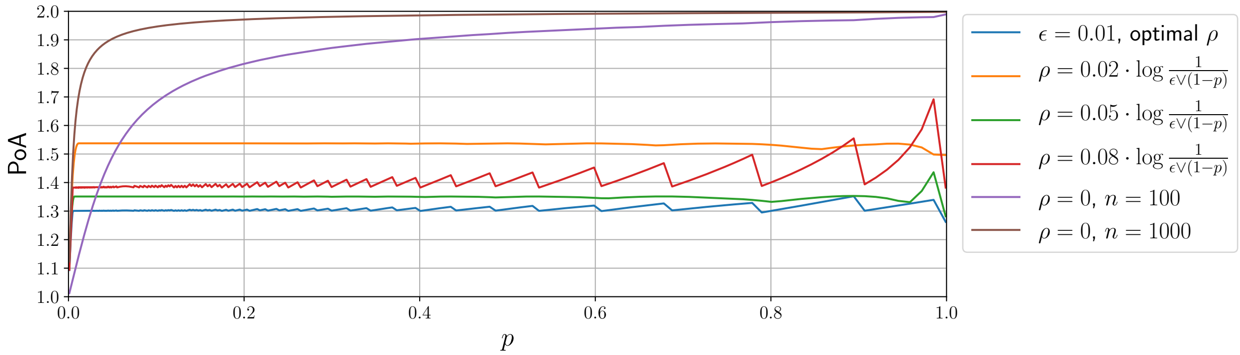

Figure 1: Evaluation of PoA for different success probability p and different values of ρ. For $\rho \neq 0$ we use bound (6) with ϵ fixed at 0.01, and for $\rho = 0$ we use the bound (5) with $\bar{n} = n$.

Proof. Let $y^\star$ be an optimal allocation, and consider the allocation y which assigns

$$Y_j = \min\left\{k \geq 0 \mid W_j\left[k\right] \geq (1-\epsilon)\, W_j\left[Y_j^\star\right]\right\}$$

players to project j. Note that this allocation is always feasible since $Y_j \leq Y_j^\star$ for every j, and that by construction $W\left(y\right) \geq (1-\epsilon)\, W\left(y^\star\right)$. Moreover, by construction we also have

$$W_j\left[Y_j - 1\right] < (1-\epsilon)\, W_j\left[Y_j^\star\right] \leq (1-\epsilon)\, w_j.$$

Using $W_j\left[k\right] = w_j\left(1 - (1-p_j)^k\right)$ we deduce that

$$1 - (1-p_j)^{Y_j - 1} < 1 - \epsilon \Rightarrow Y_j - 1 < \frac{\log \epsilon}{\log\left(1-p_j\right)} \Rightarrow Y_j \leq \left\lceil \frac{\log \epsilon}{\log\left(1-p_j\right)} \right\rceil.$$

Applying Corollary 2 on policy y and using $W\left(y\right) \geq (1-\epsilon)\, W\left(y^\star\right)$ completes the proof. □

In Figure 1 we illustrate Corollary 3 by explicitly evaluating (6) for $\epsilon = 0.01$. For $p_j \in (0,1)$, we show the bound obtained when optimizing over ρ_j, as well as when using

$$\rho_j = c \cdot \log \frac{1}{\epsilon \vee (1-p_j)}, \tag{7}$$

for $c \in \{0.02, 0.05, 0.08\}$. For comparison, we show the PoA bounds when $\rho = 0$ and $n \in \{100, 1000\}$ (the bound (6) holds for any value of n). As can be seen in the figure, the optimal value of ρ_j guarantees PoA below 1.4 for *any* value of p. Moreover, for a fairly wide range of parameters c, the heuristic (7) provides similar PoA bounds for any n and p. In contrast, with $\rho = 0$ the PoA bounds (which are tight in the worst case) depend significantly on p and n. We discuss the heuristic (7) in the end of the next section, where we generalize it to multiple players.

Unfortunately, our PoA guarantees for identical players do not extend beyond pure Nash equilibria, because of the permutation step in the beginning of the proof of Theorem 1. We can possibly address this issue using the more sophisticated techniques in [11], which produce robust PoA bounds that are also valid when every player can choose only a subset of the projects (but success probability for available projects are identical for all players).

2.2 General guarantees

We now consider the general setting, where $p_{i,j}$ and $\rho_{i,j}$ may depend on both i and j. Here we let $u_i(x)$ be any valid utility [15], satisfying $\sum_i u_i(x) \leq W(x)$ and $u_i(x) \geq W(x) - W(x_{\setminus i})$; the "equal split" (Shapley) utility is valid. In this level of generality, we were not able to prove that the price of anarchy decreases. However, we argue that the (natural generalization of) the choice of ρ that provided a PoA bound of 1.5 for identical players, will have robust PoA at most 2.5 in the general case. Without the scoop penalty the PoA is 2 in the worst case even for identical players.

First, we prove a smoothness-type result, which is a fairly trivial extension of the proof of smoothness for general valid utility games. In what follows we let x, y denote assignment vectors, and X, Y denote their respective set-valued inverses, i.e. $X_j = \{i \mid x_i = j\}$ and similarly for Y_j.

Theorem 4. *In the Project Game with penalized utility* (2) *and valid base utility u_i, any two allocations x, y satisfy*

$$\sum_{i \in [n]} \tilde{u}_i(y_i, x_{\setminus i}) \geq W(y) - (1 + \beta(y))W(x) \text{ where } \beta(y) \coloneqq \max_{j \in [m]} \sum_{i \in Y_j} \rho_{i,j}. \tag{8}$$

Therefore, if $W^\star = W(y^\star)$ is the maximal social welfare, the robust price of anarchy is at most

$$\min_y \left\{ (2 + \beta(y)) \frac{W^\star}{W(y)} \right\} \leq 2 + \beta(y^\star).$$

Proof. By definition of the scoop penalty, we have

$$\sum_{i \in [n]} \tilde{u}_i(y_i, x_{\setminus i}) = \sum_{i \in [n]} u_i(y_i, x_{\setminus i}) - \sum_{i \in [n]} \rho_{i,y_i} W_{y_i}(x_{\setminus i}).$$

The well-known analysis of valid utility games (cf. [13]) gives

$$\sum_{i \in [n]} u_i(y_i, x_{\setminus i}) \geq W(y) - W(x).$$

Moreover,

$$\sum_{i \in [n]} \rho_{i,y_i} W_{y_i}(x_{\setminus i}) = \sum_{j \in [m]} \sum_{i \in Y_j} \rho_{i,j} W_j(x_{\setminus i}) \leq \sum_{j \in [m]} \sum_{i \in Y_j} \rho_{i,j} W_j(x) \leq \sum_{j \in [m]} \beta(y) W_j(x) = \beta(y) W(x),$$

establishing the smoothness bound. Noting that $\tilde{u}_i(x) \leq u_i(x)$ for every i and x, we conclude that any coarse correlated Nash equilibrium has expected welfare at least $W(y)/(1 + \beta(y))$. Since this holds for every y, we have our claimed bound on the robust price of anarchy, where the final bound follows from taking $y = y^\star$. □

Unlike Theorem 1 for identical players, the bound of Theorem 4 strictly deteriorates as $\rho_{i,j}$ increase. However, one should note that this bound is quite loose; in the final step of the proof we replace $\tilde{u}_i(x)$ by the larger $u_i(x)$, thereby throwing away any beneficial effect that the scoop penalty could possibly have. In fact, the smoothness bound (8) implies the stronger result

$$(2 + \beta(y))W(x) \geq W(y) + \sum_j \sum_{i \in X_j} \rho_{i,j} W_j(x_{\setminus i})$$

holds in expectation for every distribution on x corresponding to a coarse correlated Nash equilibrium. Unfortunately, it is not clear how to relate the beneficial terms of the form $\rho_{i,j} W_j(x_{\setminus i})$ to either $W(x)$ or $W(y)$ in a way that improves the PoA bound.

The quantity $\beta(y)$ in Theorem 4 roughly corresponds to the term $\rho_j Y_j$ appearing in Theorem 1 for identical players. Similarly to the case of identical players, we are able to control the value of $\beta(y)$ by arguing that $(1-\varepsilon)$-optimal allocations need to succeed with probability at most $1-\varepsilon$ in any given project. We express this formally in the following (see Appendix A.2 for a proof)

Corollary 5. *In the Project Game with penalized utility* (2) *and valid base utility* u_i*, if*

$$\rho_{i,j} = c \cdot \log \frac{1}{\max\{\epsilon, 1 - p_{i,j}\}} \tag{9}$$

for some $c, \epsilon \in (0,1)$*, then the robust price of anarchy is at most*

$$\frac{2 + c \cdot \log \frac{1}{\epsilon}}{1 - \sqrt{\epsilon}}.$$

For $c = 0.05$ *and* $\epsilon = 0.01$ *(as in Figure 1), the robust price of anarchy is at most* $2.4781 \leq 2.5$*.*

2.3 Discussion

As long as $\rho_{i,j}$ is set roughly according to (9), the scoop penalty should never catastrophically affect the resulting equilibrium, and may potentially improve it. Ignoring the ϵ cutoff, the scoop penalty with (9) corresponds to the following "story". If a researcher i tries to work on project j and someone else succeeds, he looks at the resulting paper and estimates his chance $p_{i,j}$ of succeeding in the project. He then deducts from his happiness balance the quantity $w_j/\bar{n}_{i,j}$, where $\bar{n}_{i,j}$ is the number of times researcher i would have needed to choose project j in order to succeed with the overwhelming probability $1 - e^{-1/c}$ (> 0.99999999 for $c = 0.05$). Thus, the likelier the success in the scooped project, the worse the penalty gets. In our opinion, this "story" is fairly consistent with how researchers respond to results that overlap closely with their current work.

The formula (9) is quite attractive for introducing as a designed utility function (say, for robotic researchers), since $\rho_{i,j}$ depends on the problem instance only through $p_{i,j}$, which can be estimated locally. Moreover, just like in the "story" above, when agents develop their policy by repeated gameplay, the probability $p_{i,j}$ needs to be estimated only whenever another agent succeeded in the task, which might make forming an estimate even easier. For example, once a researcher sees a proof of the theorem she was working on, it is easier for her to tell how close she was.

3 Additional Model Modifications

3.1 Credit-splitting schemes

Recall that the "equal split" utility for the Project Game can be defined as the expected credit received under the following simple rule: for project j, w_j credit is divided evenly among all the researchers who succeed in this project. While this utility is conceptually simple and guarantees reasonably good social outcomes in the form of price of anarchy bounds, it is not necessarily realistic as a model for how academic credit is assigned.

In our experience, if two research groups publish very close results within a short period of time from one another, they do not derive half the benefit each would have received had the other group not published their work. For example, papers citing one paper are likely to cite the other as well, and the overall number of citing works should not decrease due to the independent discovery; it might increase! We believe that other proxies of academic credit, such as awards, promotion decision, and media exposure, are also not cut in half due to independent discovery.

For assignment x, let $K_j(x)$ denote a random variable that represents the number of researchers that succeed in project j. The equal split (Shapley) utility of player i from assignment x is

$$u_i(x) = w_j p_{i,j} \cdot \mathbb{E}\left[\frac{1}{1+K_j(x_{\setminus i})}\right] \text{ for } j = x_i.$$

To address the fact that credit is not split evenly, we may consider more general utility functions, of the form

$$u_i(x) = w_j p_{i,j} \cdot \mathbb{E} f(1+K_j(x_{\setminus i})) \text{ for } j = x_i, \tag{10}$$

where $f : \mathbb{N} \to \mathbb{R}_+$ is a non-increasing function, with the even split corresponds to $f(k) = 1/k$. By considering different functions f we can model different mechanism for dividing the credit among successful parties.

Perhaps the simplest possibility is that w_j credit is replicated among every successful researcher; $f(k) = 1$ for every k; we call this *credit replication*. The resulting utility $u_i(x) = w_{x_i} p_{i,x_i}$ is independent of $x_{\setminus i}$ and therefore there is no longer any interaction in the Project Game: the only possible equilibrium is one where every researcher i greedily picks the project j maximizing $w_j p_{i,j}$. Clearly, this can result in outcomes that are very bad from a social welfare perspective. Consider for example the extreme case where $n = m$, $p_{i,j} = 1$ for every i, j, $w_j = 1$ for every $j \geq 2$ and $w_1 = 1 + \delta$ for some $\delta \ll 1$. Under the modified utility with $f(k) = 1$, in any Nash equilibrium all players choose project $j = 1$, and the social welfare is $1 + \delta$. In contrast, the optimal social welfare $n + \delta$ (each player working on a different project). Thus, under this utility the price of anarchy is effectively unbounded.

Arguably, a utility function with $f(k) = 1$ is unrealistic, as there is some degree of loss incurred by researchers in the event of independent discovery. We may consider instead a function of the form $f(1) = 1$ and $f(k) = \beta$ for $k > 1$ and some constant $\beta < 1$, e.g. $\beta = 0.9$. Intuitively, a function f of this form seems realistic, as collision among 3 or more groups are extremely unlikely in practice and therefore values of f for $k > 2$ seem unimportant. However, a closer look reveals that such f suffers from just the same issues as $f(k) = 1$; if in the previous example we change w_1 to $1/\beta + \delta$ we will still have that in equilibrium all researchers work on project $j = 1$, while the socially optimal assignment sets each researchers to a different project, and the equilibrium is suboptimal by a factor of roughly βn. This example reveals the flaw in the above intuition justifying this form of f; massive multi-party collisions must are probably rare in practice *because* there exist a strong incentive against them.

3.2 Risk-aversion penalties

Models of the form (10) assume that researchers only seek to maximize the expectation of the utility awarded to them. However, in reality many researchers seek to guarantee at least some success, and would prefer projects with smaller expected reward, if they have a substantially larger probability of success.

Here we propose two models for such risk aversion. In the first model, every researcher i has a *risk threshold* $\underline{p}_i$, and they will only choose projects with probability of success $p_{i,j} \geq \underline{p}_i$. To formalize this, given a base utility function $u_i(x)$ we may define the penalized utility function

$$\tilde{u}_i(x) = u_i(x) \cdot 1_{\{p_{i,j} \geq \underline{p}_i\}} = \begin{cases} u_i(x) & p_{i,j} \geq \underline{p}_i \\ 0 & \text{otherwise} \end{cases} \text{ where } j = x_i.$$

A second model for risk aversion can be derived by means of a *variance penalty*, where we subtract from the utility a factor proportional to the deviation of the reward, ignoring the other players. This standard deviation equals $w_j\sqrt{(1-p_{i,j})p_{i,j}}$, and the modified utility has the form

$$\tilde{u}_i(x) = u_i(x) - \alpha_i w_j \sqrt{(1-p_{i,j})p_{i,j}} = u_i(x) - (1-p_{i,j})\alpha_i w_j \sqrt{\frac{p_{i,j}}{1-p_{i,j}}} \text{ where } j = x_i.$$

where a_i is a player-dependent risk aversion coefficient. The variance penalty admits the following probabilistic interpretation: if researcher i succeeds in her chosen project j, she obtains a reward specified by the base utility function u_i, and if she fails she pays an additional fine $\alpha_i w_j \sqrt{\frac{p_{i,j}}{1-p_{i,j}}}$. Note that the variance penalty implies a risk-threshold as well: since all the utilities we consider are at most $w_j p_{i,j}$, the penalized utility will be negative whenever

$$w_j p_{i,j} - \alpha_i w_j \sqrt{(1-p_{i,j})p_{i,j}} < 0 \Leftrightarrow p_{i,j} < \frac{1}{1+\alpha_i^2},$$

and will therefore not be chosen.

Both the risk threshold and variance penalty modifications share a clear failure mode, when the most profitable projects from a social welfare perspective have probability of success that is too low. Consider for example the extreme case where $w_1 = 1$ and $w_j = \delta \ll 1$ for every $j > 1$, but $p_{i,1} < \underline{p}_i$ (or $p_{i,1} < (1+\alpha_i^2)^{-1}$) for every $i \in [n]$. In this case, in equilibrium no one will work on project $j = 1$, even though the socially optimal assignment has everyone working on project $j = 1$ when δ is sufficiently small.

3.3 Discussion

All the model modifications proposed in this section suffer from catastrophic failure in some problem instances. This raises two questions: (1) Are such catastrophic failures realistic? (2) If not, does this mean that the proposed modeling modifications are unrealistic?

Regarding the first question, we can say with confidence that the failure mode of Section 3.1 does not occur in reality, as otherwise events where multiple people succeed in the same project would be far more commonplace, and there would be far less diversity in the projects researchers choose to work on. Whether the failure mode of Section 3.2 occur in reality is a more difficult question. Clearly, in practice unsuccessful research projects are quite common, indicating the willingness of researchers to take some risk. However, the probability of success in the research community appears to generally be quite high, with a fair fraction of research inquiries resulting in publications. It is not difficult to imagine that some very risky, very rewarding research projects are not sufficiently attended to, from a social welfare perspective. The relatively small number of theoretical computer scientists directly thinking about P=NP is perhaps an example of such an issue.

Regarding the second question, we remark that worst-case bad behavior does not necessarily imply one in practice. A more refined analysis of the effects of the modifications proposed in the previous section will involve coming up with a probabilistic model for problem instances (i.e. distribution over $\{w_j, p_{i,j}\}$) as well as penalty parameters ($\{\underline{p}_i\}$ or $\{\alpha_i\}$), and seeing how equilibria compare to socially optimal allocation *statistically* (e.g. in expectation). For example, if the distribution is such that $\text{argmax}_j\{w_j p_{i,j}\}$ turn out to be different for different i—so people have different favorite projects—one could imagine that a scheme where credit is replicated instead of split still provides decent social welfare. Similarly, if the risk-aversion constants $\{\underline{p}_i\}$ or $\{\alpha_i\}$ vary significantly among researchers, one could hope that even the riskier projects are tended to. Deeper exploration of such probabilistic analyses is left for future research. Instead, in the following section we conduct an experimental study that sheds some light on the average-case effect of these model modifications.

4 Experiments

To empirically test the our proposed model modification, we conducted experiments in which we repeatedly and independently brought a market of $n = 50$ researchers and $m = 100$ projects to an approximate equilibrium and measure the resulting welfare. We then the measured welfare by the (approximate) optimal welfare and compare the suboptimality ratios obtained by different utility functions, learning algorithms, and instance structures.

4.1 Project Game instance generation

Project importance For $j \in [m]$, we draw the *inverse* project importance importances independently from a Beta$(3, 3)$ distribution. Using the Beta distribution is convenient here because it allows both w_j and $1/w_j$ to have well-behaved densities and moments.

Success probabilities When drawing the project success probability, we consider three different "researcher types", as described below.

- **Identical researchers** We let $p_{i,j} = p_j$ for every researcher i and project j. For $j \in [m]$, we draw the project success probability p_j from Beta$(\frac{1}{w_j}, 1 - \frac{1}{w_j})$, independently from each project. This implies that the average success probability is exactly 0.5 (the expectation of $1/w_j$). Using a Beta distribution with parameter less than 1 for the probabilities implies that the expected reward $w_j p_j$ has a wide spread, and therefore that the generated game instances have few very lucrative projects over which researchers will compete.

- **Ability-based researchers** We let $p_{i,j} = \min\{2a_i p_j, 0.95\}$, where p_j is drawn as above and a_i is an ability factor drawn from Beta$(2, 2)$ independently for each researcher. This setting of probabilities keeps the mean success probability close to its previous value of $1/2$, and but creates a more realistic problem instance with non-identical researchers.

- **Independent researchers** We draw $p_{i,j} \sim$ Beta$(\frac{1}{w_j}, 1 - \frac{1}{w_j})$ independently for every $i \in [n]$ and $j \in [m]$.

Risk thresholds When the risk threshold described in Section 3.2 is used, we draw for researcher i a threshold $\underline{p}_i$ defined as the qth percentile of $p_{i,1}, p_{i,2}, \ldots, p_{i,m}$, here q is drawn from Beta$(2, 2)$ independently for every researcher. If a variance penalty of Section 3.2 is used instead, we let $\alpha_i = \sqrt{\underline{p}_i^{-1} - 1}$, with $\underline{p}_i$ drawn as before, so that it induces the same effective risk threshold.

4.2 Approximately optimizing the social welfare

Finding the allocation with maximum social welfare is NP-hard in general and also for ability-based researchers [9]. Instead, we approximate it with a greedy algorithm. The algorithm takes as input project importances $\{w_j\}_{j\in[m]}$ and success probabilities $\{p_{i,j}\}_{i\in[n],j\in[m]}$ and sequentially assigns researchers to projects by repeating the following steps steps n times:

1. For each unassigned researcher i and each project j compute the increase to the social welfare gained by assigning researcher i to project j, which equals

$$w_j p_{i,j} \cdot (1 - \mathbb{P}(\text{project solved by researchers already assigned to it})).$$

2. Find the pair $i^\star, j^\star$ of researcher and project for which is the computed increase is maximal.

3. Assign $i^\star$ to $j^\star$.

As Kleinberg and Oren [9] note, a greedy approach suffices to find the optimal allocation exactly when players are identical. We state this formally in the following.

Theorem 6. *For identical researchers, the greedy algorithm finds the maximum social welfare.*

A formal proof of this can be found in Appendix A.3.

For non-identical researchers, we conducted small scale tests with up to 4 researcher and 8 projects, where we found the optimal allocation via brute-force enumeration and compared it to the greedy solution. In the majority of the tests, the greedy algorithm achieved social welfare within 10% of the optimum.

4.3 No-regret solvers

To simulate the decisions of real-life researchers, we implemented three types of no-regret dynamics. The first of these uses Multiplicative Weights (MW) with feedback that equals the exact utility (expected credit). More precisely, at each round each player observes the utility they'd have obtained for any possible choice of projects, given the projects chosen by the other agents in this round. This can be thought of as emulating a mentor's supervision, which provides more knowledge of success probabilities than the researcher has by herself.

The second type uses MW algorithm with stochastic feedback formed by simulating the outcomes of every (hypothetical) project choice and allocating credit accordingly. Such stochastic feedback is natural, as all the utilities we consider are defined in terms of payout in each possible outcome. These feedback vectors equal the utility in expectation, and therefore using them with MW constitutes no-regret dynamics. This scenario is slightly more realistic, as real-life research outcomes are stochastic, but it still allows researchers to learn from the outcomes of all possible choice instead of just the choice that was actually made.

In the third and most realistic dynamics, researchers are given stochastic bandit feedback, i.e. they only observe the (random) amount of credit obtained from their chosen project each round, given the choices made by the other researchers. To obtain no-regret dynamics, we employ the Exp3 adversarial bandit algorithm [6].

We describe the MW and Exp3 solvers in additional detail in Appendix B.

Convergence criterion For each algorithm, we conduct the following convergence test once every 500 steps. Letting T denote the number of steps run so far, we compute for each player i the regret $r_i := \max_j \frac{2}{T} \sum_{t=T/2}^{T} \left(u_i(j, x_{\setminus i}^t) - u_i(x^t) \right)$, where u_i is player i's utility, x^t is the allocation (chosen projects) at step t, and $u_i(j, x_{\setminus i}^t)$ is i's utility when choosing project j instead of x_i^t. We declare convergence if $\sum_{i \in [n]} \max\{r_i, 0\} \leq 10^{-3} \cdot \tilde{W}_{\text{opt}}$, where $\tilde{W}_{\text{opt}}$ is the approximate optimal social welfare obtained from the greedy algorithm. We halt the solver whenever convergence is declared, or after 10^5 steps.

4.4 Experiment protocol

Each experiment run is defined by the 5-tuple (seed, researcher, credit, penalty, solver). We performed an experiment for each of the 27 000 tuples in the Cartesian product of the following sets:

- seed $\in \{100, \ldots, 349\}$ (250 different seeds).
- researcher $\in$ {identical, ability-based, independent}, as defined in Section 4.1.

- credit $\in$ {equal split (1), credit replication as in Section 3.1}.
- penalty $\in$ {none, scoop 0.05, scoop 0.1, scoop 0.2, risk threshold, variance}. Here "scoop x" means the scoop penalty (2) with coefficients as in (9) with $c = x$ and $\epsilon = 0.01$. The "risk threshold" and "variance" penalties are described in Section 3.2, and the values of the thresholds are given in Section 4.1.
- solver $\in$ {MW w/ expectations, Stochastic MW, Exp3} as described in Section 4.3.

For a given tuple (seed, researcher, credit, penalty, solver), the experiment proceeds as follows.

1. Set the random seed to seed.
2. Draw a Project Game instance with success probability distribution given by researcher.
3. Compute the approximate optimal social welfare $\tilde{W}_{\text{opt}}$ using the greedy algorithm described in Section 4.2.
4. With the utility function defined by credit and penalty, run solver as described in Section 4.3.
5. Let W_{emp} be the average social welfare in the last 100 learning iterations; output $W_{\text{emp}}/\tilde{W}_{\text{opt}}$.

4.5 Results

We summarize the results of our experiments by displaying boxplots of the distribution $W_{\text{emp}}/\tilde{W}_{\text{opt}}$ over the 250 values of seed, for each tuple (researcher, credit, penalty, solver); Figure 2 shows results for "equal split" credit allocation (1), while Figure 3 shows result for the credit replication scheme we consider in Section 3.1.

The scoop penalty improves the quality of equilibira across the board. For equal credit, we see improvements of roughly 10%, and for credit replications the improvements are even more dramatic. The coefficient $c = 0.2$ gives the best results for credit replication and equal split with identical researcher, while $c = 0.1$ performs slightly better for equal split and ability-base or independent researchers.

Figure 3 shows that, as expected, the credit replication scheme can lead to extremely poor equilibria in general and this effect lessens as researchers' abilities become more disparate. It will be interesting to explore (theoretically and experimentally) whether scoop penalties with even greater constants c can improve the situation even further.

A somewhat surprising result observable in both Figure 3 and 2 is the very high similarity of equilibria achieved by all three solvers. The only noticeable effect is the slightly higher PoA achieved in some cases by Exp3 which results from imperfect convergence of the algorithm; approximately 50% of the Exp3 trials converged before the 100000 iteration cutoff, as opposed to over 99% of the trials involving the other solvers. However, the welfare in the non-convergent trials was not significantly worse that that of the convergent ones. This result underscores the robustness of the scoop penalty to different learning dynamics.

It is also worth noting that while never as good as a well-calibrated scoop penalty, the risk threshold and variance penalty also reduce PoA when credit replication is used. With the equal-split scheme, there is an interesting inversion in which the variance penalty becomes more effective. In a few cases—namely with ability-based success probabilities—the threshold and variance penalties increase PoA. This is perhaps due to misalignment between researcher's self-perceived risk and their optimal role in the research market, leading to some researchers choosing sub-optimal research projects. It should also be noted that no attempt was made to tune or otherwise judiciously choose the distribution of the risk/variance penalty parameters.

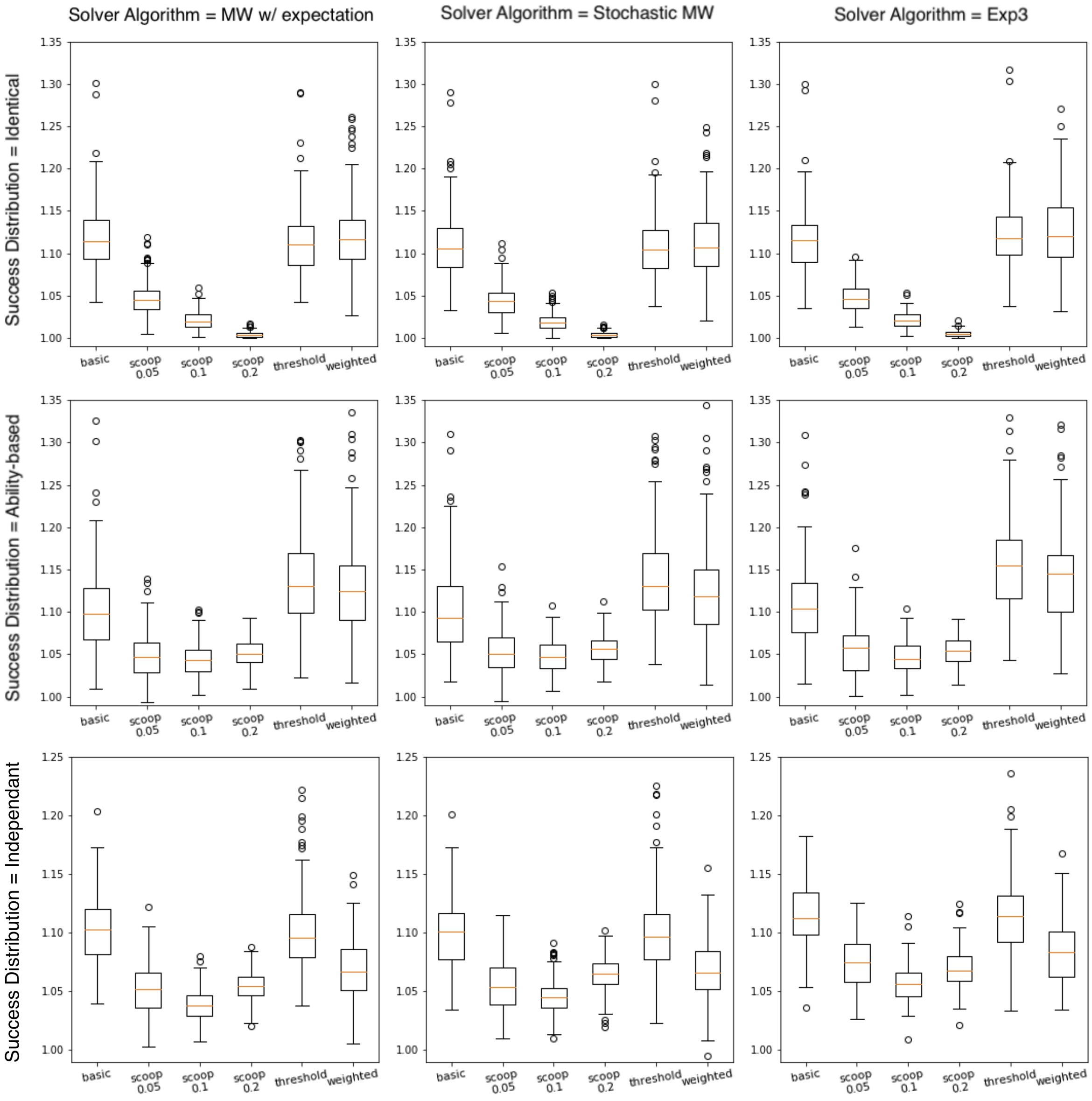

Figure 2: A comparison of PoA for different penalties with **equal split** credit allocation. Rows vary by success probability distribution type and columns by solver algorithm. Here "basic" denotes the case with no penalty and "weighted" denotes the variance penalty.

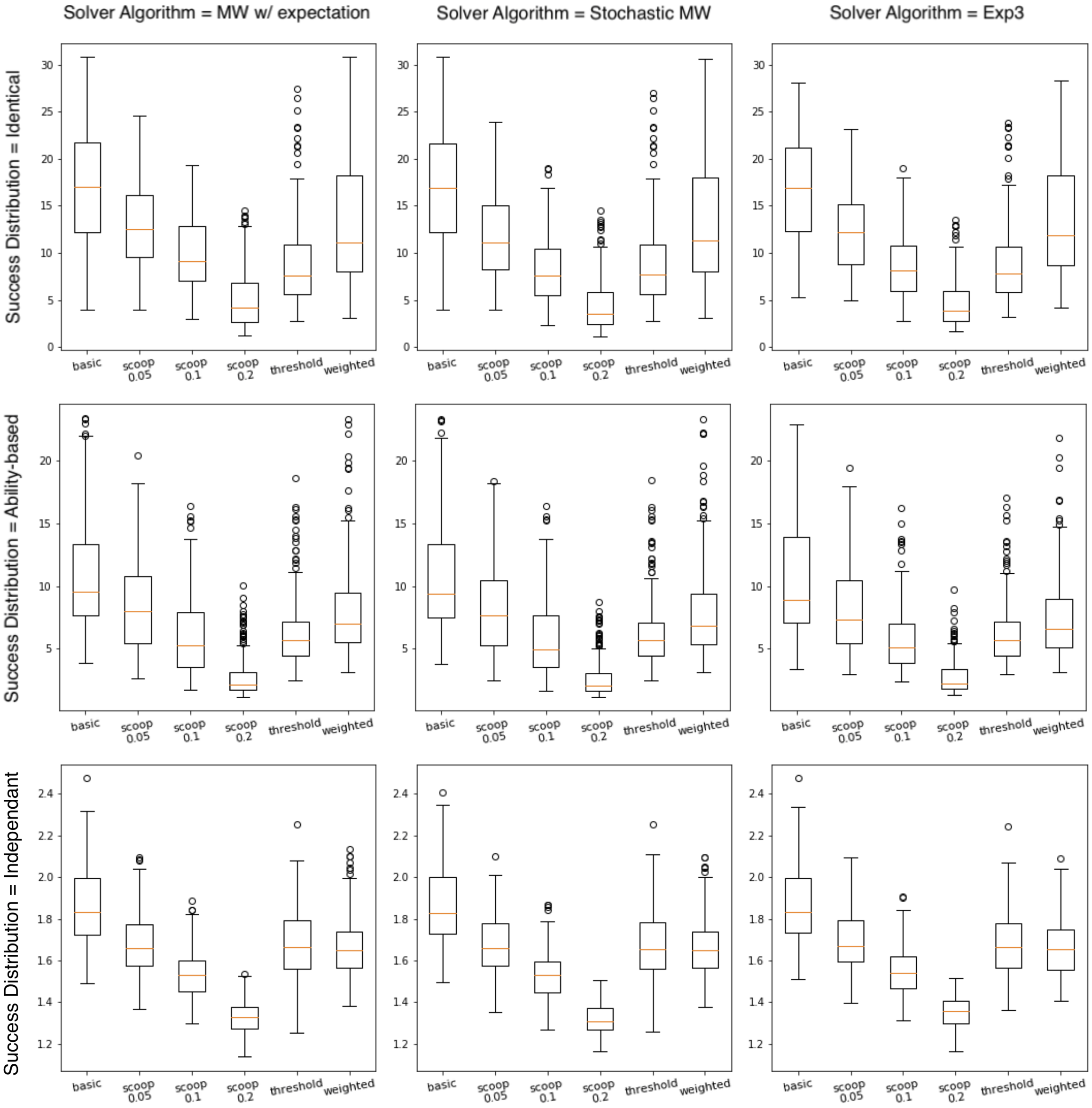

Figure 3: A comparison of PoA for different penalties given **credit replication**. Rows vary by success probability distribution type and columns by solver algorithm. Here “basic” denotes the case with no penalty and “weighted” denotes the variance penalty.

Appendices

A Proofs

A.1 Proof of Theorem 1

Theorem 1. *Let x, y be two strategies for the Project Game with Identical Players. If x is a Nash equilibrium for utility (3), then*

$$W\left(x\right) \geq \frac{W\left(y\right)}{1+\gamma\left(x,y\right)},$$

where

$$\gamma\left(x,y\right) := \max_{j\in[m]} \left\{ \max\left\{ \frac{W\left[Y_j\right]}{W\left[X_j\right]} - \frac{Y_j}{X_j} - \rho_j\left(X_j - Y_j\right)\frac{W\left[X_j-1\right]}{W\left[X_j\right]}, \rho_j\left(Y_j - X_j\right) \right\} \right\}. \tag{4}$$

Proof. Our first step is to permute the strategy y so that it is maximally aligned, i.e. for every project j , either $\{i \mid x_i = j\} \subseteq \{i \mid y_i = j\}$ or $\{i \mid y_i = j\} \subseteq \{i \mid x_i = j\}$; since players are identical, such permutation does not change $W\left(y\right)$. The permutation allows us to write the "entangled utility" in a simple form

$$\sum_{i\in[n]} \tilde{u}_i\left(y_i, x_{\setminus i}\right) = \sum_{j\in[m]} \Big(\min\left\{X_j, Y_j\right\} \tilde{U}_j\left[X_j\right] + \left[Y_j - X_j\right]_+ \tilde{U}_j\left[X_j+1\right] \Big),$$

where $\left[z\right]_+ = \max\left\{z, 0\right\}$ and $\tilde{U}_j\left[k\right] = W_j\left[k\right]/k - \rho_j W_j\left[k-1\right]$ is the per-player penalized utility from project j, when k players choose it. The above equality holds because there are exactly $\min\left\{X_j, Y_j\right\}$ players that choose project j in both y and x, and $\left[Y_j - X_j\right]_+$ players that choose project j in y but not in X. Consider a project j for which $Y_j > X_j$, the submodularity of $W_j\left[\cdot\right]$ gives

$$\left(Y_j - X_j\right)\frac{W_j\left[X_j+1\right]}{X_{j+1}} \geq \left(Y_j - X_j\right)\left(W_j\left[X_j+1\right] - W_j\left[X_j\right]\right) \geq W_j\left[Y_j\right] - W_j\left[X_j\right].$$

Letting

$$S_> = \left\{j \in [m] \mid Y_j > X_j\right\} \text{ and } S_\leq = \left\{j \in [m] \mid Y_j \leq X_j\right\},$$

we have

$$\begin{aligned}
\sum_{j\in S_>} \Big(\min\left\{X_j, Y_j\right\} \tilde{U}_j\left[X_j\right] + \left[Y_j - X_j\right]_+ \tilde{U}_j\left[X_j+1\right] \Big) &= \sum_{j\in S_>} \Big(X_j \tilde{U}_j\left[X_j\right] + \left(Y_j - X_j\right) \tilde{U}_j\left[X_j+1\right] \Big) \\
&\geq \sum_{j\in S_>} \Big(X_j \tilde{U}_j\left[X_j\right] + W_j\left[Y_j\right] - W_j\left[X_j\right] - \rho_j\left(Y_j - X_j\right) W_j\left[X_j\right] \Big) \\
&= \sum_{j\in S_>} \left(W_j\left[Y_j\right] - \rho_j\left(Y_j - X_j\right) W_j\left[X_j\right] - \rho_j X_j W_j\left[X_j - 1\right] \right),
\end{aligned}$$

and

$$\begin{aligned}
\sum_{j\in S_\leq} \Big(\min\left\{X_j, Y_j\right\} \tilde{U}_j\left[X_j\right] + \left[Y_j - X_j\right]_+ \tilde{U}_j\left[X_j+1\right] \Big) &= \sum_{j\in S_\leq} Y_j \tilde{U}_j\left[X_j\right] \\
&= \sum_{j\in S_>} \left(\frac{Y_j}{X_j} W_j\left[X_j\right] - \rho_j Y_j W_j\left[X_j - 1\right] \right).
\end{aligned}$$

Combining the two expressions and rearranging, we have

$$\begin{aligned}\sum_{i\in[n]} \tilde{u}_i\left(y_i, x_{\setminus i}\right) &\geq W\left(y\right) - \sum_{j\in S_{\leq}} \left(\frac{W\left[Y_j\right]}{W\left[X_j\right]} - \frac{Y_j}{X_j} - \rho_j\left(X_j - Y_j\right)\frac{W\left[X_j-1\right]}{W\left[X_j\right]}\right) W_j\left[X_j\right] \\ &\quad - \sum_{j\in S_{>}} \rho_j\left(Y_j - X_j\right) W_j\left[X_j\right] - \sum_{j\in[m]} \rho_j X_j W_j\left[X_j - 1\right] \\ &\geq W\left(y\right) - \gamma\left(x,y\right) W\left(x\right) - \sum_{j\in[m]} \rho_j X_j W_j\left[X_j - 1\right],\end{aligned}$$

with $\gamma\left(x,y\right)$ defined in (4). To complete the proof we simply use the fact that x is a NE to write

$$\sum_{i\in[n]} \tilde{u}_i\left(y_i, x_{\setminus i}\right) \leq \sum_{i\in[n]} \tilde{u}_i\left(x\right) = W\left(x\right) - \sum_{j\in[m]} \rho_j X_j W_j\left[X_j - 1\right].$$

Combining our upper and lower bounds on $\sum_{i\in[n]} \tilde{u}_i\left(y_i, x_{\setminus i}\right)$ and rearranging gives the result. □

A.2 Proof of Corollary 5

Corollary 5. *In the Project Game with penalized utility* (2) *and valid base utility* u_i*, if*

$$\rho_{i,j} = c \cdot \log \frac{1}{\max\{\epsilon, 1 - p_{i,j}\}} \tag{9}$$

for some $c, \epsilon \in (0,1)$*, then the robust price of anarchy is at most*

$$\frac{2 + c \cdot \log \frac{1}{\epsilon}}{1 - \sqrt{\epsilon}}.$$

For $c = 0.05$ *and* $\epsilon = 0.01$ *(as in Figure 1), the robust price of anarchy is at most* $2.4781 \leq 2.5$*.*

Proof. As in the proof of Corollary 3, we start with an optimal policy $y^\star$ and modify it by trimming "excess researchers". Fixing some $\epsilon' \in (0,1)$, we formally define our modified policy y through its inverse Y as

$$Y_j = \underset{S \subseteq Y_j^\star \text{ s.t. } W_j(S) \geq (1-\epsilon') W_j(Y_j^\star)}{\operatorname{argmin}} |S|.$$

This defines a valid assignment y as by construction $Y_j \subseteq Y_j^\star$ for every $j \in [m]$. We further have that by construction $W(y) \geq (1 - \epsilon')$.

We now proceed to bound $\beta(y) = \max_{j\in[m]} \sum_{i\in Y_j} \rho_{i,j}$. To do so, we fix $j \in [m]$ and consider separately the cases $|Y_j| = 1$ and $|Y_j| > 1$. First, if $|Y_j| = 1$, so that $Y_j = \{i'\}$ for some $i' \in [n]$, we simply have

$$\sum_{i\in Y_j} \rho_{i,j} = \rho_{i',j} \leq c \log \frac{1}{\epsilon} \tag{11}$$

by the definition (9) of $\rho_{i,j}$. Second, if $|Y_j| > 1$, we have by construction of Y_j that for every $S \subset Y_j$, $W_j(S) < (1 - \epsilon') W_j(Y_j^\star) \leq (1 - \epsilon') w_j$. Now, using $W_j(S) = w_j \left(1 - e^{-\sum_{i\in S} \log \frac{1}{1-p_{i,j}}}\right)$, we have that $W_j(S) < (1 - \epsilon') w_j$ implies

$$\sum_{i\in S} \log \frac{1}{1 - p_{i,j}} \leq \log \frac{1}{\epsilon'}.$$

Summing this inequality over every $S = Y_j \setminus \{i\}$, $i \in Y_j$, gives

$$(|Y_j| - 1) \sum_{i \in Y_j} \log \frac{1}{1 - p_{i,j}} = \sum_{i \in Y_j} \sum_{i' \in Y_j \setminus \{i\}} \log \frac{1}{1 - p_{i,j}} \leq |Y_j| \log \frac{1}{\epsilon'}.$$

Using again the definition (9) of $\rho_{i,j}$, we have

$$\sum_{i \in Y_j} \rho_{i,j} \leq c \sum_{i \in Y_j} \log \frac{1}{1 - p_{i,j}} \leq \frac{|Y_j|}{|Y_j| - 1} c \log \frac{1}{\epsilon'} \leq 2c \cdot \log \frac{1}{\epsilon'}. \tag{12}$$

Therefore, with $\epsilon' = \sqrt{\epsilon}$, the bounds (11) and (12) together guarantee that $\beta(y) \leq c \log \frac{1}{\epsilon}$, and therefore

$$(2 + \beta(y)) \frac{W(y^\star)}{W(y)} \leq \frac{2 + c \cdot \log \frac{1}{\epsilon}}{1 - \sqrt{\epsilon}},$$

and so the Corollary follows from Theorem 4. □

A.3 Proof of Theorem 6

Theorem 6. *For identical researchers, the greedy algorithm finds the maximum social welfare.*

Proof. We will prove by induction that we never eliminate an optimal allocation.

For the base case, we know that there exists at least one optimal allocation that maximizes social welfare. This cannot have been eliminated before we start assigning researchers.

Assume that the algorithm worked until the kth step. Say we assign researcher i to project j. Case 1: if we look at some optimal, we immediately see that there are more people assigned to j than we have so far. Then we can assign i to j immediately, swapping i with some extra researcher assigned to j in the optimal. We know this is still optimal, since relabeling doesn't affect the distribution when the researchers are identical.

Case 2: there are no extra researchers assigned to project j. Assume the contribution of this assignment was c. All other options were at most as good as the assignment we made, so their contribution is $\leq c$. Since, once we assign a researcher to a project the reward for assigning more researchers to that project can only decrease, we know that all assignments made after this point, contributed at most c, regardless of the order they were assigned in. In particular, the contribution for assigning i to wherever it is in the optimal must be at most c. Regardless of what time step i was assigned at, assigning i to j will give contribution c (since no extra researchers were assigned to j in the meantime), and assigning i anywhere else would give contribution at most c. Then, changing i to be assigned to j can only increase the social optimum.

Thus, at each step in the algorithm, we never eliminate all social optimal allocations. Thus, at the end of the algorithm, the allocation generated must be optimal. □

B Solvers in detail

B.1 Multiplicative weights

The Multiplicative Weights algorithm [4] is described by the following procedure for each researcher:

1. Initialize $w^1(r) = 1$ for all research projects $r \in R$
2. For $t = 1, 2, ..., T$:

   (a) Choose a project according to the distribution $p^t := w^t/\Gamma^t$ where $\Gamma^t = \sum_{r \in R} w^t(r)$
   (b) Given cost vector c^t, decrease weights using the formula $w^{t+1}(r) = w^t(r) \cdot (1-\epsilon)^{c^t(r)}$ for every project $r \in R$.

In the non-stochastic case the cost vectors c^t will be calculated as the negative expected utility of each action. In the stochastic case, the cost vectors are sampled negative utilities given each action. We should expect slightly faster convergence in the non-stochastic case as it most accurately reflects the utility of an action.

B.2 Exp3

In the least observed setting, we consider the research market for each individual researcher as an adversarial bandit, which is an extension of the multi-armed bandit model to competitive games. Exp3 is a well-known algorithm for adversarial bandit learning with bounded regret [6]. The procedure for Exp3 is shown below:

1. Initialize $w^1(r) = 1$ for all research projects $r \in R, |R| = m$
2. For $t = 1, 2, ..., T$:
   (a) Set $p^t := (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{m} w_j(t)} + \frac{\gamma}{m}$ for each i.
   (b) Choose a project r_t according to p^t
   (c) Given reward $x^t(r_t)$, update weights using the formula $w^{t+1}(r_t) = w^t(r_t)e^{\frac{\gamma x^t(r_t)}{m p^t(r_t)}}$.
   (d) For all $j \neq r_t$, $w^{t+1}(j) = w^t(j)$.

As in the stochastic case with the Multiplicative Weights solver, the reward for an action is simulated given the chosen action.

References

[1] M. Ackerman and S. Brânzei. Research quality, fairness, and authorship order. *CoRR abs/1208.3391*, 2012.

[2] E. Anshelevich and S. Sekar. Computing stable coalitions: Approximation algorithms for reward sharing. In *International Conference on Web and Internet Economics*, pages 31–45. Springer, 2015.

[3] E. Anshelevich, O. Bhardwaj, and M. Hoefer. Friendship, altruism, and reward sharing in stable matching and contribution games. *arXiv preprint arXiv:1204.5780*, 2012.

[4] S. Arora, E. Hazan, and S. Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(1):121–164, 2012.

[5] K. J. Arrow. Economic welfare and the allocation of resources for invention. In *Readings in Industrial Economics*, pages 219–236. Springer, 1972.

[6] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[7] Y. Bachrach, V. Syrgkanis, and M. Vojnović. Incentives and efficiency in uncertain collaborative environments. In *International Conference on Web and Internet Economics*, pages 26–39. Springer, 2013.

[8] J. G. Foster, A. Rzhetsky, and J. A. Evans. Tradition and innovation in scientists' research strategies. *American Sociological Review*, 80(5):875–908, 2015.

[9] J. Kleinberg and S. Oren. Mechanisms for (mis) allocating scientific credit. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 529–538. ACM, 2011.

[10] Q. Ma, S. Muthukrishnan, B. Thompson, and G. Cormode. Modeling collaboration in academia: A game theoretic approach. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 1177–1182. ACM, 2014.

[11] J. R. Marden and T. Roughgarden. Generalized efficiency bounds in distributed resource allocation. *IEEE Transactions on Automatic Control*, 59(3):571–584, 2014.

[12] R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2(1):65–67, 1973.

[13] T. Roughgarden. Intrinsic robustness of the price of anarchy. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 513–522. ACM, 2009.

[14] A. Rzhetsky, J. G. Foster, I. T. Foster, and J. A. Evans. Choosing experiments to accelerate collective discovery. *Proceedings of the National Academy of Sciences*, 112(47):14569–14574, 2015.

[15] A. Vetta. Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions. In *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, pages 416–425. IEEE, 2002.