Diversified Strategies for Mitigating Adversarial Attacks in Multiagent Systems

Maria-Florina Balcan
Carnegie Mellon University
ninamf@cs.cmu.edu

Avrim Blum
TTI-Chicago
avrim@ttic.edu

Shang-Tse Chen
Georgia Institute of Technology
schen351@gatech.edu

ABSTRACT

In this work we consider online decision-making in settings where players want to guard against possible adversarial attacks or other catastrophic failures. To address this, we propose a solution concept in which players have an additional constraint that at each time step they must play a *diversified* mixed strategy: one that does not put too much weight on any one action. This constraint is motivated by applications such as finance, routing, and resource allocation, where one would like to limit one's exposure to adversarial or catastrophic events while still performing well in typical cases. We explore properties of diversified strategies in both zero-sum and general-sum games, and provide algorithms for minimizing regret within the family of diversified strategies as well as methods for using taxes or fees to guide standard regret-minimizing players towards diversified strategies. We also analyze equilibria produced by diversified strategies in general-sum games. We show that surprisingly, requiring diversification can actually lead to higher-welfare equilibria, and give strong guarantees on both price of anarchy and the social welfare produced by regret-minimizing diversified agents. We additionally give algorithms for finding optimal diversified strategies in distributed settings where one must limit communication overhead.

KEYWORDS

Game theory; regret minimization; adversarial multiagent systems; diversified strategies; risk mitigation; general-sum games

ACM Reference Format:
Maria-Florina Balcan, Avrim Blum, and Shang-Tse Chen. 2018. Diversified Strategies for Mitigating Adversarial Attacks in Multiagent Systems. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

A common piece of advice when one needs to make decisions in the face of unknown future events is "Don't put all your eggs in one basket." This is especially important when there can be an adversarial attack or catatrophic failure. We consider game-theoretic problems from this perspective, design online learning algorithms with good performance subject to such exposure-limiting constraints on behavior, and analyze the effects of these constraints on the expected value obtained (in zero-sum games) and the overall social welfare produced (in general-sum games).

As an example, consider a standard game-theoretic scenario: an agent must drive from point A to point B and has n different routes it can take. We could model this as a game M where rows correspond to the n routes, columns correspond to m possible traffic patterns, and entry $M(i,j)$ is the cost for using route i under traffic pattern j. However, suppose the agent is carrying valuable documents and is concerned an adversary might try to steal them. In this case, to reduce the chance of this happening, we might require that no route have more than (say) 10% probability. The agent then wants to minimize expected travel time subject to this requirement. Or in an investment scenario, if rows correspond to different investments and columns to possible market conditions, we might have an additional worry that perhaps one of the investment choices is run by a crook. In this case, we may wish to restrict the strategy space to allocations of funds that are not too concentrated.

To address such scenarios, for $\epsilon \in [\frac{1}{n}, 1]$ let us define a probability distribution (or allocation) P to be *ϵ-diversified* if $P(i) \leq \frac{1}{\epsilon n}$ for all i. For example, for $\epsilon = \frac{1}{n}$ this is no restriction at all, for $\epsilon = 1$ this requires the uniform distribution, and for intermediate values of ϵ this requires an intermediate level of diversification. We then explore properties of such diversified strategies in both zero-sum and general-sum games as well as give algorithmic guarantees.

For zero-sum games, define v_ϵ to be the minimax-optimal value of the game in which the row player is restricted to playing ϵ-diversified mixed strategies. Natural questions we address are: Can one design adaptive learning algorithms that maintain ϵ-diversified distributions and minimize regret within this class so they never perform much worse than v_ϵ? Can a central authority "nudge" a generic non-diversified regret-minimizer into using diversified strategies via fines or taxes (extra loss vectors strategically placed into the event stream) such that it maintains low-regret over the original sequence? And for reasonable games, how much worse is v_ϵ compared to the non-diversified minimax value v? We also consider a dual problem of producing a strategy Q for the column player that achieves value v_ϵ against all but an ϵ fraction of the rows (which an adversary can then aim to attack).

One might ask why not model such an adversary directly within the game, via additional columns that each give a large loss to one of the rows. The main reason is that these would then dominate the minimax value of the game. (And they either would not have values within the usual $[0,1]$ range assumed by regret-minimizing learners, or, if they were scaled to lie in this range, they would cause all other events to seem roughly the same). Instead, we want to consider learning algorithms that optimize for more common events, while keeping to the constraint of maintaining diversified strategies. We also remark that one could also make diversification a soft constraint by adding a loss term for not diversifying.

We next consider general-sum games, such as routing games and atomic congestion games, in which k players interact in ways that lead to various costs being incurred by each player. We show

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

that surprisingly, requiring a player to use diversified strategies can actually improve its performance in equilibria in such games. We then study the ϵ-*diversified price of anarchy*: the ratio of the social cost of the worst equilibrium subject to all players being ϵ-diversified to the social cost of the socially-best set of ϵ-diversified strategies. We show that in some natural games, even requiring a small amount of diversification can dramatically improve the price of anarchy of the game, though we show there also exist games where diversification can make the price of anarchy worse. We also bring several threads of this investigation together by showing that for the class of *smooth games* defined by Roughgarden [25], for any diversification parameter $\epsilon \in [\frac{1}{n}, 1]$, the ϵ-diversified price of anarchy is no worse than the smoothness of the game, and moreover, players using diversified regret-minimizing strategies will indeed approach this bound. Thus, we get strong guarantees on the quality of interactions produced by self-interested diversified play. Finally, we consider how much diversification can hurt *optimal* play, showing that in random unit-demand congestion games, diversification indeed incurs a low penalty.

Lastly, we consider an information-limited, distributed, "big-data" setting in which the number of rows and columns of the matrix M is very large and we do not have it explicitly. Specifically, we assume the n rows are distributed among r processors, and the only access to the matrix M we have is via an oracle for the column player that takes in a sample of rows and outputs the column player's best response. What we show is how in such a setting to produce near optimal strategies for each player in the sense described above, from very limited communication among processors.

In addition to our theoretical results, we also present experimental simulations for both zero-sum and general-sum games.

1.1 Related Work

There has been substantial work on design of "no-regret" learning algorithms for repeated play of zero-sum games [7, 10, 15]. *Multiplicative Weight Update* methods [2, 21] are a specific type of no-regret algorithm that have received considerable attention in game theory [15, 16], machine learning [11, 15], and many other research areas [1, 20], due to their simplicity and elegance.

We consider the additional constraint that players play diversified mixed strategies, motivated by the goal of reducing exposure to adversarial attacks. The concept of diversified strategies, sometimes called "smooth distributions", appears in a range of different areas [12, 17, 20]. [9] considers a somewhat related notion where there is a penalty for deviation from a given fixed strategy, and shows existence of equilibria in such games. Also related is work on adversarial machine learning, e.g., [13, 18, 27]; however, in this work we are instead focused on decision-making scenarios.

Our distributed algorithm is inspired by prior work in distributed machine learning [4, 11, 14], where the key idea is to perform weight updates in a communication efficient way. Other work on the impact of adversaries in general-sum games appears in [3, 5, 6].

2 ZERO-SUM GAMES

We begin by studying two-player zero-sum games. Recall that a two-player zero-sum game is defined by a $n \times m$ matrix M. In each round of the game, the row player chooses a distribution P over the rows of M, and the column player chooses a distribution Q over the columns of M. The expected loss of the row player is

$$M(P, Q) = P^T MQ = \sum_{i,j} P(i)M(i, j)Q(j),$$

where $M(i, j) \in [0, 1]$ is the loss suffered by the row player if the row player plays row i and the column player plays column j. The goal of the row player is to minimize its loss, and the goal of the column player is to maximize this loss. The minimax value v of the game is:

$$v = \min_P \max_Q M(P, Q) = \max_Q \min_P M(P, Q).$$

Algorithm 1 Multiplicative Weights Update algorithm with Restricted Distributions

```
Initialization: Fix a $\gamma \leq \frac{1}{2}$. Set $P^{(1)}$ to be the uniform distribution.
for $t = 1, 2, \ldots, T$ do
    (1) Choose distribution $P^{(t)}$
    (2) Receive the pure strategy $j_t$ for the column player
    (3) Compute the multiplicative update rule
            $\hat{P}_i^{(t+1)} = P_i^{(t)}(1-\gamma)^{M(i,j_t)}/Z^{(t)}$
        where $Z^{(t)} = \sum_i P_i^{(t)}(1-\gamma)^{M(i,j_t)}$ is the normalization factor.
    (4) Project $\hat{P}^{(t+1)}$ into $\mathcal{P}_\epsilon$
            $P^{(t+1)} = \arg\min_{P \in \mathcal{P}_\epsilon} RE(P \parallel \hat{P}^{(t+1)})$
end for
```

2.1 Multiplicative Weights and Diversified Strategies

We now consider row players restricted to only playing diversified distributions, defined as follows.

Definition 2.1. A distribution $\mathbf{p} \in \Delta_n$ is called ϵ-diversified if $\max_i p_i \leq \frac{1}{\epsilon n}$.

Let $\mathcal{P}_\epsilon$ be the set of all ϵ-diversified distributions, and let v_ϵ be the minimax value of the game subject to the row player restricted to playing in $\mathcal{P}_\epsilon$. Note that the range of ϵ is between $1/n$ and 1. It is easy to verify that $\mathcal{P}_\epsilon$ is a convex set. As a result, the minimax theorem applies to $\mathcal{P}_\epsilon$ [26], and we call the minimax value v_ϵ:

$$v_\epsilon = \min_{P \in \mathcal{P}_\epsilon} \max_Q M(P, Q) = \max_Q \min_{P \in \mathcal{P}_\epsilon} M(P, Q).$$

The multiplicative weights update algorithm [19, 21] can be naturally adapted to maintain diversified strategies by projecting its distributions into the class $\mathcal{P}_\epsilon$ if they ever step outside of it. This is shown in Algorithm 1. By adapting the analysis of [19] to this case, we arrive at the following regret bound.

THEOREM 2.2. *For any $0 < \gamma \leq 1/2$ and any positive integer T, Algorithm 1 generates distributions $P^{(1)}, \ldots, P^{(T)} \in \mathcal{P}_\epsilon$ to responses $j_1, \ldots, j_T$, such that for any $P \in \mathcal{P}_\epsilon$,*

$$\textstyle\sum_{t=1}^T M(P^{(t)}, j_t) \leq (1+\gamma) \sum_{t=1}^T M(P, j_t) + \frac{RE(P \| P^{(1)})}{\gamma},$$

where $RE(p \parallel q) = \sum_i p_i \ln(p_i/q_i)$ is relative entropy.

By combining Algorithm 1 with a best-response oracle for the column player, and applying Theorem 2.2 and a standard argument [2, 16] we have:

Theorem 2.3. *Running Algorithm 1 for T steps against a best-response oracle, one can construct mixed strategies $\bar{P}$ and $\bar{Q}$ s.t.*

$$\max_Q M(\bar{P}, Q) \le v_\epsilon + \Delta_T \quad \textit{and} \quad \min_{P \in \mathcal{P}_\epsilon} M(P, \bar{Q}) \ge v_\epsilon - \Delta_T,$$

for $\Delta_T = 2\sqrt{\frac{\ln(1/\epsilon)}{T}}$, *where* $\bar{P} = \frac{1}{T}\sum_{t=1}^T P^{(t)}$ *and* $\bar{Q} = \frac{1}{T}\sum_{t=1}^T j_t$.

Proof. We can sandwich the desired inequalities inside a proof of the minimax theorem as follows:

$$\begin{aligned}
\min_{P\in\mathcal{P}_\epsilon} \max_Q M(P,Q) &\le \max_Q M(\bar{P},Q) = \max_Q \frac{1}{T}\sum_{t=1}^T M(P^{(t)},Q) \\
&\le \frac{1}{T}\sum_{t=1}^T \max_Q M(P^{(t)},Q) = \frac{1}{T}\sum_{t=1}^T M(P^{(t)},j_t) \\
&\le \min_{P\in\mathcal{P}_\epsilon} \frac{1+\gamma}{T}\sum_{t=1}^T M(P,j_t) + \frac{\ln(1/\epsilon)}{\gamma T} \\
&\le \min_{P\in\mathcal{P}_\epsilon} M(P,\bar{Q}) + \gamma + \frac{\ln(1/\epsilon)}{\gamma T} \\
&\le \max_Q \min_{P\in\mathcal{P}_\epsilon} M(P,Q) + \gamma + \frac{\ln(1/\epsilon)}{\gamma T}
\end{aligned}$$

If we set $\gamma = \sqrt{\frac{\ln(1/\epsilon)}{T}}$, then $\Delta_T = \gamma + \frac{\ln(1/\epsilon)}{\gamma T} = 2\sqrt{\frac{\ln(1/\epsilon)}{T}}$. The two inequalities in the theorem follow by skipping the first and the last inequalities from the proof above, respectively. □

The next theorem shows that the distribution $\bar{Q}$ in Theorem 2.3 is also a good mixed strategy for the column player against any row-player strategy if we remove a small fraction of the rows.

Theorem 2.4. *By running Algorithm 1 for T steps against a best-response oracle, we can construct a mixed strategy $\bar{Q}$ such that for all but an ϵ fraction of the rows i, $M(i,\bar{Q}) \ge v_\epsilon - \gamma$. Moreover we can do this with at most $T = O\left(\frac{\log(1/\epsilon)}{\gamma^2(1+\gamma-v_\epsilon)}\right)$ oracle calls.*

Proof. We generate distributions $P^{(1)}, \dots, P^{(T)} \in \mathcal{P}_\epsilon$ by using Algorithm 1. Let j_t be the column returned by the oracle with the input $P^{(t)}$. After $T = \left\lceil \frac{\log(1/\epsilon)}{\gamma^2(1+\gamma-v_\epsilon)} \right\rceil + 1$ rounds, we set the mixed strategy $\bar{Q} = \frac{1}{T}\sum_{t=1}^T j_t$. Set $E = \{i | M(i,\bar{Q}) < v_\epsilon - \gamma\}$. Suppose for contradiction that $|E| \ge \epsilon n$. Let $P = u_E$, the uniform distribution on E and 0 elsewhere. It is easy to see that $u_E \in \mathcal{P}_\epsilon$, since $|E| \ge \epsilon n$.

By the assumption of the oracle, we have $v_\epsilon T \le \sum_{t=1}^T M(P^{(t)}, j_t)$. In addition, by Theorem 2.2, we have

$$\sum_{t=1}^T M(P^{(t)}, j_t) \le (1+\gamma)\sum_{t=1}^T M(P, j_t) + \frac{RE(P \parallel P^{(1)})}{\gamma}.$$

For any $i \in E$, $\sum_{t=1}^T M(i, j_t) = T \cdot M(i,\bar{Q}) < (v_\epsilon - \gamma)T$. Since P is the uniform distribution on E, we have $\sum_{t=1}^T M(P, j_t) < (v_\epsilon - \gamma)T$. Furthermore, since $|E| \ge \epsilon n$, we have

$$RE(P \parallel P^{(1)}) = RE(u_E \parallel u) \le \ln(1/\epsilon).$$

Putting these facts together, we get $v_\epsilon T \le (1+\gamma)(v_\epsilon - \gamma)T + \frac{\ln(1/\epsilon)}{\gamma}$, which implies $T \le \frac{\ln(1/\epsilon)}{\gamma^2(1+\gamma-v_\epsilon)}$, a contradiction. □

Algorithm 2 Multiplicative Weights Update algorithm with Interventions

```
Initialization: Fix a $\gamma \le \frac{1}{2}$. Set $P^{(1)}$ to be the uniform distribution.
for $t = 1, 2, \dots, T$ do
    (1) Choose distribution $P^{(t)}$
    (2) Receive the pure strategy $j_t$ for the column player
    (3) Compute the multiplicative update rule
            $P_i^{(t+1)} = P_i^{(t)}(1-\gamma)^{M(i,j_t)}/Z^{(t)}$
        where $Z^{(t)} = \sum_i P_i^{(t)}(1-\gamma)^{M(i,j_t)}$ is the normalization factor.
    (4) While $P^{(t+1)}$ is not $(1-\gamma)\epsilon$-diversified, run multiplicative update (Step 3) on fake loss vector $\ell$ defined as:
            $\ell_i = \begin{cases} 1 & \text{if } P_i^{(t+1)} > \frac{1}{(1-\gamma)\epsilon n} \\ 0 & \text{if } P_i^{(t+1)} \le \frac{1}{(1-\gamma)\epsilon n} \end{cases}$
end for
```

2.2 Diversifying Dynamics

Theorem 2.2 shows that it is possible for a player to maintain an ϵ-diversified distribution at all times while achieving low regret with respect to the entire family $\mathcal{P}_\epsilon$ of ϵ-diversified distributions. However, suppose a player, who is allocating an investment portfolio among n investments, does not recognize the need for maintaining a diversified distribution and simply uses the standard multiplicative-weights algorithm to minimize regret. For example, the player might not realize that the matrix M only represents "typical" behavior of investments, and that a crooked portfolio manager or clever hacker could cause an entire investment to be wiped out. This player might quickly reach a dangerous non-diversified portfolio in which nearly all of its weight is just on one row.

Suppose, however, that an investment advisor or helpful authority has the ability to charge fees on actions whose weights are too high, that can be viewed as inserting fake loss vectors into the stream of loss vectors observed by the player's algorithm. We show here that by doing so in an appropriate manner, this advisor or authority can ensure that the player both (a) maintains diversified distributions, and (b) incurs low regret with respect to the family $\mathcal{P}_\epsilon$ over the sequence of real loss vectors. Viewed another way, this can be thought of as an alternative to Algorithm 1 with slightly weaker guarantees but that does not require the projection. The algorithm remains efficient.

Theorem 2.5. *Algorithm 2 generates distributions $P^{(1)}, \dots, P^{(T)}$ such that*

(a) $P^{(t)} \in \mathcal{P}_{(1-\gamma)\epsilon}$ for all t, and

(b) for any $P \in \mathcal{P}_\epsilon$ we have

$$\sum_{t=1}^T M(P^{(t)}, j_t) \le (1+\gamma)\sum_{t=1}^T M(P, j_t) + \frac{RE(P \parallel P^{(1)})}{\gamma}.$$

Proof. For part (a) we just need to show that the while loop in Step 4 of the algorithm halts after a finite number of loops. To show this, we show that each time a fake loss vector is applied, the gap between the maximum and minimum total losses (including both actual losses and fake losses) over the rows i is reduced. In particular, the multiplicative-weights algorithm has the property that the probability on an action i is proportional to $(1-\gamma)^{L^i_{total}}$ where L^i_{total} is the total loss (actual plus fake) on action i so far; so, the actions of highest probability are also the actions of lowest total loss. This means that in Step 4, there exists some threshold τ such that $\ell_i = 1$ for all i of total loss at most τ and $\ell_i = 0$ for all i of total loss greater than τ. Since we are adding 1 to those actions of total loss at most τ, this means that the gap between the maximum and minimum total loss over all the actions is decreasing, so long as that gap was greater than 1. However, note that if $P^{(t+1)}$ is not $(1-\gamma)\epsilon$-diversified then the gap between maximum and minimum total loss must be greater than 1, by definition of the update rule and using the fact that $\epsilon \le 1$. Therefore, the gap between maximum and minimum total loss is strictly reduced on each iteration (and reduced by at least 1 if any row is ever updated twice) until $P^{(t+1)}$ becomes $(1-\gamma)\epsilon$-diversified.

For part (b), define $L^{alg}_{actual} = \sum_{t=1}^{T} M(P^{(t)}, j_t)$ to be the actual loss of the algorithm and define $L^P_{actual} = \sum_{t=1}^{T} M(P, j_t)$ to be the actual loss of some ϵ-diversified distribution P. We wish to show that L^{alg}_{actual} is not too much larger than L^P_{actual}. To do so, we begin with the fact that, by the usual multiplicative weights analysis, the algorithm has low regret with respect to any fixed strategy over the entire sequence of loss vectors (actual and fake). Say the algorithm's total loss is $L^{alg}_{total} = L^{alg}_{actual} + L^{alg}_{fake}$ and the total loss of P is $L^P_{total} = L^P_{actual} + L^P_{fake}$. We know that

$$L^{alg}_{total} \le (1+\gamma)L^P_{total} + \frac{RE(P\|P^{(1)})}{\gamma},$$

which we can rewrite as:

$$L^{alg}_{actual} + L^{alg}_{fake} \le (1+\gamma)L^P_{actual} + (1+\gamma)L^P_{fake} + \frac{RE(P\|P^{(1)})}{\gamma}.$$

Thus, to prove part (b) it suffices to show that $L^{alg}_{fake} \ge (1+\gamma)L^P_{fake}$. But notice that on each fake loss vector, for each index i such that $\ell_i = 1$, the algorithm has strictly more than $\frac{1}{(1-\gamma)\epsilon n} > \frac{1+\gamma}{\epsilon n}$ probability mass on row i. In contrast, P has at most $\frac{1}{\epsilon n}$ probability mass on row i, since P is ϵ-diversified. Therefore $L^{alg}_{fake} \ge (1+\gamma)L^P_{fake}$ and the proof is complete. □

This analysis can be extended to the case of an advisor who only periodically monitors the player's strategy. If the advisor monitors the strategy every k steps, then in the meantime the maximum probability that any row i can reach is $\frac{1}{(1-\gamma)^k \epsilon n}$. So, part (a) of Theorem 2.5 would need to be relaxed to $P^{(t)} \in \mathcal{P}_{(1-\gamma)^k\epsilon}$. However, part (b) of Theorem 2.5 holds as is.

2.3 How close is v_ϵ to v?

Restricting the row player to play ϵ-diversified strategies can of course increase its minimax loss, i.e., $v_\epsilon \ge v$. In fact, it is not hard to give examples of games where the gap is quite large. For example, suppose the row player has one action that always incurs loss 0, and the remaining $n-1$ actions always incur loss 1 (whatever the column player does). Then $v = 0$ but for $\epsilon \in [\frac{1}{n}, 1]$, $v_\epsilon = 1 - \frac{1}{\epsilon n}$.

However, we show here that for *random* matrices M, the gap between the two is quite small. I.e., the additional loss incurred due to requiring diversification is low. A related result, in a somewhat different model, appears in [22].

Theorem 2.6. *Consider a random $n \times n$ game M where each entry $M(i,j)$ is drawn i.i.d. from some distribution D over* $[0,1]$. *With probability $\ge 1 - \frac{1}{n}$, for any $\epsilon \le 1$, we have $v_\epsilon - v = O\left(\sqrt{\frac{\log n}{n}}\right)$.*

Proof. Let $\mu = \mathbf{E}_{x\sim D}[x]$ be the mean of distribution D. We will show that v and v_ϵ are both close to μ. To argue this, we will examine the value of the uniform distribution P_{unif} for the row player, and the value of the uniform distribution Q_{unif} for the column player. In particular, notice that $v \ge \min_i M(i, Q_{\text{unif}})$ because Q_{unif} is just one possible strategy for the column player, and by definition, $v = \min_i M(i, Q^*)$ where Q^* is the minimax optimal strategy for the column player, and the row player's loss under Q^* is greater than or equal to the row player's loss under Q_{unif} since the column player is trying to maximize the row player's loss. Similarly, $v_\epsilon \le \max_j M(P_{\text{unif}}, j)$ since P_{unif} is just one possible diversified strategy for the row player, and by definition $v_\epsilon = \max_j M(P^*, j)$ where P^* is the minimax optimal diversified strategy for the row player and the row player is trying to minimize loss. So, we have

$$\min_i M(i, Q_{\text{unif}}) \le v \le v_\epsilon \le \max_j M(P_{\text{unif}}, j).$$

Thus, if we can show that with high probability $\min_i M(i, Q_{\text{unif}})$ and $\max_j M(P_{\text{unif}}, j)$ are both close to μ, then this will imply that v and v_ϵ are close to each other.

Let us begin with P_{unif}. Notice that $M(P_{\text{unif}}, j)$ is just the average of the entries in the jth column. So, by Hoeffding bounds, there exists a constant c such that for any given column j,

$$\Pr\left[M(P_{\text{unif}}, j) > \mu + c\sqrt{\frac{\log n}{n}}\right] \le \frac{1}{2n^2},$$

where the probability is over the random draw of M. By the union bound, with probability at least $1 - \frac{1}{2n}$, this inequality holds simultaneously for all columns j. Since P_{unif} is ϵ-diversified, as noted above this implies that $v_\epsilon \le \mu + c\sqrt{(\log n)/n}$ with probability at least $1 - \frac{1}{2n}$.

On the other hand, by the same reasoning, with probability at least $1 - \frac{1}{2n}$ the uniform distribution Q_{unif} for the column player has the property that for all rows i, $M(i, Q_{\text{unif}}) \ge \mu - c\sqrt{\frac{\log n}{n}}$. This implies as noted above that $v \ge \mu - c\sqrt{\frac{\log n}{n}}$. Therefore, with probability at least $1 - \frac{1}{n}$, $v_\epsilon - v \le 2c\sqrt{\frac{\log n}{n}}$ as desired. □

3 GENERAL-SUM GAMES

We now consider k-player general-sum games. Instead of minimax optimality, the natural solution concept now is a Nash equilibrium. We begin by showing that unlike zero-sum games, it is now possible for the payoff of a player at equilibrium to actually be improved by requiring it to play a diversified strategy. This is a bit peculiar because constraining a player is actually helping it.

We then consider the relationship between the social cost at equilibrium and the optimal social cost, when all players are required to use diversified strategies. We call the ratio of these two quantities the *diversified price of anarchy* of the game, in analogy to the usual *price of anarchy* notion when there is no diversification constraint. We show that in some natural games, even requiring a small amount of diversification can significantly improve the price of anarchy of the game, though there also exist games where diversification can make the price of anarchy worse. Finally, we bring several threads of this investigation together by showing that for the class of *smooth games* defined by Roughgarden [25], for any diversification parameter $\epsilon \in [\frac{1}{n}, 1]$, the ϵ-diversified price of anarchy is no worse than the smoothness of the game, and moreover that players using diversified regret-minimizing strategies (such as those in Sections 2.1 and 2.2) will indeed approach this bound.

3.1 The Benefits of Diversification

First, let us formally define the notion of a *Nash equilibrium subject to a (convex) constraint C*, where C could be a constraint such as "the row player must use an ϵ-diversified strategy".

Definition 3.1. A set of mixed strategies $(P_1, \ldots, P_k)$ is a *Nash equilibrium subject to constraint C* if no player can unilaterally deviate to improve its payoff without violating constraint C. We will just call this a *Nash equilibrium* when C is clear from context.

We now consider the case of $k = 2$ players, and examine how requiring the row player to diversify can affect its payoff at equilibrium. For zero-sum games, the value v_ϵ was always no better than the minimax value v of the game, since constraining the row player can never help it. We show here that this is not the case for general-sum games: requiring a player to use a diversified strategy can in some games *improve* its payoff at equilibrium.

Theorem 3.2. *There exist 2-player general-sum games for which a diversification constraint on the row player* lowers *the row player's payoff at equilibrium, and games for which such a constraint* increases *the row player's payoff at equilibrium.*

Proof. Consider the following two bimatrix games (entries here represent payoffs rather than losses):

$$\text{Game } A : \begin{array}{|c|c|}\hline 2,2 & 1,1 \\ \hline 1,1 & 0,0 \\ \hline \end{array} \qquad \text{Game } B : \begin{array}{|c|c|}\hline 1,1 & 3,0 \\ \hline 0,0 & 1,3 \\ \hline \end{array}$$

In Game A, the unique Nash equilibrium has payoff of 2 to each player, and requiring the row player to be diversified strictly lowers both player's payoffs. On the other hand, diversification helps the row player in Game B. Without a diversification constraint, in Game B the row player will play the top row and the column player will therefore play the left column, giving both players a payoff of 1. However, requiring the row-player to put probability $\frac{1}{2}$ on each row will cause the column player to choose the right column, giving the row player a payoff of 2 and the column player a payoff of 1.5. □

Routing games [24] are an interesting class of many-player games where requiring all players to diversify can actually improve the quality of the equilibrium for everyone. An example is Braess' paradox [8] shown in Figure 1. In this example, k players need to travel from s to t and wish to take the cheapest route. Edge costs are given in the figure, where k_e is the number of players using edge e. At Nash equilibrium, all players choose the route s-a-b-t and incur a cost of 2. However, if they must put equal probability on the three routes they can choose from, the expected cost of each player approaches only $\frac{1}{3}(\frac{2}{3}+1)+\frac{1}{3}(\frac{2}{3}+\frac{2}{3})+\frac{1}{3}(1+\frac{2}{3}) = 1+\frac{5}{9}$. Thus, even though from an individual player's perspective, diversification is a restriction that increases robustness at the expense of higher average loss, overall, diversification can actually improve the quality of the resulting equilibrium state. In the next section, we discuss the social cost of diversified equilibria in many-player games in more detail, analyzing what we call the *diversified price of anarchy* as well as the social cost that results from all players using diversified regret-minimizing strategies.

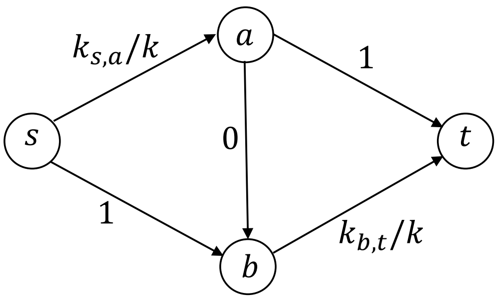


Figure 1: Braess' paradox. Here, k players wish to travel from s to t, and requiring all players to use diversified strategies improves the quality of the equilibrium for everyone.

3.2 The Diversified Price of Anarchy

We now consider structured general-sum games with $k \geq 2$ players. In these games, each player i chooses some strategy s_i from a strategy space $\mathcal{S}_i$. The combined choice of the players $\mathbf{s} = (s_1, \ldots, s_k)$, which we call the *outcome*, determines the cost that each player incurs. Specifically, let $cost_i(\mathbf{s})$ denote the cost incurred by player i under outcome $\mathbf{s}$, and let $cost(\mathbf{s}) = \sum_{i=1}^{k} cost_i(\mathbf{s})$ denote the overall social cost of $\mathbf{s}$. Let $\mathbf{s}^* = \text{argmin}_{\mathbf{s}} cost(\mathbf{s})$, i.e., the outcome of optimum social cost. The *price of anarchy* of a game is defined as the maximum ratio $cost(\mathbf{s})/cost(\mathbf{s}^*)$ over all Nash equilibria $\mathbf{s}$. A low price of anarchy in a game means that all Nash equilibria have social cost that is not too much worse than the optimum. We can analogously define the *ϵ-diversified price of anarchy*:

Definition 3.3. Let $\mathbf{s}_\epsilon^*$ denote the outcome of optimum social cost subject to each player choosing an ϵ-diversified strategy. The *ϵ-diversified price of anarchy* is the maximum ratio $cost(\mathbf{s}_\epsilon)/cost(\mathbf{s}_\epsilon^*)$ over all outcomes $\mathbf{s}_\epsilon$ that are Nash equilibria subject to all players playing ϵ-diversified strategies.

Note that for any game, the 1-diversified price of anarchy equals 1, because players are all required to play the uniform distribution. This suggests that as we increase ϵ, the ϵ-diversified price of anarchy should drop, though as we show, in some games it is not monotone.

Examples. In *consensus games*, each player i is a distinct node in a k-node graph G. Players each choose one of two colors, red or blue, and the cost of player i is the number of neighbors it has of color different from its own. The social cost is the sum of the players' costs, and to keep ratios finite we add 1 to the total. The optimal $\mathbf{s}^*$ is either "all blue" or "all red" in which each player has a cost of 0, so the social cost is 1. However, if the graph is a complete graph minus a matching, then there exists an equilibrium in which half of the players choose red and half the players choose blue. Each player has $\frac{k}{2} - 1$ red neighbors and $\frac{k}{2} - 1$ blue neighbors, so the social cost of this equilibrium is $\Theta(k^2)$. This means the price of anarchy is $\Theta(k^2)$. However, if we require players to play ϵ-diversified strategies for any constant $\epsilon > \frac{1}{2}$ (i.e., they cannot play pure strategies), then for any m-edge graph G, even the optimum outcome has cost $\Omega(m)$ since every edge has a constant probability of contributing to the cost. So the diversified price of anarchy is $O(1)$.

As another example, consider *atomic congestion games* [23]. Here, we have a set R of *resources* (e.g., edges in a graph G) and each player i has a strategy set $\mathcal{S}_i \subseteq 2^R$ (e.g., all ways to select a path between two specified vertices in G). The cost incurred by a player is the sum of the costs of the resources it uses (the cost of its path). Each resource j has a cost function $c_j(k_j)$ where k_j is the number of players who are using resource j. The cost functions c_j could be increasing, such as in packet routing where latency increases with the number of users of an edge, or decreasing, such as players splitting the cost of a shared printer. When examining diversified strategies, we sometimes view players as making fractional choices, such as sending half their packets down one path and half of them down another. The quantity k_j then denotes the total fractional usage of resource j (or equivalently, the expected number of users of that resource).

Non-monotonicity. An example of an atomic congestion game where some diversification can initially increase the price of anarchy is the following. Suppose there are four resources, and each player just needs to choose one of them. The costs of the resources behave as follows:

$$c_1(k_1) = 1, \quad c_2(k_2) = 5, \quad c_3(k_3) = 6/k_3, \quad c_4(k_4) = 6/k_4.$$

Assume the total number of players k is at least 13. The optimal outcome $\mathbf{s}^*$ is for all players to choose resource 3 (or all choose resource 4) for a total social cost of 6. The optimal ϵ-diversified outcome for $\epsilon = \frac{1}{2}$ (i.e., each player can put weight at most $\frac{1}{2}$ on any given resource) is for all players to put half their weight on strategy 3 and half their weight on strategy 4, for a total cost of 12. The worst Nash equilibrium is for all players to choose strategy 1, for a total cost of k, giving a price of anarchy of $k/6$. However if we require players to be ϵ-diversified for $\epsilon = \frac{1}{2}$, there is now a worse equilibrium where each player puts half its weight on strategy 1 and half its weight on strategy 2, for a total cost of $3k$ and a diversified price of anarchy of $3k/12 = k/4$. So, increasing ϵ from $\frac{1}{4}$ up to $\frac{1}{2}$ increases the price of anarchy, and then increasing ϵ further to 1 will then decrease the price of anarchy to 1.

3.2.1 General Bounds. We now present a general bound on the diversified price of anarchy for games, as well as for the social welfare when all players use diversified regret-minimizing strategies such as given in Sections 2.1 and 2.2, using the smoothness framework of Roughgarden [25].

Definition 3.4. [25] A general-sum game is (λ, μ)*-smooth* if for any two outcomes $\mathbf{s}$ and $\mathbf{s}^*$,

$$\sum_{i=1}^{k} cost_i(s_i^*, \mathbf{s}_{-i}) \le \lambda \, cost(\mathbf{s}^*) + \mu \, cost(\mathbf{s}).$$

Here, $(s_i^*, \mathbf{s}_{-i})$ means the outcome in which player i plays its action in $\mathbf{s}^*$ but all other players play their action in $\mathbf{s}$.

Theorem 3.5. *If a game is (λ, μ)-smooth, then for any ϵ, the ϵ-diversified price of anarchy is at most $\frac{\lambda}{1-\mu}$.*

Proof. Let $\mathbf{s} = \mathbf{s}_\epsilon$ be some Nash equilibrium subject to all players playing ϵ-diversified strategies, and let $\mathbf{s}^* = \mathbf{s}_\epsilon^*$ be an outcome of optimum social cost subject to all players choosing ϵ-diversified strategies. Since $\mathbf{s}$ is an equilibrium, no player wishes to deviate to their action in $\mathbf{s}^*$; here we are using the fact that $\mathbf{s}^*$ includes only ϵ-diversified strategies, so such a deviation would be legal. Therefore $cost(\mathbf{s}) \le \sum_{i=1}^{k} cost_i(s_i^*, \mathbf{s}_{-i}) \le \lambda \, cost(\mathbf{s}^*) + \mu \, cost(\mathbf{s})$. Rearranging, we have $(1-\mu) cost(\mathbf{s}) \le \lambda \, cost(\mathbf{s}^*)$, so $cost(\mathbf{s})/cost(\mathbf{s}^*) \le \frac{\lambda}{1-\mu}$. □

Roughgarden [25] shows that atomic congestion games with affine cost functions, i.e., cost functions of the form $c_j(k_j) = a_j k_j + b_j$, are $(\frac{5}{3}, \frac{1}{3})$-smooth. So, their ϵ-diversified price of anarchy is at most 2.5. We now adapt the proof in Roughgarden [25] to show that players with vanishing regret with respect to diversified strategies will also approach the bound of Theorem 3.5.

Theorem 3.6. *Suppose that in repeated play of a (λ, μ)-smooth game, each player i uses a sequence of mixed strategies $s_i^{(1)}, \ldots, s_i^{(T)}$ such that for any ϵ-diversified strategy s_i^* we have:*

$$\frac{1}{T}\sum_{t=1}^{T} cost_i(\mathbf{s}^{(t)}) \le \frac{1}{T}\sum_{t=1}^{T} cost_i(s_i^*, \mathbf{s}_{-i}^{(t)}) \ + \ \Delta_T.$$

Then the average social cost over the T steps satisfies

$$\frac{1}{T}\sum_{t=1}^{T} cost(\mathbf{s}^{(t)}) \le \frac{\lambda}{1-\mu} cost(\mathbf{s}^*) + \frac{k\Delta_T}{1-\mu}.$$

In particular, if $\Delta_T \to 0$ then the average social cost approaches the bound of Theorem 3.5.

Proof. Combining the assumption of the theorem, the definition of social cost, and the smoothness definition we have:

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T} cost(\mathbf{s}^{(t)}) \quad &= \quad \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{k} cost_i(\mathbf{s}^{(t)}) \\
&\qquad \text{(definition of social cost)} \\
&\le \quad \sum_{i=1}^{k}\left[\frac{1}{T}\sum_{t=1}^{T} cost_i(s_i^*, \mathbf{s}_{-i}^{(t)}) + \Delta_T\right] \\
&\qquad \text{(assumption of theorem)} \\
&= \quad \frac{1}{T}\sum_{t=1}^{T}\left[\sum_{i=1}^{k} cost_i(s_i^*, \mathbf{s}_{-i}^{(t)})\right] + k\Delta_T \\
&\qquad \text{(rearranging)} \\
&\le \quad \frac{1}{T}\sum_{t=1}^{T}\left[\lambda \, cost(\mathbf{s}^*) + \mu \, cost(\mathbf{s}^{(t)})\right] + k\Delta_T. \\
&\qquad \text{(applying smoothness)}
\end{aligned}$$

Rearranging, we have:

$$(1-\mu)\frac{1}{T}\sum_{t=1}^{T} cost(\mathbf{s}^{(t)}) \quad \leq \quad \lambda\, cost(\mathbf{s}^*) + k\Delta_T,$$

which immediately yields the result of the theorem. □

3.3 The Cost of Diversification

We now complement the above results by considering how much worse $cost(s^*_\epsilon)$ can be compared to $cost(s^*)$ in natural games. We focus here on *unit-demand* congestion games where each strategy set $\mathcal{S}_i \subseteq R$; that is, each player i selects a single resource in $\mathcal{S}_i$. In particular, we focus on two important special cases: (a) $c_j(k_j) = 1/k_j\ \forall j$ (players share the cost of their resource equally with all others who make the same choice; this can be viewed as a game-theoretic distributed hitting-set problem), and (b) $c_j(k_j) = k_j\ \forall j$, i.e., linear congestion games. To avoid unnecessary complication, we assume all $\mathcal{S}_i$ have the same size n, i.e., every player has n choices. We will also think of the number of choices per player n as $O(1)$, whereas the number of players k and the total number of resources R may be large.

Unfortunately, in both cases (a) and (b), the cost of diversification can be very high in the worst case. For case (a) (cost sharing), a bad scenario is if there is a single element j^* such that $\mathcal{S}_i \cap \mathcal{S}_{i'} = j^*$ for all pairs $i \neq i'$. Here, $cost(s^*) = 1$ since all players can choose j^*, but for any $\epsilon \in [\frac{2}{n}, 1]$, we have $E[cost(s^*_\epsilon)] = \Omega(k)$, since even in the best solution each player has a 50% chance of choosing a resource that no other player chose. For case (b) (linear congestion), a bad scenario is if there are $n-1$ elements $j^*_1, \ldots, j^*_{n-1}$ such that $\mathcal{S}_i \cap \mathcal{S}_{i'} = \{j^*_1, \ldots, j^*_{n-1}\}$ for all pairs $i \neq i'$. Here, $cost(s^*) = k$ since each player can choose a distinct resource, but for any $\epsilon \in [\frac{2}{n}, 1]$, we have $E[cost(s^*_\epsilon)] = \Omega(k^2/(n-1))$, which is $\Omega(k^2)$ for $n = O(1)$. So, in both cases, the ratio $cost(s^*_\epsilon)/cost(s^*) = \Omega(k)$.

However, in the *average case* (each $\mathcal{S}_i$ consists of n *random* elements from R) the cost of diversification is only $O(1)$.

Theorem 3.7. *For both (a) unit-demand cost-sharing and (b) unit-demand linear congestion games, with $n = O(1)$ strategies per player and random strategy sets $\mathcal{S}_i$, $E[cost(s^*_\epsilon)] = O(E[cost(s^*)])$.*

Proof. Let us first consider (a) unit-demand cost-sharing. One lower bound on $cost(s^*)$ is that it is at least the cardinality of the largest collection of disjoint strategy sets $\mathcal{S}_i$; for $n = 2$ this is the statement that the smallest vertex cover in a graph is at least the size of the maximum matching. Now consider selecting the random sets $\mathcal{S}_i$ one at a time. For $i \leq R/n^2$, the first i sets cover at most R/n resources, so set $\mathcal{S}_{i+1}$ has at least a constant probability of being disjoint from the first i. This means that the expected size of the largest collection of disjoint strategy sets is at least $\Omega(\min\{k, R/n^2\})$. On the other hand, a trivial upper bound on $cost(s^*_\epsilon)$, even for $\epsilon = 1$, is $\min\{k, R\}$, since at worst each player takes a separate resource until all resources are used. Thus, for $n = O(1)$, we have $E[cost(s^*_\epsilon)] = O(E[cost(s^*)])$.

Now let us consider (b) unit-demand linear congestion. In this case, a lower bound on $cost(s^*)$ is the best-case allocation of all resources equally divided. In this case we have k/R usage per resource for a total cost of $R \times (k/R)^2 = k^2/R$. Another lower bound is simply k, so we have $cost(s^*) \geq \max\{k, k^2/R\}$. On the other hand, we can notice that s^*_ϵ for a fully-diversified $\epsilon = 1$ and random sets $\mathcal{S}_i$ is equivalent to players choosing resources independently at random. In this case, the social cost is identical to the analysis of random hashing: $E[cost(s^*_1)] = E[\sum_j k_j^2] = k + k(k-1)/R$. Thus, $E[cost(s^*_\epsilon)] = O(E[cost(s^*)])$ as desired. □

4 DISTRIBUTED SETTING

We now consider a distributed setting where the actions of the row player are partitioned among k entities, such as subdivisions within a company or machines in a distributed system. At each time step, the row player asks for a number of actions from each entity, and plays a mixed strategy over them. However, asking for actions requires communication, which we would like to minimize.

Our aim is to obtain results similar to Theorem 2.4 with low communication complexity, measured by the number of actions requested and any additional constant-sized words communicated. Let $d \leq \log m$ denote the VC-dimension or pseudo-dimension of the set of columns H, viewing each row as an example and each column as a hypothesis. A baseline approach is that the row player samples $O(\frac{d}{\epsilon^2}\log\frac{1}{\epsilon})$ times, from the uniform multinomial distribution over $\{1, \ldots, k\}$ and asks each entity to send the corresponding number of actions to the row player. The row player can then use Algorithm 1 as in the centralized setting over the sampled actions, and will lose only an additional ϵ in the value of its strategy. The communication complexity of this method is $O(\frac{d}{\epsilon^2}\log\frac{1}{\epsilon})$ actions plus $O(k)$ additional words. Here, we provide an algorithm that reduces communication to $O(d\log(1/\epsilon))$. The idea is to show that in Algorithm 1, each iteration of the multiplicative weight update can be simulated in the distributed setting with $O(d)$ communication. Then, since there are at most $O(\log(1/\epsilon))$ iterations, the desired result follows. More specifically, we show that in each iteration, we can do the following two actions communication efficiently:

(1) For any distribution P over the rows partitioned across k entities, obtain a column j such that $M(P, j) \geq v_\epsilon$.
(2) Update the distribution using the received column j.

To achieve the first statement, assume there is a centralized oracle, which for any ϵ-diversified distribution P returns a column j such that $M(P, j) \geq v_\epsilon$. For any distribution P partitioned across k entities, each agent first sends its sum of weights to the row player. Then, the row player samples $O(\frac{d}{(1-\alpha)^2 v_\epsilon^2})$ actions $(0 < \alpha < 1)$ across the k agents proportional to their sum of weights, where d is the VC-dimension of H. By the standard VC-theory, a mixed strategy P' of choosing a uniform distribution over the sampled actions is a $(1-\alpha)v_\epsilon$-approximation for H, i.e. $M(P', j) \geq M(P, j) - (1-\alpha)v_\epsilon \geq \alpha v_\epsilon$ for all column $j \in H$. The communication complexity of this step is $O(\frac{d}{(1-\alpha)^2 v_\epsilon^2})$ actions plus $O(k)$ additional words. For (2), we show steps 3 and 4 in Algorithm 1 can be simulated with low communication. Step 3 is easy: just send column j to all entities, and each entity then updates its own weights. What is left is to show that the projection step in Algorithm 1 can be simulated in the distributed setting. Fortunately, this projection step has been studied before in the distributed machine learning literature [11], where an efficient algorithm with $O(k\log^2(d/\epsilon))$ words of comunication is proposed. We summarize our results for the distributed setting with the following theorem.

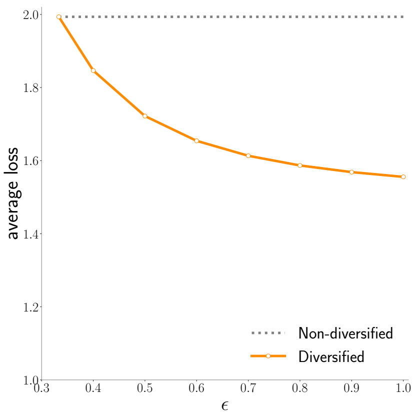


Figure 2: Simulated resuls of Braess' paradox after $T = 10,000$ rounds. A more diversified strategy leads to lower loss.

Theorem 4.1. *Given a centralized oracle, which for any ϵ-diversified distribution P returns a column j such that $M(P, j) \geq v_\epsilon$. If the actions of the row players are distributed across k entities, there is an algorithm that constructs a mixed strategy Q such that for all but an ϵ fraction of the rows i, $M(i, Q) \geq \alpha v_\epsilon - \gamma$, $0 < \alpha < 1$. The algorithm requests at most $O(\frac{\log(1/\epsilon)}{\gamma^2(1+\gamma-\alpha v_\epsilon)} \cdot \frac{d}{(1-\alpha)^2 v_\epsilon^2})$ actions and uses an additional $O(\frac{\log(1/\epsilon)}{\gamma^2(1+\gamma-\alpha v_\epsilon)} \cdot k \log^2(d/\epsilon))$ words of communication.*

5 EXPERIMENTS

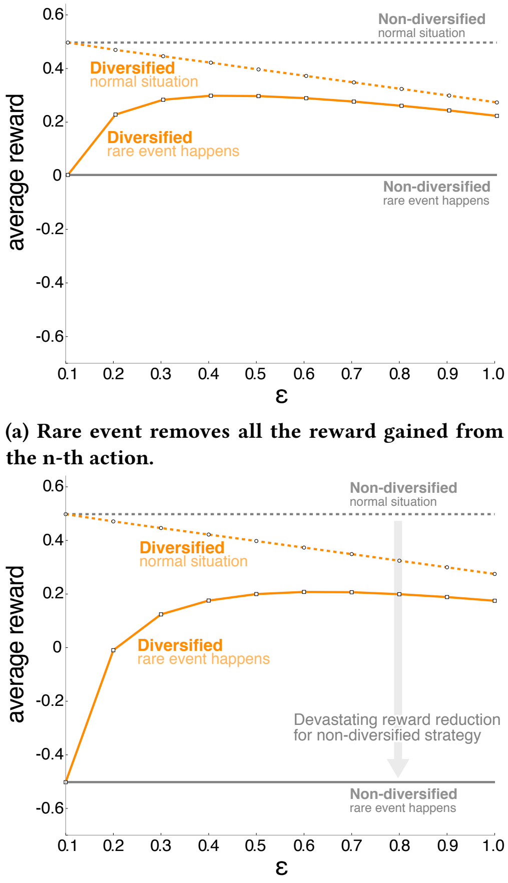
To better understand the benefit of diversified strategies, we give some empirical simulations for both two-player zero-sum games and general-sum games. For all the experiments, we fix $\gamma = 0.2$ and show the results of using different values of ϵ.

Two-player zero-sum games. The row player has $n = 10$ actions to choose from, where each round, each action a_i returns a uniformly random reward $r_i \in [i/n, 1]$. The game is played for $T = 10,000$ rounds. Note that the n-th action has the highest expected reward.

We consider two scenarios in which a rare but catastrophic event occurs. The first scenario is that at time T, the cumulative reward gained from choosing the n-th action becomes zero. The second scenario is that the n-th action incurs a large negative reward of $-T$ in time step T. Both of these can be viewed as different ways of simulating a bad event where, for instance, the shares of a company become worthless when the company goes bankrupt.

The results for both scenarios, averaged over 10 independent trials, are shown in Figure 3. One can see that as expected, in the normal situation, the diversified strategy gains less reward. However, when the rare event happens, the non-diversified strategy gains very low reward. In both cases, a modest value of $\epsilon = 0.4$ achieves a high reward whether the bad event happens or not.

General-sum games. We play the routing game defined in Braess' paradox (see Figure 1). Each player has three routes to choose from (s-a-b-t, s-a-t, and s-b-t) in each round, so $\epsilon \in [1/3, 1]$. As anlyzed in Section 3.1, without the diversified constraint (i.e., $\epsilon = 1/3$), the game quickly converges to the Nash equilibrium where all players choose the route s-a-b-t and incur a loss of 2. The best strategy in this case is to play the 1-diversified strategy, which incur a lower loss of about 1.55. See Figure 2 for the results using other ϵ values.



(a) Rare event removes all the reward gained from the n-th action.

(b) Rare event changes the reward of the n-th action to $-T$ in the last round.

Figure 3: Average reward over $T = 10,000$ rounds with different values of ϵ. When the rare event happens, the non-diversified strategy gains very low (even negative) reward.

6 CONCLUSION

We consider games in which one wants to play well without choosing a mixed strategy that is too concentrated. We show that such a diversification restriction has a number of benefits, and give adaptive algorithms to find diversified strategies that are near-optimal, also showing how taxes or fines can be used to keep a standard algorithm diversified. Further, our algorithms are simple and efficient, and can be implemented in a distributed setting. We also analyze properties of diversified strategies in both zero-sum and general-sum games, and give general bounds on the diversified price of anarchy as well as the social cost achieved by diversified regret-minimizing players.

Acknowledgements: This work is supported in part by NSF grants CCF-1101283, CCF-1451177, CCF-1535967, CCF-1422910, CCF-1800317, CNS-1704701, ONR grant N00014-09-1-0751, AFOSR grant FA9550-09-1-0538, and a gift from Intel.

REFERENCES

[1] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2005. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*. IEEE, 339–348.

[2] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2012. The Multiplicative Weights Update Method: a Meta-Algorithm and Applications. *Theory of Computing* 8, 1 (2012), 121–164.

[3] Moshe Babaioff, Robert Kleinberg, and Christos H Papadimitriou. 2007. Congestion games with malicious players. In *Proceedings of the 8th ACM conference on Electronic commerce*. ACM, 103–112.

[4] M.-F. Balcan, A. Blum, S. Fine, and Y. Mansour. 2012. Distributed Learning, Communication Complexity and Privacy. *Journal of Machine Learning Research - Proceedings Track* 23 (2012), 26.1–26.22.

[5] Maria-Florina Balcan, Avrim Blum, and Yishay Mansour. 2009. The price of uncertainty. In *Proceedings of the 10th ACM conference on Electronic commerce*. ACM, 285–294.

[6] Maria-Florina Balcan, Florin Constantin, and Steven Ehrlich. 2011. The snowball effect of uncertainty in potential games. In *International Workshop on Internet and Network Economics*. Springer, 1–12.

[7] Avrim Blum and Yishay Mansour. 2007. *Learning, Regret Minimization, and Equilibria*. Cambridge University Press, 79–102. https://doi.org/10.1017/CBO9780511800481.006

[8] Dietrich Braess. 1968. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung* 12 (1968), 258–268.

[9] Ioannis Caragiannis, David Kurokawa, and Ariel D Procaccia. 2014. Biased Games. In *AAAI*. 609–615.

[10] Nicolo Cesa-Bianchi and Gábor Lugosi. 2006. *Prediction, learning, and games*. Cambridge university press.

[11] S.-T. Chen, M.-F. Balcan, and D. Chau. 2016. Communication Efficient Distributed Agnostic Boosting. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016*. 1299–1307.

[12] S.-T. Chen, H.-T. Lin, and C.-J. Lu. 2012. An Online Boosting Algorithm with Theoretical Justifications. In *Proceedings of ICML*. 1007–1014.

[13] N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma. 2004. Adversarial classification. In *KDD*.

[14] Hal Daumé, Jeff M. Phillips, Avishek Saha, and Suresh Venkatasubramanian. 2012. Efficient Protocols for Distributed Classification and Optimization. In *Proceedings of the 23rd International Conference on Algorithmic Learning Theory (ALT'12)*. 154–168.

[15] Y. Freund and R. E. Schapire. 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. System Sci.* 55, 1 (1997), 119–139.

[16] Yoav Freund and Robert E Schapire. 1999. Adaptive game playing using multiplicative weights. *Games and Economic Behavior* 29, 1 (1999), 79–103.

[17] Dmitry Gavinsky. 2003. Optimally-smooth Adaptive Boosting and Application to Agnostic Learning. *J. Mach. Learn. Res.* 4 (2003), 101–117.

[18] Matthias Hein and Maksym Andriushchenko. 2017. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Advances in Neural Information Processing Systems*. 2263–2273.

[19] Mark Herbster and Manfred K. Warmuth. 2001. Tracking the Best Linear Predictor. *J. Mach. Learn. Res.* 1 (2001), 281–309.

[20] Russell Impagliazzo. 1995. Hard-core distributions for somewhat hard problems. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. IEEE, 538–545.

[21] Nick Littlestone and Manfred K Warmuth. 1989. The weighted majority algorithm. In *Foundations of Computer Science, 1989., 30th Annual Symposium on*. IEEE, 256–261.

[22] Panagiota N Panagopoulou and Paul G Spirakis. 2014. Random bimatrix games are asymptotically easy to solve (A simple proof). *Theory of Computing Systems* 54, 3 (2014), 479–490.

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

[24] Tim Roughgarden. 2007. Routing games. *Algorithmic game theory* 18 (2007), 459–484.

[25] Tim Roughgarden. 2015. Intrinsic Robustness of the Price of Anarchy. *J. ACM* 62, 5 (2015), 32:1–32:42. https://doi.org/10.1145/2806883

[26] Maurice Sion. 1958. On general minimax theorems. *Pacific J. Math.* 8, 1 (1958), 171–176.

[27] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. 2014. Intriguing properties of neural networks. In *ICLR*.