Equilibrium Finding in Matrix Games Via Greedy Regret Minimization

Hugh Zhang[1*], **Adam Lerer**[2], **Noam Brown**[2]

[1]Harvard University, [2]Facebook AI Research
hughzhang@fas.harvard.edu, alerer@fb.com, noambrown@fb.com

Abstract

We extend the classic regret minimization framework for approximating equilibria in matrix games by greedily weighing iterates based on the regrets observed at runtime. Theoretically, our method retains all previous convergence rate guarantees. Empirically, experiments on large random matrix games and normal-form subgames of the AI benchmark Diplomacy show that greedy weights outperforms previous methods whenever sampling is used, sometimes by several orders of magnitude.

Introduction

Constructing algorithms that efficiently converge to equilibria is one of the central goals of computational game theory. In recent years, regret minimization techniques for approximating equilibria via self play have led to a number of major successes in games like poker (Bowling et al. 2015; Moravčík et al. 2017; Brown and Sandholm 2018, 2019b), Avalon (Serrino et al. 2019), and Diplomacy (Gray et al. 2020). Regret minimization is now the state-of-the-art approach for computing equilibria in large games, especially those with a large number of actions or in settings where queries to the payoff matrix are too expensive to compute an exact solution via linear programming.

However, past uses of regret minimization algorithms for self play in games typically treat each player as an independent online learner, which has led to missed opportunities for accelerating convergence. In particular, in online learning settings each iteration is typically weighed equally. In this paper, we show it is possible to empirically achieve faster convergence, while still maintaining the same worst-case regret bound, by greedily weighing each new iteration to minimize the potential function: a function of regret measuring distance to the set of desired equilibria. Previous algorithms such as CFR+ (Tammelin 2014) and Linear CFR (Brown and Sandholm 2019a) have also shown that faster performance is achievable by weighing iterates non-uniformly. However, in all previous algorithms, iterates were weighed according to a fixed, pre-determined schedule. In contrast, we introduce **greedy weights**, the first equilibrium-finding regret minimization algorithm where iterates are weighed *dynamically* using information available at *runtime*. We benchmark greedy weights against past techniques for computing minimax equilibria (in two-player zero-sum games), coarse-correlated equilibria, and correlated equilibria in large, randomly generated matrix games. Additionally, we conduct experiments in subgames of Diplomacy, an important AI benchmark which has a long history as a benchmark for AI research (Kraus and Lehmann 1988; Kraus, Ephrati, and Lehmann 1994; Kraus and Lehmann 1995; Johansson and Håård 2005; Ferreira, Cardoso, and Reis 2015) and has been a particularly active domain for research in recent years (Paquette et al. 2019; Anthony et al. 2020; Gray et al. 2020). We find that greedy weights significantly improves the rate of convergence whenever sampling is used, in some cases by several orders of magnitude. Finally, we find that equilibria discovered by greedy weights in general-sum games typically have higher overall social welfare than those found by past methods.

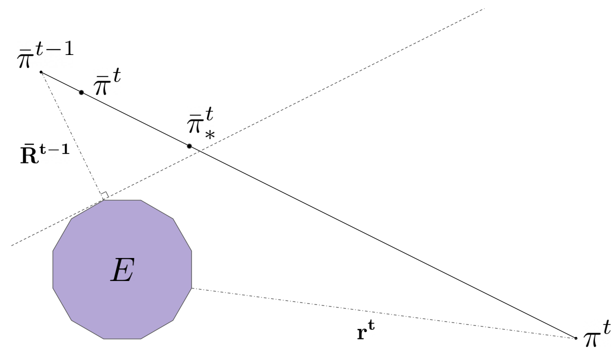


Figure 1: Vanilla regret minimization methods always move the rolling weighted average policy profile discovered so far $\bar{\pi}^{t-1}$ (with average regret represented by $\bar{\mathbf{R}}^{\mathbf{t-1}}$) a fixed step towards the latest iteration of the procedure π^t (with instantaneous regret represented by $\mathbf{r^t}$) to the point represented by $\bar{\pi}^t$. In contrast, greedy weights will choose the point on the line $\bar{\pi}^t_*$ that minimizes the potential function measuring the distance to the set of equilibria we wish to approach (represented by E), often resulting in accelerated convergence.

Notation and Background

In a strategic-form (also called normal-form or matrix) game, each of $\mathcal{P}$ players simultaneously choose their actions with-

[*]Work done primarily while the author was at Facebook AI Research.

out observing the other players' choices. Each player then receives a reward determined by a function of all players' actions. Let Δ represent the difference between the highest and lowest possible payoff of any player in the game. Let A_i denote the set of actions for player i and A the set of joint actions for all players. We denote the set of joint actions for all players *except* i as A_{-i}. Let Σ_i represent the set of probability distributions over actions in A_i (i.e., the set of mixed policies, also known as strategies). Σ is the set of joint policies across all players, and Σ_{-i} is the set of joint policies for all players other than i. $\pi_i \in \Sigma_i$ denotes player i's policy, which is a probability distribution over actions in A_i. The probability of action a_i in π_i is denoted $\pi_i(a_i)$. Similarly, π_{-i} and π denote the policies for all players other than i and for all players, respectively. The payoff player i receives when all players play joint action $a \in A$ is denoted $u_i(a) = u_i(a_i, a_{-i})$. Analogously, the expected payoff to player i when all players play policy profile π is denoted $u_i(\pi) = u_i(\pi_i, \pi_{-i})$.

Equilibria in Games

Perhaps the most well-known equilibrium concept for games is the Nash equilibrium (NE) (Nash 1951). A NE is a tuple of policies (one for each player) in which no player can improve by deviating to a different policy. Formally, a policy profile π is a NE if it satisfies:

$$\max_{i \in \mathcal{P}} \max_{\pi' \in \Sigma_i} u_i(\pi', \pi_{-i}) - u_i(\pi_i, \pi_{-i}) \leq 0 \tag{1}$$

Well-known results in complexity theory have suggested that discovering (or even approximating) a NE in general games is computationally hard (Chen, Deng, and Teng 2009; Daskalakis, Goldberg, and Papadimitriou 2009; Rubinstein 2019). As a result, researchers often also consider the correlated equilibrium (CE) (Aumann 1974), an alternative solution concept which is efficiently computable in all matrix games. Whereas a NE is a probability distribution over actions A_i for each player i, a CE is a probability distribution p over the set of joint actions A that satisfies certain incentive constraints. In order for a probability distribution over joint actions to be a CE, it must be the case that if a mediator were to sample a joint action a from that distribution and privately share with each player i their action a_i that is part of the joint action, then no player could gain by deviating from that action. Formally, a CE is a probability distribution over joint actions in A satisfying

$$\max_{i \in \mathcal{P}} \max_{\phi: A_i \to A_i} \sum_{a \in A} p(a)(u_i(\phi(a_i), a_{-i}) - u_i(a_i, a_{-i})) \leq 0$$

Finally, a *coarse*-correlated equilibrium (CCE) (Hannan 1957) is also a probability distribution over joint actions but with a weaker incentive constraint than the correlated equilibrium. In order for a probability distribution over joint actions to be a CCE, it must be the case that if a mediator were to sample a joint action a from that distribution and each player i is forced to play their action a_i that is part of the joint action, then no player could gain by refusing to receive an action from the mediator and choosing an action on their own instead. Formally, a CCE satisfies:

$$\max_{i \in \mathcal{P}} \max_{a'_i \in A_i} \sum_{a \in A} p(a)(u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})) \leq 0 \tag{2}$$

We can define the ϵ-versions of all of the above equilibria by replacing the 0 on the right hand side of the equations with an ϵ. In two-player zero-sum games, NE, CE, and CCE can be shown to be payoff equivalent to one another via the minimax theorem (Neumann 1928).

Regret Minimization

There exist several polynomial-time algorithms for computing CEs and CCEs. This paper focuses on the leading approach for large games: regret minimization algorithms. In addition to their theoretical guarantees of convergence to equilibria, regret minimization algorithms have been behind recent empirical successes in large-scale game benchmarks such as many forms of poker (including non-two-player poker) (Bowling et al. 2015; Moravčík et al. 2017; Brown and Sandholm 2018, 2019b), Avalon (Serrino et al. 2019), and Diplomacy (Gray et al. 2020).

For any sequence of policies $\pi^1 \dots \pi^T$ in a game G, player i's weighted **external regret** for not having played action $a'_i \in A_i$ is

$$R_i^{E,T}(a'_i) = \sum_{t=1}^{T} w_t\big(u_i(a'_i, \pi^t_{-i}) - u_i(\pi^t)\big)$$

We can thus define the overall average external regret for player i as

$$\bar{R}_i^{E,T} = \max_{a'_i \in A_i} \frac{R_i^{E,T}(a'_i)}{\sum_{t=1}^{T} w_t} \tag{3}$$

Analogously, we can define player i's weighted **internal regret** for not swapping to action a'_i every time she actually played action a_i^A as

$$R_i^{I,T}(a_i^A, a'_i) = \sum_{t=1}^{T} \mathbb{1}[a_i^t = a_i^A] w_t(u_i(a'_i, a^t_{-i}) - u_i(a^t))$$

and her overall average internal regret as

$$\bar{R}_i^{I,T} = \max_{a'_i, a_i^A \in A_i} \frac{R_i^{I,T}(a_i^A, a'_i)}{\sum_{t=1}^{T} w_t}$$

We denote the vector (for all players) of average external regrets by $\bar{\mathbf{R}}^{E,T}$ and average internal regret by $\bar{\mathbf{R}}^{I,T}$. Additionally, whenever we use the notation R_+, we refer to the positive regrets defined by $\max(0, R)$ for whatever regret R represents. For example, $R_{i,+}^{E,t}(a_i) = \max\Big(0, R_i^{E,t}(a_i)\Big)$.

Celebrated past results have shown that for any strategic-form game, minimizing average external regret for all players leads to a CCE while minimizing average internal regret for all players leads to a CE (Cesa-Bianchi and Lugosi 2006). Many well-known methods exist for minimizing both internal and external regret. For our experiments in external regret minimization, we use Blackwell's regret minimization (Blackwell 1956), which asks every player to select their next action proportional to how much they regret having not selected that action in the past. Formally, each player selects action $a_i \in A_i$ at timestep $t+1$ with probability

$$\Pr(a_i^{t+1} = a_i) = \frac{R_{i,+}^{E,t}(a_i)}{\sum_{a_i' \in A_i} R_{i,+}^{E,t}(a_i')}$$

except in the case where all regrets are nonpositive upon which a_i^{t+1} is chosen uniformly at random from A_i.

For internal regret minimization, we primarily use an extension of Blackwell's regret minimization given by Hart and Mas-Colell (2000) also known as **regret matching**. Regret matching also selects its policy with probability "matching" its past regrets of not switching to that action in the past, but it differs in that it additionally uses a fixed inertia parameter α and thus always retains a positive probability of staying in place, with probability approaching 1 as the overall regrets vanish[1]. Formally, we have

$$\Pr(a_i^{t+1} = a_i) = \begin{Bmatrix} \frac{\alpha}{\alpha + \sum_{a_i' \in A_i} R_{i,+}^{I,t}(a_i^t, a_i')}, & \text{if } a_i = a_i^t \\ \frac{R_{i,+}^{I,t}(a_i^t, a_i)}{\alpha + \sum_{a_i' \in A_i} R_{i,+}^{I,t}(a_i^t, a_i')}, & \text{otherwise} \end{Bmatrix}$$

Due to the substantial similarities between the two algorithms, we refer to both as regret matching. Both forms of regret matching have been shown to be special cases of a general class of potential-based minimizers (Cesa-Bianchi and Lugosi 2006; Hart and Mas-Colell 2001; Cesa-Bianchi and Lugosi 2001). Specifically, many of their theoretical properties can be proved via careful examination of the potential function, defined as the sum of the squared positive regrets. The potential functions that regret matching minimizes for external and internal regret, respectively, are:

$$\phi(\bar{\mathbf{R}}_+^{E,T}) = \sum_{i \in P} \sum_{a_i' \in A_i} \left(\bar{R}_{i,+}^{E,T}(a_i')\right)^2$$

$$\phi(\bar{\mathbf{R}}_+^{I,T}) = \sum_{i \in P} \sum_{a_i^A, a_i^B \in A_i} \left(\bar{R}_{i,+}^{I,T}(a_i^A, a_i^B)\right)^2$$

External regret matching guarantees that $\phi(\bar{\mathbf{R}}_+^{E,T}) \leq \frac{|P|\Delta^2|A|}{T}$, which in turn guarantees that $\max_{i \in P} \max_{a_i \in A_i} \bar{R}_i^{E,T}(a_i) \leq \frac{\Delta\sqrt{|P||A|}}{\sqrt{T}}$, with similar guarantees for internal regret. If all players' average regret is bounded by ϵ, then the empirical distribution of play is an $O(\epsilon)$-equilibrium.

Greedy Weights

Blackwell's original regret minimization procedure (Blackwell 1956) and its various extensions (e.g. Hart and Mas-Colell (2000), Blum and Mansour (2007), Zinkevich et al. (2008)) and applications to online learning (Abernethy, Bartlett, and Hazan 2011) typically assign equal weight to each iteration of the procedure. Recently, Brown and Sandholm (2019a) demonstrated that modifying the weight schedule of regret matching empirically resulted in faster convergence to equilibria while maintaining a similar worst-case convergence bound. However, this modified schedule was fixed and pre-determined before the start of the procedure. Our algorithm extends this direction of inquiry by greedily choosing the iteration weights to minimize a function of regret at runtime.

Algorithm 1: Greedy Weights

Input: total timesteps T, game G, regret minimizer M
Initialize a randomly and compute immediate regret $\mathbf{r}$
Set $\bar{\pi} = a$, $w_{\text{sum}} = 1$, $\mathbf{R} = \mathbf{r}$
for $t = 1$ **to** T **do**
  $\pi, \mathbf{r} \leftarrow M(G, \phi, \mathbf{R})$
  $w \leftarrow \underset{w}{\operatorname{argmin}}\, \phi\left((\mathbf{R} + w\mathbf{r})/(w_{\text{sum}} + w)\right)$
  $\mathbf{R} \leftarrow \mathbf{R} + w\mathbf{r}$
  $\bar{\pi} \leftarrow (w_{\text{sum}}\bar{\pi} + w\pi)/(w_{\text{sum}} + w)$
  $w_{\text{sum}} \leftarrow w_{\text{sum}} + w$
end for
return $\bar{\pi}$

Figure 1 demonstrates why this might prove useful. We represent the set of equilibria we wish to approach as E. At iteration t, let $\bar{\pi}^{t-1}$ represent the weighted average policy thus far and π^t the policy played at iteration t of the procedure. Their regrets are denoted by $\bar{\mathbf{R}}^{\mathbf{t-1}}$ and $\mathbf{r^t}$, respectively. Vanilla regret minimization would give an overall weight of $\frac{1}{t}$ to iteration t, regardless of whether an alternative weighting would result in an average policy closer to an equilibrium. In contrast, greedy weights would choose the relative weighting between $\bar{\pi}^{t-1}$ and π^t that minimizes the potential function measuring the distance of the resulting new average policy $\bar{\pi}_*^t$ to the set of equilibria E. Formally, greedy weights picks the weight of each iteration to greedily minimize $\phi_i^{E,T}$ or $\phi_i^{I,T}$, depending on whether external or internal regret is being minimized.

We describe our full greedy weights procedure in Algorithm 1. On each iteration the regret vectors for all players, denoted by $\mathbf{R}$, is used to determine the policy profile π played in the regret minimization algorithm and the resulting instantaneous regret vectors for all players, denoted by $\mathbf{r}$ (which in typical regret minimization would be added to $\mathbf{R}$, leading to the new regret vectors). Next, the weight w for $\mathbf{r}$ that minimizes the potential function is computed. Both the update to the regret vectors and the update to the average policy profile is weighed by w, and the process repeats. To reduce the risk of numerical instability and overflow, it is sometimes beneficial to discount the previously computed regrets and average policy profiles rather than upweight the next iteration. In our experiments, if $w > 1$ then rather than weigh the next iteration by w we instead discount all previous iterations by $\frac{1}{w}$ and weigh the next iteration by 1.

As with many greedy algorithms, greedy weights is not guaranteed to converge faster than vanilla regret matching. In particular, we observe in two-player zero-sum games that

[1] Note that our formulation is slightly different from the formulation given in their original paper in that it allows us to use a very low inertia constant ($\alpha = 10^{-10}$) and thus prevents the procedure from repeating actions until the regrets become very low. We empirically observe that this dramatically accelerates convergence to CE in our experiments with large random matrix games.

setting a weight floor of $\frac{w_{\text{sum}}}{2t}$ is often useful for speeding up convergence. In all other settings, we did not observe a floor to be beneficial. In the Appendix, we describe ablations measuring the performance for different weight floors.

Computing the optimal weight is essentially a line search procedure. It can be approximated via binary search or computed exactly by checking $O(|\mathcal{P}||A|)$ points. This is especially useful in cases where $\mathbf{r}$ may be expensive to compute (e.g. when evaluated using a neural network value function), because this line search only requires a single evaluation of $\mathbf{r}$ and is then able to compute $\phi(\mathbf{R}+w\mathbf{r})$ as a simple algebraic function of $\mathbf{R}, \mathbf{r}, w$ without any further queries to the reward function.

Greedy weights retains the convergence guarantees of previous regret minimization methods, as demonstrated by the following theorem.

Theorem 1. *If the policies for each player and weights for each iteration are selected according to Algorithm 1 and run for T iterations, the resulting average distribution of plays $\bar{\pi}^T$ is guaranteed to be an $O(\frac{1}{\sqrt{T}})$ -equilibrium.*

The full proof for external and internal regret matching is provided in the Appendix.

Experimental Results

We benchmark **greedy weights** against the state-of-the-art algorithms for regret minimization on randomly generated matrix games and on subgames from the benchmark seven-player game Diplomacy. In our experiments we evaluate the following regret minimization methods:

1. **Regret Matching (RM)** (Blackwell 1956; Hart and Mas-Colell 2000). Blackwell's was the the original (external) regret minimization procedure, where each player chooses their next policy proportionally to their positive regrets. Hart and Mas-Colell (2000) extended their procedure to internal regret by adding a small inertia parameter that causes players to tend to stay in place as their regrets go to 0.
2. **RM+** (Tammelin 2014). RM+ makes two changes to RM. The first is that a distinction is made between the "true" regrets and the "guiding" regrets, where the guiding regrets are used to determine the policies on the next iteration. After every iteration, any negative guiding regrets are set to zero. The second is that in RM+ iteration t is given weight t when computing the final average policy (but not when computing the next iteration's policy). RM+ does well with mixed strategies, but does poorly with sampling (Burch 2017).
3. **Linear RM** (Brown and Sandholm 2019a). Linear RM is identical to vanilla RM except that iteration t is given weight t (both when computing the average policy and when computing the next iteration's policy).

Additionally, prior work has discovered several additional modifications to regret minimization that significantly improves the convergence rate in two-player zero-sum games in practice. We additionally evaluate these methods.

1. **Alternating Updates** (Tammelin 2014). For two-player zero-sum games, each player's guiding regrets are updated only once every other iteration. We generalize this procedure to n-player games by updating each player's regrets only once every n iterations.
2. **Optimism** (Syrgkanis et al. 2015; Farina, Kroer, and Sandholm 2020). The guiding regrets are modified such that the latest iteration is temporarily counted twice. This boost is subtracted away from the guiding regrets immediately after the next iteration's strategy is determined and the equilibrium regrets remain unchanged.

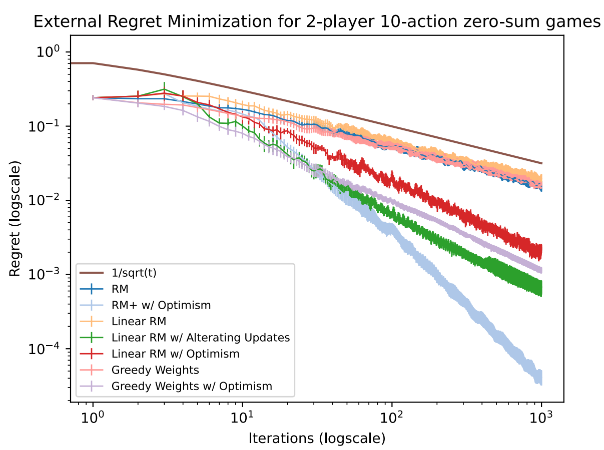
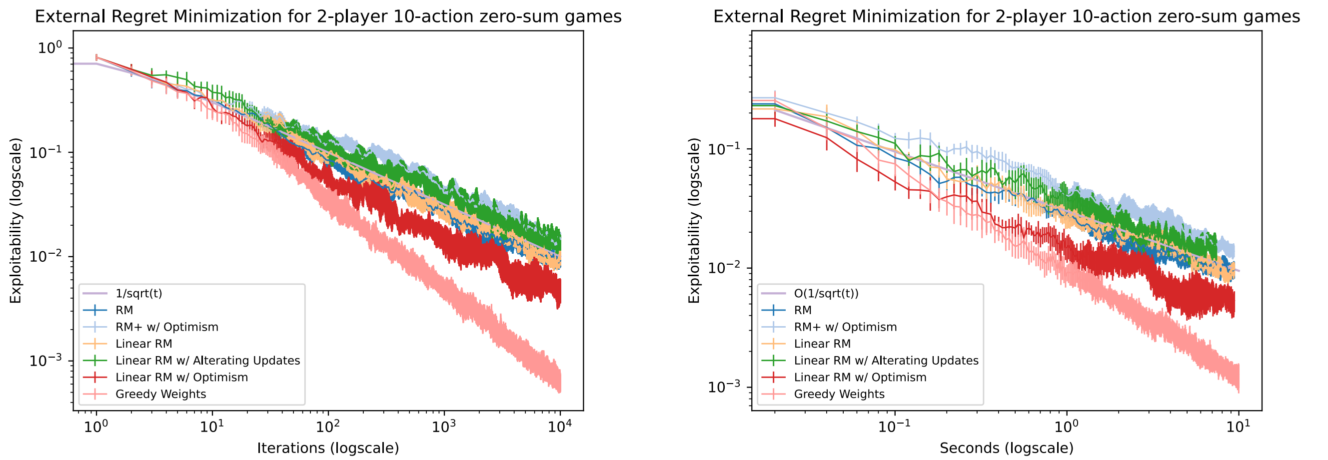


Figure 2: In the special case of two-player zero-sum games where mixed strategies are used at each iteration is applied, greedy weights (combined with optimism (Syrgkanis et al. 2015)) outperforms many but not all previous methods for minimizing external regret. However, this trick is not feasible for general computation of equilibria or in games with a large number actions where full queries to the payoff matrix are too expensive.

Pure vs Mixed Policies at each Iteration

In order to interpret the results in the following sections, it is necessary to discuss a technique often used in regret matching in the two-player zero-sum setting. When computing a Nash equilibrium in two-player zero-sum games, it is possible to simulate each player playing their *mixed* policy at each iteration rather than sampling a single action to play. This modification leads to much faster convergence with certain regret matching schemes. However, using mixed policies comes with some a major computational drawback in games with more than two players, as the computational cost of each iteration scales as $O(|P|\,|A|^{|P|})$ for mixed-strategy RM as opposed to $O(|A|\,|P|)$ for standard RM, making it intractable for games with more than 2-3 players, games with a large action space, or games where evaluating the payoff matrix is expensive. A game such as Diplomacy - an important AI challenge problem that has previously been tackled with RM (Gray et al. 2020) - has all three of these properties, making mixed-strategy RM untenable. We show that while some regret matching variants outperform greedy RM in the mixed-strategy two-player zero-sum setting (see Figure 2), only greedy regret matching improves convergence in the general setting where mixed-strategy RM cannot be applied.



(a) Exploitability v. Iteration

(b) Exploitability v. Time

Figure 3: We generate 10 two-player zero-sum games with P1's matrix payoff entries selected uniformly at random from $[0, 1)$ and run the best of the described approaches for minimizing external regret in the sampled setting. Greedy weights outperforms all methods by almost an order of magnitude. Figure 3a shows the results as a function of the number of iterations used, while Figure 3b shows the same data as a function of time for fair comparison as iterations of greedy weights cost slightly more than previous methods. Exploitability is the distance to a Nash equilibrium. Both axes are logscale and error bars are shown at 95% confidence.

Two-Player Zero-Sum Random Matrix Games

We first evaluate greedy weights on the special case of two-player zero-sum games, where external regret minimization finds a Nash equilibrium. The results are shown in Figure 3. Unlike the mixed case depicted in Figure 2, no prior methods achieve a faster asymptotic convergence rate than $O(\frac{1}{\sqrt{T}})$ in the pure strategy setting. However, greedy RM empirically displays asymptotically faster convergence, and achieves regret over an order of magnitude lower after around 10^3 iterations, even when accounting for the additional computational overhead of computing the optimal iteration weight. All random matrix experiments (both zero-sum and general-sum) were done on a single CPU core.

General-Sum Matrix Games

In Figure 4, we compare the convergence of different RM weighting schemes for computing correlated equilibria in random matrix games. We observe again that no prior methods achieve asymptotically faster convergence to correlated equilibria than $O(\frac{1}{\sqrt{T}})$ in general-sum games in the pure strategy setting. Our proposed greedy weighting scheme, however, dramatically improves convergence to correlated equilibria in large general-sum games. These results are robust to variance between games (error bars listed at 95% confidence) and hold across a large spectrum of games with varying numbers of players and actions. For the sake of space, we have relegated most of these plots to the Appendix.

We also investigate whether the equilibria that greedy weights discovers in general-sum games increase total expected value (i.e., social welfare) compared to equilibria found using vanilla regret minimization. Intuitively, we might expect this to be the case since iterations where players receive higher overall rewards are also likely to be iterations with lower overall regret, and would thus be upweighted by the greedy weights algorithm.

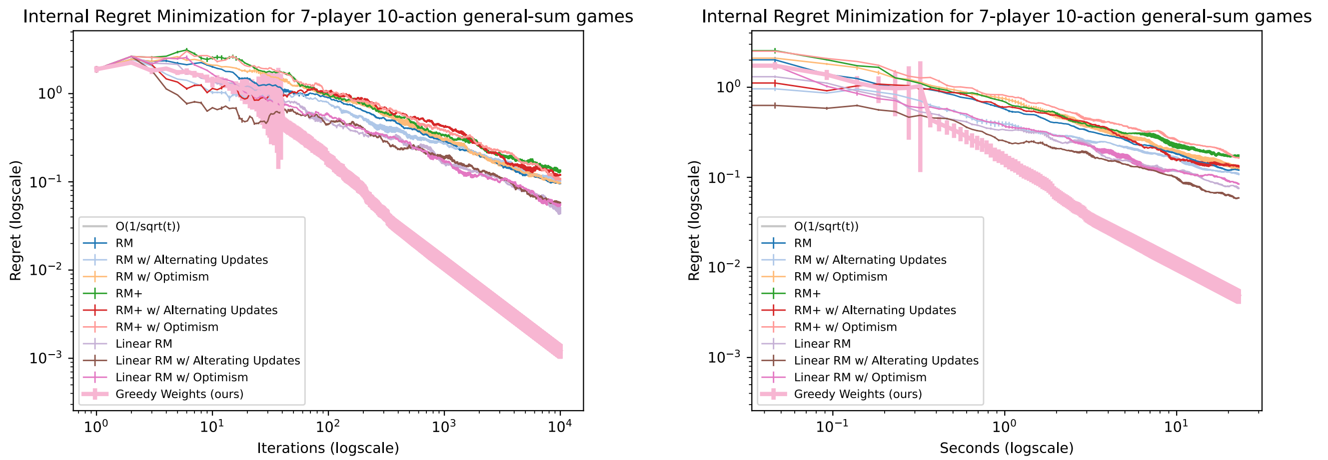
We generated 100 random 7-player 10-action general-sum games with payoff entries randomly sampled between 0 and 1 and ran both the standard and greedy weights variants of regret matching for 1000 iterations each. Indeed, greedy weights RM converges to equilibria with higher social welfare: it finds equilibrium with average welfare of 4.16 ± 0.023, while vanilla RM finds equilibria with average welfare 3.50 ± 0.005 (95% confidence intervals).

Additionally, we evaluate greedy weights on all the normal-form games included in the popular game theory library OpenSpiel (Lanctot et al. 2019). For space reasons, these plots have been relegated to the appendix.

Results in Diplomacy

In addition to running experiments on randomly generated matrix games, we also benchmark greedy weights on subgames of the benchmark game of Diplomacy.

Diplomacy is a popular seven-player zero-sum board game that involves simultaneous moves and both cooperation and competition. Players decide whom to support and whom to betray in pursuit of majority control of the board. Diplomacy has a long history as a benchmark for AI research (Kraus and Lehmann 1988; Kraus, Ephrati, and Lehmann 1994; Kraus and Lehmann 1995; Johansson and Håård 2005; Ferreira, Cardoso, and Reis 2015) and has been a particularly active domain for research in recent years (Paquette et al. 2019; Anthony et al. 2020; Gray et al. 2020). Since the game state is fully observable each turn, and players act simultaneously,



(a) Regret v. Iteration

(b) Regret v. Time

Figure 4: We generate 10 seven-player general-sum games with the matrix payoff entries selected uniformly at random from $[0, 1)$ and run all the described state-of-the-art approaches for minimizing internal regret in the sampled setting. All methods except our newly proposed greedy weights roughly converge at the worst case bound rate of $O(\frac{1}{\sqrt{T}})$, while greedy weights converges several orders of magnitude faster. Figure 4a shows the results as a function of the number of iterations used, while Figure 4b shows it as a function of time for fair comparison as each iteration of greedy weights costs slightly more than previous methods for regret minimization. Note that both axes are logscale. Error bars are shown at 95% confidence.

each turn in Diplomacy can be viewed as a matrix game if there is a defined state value function. Moreover, since players are able to communicate before acting, it is possible for players' actions to be correlated.

Prior work on no-press Diplomacy (the variant where players are unable to communicate) has achieved human-level performance by using regret matching on each turn to approximate equilibrium play for just the current turn (Gray et al. 2020). In other words, each turn is viewed as a matrix game in which the payoffs to each player for each joint action are determined via a pre-trained value network. Since queries to this value network are relatively expensive and the number of players in Diplomacy is large, mixed-policy techniques that do not use sampling are intractable.

We run experiments on the computation of correlated equilibria and coarse correlated equilibria in Diplomacy. Additionally, we measure convergence to a Nash equilibrium in a two-player zero-sum variant of Diplomacy called France vs. Austria (FvA). In all of our Diplomacy experiments, each player chooses between the 10 actions that have highest probability in the publicly available policy network from (Gray et al. 2020). We use the value network from (Gray et al. 2020) to determine payoffs.

Our results in Figure 5 indicate faster convergence to both a coarse-correlated and a correlated equilibrium when compared to Linear RM. In the case of CCE, greedy weights is orders of magnitude faster. Additionally, Figure 6 demonstrates that in two-player zero-sum France vs. Austria subgames, greedy weights reaches the same level of convergence to a Nash equilibrium with 2-3x fewer iterations than Linear RM when using pure policies. Experiments in Diplomacy used a single CPU core and a single NVIDIA V100 GPU. The overhead time necessary for computing an optimal weight for greedy weights was negligible relative to the cost of querying the value neural network.

Conclusions and Future Work

We introduce greedy weights, a novel generalization of the regret minimization framework for learning equilibria in which each new iteration is greedily weighed to minimize the procedure's average potential function (which is a function of all player's regret). In contrast, all prior regret minimization algorithms weighed iterates according to a fixed schedule. In randomly generated matrix games, we demonstrate that greedy weights empirically converges to correlated equilibria several orders of magnitude faster than existing methods, as well as faster convergence to Nash in two-player zero-sum games. We also find that greedy weights tends to learn equilibria with higher social welfare than vanilla regret minimization methods in general-sum games. Finally, in the large-scale AI benchmark of Diplomacy, we show speedups in convergence to all of NE, CE and CCE compared to existing methods.

Several important directions remain for future work. In this paper we focused our evaluation of greedy weights on strategic-form games. While we evaluated greedy weights on normal-form subgames of the sequential game Diplomacy, a natural question is whether similar results can be obtained for general sequential games that cannot be modeled as a sequence of matrix games by extending greedy weights to counterfactual regret minimization (Zinkevich et al. 2008). We show some preliminary positive results in the Appendix for the sequential games of Kuhn and Leduc poker but describe several difficult remaining challenges in scaling to large sequential games, which we believe to be worthy of

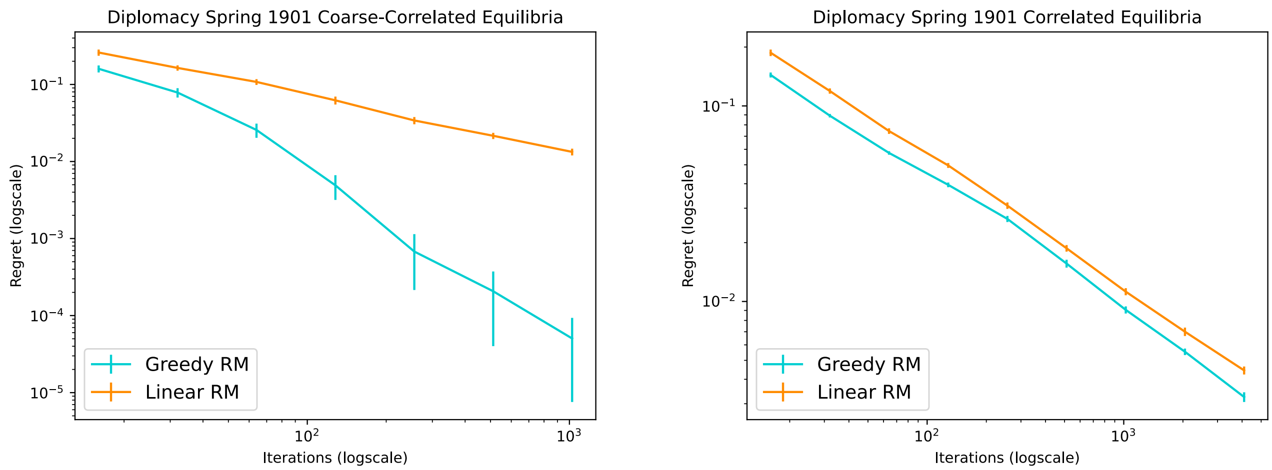
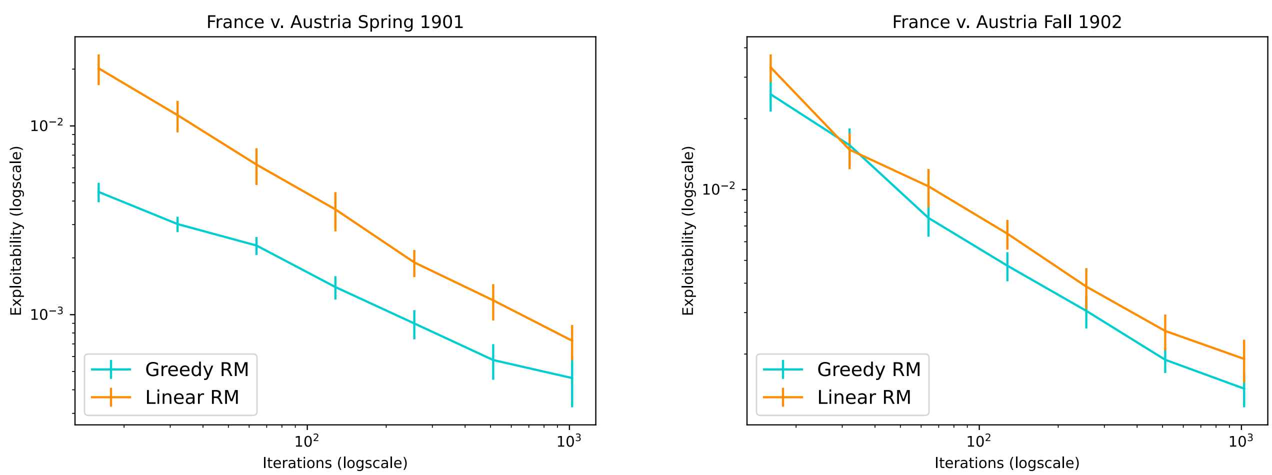


Figure 5: We benchmark greedy weights for computing both CCE and CE on the first turn of the seven-player game of Diplomacy which are computed by minimizing external and internal regret respectively. Greedy weights is significantly faster than in all cases considered, and when computing CCE, we find gains of several orders of magnitude. Note that both axes are logscale. Error bars denote 95% confidence intervals. Additional plots for Diplomacy can be found in the Appendix.



(a) Spring 1901 FvA

(b) Fall 1902 FvA

Figure 6: We also benchmark greedy weights on a two-player zero-sum variant of diplomacy called France v. Austria. Greedy weights converges 2-3x faster than Linear RM for computing Nash equilibria in this setting. Exploitability is the ϵ of an ϵ-Nash equilibrium. Note that the axes are both logscale. Error bars are at 95% confidence.

future investigation.

Another interesting direction for future work is investigating alternative dynamic weighting schemes. In this paper we describe dynamically weighing iterates by greedily minimizing the potential function. This is the first regret minimization algorithm to be introduced that dynamically adjusts the weights of iterations based on information obtained at runtime. However, in theory there are numerous other ways to dynamically weigh the iterates, such as adjusting the weights of past iterates or searching ahead to future iterates. It remains to be seen whether other algorithms that dynamically weigh iterates can lead to even better performance.

Finally, greedy weights (and regret minimization algorithms in general) cannot guarantee discovery of any *particular* equilibrium. While we show that greedy weights tends to find CE in general-sum game with higher average welfare , the question of whether regret minimization procedures can be extended to find specific desired equilibria, such as Pareto optimal equilibria or the equilibrium that optimizes the sum of player utilities, remains open for resolution.

References

Abernethy, J.; Bartlett, P. L.; and Hazan, E. 2011. Blackwell approachability and no-regret learning are equivalent. In *Proceedings of the 24th Annual Conference on Learning Theory*, 27–46. JMLR Workshop and Conference Proceedings.

Anthony, T.; Eccles, T.; Tacchetti, A.; Kramár, J.; Gemp, I.; Hudson, T. C.; Porcel, N.; Lanctot, M.; Pérolat, J.; Everett, R.; et al. 2020. Learning to play no-press diplomacy with best response policy iteration. *arXiv preprint arXiv:2006.04635*.

Aumann, R. J. 1974. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics*, 1(1): 67–96.

Blackwell, D. 1956. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1): 1–8.

Blum, A.; and Mansour, Y. 2007. From external to internal regret. *Journal of Machine Learning Research*, 8(Jun): 1307–1324.

Bowling, M.; Burch, N.; Johanson, M.; and Tammelin, O. 2015. Heads-up limit hold'em poker is solved. *Science*, 347(6218): 145–149.

Brown, N.; Lerer, A.; Gross, S.; and Sandholm, T. 2019. Deep counterfactual regret minimization. In *International conference on machine learning*, 793–802. PMLR.

Brown, N.; and Sandholm, T. 2018. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374): 418–424.

Brown, N.; and Sandholm, T. 2019a. Solving imperfect-information games via discounted regret minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 1829–1836.

Brown, N.; and Sandholm, T. 2019b. Superhuman AI for multiplayer poker. *Science*, 365(6456): 885–890.

Burch, N. 2017. *Time and Space: Why Imperfect Information Games are Hard*. Ph.D. thesis, University of Alberta.

Cesa-Bianchi, N.; and Lugosi, G. 2001. Potential-based algorithms in online prediction and game theory. In *International Conference on Computational Learning Theory*, 48–64. Springer.

Cesa-Bianchi, N.; and Lugosi, G. 2006. *Prediction, learning, and games*. Cambridge university press.

Chen, X.; Deng, X.; and Teng, S.-H. 2009. Settling the complexity of computing two-player Nash equilibria. *Journal of the ACM (JACM)*, 56(3): 1–57.

Daskalakis, C.; Goldberg, P. W.; and Papadimitriou, C. H. 2009. The complexity of computing a Nash equilibrium. *SIAM Journal on Computing*, 39(1): 195–259.

Farina, G.; Kroer, C.; and Sandholm, T. 2020. Faster Game Solving via Predictive Blackwell Approachability: Connecting Regret Matching and Mirror Descent. *arXiv preprint arXiv:2007.14358*.

Ferreira, A.; Cardoso, H. L.; and Reis, L. P. 2015. Dipblue: A diplomacy agent with strategic and trust reasoning. In *ICAART 2015-7th International Conference on Agents and Artificial Intelligence, Proceedings*.

Gray, J.; Lerer, A.; Bakhtin, A.; and Brown, N. 2020. Human-Level Performance in No-Press Diplomacy via Equilibrium Search. *arXiv preprint arXiv:2010.02923*.

Hannan, J. 1957. Approximation to Bayes risk in repeated play. *Contributions to the Theory of Games*.

Hart, S.; and Mas-Colell, A. 2000. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5): 1127–1150.

Hart, S.; and Mas-Colell, A. 2001. A general class of adaptive strategies. *Journal of Economic Theory*, 98(1): 26–54.

Johansson, S. J.; and Håård, F. 2005. Tactical coordination in no-press diplomacy. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, 423–430.

Kalai, A.; and Kalai, E. 2013. Cooperation in Strategic Games Revisited*. *The Quarterly Journal of Economics*, 128(2): 917–966.

Kraus, S.; Ephrati, E.; and Lehmann, D. 1994. Negotiation in a non-cooperative environment. *Journal of Experimental & Theoretical Artificial Intelligence*, 3(4): 255–281.

Kraus, S.; and Lehmann, D. 1988. Diplomat, an agent in a multi agent environment: An overview. In *IEEE International Performance Computing and Communications Conference*, 434–435. IEEE Computer Society.

Kraus, S.; and Lehmann, D. 1995. Designing and building a negotiating automated agent. *Computational Intelligence*, 11(1): 132–171.

Kuhn, H. W. 2016. 9. A SIMPLIFIED TWO-PERSON POKER. In *Contributions to the Theory of Games (AM-24), Volume I*, 97–104. Princeton University Press.

Lanctot, M.; Lockhart, E.; Lespiau, J.-B.; Zambaldi, V.; Upadhyay, S.; Pérolat, J.; Srinivasan, S.; Timbers, F.; Tuyls, K.; Omidshafiei, S.; et al. 2019. OpenSpiel: A framework for reinforcement learning in games. *arXiv preprint arXiv:1908.09453*.

McMahan, H. B.; Gordon, G. J.; and Blum, A. 2003. Planning in the presence of cost functions controlled by an adversary. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 536–543.

Monderer, D.; and Shapley, L. S. 1996. Potential games. *Games and economic behavior*, 14(1): 124–143.

Moravčík, M.; Schmid, M.; Burch, N.; Lisỳ, V.; Morrill, D.; Bard, N.; Davis, T.; Waugh, K.; Johanson, M.; and Bowling, M. 2017. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337): 508–513.

Nash, J. 1951. Non-cooperative games. *Annals of mathematics*, 286–295.

Neumann, J. v. 1928. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1): 295–320.

Paquette, P.; Lu, Y.; Bocco, S.; Smith, M. O.; Ortiz-Gagné, S.; Kummerfeld, J. K.; Singh, S.; Pineau, J.; and Courville, A. 2019. No Press Diplomacy: Modeling Multi-Agent Gameplay. *arXiv preprint arXiv:1909.02128*.

Rubinstein, A. 2019. *Hardness of Approximation Between P and NP*. Morgan & Claypool.

Serrino, J.; Kleiman-Weiner, M.; Parkes, D. C.; and Tenenbaum, J. 2019. Finding Friend and Foe in Multi-Agent Games. *Advances in Neural Information Processing Systems*, 32: 1251–1261.

Southey, F.; Bowling, M. P.; Larson, B.; Piccione, C.; Burch, N.; Billings, D.; and Rayner, C. 2012. Bayes' bluff: Opponent modelling in poker. *arXiv preprint arXiv:1207.1411*.

Syrgkanis, V.; Agarwal, A.; Luo, H.; and Schapire, R. E. 2015. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems*, 2989–2997.

Tammelin, O. 2014. Solving large imperfect information games using CFR+. *arXiv preprint arXiv:1407.5042*.

Zinkevich, M.; Johanson, M.; Bowling, M.; and Piccione, C. 2008. Regret minimization in games with incomplete information. In *Advances in neural information processing systems*, 1729–1736.