Regret-Grow CFR: A Novel Iterative Algorithm to Non-zero-Sum Game Nash Equilibrium

Anonymous Authors

Abstract

The two-player, imperfect information, poker card game Goofspiel is one of the most commonly-used benchmarks for testing equilibrium-finding algorithms. While Goofspiel is a qualified instance of imperfect information decision problems, it considers zero-sum cases exclusively, which is classified as one of the major limitations. And even non-zero-sum games are more general cases, they haven't received sufficient attention like their zero-sum counterparts for years. In this work, we examined how the traditional CFR algorithm behaves in selected information sets, and show that there are potential equilibrium points not reachable by CFR iterating. Then we characterized non-zero-sum games and objective functions and reformulated the game into a single-objective optimizing problem. It turns out that the problem generally falls into linear-quadratic programming category, whose convexity is typically not guaranteed. We also provided an iterative approach to converge to these equilibrium points, and compare them with CFR algorithm. It turns out that our iterative method is capable of finding equilibrium points that CFR sometimes fails to converge to, at a cost of augmenting the traditional iterative procedure by adding the exploitability minimizing mechanism, but computational overhead is still comparable with existing CFR.

Introduction

Non-zero-sum, imperfect information game is a theorized model to formulate many sequential move real-life decision problems. Imperfect-information games (IIGs) model strategic interactions among a set of participants who make decisions with imperfect information. Non-zero-sum (NZS) game is a game in which the sum of players' payoff is not zero, and typically not a constant. While quite a few poker variants are instances and common benchmarks for IIGs, few of them are capable of modeling the feature of NZS problems. In a NZS setting, the players' payoffs are mutually independent. Even solving a two-player NZS-IIG is harder than its zero-sum counterpart, because few existing poker variants pay attention to an NZS game setting.

Extensive form layout is common in large imperfect information games, Which resembles a decision tree in that players make decisions based on probability. For sufficiently large IIGs that are infeasible for a linear program, iterative algorithms are preferred because players' strategies often keep optimizing during a single affordable iteration. The ultimate goal is commonly to find a Nash Equilibrium (Nash 1950) in which no player can improve by unilaterally adjusting the strategy. Some special types of problems like zero-sum IIGs, already enjoy acceptable solutions by multiple algorithms. For example, 6-card and 10-card Leduc can be essentially be solved by First Order Method (Kroer et al. 2015). And CFR+ to solve heads-up limite Texas hold'em problem (Brown and Sandholm 2018) and find the baseline solution of heads-up no-limit Texas hold'em (HUNL) endgames in Libratus (Brown and Sandholm 2019). In order to solve the problem that exhibits ill-condition values, Tuomas el. al (Noam et al. 2019) proposed an discount mechanism for assign different wights for every iterations.(Silver et al. 2016, 2017, 2018; Schrittwieser et al. 2020)

Two major features made problems featured as NZS and IIG challenging. Firstly, the imperfect information makes players' payoff rely on their opponents. Since an action's payoff is an expectation from all possible cases, a strategy update from the opponent may cause a change in these cases' probability and that expectation. Thus players cannot simply take action with the highest payoff. Meanwhile, although there are known algorithms that can solve zero-sum games efficiently, whether these algorithms are still qualified for NZS-IIGs should be speculated. Since the independence of player payoffs, many algorithms, for example, Mini-Max, are no longer providing optimal solutions, because it assumes that one player's payoff is the opposite of the other's. Even for algorithms that are capable of approaching a Nash Equilibrium, the behavior of these algorithms near the equilibrium points, i.e. time complexity and stability should be studied rigorously.

In this paper, we focus on a two-player non-zero-sum game. To make things familiar, we customized the existing zero-sum poker card game goofspiel to a non-zero-sum variant. We proposed a hybrid iterating method inspired by Counterfactual Regret Minimization and Exploitability Descending. As all the players have their mixed strategies comprised of pure actions, the actions should have maximal therefore equal payoffs(Nash 1950). This allows solving normal form games by picking two (or more) pure actions, finding the opponent's probability distribution when these pure actions' payoffs coresult algorithmscking whether other not picked actions are all sub-optimal(Marc et al. 2009; Martin et al. 2019). Then the CFR algorithm is applied to solve customized goofspiel, we show how CFR behaves both at iterating and near equilibrium, and then there are cases in that CFR may skip and miss some equilibrium points and deviate toward other equilibrium. Then we proposed a novel methodology that fusion exploitability minimizing with existing CFR. Finally, we tested the algorithm on the same customized goofspiel and the algorithm also exhibits excellent converging behavior.

Related Work

Finding a Nash equilibrium is an important, interesting, and well-studied problem. Like two-player zero-sum games can be solved in polynomial time, finding (even an approximate) Nash equilibrium in a two-player non-zero-sum game is PPAD-complete (Chen, Deng, and Teng 2009), and a correlated equilibrium can be computed efficiently (Jiang and Leyton-Brown 2011). There are popular methods proposed for solving non-zero-sum imperfect information games, most of them can be categorized into vertex enumerating methods, reactive module methods, and no-regret methods.

The polytope vertex enumerating methods focus on extremely small-scaled games that can be converted into norm-form games. One of the methods, proposed by (Avis et al. 2009), is to enumerating all the possible combinations for actions with non-zero probability. The combinations of both sides should contain same amount actions. By establishing equations that let actions share the same payoff, the opponent's probability on each actions can be calculated. This method is based on the fact that the actions presented in a mixed strategy should have their payoffs equal, and equal to maximum of all legal actions' payoff. While this algorithm has rigorous mathematical basis and can be used to examine whether a strategy profile is a Nash equilibrium, two fatal limitations makes it not suitable for large non-zero-sum imperfect-information games: the algorithm is native to norm-form but not friendly to extensive-form, and even in norm-form, the computational time for an $n \times n$ norm-form is 4^n and not a polynomial time complexity. In principle, one can find all equilibria since the nonlinear equations are polynomials. The idea is to enumerate all supports, solve all roots of the polynomial equations, and select the solutions that correspond to probability distributions. The methods of finding all equilibria are probabilistic, that is, they will find all solutions with given probability when they are run for at least some amount of time (which depends on the probability). There are exponentially many supports in the game and there can be exponentially many equilibria. Moreover, the homotopy methods (global Newton, tracing procedure, or quantal response method) are not guaranteed to find all equilibria.

The homotopy methods that use the global Newton method do not converge globally. Govindan and Wilson observe that the iterated polymatrix approximation method typically converges globally but is not failsafe and may get stuck in some games. They find that the problem with homotopy methods is that they need to traverse nonlinear paths and require many small steps in order to obtain reasonable accuracy. They also ob- serve that the homotopy path may have many twists and reversals. Goldberg et al. construct examples where homotopy methods will not only need an exponential number of pivots but also an exponential number of direction reversals. Herings and van den Elzen and Herings and Peeters present a globally convergent homotopy method but note that the triangulations must have very refined mesh and the homotopy path must be traced numerically. Reinforcement learning, focusing on end-to-end solution, are also used in NZSGs. Gutierrez et. al (Gutierrez Julian and Michael 2000) studied non-zero-sum n-player games in which the choices available to players are defined using the Simple Reactive Modules Language (SRML), a subset of Reactive Modules (Alur and Henzinger 1999), a popular and expressive system modeling language that is used in several practical modeling checking systems (e.g., MOCHA (Alur et al. 1998) and Prism (Kwiatkowska, Norman, and Parker 2011)). Reactive Modules support succinct and high-level modeling of concurrent and multi-agent systems. In the games we study, the preferences of system components are specified by associating with each player in the game a temporal logic (LTL) formula that the player desires to be satisfied. Reactive Modules Games with perfect information (where each player can see the entire system state) have been extensively studied (Gutierrez, Harrenstein, and Wooldridge 2015a).

The regret based methods are proposed in early 2010s, by making strategies proportional to the positive regret values. And by minimizing all actions' regret on all the information sets, the global regret value can be minimized by minimizing the immediate regret values, or simply put, per-information-set no-regret method. This makes the actions easily computed at drastically low computational overhead. However, Zinkevich et al uses average strategy as output, while instant strategy at each iterations (its state variable) varies wildly, which is detrimental of numerical stability and convergence when the algorithm are applied on non-zero-sum games. Also, the naive method of taking average strategy makes it hard and intractable to rule out sub-optimal actions. While the algorithms runs good on zero-sum game, its intrinsic numerical defects brings it into technical difficulties on non-zero-sum games. This is because of the independence of payoffs breaks the correlation of players' payoff, and hence near-equilibrium payoff has one-degree-lower flatness than non-zero-sum games. What makes it more challenging is that the backward induction, what was used in perfect games, is unable to find the best action. This is because the perfect information games allows induction, which always go extreme and produces pure strategies as their equilibrium points, which is typically not the case in imperfect games. Although there are proofs shows that CFR can converge to Nash equilibrium in zero-sum games, and the necessary condition for CFR converges is exactly Nash equilibrium, the proof for sufficiency that CFR will converge is still absent.

Preliminaries

An **imperfect-information** game have both normal form and extensive-form, in this paper we use both of them. For extensive-form, the game is represented by a decision tree start from the root. There is a set for all the **players** called P. Each node is identified by a sequence of all actions taken through the path root to themselves called h for **history**, root has its history empty. In classical definition, each node has a player, who makes the **action** $a \in A$ if any actions are available. Joint decision nodes which have multiple players make decisions simultaneously are possible, which is a embedded norm form game into extensive form, and can drastically reduce the complexity when the game has sub-games. Every actions leads to child nodes that represent game states after they are committed. Let H to be the set of all the histories, for nodes identified as h and h', if node h' is child or n-th generation child node of h, then it is called $h \sqsubseteq h'$. For the **leaf nodes** who has no available actions and terminates the game, therefore no child nodes, their history sequences are not any prefix of other histories, we use $Z \in H$ for represent these nodes. All players will receive a **payoff** or reward when the game reaches leaf nodes. We call $u_i(z)$ for what player i can receive at leaf node z. We denote the range of payoffs in the game by Δ, and Δ_i represent the difference between maximal and minimal payoffs for player $i \in P$.

In imperfect-information games, since actions are not guaranteed to be observed by all the players, there are different nodes whose history appears identical view by some players. Such set of nodes are called **information sets** S. Apparently, all nodes $n \in S$ have same player i, which is not the case conversely. However, all nodes of same player can be first aggregated into information sets, and all information sets can be aggregated into **information collection** I_i for every player i. It will later show that the information sets, not nodes, are the minimum units for formulating strategy problems, which we call $A(S)$ that all available actions on information sets.

In extensive game, the player choose action by a stochastic manner, at each information set S, all the players assign a distribution on each available action a. Every player has its **strategy** σ_i that is a mapping that maps every information sets $S \in I_i$ with a vector $\mathbf{R}^{|A(S)|}$, namely, $\sigma_i(S) \in \mathbf{R}^{|A(S)|}$. It is common that σ_{-i} is used as other players' strategy. The set of all players' strategy, σ, is called strategy profile.

Nash Equilibrium

Let $u_i(\sigma_i, \sigma)$ to be the player i's payoff. The Nash equilibrium is a strategy profile σ that every unilateral changes in σ_i profile will not increase $u_i(\sigma_i, \sigma_{-i})$, i.e.

$$\forall i, u_i(\sigma_i, \sigma_{-i}) = \max_{\sigma_i'} u_i(\sigma_i', \sigma_{-i}) \tag{1}$$

For measuring how far the players are deviating from the equilibrium, exploitablility is defined as:

$$ep_i(\sigma_i, \sigma_{-i}) = \max_{\sigma_i'} u_i(\sigma_i', \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) \tag{2}$$

$$ep(\sigma) = \sum_{i \in P} ep_i(\sigma_i, \sigma_{-i}) \tag{3}$$

By definition, we have:

$$ep(\sigma^*) = \sum_{i \in P} ep_i(\sigma_i^*, \sigma_{-i}^*) = 0 \tag{4}$$

The two-player norm form games focus on single information set. Each player's payoff can be defined as entries in two matrices, let (A, B) be a binary tuple of m-by-n matrices, therefore m and n are numbers of their available actions. Let x and y as strategies of both players. both have their entities non-negative and sum to be exactly 1. As a natural result, u and v as their payoffs.

Best response condition

Let x and y be the mixed strategies of both players. Then those actions whose probabilities are non-zero have their payoffs to be maximal, and therefore mutually equal to others.

$$x_i > 0 \iff a_i^T y = u = \max_i(a_i^T y) \tag{5}$$

where a_i are row vectors of matrix A, and:

$$y_j > 0 \iff b_j^T x = v = \max_j(b_j^T x) \tag{6}$$

where b_j are column vectors of matrix B.

What it has alleviated is the infinite mixed strategy problem to finite-dimension inequalities formation, which however at the cost of numerical behavior of best responses. The collection of best responses would drop at almost all but one of its elements even if an opponent's strategy deviates a little. Nevertheless, the algorithm can be used reversely, say, for example, player 1, not to find what pure actions are the collection I of best pure strategies should play 1 player against player 2's y, but when I is potentially possible to become the collections of player's y.

Methodology

To make both eq(5) and eq(6) have a unique solution, for example, if there are k non-zero entries in x, namely x_1, and the rest of zero-entries x_0, the linear problem should contain exactly k equations. Let y_1 be a non-zero part of y. From $\mathbf{C_n^k}$ possible different y_1s, they form $k \times k$ linear equation, which is required by the uniqueness of the solution.

Clearly, this method provides Nash equilibrium points at the cost of NP-hard, by enumerating all the 1 to $\min(m, n)$, it requires all the $2^{\min(m,n)}$. Conversely, the counterfactual minimization method provides $\mathcal{O}(\frac{1}{\epsilon^2})$. So the NonZeroSumMatrix method is only tractable in small sized information sets, and should be act as benchmark to test whether other algorithms could find equilibrium in test-size problem.

In a nonzero-sum game, minimax is no longer optimal, because it wrongly assumes that both play-ers use the same payoff function. Nonetheless, A's minimax does guarantee the worst-case outcome for A, because it proceeds as if B would always choose the worst possible moves against A. Therefore, minimax is used as the baseline for comparisons in our examples. More generally, we consider imperfect information nonzero-sum games, in which players can have incomplete mutual knowledge and thus SPE does not apply.

Algorithm 1: Regret Grow CFR

Input: A, B

Output: strategy profile x, y

```
function Non-zero-sum-Matrix ($A, B$):
$m, n$ = A.shape
for $k \leftarrow 1$ to $\min(m, n)$ do
    for $I : sum(I) = k, I \in R^m, I_i \in 0, 1$ do
        for $J : sum(J) = k, J \in R^n, J_j \in 0, 1$ do
            $y = A^{-1}_{[I,J]}\mathbf{1}$
            if $0 \leq y \leq 1, Ay \leq 1$ then
                $I_{best} = y$
            end if
        end for
    end for
    for $J : sum(J) = k, J \in R^n, J_j \in 0, 1$ do
        for $I : sum(I) = k, I \in R^m, I_i \in 0, 1$ do
            $x = B^{-1}_{[I,J]}\mathbf{1}$
            if $0 \leq x \leq 1, B^T x \leq 1$ then
                $J_{best} = x$
            end if
        end for
    end for
    for $I : sum(I) = k, I \in R^m, I_i \in 0, 1$ do
        for $J : sum(J) = k, J \in R^n, J_j \in 0, 1$ do
            $J' = I_{best-J} = where(I_{best} > 0)$
            $I' = J'_{best-I} = where(J'_{best} > 0)$
            if $I' = I$ then
                $x = I_{best}$
                $y = J_{best}$
            end if
        end for
    end for
end for
return $x, y$
```

Inspired the existing algorithms, the counterfactual regret minimization is slightly different naively applying gradient-based optimization method, but used convex-combinations instead. For any x

$$\sum_{i=1}^{m} x_i = 1, 0 \leq x \leq 1 \tag{7}$$

and similar y, there exist:

$$u(x, y) = x^T Ay \tag{8}$$

$$v(x, y) = x^T By \tag{9}$$

In typical CFR+ algorithm, since Taylor expansions is valid when iteration T approaching to infinity, let R as regret on all the actions, as the rule of the iteration have:

$$x' = \frac{R + r}{\sum R + \sum r} \tag{10}$$

let

$$p = \frac{r}{\sum r} \tag{11}$$

$$x' = \frac{\sum Rx + \sum rp}{\sum R + \sum r} \tag{12}$$

$$x' = \frac{\sum Rx + \sum rx - \sum rx + \sum rp}{\sum R + \sum r} \tag{13}$$

$$x' = x + \frac{\sum r}{\sum R + \sum r} p \tag{14}$$

$$x' \approx x + \frac{\sum r}{\sum R} p \tag{15}$$

Since the all the regret vector r comes from strictly positive actions that has better response for opponent, the payoff functions $u(x', y) > u(x, y)$ always holds.

$$R^{T+1}(I, a) = \begin{cases} R^T_+(I, a) + r^T(I, a), & \text{if } r^T(I, a) \geq 0 \\ R^T_+(I, a) + \epsilon r^T(I, a), & \text{otherwise} \end{cases} \tag{16}$$

where $\epsilon \in [0, 1]$, and R is truncated if negative, i.e.

$$R^{T+1}_+(I, a) = \begin{cases} R^{T+1}(I, a), & \text{if } R^{T+1}(I, a) \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

Definition 1 Let f_i from actions $i \in \{1, 2, ..., m\}$ to be approximation of best strategy y, player's strategy $\hat{p}$ is convex combination of f_i, and a non-negative loss function $\ell(\hat{p}, y)$. Then the instantaneous regret value for $\hat{p}$ deviating away from f_i is defined as $r_i = \ell(\hat{p}) - \ell(f_i)$.

Definition 2 *If approximation above is repeated for n times, then the cumulative loss functions for player and advisor $i \in \{1, 2, ..., m\}$ are defined as $\hat{L}_n = \sum\limits_{t=1}^{n} \ell(\hat{p}_t, y_t)$ and $L_{i,n} = \sum\limits_{t=1}^{n} \ell(f_{i,t}, y_t)$ respectively, and the cumulative regret is defined as $R_{i,n} = \sum\limits_{t=1}^{n} r_{i,t} = \hat{L}_n - L_{i,n}$*

Theorem 1 *Let ϕ to be function from R to R_+ is a non-negative, convex and increasing function, then* $\sup\limits_{y_t} \sum\limits_{i=1}^{N} r_{i,t}\phi'(R_{i,t-1}) \leq 0$

Proof:

Since $\phi'(R_{i,t-1}) > 0$, using Jensen's inequality for all y,

$$\ell(\hat{p}_t, y) = \ell\left(\frac{\sum\limits_{i=1}^{N} \phi'(R_{i,t-1}) f_{i,t}}{\sum\limits_{j=1}^{N} \phi'(R_{j,t-1})}, y\right) \leq \frac{\sum\limits_{i=1}^{N} \phi'(R_{i,t-1})\ell(f_{i,t}, y)}{\sum\limits_{i=1}^{N} \phi'(R_{j,t-1})}$$

Lemma 1 *Let $\boldsymbol{r}_t = (r_{1,t}, r_{2,t}, ..., r_{N,t}) \in R^N$ to be instantaneous regret vector, and cumulative regret vector $\boldsymbol{R}_n = \sum\limits_{t=1}^{n} \boldsymbol{r}_t$. Then the potential function $\Phi : R^N \rightarrow R_+$ is defined as $\Phi(\boldsymbol{u}) = \psi\left(\sum\limits_{i=1}^{N} \phi(u_i)\right)$, where $\phi : R \rightarrow R_+$ is any non-negative increasing function, and $\psi : R_+ \rightarrow R_+$ is any non-negative function for scaling purpose with strictly increasing and concave properties. Then $\hat{p}_t = \frac{\nabla\Phi(\boldsymbol{R}_{t-1}) \cdot \boldsymbol{f}_t}{\sum\limits_{j=1}^{N} \nabla\Phi(\boldsymbol{R}_{t-1})_j}$, and theorem 7.3 is equivalent to* $\sup\limits_{y_t} \boldsymbol{r}_t \cdot \nabla\Phi(\boldsymbol{R}_{t-1}) \leq 0$

Lemma 2 $\Phi(\boldsymbol{R}_n) \leq \Phi(0) + \frac{1}{2}\sum_{t=1}^{n} C(\boldsymbol{r}_t)$, *where* $C(\boldsymbol{r}_t) = \sup_{\boldsymbol{u}\in R^N}\left[\psi'\left(\sum_{i=1}^{N}\phi(u_i)\right)\sum_{i=1}^{N}\phi''(u_i)r_{i,t}^2\right]$

Proof:

$$\Phi(\mathbf{R}_t) = \Phi(\mathbf{R}_{t-1} + \mathbf{r}_t)$$

$$= \Phi(\mathbf{R}_{t-1}) + \nabla\Phi(\mathbf{R}_{t-1})\cdot\mathbf{r}_t + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial^2\Phi}{\partial u_i\partial u_j}r_{i,t}r_{j,t}$$

$$\leq \Phi(\mathbf{R}_{t-1}) + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial^2\Phi}{\partial u_i\partial u_j}r_{i,t}r_{j,t}$$

where the second-order term of Taylor expansion shows that

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial^2\Phi}{\partial u_i\partial u_j}r_{i,t}r_{j,t}$$

$$\leq \psi''\left(\sum_{i=1}^{N}\psi(\xi_i)\right)\sum_{i=1}^{N}\sum_{j=1}^{N}\psi'(\xi_i)\psi'(\xi_j)r_{i,t}r_{j,t}$$

$$+\psi'\left(\sum_{i=1}^{N}\psi(\xi_i)\sum_{i=1}^{N}\psi''(\xi_i)r_{i,t}^2\right)$$

$$= \psi''\left(\sum_{i=1}^{N}\psi(\xi_i)\right)\left(\sum_{i=1}^{N}\psi'(\xi_i)r_{i,t}\right)^2$$

$$+\psi'\left(\sum_{i=1}^{N}\psi(\xi_i)\sum_{i=1}^{N}\psi''(\xi_i)r_{i,t}^2\right)$$

$$\leq C(\mathbf{r}_t)$$

Theorem 2 *For any convex loss function* ℓ*, if it takes values in [0, 1], if scaling function* ψ *is polynomial weighted function, then for any sequence* $y_1, y_2, ..., y_n$ *the loss function have* $\hat{L}_n - \min_{i=1,2,...,N} L_{i,n} \leq \sqrt{n(p-1)N^{2/p}}$*, which also means that regret value is* $o(n)$ *when* $n \rightarrow \infty$

Proof:

Since $\psi'(x) = (x^{\frac{2}{p}})' = \frac{2}{px^{(p-2)/p}}$, and $\phi''(x) = (x_+^p)'' = p(p-1)x_+^{p-2}$, where x_+ floors negative components to zero while keeps the positive component. By Holder inequality,

$$\sum_{i=1}^{N}\psi''(u_i)r_{i,t}^2$$

$$\leq p(p{-}1)\left(\sum_{i=1}^{N}\left((u_i)_+^{p-2}\right)^{p/(p-2)}\right)^{(p-2)/p}\left(\sum_{i=1}^{N}|r_{i,t}|^p\right)^{2/p}$$

Thus,

$$\psi\left(\sum_{i=1}^{N}\psi(u_i)\right)\sum_{i=1}^{N}\psi''(u_i)r_{i,t}^2$$

$$\leq 2(p-1)\left(\sum_{i=1}^{N}|r_{i,t}|^p\right)^{2/p}$$

which means that

$$\Phi_p(\mathbf{R}_n) \leq (p-1)\sum_{i=1}^{N}||\mathbf{r}_t||_p^2$$

$\leq n(p-1)N^{2/p}$ which means that the regret grows only sub-linearly. i.e. $\frac{\mathbf{R}_T}{T} \rightarrow 0$ when $T \rightarrow \infty$, it's asymptotically approaching to best response of player should follow with.

Intuition of the Regret-Grow CFR

What inspired the CFR is the convex combination with better actions, however, what to be maximized is a multiple-objective $u_1(x,y)$ and $u_2(x,y)$. The exploitability provides no-exploitability method just like existing CFR+ algorithm, which can also reformulate two-player non-zero-sum games into single object optimization problem:

$$\max_{x,y}(u(x,y) - \max_{p}u(p,y) + v(x,y) - \max_{q}v(x,q)) \quad (18)$$

For the iterating method, we added exploitability terms for augmenting the existing CFR method:

$$r_{xi} = u(x_i, y) - u(x, y) \quad (19)$$

$$r_{x\,epy\,i} = -\max_{q} x_i, q \quad (20)$$

$$x' = x + \frac{r_x}{\sum R_x} + \frac{r_{x\,epy}}{\sum R_{x\,epy}} \quad (21)$$

$$y' = y + \frac{r_y}{\sum R_y} + \frac{r_{y\,epx}}{\sum R_{y\,epx}} \quad (22)$$

This will perform maximize the player's own payoff, and minimize the opponent's exploitability, which degenerates to existing CFR when the problem is just zero-sum cases.

Experiments

We evaluate the performance of regret-grow CFR on Goofspiel. Goofspiel is a bidding card game where players have a hand of cards numbered 1 to N, and take turns secretly bidding on the top point-valued card in a point card stack using cards in their hands. While all of these games have imperfect information and are roughly of similar size, they are a diverse set of games, varying both in the degree (the ratio of the number of information sets to the number of histories) and nature (whether due to chance or opponent actions) of imperfect information.

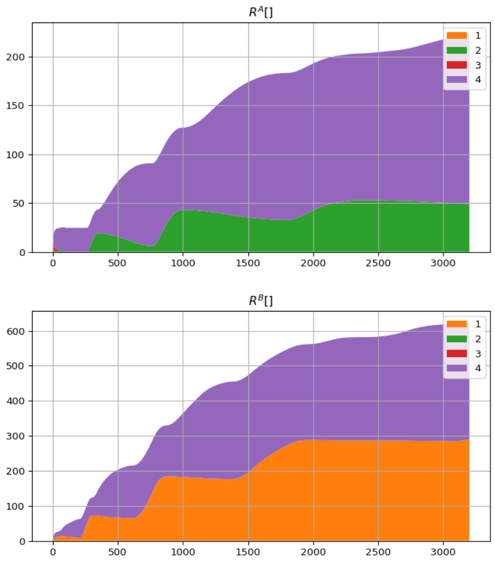

Figure 1: Strategies and regrets for players to decide their first cards.

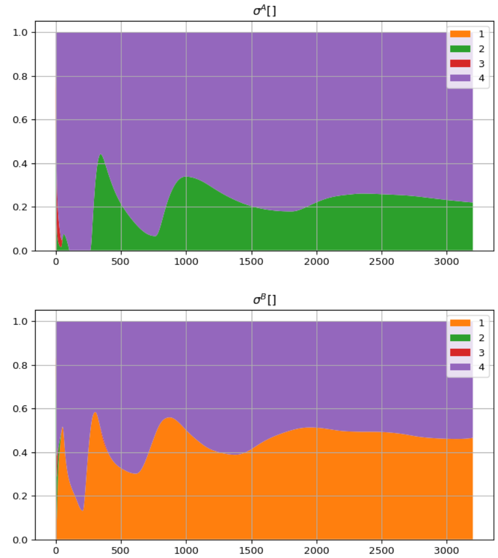

Figure 2: Strategies and regrets for players to decide their first cards.

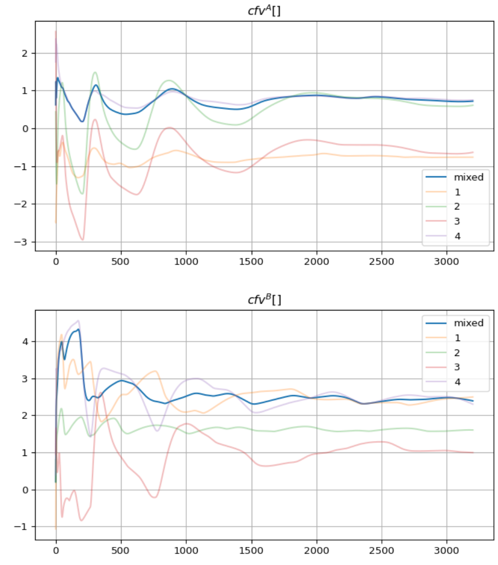

Figure 3: How payoff of pure and mixed strategies evolves at the first card.

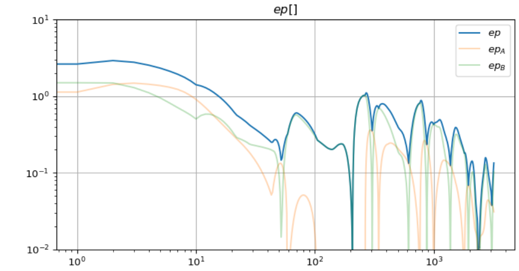

Figure 4: How exploitability or A and B and their sum are minimized at the first card.

The equilibrium points

We also used subgames of Goofspiel variation as a test bed for the techniques introduced in this paper. First, for finding the equilibrium points, we used a different assessment of the effectiveness of the algorithm in terms of availability than the original one. The new evaluation criterion is defined as a head-to-head comparison between the adversarial sides, weighted differently, and the contrasting algorithms respectively, counting the final benefit of both sides.

Since that customized variant of Goofspiel was used as a test-bed for the techniques of Regret-Grow-CFR, in this experiment, all the goofspiel upcards are treated as 1, but weighted as [5.00, 1.33, 2.71, 4.27] for player 1, and [4.10, 6.28, 3.33, 3.84] for player 2. When one player wins a card, this contributes their payoff by how the card weighted by this player, while the other's decrease by how this card weighted by that player.

The first chart reveals how both players make decision at first card, this is a mixed strategy profile, which suggests both the players bet their largest card. The empirical converge rate at won't take effect on initial few turns, rather, it diverge away final equilibrium point by 0.771 for player 1 and 0.479 for player 2.

The result of how the players deal with their second card is presented in Figure 3 and 4. First, their strategies are converging, and therefore the equilibrium point's strategy profile is found. This can also be verified from the view of regret controlling - for both players and at each point where they make decisions, the sum of regrets for all available actions grows sub-linearly, this is also empirically verified the regret-based theories.

An interesting result from Figure 3 and Figure 4 is, once entered into equilibrium, both players' strategies, typically a mixed one, is consisted of the same number of pure strategies, however, this is a natural result of Nash theorem, since the pure strategy, will almost always have a unique best response from the opponent, which will never make opponent take a mixed strategy.

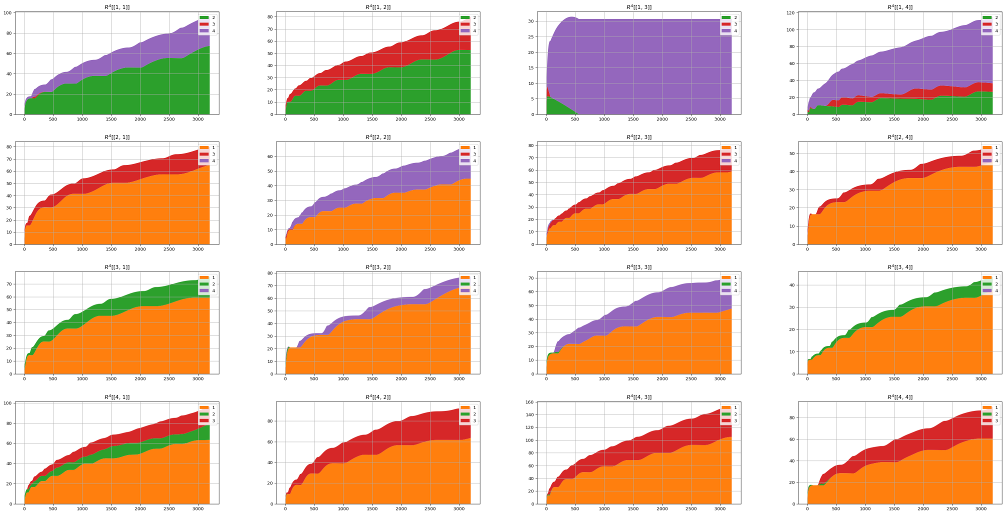

Figure 5: player A, regret values for available actions

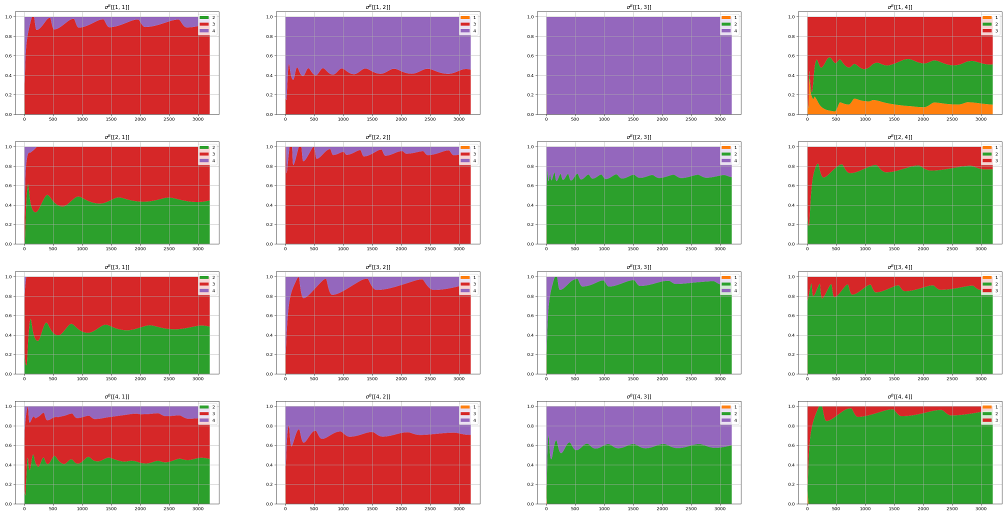

Figure 6: player B, strategy distribution on available actions

For example, an equilibrium that both players' strategies are mixed strategy that contains 2 actions: If player A adopts 2 actions in his mixed strategy, then player B will almost always adopt a mixed strategy that contains 2 actions. The reason can be discussed by considering 3 cases.

(1) If B's strategy contains only one action, i.e. pure strategy, then that is impossible. Because the action that B chose will typically not make 2 actions of A have same payoffs.

(2) Mixed strategy response from B that contain 2 actions is possible, since B's some partitions of probability can make A's two actions have the same payoff.

(3) Mixed strategy response from B that contain 3 actions or beyond is also impossible. If the additional action is inferior to 2 existing actions, this sub-optimal action will be ruled out, resulting in a 2 × 2 mixed strategy equilibrium. If the additional action is better than the 2 existing actions, then B will prefer the new action, shifting and eventually landing at a new equilibrium.

In all the cases above, strategies from either side must contain the same number of pure strategies, or actions.

The result of how the players deal with their second card is presented in Figure 3 and 4. First, their strategies are converging, and therefore the equilibrium point's strategy profile is found. This can also be verified from the view of regret controlling - for both players and at each point where they make decisions, the sum of regrets for all available actions grows sub-linearly, this is also empirically verified the regret-based theories.

From figure 3, 7, and 8, it can be verified that the curve of cfv values always try to follow the curve of actions with the highest payoff, using a steady yet fast method to follow. If two or more actions are the best responses that comprised the mixed strategy, their payoffs compete and take turns to lead other actions. Be sure to not confuse it with a

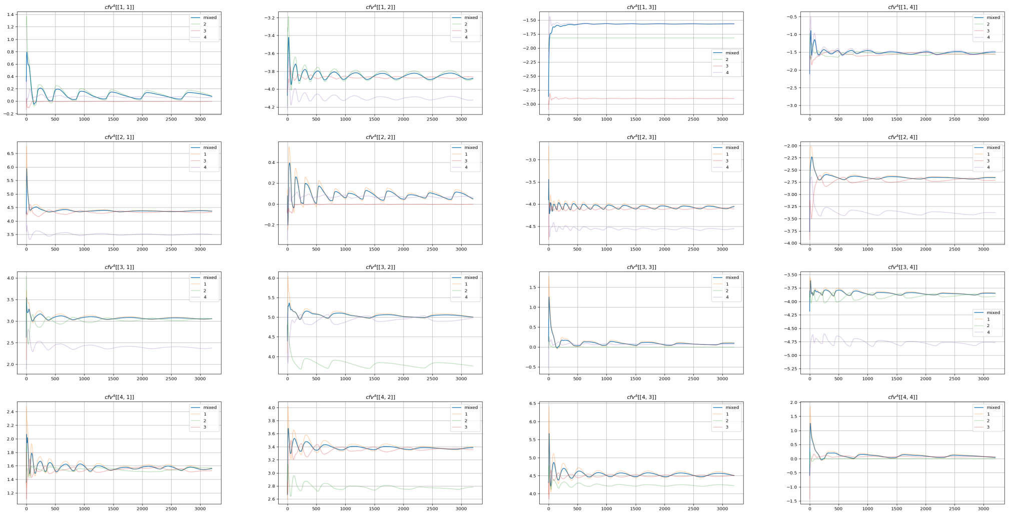

Figure 7: player A in subgames, payoff for strategies and actions

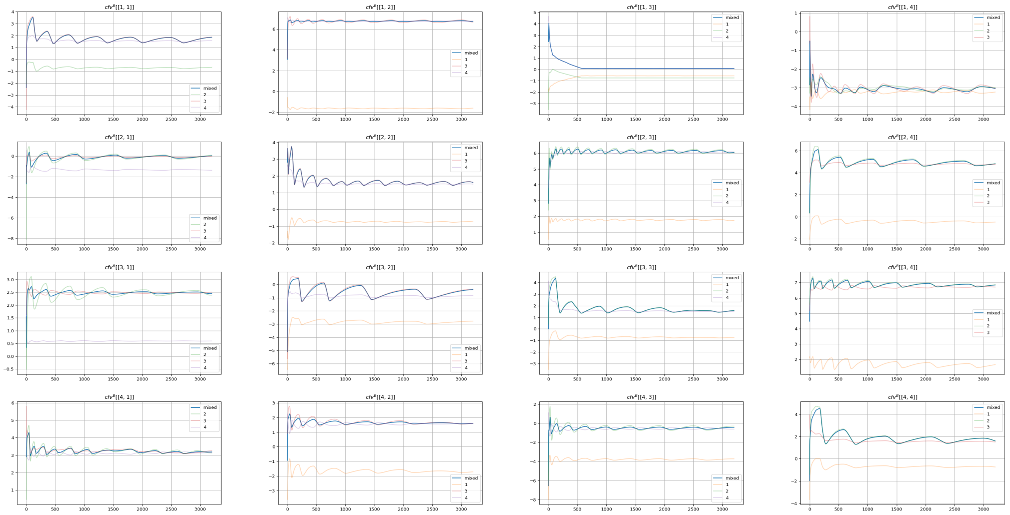

Figure 8: player B in subgames, payoff for strategies and actions

truly sub-optimal action, which disadvantage is permanent and can never be overturned.

While the payoffs of actions can be oscillating wildly, the curve of cfv adopts a fast-yet-smooth pattern to realize payoff maximization and guarantee convergence. This is because the negative instant regret is vanishing and ruling out the sub-optimal action, and also because the regrets sum across the actions' are growing yet sub-linearly, which makes the updating step length smaller across the times, yet allow the significant updates influencing the subsequent iterations. The limited-forgetting mechanism makes the regret grows over time, and limiting the step length of updating strategy; yet still makes the regret grows strictly slower than linear increasing, which may force the strategy not to be a sub-optimal one, and therefore guarantees converging to a Nash equilibrium.

Behavior near Subgame equilibrium

Since we considered the problem of computing an equilibrium solution for non-zero-sum games. The most common solution concept is the Nash equilibrium. For $\epsilon \geq 0$, a strategy profile σ is an ϵ - Nash equilibrium if no player deviates from it.

The importance of deeper analyzing the endgame of Goofspiel variation lies in testing the behavior of Regret-Grow-CFR, especially, at some vicinity of any possible equilibrium points. In our experiments, this results in a different assessment of the effectiveness of the algorithm in terms of availability than the original one. The new evaluation criterion is defined as a head-to-head comparison between the adversarial sides, weighted differently, and the contrasting algorithms respectively, counting the final benefit of both sides.

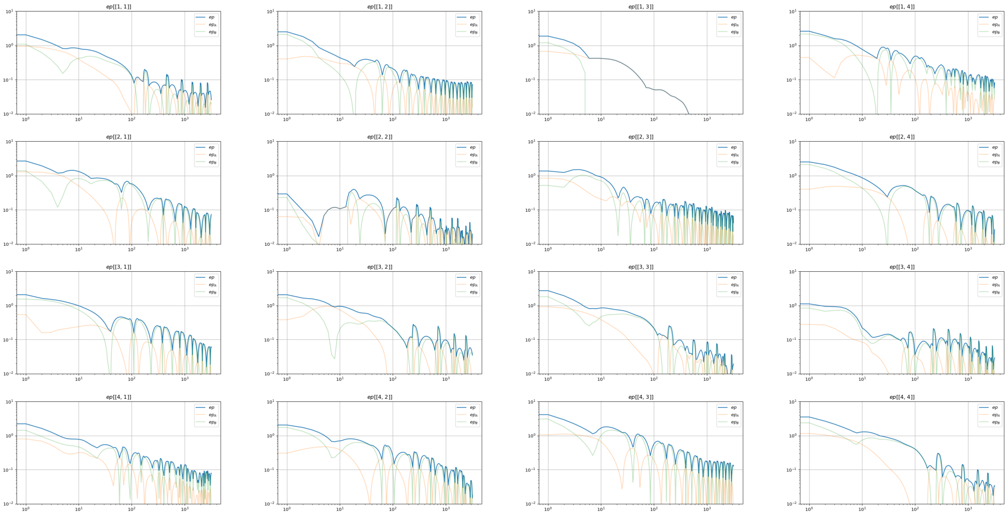

Figure 9: exploitability, for both players and their sum

Figure 5 and 6 have shown 16 distinct subgames in which player A and B deal with high loss, and show how their strategies evolved. They had also shown how the negative influence of bad initial guess is dissolved when the strategy was significantly off-equilibrium, which results can be seen in Figure 7 and 8.

While higher card weight draws much attention from both players, the chaining logic makes players deliberately give up the high-weighted cards by throwing low-ranking cards, this can be verified in the subgame [(1, 1)]. For the converging process, the average time lengths for ruling out dominated strategy and perform a fully-updated strategy for both players are 33.94 and 59.67 epochs respectively. More specifically, as Figure 1, 5, and 6 shown, player A bet more often his card-4 for a 5.00 reward, while player B bet a little mixed strategy, which throws card-1 at 34.7%, but concentrates more on his second card for a 6.28 reward. In all the selected 16 subgames, both players' exploitability value and that of the sum are shrinking, and that time complexity is no more than the time complexity of $O(1/T^2)$, which is plotted in dual-logarithm coordinates in Figure 9.

In the experiment, The first chart reveals how both players make a decision at the first card, this is a mixed strategy profile, which suggests both the players bet their largest card. The empirical converge rate at won't take effect on the initial few turns, rather, it diverges away the final equilibrium point by 0.771 for player A and 0.479 for player B.

The game goofspiel variant is a two-player imperfect game that have multiple subgame nodes, on which all players reveal their actions so that players have full knowledge about their current situations. This make goofspiel simple yet capable to catch the complexity of the imperfect games. From equation (15), it is clear that the process of exploitability minimizing is the only necessary condition for a ϵ-Nash equilibrium. This effectively boost the process to both inspect and verify the convergent behavior and their speed near Nash equilibrium.

The exploitability of both players at the subgame [(1, 3), (2, 3)] has three possible equilibrium points, however, only (1, 0) became the converging limit of CFR algorithm.

For example, player A and B at the information set have thrown cards (1, 3) and (2, 3) respectively, the strategy profile approaching to the equilibrium point (0.825, 0.175), (0.318, 0.682) at first, but since it is not a CFR-stable saddle, the CFR iteration process shift away and headed towards (1, 0), (0, 1) instead. The numerical error terms mainly come from our algorithm accumulating regret in the very beginning of the game is played, they should shrink to 0 asymptotically when T approaches infinity.

Conclusion and Future Works

In this paper, we introduced a novel variant of CFR whose latest iteration is capable of converging to the Nash Equilibrium. In order to efficiently rule out dominated strategy, regret-grow CFR adopts a partial-forget mechanism to update the cumulative regrets, yet guarantees the cumulative regret vector's element sum increasing. We tested regret-grow CFR on non-zero sum Goofspiel and its subgames, which showed the exploitability values for both players are descending and in the magnitude of ep $O(1/T^2)$. Since the regret-grow CFR does not require averaging the previous strategies, dominated strategies can be ruled out, which makes probability distribution on information sets a sparse vector most of the time.

In further research, we would like to examine how the payoff of the game can affect the algorithms' convergence. Such analysis could offer further algorithmic or theoretical improvements on algorithm stability. It is also promising yet challenging to adapt regret-grow CFR with the Monte-Carlo sampling traversing method. If the nondeterministic character does not affect the stability of regret-grow CFR significantly, it would be interesting to compare the approach, in terms of the payoff of both players and exploitability values.

References

Avis, D.; Rosenberg, G. D.; Savani, R.; and von Stengel, B. 2009. Enumeration of nash equilibria for two-player games. *Economic Theory*, 42(1): 9–37.

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. 2019. Superhuman AI for multiplayer poker. *Science*, 365(6456): 885–890.

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

Gutierrez Julian, P. G.; and Michael, W. 2000. Imperfect information in reactive modules games. *Association for the Advancement of Artificial Intelligence*.

Jiang, A. X.; and Leyton-Brown, K. 2011. Polynomial-time computation of exact correlated equilibrium in compact games. *Proceedings of the 12th ACM conference on Electronic commerce - EC '11*.

Kroer, C.; Waugh, K.; Kilinç-Karzan, F.; and Sandholm, T. 2015. Faster First-Order Methods for Extensive-Form Game Solving. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15, 817–834. New York, NY, USA: Association for Computing Machinery. ISBN 9781450334105.

Marc, L.; Kevin, W.; Martin, Z.; and H, B. M. 2009. Monte Carlo Sampling for Regret Minimization in Extensive Games. In *23th Conference on Neural Information Processing Systems*, 1078–1086. Vancouver, British Columbia, Canada.

Martin, S.; Neil, B.; Marc, L.; Matej, M.; Rudolf, K.; and Michael, B. 2019. Variance reduction in monte carlo counterfactual regret minimization (VR-MCCFR) for extensive form games using baselines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2157–2164. Honolulu, Hawaii USA: PKP.

Nash, J. F. 1950. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1): 48–49.

Noam, B.; Adam, L.; Sam, G.; and Tuomas, S. 2019. Deep counterfactual regret minimization. In *International conference on machine learning*, 793–802. Long Beach, Calif, USA: International Conference on Machine Learning (ICML).

Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. 2020. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489.

Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419): 1140–1144.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359.