Online Convex Optimization for Sequential Decision Processes and Extensive-Form Games[*]

Gabriele Farina and **Christian Kroer** and **Tuomas Sandholm**
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
{gfarina,ckroer,sandholm}@cs.cmu.edu

Abstract

Regret minimization is a powerful tool for solving large-scale extensive-form games. State-of-the-art methods rely on minimizing regret locally at each decision point. In this work we derive a new framework for regret minimization on sequential decision problems and extensive-form games with general compact convex sets at each decision point and general convex losses, as opposed to prior work which has been for simplex decision points and linear losses. We call our framework *laminar regret decomposition*. It generalizes the CFR algorithm to this more general setting. Furthermore, our framework enables a new proof of CFR even in the known setting, which is derived from a perspective of decomposing polytope regret, thereby leading to an arguably simpler interpretation of the algorithm. Our generalization to convex compact sets and convex losses allows us to develop new algorithms for several problems: regularized sequential decision making, regularized Nash equilibria in extensive-form games, and computing approximate extensive-form perfect equilibria. Our generalization also leads to the first regret-minimization algorithm for computing reduced-normal-form quantal response equilibria based on minimizing local regrets. Experiments show that our framework leads to algorithms that scale at a rate comparable to the fastest variants of counterfactual regret minimization for computing Nash equilibrium, and therefore our approach leads to the first algorithm for computing quantal response equilibria in extremely large games. Finally we show that our framework enables a new kind of scalable opponent exploitation approach.

Introduction

Counterfactual regret minimization (CFR) (Zinkevich et al., 2007), and the newest variant *CFR$^+$* (Tammelin et al., 2015), have been a central component in several recent milestones in solving imperfect-information *extensive-form games (EFGs)*. Bowling et al. (2015) used CFR$^+$ to near-optimally solve heads-up limit Texas hold'em. Brown and Sandholm (2017) and Moravčík et al. (2017) used CFR variants, along with other scalability techniques, to create AIs that beat professional poker players at the larger game of heads-up no-limit Texas hold'em.

We can view the CFR approach more generally as a methodology for setting up regret minimization for sequential decision problems (whether single- or multi-agent), where each decision point requires selecting either an action or a point from the probability distribution over actions. The crux of CFR is counterfactual regret, which leads to a definition of regret local to each decision point. CFR can then be viewed as the observation, and proof, that bounds on counterfactual regret, which can be minimized locally, lead to bounds on the overall regret. To minimize local regret, the framework relies on regret minimizers that operate on a simplex (typically of probabilities over the available actions), such as *regret matching* (RM) (Blackwell, 1956) or the newer variant *regret matching$^+$* (RM$^+$) (Tammelin et al., 2015).

In this paper we consider the more general problem of how to minimize regret over a sequential decision-making (SDM) polytope, where we allow arbitrary compact convex subsets of simplexes at each decision point (as opposed to only simplexes in CFR), and general convex loss functions (as opposed to only linear losses in CFR). This allows us to model a form of online convex optimization over SDM polytopes. We derive a decomposition of the polytope regret into local regret at each decision point. This allows us to minimize regret locally as with CFR, but for general compact convex decision points and convex losses. We call our decomposition *laminar regret decomposition (LRD)*. We call our overall framework for convex losses and compact convex decision points *laminar regret minimization (LRM)*. As a special case, our framework provides an alternate view of why CFR works—one that may be more intuitive for those with a background in online convex optimization.

Our generalization to general compact convex sets (we restrict our attention to convex subsets of simplexes, but this is without loss of generality due to the convexity-preserving properties of affine transformations) allows us to model entities such as ϵ-perturbed simplexes (Farina and Gatti, 2017; Farina, Kroer, and Sandholm, 2017; Kroer, Farina, and Sandholm, 2017), and thus yields new algorithms for computing approximate equilibrium refinements for EFGs.

General convex losses in SDM and EFG contexts have, to the best of our knowledge, not been considered before. This generalization enables fast algorithms for many new settings. One is to compute regularized zero-sum equilibria. If we apply a convex regularization function at each simplex, we can apply our framework to solve the resulting game. For the negative entropy regularizer this is equivalent to the dilated

[*] Will appear at AAAI'19.

entropy distance function used for solving EFGs with first-order methods (Hoda et al., 2010; Kroer et al., 2015, 2017). Ling, Fang, and Kolter (2018) show that dilated-entropy-regularized EFGs are equivalent to quantal response equilibria (QRE) in the corresponding reduced normal-form game. Thus our result yields the first regret-minimization algorithm for computing reduced-normal-form quantal response equilibria in EFGs.

Our experiments show that QREs and ℓ_2-regularized equilibria can be computed at a rate that is competitive with that of CFR$^+$ for computing Nash equilibria, and substantially faster in some cases. This shows that our approach can be used to compute regularized equilibria in extremely large games such as real poker games. We go on to show that our framework also enables a new kind of opponent-exploitation approach for extremely large games, by adding a convex regularizer that penalizes the exploiter for being far away from a pre-computed Nash equilibrium, and thus potentially exploitable herself.

Regret Minimization

We work the online learning framework called *online convex optimization* (Zinkevich, 2003). In this setting, a decision maker repeatedly plays against an unknown environment by making a sequence of decisions $x^1, x^2, \ldots$. As customary, we assume that the set $X \subseteq \mathbb{R}^n$ of all possible decisions for the decision maker is convex and compact. The outcome of each decision x^t is evaluated as $\ell^t(x^t)$, where ℓ^t is a convex function *unknown* to the decision maker until the decision is made. Abstractly, a *regret minimizer* is a device that supports two operations:

- it gives a *recommendation* for the next decision $x^{t+1} \in X$;
- it receives/observes the convex loss function ℓ^t used to "evaluate" decision x^t.

The learning is *online* in the sense that the decision maker/regret minimizer's next decision, x^{t+1}, is based only on the previous decisions $x^1, \ldots, x^t$ and corresponding loss observations $\ell^1, \ldots, \ell^t$.

In this paper, we adopt (external) *regret* as a way to evaluate the quality of the regret minimizer. Formally, the *cumulative regret* at time T is defined as

$$R^T := \sum_{t=1}^{T} \ell^t(x^t) - \min_{\hat{x} \in X} \sum_{t=1}^{T} \ell^t(\hat{x}),$$

It measures the difference between the loss cumulated by the sequence of decisions $x^1, \ldots, x^T$ and the loss that would have been cumulated by playing the best time-independent decision $\hat{x}$ in hindsight. A desirable property of a regret minimizer is *Hannan consistency*: the average regret approaches zero, that is, R^T grows at a *sublinear* rate in T.

We now review a particular very general regret-minimization algorithm: *online mirror descent* (OMD). The generality of OMD arises because it performs updates in the dual space, where the duality is given by a *mirror map* Φ, a strongly-convex differentiable function on X which defines a vector field in which gradient updates are performed. At each time step OMD performs the following update:

$$\nabla\Phi(y_{t+1}) = \nabla\Phi(y_t) - \eta\nabla\ell^t(x_t),$$

and then recommends the point

$$x_{t+1} = \arg\min_{x \in X} \Phi(x) - \langle \nabla\Phi(y_{t+1}), x\rangle.$$

If OMD is initialized with $\nabla\Phi(y_1) = 0$ and x_1 as the corresponding minimizer, it satisfies the regret bound

$$R^T \leq \max_{u,v \in X} \{\Phi(u) - \Phi(v)\} + \eta \sum_{t=1}^{T} \|\nabla\ell^t(x_t)\|_*^2,$$

where $\|\cdot\|_*$ is the dual norm with respect to which Φ is strongly convex. OMD is very general in the sense that we can choose Φ and the norm for measuring strong convexity so that it fits the problem at hand. For example, this allows only a logarithmic dependence on the dimension of X when $X = \Delta_n$ and Φ is the negative entropy. By specific choices of η and Φ it is possible to show that this algorithm generalizes online variants of gradient descent, exponential weights, and regularized follow-the-leader (Zinkevich, 2003; Hazan and Kale, 2010; Hazan, 2016). The regret generally grows at a rate of $T^{-1/2}$ for these algorithms.

We could also run OMD with X being the entire SDM polytope. For example, we could do that by applying the *distance generating function (DGF)* of Kroer et al. (2017). However, decomposition into local regret minimization at each decision point has been dramatically more effective in practice, possibly because this allows better leveraging of the structure of the problem.

Linear losses and games. Regret minimization methods for normal-form and extensive-form games usually involve minimizing the regret induced by linear loss functions. When the domain at each decision point X_j is the n_j-dimensional simplex Δ_{n_j}, the two most successful regret-minimizers in practice have been *regret matching* (Blackwell, 1956) and *regret matching*$^+$ (Tammelin et al., 2015). These regret minimizers also have regret that grows at a rate $T^{-1/2}$ as with OMD, but they have a worse dependence on the dimension n_j. Nonetheless, they seem to perform better in practice when coupled with CFR.

Sequential Decision Making

It turns out that the results of this paper can be proven in a general setting which we call a sequential decision making. At each stage, the agent chooses a point in a simplex (or a subset of it). The chosen point incurs a convex loss and defines a probability distribution over the actions of the simplex. An action is sampled according to the chosen distribution, and the agent then arrives at a next decision point, potentially randomly selected out of several candidates. The reason the agent chooses points in the convex hull of actions, rather than simply an action, is that this gives us greater flexibility in representing decision points where agents wish to randomize over actions. This is the case for example in game-theoretic equilibria or when solving the decision-making problem with an iterative optimization algorithm.

Formally, we assume that we have a set of decision points $\mathcal{J}$. Each decision point $j \in \mathcal{J}$ has a set of actions A_j of size n_j. The decision space at each decision point j is represented by a convex set $X_j \subseteq \Delta_{n_j}$. A point $x_j \in X_j$ represents a probability distribution over A_j. When a point x_j is chosen, an action is sampled randomly according to x_j. Given a specific action at j, the set of possible decision points that the agent may next face is denoted by $\mathcal{C}_{j,a}$. It can be an empty set if no more actions are taken after j, a. We assume that the decision points form a tree, that is, $\mathcal{C}_{j,a} \cap \mathcal{C}_{j',a'} = \emptyset$ for all other convex sets and action choices j', a'. This condition is equivalent to the perfect-recall assumption in extensive-form games, and to conditioning on the full sequence of actions and observations in a finite-horizon partially-observable decision process. In our definition, the decision space starts with a root decision point, whereas in practice multiple root decision points may be needed, for example in order to model different starting hands in card games. Multiple root decision points can be modeled in our framework by having a dummy root decision point with only a single action.

The set of possible next decision points after choosing action $a \in A_j$ at X_j, denoted $\mathcal{C}_{j,a}$, can be thought of as representing the different decision points that an agent may face after taking action a and then making an observation on which she can condition her next action choice. For example, in a card game an action may be to raise (that is, put money into the pot), and an observation could be the set of actions taken by the other players, as well as any cards dealt out, until the agent acts again. Each specific observation of actions and cards then corresponds to a specific decision point in $\mathcal{C}_{j,a}$.

We will relate the regret over the whole decision space to regret at subtrees in the decision space and individual convex sets. In order to do that we need ways to refer to each of these structures. Given a strategy x, x_j is the (sub)vector belonging to the decision space X_j at decision point j. Similarly, $x_{j,a}$ is the scalar associated with action $a \in A_j$ at decision point j, and in typical applications it is the probability of choosing action a at decision point j. Subscript $\triangle_j$ denotes the portion of x containing the decision variables for decision point j and all its descendants. Finally, x refers to the vector for the whole treeplex, which corresponds to subscript $\triangle_r$ where r is the root of the tree.

As an illustration, consider the game of Kuhn poker (Kuhn, 1950). Kuhn poker consists of a three-card deck: king, queen, and jack. Each player is dealt one of the three cards and a single round of betting occurs. A complete game description is given in the online appendix. The action space for the first player is shown in Figure 1. For instance, we have: $\mathcal{J} = \{0, 1, 2, 3, 4, 5, 6\}$; $n_0 = 1, X_0 = \Delta_1 = \{1\}$; $n_j = 2, X_j = \Delta_2$ for all $j \in \mathcal{J} \setminus \{0\}$; $A_0 = \{\text{start}\}$, $A_1 = A_2 = A_3 = \{\text{check}, \text{raise}\}$, $A_4 = A_5 = A_6 = \{\text{fold}, \text{call}\}$; $\mathcal{C}_{0,\text{start}} = \{1, 2, 3\}$, $\mathcal{C}_{1,\text{raise}} = \emptyset$, $\mathcal{C}_{3,\text{check}} = \{6\}$; $X_{\triangle_1} = X_1 \times X_4$, $X_{\triangle_4} = X_4$; $x_1 = [x_{1,\text{check}}, x_{1,\text{raise}}]$; $x_{\triangle_1} = [x_1; x_4]$, etc.

In addition to games, our model captures, for example, POMDPs and MDPs where we condition on the entire history of observations and actions.

Regret in Sequential Decision Making

We assume that we are playing a sequence of T iterations of a sequential decision process. At each iteration t we choose a strategy $x \in X$ and are then given a loss function of the form

$$\ell^t(x) := \sum_{j \in \mathcal{J}} \pi_j(x) \ell^t_j(x_j), \tag{1}$$

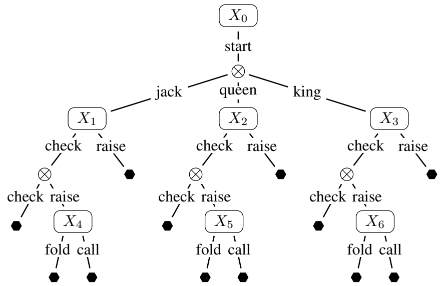

Figure 1: Sequential action space for the first player in the game of Kuhn poker. ⊗ denotes an observation point; ⬢ represents the end of the decision process.

where $\ell^t_j : X_j \to \mathbb{R}$ is a convex function for each $j \in \mathcal{J}$. We coin loss functions of this form *separable*, and they will play an important role in our results. Our goal is to compute a new strategy vector x^t such that the regret across all T iterations is as low as possible against any sequence of loss functions.

We now summarize definitions for the value and regret associated with convex sets and strategies. First we have the value of convex set j at iteration t when following strategy $\hat{x}$:

$$\hat{V}^t_{\triangle_j}(\hat{x}_{\triangle_j}) := \ell^t_j(\hat{x}^t_j) + \sum_{a \in A_j} \sum_{j' \in \mathcal{C}_{j,a}} \hat{x}_{j,a} \hat{V}^t_{\triangle_{j'}}(\hat{x}_{\triangle_{j'}}).$$

This definition denotes the utility associated with starting at convex set X_j rather than at the root. Thus we have exchanged the term $\pi_j(x)$ with one for ℓ^t_j and with $\hat{x}_{j,a}$ for $V^t_{\triangle_{j'}}$; this allows us to write the value as a recurrence. We will be particularly interested in the value of x^t, which we denote $V^t_{\triangle_j} := \hat{V}^t_{\triangle_j}(x^t_{\triangle_j})$.

Now we can define the cumulative regret at convex set j across all T iterations as

$$R^T_{\triangle_j} := \sum_{t=1}^{T} V^t_{\triangle_j} - \min_{\hat{x}_{\triangle_j}} \sum_{t=1}^{T} .\hat{V}^t_{\triangle_j}(\hat{x}_{\triangle_j}),$$

This can equivalently be stated as

$$\min_{\hat{x}_{\triangle_j}} \sum_{t=1}^{T} \hat{V}^t_{\triangle_j}(\hat{x}_{\triangle_j}) = \sum_{t=1}^{T} V^t_{\triangle_j} - R^T_{\triangle_j}. \tag{2}$$

Finally, *average regret* is $\bar{R}^T_{\triangle_j} = \frac{1}{T} R^T_{\triangle_j}$.

Laminar Regret Decomposition

We now define a new parameterized class of loss functions for each subtree X_j which we will show can be used to minimize regret over X by minimizing that loss function independently at each convex set X_j. The loss function is

$$\hat{\ell}^t_j(x_j) := \ell^t_j(x_j) + \sum_{a \in A_j} \sum_{j' \in \mathcal{C}_{j,a}} x_{j,a} V^t_{\triangle_{j'}}. \tag{3}$$

It is convex since ℓ^t_j is convex by hypothesis and we are only adding a linear term to it. Strict convexity is also preserved, and for strongly convex losses, the strong convexity parameter remains unchanged.

We now prove that the regret at information set j decomposes into regret terms depending on $\hat{\ell}^t_j$ and a sum over the regret at child convex sets:

Theorem 1. *The cumulative regret at a decision point j can be decomposed as*

$$R^T_{\triangle_j} = \sum_{t=1}^{T} \hat{\ell}^t_j(x^t) - \min_{\hat{x}_j \in X_j} \left\{ \sum_{t=1}^{T} \hat{\ell}^t_j(\hat{x}_j) - \sum_{a \in A_j} \sum_{j' \in \mathcal{C}_{j,a}} \hat{x}_{j,a} R^T_{\triangle_{j'}} \right\}$$

Proof. By definition, the cumulative regret $R^T_{\triangle_j}$ at time T for decision point j is:

$$\sum_{t=1}^{T} V^t_{\triangle_j} - \min_{\hat{x}_{\triangle_j}} \left\{ \sum_{t=1}^{T} \ell^t_j(\hat{x}_j) + \sum_{t=1}^{T} \sum_{a \in A_j} \sum_{j' \in \mathcal{C}_{j,a}} \hat{x}_{j,a} \hat{V}^t_{\triangle_{j'}}(\hat{x}_{\triangle_{j'}}) \right\}$$

$$= \sum_{t=1}^{T} V^t_{\triangle_j} - \min_{\hat{x}_j \in X_j} \left\{ \sum_{t=1}^{T} \ell^t_j(\hat{x}_j) \right.$$

$$\left. + \sum_{a \in A_j} \sum_{j' \in \mathcal{C}_{j,a}} \hat{x}_{j,a} \min_{\hat{x}_{\triangle_{j'}}} \sum_{t=1}^{T} \hat{V}^t_{\triangle_{j'}}(\hat{x}_{\triangle_{j'}}) \right\}, \qquad (4)$$

where the equalities follow first from expanding the definitions of $R^T_{\triangle_j}$ and $\hat{V}^t_{\triangle_j}(\hat{x}_{\triangle_j})$, and then using the fact that we can sequentially minimize first over choices at j and then over choices for child information sets.

Now we can use (2) to get that (4) is equal to

$$= \sum_{t=1}^{T} V^t_{\triangle_j} - \min_{\hat{x}_j \in X_j} \left(\sum_{t=1}^{T} \hat{\ell}^t_j(\hat{x}_j) - \sum_{a \in A_j} \hat{x}_{j,a} \sum_{j' \in \mathcal{C}_{j,a}} R^t_{\triangle_{j'}} \right). \quad (5)$$

Since $V^t_{\triangle_j}$ already depends on $V^t_{\triangle_{j'}}$ for each child decision point j' we have $V^t_{\triangle_j} = \hat{\ell}^t_j(x^t_j)$, where the equality follows by the definition of $\hat{\ell}^t_j$. Substituting this equality in (5) yields the statement. □

Theorem 1 justifies the introduction of the concept of *laminar regret* at each decision point $j \in \mathcal{J}$:

$$\hat{R}^T_j := \sum_{t=1}^{T} \hat{\ell}^t_j(x^t_j) - \min_{\hat{x}_j \in X_j} \sum_{t=1}^{T} \hat{\ell}^t_j(\hat{x}_j).$$

With this, we can write the cumulative subtree regret at decision point j as a sum of laminar regret at j plus a recurrence term for each child decision point. Applying this inductively gives the following theorem which tells us how one can apply regret minimization locally on laminar regrets in order to minimize regret in SDMs:

Theorem 2. *The cumulative regret on X satisfies*

$$R^T \leq \max_{\hat{x} \in X} \sum_{j \in \mathcal{J}} \pi_j(\hat{x}) \hat{R}^T_j.$$

Corollary 1. *If each individual laminar regret $\hat{R}^T_j$ on each of the convex domains X_j grows sublinearly, overall regret on X grows sublinearly.*

Theorem 2 shows that overall regret can be minimized by minimizing each laminar regret separately. In particular, this means that if we have a regret minimizer for each decision point j that can handle the structure of the convex set X_j and the convex loss from (3), then we can apply those regret minimizers individually at each information set, and Theorem 2 guarantees that overall regret will be bounded by a weighted sum over those local regrets. For example, if each local regret minimizer has regret that grows at a particular sublinear rate, then the overall regret is also guaranteed to grow only at that sublinear rate.

Our result gives an alternative proof of CFR. This is arguably simpler than existing proofs, because we show directly why regret over a sequential decision-making space decomposes into individual regret terms, as opposed to bounding terms in order to fit the CFR framework. Finally, our result also generalizes CFR to new settings: we show how CFR can be implemented on arbitrary convex subsets of simplexes and with convex losses rather than linear.

Sequence form for sequential decision processes

So far we have described the decision space as a product of convex sets where the choice of each action is taken from a subset of a simplex $X_j \subseteq \Delta_{n_j}$. This formulation has a drawback: the expected-value function for a given strategy is not linear. Consider taking action a at decision point j. In order to compute the expected overall contribution of that decision, its local payoff $g_{j,a}$ has to be weighted by the product of probabilities of all actions on the path to j and by $x_{j,a}$. So, the overall expected utility is nonlinear and non-convex. We now present a well-known alternative representation of this decision space which preserves linearity. While we will mainly be working in the product space X, it will occasionally be useful to move to this equivalent representation to preserve linearity.

The alternative formulation is called the *sequence form*. In that representation, every convex set $j \in \mathcal{J}$ is scaled by the parent variable leading to j. In other words, the sum of values at j now sum to the value of the parent variable. In this formulation, the value of a particular action then represents the probability of playing the whole *sequence* of actions from the root to that action. This allows each term in the expected loss to be weighted only by the sequence ending in the corresponding action. The sequence form has been used to instantiate linear programming (von Stengel, 1996) and first-order methods (Hoda et al., 2010; Kroer et al., 2015, 2017) for computing Nash equilibria of zero-sum EFGs. There is a straightforward mapping between any $x \in X$ to its corresponding sequence form: simply assign each sequence the product of probabilities in the sequence. Likewise, going from sequence form to X can be done by dividing each $x_{j,a}$ by the value x_{p_j} where p_j is the entry in x corresponding to the parent of j. We let μ be a function that maps each $x \in X$ to its corresponding sequence-form vector. For the reverse direction μ^{-1}, there is ambiguity because μ is not injective. Nonetheless, an inverse can be computed in linear time.

Application Domains

Because we only need a finite sequential tree structure of the decision space, our framework captures a very broad class of SDM problems. In this section we describe how our framework can be applied to a number of prominent applications, such as POMDPs and EFGs. In general, our framework can be applied to any SDM problem where one or more agents are faced with a finite sequence of decisions that form a tree, such that agents always remember all past actions. The specific decision problem at each stage may depend on the past decisions as well as stochasticity. The fact that we require the decision space to be tree structured might seem limiting from the perspective of compactly representing the decision space. However, this has successfully been dealt with in applications by using state- or value-estimation techniques, rather than fully representing the original problem (Moravčík et al., 2017; Jin, Levine, and Keutzer, 2018).

One example class of a single-agent decision problems that we can model is finite-horizon POMDPs where the history of states and actions is remembered. In that case, each decision point corresponds to a specific sequence of actions and observations made by the agent. This setting is reminiscent of the POMDP setting considered by Jin, Levine, and Keutzer (2018). This type of model can be used to model sequential medical treatment planning when combined with results on imperfect-recall abstraction (Lanctot et al., 2012; Chen and Bowling, 2012; Kroer and Sandholm, 2016a), and has potential applications in steering evolutionary adaptation (Sandholm, 2015; Kroer and Sandholm, 2016b). Our framework allows more general models for such problems via our generalization to convex decision points and convex losses; for example our framework could be used for regularized models. For instance in a medical settings, we may want to regularize the complexity of the treatment plan.

Extensive-form games with convex-concave saddle-point structure

In an extensive-form game with perfect recall each player faces a sequential decision-making problem, of the type described in the previous section and in Figure 1. The set of next potential decision points $\mathcal{C}_{j,a}$ is based on observations of stochastic outcomes and actions taken by the other players.

Here, we will focus on two-player zero-sum EFGs with perfect recall, but with slightly more general utility structure than is usually considered. In particular, we assume that we are solving a convex-concave saddle-point problem of the following form:

$$\min_{x\in X}\max_{y\in Y}\left\{\mu(x)^\top A\mu(y) + d_1(\mu(x)) - d_2(\mu(y))\right\}, \qquad (6)$$

where X is the SDM polytope for Player 1 and Y is the SDM polytope of Player 2. Each d_i is assumed to be a dilated convex function of the form

$$d_i(\mu(x)) = \sum_{j\in\mathcal{J}} \mu(x)_{p_j} d_j\left(\frac{\mu(x)_j}{\mu(x)_{p_j}}\right) = \sum_{j\in\mathcal{J}} \pi_j(x)\ell_j(x_j),$$

that is, in the form given in (1).

In standard EFGs, the loss function for each player at each iteration t is defined to be the negative payoff vector associated with the sequence-form strategy of the other player at that iteration; since we additionally allow a *regularization* term we also get a nonlinear convex term. More formally, at each iteration t, the loss functions $\ell_X^t : X \to \mathbb{R}$ and $\ell_Y^t : Y \to \mathbb{R}$ for player 1 and 2 respectively are defined as

$$\ell_X^t : x \mapsto \langle -A\mu(y^t), \mu(x)\rangle + d_1(x),$$
$$\ell_Y^t : y \mapsto \langle A^\top\mu(x^t), \mu(y)\rangle + d_2(y),$$

where A is the sequence-form payoff matrix of the game (von Stengel, 1996). Some simple algebra shows that ℓ_X^t and ℓ_Y^t are indeed separable (that is, they can be written in the form of Equation 1), where each decision-point-level loss $\ell_{j,X}^t$ and $\ell_{j,Y}^t$ is a convex function.

This choice of loss function is justified by the fact that the induced regret-minimizing dynamics for the two players lead to a convex-concave saddle-point problem. Specifically, assume the two players play the game T times, accumulating regret after each iteration as in Figure 2.

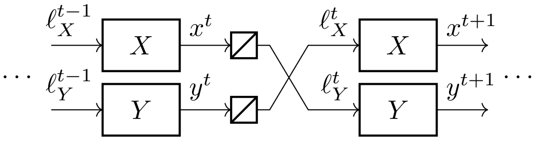

Figure 2: The flow of strategies and losses in regret minimization for games. The symbol $\boxslash$ denotes computation/construction of the loss function.

A folk theorem explains the tight connection between low-regret strategies and approximate Nash equilibria. We will need a more general variant of that theorem generalized to (6). The convergence criterion we are interested in is the *saddle-point residual (or gap)* ξ of $(\bar{x}, \bar{y})$, defined as

$$\xi = \max_{\hat{y}}\{d_1(\bar{x}) - d_2(\hat{y}) + \langle\bar{x}, A\hat{y}\rangle\} - \min_{\hat{x}}\{d_1(\hat{x}) - d_2(\bar{y}) + \langle\hat{x}, A\bar{y}\rangle\}$$

We show that playing the average of a sequence of regret-minimizing strategies leads to a bounded saddle-point residual. This result is probably known, but it is unclear whether it has been stated in the form here. We provide a proof in the online appendix. A closely related form is used for averaged strategy iterates in a first-order method by Nemirovski (2004).

Theorem 3. *If the average regret accumulated on X and Y by the two sets of strategies $\{x_t\}_{t=1}^T$ and $\{y_t\}_{t=1}^T$ is ϵ_1 and ϵ_2, respectively, then any strategy profile $(\bar{x}, \bar{y})$ such that $\mu(\bar{x}) = \frac{1}{T}\sum_{t=1}^T \mu(x^t)$, $\quad \mu(\bar{y}) = \frac{1}{T}\sum_{t=1}^T \mu(y^t)$ has a saddle-point residual bounded by $\epsilon_1 + \epsilon_2$.*

The above averaging is performed in the sequence-form space, which works because that space is also convex. After averaging we can easily compute $\bar{x}$ in linear time. Hence, by applying LRD to the decision spaces X and Y, we converge to a small saddle-point residual. The fact that the averaging of the strategies is performed in sequence form explains why the traditional CFR presentation requires averaging with weights based on the player's reaches π_j at each decision point j.

Theorem 3 shows that a Nash equilibrium can be computed by taking the uniform distribution over sequence-form strategy iterates. However, in the practical EFG-solving literature another approach called *linear averaging* has been popular Tammelin et al. (2015), especially for the CFR$^+$ algorithm. In linear averaging a weighted average strategy is constructed, where each strategy $\mu(x^t)$ is weighted by t. Tammelin et al. (2015) show that this is guaranteed to converge specifically when using the RM$^+$ regret minimizer. It would be interesting to prove when this works more generally. Here we make the simple observation that we can compute *both* averages, and simply use the one with better practical performance, even in settings where only the uniform average is guaranteed to converge.

Quantal response equilibrium (QRE) Ling, Fang, and Kolter (2018) show that a reduced-normal-form QRE can be expressed as the convex-concave saddle-point problem (6) where d_1 and d_2 are the (convex) dilated entropy functions usually used in first-order methods (FOMs) for solving EFGs (Hoda et al., 2010; Kroer et al., 2015, 2017). This saddle-point problem can be solved using FOMs, which would lead to fast convergence rate due to the strongly convex nature of the dilated entropy distance (Kroer et al., 2017). However, until now, no algorithms based on local regret minimization at each decision point have been known for this problem. Because the dilated entropy function separates into a sum over negative entropy terms at each decision point it can be incorporated as a convex loss in LRD. Combined with any regret-minimization algorithm that allows convex functions over the simplex, this leads to the first regret-minimization algorithm for computing (reduced-normal-form) QREs.

Perturbed EFGs and equilibrium refinement Equilibrium refinements are Nash equilibria with additional important rationality properties. Such equilibria have rarely been used in practice due to scalability issues. Recently, fast algorithms for computing approximate refinements were introduced Kroer, Farina, and Sandholm (2017); Farina, Kroer, and Sandholm (2017). Theorem 1 gives a new tool for constructing such methods: it immediately implies correctness of the method of Farina, Kroer, and Sandholm (2017), while also allowing new types of refinements and regret minimizers.

Erratum about Alternation in CFR$^+$

Several tweaks to speed up the convergence of CFR have been proposed. The state of the art is CFR$^+$ (Tammelin et al., 2015). CFR$^+$ consists of three tweaks: the RM$^+$ regret minimizer, linear averaging, and alternation. RM$^+$ can be applied in our setting as well; it is simply an alternative regret minimizer for linear losses over a simplex. We described linear averaging earlier in this paper. Finally, *alternation* is the idea that at iteration t, we provide Player 2 with the utility vector associated with the *current* iterate of Player 1, rather than that of the previous iteration, as is normally done in regret minimization. Figure 3 illustrates how this works, in contrast with Figure 2 which shows the usual flow. Tammelin et al. (2015) state that they prove convergence of CFR$^+$; however their proof relies on the folk theorem linking Nash equilibrium and regret. That folk theorem is only proven for the case where no alternation is applied. We show below that the theorem does not hold with alternation!

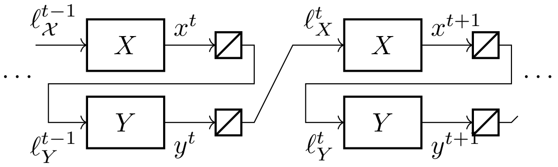

Figure 3: The alternation method for CFR in games. The loss at iteration t for y is computed with x^t. The symbol ⧄ denotes computation/construction of the loss function.

Observation 1. *Let the action spaces for the players be* $X = Y = [0, 1]$*, and let* $\ell_X^t : x \mapsto x \cdot y^t$, $\ell_Y^t : y \mapsto -y \cdot x^{t+1}$ *be bilinear loss functions (the superscript* $t + 1$ *comes from the use of alternation—see Figure 3). Consider the sequence of strategies* $x^t = t \mod 2, y^t = (t + 1) \mod 2$*. A simple check reveals that after* $2T$ *iterations, the average regrets of the two players are both 0. Yet, the average strategies* $\bar{x}^{2T} = \bar{y}^{2T} = 0.5$ *do not converge to a saddle point of* xy*.*

This observation should be seen as more of a theoretical issue than practical; alternation has been used extensively in practice, and the problem that we show does not seem to come up for nondegenerate iterates (at least for CFR$^+$; it may explain some erratic behavior that we have anecdotally observed with other regret minimization algorithms when using alternation).

Experiments

We conducted multiple kinds of experiments on two EFG settings. The first game is Leduc 5 poker (Southey et al., 2005), a standard benchmark in imperfect-information game solving. There is a deck consisting of 5 unique cards with 2 copies of each. There are two rounds. In the first round, each player places an ante of 1 in the pot and receives a single private card. A round of betting then takes place with a two-bet maximum, with Player 1 going first. A public shared card is then dealt face up and another round of betting takes place. Again, Player 1 goes first, and there is a two-bet maximum. If one of the players has a pair with the public card, that player wins. Otherwise, the player with the higher card wins. All bets in the first round are 1, while all bets in the second round are 2. The second game is a variant of Goofspiel (Ross, 1971), a bidding game where each player has a hand of cards numbered 1 to N. A third stack of N cards is shuffled and used as prizes: each turn a prize card is revealed, and the players each choose a private card to bid on the prize, with the high card winning, the value of the prize card is split evenly on a tie. After N turns all prizes have been dealt out and the payoff to each player is the sum of prize cards that they win. We use $N = 4$ in our experiments.

First we investigate a setting where no previous regret-minimization algorithms based on minimizing regret locally existed: the computation of QREs via LRM and our more general convex losses. Ling, Fang, and Kolter (2018) use Newton's method for this setting, but, as with standard Nash equilibrium, second-order algorithms do not scale to large games (this is why CFR$^+$ has been so successful for creating human-level poker AIs). We compare how quickly we can compute QREs compared to how quickly Nash equilibria can be computed, in order to understand how large games we can expect to find QREs for with our approach. To do this we run LRM with online gradient descent (OGD) at each decision point. Because OGD is not guaranteed to stay within the simplex at each iteration we need to project; this can be implemented via binary search for decision points with large dimension (Duchi et al., 2008), and via a constant-size decision tree for low-dimension decision points. The results are shown in Figure 4. We see that LRM performs extremely well; in Goofspiel it converges vastly faster than CFR$^+$, and in Leduc 5 it converges at a rate comparable to CFR$^+$ and eventually becomes faster. This shows that QRE computation via LRM likely scales to extremely large EFGs, such as real-world-sized poker games (since CFR$^+$ is known to scale to such games).

In the second set of experiments we investigate the speed of convergence for solving ℓ_2-regularized EFGs. Again we include the convergence rate of standard CFR and CFR$^+$ for Nash equilibrium computation as a benchmark. The results for Leduc 5 are in Figure 5. Solving the regularized game is much faster than computing an Nash equilibrium via CFR$^+$ except for extremely small amounts of regularization. Results for Goofspiel are in the online appendix; the results are very similar to the ones for Leduc 5.

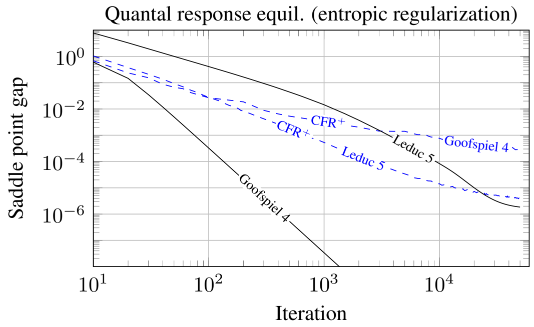

Figure 4: The QRE saddle-point gap as a function of the number of iterations for each game. The convergence rates of CFR$^+$ for Nash equilibrium is shown for reference.

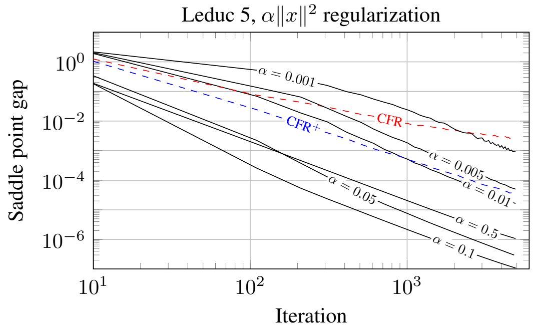

Figure 5: The saddle-point gap as a function of the number of iterations for ℓ_2-regularized Leduc 5 for varying regularization amounts. The convergence rates of CFR$^+$ for Nash equilibrium is shown for reference.

In the third set of experiments we investigate the performance of LRM in a single-agent-learning setting: learning how to exploit a static opponent where we observe repeated samples from their strategy. We consider a setting where the exploiter wishes to maximally exploit subject to staying near a pre-computed Nash equilibrium in order to avoid opening herself up to exploitability (Ganzfried and Sandholm, 2011). We model this in a new way: as a regularized online SDM problem, where the loss for the exploiter is

$$\ell^t(x) := \langle -A\mu(y^t), \mu(x)\rangle + \alpha D(x\|x^{NE}) \qquad (7)$$

where y^t is the t'th observation from the opponent's strategy and $D(x\|x^{NE})$ is the dilated ℓ_2-based Bregman divergence between the NE strategy x^{NE} and x. The opponent's suboptimal strategy was computed by running CFR$^+$ until a gap of 0.1 was reached. We stop the training of the exploiter after 5000 iterations or when an average regret of 0.0005 was reached, whichever happens first. Figure 6 shows the results. The "utility increase" line shows how much the agent gains by moving away from the Nash equilibrium and towards an exploitative strategy, while the "exploitability" shows to what extent the agent thereby opens herself up to being exploited by an optimal adversary. We see that this model can indeed be used as a scalable proxy for trading off exploitation and exploitability by varying α.

Our experiments are preliminary: we use simple OGD with very little tuning, and focus on iteration complexity rather than runtime. There are several reasons to expect that LRM for regularized games could be made much faster. For one, the laminar losses are strongly convex, so accelerated methods could be employed. This leads to significantly better theoretical convergence rate than that of CFR$^+$ for Nash equilibrium, and could likely be exploited in practice also. Furthermore, we used OGD for convenience, but one could most likely employ a projection-free algorithm and thus have the computational cost at each decision point be the same as for CFR$^+$ while getting a faster convergence rate.

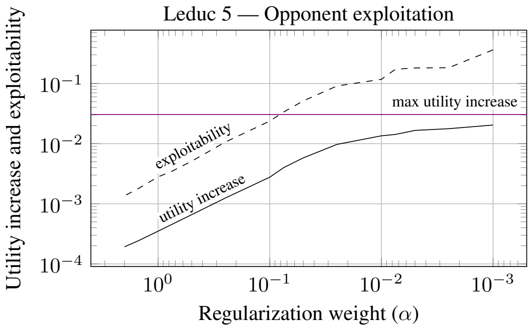

Figure 6: The utility increase from exploitation, and the resulting increase in the agent's own exploitability, as a function of decreasing penalization on distance from Nash equilibrium in (7). The straight line shows the value of best responding (i.e. maximally exploiting) to the opponent strategy.

Conclusions and Future Research

We presented LRD, a new decomposition of the regret associated with a sequential action space into regrets associated with individual decision points. We developed our technique for general compact convex sets and convex losses at each decision point, thus providing a generalization of CFR beyond simplex decision points and linear loss. We then showed that our results lead to a new class of regret-minimization algorithms that solve SDM problems by minimizing regret locally at each decision point. Although more general, our proof also provides a new perspective on the CFR algorithm in terms of our regret decomposition, and we explained the need for weighting by reach as a consequence of averaging in sequence form. We then showed that our approach can be used to compute regularized equilibria as well as Nash equilibrium refinements, and gave the first regret-minimization algorithm for computing (reduced-normal-form) quantal response equilibrium (QRE) (based on local regrets). We showed experimentally that even a preliminary variant of LRM can be used to compute QREs in EFGs at a rate that is comparable to Nash equilibrium finding using CFR$^+$, thus yielding the first algorithm for computing QREs in extremely large EFGs. We similarly showed that ℓ_2-regularized equilibrium can be computed very quickly with out method. Finally, we showed how our approach can be used as a new approach to opponent exploitation, and to control the tradeoff between exploitation and exploitability.

It would be interesting to investigate tweaks to the algorithms that use LRM in order to understand what variants yield best practical performance. It would also be interesting to find further applications where our new types of decision spaces and loss functions can be used.

References

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

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

Brown, N., and Sandholm, T. 2017. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science* eaao1733.

Chen, K., and Bowling, M. 2012. Tractable objectives for robust policy optimization. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*.

Duchi, J.; Shalev-Shwartz, S.; Singer, Y.; and Chandra, T. 2008. Efficient projections onto the l 1-ball for learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, 272–279. ACM.

Farina, G., and Gatti, N. 2017. Extensive-form perfect equilibrium computation in two-player games. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Farina, G.; Kroer, C.; and Sandholm, T. 2017. Regret minimization in behaviorally-constrained zero-sum games. In *International Conference on Machine Learning (ICML)*.

Ganzfried, S., and Sandholm, T. 2011. Game theory-based opponent modeling in large imperfect-information games. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.

Hazan, E., and Kale, S. 2010. Extracting certainty from uncertainty: Regret bounded by variation in costs. *Machine learning* 80(2-3):165–188.

Hazan, E. 2016. Introduction to online convex optimization. *Foundations and Trends in Optimization* 2(3-4):157–325.

Hoda, S.; Gilpin, A.; Peña, J.; and Sandholm, T. 2010. Smoothing techniques for computing Nash equilibria of sequential games. *Mathematics of Operations Research* 35(2).

Jin, P. H.; Levine, S.; and Keutzer, K. 2018. Regret minimization for partially observable deep reinforcement learning. In *International Conference on Machine Learning (ICML)*.

Kroer, C., and Sandholm, T. 2016a. Imperfect-recall abstractions with bounds in games. In *Proceedings of the ACM Conference on Economics and Computation (EC)*.

Kroer, C., and Sandholm, T. 2016b. Sequential planning for steering immune system adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Kroer, C.; Waugh, K.; Kılınç-Karzan, F.; and Sandholm, T. 2015. Faster first-order methods for extensive-form game solving. In *Proceedings of the ACM Conference on Economics and Computation (EC)*.

Kroer, C.; Waugh, K.; Kılınç-Karzan, F.; and Sandholm, T. 2017. Theoretical and practical advances on smoothing for extensive-form games. In *Proceedings of the ACM Conference on Economics and Computation (EC)*.

Kroer, C.; Farina, G.; and Sandholm, T. 2017. Smoothing method for approximate extensive-form perfect equilibrium. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Kuhn, H. W. 1950. A simplified two-person poker. In Kuhn, H. W., and Tucker, A. W., eds., *Contributions to the Theory of Games*, volume 1 of *Annals of Mathematics Studies, 24*. Princeton, New Jersey: Princeton University Press. 97–103.

Lanctot, M.; Gibson, R.; Burch, N.; Zinkevich, M.; and Bowling, M. 2012. No-regret learning in extensive-form games with imperfect recall. In *International Conference on Machine Learning (ICML)*.

Ling, C. K.; Fang, F.; and Kolter, J. Z. 2018. What game are we playing? End-to-end learning in normal and extensive form games. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Moravčík, M.; Schmid, M.; Burch, N.; Lisý, 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).

Nemirovski, A. 2004. Prox-method with rate of convergence O(1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization* 15(1).

Ross, S. M. 1971. Goofspiel–the game of pure strategy. *Journal of Applied Probability* 8(3):621–625.

Sandholm, T. 2015. Steering evolution strategically: Computational game theory and opponent exploitation for treatment planning, drug design, and synthetic biology. In *AAAI Conference on Artificial Intelligence (AAAI)*. Senior Member Track.

Southey, F.; Bowling, M.; Larson, B.; Piccione, C.; Burch, N.; Billings, D.; and Rayner, C. 2005. Bayes' bluff: Opponent modelling in poker. In *Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence (UAI)*.

Tammelin, O.; Burch, N.; Johanson, M.; and Bowling, M. 2015. Solving heads-up limit Texas hold'em. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*.

von Stengel, B. 1996. Efficient computation of behavior strategies. *Games and Economic Behavior* 14(2):220–246.

Zinkevich, M.; Bowling, M.; Johanson, M.; and Piccione, C. 2007. Regret minimization in games with incomplete information. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*.

Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning (ICML)*, 928–936.