Faster First-Order Methods for Extensive-Form Game Solving

Christian Kroer, Computer Science Department, Carnegie Mellon University
Kevin Waugh, Computer Science Department, Carnegie Mellon University
Fatma Kılınç-Karzan, Tepper School of Business, Carnegie Mellon University
Tuomas Sandholm, Computer Science Department, Carnegie Mellon University

We study the problem of computing a Nash equilibrium in large-scale two-player zero-sum extensive-form games. While this problem can be solved in polynomial time, first-order or regret-based methods are usually preferred for large games. Regret-based methods have largely been favored in practice, in spite of their theoretically inferior convergence rates. In this paper we investigate the acceleration of first-order methods both theoretically and experimentally. An important component of many first-order methods is a distance-generating function. Motivated by this, we investigate a specific distance-generating function, namely the dilated entropy function, over treeplexes, which are convex polytopes that encompass the strategy spaces of perfect-recall extensive-form games. We develop significantly stronger bounds on the associated strong convexity parameter. In terms of extensive-form game solving, this improves the convergence rate of several first-order methods by a factor of $O(\frac{\#\text{information sets}\cdot\text{depth}\cdot M}{2^{\text{depth}}})$ where M is the maximum value of the ℓ_1 norm over the treeplex encoding the strategy spaces.

Experimentally, we investigate the performance of three first-order methods (the excessive gap technique, mirror prox, and stochastic mirror prox) and compare their performance to the regret-based algorithms. In order to instantiate stochastic mirror prox, we develop a class of gradient sampling schemes for game trees. Equipped with our distance-generating function and sampling scheme, we find that mirror prox and the excessive gap technique outperform the prior regret-based methods for finding medium accuracy solutions.

Categories and Subject Descriptors: I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems; J.4.a [**Social and Behavioral Sciences**]: Economics

General Terms: Algorithms, Economics, Theory

Additional Key Words and Phrases: Extensive-form game; equilibrium computation; convex optimization; convex analysis

1. INTRODUCTION

Extensive-form games (EFGs) model sequential interaction, imperfect information, and outcome uncertainty; and are thus one of the broadest classes of game models. Nash equilibria prescribe a particular notion of rational behavior in such games. For the specific case of two-player zero-sum EFGs with perfect recall, an exact Nash equilibrium can be computed in polynomial time using a Linear Program (LP). But the size of the resulting LP is linear in the size of the game tree [von Stengel 1996]; and for very-large scale games, it is prohibitively expensive [Sandholm 2010] to solve it exactly. This has motivated the design of iterative algorithms that converge to a Nash equilibrium in the limit. Such algorithms are mainly categorized as first-order methods (FOMs) [Hoda et al. 2010] and regret-based [Zinkevich et al. 2007] approaches.

This work is supported by the National Science Foundation under grant IIS-1320620. Authors addresses: C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA; email: {ckroer,waugh,sandholm}@cs.cmu.edu, fkilinc@andrew.cmu.edu

EC'15, June 15–19, 2015, Portland, OR, USA. ACM 978-1-4503-3410-5/15/06 ...$15.00.

The current state-of-the-art for practical game solving is a regret-based stochastic algorithm [Lanctot et al. 2009], with an $O(\frac{1}{\epsilon^2})$ convergence rate. Hoda et al. [2010] have studied first-order methods (FOMs) with an $O(\frac{1}{\epsilon})$ rate of convergence. While such approaches have more desireable theoretical guarantees, they have yet to become the norm in practice. This paper investigates the acceleration of such methods, both from a theoretical and an experimental perspective.

Our approach is motivated by the fact that many FOMs work with a *distance-generating function* (d.g.f.), a measure of distance over the search space, which is required to be strongly convex. Moreover, the convergence rate of such FOMs usually depends on the ratio of the diameter of the search space measured by the d.g.f., scaled by the strong convexity parameter of the same function. Thus, good lower bounds on this strong convexity parameter are desirable. On the theoretical side, we investigate a class of distance-generating functions, namely the *dilated entropy function* over the class of *treeplexes*, a convex polytope that generalizes the strategy spaces of the players in perfect-recall EFGs. Hoda et al. [2010] developed a generic scheme for constructing such functions for EFGs based on standard d.g.f.s used for the simplex domain. However, the generic scheme from Hoda et al. [2010] leads to very weak strong convexity parameters, resulting in slow convergence rates. In this paper, we specifically analyze the dilated entropy function as a d.g.f. and establish its best known strong convexity lower bound over treeplexes. Our result, in particular, improves this bound on the strong convexity parameter to the square root of the prior bound, thereby resulting in an improvement in the convergence rate of the same order for many FOMs such as Nesterov's excessive gap technique (EGT) [Nesterov 2005] which was applied to EFG solving by Hoda et al. [2010], as well as for any other algorithm based on d.g.f.s over treeplexes. Moreover, such an improvement of the strong convexity parameter is especially critical for stochastic FOMs, where it is not possible to speed up the search through line search techniques.

The top poker bots at the Annual Computer Poker Competition are created with the monte-carlo counterfactual regret minimization (MCCFR) algorithm [Lanctot et al. 2009], which is a sampling variant of the counterfactual regret minimization algorithm (CFR) [Zinkevich et al. 2007; Brown et al. 2015]. Inspired by this success, we also describe a family of sampling schemes that lead to unbiased gradient estimators for EFGs, and suggest their use in a stochastic FOM.

Finally, we perform a preliminary experimental investigation of the speed up of FOMs with convergence rate $O(\frac{1}{\epsilon})$. Specifically, using our theoretical results, we instantiate the mirror prox (MP) [Nemirovski 2004], stochastic mirror prox (SMP) [Juditsky et al. 2011], and EGT [Nesterov 2005] algorithms to EFG solving. We also compare the performance of these algorithms with the premier regret-based methods CFR and MCCFR.

2. RELATED WORK

Nash equilibrium computation is a topic that has received much attention in the literature [Littman and Stone 2003; Lipton et al. 2003; Gilpin and Sandholm 2007; Zinkevich et al. 2007; Jiang and Leyton-Brown 2011; Kroer and Sandholm 2014a; Daskalakis et al. 2014]. The specific equilibrium-finding problems further vary quite a bit. Here we restrict our attention to two-player zero-sum sequential games.

Koller et al. [1996] present an LP whose size is linear in the size of the game tree. This approach, coupled with lossless abstraction techniques, was used to solve Rhode-Island hold'em [Shi and Littman 2002; Gilpin and Sandholm 2007], a game with 3.1 billion nodes (roughly size $5 \cdot 10^7$ after lossless abstraction). However, for games larger than this, the resuting LPs tend not to fit in the computer memory, thus requiring approximate techniques. These techniques fall into two categories: iterative ϵ-Nash equilibrium-finding algorithms, and abstraction techniques [Sandholm 2010].

The most popular iterative Nash equilibrium algorithm is undoubtedly the counterfactual regret minimization (CFR) algorithm [Zinkevich et al. 2007] and its sampling-based variant monte-carlo CFR (MCCFR) [Lanctot et al. 2009]. Both of these regret-minimization algorithms perform local regret-based updates at each information set. Despite their slow convergence rate of $O(\frac{1}{\epsilon^2})$, they perform very well in pratice, likely owing to their very cheap iterations. As a stochastic algorithm MCCFR touches only the sampled part of the game tree. Also, as experimentally shown by Lanctot et al. [2009], even CFR can prune large parts of the game tree due to actions with probability zero. Recently, Waugh and Bagnell [2015] showed that CFR can be interpreted as a FOM with $O(\frac{1}{\epsilon^2})$ rate. Nonetheless, in this paper we make a distinction between regret-based methods and $O(\frac{1}{e})$ FOMs for ease of exposition. Hoda et al. [2010] has suggested the only other FOM, a customization of EGT with a convergence rate of $O(\frac{1}{\epsilon})$, that scales to solve large games. Gilpin et al. [2012] give an algorithm with convergence rate $O(\ln(\frac{1}{\epsilon}))$. But their bound has a dependence on a certain condition number of the sequence-form payoff matrix, which is difficult to estimate, and as a result they show a $O(\frac{1}{\epsilon})$ bound in the worst case. Finally, Bosansky et al. [2014] develop an iterative double-oracle algorithm for exact equilibrium computation. However, this algorithm only scales for games where it can identify an equilibrium of small support, and thus suffers from the same performance issues as the general LP approach for large EFGs.

In addition to equilibrium-finding algorithms, another central topic in large-scale game solving has been automated abstraction [Sandholm 2010]. Initially, this was used mostly for information abstraction [Gilpin and Sandholm 2007; Shi and Littman 2002; Zinkevich et al. 2007]. Lately, action abstraction approaches have gained considerable interest [Hawkin et al. 2011, 2012; Brown and Sandholm 2014; Kroer and Sandholm 2014a,b]. Sequential game abstraction approaches with solution quality bounds have also emerged for stochastic [Sandholm and Singh 2012] and extensive-form [Lanctot et al. 2012; Kroer and Sandholm 2014a,b] games more recently.

2.1. Sequence form

It is well-known [Romanovskii 1962; Koller et al. 1996; von Stengel 1996] that the strategy space of each player can be formulated in a way that leads to the following bilinear saddle-point formulation of the Nash equilibrium problem:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\langle x, Ay\rangle = \max_{y\in\mathcal{Y}}\min_{x\in\mathcal{X}}\langle x, Ay\rangle. \tag{1}$$

In this formulation, $\mathcal{X},\mathcal{Y}$ are convex polyhedral reformulations of the sequential strategy space of players 1 and 2 respectively. Let x, y be non-negative strategy vectors for players 1 and 2. Then $\mathcal{X},\mathcal{Y}$ are defined by the constraints $Ex = e, Fy = f$, where e, f are $|\mathcal{I}_1|, |\mathcal{I}_2|$-dimensional vectors of the form $[1, 0 \ldots,]$, and $\mathcal{I}_i$ is the number of information sets for player i. Likewise, E, F have as many rows, with each row encoding part of the sequential nature of the strategy vectors. For a complete treatment of this formulation, see von Stengel [1996]. Next, we develop our theoretical results for a more general class of convex polytopes that contain the sequence form strategy space as a special case.

3. OPTIMIZATION SETUP

Given a Bilinear Saddle Point Problem (BSPP), we first describe our general setup, independent of EFGs because our theoretical results apply to any optimization problem over treeplexes. We follow the presentation and notation of Juditsky and Nemirovski [2011a,b].

Throughout this paper, we use Matlab notation to denote vector and matrices, i.e., $[x; y]$ denotes the concatenation of two column vectors x, y.

In its most general form a BSPP is defined as

$$\max_{y\in\mathcal{Y}}\min_{x\in\mathcal{X}}\phi(x,y), \qquad (\mathcal{S})$$

where $\phi(x,y) = \upsilon + \langle a_1, x\rangle + \langle a_2, y\rangle + \langle y, Ax\rangle$, where $\mathcal{X}, \mathcal{Y}$ are nonempty convex compact sets in Euclidean spaces E_x, E_y and $\mathcal{Z} := \mathcal{X}\times\mathcal{Y}$, so $\phi(x,y) : Z \to \mathbf{R}$. For EFG solving, $\phi(x,y)$ is simply the inner product given in (1).

Note that $(\mathcal{S})$ gives rise to two convex optimization problems that are dual to each other:

$$\begin{array}{rcll}\mathrm{Opt}(P) & = & \min_{x\in\mathcal{X}}[\overline{\phi}(x) := \max_{y\in\mathcal{Y}}\phi(x,y)] & (P)\\ \mathrm{Opt}(D) & = & \max_{y\in\mathcal{Y}}[\underline{\phi}(y) := \min_{x\in\mathcal{X}}\phi(x,y)] & (D)\end{array}$$

with $\mathrm{Opt}(P) = \mathrm{Opt}(D) = \mathrm{Opt}$. It is well known that the solutions to $(\mathcal{S})$ — the saddle points of ϕ on $\mathcal{X}\times\mathcal{Y}$ — are exactly the pairs $z = [x; y]$ comprised of optimal solutions to problems (P) and (D). The accuracy of a candidate solution $z = [x; y]$ is quantified by the *saddle point residual*

$$\epsilon_{\mathrm{sad}}(z) \;=\; \overline{\phi}(x) - \underline{\phi}(y) = \underbrace{\left[\overline{\phi}(x) - \mathrm{Opt}(P)\right]}_{\geq 0} + \underbrace{\left[\mathrm{Opt}(D) - \underline{\phi}(y)\right]}_{\geq 0}.$$

In an EFG, $\epsilon_{\mathrm{sad}}(z)$ is proximity to being an ϵ-Nash equilibrium.

The problems (P) and (D) also give rise to the *variational inequality*: find $z_* \in Z$ s.t.

$$\langle F(z), z - z_*\rangle \geq 0 \;\text{ for all } z \in Z, \qquad (2)$$

where $F : \mathcal{Z} \mapsto E_x \times E_y$ is the affine monotone operator defined by

$$F(x,y) = \left[F_x(y) = \frac{\partial\phi(x,y)}{\partial x}; F_y(x) = -\frac{\partial\phi(x,y)}{\partial y}\right].$$

For EFG-solving purposes, $F(x,y)$ corresponds to the map returning the gradients $[Ay, -A^T x]$ from (1). Then (2) states that for any other strategy pair $z = [x,y]$, $z_* = [x_*, y_*]$ are best responses.

3.1. General framework for FOMs

Most FOMs capable of solving BSPP $(\mathcal{S})$ are quite flexible in terms of adjusting to the geometry of the problem characterized by the domain $\mathcal{X}, \mathcal{Y}$ of the BSPP $(\mathcal{S})$. The following components are standard in forming the setup for such FOMs:

- *Norm*: $\|\cdot\|$ on the Euclidean space E where the domain $\mathcal{Z} = \mathcal{X}\times\mathcal{Y}$ of $(\mathcal{S})$ lives, along with the dual norm $\|\zeta\|_* = \max\limits_{\|z\|\leq 1}\langle\zeta, z\rangle$.
- *Distance-Generating Function* (d.g.f.): A function $\omega(z) : \mathcal{Z} \to \mathbb{R}$, which is convex and continuous on $\mathcal{Z}$, and admits a continuous selection of subgradients $\omega'(z)$ on the set $\mathcal{Z}^o = \{z \in \mathcal{Z} : \partial\omega(z) \neq \emptyset\}$ (here $\partial\omega(z)$ is a subdifferential of ω taken at z), and is strongly convex with modulus 1 w.r.t. $\|\cdot\|$:

$$\forall z', z'' \in \mathcal{Z}^o : \; \langle\omega'(z') - \omega'(z''), z' - z''\rangle \geq \|z' - z''\|^2. \qquad (3)$$

- *Bregman distance*: $V_z(u) = \omega(u) - \omega(z) - \langle\omega'(z), u - z\rangle$, where $z \in \mathcal{Z}^o$ and $u \in \mathcal{Z}$.
- *Prox-mapping*: Given a *prox center* $z \in \mathcal{Z}^o$,

$$\mathrm{Prox}_z(\xi) = \operatorname*{argmin}_{w\in\mathcal{Z}}\{\langle\xi, w\rangle + V_z(w)\} : E \to \mathcal{Z}^o.$$

Given a properly chosen stepsize, the prox-mapping becomes a contraction, which is critical in the convergence analysis of FOMs. Furthermore, when the squared ℓ_2 norm is used as a d.g.f., the prox mapping becomes the usual projection operation of the vector $z - \xi$ onto $\mathcal{Z}$.

- *ω-center*: $z_\omega = \operatorname{argmin}_{z \in \mathcal{Z}} \omega(z) \in \mathcal{Z}^o$ of $\mathcal{Z}$.
- *Set width*: $\Omega = \Omega_z := \max\limits_{z \in \mathcal{Z}} V_{z_\omega}(z) \le \max\limits_{z \in \mathcal{Z}} \omega(z) - \min\limits_{z \in \mathcal{Z}} \omega(z)$.
- *Lipschitz constant*: $\mathcal{L}$ of F from $\|\cdot\|$ to $\|\cdot\|_*$, satisfying $\|F(z) - F(z')\|_* \le \mathcal{L}\|z - z'\|, \forall z, z'$.

The standard customization of a FOM for BSPP ($\mathcal{S}$) is based on associating a norm $\|\cdot\|_x$ and a d.g.f. $\omega_x(\cdot)$ with domain $\mathcal{X}$, and similarly $\|\cdot\|_y$, $\omega_y(\cdot)$ with domain $\mathcal{Y}$. Then, given two scalars $\alpha_x, \alpha_y > 0$, we build the d.g.f. and ω-center, z_ω, for $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ as:

$$\omega(z) = \alpha_x \omega_x(x) + \alpha_y \omega_y(y) \quad \text{and} \quad z_\omega = [x_{\omega_x}; y_{\omega_y}],$$

where $\omega_x(\cdot)$ and $\omega_y(\cdot)$ as well as x_{ω_x} and y_{ω_y} are customized based on the geometry of the domains $\mathcal{X}, \mathcal{Y}$. Then by letting $\xi = [\xi_x; \xi_y]$, $z = [x; y]$, the prox mapping becomes decomposable as

$$\text{Prox}_z(\xi) = \left[\text{Prox}_x^{\omega_x}\left(\frac{\xi_x}{\alpha_x}\right); \text{Prox}_y^{\omega_y}\left(\frac{\xi_y}{\alpha_y}\right) \right],$$

where $\text{Prox}_x^{\omega_x}(\cdot)$ and $\text{Prox}_y^{\omega_y}(\cdot)$ are respectively prox mappings w.r.t. $\omega_x(x)$ in domain $\mathcal{X}$ and $\omega_y(y)$ in domain $\mathcal{Y}$.

Based on this setup, the *Mirror Prox* (MP) algorithm is given in Algorithm 1.

ALGORITHM 1: Mirror Prox

```
   input  : $\omega$-center $z_\omega$, step size $\{\gamma_t\}$ and $\epsilon$
   output: $z^t (= [x^t; y^t])$
1  $t = 1; z_1 := z_\omega$;
2  while $\epsilon_{\text{sad}}(z^t) > \epsilon$ do
3      $v_t = \text{Prox}_{z_t}(\gamma_t F(z_t))$;
4      $z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(v_t))$;
5      $z^t = \left[\sum_{s=1}^t \gamma_s\right]^{-1} \sum_{s=1}^t \gamma_s v_s$;
6      $t = t + 1$;
```

Suppose the step sizes in the MP algorithm satisfy $\gamma_t = \mathcal{L}^{-1}$. Then, at every iteration $t \ge 1$, the corresponding solution $z^t = [x^t; y^t]$ satisfies $x^t \in \mathcal{X}$, $y^t \in \mathcal{Y}$, and we have $O(1)\frac{\Omega\mathcal{L}}{\epsilon}$ convergence rate [Nemirovski 2004]. Specifically:

$$\overline{\phi}(x^t) - \underline{\phi}(y^t) = \epsilon_{\text{sad}}(z^t) \le \frac{\Omega\mathcal{L}}{t}.$$

The EGT algorithm [Nesterov 2005] can be set up using the same framework, to achieve a convergence rate of $O(1)\frac{\Omega\mathcal{L}}{\epsilon}$.

3.2. Stochastic FOMs

In practice, each iteration of an FOM might be prohibitively expensive, usually because the prox function, gradient, or both, are expensive to compute. In such a case, a stochastic FOM based on sampling-based is often advantageous. Juditsky et al. [2011] discuss a stochastic variant of MP for BSSP, namely *Stochastic MP* (SMP). Algorithm 1 is modified as follows. Steps 3 and 4 do not use the exact gradients, but rather unbiased estimators thereof. Specifically, for every $u \in \mathcal{X}$, we assume access to a probability distribution Π_u such that $E_{\xi \sim \Pi_u}[\xi] = u$. Similarly, we assume access to P_v for $v \in \mathcal{Y}$ such that $E_{\eta \sim P_v}[\eta] = v$. We define variance as follows:

$$\sigma_x^2 = \sup_{u \in \mathcal{X}} E\left\{ \| A[\xi_u - u] \|_{y,*}^2 \right\}, \quad \sigma_y^2 = \sup_{v \in \mathcal{Y}} E\left\{ \| A^T[\eta_v - v] \|_{x,*}^2 \right\}$$

Juditsky et al. [2011] prove that after T iterations of SMP with stepsizes $\gamma_t = \min\left[\frac{1}{\sqrt{3}\mathcal{L}}, \sqrt{\frac{4\Omega}{7T(\sigma_x^2+\sigma_y^2)}}\right]$, the solution satisfies:

$$\mathbb{E}\left[\epsilon_{sad}(z^T)\right] \leq \max\left[\frac{7\Omega\mathcal{L}}{2T}, 7\sqrt{\frac{2\Omega(\sigma_x^2+\sigma_y^2)}{3T}}\right]. \tag{4}$$

4. TREEPLEXES

Hoda et al. [2010] introduce the *treeplex*, a class of convex polytopes that encompass the sequence-form description of strategy spaces in perfect-recall sequential games:

Definition 4.1. Treeplexes are described recursively:

(1) *Basic sets*: The standard simplex $\Delta_m = \{x \in [0,1]^m : \sum_{k=1}^m x_k = 1\}$ is a treeplex.
(2) *Cartesian product*: If $Q_1, \ldots, Q_k$ are treeplexes, then $Q_1 \times \cdots \times Q_k$ is a treeplex.
(3) *Branching*: If $P \subseteq [0,1]^p$ and $Q = \{Q_1, \ldots, Q_k\}$, where $Q_j \subseteq [0,1]^{q_j}$ are treeplexes and $l = \{l_1, \ldots, l_k\} \subseteq \{1, \ldots, p\}$, then

$$P \boxed{1} Q := \left\{ (x, y_1, \ldots, y_k) \in \mathbb{R}^{p + \sum_j q_j} : x \in P,\, y_1 \in x_{l_1} \cdot Q_1,\, \ldots, y_k \in x_{l_k} \cdot Q_k \right\}$$

is a treeplex and x_{l_j} are the branching variables for Q_j.

A treeplex is a tree of simplices, where children are connected to parents through the branching operation, which scales the child simplex by the value of the parent branching variable. This intuition is important because the proof of our main result relies on induction over these simplices. For EFGs, the simplices correspond to the information sets of a single player and the whole treeplex represents that player's strategy space. The branching operation has a sequential interpretation: The vector x represents the decision variables at some stage, while the vectors y_j represent the decision variables at the k potential following stages, depending on external outcomes. Here $k \leq p$, as some variables in x may not have subsequent decisions. As pointed out by von Stengel [1996], the treeplex can be represented by the linear equations $Ex = e$ for a matrix E with entries in $\{-1, 0, 1\}$ and a vector e with entries in $\{0, 1\}$.

For a treeplex Q, we denote by S_Q the index set of the set of simplices contained in Q (in an EFG S_Q is the set of information sets belonging to the player). For each vector $q \in Q$ and simplex Δ^j we let $\mathbb{I}_j$ denote the set of indices of q that index to variables from Δ^j, and q^j denote the subset of q referring to variables in $\mathbb{I}_j$. For each simplex Δ^j and branch $i \in \mathbb{I}_j$ we let $\mathcal{D}_j^i$ denote the set of indices of simplices reached immediately after Δ^j by taking branch i (in an EFG $\mathcal{D}_j^i$ is the set of potential next-step information sets for the player). For a vector $q \in Q$, simplex Δ^j, index $i \in \mathbb{I}_j$, and each child simplex Δ^k where $k \in \mathcal{D}_j^i$, Δ^k is scaled by q_i. Conversely, for a given simplex Δ^j we let p_j denote the index in q of the parent branching variable q_{p_j} that Δ^j is scaled by. We use the convention that $x_{p_j} = 1$ if Q is such that no branching operation precedes Δ^j. For each $j \in S_Q$, d_j is the maximum depth of the treeplex rooted at Δ^j, that is, the maximum number of simplices reachable through a series of branching operations at Δ^j.

An example treeplex is given in Figure 1. Let us denote this treeplex by Q. Q is constructed from eight two-to-three-dimensional simplices $\Delta^1, \ldots, \Delta^8$. The two simplices

Δ^1, Δ^2 are both at depth 1 through a Cartesian product operation, denoted by ×. We have maximum depths $d_1 = 2, = d_2 = 1$ beneath them. Since there are no preceding branching operations their parent variables are constants equal to 1, i.e., $q_{p_1} = q_{p_2} = 1$. For Δ^1, the corresponding set of indices in the q vector is $\mathbb{I}_1 = \{1, 2\}$, while for Δ^2 it is $\mathbb{I}_2 = \{3, 4\}$. At depth 2 we have the simplices $\Delta^3, \dots, \Delta^6$, each of which is scaled by the parent variable q_{p_j} that the branching operation was performed on. For example, Δ^3 is scaled by $q_{p_3} = q_1$.

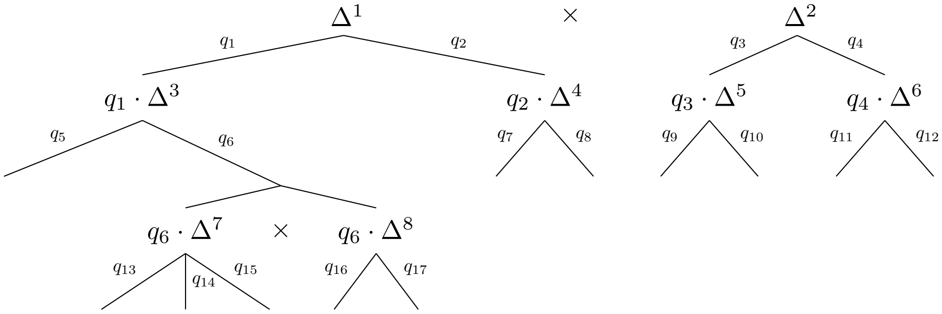

Fig. 1: An example treeplex constructed from 8 simplices. Cartesian product operation is denoted by ×.

Because we allow more than two-way branches, our formulation of a treeplex differs from that of Hoda et al. [2010]. It is possible to model the sequence-form using only two-way branches. Yet, this can in general cause a large increase in the depth of the treeplex, thus leading to significant degradation in the strong convexity parameter. Our generalization of multi-way branches allows us to better take into account the structure of the sequence-form game, thereby deriving better strong convexity results.

5. DILATED ENTROPY FUNCTION WITH BOUNDED STRONG CONVEXITY

Our goal is to construct d.g.f.s that are *strongly convex* with respect to the ℓ_1 norm on any treeplex Q, c.f. (3). Our analysis is based on the following alternative characterization of strongly convex twice differentiable functions:

FACT 5.1. *A twice-differentiable function f is strongly convex with modulus φ with respect to a norm $\|\cdot\|$ on nonempty convex set $C \subset \mathbb{R}^n$ if $h^T \nabla^2 f(z) h \geq \varphi \|h\|$, $\forall h \in \mathbb{R}^n, z \in C$ where $\nabla^2 f(z)$ is the Hessian of f at z.*

The basic building block in our construction is the well-known *entropy* d.g.f. for the simplex Δ_n, which is given by $\omega_E(z) = -\sum_{i=1}^n z_i \log(z_i)$. It is well-known that this function is strongly convex with modulus 1 with respect to the ℓ_1 norm on Δ_n. We will show that a suitable modification of this function achieves a desireable strong convexity parameter for treeplex domains.

Our analysis also requires a measure of the size of a treeplex Q. Thus, we define

$$M_Q := \max_{q \in Q} \|q\|_1. \tag{5}$$

For EFGs, M_Q is the maximum number of information sets that can have nonzero probability of being reached when the player whose strategy space is encoded by Q has to choose a pure strategy, while the other player can choose a mixed strategy.

The treeplex structure is naturally related to the *dilation operation* [Hiriart-Urruty and Lemaréchal 2001] defined as follows: First, given a compact set $K \subseteq \mathbb{R}^d$ and a function $f : K \to \mathbb{R}$, we define the set $\bar{K} \subseteq \mathbb{R}^{d+1}$ as

$$\bar{K} := \left\{(t, z) \in \mathbb{R}^{d+1} : t \in [0, 1], z \in t \cdot K\right\}.$$

Definition 5.2. Given a function $f(z)$, the *dilation operation* is the function $\bar{f} : \bar{K} \to \mathbb{R}$ given by

$$\bar{f}(z, t) = \begin{cases} t \cdot f(z/t) & \text{if } t > 0 \\ 0 & \text{if } t = 0 \end{cases}.$$

Based on this dilation operation, we define the following dilated entropy function over the simplices of a treeplex:

Definition 5.3 (*Dilated entropy function*). Given a treeplex Q and the weights β_j for each $j \in S_Q$, we let

$$\omega(q) = \sum_{j \in S_Q} \beta_j \sum_{i \in \mathbb{I}_j} q_i \log \frac{q_i}{q_{p_j}} \quad \text{for any } q \in Q,$$

where p_j is the index of the branching variable preceding Δ^j, with the convention that $q_{p_j} = 1$ if Δ^j has no branching operation preceding it.

This class of d.g.f.s is a small subset of the d.g.f.s considered by Hoda et al. [2010]. Nevertheless, by focusing on the dilated entropy function and the ℓ_1 norm, we establish much better strong convexity bounds. Our main theoretical result is:

THEOREM 5.4. *For any treeplex Q, the dilated entropy function with weights* $\beta_j = 2^{d_j} M_{Q_j}$ *is strongly convex modulus* $\frac{1}{|S_Q|}$ *with respect to the* ℓ_1 *norm.*

The proof of Theorem 5.4 is given in Section 5.2.

This is, to our knowledge, the first strong convexity bounds for general treeplexes. Hoda et al. [2010] have proven strong convexity of this scheme, under weaker assumptions on the norm. But their analysis is for only the degradation associated with a single two-way branching step and fails to provide explicit strong convexity parameters. For the special case of *uniform treeplexes* (a significant restriction on the treeplex construction), Hoda et al. [2010] do give explicit bounds. Nevertheless, in Section 5.3 we show that their bounds are significantly inferior to ours even for this special case.

Theorem 5.4 implies that for any treeplex Q, the dilated entropy function with weights $\beta_j = S_Q 2^{d_j} M_{Q_j}$ is strongly convex with modulus 1 with respect to the ℓ_1 norm. Note that $\omega(\cdot)$ defined in Theorem 5.4 is twice differentiable in the interior of treeplex Q and admits a continuous gradient selection. Thus, the dilated entropy function is compatible with the ℓ_1 norm, as required by the MP setup.

5.1. Preliminaries for the proof of Theorem 5.4

We start with some simple facts and a few technical lemmas that will streamline the proof of Theorem 5.4.

FACT 5.5. *For any* q *in the relative interior* Δ_n^0 *of the* n*-dimensional simplex* Δ_n *and any* $h \in \mathbb{R}^n$*, the Cauchy-Schwarz inequality implies*

$$\sum_{i=1}^n \frac{h_i^2}{q_i} = \underbrace{\left(\sum_{j=1}^n q_j\right)}_{=1} \left(\sum_{i=1}^n \frac{h_i^2}{q_i}\right) \geq \left(\sum_{i=1}^n \frac{|h_i|}{\sqrt{q_i}} \sqrt{q_i}\right)^2 = \|h\|_1^2.$$

FACT 5.6. *For any $a, b \in \mathbb{R}$ and index set J over variables a_j we have*

(i) $a^2 + b^2 \geq 2ab$, *and thus* $\sum_{j \in J} \sum_{i \neq j} a_i^2 \geq \sum_{j \in J} \sum_{i \neq j} a_i a_j$.

(ii) $2\left(a-b\right)^2 + 2b^2 \geq \left(a-b\right)^2 + b^2 + 2b\left(a-b\right) = \left(\left(a-b\right)+b\right)^2 = a^2$.

LEMMA 5.7. *Given a treeplex Q, for all $j \in S_Q$:*

$$M_{Q_j} = 1 + \max_{i \in \mathbb{I}_j} \sum_{l \in \mathcal{D}_j^i} M_{Q_l},$$

$$M_{Q_j} \geq 1 + \sum_{l \in \mathcal{D}_j^i} M_{Q_l} \quad \forall i \in \mathbb{I}_j.$$

PROOF. The equality follows from the definition of M_{Q_j} and the fact that Δ^j is a simplex. The inequality follows by every $i \in \mathbb{I}_j$ being lower bounded by the maximum. □

LEMMA 5.8. *For any q in the relative interior of a treeplex Q and for all $j \in S_Q$:*

$$\sum_{i \in \mathbb{I}_j} \left(\frac{h_i^2}{q_i} - \frac{2 h_i h_{p_j}}{q_{p_j}} \right) + \frac{2 h_{p_j}^2}{q_{p_j}} \geq \frac{1}{2} \sum_{i \in \mathbb{I}_j} \frac{h_i^2}{q_i}. \tag{6}$$

PROOF. Consider a treeplex Q and $j \in S_Q$. Then for Δ^j of dimension n_j, q^j is taken from the scaled simplex $\Delta_{n_j} \cdot q_{p_j}$. Hence, the vector $\bar{q}^j = \frac{1}{q_{p_j}} q^j$ is from Δ_{n_j} and

$$\sum_{i \in \mathbb{I}_j} \left(\frac{h_i^2}{q_i} - \frac{2 h_i h_{p_j}}{q_{p_j}} \right) + \frac{2 h_{p_j}^2}{q_{p_j}} = \sum_{i \in \mathbb{I}_j} \left[\frac{1}{q_{p_j}} \left(\frac{h_i^2}{\bar{q}_i} - 2 h_i h_{p_j} + \bar{q}_i h_{p_j}^2 \right) + \frac{\bar{q}_i h_{p_j}^2}{q_{p_j}} \right]$$

$$= \sum_{i \in \mathbb{I}_j} \left[\frac{1}{q_{p_j}} \left(\frac{h_i}{\sqrt{\bar{q}_i}} - \sqrt{\bar{q}_i} h_{p_j} \right)^2 + \frac{\bar{q}_i h_{p_j}^2}{q_{p_j}} \right] \geq \sum_{i \in \mathbb{I}_j} \left[\frac{1}{q_{p_j}} \frac{h_i^2}{2 \bar{q}_i} \right] = \frac{1}{2} \sum_{i \in \mathbb{I}_j} \frac{h_i^2}{q_i},$$

where the first equality follows from rearranging the terms in the summation by moving q_{p_j} outside, the second equality from completing the square, and the inequality from Fact 5.6 (ii) with $a = \frac{h_i}{\sqrt{\bar{q}_i}}, b = \sqrt{\bar{q}_i} h_{p_j}$. □

5.2. Proof of Theorem 5.4

We are now ready to prove Theorem 5.4.

PROOF. Given a treeplex Q, for any $q \in Q^0$ and for any h, we will first prove

$$h^T \nabla \omega^2(q) h \geq \sum_{j \in S_Q} \|h^j\|_1^2, \tag{7}$$

from which the strong convexity of $\omega(\cdot)$ w.r.t. ℓ_1 norm on Q will follow. For each $j \in S_Q, i \in \mathbb{I}_j$ the second-order partial derivates of $\omega(\cdot)$ w.r.t. q_i are:

$$\omega''_{q_i^2}(q) = \frac{\beta_j}{q_i} + \sum_{k \in \mathcal{D}_j^i} \sum_{l \in \mathbb{I}_k} \frac{\beta_k q_l}{q_i^2} = \frac{\beta_j}{q_i} + \sum_{k \in \mathcal{D}_j^i} \frac{\beta_k}{q_i}, \tag{8}$$

where the last equality holds because $q^k \in \Delta^k \cdot q_i$. For each $j \in S_Q, i \in \mathbb{I}_j, k \in \mathcal{D}_j^i$, and $l \in \mathbb{I}_k$ we also need the second-order partial derivates w.r.t q_i, q_l:

$$\omega''_{q_i,q_l}(q) = \omega''_{q_l,q_i}(q) = -\frac{\beta_k}{q_i}. \tag{9}$$

Equations (8) and (9) together imply

$$\begin{aligned} h^T \nabla^2 \omega(q) h &= \sum_{j \in S_Q} \sum_{i \in \mathbb{I}_j} \left[h_i^2 \left(\frac{\beta_j}{q_i} + \sum_{k \in \mathcal{D}_j^i} \frac{\beta_k}{q_i} \right) - \sum_{k \in \mathcal{D}_j^i} \sum_{l \in \mathbb{I}_k} h_i h_l \frac{2\beta_k}{q_i} \right] && (10) \\ &= \sum_{j \in Q} \sum_{i \in \mathbb{I}_j} \beta_j \left[\left(\frac{h_i^2}{q_i} - \frac{2 h_i h_{p_j}}{q_{p_j}} \right) + \frac{h_{p_j}^2}{q_{p_j}} \right], && (11) \end{aligned}$$

where the last equality is obtained by shifting the last two terms on the right-hand side of (10) to be counted at the child indices in the sum $\sum_{j \in S_Q}$. So $h_i^2 \frac{\beta_k}{q_i}$ becomes $\beta_k \frac{h_{p_{j'}}^2}{q_{p_{j'}}}$, and $h_i h_l \frac{2\beta_k}{q_i}$ becomes $\beta_k \frac{2 h_i h_{p_{j'}}}{q_{p_{j'}}}$, where j' is the index of the child simplex. For Δ^j at depth 1, there is no preceding branching operation, so technically the variables h_{p_j}, q_{p_j} do not exist. We circumvent this with the convention $h_{p_j} = 0, q_{p_j} = 1$ for such j.

We are now ready to prove equation (7) by induction. Starting from the value of the quadratic at the root, $2^d M_Q \sum_{i \in \mathbb{I}_Q} \frac{h_i^2}{q_i}$, we show that the multipliers $2^d M_Q$ are large enough to "push down" $2^{d-1} M_{Q_j} \frac{h_i^2}{q_i}$ to each descendant simplex of the branching operation at Δ^j. Our inductive assumption for depth k is:

$$\begin{aligned} &\sum_{j \in S_Q, d_j \le k} \beta_j \left[\sum_{i \in \mathbb{I}_j} \left(\frac{h_i^2}{q_i} - \frac{2 h_i h_{p_j}}{q_{p_j}} \right) + \frac{h_{p_j}^2}{q_{p_j}} \right] \\ \ge \quad &\sum_{j \in S_Q, d_j \le k} \|h^j\|_1^2 + \sum_{j \in S_Q, d_j = k} \left[2^{d-k-1} (M_{Q_j} - 1) \sum_{i \in \mathbb{I}_j} \frac{h_i^2}{q_i} \right]. \end{aligned}$$

Base case: let $k = 1$. Since this is the first set of branching operations, $h_{p_j} = 0$, so

$$\begin{aligned} \sum_{j \in S_Q, d_j = 1} \beta_j \left[\sum_{i \in \mathbb{I}_j} \left(\frac{h_i^2}{q_i} - \frac{2 h_i h_{p_j}}{q_{p_j}} \right) + \frac{h_{p_j}^2}{q_{p_j}} \right] &= \sum_{j \in S_Q, d_j = 1} \left[2^d M_{Q_j} \sum_{i \in \mathbb{I}_j} \frac{h_i^2}{q_i} \right] && (12) \\ \ge \sum_{j \in S_Q, d_j = 1} \left[2^d \left(M_{Q_j} - 1 \right) \sum_{i \in \mathbb{I}_j} \frac{h_i^2}{q_i} + 2^d \|h^j\|_1^2 \right], & && (13) \end{aligned}$$

where the last inequality follows from Fact 5.5.

Inductive step: Given our inductive assumption for depth k, we show that it then holds for $k+1$. From the inductive assumption, we get

$$\begin{aligned}
&\sum_{j\in S_Q, d_j\le k+1} \beta_j \left[\sum_{i\in\mathbb{I}_j}\left(\frac{h_i^2}{q_i} - \frac{2h_i h_{p_j}}{q_{p_j}}\right) + \frac{h_{p_j}^2}{q_{p_j}}\right] \\
&\ge \sum_{j\in S_Q, d_j\le k} \|h^j\|_1^2 + \sum_{j\in S_Q, d_j=k}\left[2^{d-k-1}(M_{Q_j}-1)\sum_{i\in\mathbb{I}_j}\frac{h_i^2}{q_i}\right] \\
&\quad + \sum_{j\in S_Q, d_j=k+1} 2^{d-k-1} M_{Q_j}\left[\sum_{i\in\mathbb{I}_j}\left(\frac{h_i^2}{q_i} - \frac{2h_i h_{p_j}}{q_{p_j}}\right) + \frac{h_{p_j}^2}{q_{p_j}}\right]. \qquad (14)
\end{aligned}$$

By Lemma 5.7, $M_{Q_j} \ge 1 + \sum_{l\in\mathcal{D}_j^i} M_{Q_l}$, so $\left(M_{Q_j}-1\right)\sum_{i\in\mathbb{I}_j}\frac{h_i^2}{q_i} \ge \sum_{i\in\mathbb{I}_j}\sum_{l\in\mathcal{D}_j^i} M_{Q_l}\frac{h_{p_j}^2}{q_{p_j}}$. By counting the sum over $j\in S_Q, d_j = k$ in the RHS of (14) at the children we get

$$\ge \sum_{j\in S_Q, d_j\le k} \|h^j\|_1^2 + \sum_{j\in S_Q, d_j=k+1} 2^{d-k-1} M_{Q_j}\left[\sum_{i\in\mathbb{I}_j}\left(\frac{h_i^2}{q_i} - \frac{2h_i h_{p_j}}{q_{p_j}}\right) + \frac{2h_{p_j}^2}{q_{p_j}}\right]. \qquad (15)$$

Then, using Lemma 5.8 in (15) gives

$$\begin{aligned}
&\ge \sum_{j\in S_Q, d_j\le k} \|h^j\|_1^2 + \sum_{j\in S_Q, d_j=k+1} 2^{d-k-2} M_{Q_j}\sum_{i\in\mathbb{I}_j}\frac{h_i^2}{q_i} \\
&\ge \sum_{j\in S_Q, d_j\le k+1} \|h^j\|_1^2 + \sum_{j\in S_Q, d_j=k+1} 2^{d-k-2}(M_{Q_j}-1)\sum_{i\in\mathbb{I}_j}\frac{h_i^2}{q_i},
\end{aligned}$$

where the last inequality is obtained by applying the same trick that we used to go from (12) to (13). This concludes the inductive step, and establishes (7) because $M_{Q_j}\ge 1$ for all $j\in S_Q$ and all other terms are nonnegative.

Then, by bounding the right-hand side of (7), we get

$$\begin{aligned}
h^T\nabla^2\omega(q)h &\ge \sum_{j\in S_Q}\|h^j\|_1^2 = \sum_{j\in S_Q}\left[\frac{1}{|S_Q|}\|h^j\|_1^2 + \sum_{i\ne j}\frac{1}{|S_Q|}\|h^j\|_1^2\right] \\
&\ge \sum_{j\in S_Q}\left[\frac{1}{|S_Q|}\|h^j\|_1^2 + \frac{1}{|S_Q|}\sum_{i\ne j}\|h^j\|_1\|h^i\|_1\right] = \frac{1}{|S_Q|}\|h\|_1^2, \qquad (16)
\end{aligned}$$

where the last inequality follows by applying Fact 5.6 (i) to each pair $\|h^i\|_1^2, \|h^j\|_1^2$.

Since the dilated entropy function is twice differentiable in the relative interior of Q, using Fact 5.1, strong convexity with modulus φ is equivalent to

$$h^T\nabla^2\omega(q)h \ge \varphi\|h\|_1^2 \qquad (17)$$

for all $q\in Q^0, h\in\mathbb{R}^n$. Applying this to (16) with $\varphi = \frac{1}{|S_Q|}$ proves the theorem. □

5.3. Bounding the set width of treeplexes

As described in Section 3.1, MP, and other FOMs, rely on the set width as measured by the d.g.f.. In particular, their rate of convergences rely on the ratio of the two: the strong convexity parameter can be made smaller or larger by scaling the d.g.f., but this leads to increases or decreases in the set width. Thus, WLOG we can assume that the strong convexity modulus is 1.

Theorem 5.4 shows that the dilated entropy function with weights $\beta_j = |S_Q|\, 2^{d_j} M_{Q_j}$ for each $j \in S_Q$ when used as d.g.f. over a treeplex has a strong convexity modulus $\varphi \geq 1$ w.r.t. the ℓ_1 norm. We now investigate the set width over a treeplex when this d.g.f. is used.

THEOREM 5.9. *For a treeplex Q and weights $\beta_j = |S_Q|\, 2^{d_j} M_{Q_j}$ for each $j \in S_Q$, the maximum distance Ω between any two points in the treeplex as measured by the dilated entropy function is bounded by*

$$\frac{\Omega}{\varphi} \leq \Omega \leq |S_Q|\, d\, 2^d M_Q \log(m),$$

where d is the depth of Q and m is the maximum dimension of any Δ^j for $j \in S_Q$.

PROOF. The inequality $\frac{\Omega}{\varphi} \leq \Omega$ follows from Theorem 5.4 since we scale by an additional $|S_Q|$ compared to the weights used there. Now we bound Ω for any $q \in Q$. Note that the maximum value $\max_{q \in Q} \omega(q) = 0$ because we can choose a vector q where a single variable has value 1 for each simplex Δ^j such that $q_{p_j} > 0$.

For $\min_{q \in Q} \omega(q)$, first note that the entropy d.g.f. ω_E satisfies $\max_{z \in \Delta_n} \omega_E(z) \leq \log(n)$, and the dimension of each Δ^j can be bounded above by some m. Thus, we have

$$\sum_{j \in S_Q} \beta_j \sum_{i \in \mathbb{I}_j} q_i \log \frac{q_i}{q_{p_j}} = \sum_{j \in S_Q} \beta_j q_{p_j} \sum_{i \in \mathbb{I}_j} \frac{q_i}{q_{p_j}} \log \frac{q_i}{q_{p_j}} \geq - \sum_{j \in S_Q} \beta_j q_{p_j} \log(m).$$

Now consider any depth d'. By Lemma 5.7, we have

$$\begin{aligned} &- \sum_{j \in S_Q} \beta_j q_{p_j} \log(m) = - \sum_{j \in S_Q, d_j = d'} q_{p_j} |S_Q|\, 2^{d-d'} M_{Q_j} \log(m) \\ &\geq - \sum_{j \in S_Q, d_j = d'} q_{p_j} |S_Q|\, 2^{d-d'} (M_Q - d) \log(m) \geq - \sum_{j \in S_Q, d_j = d'} q_{p_j} |S_Q|\, 2^d M_Q \log(m). \end{aligned} \tag{18}$$

Summing up (18) over all depths gives

$$\min_{q \in Q} \omega(q) \geq - |S_Q|\, d\, 2^d M_Q \log(m).$$

Now we plug in our values for the maximum and minimum to get

$$\Omega \leq \max_{q \in Q} \omega(q) - \min_{q \in Q} \omega(q) \leq 0 - \big(- |S_Q|\, d\, 2^d M_Q \log(m)\big) = |S_Q|\, d\, 2^d M_Q \log(m),$$

which completes the proof. □

The ratio $\frac{\Omega}{\varphi}$ of maximum distance over the strong convexity parameter is important for FOMs that rely on a prox function, such as Nesterov's EGT, MP, and SMP. As mentioned previously, Hoda et al. [2010] only proved explicit bounds for the special case of uniform treeplexes. We next compare our bound for the general case to their bound for this specific case of uniform treeplexes.

Intuitively, uniform treeplexes start from a base treeplex, and construct a uniform treeplex by having each level have the same branching factor, and for every level, the subtrees rooted at those levels have the exact same structure, while branching is performed only on the base treeplex. Given bounds Ω_b, φ_b for the base treeplex, Hoda et al. [2010] achieve a bound of

$$\frac{\Omega}{\varphi} \leq O\left(|S_Q|^2\, d^2 M_Q^2 \frac{\Omega_b}{\varphi_b} \right)$$

for the overall treeplex. For the special case where the base treeplex is a simplex of dimension m, their bound for the dilated entropy function becomes

$$\frac{\Omega}{\varphi} \leq O\left(|S_Q|^2 \, d^2 M_Q^2 \log(m)\right).$$

Comparing Theorem 5.9 to their bound we see that our bound for the general class trades a factor of $O(|S_Q| \, d \, M_Q)$ for a factor $O(2^d)$. In many games, either of the terms $|S_Q|$ and M_Q would individually be much larger than 2^d.

5.4. Mirror prox algorithm for extensive-form game solving

We now describe how to instantiate MP for solving two-player zero-sum EFGs of the form given in (1) when $\mathcal{X}, \mathcal{Y}$ are treeplexes. This requires us to instantiate all the definitions from Section 3 for our problem. Because $\mathcal{X}$ and $\mathcal{Y}$ are treeplexes, it is immediately apparent that they are closed, convex, and bounded. We use the ℓ_1 norm on both of the embedding spaces E_x, E_y.

We use the dilated entropy function scaled with weights given in Theorem 5.9 as our d.g.f. for $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ that is compatible with the ℓ_1 norm. Then, Theorem 5.9 gives our bound on Ω_z. Because the dual norm of ℓ_1 norm is the ℓ_∞ norm, an upper bound on the Lipschitz constant of our monotone operator, namely $\mathcal{L}$, is given by:

$$\mathcal{L} = \max\left\{\mathcal{L}_1, \mathcal{L}_2\right\}, \text{where}$$

$$\mathcal{L}_1 = \max_{y \in \mathcal{Y}} \left| \max_j (Ay)_j - \min_j (Ay)_{ij} \right|, \mathcal{L}_2 = \max_{x \in \mathcal{X}} \left| \max_i (A^T x)_i - \min_i (A^T x)_i \right|.$$

Note that $\mathcal{L}$ is not at the scale of the maximum payoff difference in the original game. The differences involved in $\mathcal{L}_1, \mathcal{L}_2$ are scaled by the probability of the observed nature outcomes on the path of each sequence. Thus our Lipschitz constant $\mathcal{L}$ is exponentially smaller (in the number of observed nature steps on the path to the maximizing sequence) than the maximum payoff difference in the original EFG.

With our setup the convergence rate of MP with the dilated entropy function for solving BSPPs over treeplex domains (and EFG solving) is

$$O\left(\frac{|S_{\mathcal{Z}}| \, d \, 2^d M_{\mathcal{Z}} \mathcal{L} \max_{j \in S_{\mathcal{Z}}} \log(|\mathbb{I}_j|)}{T}\right).$$

This rate immediately improves the constants involved in many state-of-the-art FOMs with $O(1/\epsilon)$ convergence rate for EFGs. In particular, the convergence rate of the EGT algorithm is improved by the same amount by applying our d.g.f. and theoretical result. For SMP we further need unbiased gradient estimators.

6. SAMPLING

We now introduce a broad class of unbiased gradient estimators for EFGs. We describe how to generate a gradient estimate η of $x^T A$ given $x \in \mathcal{X}$, that is, the gradient for player 2. Given $y \in \mathcal{Y}$, the estimate ξ of $A^T y$ is generated analogously. This class explicitly uses the tree structure representation of EFGs, so we introduce some notation for it. We define H to be the set of nodes (or equivalently, histories) in the game tree. For any node h, we define $A(h)$ to be the set of actions available at h. The node reached by taking action $a \in A(h)$ at h is denoted by $h[a]$. We let $p_{h,x}$ to be the probability distribution over actions $A(h)$ given by the current iterate x. If h is a chance node, x has no effect on the distribution. We let $u_2(h)$ be the utility of player 2 for reaching a terminal node h, and η_h be the index in η corresponding to a given terminal node h.

An *EFG gradient estimator* is defined by a sampling-description function $\mathcal{C} : H \to \mathbb{N} \cup \{all\}$, where $\mathcal{C}(h)$ gives the number of samples drawn at the node $h \in H$. The estimate η of $x^T A$ is then generated using the following recursive sampling scheme:

$$\mathbf{Sample}(h,\tau) : \begin{cases} \text{if } h \text{ is a terminal node} : \eta_h = \tau \cdot u_2(h) \\ \text{else if } h \text{ belongs to player } 2\text{: } \forall a \in A(h) : \mathbf{Sample}(h[a],\tau) \\ \text{else if } \mathcal{C}(h) = all\text{: } \forall a \in A(h) : \mathbf{Sample}(h[a], p_{h,x}(a) \cdot \tau) \\ \text{else: } \begin{cases} \text{Draw } \left\{a_1, \ldots, a_{\mathcal{C}(h)}\right\} \sim p_{h,x}, \\ \forall j = 1, \ldots, \mathcal{C}(h) : \mathbf{Sample}(h[a_j], \tau \cdot \frac{1}{\mathcal{C}(h)}) \end{cases} \end{cases} .$$

Sampling is initiated with $h = r, \tau = 1$, where r is the root node of the game tree. From the definition of this sampling scheme, it is clear that the expected value of any EFG estimator is exactly $x^T A$ as desired. It is possible to generalize this class to sample $A(h)$ at nodes h belonging to player 2 as well.

Lanctot et al. [2009] introduced several unbiased sampling schemes for EFGs. These are easily described in terms of our general sampling scheme. *Chance sampling* is where a single sample is drawn if the node is a nature node, and otherwise all actions are chosen. This corresponds to the following EFG estimator:

$$\mathcal{C}_c(h) = \begin{cases} 1 & \text{if } h \text{ is a chance node} \\ all & \text{else} \end{cases} .$$

In *external sampling*, a single sample is drawn for the nodes that do not belong to the current player. So if we are estimating $x^T A$, $\mathcal{C}_e(h) = 1$.

In our experiments, for a given k, we focus on the following estimator

$$\mathcal{C}_{c,k}(h) = \begin{cases} k & \text{if } h \text{ is a chance node} \\ all & \text{else} \end{cases} . \tag{19}$$

7. THEORETICAL COMPARISON OF ALGORITHMS

Theorems 5.4 and 5.9 lead to the strongest known results on strong convexity parameter and set width of prox functions over treeplexes. As previously discussed, these results are not only relevant for the MP and SMP algorithms, but all FOMs based on d.g.f.s. In particular, they immediately improve the convergence rate of the EGT algorithm as well.

CFR is the most popular algorithm for practical large-scale equilibrium finding. It has a $O(\frac{1}{\epsilon^2})$ convergence rate; but its dependence on the number of information sets is only linear (and sometimes sublinear [Lanctot et al. 2009]). Thus, it can be more attractive for obtaining low-quality solutions quickly for games with many information sets. MCCFR has a similar convergence rate. When we apply SMP with our results, we achieve an expected convergence rate of $O(\frac{1}{\epsilon^2})$, but with only a square root dependence on the set width Ω in (4), similar to that of MCCFR. The SMP and MCCFR algorithms have much cheaper iterations compared to the deterministic algorithms.

There is a fundamental difference in how the constants of these algorithms depend on the number of information sets in the game. When the strategy space of player i is represented as a treeplex Q, the constant M_Q in our formulation essentially measures the branching factor over the other player and nature while ignoring branching from player i. The CFR bound has a similar constant, but it measures the branching over player i, while essentially taking the square root of the branching factor over the other player and nature. If players $1, 2$ and nature all have equal amounts of branching, then the two constants are the same. Otherwise, depending on the game, either one can be superior. For all of the above, we only measure nature branching in terms of outcomes observed by player i.

Gilpin et al. [2012] give an equilibrium-finding algorithm presented as $O(\ln(\frac{1}{\epsilon}))$, but this form of their bound has a dependence on a certain condition number of the A matrix: Their iteration bound for sequential games is $O(\frac{\|A\|_2 \cdot \ln(\|A\|_{2,2}/\epsilon) \cdot \sqrt{D}}{\delta(A)})$, where $\delta(A)$ is the condition number of A, $\|A\|_2 = \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}$ is the Euclidean matrix norm, and $D = \max_{x,\bar{x} \in \mathcal{X}, y, \bar{y} \in \mathcal{Y}} \|(x,y) - (\bar{x},\bar{y})\|_2^2$. Unfortunately, the condition number is only shown to be finite. Without any such unknown quantities based on condition numbers, Gilpin et al. [2012] show an upper bound of $O(\frac{\|A\|_2 \cdot D}{\epsilon})$ convergence rate. This bound, while at a $(\frac{1}{\epsilon})$ convergence rate as ours, suffers from worse constants. In particular, there exist matrices such that $\|A\|_2 = \sqrt{\|A\|_1 \|A\|_\infty}$, where the $\|A\|_1, \|A\|_\infty$ matrix norms are the maximum absolute column and row sums, respectively. Together with the value of D, this leads to a cubic dependence on the dimension of Q. For games where the players have roughly equal-size strategy spaces, this is equivalent to an M_Q^4 constant.

CFR, EGT, MP, and SMP all need to keep track of a constant number of current and/or average iterates, so the memory usage of all four algorithms is of the same order: When gradients are computed using an iterative approach as opposed to storing matrices or matrix decompositions, each algorithm requires a constant times the number of sequences in the sequence-form representation.

8. NUMERICAL COMPARISON OF ALGORITHMS

Based on the results derived in the previous sections, we investigate the practical performance of several first-order methods on EFGs. The algorithms that we consider are: EGT, MP, SMP, CFR, and MCCFR. For EGT, MP, and SMP we employ the dilated entropy function, with weights M_{Q_j} on each $j \in S_Q$, which, on a per-level basis, is proportional to the weights in Theorem 5.4. For EGT, we employ the balancing heuristic described by Hoda et al. [2010]. We run CFR with pruning, which was shown to improve performance by Lanctot et al. [2009]. For MCCFR and SMP we tried 4 different EFG gradient estimators: $\mathcal{C}_{c,1}, \mathcal{C}_{c,5}, \mathcal{C}_{c,10}, \mathcal{C}_{c,20}$ (c.f. (19)). In our experiments we only report the results of the best for each: MCCFR was best with $\mathcal{C}_{c,1}$ while SMP was best with $\mathcal{C}_{c,20}$. For SMP, rather than employing the very small fixed stepsize given in (4), we use a decreasing stepsize that approaches this value from above.

We test these algorithms on a scaled up variant of the poker game Leduc holdem [Southey et al. 2005], a benchmark problem in the incomplete-information game-solving community. In our version, the deck consists of k pairs of cards $1 \dots k$, for a total deck size of $2k$. Each player initially pays one chip to the pot, and is dealt a single private card. After a round of betting, a community card is dealt face up. After a subsequent round of betting, if neither player has folded, both players reveal their private cards. If either player pairs their card with the community card they win the pot. Otherwise, the player with the highest private card wins. In the event both players have the same private card, they draw and split the pot.

The results are shown in Figure 2. Each graph is a loglog plot that shows the results for a particular instance of Leduc with $6, 10, 16$ and 30 card decks, respectively. For each graph, we show the performance of all five algorithms, with the x-axis showing the number of nodes in the game tree touched, and the y-axis showing the maximum regret over the two players. By using the number of nodes touched, we can account for the pruning in the CFR version that we run, and constants such as the fact that the algorithms perform different numbers of tree traversals per iteration (for example, MP and SMP require two prox function applications). In line with what the theory predicts, the $O(\frac{1}{\epsilon})$ algorithms EGT and MP show a better convergence rate, but we also

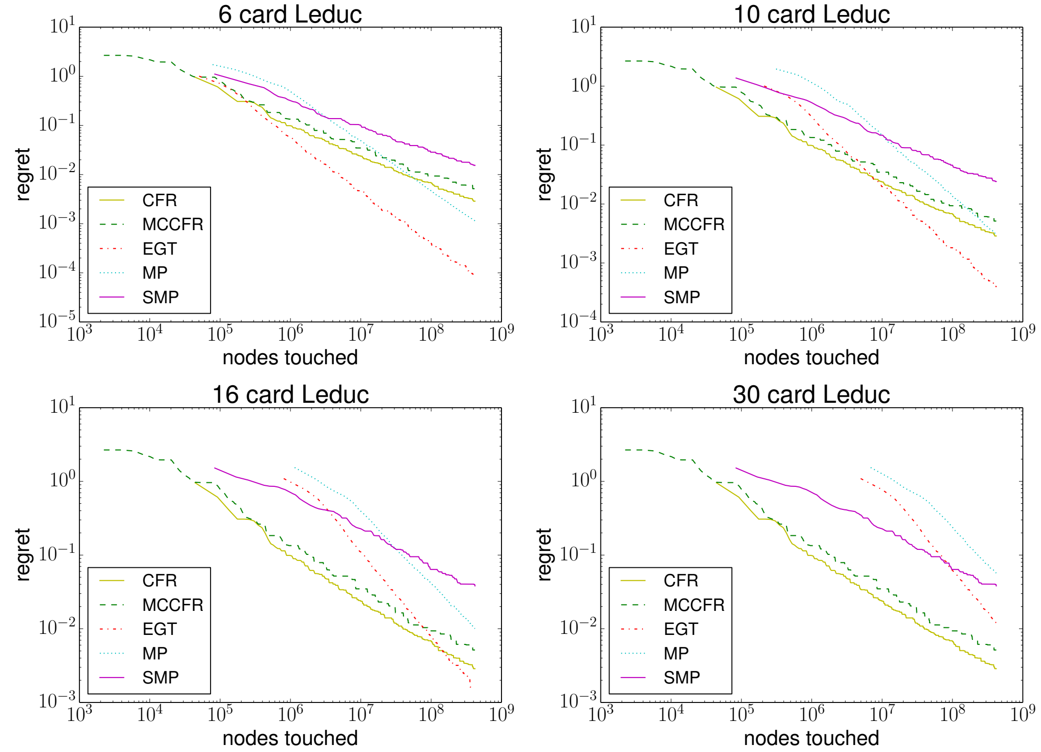

Fig. 2: Regret as a function of the number of nodes touched in four different variants of Leduc holdem for the CFR, MCCFR, EGT, MP, and SMP algorithms.

see that they initially start out slower. These results suggest that they might be better suited for high-precision equilibrium computation. SMP does not perform as well; but its performance improves as the instances get larger. While it is initially preferable to EGT and MP for the 30-card game, it generally performs poorly. One reason for this could be the small stepsizes required.

For these experiments we did not perform much engineering of the parameters of these algorithms. However, we did find that both MP and SMP are very sensitive to parameter selection. This suggests the possibility of improving their practical performance with a more thorough experimental investigation. We leave this as future work.

9. CONCLUSIONS

This paper investigated the problem of computing Nash equilibria in two-player zero-sum perfect-recall EFGs. On the theoretical side, we analyzed the strong convexity properties of the dilated entropy function over treeplexes. By introducing specific weights that are tied to the structure of the treeplex, we established much stronger bounds on the strong convexity parameter of this function over any given treeplex. In particular, these weights on each simplex in the treeplex are proportional to (1) the depth of the treeplex under the simplex, and (2) the maximum value of the ℓ_1 norm under the simplex. We then investigated the ratio $\frac{\Omega}{\varphi}$, the maximum distance measured by the dilated entropy function Ω over the strong convexity parameter φ. We showed that our new analysis trades off a factor $O(|S_Q|\, d\ M_Q)$ for a factor of 2^d, thus vastly improving the previous bounds on this ratio. Our results generalize to any treeplex, whereas the prior results were only for uniform treeplexes, a significant restriction. These results lead to significant improvements in the convergence rates of EGT, MP, and SMP for EFG solving. Finally, we designed a class of sampling schemes for gradient estimation, generalizing prior schemes considered for MCCFR.

We numerically investigated the performance of EGT, MP, SMP, CFR, and MCCFR. Our experiments showed that EGT and MP have desirable convergence rates, but also that they start out slower. Both MP and SMP are much more customizable than the others. Here we only considered their basic variants. It is of interest to investigate various step sizes and parameter choices for MP to see if it achieves similar or better performance than EGT. Our experiments also suggest that there might be a benefit to applying FOMs such as these in large-scale game solving as opposed to the current common practice which centers around CFR variants.

Another question that we leave open, theoretically and experimentally, is how well our approach interacts with the abstraction approaches mentioned in Section 2. CFR works well on such abstractions even though no theoretical justification exists [Lanctot et al. 2012]. A promising research direction is to develop similar results for FOMs.

REFERENCES

BOSANSKY, B., KIEKINTVELD, C., LISY, V., AND PECHOUCEK, M. 2014. An exact double-oracle algorithm for zero-sum extensive-form games with imperfect information. *Journal of Artificial Intelligence Research*, 829–866.

BROWN, N., GANZFRIED, S., AND SANDHOLM, T. 2015. Hierarchical abstraction, distributed equilibrium computation, and post-processing, with application to a champion no-limit texas hold 'em agent. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.

BROWN, N. AND SANDHOLM, T. 2014. Regret transfer and parameter optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

DASKALAKIS, C., DECKELBAUM, A., AND KIM, A. 2014. Near-optimal no-regret algorithms for zero-sum games. *Games and Economic Behavior*.

GILPIN, A., PEÑA, J., AND SANDHOLM, T. 2012. First-order algorithm with $\mathcal{O}(\ln(1/\epsilon))$ convergence for ϵ-equilibrium in two-person zero-sum games. *Mathematical Programming 133,* 1–2, 279–298. Conference version appeared in AAAI-08.

GILPIN, A. AND SANDHOLM, T. 2007. Lossless abstraction of imperfect information games. *Journal of the ACM 54,* 5. Early version in EC-06.

HAWKIN, J., HOLTE, R., AND SZAFRON, D. 2011. Automated action abstraction of imperfect information extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

HAWKIN, J., HOLTE, R., AND SZAFRON, D. 2012. Using sliding windows to generate action abstractions in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

HIRIART-URRUTY, J.-B. AND LEMARÉCHAL, C. 2001. *Fundamentals of convex analysis*.

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, 494–512. Conference version appeared in WINE-07.

JIANG, A. AND LEYTON-BROWN, K. 2011. Polynomial-time computation of exact correlated equilibrium in compact games. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*.

JUDITSKY, A. AND NEMIROVSKI, A. 2011a. First order methods for nonsmooth convex large-scale optimization, i: general purpose methods. *Optimization for Machine Learning*, 121–148.

JUDITSKY, A. AND NEMIROVSKI, A. 2011b. First order methods for nonsmooth convex large-scale optimization, ii: utilizing problems structure. *Optimization for Machine Learning*, 149–183.

JUDITSKY, A., NEMIROVSKI, A., AND TAUVEL, C. 2011. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems 1,* 1, 17–58.

KOLLER, D., MEGIDDO, N., AND VON STENGEL, B. 1996. Efficient computation of equilibria for extensive two-person games. *Games and Economic Behavior*.

KROER, C. AND SANDHOLM, T. 2014a. Extensive-form game abstraction with bounds. In *Proceedings of the ACM Conference on Economics and Computation (EC)*.

KROER, C. AND SANDHOLM, T. 2014b. Extensive-form game imperfect-recall abstractions with bounds. arXiv.

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)*.

LANCTOT, M., WAUGH, K., ZINKEVICH, M., AND BOWLING, M. 2009. Monte Carlo sampling for regret minimization in extensive games. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*. 1078–1086.

LIPTON, R., MARKAKIS, E., AND MEHTA, A. 2003. Playing large games using simple strategies. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*.

LITTMAN, M. AND STONE, P. 2003. A polynomial-time Nash equilibrium algorithm for repeated games. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*.

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, 229–251.

NESTEROV, Y. 2005. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal of Optimization 16,* 1, 235–249.

ROMANOVSKII, I. 1962. Reduction of a game with complete memory to a matrix game. *Soviet Mathematics 3*, 678–681.

SANDHOLM, T. 2010. The state of solving large incomplete-information games, and application to poker. *AI Magazine*, 13–32. Special issue on Algorithmic Game Theory.

SANDHOLM, T. AND SINGH, S. 2012. Lossy stochastic game abstraction with bounds. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*.

SHI, J. AND LITTMAN, M. 2002. Abstraction methods for game theoretic poker. In *CG '00: Revised Papers from the Second International Conference on Computers and Games*.

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)*. 550–558.

VON STENGEL, B. 1996. Efficient computation of behavior strategies. *Games and Economic Behavior 14,* 2, 220–246.

WAUGH, K. AND BAGNELL, J. A. 2015. A unified view of large-scale zero-sum equilibrium computation. In *AAAI Workshop on Computer Poker and Imperfect Information*.

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)*.