

Pseudonorm Approachability and Applications to Regret Minimization

Christoph Dann

Google Research, Zurich, Switzerland

CDANN@CDANN.NET

Yishay Mansour

Google Research, Tel Aviv, Israel

MANSOUR@GOOGLE.COM

Mehryar Mohri

Google Research, New York, NY 10011

MOHRI@GOOGLE.COM

Jon Schneider

Google Research, New York, NY 10011

JSCHNEI@GOOGLE.COM

Balubramanian Sivan

Google Research, New York, NY 10011.

BALUSIVAN@GOOGLE.COM

Editors: Shipra Agrawal and Francesco Orabona

Abstract

Blackwell’s celebrated approachability theory provides a general framework for a variety of learning problems, including regret minimization. However, Blackwell’s proof and implicit algorithm measure approachability using the ℓ_2 (Euclidean) distance. We argue that in many applications such as regret minimization, it is more useful to study approachability under other distance metrics, most commonly the ℓ_∞ -metric. But, the time and space complexity of the algorithms designed for ℓ_∞ -approachability depend on the dimension of the space of the vectorial payoffs, which is often prohibitively large. Thus, we present a framework for converting high-dimensional ℓ_∞ -approachability problems to low-dimensional *pseudonorm* approachability problems, thereby resolving such issues. We first show that the ℓ_∞ -distance between the average payoff and the approachability set can be equivalently defined as a *pseudodistance* between a lower-dimensional average vector payoff and a new convex set we define. Next, we develop an algorithmic theory of pseudonorm approachability, analogous to previous work on approachability for ℓ_2 and other norms, showing that it can be achieved via online linear optimization (OLO) over a convex set given by the Fenchel dual of the unit pseudonorm ball. We then use that to show, modulo mild normalization assumptions, that there exists an ℓ_∞ -approachability algorithm whose convergence is independent of the dimension of the original vectorial payoff. We further show that that algorithm admits a polynomial-time complexity, assuming that the original ℓ_∞ -distance can be computed efficiently. We also give an ℓ_∞ -approachability algorithm whose convergence is logarithmic in that dimension using an FTRL algorithm with a maximum-entropy regularizer. Finally, we illustrate the benefits of our framework by applying it to several problems in regret minimization.

Keywords: Blackwell’s approachability, regret minimization, swap regret.

1. Introduction

The notion of approachability introduced by Blackwell (1956) can be viewed as an extension of von Neumann’s minimax theorem (von Neumann, 1928) to the case of vectorial payoffs. Blackwell gave a simple example showing that the straightforward analog of von Neumann’s minimax theorem does not hold for vectorial payoffs. However, in contrast with this negative result for one-shot games, he proved that, in repeated games, a player admits an adaptive strategy guaranteeing that their average payoff *approaches* a closed convex set in the limit, provided that the set satisfies a natural separability condition.

The theory of Blackwell approachability is intimately connected with the field of online learning for the reason that the problem of *regret minimization* can be viewed as an approachability problem: in particular, the learner would like their vector of regrets (with respect to each competing benchmark) to converge to a non-positive vector. In this vein, Abernethy et al. (2011) demonstrated how to use algorithms for approachability to solve a general class of regret minimization problems (and conversely, how to use regret minimization to construct approachability algorithms). However, applying their reduction sometimes leads to suboptimal regret guarantees – for example, for the specific case of minimizing external regret over T rounds with K actions, their reduction results in an algorithm with $O(\sqrt{TK})$ regret (instead of the optimal $O(\sqrt{T \log K})$ regret bound achievable by e.g. multiplicative weights).

One reason for this suboptimality is the choice of distance used to define approachability. Both Blackwell and Abernethy et al. consider approachability algorithms that minimize the Euclidean (ℓ_2) distance between their average payoff and the desired set. We argue that, for applications to regret minimization, it is often more useful to study approachability under other distance metrics, most commonly approachability under the ℓ_∞ metric to the *non-positive orthant*, which is well suited to capture the fact that regret is a *maximum* over various competing benchmarks. This has been observed in the past in several recent publications (Perchet, 2015; Shimkin, 2016; Kwon, 2021). In particular, by constructing algorithms for ℓ_∞ approachability, it is possible to naturally recover a $O(\sqrt{T \log K})$ external regret learning algorithm (and algorithms with optimal regret guarantees for many other problems of interest).

However, there is still one significant problem with developing regret minimization algorithms via ℓ_∞ approachability (or any of the forms of approachability previously mentioned): the time and space complexity of these algorithms depends polynomially on the dimension d of the space of vectorial payoffs, which in turn equals the number of benchmarks we compete against in our regret minimization problem. In some regret minimization settings, this can be prohibitively expensive. For example, in the setting of swap regret (where the benchmarks are parameterized by the $d = K^K$ swap functions mapping $[K]$ to $[K]$), this results in algorithms with complexity exponential in K . On the other hand, there exist algorithms, e.g. (Blum and Mansour, 2007), which are both efficient (poly(K) time and space) and obtain optimal regret guarantees.

1.1. Main results

In this paper, we present a framework for converting *high-dimensional* ℓ_∞ approachability problems to *low-dimensional* “pseudonorm” approachability problems, in turn resolving many of these issues. To be precise, recall that the setting of approachability can be thought of as a T -round repeated game, where in round t the learner chooses an action p_t from some convex “action” set $\mathcal{P} \subseteq \mathbb{R}^n$, the adversary simultaneously chooses an action ℓ_t from some convex “loss” set $\mathcal{L} \subseteq \mathbb{R}^m$, and the

learner receives a vector-valued payoff $u(p_t, \ell_t)$, where $u: \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}^d$ is a d -dimensional bilinear function. The learner would like the ℓ_∞ distance between their average payoff $\frac{1}{T} \sum u(p_t, \ell_t)$ and some convex set¹ \mathcal{S} to be as small as possible.

We first demonstrate how to construct a new d' -dimensional bilinear function $\tilde{u}: \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}^{d'}$, a new convex set $\mathcal{S}' \subseteq \mathbb{R}^{d'}$, and a pseudonorm² f such that the “pseudodistance” between the average modified payoff $\frac{1}{T} \sum \tilde{u}(p_t, \ell_t)$ and \mathcal{S}' is equal to the ℓ_∞ distance between the original average payoff and \mathcal{S} . Importantly, the new dimension d' is equal to mn and is independent of the original dimension d .

We then develop an algorithmic theory of pseudonorm approachability analogous to that developed in (Abernethy et al., 2011) for the ℓ_2 norm and (Shimkin, 2016; Kwon, 2021) for other norms, showing that, in order to perform pseudonorm approachability, it suffices to be able to perform on-line linear optimization (OLO) over a convex set given by the Fenchel dual of the unit pseudonorm ball (and that the rate of approachability is directly related to the regret guarantees of this OLO subalgorithm). This has the following consequences for approachability:

- First, by solving this OLO problem with a quadratic regularized Follow-The-Regularized-Leader (FTRL) algorithm, we show (modulo mild normalization assumptions on the sizes of \mathcal{P} , \mathcal{L} , and u) that there exists a pseudonorm approachability algorithm (and hence an ℓ_∞ approachability algorithm for the original problem) which converges at a rate of $O(nm/\sqrt{T})$. We additionally provide a stronger bound on the rate which scales as $O(D_u D_p D_\ell / \sqrt{T})$ where $D_p = \text{diam } \mathcal{P}$, $D_\ell = \text{diam } \mathcal{L}$, and D_u is the maximum ℓ_2 norm of the set of vectors formed by taking the coefficients of components of u (Theorem 13). In comparison, the best-known generic guarantee for ℓ_∞ -approachability prior to this work converged at a d -dependent rate of $O(\sqrt{(\log d)/T})$.
- Second, we show that as long as we can evaluate the original ℓ_∞ -distance between $\frac{1}{T} \sum u(p_t, \ell_t)$ and $(-\infty, 0]^d$ efficiently, we can implement the above algorithm in $\text{poly}(m, n)$ time per round (Theorem 19). This has the following natural consequence for the class of regret minimization problems that can be written as ℓ_∞ -approachability problems: *if it is possible to efficiently compute some notion of regret for a sequence of losses and actions, then there is an efficient (in the dimensions of the actions and losses) learner that minimizes this regret.*
- Finally, in some cases, the $O(\sqrt{(\log d)/T})$ approachability rate from (inefficient) ℓ_∞ -approachability outperforms the rate obtained by the quadratic regularized FTRL algorithm. We define a new regularizer whose value is given by finding the maximum entropy distribution of a subset of distributions of support d , and show that, by using this regularizer, we recover this $O(\sqrt{(\log d)/T})$ rate. In particular, whenever we can efficiently compute this maxent regularizer, there is an efficient learning algorithm with a $O(\sqrt{(\log d)/T})$ approachability rate.

1. For simplicity, throughout this paper we assume \mathcal{S} to be the negative orthant $(-\infty, 0]^d$, as this is the case most relevant to regret minimization. However, much of our results extend straightforwardly to arbitrary convex sets.

2. In this paper a *pseudonorm* is a function $f: \mathbb{R}^{d'} \rightarrow \mathbb{R}_{\geq 0}$ which satisfies most of the properties of a norm (e.g. positive homogeneity, triangle inequality), but may be asymmetric (there may exist x where $f(x) \neq f(-x)$) and may not be definite (there may exist $x \neq 0$ where $f(x) = 0$). Just as a norm $\|\cdot\|$ defines a distance between $x, y \in \mathbb{R}^{d'}$ via $\|x - y\|$, a pseudonorm f defines the pseudodistance $f(x - y)$.

We then apply our framework to various problems in regret minimization:

- We show that our framework straightforwardly recovers a regret-optimal and efficient algorithm for swap regret minimization. Doing so requires computing the above maximum entropy regularizer for this specific case, where we show that it has a nice closed form. In particular, to our knowledge, this is the first approachability-based algorithm for swap regret that both is efficient and has the optimal minimax regret.
- In Section 4.3, we apply our framework to develop the first efficient contextual learning algorithms with low *Bayesian swap regret*. Such algorithms have the property that if learners employ them in a repeated Bayesian game, the time-average of their strategies will converge to a Bayesian correlated equilibrium, a well-studied equilibrium notion in game theory (see e.g. [Bergemann and Morris \(2016\)](#)).

This notion of Bayesian swap regret was recently introduced by [Mansour et al. \(2022\)](#), who also provided an algorithm with low Bayesian swap regret, albeit one that is not computationally efficient. By applying our framework, we easily obtain an efficient contextual learning algorithm with $O(CK\sqrt{T})$ Bayesian swap regret, resolving an open question of [Mansour et al. \(2022\)](#) (here C is the number of “contexts” / “types” of the learner).

- In Section 4.4, we further analyze the application of our general ℓ_∞ -approachability theory and algorithm to the analysis of reinforcement learning (RL) with vectorial losses. We point out how our framework can provide a general solution in the full information setting with known transition probabilities and how we can recover the best known solution for standard regret minimization in episodic RL. More importantly, we show how our framework and algorithm can lead to an algorithm for constrained MDPs with a significantly more favorable regret guarantee, logarithmic in the number of constraints k , in contrast with the \sqrt{k} -dependency of the results of [Miryoosefi et al. \(2019\)](#).

1.2. Related Work

There is a wide literature dealing with various aspects of Blackwell’s approachability, including its applications to game theory, regret minimization, reinforcement learning, and multiple extensions.

[Hart and Mas-Colell \(2000\)](#) described an adaptive procedure for players in a game based on Blackwell’s approachability, which guarantees that the empirical distributions of the plays converges to the set of a correlated equilibrium. This procedure is related to internal regret minimization, for which, as shown by [Foster and Vohra \(1999\)](#), the existence of an algorithm follows the proof of [Hart and Mas-Colell \(2000\)](#). [Hart and Mas-Colell \(2001\)](#) further gave a general class of adaptive strategies based on approachability. Approachability has been widely used for calibration ([Dawid, 1982](#)), see [Foster and Hart \(2018\)](#) for a recent work on the topic. Approachability and partial monitoring were studied in a series of publications by [Perchet \(2010\)](#); [Mannor et al. \(2014a,b\)](#); [Perchet and Quincampoix \(2015, 2018\)](#); [Kwon and Perchet \(2017\)](#). More recently, approachability has also been used in the analysis of fairness in machine learning ([Chzhen et al., 2021](#)).

Approachability has also been extensively used in the context of reinforcement learning. [Mannor and Shimkin \(2003\)](#) discussed an extension of regret minimization in competitive Markov decision processes (MDPs) whose analysis is based on Blackwell’s approachability theory. [Mannor and](#)

Shimkin (2004) presented a geometric approach to multiple-criteria reinforcement learning formulated as approachability conditions. Kalathil et al. (2014) presented strategies for approachability for MDPs and Stackelberg stochastic games based on Blackwell’s approachability theory. More recently, Miryoosefi et al. (2019) used approachability to derive solutions for reinforcement learning with convex constraints.

The notion of approachability was further extended in several studies. Vieille (1992) used differential games with a fixed duration to study weak approachability in finite dimensional spaces. Spinat (2002) formulated a necessary and sufficient condition for approachability of non-necessary convex sets. Lehrer (2003) extended Blackwell’s approachability theory to infinite-dimensional spaces.

The most closely related work to this paper, which we build upon, is that of Abernethy et al. (2011) who showed that, remarkably, any algorithm for Blackwell’s approachability could be converted into one for online convex optimization and vice-versa. Bernstein and Shimkin (2015) also discussed a related response-based approachability algorithm.

Perchet (2015) presented a specific study of ℓ_∞ approachability, for which they gave an exponential weight algorithm. Shimkin (2016, Section 5) studied approachability for an arbitrary norm and gave a general duality result for an arbitrary norm using Sion’s minmax theorem. The pseudonorm duality theorem we prove, using Fenchel duality, can be viewed as a generalization. Kwon (2016, 2021) also presented a duality theorem similar to that of Shimkin (2016) which they used to derive a FTRL algorithm for general norm approachability. They further treated the special case of internal and swap regret. However, unlike the algorithms derived in this work, the computational complexity of their swap regret algorithm is in $O(K^K)$. This is also true of the paper of Perchet (2015) which also analyzes the swap regret problem.

It is known that if all players follow a swap regret minimization algorithm, then the empirical distribution of their play converges to a correlated equilibrium (Blum and Mansour, 2007). Hazan and Kale (2008) showed a result generalizing this property to the case of Φ -regret and Φ -equilibrium, where the Φ -regret is the difference between the cumulative expected loss suffered by the learner and that of the best Φ -modification of the sequence in hindsight. Gordon et al. (2008) further generalized the results of Hazan and Kale (2008) to a more general class of Φ -modification regrets. The algorithms discussed in (Gordon et al., 2008) are distinct from those discussed in this paper (they do not clearly extend to the general approachability setting, and they require significantly different computational assumptions than ours). Nevertheless, they bare some similarity with our work.

2. Preliminaries

Notation. We use $[n]$ as a shorthand for set $\{1, 2, \dots, n\}$. We write $\Delta_d = \{x \in \mathbb{R}^d \mid x_i \geq 0, \sum x_i = 1\}$ to denote the simplex over d dimensions and $\bar{\Delta}_d = \{x \in \mathbb{R}^d \mid x_i \geq 0, \sum x_i \leq 1\}$ to denote the convex hull of the d -simplex with the origin. $\text{conv}(S)$ denotes the convex hull of the points in S , and $\text{cone}(S) = \{\alpha x \mid \alpha \geq 0, x \in \text{conv}(S)\}$ the convex cone generated by the points in S .

Some of the more standard proofs have been deferred to Appendix A.

2.1. Blackwell approachability and regret minimization

We begin by illustrating the theory of Blackwell approachability for the specific case of the ℓ_∞ -distance; this case is both particularly suited to the application of regret minimization, and will play an important role in the results (e.g. reductions to pseudonorm approachability) that follow.

We consider a repeated game setting, where every round t a learner chooses an action p_t belonging to a bounded³ convex set $\mathcal{P} \subseteq [-1, 1]^n$, and an adversary simultaneously chooses a loss ℓ_t belonging to a bounded convex set $\mathcal{L} \subseteq [-1, 1]^m$. Let $u: \mathcal{P} \times \mathcal{L} \rightarrow [-1, 1]^d$ be a bounded bilinear⁴ vector-valued *payoff function*, and let $\mathcal{S} \subseteq \mathbb{R}^d$ be a closed convex set with the property that for every $\ell \in \mathcal{L}$, there exists a $p \in \mathcal{P}$ such that $u(p, \ell) \in \mathcal{S}$ (we say that such a set is “separable”). When $d = 1$, the minimax theorem implies that there exists a single $p \in \mathcal{P}$ such that for all ℓ , $u(p, \ell) \in \mathcal{S}$.

This is not true for $d > 1$, but the theory of Blackwell approachability provides the following algorithmic analogue of this statement. Define a *learning algorithm* \mathcal{A} to be a collection of functions $\mathcal{A}_t: \mathcal{L}^{t-1} \rightarrow \mathcal{P}$ for each $t \in [T]$, where \mathcal{A}_t describes how the learner decides their action p_t as a function of the observed losses $\ell_1, \ell_2, \dots, \ell_{t-1}$ up until time $t - 1$. Blackwell approachability guarantees that there exists a learning algorithm \mathcal{A} such that when \mathcal{A} is run on any loss sequence ℓ the resulting action sequence \mathbf{p} has the property that:

$$\lim_{T \rightarrow \infty} d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), \mathcal{S} \right) = 0. \quad (1)$$

(Here for $v, w \in \mathbb{R}^d$, $d_\infty(v, w) = \max_i |v_i - w_i|$ represents the ℓ_∞ distance between v and w).

As mentioned, one of the main motivations for studying Blackwell approachability is its connections to regret minimization. In particular, for a fixed choice of u , define

$$\text{Reg}(\mathbf{p}, \ell) = \max \left(\max_{i \in [d]} \left(\sum_{t=1}^T u_i(p_t, \ell_t) \right), 0 \right). \quad (2)$$

Note first that this definition of “regret” is exactly T times the ℓ_∞ approachability distance in the case where $\mathcal{S} = (\infty, 0]^d$ is the negative orthant; that is,

$$\text{Reg}(\mathbf{p}, \ell) = T \cdot d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d \right). \quad (3)$$

But secondly, note that by choosing \mathcal{P} , \mathcal{L} , and u carefully, the definition (2) can capture a wide variety of forms of regret studied in regret minimization. For example:

- When $\mathcal{P} = \Delta_K$, $\mathcal{L} = [0, 1]^K$, $d = K$, and $u(p, \ell)_i = \langle p, \ell \rangle - \ell_i$, $\text{Reg}(\mathbf{p}, \ell)$ is the *external regret* of playing action sequence \mathbf{p} against loss sequence ℓ ; i.e., it measures the regret compared to the best single action.

3. We bound the entries of \mathcal{P} , \mathcal{L} , and u within $[-1, 1]$ for convenience, but it is generally easy to translate between different boundedness assumptions (since almost all relevant quantities are linear). We express the majority of our theorem statements (with the notable exception of Theorem 13) in a way that is independent of the choice of bounds.

4. We briefly note that all our results also hold for *biaffine* functions u_i ; in particular, extending the loss and action sets slightly (by replacing \mathcal{P} and \mathcal{L} with $\mathcal{P} \times \{1\}$ and $\mathcal{L} \times \{1\}$) allows us to write any biaffine function over the original sets as a bilinear function over the extended sets.

- When $\mathcal{P} = \Delta_K$, $\mathcal{L} = [0, 1]^K$, $d = K^K$, and (for each function $\pi: [K] \rightarrow [K]$), $u(p, \ell)_\pi = \sum_{i=1}^K p_i(\ell_i - \ell_{\pi(i)})$, $\text{Reg}(p, \ell)$ is the *swap regret* of playing action sequence p against loss sequence ℓ ; i.e., it measures the regret compared to the best action sequence p' obtained by applying a fixed swap function to sequence p .
- When $\mathcal{P} \subseteq \mathbb{R}^n$ is a convex polytope, $\mathcal{L} = [0, 1]^n$, $d = |V(\mathcal{P})|$ (where $V(\mathcal{P})$ is the set of vertices of \mathcal{P}), and (for each vertex $v \in V(\mathcal{P})$), $u(p, \ell)_v = \langle p, \ell \rangle - \langle v, \ell \rangle$, this captures the (external) regret from performing online linear optimization over the polytope \mathcal{P} (see Section 2.2).
- Finally, to illustrate the power of this framework, we present an unusual swap regret minimization that we call “Procrustean swap regret minimization” (after the orthogonal Procrustes problem, see (Gower and Dijkstra, 2004)). Let $\mathcal{P} = \{x \in \mathbb{R}^n : \|x\|_2 \leq 1\}$ be the unit ball in n dimensions, $\mathcal{L} = [-1, 1]^n$, and, for each orthogonal matrix⁵ $Q \in O(n)$, let $u(p, \ell)_Q = \langle p, \ell \rangle - \langle Qp, \ell \rangle$.

When the negative orthant $(-\infty, 0]^d$ is separable with respect to u , which is true in all of the above examples, the theory of Blackwell approachability immediately guarantees the existence of a sublinear regret learning algorithm \mathcal{A} for the corresponding notion of regret. Specifically, define the regret of a learning algorithm \mathcal{A} to be the worst-case regret over all possible loss vectors $\ell \in \mathcal{L}^T$; i.e.,

$$\text{Reg}(\mathcal{A}) = \max_{\ell \in \mathcal{L}^T} \text{Reg}(p, \ell), \quad (4)$$

where $p_t = \mathcal{A}_t(\ell_1, \dots, \ell_{t-1})$. Then (1) implies that the same algorithm \mathcal{A} satisfies $\text{Reg}(\mathcal{A}) = o(T)$. Motivated by this application, we will restrict our attention for the remainder of this paper to the setting where $\mathcal{S} = (-\infty, 0]^d$ and will assume (unless otherwise specified) that this \mathcal{S} is separable with respect to the bilinear function u we consider.

In fact, the theory of ℓ_∞ -approachability is constructive and allows us to write down explicit algorithms \mathcal{A} along with explicit (and in many cases, near optimal) regret bounds. However, before we introduce these algorithms, we will need to introduce the problem of online linear optimization.

2.2. Online linear optimization

Here we discuss algorithms for *online linear optimization* (OLO), a special case of online convex optimization where all the loss functions are linear functions of the learner’s action. These will form an important primitive of our algorithms for approachability.

Let \mathcal{X}, \mathcal{Y} be two bounded convex subsets of \mathbb{R}^r and consider the following learning problem. Every round t (for T rounds) our learning algorithm must choose an element $x_t \in \mathcal{X}$ as a function of y_1, y_2, \dots, y_{t-1} . The adversary simultaneously chooses a loss “function” $y_t \in \mathcal{Y}$. This causes the learner to incur a loss of $\langle x_t, y_t \rangle$ in round t . The goal of the learner is to minimize their regret, defined as the difference between their total loss and the loss they would have incurred by playing the best fixed action in hindsight, i.e.,

5. Technically, this leads to an infinite dimensional u (since the group of orthogonal matrices is infinite), but one can instead take an arbitrarily fine discrete approximation of the set. Indeed, one of the advantages of the results we present is that they are largely independent of the dimension d of u .

$$\text{Reg}(\mathbf{x}, \mathbf{y}) = \max_{x^* \in \mathcal{X}} \left(\sum_{t=1}^T \langle x_t, y_t \rangle - \sum_{t=1}^T \langle x^*, y_t \rangle \right).$$

There are many similarities between this problem and the approachability and regret minimization problems discussed in Section 2.1. For example, if we take $\mathcal{X} = \mathcal{P} = \Delta_K$ and $\mathcal{Y} = \mathcal{L} = [0, 1]^K$, then OLO is equivalent to the problem of external regret minimization. However, not all regret minimization problems can be written directly as an instance of OLO – for example, there is no clear way to write swap regret minimization as an OLO instance. Eventually we will demonstrate how to apply OLO as a subroutine to solve any regret minimization problem, but this will involve a reduction to Blackwell approachability and will require running OLO on different spaces than the action/loss sets \mathcal{P} and \mathcal{L} directly (which is why we distinguish the action/loss sets for OLO as \mathcal{X} and \mathcal{Y} respectively).

There is an important subclass of algorithms for OLO known as *Follow the Regularized Leader* (FTRL) algorithms (Shalev-Shwartz, 2007; Abernethy et al., 2008). An FTRL algorithm is completely specified by a strongly convex function $R: \mathcal{X} \rightarrow \mathbb{R}$, and plays the action

$$x_t = \underset{x \in \mathcal{X}}{\text{argmin}} \left(R(x) + \sum_{s=1}^{t-1} \langle x, y_s \rangle \right).$$

In words, this algorithm plays the action that minimizes the total loss on the rounds until the present (“following the leader”), subject to an additional regularization term. It is possible to characterize the worst-case regret of an FTRL algorithm in terms of properties of the regularizer R and the sets \mathcal{X} and \mathcal{Y} (see e.g. Theorem 15 of Hazan (2016)). For our purposes, we will only need the following two results for specific regularizers.

Lemma 1 (Quadratic regularizer, (Zinkevich, 2003)) *Let $D_x = \text{diam } \mathcal{X}$ and $D_y = \max_{y \in \mathcal{Y}} \|y\|_2$. Let x_0 be an arbitrary element of \mathcal{X} , and let $R(x) = \|x - x_0\|^2$. Then, the FTRL algorithm with regularizer R incurs worst-case regret at most $O(D_x D_y \sqrt{T})$.*

Lemma 2 (Negative entropy regularizer, (Kivinen and Warmuth, 1995)) *Let $\mathcal{X} = \Delta_d$, $\mathcal{Y} = [0, 1]^d$, and let $R(x) = \sum_{i=1}^d x_i \log x_i$ (where we extend this to the boundary of \mathcal{X} by letting $0 \log 0 = 0$). Then, the FTRL algorithm with regularizer R incurs worst-case regret at most $O(\sqrt{T \log d})$.*

2.3. Algorithms for ℓ_∞ -approachability

We can now write down an explicit description of our algorithm for ℓ_∞ -approachability (in terms of a blackbox OLO algorithm) and get quantitative bounds on the rate of convergence in the LHS of (1). Let \mathcal{F} be an OLO algorithm for the sets $\mathcal{X} = \overline{\Delta}_d$ and $\mathcal{Y} = [-1, 1]^d$. Then we can describe our algorithm \mathcal{A} for ℓ_∞ -approachability as Algorithm 1.

It turns out that we can relate the ℓ_∞ approachability distance to $(-\infty, 0]^d$ (and hence the regret of this algorithm \mathcal{A}) to the regret of our OLO algorithm \mathcal{F} .

Theorem 3 *We have that*

$$\text{Reg}(\mathcal{A}) = T \cdot d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d \right) \leq \text{Reg}(\mathcal{F}).$$

Algorithm 1: Description of Algorithm \mathcal{A} for ℓ_∞ -approachability

Initialize θ_1 to an arbitrary point in $\overline{\Delta}_d$;

for $t \leftarrow 1$ **to** T **do**

Choose $p_t \in \mathcal{P}$ so that for all $\ell \in \mathcal{L}$, $\langle \theta_t, u(p_t, \ell) \rangle \leq 0$. Such a p_t is guaranteed to exist by the separability condition on u , since $\langle \theta, s \rangle \leq 0$ for any $\theta \in \overline{\Delta}_d$ and $s \in (-\infty, 0]^d$;

Play action p_t and receive as feedback $\ell_t \in \mathcal{L}$;

Set $y_t = -u(p_t, \ell_t)$;

Set $\theta_{t+1} = \mathcal{F}(y_1, y_2, \dots, y_t)$.

end

If we let \mathcal{F} be the negative entropy FTRL algorithm (Lemma 2)⁶, we obtain the following regret guarantee for \mathcal{A} .

Corollary 4 *For any bilinear regret function $u: \mathcal{P} \times \mathcal{L} \rightarrow [-1, 1]^d$, there exists a regret minimization algorithm \mathcal{A} with worst-case regret $\text{Reg}(\mathcal{A}) = O(\sqrt{T \log d})$.*

Equivalently, Corollary 4 can be interpreted as saying that there exists an ℓ_∞ -approachability algorithm which approaches the negative orthant at an average rate of $O(\sqrt{(\log d)/T})$. In general, (3) lets us straightforwardly convert between results for approachability and results for regret minimization. Throughout the remainder of the paper we will primarily phrase our results in terms of $\text{Reg}(\mathcal{A})$, but switch between quantities of interest when convenient.

3. Main Results

Already Corollary 4 leads to a number of impressive consequences. For example, when applied to the problem of swap-regret minimization (where u has dimension $d = K^K$), it leads to a learning algorithm with $O(\sqrt{KT \log K})$ regret, matching the best known regret bounds for this problem (Blum and Mansour, 2007). However, the algorithm we obtain in this way has two unfortunate properties.

First, since u is d -dimensional, implementing the algorithm as written above requires $O(\text{poly}(d))$ time and space complexity (even storing any specific y_t or θ_t requires $\Omega(d)$ space). This is fine if d is small, but in many of our applications d is much (e.g., exponentially) larger than the dimensions m and n of the loss and action sets. For example, for swap regret we have $d = K^K$ but $m = n = K$. Although Corollary 4 gives us an optimal $O(\sqrt{KT \log K})$ swap regret algorithm, it takes exponential time / space to implement (in contrast to other known swap regret algorithms, such as that of Blum and Mansour (2007)).

Secondly, although Corollary 4 has only a logarithmic dependence on d , sometimes even this may be too large (for example when we want to compete against an uncountable set of benchmarks). In such cases, we would ideally like a regret bound that depends on the action and loss sets but not directly on d .

6. There is a slight technical difference between the sets $\mathcal{X} = \overline{\Delta}_d$ and $\mathcal{Y} = [-1, 1]^d$ in Algorithm 1 and the sets $\mathcal{X}' = \Delta_d$ and $\mathcal{Y}' = [0, 1]^d$ in Lemma 2. However, note that if we map $x \in \mathcal{X}$ to $(x_1, x_2, \dots, x_n, 1 - \sum x_i) \in \Delta_{d+1}$ and $y \in [-1, 1]^d$ to $((1 + y_1)/2, (1 + y_2)/2, \dots, (1 + y_d)/2, 0) \in [0, 1]^{d+1}$, we preserve the inner product of x and y up to an additive constant, which disappears when computing regret, and a factor of $1/2$.

In the following subsections, we will demonstrate a framework for regret minimization that allows us to achieve both of these goals (under some fairly light computational assumptions on u).

3.1. Approachability for pseudonorms

In Section 2.3, we described the theory of Blackwell approachability for a distance metric defined by the ℓ_∞ norm (i.e., $\|z\|_\infty = \max_i |z_i|$). We begin here by describing a generalization of this approachability theory to functions we refer to as *pseudonorms* and *pseudodistances*. A function $f: \mathbb{R}^d \rightarrow \mathbb{R}_{\geq 0}$ is a pseudonorm if $f(0) = 0$, f is positive homogeneous (for all $z \in \mathbb{R}^d$, $\alpha \in \mathbb{R}_{\geq 0}$, $f(\alpha z) = \alpha f(z)$), and f satisfies the triangle inequality ($f(z + z') \leq f(z) + f(z')$ for all $z, z' \in \mathbb{R}^d$). Note that unlike norms, pseudonorms may not satisfy definiteness and are not necessarily symmetric; it may be the case that $f(z) \neq f(-z)$. However, all norms are pseudonorms. Note also that by the positive homogeneity and the triangle inequality, a pseudonorm is a convex function. A pseudonorm f defines a *pseudodistance function* d_f via: $\forall z, z' \in \mathbb{R}^d$, $d_f(z, z') = f(z - z')$.

In order to effectively work with pseudodistances, it will be useful to define the dual set \mathcal{T}_f^* associated to f as follows: $\mathcal{T}_f^* = \{\theta: \forall z \in \mathbb{R}^d, \langle \theta, z \rangle \leq f(z)\}$. This coincides with the traditional notion of duality in convex analysis; for example, when f is a norm, \mathcal{T}_f^* coincides with the dual ball of radius one: $\mathcal{T}_f^* = \{\theta: \|\theta\|_* \leq 1\}$ (e.g., when f is the d -dimensional ℓ_∞ norm, \mathcal{T}_f^* is the d -dimensional ℓ_1 -ball). The following theorem relates the pseudodistance between z and a convex set \mathcal{S} to a convex optimization problem over the dual set.

Theorem 5 *For any closed convex set $\mathcal{S} \subset \mathbb{R}^d$, the following equality holds for any $z \in \mathbb{R}^d$:*

$$d_f(z, \mathcal{S}) = \inf_{s \in \mathcal{S}} f(z - s) = \sup_{\theta \in \mathcal{T}_f^*} \left\{ \theta \cdot z - \sup_{s \in \mathcal{S}} \theta \cdot s \right\}.$$

Proof We adopt the standard definition and notation in optimization for an *indicator function* $I_{\mathcal{X}}$ of a set \mathcal{X} : for any x , $I_{\mathcal{X}}(x) = 0$ if x is in \mathcal{X} , $+\infty$ otherwise. Define \tilde{f} by $\tilde{f}(s) = f(z - s)$ for all $s \in \mathbb{R}^d$ and set $g = I_{\mathcal{S}}$.

By definition, the conjugate function of f is defined by: $\forall y \in \mathbb{R}^d$, $f^*(y) = \sup_{x \in \mathbb{R}^d} x \cdot y - f(x)$. Now, if y is in \mathcal{T}_f^* , then we have $f^*(y) \leq f(x) - f(x) = 0$. Thus, since $f(0) = 0$, the supremum in the definition of f^* is achieved for $x = 0$ and $f^*(0) = 0$. Otherwise, if $y \notin \mathcal{T}_f^*$, there exists $x \in \mathbb{R}^d$ such that $x \cdot y > f(x)$. For that x , for any $t > 0$, by the positive homogeneity of f , we have $(tx) \cdot y - f(tx) = t(x \cdot y - f(x)) > 0$. Taking the limit $t \rightarrow +\infty$, this shows that $f^*(y) = +\infty$. Thus, we have $f^*(y) = I_{\mathcal{T}_f^*}$.

By definition, the conjugate function \tilde{f}^* is defined for all θ by $\tilde{f}^*(\theta) = \sup_{x \in \mathbb{R}^d} x \cdot \theta - \tilde{f}(z - x) = \sup_{u \in \mathbb{R}^d} (z - u) \cdot \theta - f(u) = f^*(-\theta) + z \cdot \theta = I_{\mathcal{T}_f^*}(-\theta) + z \cdot \theta$, which can also be derived from the conjugate function calculus in Table B.1 of (Mohri et al., 2018).

It is also known that the conjugate function of the indicator function g is defined by $g^*: \theta \mapsto \sup_{s \in \mathcal{S}} \theta \cdot s$ (Boyd and Vandenberghe, 2014). Since $\text{dom}(\tilde{f}) = \mathbb{R}^d$ and $\text{cont}(g) = \mathcal{S}$, we have $\text{dom}(\tilde{f}) \cap \text{cont}(g) = \mathcal{S} \neq \emptyset$. Thus, for any convex and bounded set \mathcal{S} in \mathbb{R}^d , by Fenchel duality

(Theorem 25, Appendix B), we can write:

$$\begin{aligned}
 d_f(z, \mathcal{S}) &= \inf_{s \in \mathcal{S}} f(z - s) \\
 &= \inf_{s \in \mathbb{R}^d} \{f(z - s) + I_{\mathcal{S}}(s)\} && \text{(def. of } I_{\mathcal{S}}) \\
 &= \sup_{\theta \in \mathbb{R}^d} \left\{ -\left(I_{\mathcal{T}_f^*}(-\theta) + \theta \cdot z \right) - \sup_{s \in \mathcal{S}} \{-\theta \cdot s\} \right\} && \text{(Fenchel duality theorem)} \\
 &= \sup_{-\theta \in \mathcal{T}_f^*} \left\{ -\theta \cdot z - \sup_{s \in \mathcal{S}} \{-\theta \cdot s\} \right\} && \text{(def. of } I_{\mathcal{T}_f^*}) \\
 &= \sup_{\theta \in \mathcal{T}_f^*} \left\{ \theta \cdot z - \sup_{s \in \mathcal{S}} \theta \cdot s \right\}. && \text{(change of variable)}
 \end{aligned}$$

This completes the proof. ■

We will primarily be concerned with the case where \mathcal{S} is a convex cone. A *convex cone* is a set \mathcal{C} such that if $x \in \mathcal{C}$, $\alpha x \in \mathcal{C}$ for all $\alpha \geq 0$. In this case, Theorem 5 can be simplified to write $d_f(z, \mathcal{C})$ as a *linear* optimization problem over the intersection of \mathcal{T}_f^* and the polar cone of \mathcal{C} .

Corollary 6 *Let \mathcal{C} be a convex cone, and let $\mathcal{C}^\circ = \{y \mid \langle y, x \rangle \leq 0 \forall x \in \mathcal{C}\}$ be the polar cone of \mathcal{C} . Fix a d -dimensional pseudonorm f and let $\mathcal{T}_f^{\mathcal{C}} = \mathcal{T}_f^* \cap \mathcal{C}^\circ$. Then for any $z \in \mathbb{R}^d$,*

$$d_f(z, \mathcal{C}) = \sup_{\theta \in \mathcal{T}_f^{\mathcal{C}}} (\theta \cdot z).$$

3.2. From high-dimensional ℓ_∞ approachability to low-dimensional pseudonorm approachability

In Section 2.3, we expressed the regret for a general regret minimization problem as the ℓ_∞ distance

$$\text{Reg} = T \cdot d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d \right). \quad (5)$$

In this section, we will demonstrate how to rewrite this in terms of a lower-dimensional pseudodistance.

Consider the bilinear ‘‘basis map’’ $\tilde{u} : \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}^{nm}$ given by $\tilde{u}(p, \ell)_{i,j} = p_i \ell_j$. Note that every bilinear function $b : \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}$ can be written in the form $b(p, \ell) = \langle \tilde{u}(p, \ell), v \rangle$ for some vector $v \in \mathbb{R}^{nm}$ (i.e., the monomials in \tilde{u} form a basis for the set of biaffine functions on $\mathcal{P} \times \mathcal{L}$)⁷.

Let $v_i \in \mathbb{R}^{nm}$ be the vector of coefficients corresponding to the component u_i of u and consider the function $f(x) = \max(\max_{i \in [d]} \langle v_i, x \rangle, 0)$. Note that f is a pseudonorm on \mathbb{R}^{nm} ; indeed, it’s straightforward to see that f is non-negative and $f(\alpha x) = \alpha f(x)$ for any $\alpha \geq 0$. In addition, let \mathcal{C} be the convex cone defined by $\mathcal{C} = \{x \mid \langle x, v_i \rangle \leq 0 \forall i \in [d]\}$. We claim that that we can rewrite the ℓ_∞ distance in (5) in the following way.

7. If it is helpful, one can think of \tilde{u} as the natural map from $\mathcal{P} \times \mathcal{L}$ to the tensor product $\mathcal{P} \otimes \mathcal{L}$. The vectors v_i can then be thought of as elements of the dual space, each representing a linear functional on $\mathcal{P} \otimes \mathcal{L}$.

Theorem 7 *We have that*

$$d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d \right) = d_f \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), \mathcal{C} \right). \quad (6)$$

The dual set \mathcal{T}_f^* associated to this pseudonorm is given by:

$$\mathcal{T}_f^* = \left\{ \tilde{\theta} \in \mathbb{R}^{nm}: \forall x \in \mathbb{R}^{nm}, \langle \tilde{\theta}, x \rangle \leq \max_{i \in [d]} \langle v_i, x \rangle \right\}.$$

The following two properties of this dual set will be useful in the sections that follow. First, we show that we can alternately think of \mathcal{T}_f^* as the convex hull of the v_i .

Lemma 8 *The dual set \mathcal{T}_f^* coincides with the convex hull of v_i s: $\mathcal{T}_f^* = \text{conv}\{v_1, \dots, v_d\}$.*

Secondly, we show that the dual set \mathcal{T}_f^* is contained within the polar cone \mathcal{C}° of \mathcal{C} .

Lemma 9 *We have that $\mathcal{T}_f^* \subseteq \mathcal{C}^\circ$, where $\mathcal{C}^\circ = \{y \mid \langle y, x \rangle \leq 0 \forall x \in \mathcal{C}\}$.*

Proof By Lemma 8, it suffices to show that each $v_i \in \mathcal{C}^\circ$, i.e., for each v_i , that $\langle v_i, x \rangle \leq 0$ for all $x \in \mathcal{C}$. However, this immediately follows by the definition of \mathcal{C} , since each $x \in \mathcal{C}$ satisfies $\langle x, v_i \rangle \leq 0$ for all $i \in [d]$. (An equivalent way of thinking about this is that \mathcal{C} is already the polar cone of the cone generated by the v_i , so the v_i must lie in the polar to \mathcal{C}). \blacksquare

This allows us to simplify Corollary 6 even further.

Corollary 10 *For this convex cone \mathcal{C} and pseudonorm f , we have that for any $z \in \mathbb{R}^{mn}$,*

$$d_f(z, \mathcal{C}) = \sup_{\tilde{\theta} \in \mathcal{T}_f^*} \langle \tilde{\theta}, z \rangle.$$

Finally, we prove the following ‘‘separability’’ conditions for this approachability problem.

Lemma 11 *The following two statements are true.*

1. *For any $\ell \in \mathcal{L}$, there exists a $p \in \mathcal{P}$ such that $\tilde{u}(p, \ell) \in \mathcal{C}$.*
2. *For any $\tilde{\theta} \in \mathcal{T}_f^*$, there exists a $p \in \mathcal{P}$ such that $\langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$.*

3.3. Algorithms for pseudonorm approachability

We now present an algorithm for pseudonorm approachability in the setting of Section 3.2 (i.e., for the bilinear function \tilde{u} and the convex cone \mathcal{C}). Just as the algorithm in Section 2.3 for ℓ_∞ -approachability required an OLO algorithm for the simplex, this algorithm will assume we have access to an OLO algorithm $\tilde{\mathcal{F}}$ for the sets $\mathcal{X} = \mathcal{T}_f^*$, and $\mathcal{Y} = -\text{conv}(\tilde{u}(\mathcal{P}, \mathcal{L}))$ (recall that $\text{conv}(\tilde{u}(\mathcal{P}, \mathcal{L}))$ denotes the convex hull of the points $\tilde{u}(p, \ell) \in \mathbb{R}^{nm}$ for $p \in \mathcal{P}$ and $\ell \in \mathcal{L}$).

Our algorithm $\tilde{\mathcal{A}}$ is summarized above in Algorithm 2. We now have the following analogue of Theorem 3.

Algorithm 2: Description of Algorithm $\tilde{\mathcal{A}}$ for f -approachability

Initialize $\tilde{\theta}_1$ to an arbitrary point in \mathcal{T}_f^* ;

for $t \leftarrow 1$ **to** T **do**

Choose $p_t \in \mathcal{P}$ so that for all $\ell \in \mathcal{L}$, $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell) \rangle \leq 0$ (such a p_t is guaranteed to exist by the second statement in Lemma 11);

Play action p_t and receive as feedback $\ell_t \in \mathcal{L}$;

Set $y_t = -\tilde{u}(p_t, \ell_t)$;

Set $\tilde{\theta}_{t+1} = \tilde{\mathcal{F}}(y_1, y_2, \dots, y_t)$.

end

Theorem 12 *The following guarantee holds for the pseudonorm approachability algorithm $\tilde{\mathcal{A}}$:*

$$\text{Reg}(\tilde{\mathcal{A}}) = T \cdot d_f \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), \mathcal{C} \right) \leq \text{Reg}(\tilde{\mathcal{F}}).$$

Proof The first equality follows as a direct consequence of Theorem 7 (which proves that the d_f distance to \mathcal{C} is equal to the analogous ℓ_∞ distance to $(-\infty, 0]^d$) and Theorem 3 (which shows that this ℓ_∞ distance is equal to the regret of our regret minimization problem). It therefore suffices to prove the second equality.

Note that

$$\begin{aligned} d_f \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), \mathcal{C} \right) &= \sup_{\tilde{\theta} \in \mathcal{T}_f^*} \left\langle \tilde{\theta}, \frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t) \right\rangle \\ &= - \inf_{\tilde{\theta} \in \mathcal{X}} \left\langle \tilde{\theta}, \frac{1}{T} \sum_{t=1}^T (-\tilde{u}(p_t, \ell_t)) \right\rangle \\ &= \frac{1}{T} \text{Reg}(\tilde{\mathcal{F}}) - \frac{1}{T} \sum_{t=1}^T \langle \tilde{\theta}_t, -\tilde{u}(p_t, \ell_t) \rangle \\ &\leq \frac{1}{T} \text{Reg}(\tilde{\mathcal{F}}). \end{aligned}$$

Here, the first equality holds as a consequence of Corollary 10, and the last inequality holds since (by choice of p_t in step 2a) $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell_t) \rangle \leq 0$ for all $t \in [T]$. \blacksquare

If we choose $\tilde{\mathcal{F}}$ to be FTRL with a quadratic regularizer, Lemma 1 implies the following result.

Theorem 13 *Let $u: \mathcal{P} \rightarrow \mathcal{L}$ be a bilinear regret function. Then there exists a regret minimization algorithm $\tilde{\mathcal{A}}$ for u with regret*

$$\text{Reg}(\tilde{\mathcal{A}}) = O(D_z(\text{diam } \mathcal{P})(\text{diam } \mathcal{L})\sqrt{T}),$$

where $D_z = \max_{i \in [d]} \|v_i\|$. If we let

$$\lambda = \max_{i \in [d]} \|v_i\|_\infty = \max_{i \in [d], j \in [m], k \in [n]} \left| \frac{\partial^2 u_i}{\partial p_j \partial \ell_k} \right|,$$

then this regret bound further satisfies

$$\text{Reg}(\tilde{\mathcal{A}}) = O(\lambda nm \sqrt{T}).$$

Proof To see the first result, note that Lemma 1 directly implies a bound of $O(\text{diam}(\mathcal{X}) \text{diam}(\mathcal{Y}) \sqrt{T})$, where $\mathcal{X} = \mathcal{T}_f^*$ and $\mathcal{Y} = -\text{conv}(\tilde{u}(\mathcal{P}, \mathcal{L}))$. We'll now proceed by simplifying $\text{diam}(\mathcal{X})$ and $\text{diam}(\mathcal{Y})$. First, for \mathcal{X} , recall from Lemma 8 that $\mathcal{X} = \mathcal{T}_f^*$ is the convex hull of the d vectors $v_i \in \mathbb{R}^{nm}$, so $\text{diam}(\mathcal{X}) = O(\max_{i \in [d]} \|v_i\|) = O(D_z)$. Second, for \mathcal{Y} , note that $\text{diam}(\mathcal{Y}) = O(\max_{y \in \mathcal{Y}} \|y\|) = O(\max_{p \in \mathcal{P}, \ell \in \mathcal{L}} \|\tilde{u}(p, \ell)\|)$. But $\|\tilde{u}(p, \ell)\| = \|p\| \cdot \|\ell\|$, so $\text{diam}(\mathcal{Y}) = O(\max_{p \in \mathcal{P}} \|p\| \cdot \max_{\ell \in \mathcal{L}} \|\ell\|) = O((\text{diam } \mathcal{P})(\text{diam } \mathcal{L}))$. Combining these, we obtain the first inequality.

The second result directly follows from the following three facts: i. $\text{diam } \mathcal{P} = O(\sqrt{n})$ (since $\mathcal{P} \subseteq [-1, 1]^n$), ii. $\text{diam } \mathcal{L} = O(\sqrt{m})$ (since $\mathcal{L} \subseteq [-1, 1]^m$), and iii. $D_z \leq \lambda \sqrt{mn}$. ■

Theorem 13 shows that in settings where λ is constant (which is true in all the settings we consider), there exists an algorithm where $\text{Reg}(\tilde{\mathcal{A}}) = \text{poly}(n, m) \sqrt{T}$; notably, $\text{Reg}(\tilde{\mathcal{A}})$ does not depend on the dimension d , which in many cases can be thought of as the number of benchmarks of comparison for our learning algorithm.

In the following two sections, we will show how to strengthen this result in two different ways. First, we will show that (modulo some fairly mild computational assumptions) it is possible to efficiently implement the algorithm of Theorem 13. Second, we will show that by using a different choice of regularizer, we can recover exactly the regret obtained in Corollary 4.

3.4. Efficient algorithms for pseudonorm approachability

3.4.1. COMPUTATIONAL ASSUMPTIONS

In this section we will discuss how to transform the algorithm in Theorem 13 into a computationally efficient algorithm. Note that without any constraints on \mathcal{L} , \mathcal{P} , and u or how they are specified, performing any sort of regret minimization efficiently is a hopeless task (e.g., consider the case where it is computationally hard to even determine membership in \mathcal{P}). We'll therefore make the following three structural / computational assumptions on \mathcal{L} , \mathcal{P} , and u .

First, we will restrict our attention to cases where the loss set \mathcal{L} is “orthant-generating”. A convex subset S of \mathbb{R}^d is *orthant-generating* if $S \subseteq [0, \infty)^d$ and for each $i \in [d]$, there exists an $\lambda_i > 0$ such that $\lambda_i e_i \in S$ (as part of this assumption, we will also assume we have access to the values λ_i). Note that many common choices for \mathcal{L} (e.g., $[0, 1]^m$, Δ_m , intersections of other ℓ_p balls with the positive orthant) are all orthant-generating.

Second, we will assume we have an *efficient separation oracle* for the action set \mathcal{P} ; that is, an oracle which takes a $p \in \mathbb{R}^n$ and outputs (in time $\text{poly}(n)$) either that $p \in \mathcal{P}$ or a separating hyperplane $h \in \mathbb{R}^n$ such that $\langle p, h \rangle > \max_{p' \in \mathcal{P}} \langle p', h \rangle$.

Finally, we will assume we have access to what we call an *efficient regret oracle* for u . Given a collection of R action/loss pairs $(p_r, \ell_r) \in \mathcal{P} \times \mathcal{L}$ and R positive constants $\alpha_r \geq 0$, an efficient regret oracle can compute (in time $\text{poly}(n, m, R)$) the value of $\max_{i \in [d]} \sum_{r=1}^R \alpha_r u_i(p_r, \ell_r)$. This can be thought of as evaluating the function $\text{Reg}(p, \ell)$ for a pair of action and loss sequences that take on the action/loss pair (p_r, ℓ_r) an $\alpha_r / \sum_{r'} \alpha_{r'}$ fraction of the time. At a higher level, having access to an efficient regret oracle means that a learner can efficiently compute their overall regret at the end of T rounds (it is hard to imagine how one could efficiently minimize regret without being able to compute it).

3.4.2. EXTENDING THE DUAL SET

One of the ingredients we will need to implement algorithm $\tilde{\mathcal{A}}$ is a membership oracle for the dual set \mathcal{T}_f^* (e.g. in order to perform OLO over this dual set). To check whether $\tilde{\theta} \in \mathcal{T}_f^*$, it suffices to check whether $\langle \tilde{\theta}, x \rangle \leq \max_{i \in [d]} \langle v_i, x \rangle$ for all $x \in \mathbb{R}^{nm}$, so in turn, it will be useful to be able to compute the function $\max_{i \in [d]} \langle v_i, x \rangle$ for any $x \in \mathbb{R}^{nm}$.

Computing this maximum is *very similar*⁸ to what is provided by our regret oracle: note that by writing $u_i(p_r, \ell_r)$ as $\langle v_i, \tilde{u}(p_r, \ell_r) \rangle$, we can think of the regret oracle providing the value of:

$$\max_{i \in [d]} \left\langle v_i, \sum_{r=1}^R \alpha_r \tilde{u}(p_r, \ell_r) \right\rangle.$$

If it were possible to write any $x \in \mathbb{R}^d$ in the form $\sum_{r=1}^R \alpha_r \tilde{u}(p_r, \ell_r)$, we would be done. However, this may not be possible: in particular, $\sum_{r=1}^R \alpha_r \tilde{u}(p_r, \ell_r)$ must lie in the convex cone $\text{cone}(\tilde{u}(\mathcal{P}, \mathcal{L}))$ generated by all points of the form $\tilde{u}(p, \ell)$.

We will therefore briefly generalize the theory of Section 3.1 to cases where we are only able to optimize over an extension of the dual set. Given a convex cone \mathcal{Z} , let $\mathcal{T}_f^*(\mathcal{Z}) = \{\theta \mid \forall z \in \mathcal{Z}, \langle \theta, z \rangle \leq f(z)\}$. Note that $\mathcal{T}_f^* \subseteq \mathcal{T}_f^*(\mathcal{Z})$ (since in \mathcal{T}_f^* , $\langle \theta, z \rangle \leq f(z)$ must hold for all z in the ambient space). The following lemma shows that if z is in \mathcal{Z} , to maximize a linear function over \mathcal{T}_f^* it suffices to maximize it over $\mathcal{T}_f^*(\mathcal{Z})$.

Lemma 14 *Let \mathcal{Z} be an arbitrary convex cone. Then if $z \in \mathcal{Z}$, the following equalities hold:*

$$\sup_{\theta \in \mathcal{T}_f^*} \langle \theta, z \rangle = \sup_{\theta \in \mathcal{T}_f^*(\mathcal{Z})} \langle \theta, z \rangle = f(z).$$

Now, consider the specific convex cone $\mathcal{Z} = \text{cone}(\tilde{u}(\mathcal{P}, \mathcal{L}))$. Given Lemma 14, it is straightforward to check that Theorem 12 continues to hold even if the domain \mathcal{X} of the OLO algorithm $\tilde{\mathcal{F}}$ is set to $\mathcal{T}_f^*(\mathcal{Z})$ instead of \mathcal{T}_f^* . In particular, the first equality in the proof of Theorem 12 is still true when \mathcal{T}_f^* is replaced by $\mathcal{T}_f^*(\mathcal{Z})$, since $\frac{1}{T} \sum_t \tilde{u}(p_t, \ell_t) \in \mathcal{Z}$.

There is one other issue we must deal with: it is possible that $\mathcal{T}_f^*(\mathcal{Z})$ is significantly larger than \mathcal{T}_f^* , and therefore an OLO algorithm with domain $\mathcal{T}_f^*(\mathcal{Z})$ might incur more regret than that of \mathcal{T}_f^* . In fact, there are cases where the set $\mathcal{T}_f^*(\mathcal{Z})$ is unbounded. Nonetheless, the following lemma shows that for OLO with a quadratic regularizer, we will never encounter very large values of $\tilde{\theta}$.

Lemma 15 *Let $R(\tilde{\theta}) = \|\tilde{\theta}\|^2$, and fix a $z \in \mathcal{Z}$ and $\eta > 0$. Let*

$$\tilde{\theta}_{opt} = \underset{\tilde{\theta} \in \mathcal{T}_f^*(\mathcal{Z})}{\operatorname{argmin}} (\eta R(\tilde{\theta}) - \langle \tilde{\theta}, z \rangle).$$

Then $\|\tilde{\theta}_{opt}\| \leq \operatorname{diam}(\mathcal{T}_f^)$.*

8. In fact, in almost all practical cases where we have an efficient regret oracle, it is possible by a similar computation to directly compute this maximum. Here we describe an approach that works in a blackbox way given only a strict regret oracle – if you accept the existence of an oracle that can optimize linear functions over the v_i , you can skip this subsection.

Lemma 15 therefore implies that a quadratic-regularized OLO algorithm $\tilde{\mathcal{F}}$ will only output values of $\tilde{\theta}$ with $\|\tilde{\theta}\| \leq \text{diam}(\mathcal{T}_f^*)$, and therefore we can safely set $\mathcal{X} = \mathcal{T}_f^* \cap \{\theta \mid \|\theta\| \leq \rho\}$, where $\rho = \Theta(\text{diam}(\mathcal{T}_f^*))$ is an efficiently computable upper bound on $\text{diam}(\mathcal{T}_f^*)$. This guarantees that $\text{diam}(\mathcal{X}) = O(\text{diam}(\mathcal{T}_f^*))$, and thus the regret guarantees of Theorem 13 remain unchanged.

3.4.3. CONSTRUCTING A MEMBERSHIP ORACLE

We will now demonstrate how to use our regret oracle to construct a membership oracle for the expanded set \mathcal{X} defined in the previous section. We first show that it is possible to check for membership in \mathcal{Z} (and further, when $z \in \mathcal{Z}$, write z as a convex combination of the generators of \mathcal{Z}).

Lemma 16 *Given a point $z \in \mathbb{R}^{nm}$, we can efficiently check whether $z \in \mathcal{Z}$. If $z \in \mathcal{Z}$, we can also efficiently write z in the form $\sum_{k=1}^m \alpha_k \tilde{u}(p_k, \ell_k)$ (for an explicit choice of p_k, ℓ_k , and α_k).*

Note that expressing z in the form $\sum_{k=1}^m \alpha_k \tilde{u}(p_k, \ell_k)$ allows us to directly apply our efficient regret oracle (with $R = m$). We therefore gain the following optimization oracle as a corollary.

Corollary 17 *Given an efficient regret oracle, for any $z \in \mathcal{Z}$ we can efficiently (in time $\text{poly}(n, m)$) compute $\max_{i \in [d]} \langle v_i, z \rangle$.*

Finally, we will show that given these two results, we can efficiently construct a membership oracle for our set \mathcal{X} . To do this, we will need the following fact (loosely stated; see Lemma 26 in Appendix B for an accurate statement): it is possible to minimize a convex function over a convex set as long as one has an evaluation oracle for the function and a membership oracle for the convex set.

Lemma 18 *Given an efficient regret oracle for u , we can construct an efficient membership oracle for the set \mathcal{X} .*

3.4.4. IMPLEMENTING REGRET MINIMIZATION

Equipped with this membership oracle, we can now state and prove our main theorem.

Theorem 19 *Assume that:*

1. *The convex set \mathcal{L} is orthant-generating.*
2. *We have an efficient separation oracle for the convex set \mathcal{P} .*
3. *We have an efficient regret oracle for the regret function u .*

Then it is possible to implement the algorithm of Theorem 13 in $\text{poly}(n, m)$ time per round.

Proof There are two steps of unclear computational complexity in the description of the algorithm in Section 3.3: step 2a, where we find a p_t such that $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$, and step 2d, where we have to run our OLO subalgorithm over \mathcal{T}_f^* (which we specifically instantiate as FTRL with a quadratic regularizer).

We begin by describing how to perform step 2a efficiently. Fix a $\tilde{\theta}_t$. Note that since \mathcal{L} is orthant-generating, to check whether $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$, it suffices to check whether

$\tilde{u}(p_t, e_k) \leq 0$ for each unit vector $e_k \in \mathbb{R}^m$. Therefore, to find a valid p_t we must find a point $p_t \in \mathcal{P}$ that satisfies an additional m explicit linear constraints. Since we have an efficient separation oracle for \mathcal{P} , this is possible in $\text{poly}(n, m)$ time.

To implement the FTRL algorithm \mathcal{F} over the set $\mathcal{X} = \mathcal{T}_f^*$ with a quadratic regularizer, each round we must find the minimizer of the convex function $g(\tilde{\theta}) = R(\tilde{\theta}) + \eta \sum_{s=1}^{t-1} \langle \tilde{\theta}, \tilde{u}(p_s, \ell_s) \rangle$ over the convex set \mathcal{X} . To do this, it suffices to exhibit an evaluation oracle for g and a membership oracle for \mathcal{X} . To evaluate g , note that we simply need to be able to compute $R(\tilde{\theta})$ (which is a Euclidean distance in \mathbb{R}^{mn}) and $\sum_{s=1}^{t-1} \langle \tilde{\theta}, \tilde{u}(p_s, \ell_s) \rangle$ which we can do in $O(mn)$ time by keeping track of the cumulative sum $\sum_{s=1}^{t-1} \tilde{u}(p_s, \ell_s)$ and computing a single inner product in \mathbb{R}^{mn} . A membership oracle for \mathcal{X} is provided by Lemma 18. \blacksquare

3.5. Recovering ℓ_∞ regret via maxent regularizers

One interesting aspect of this reduction to pseudonorm approachability is that, in some cases, the regret bound of $O(\sqrt{T \log d})$ achievable via ℓ_∞ -approachability (Corollary 4) sometimes outperforms the regret bound of $O(\lambda mn \sqrt{T})$ achieved by pseudonorm approachability (Theorem 13), for example when $d = \text{poly}(m, n)$. Of course, this comparison is not completely fair: in both cases there is flexibility to specify the underlying OLO algorithm, and the $O(\sqrt{T \log d})$ bound uses an FTRL algorithm with a negative entropy regularizer, whereas our pseudonorm approachability bound uses an FTRL algorithm with a quadratic regularizer. After all, there are well-known cases (e.g. for OLO over a simplex, as in ℓ_∞ -approachability) where the negative entropy regularizer leads to exponentially better (in d) regret bounds than the quadratic regularizer.

In this subsection, we will show that there exists a different regularizer for pseudonorm approachability – one we call a *maxent regularizer* – which recovers the $O(\sqrt{T \log d})$ regret bound of Corollary 4. In doing so, we will also better understand the parallels between the regret minimization algorithm \mathcal{A} (which works via reduction to ℓ_∞ -approachability in a d -dimensional space) and the regret minimization algorithm $\tilde{\mathcal{A}}$ (which works via reduction to pseudonorm approachability in an nm -dimensional space).

Let V be the d -by- nm matrix whose i th row equals v_i . Note that V allows us to translate between analogous concepts/quantities for \mathcal{A} and $\tilde{\mathcal{A}}$ in the following way.

Lemma 20 *The following statements are true:*

- For any $p \in \mathcal{P}$ and $\ell \in \mathcal{L}$, $u(p, \ell) = V \tilde{u}(p, \ell)$.
- The dual set $\mathcal{T}_f^* = V^T \Delta_d$ (i.e., $\tilde{\theta} \in \mathcal{T}_f^*$ iff there exists a $\theta \in \Delta_d$ such that $\tilde{\theta} = V^T \theta$).
- If $\theta \in \Delta_d$ and $\tilde{\theta} = V^T \theta \in \mathcal{T}_f^*$, then for any $p \in \mathcal{P}$ and $\ell \in \mathcal{L}$, $\langle \theta, u(p, \ell) \rangle = \langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle$.
- Fix $\theta_t \in \Delta_d$ and let $\tilde{\theta} = V^T \theta$. If p_t satisfies $\langle \theta_t, u(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$, then p_t also satisfies $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$.

Proof The first statement follows from the fact that $V \tilde{u}(p, \ell)_i = \langle v_i, \tilde{u}(p, \ell) \rangle = u(p, \ell)_i$. The second claim follows as a consequence of Lemma 8 (that \mathcal{T}_f^* is the convex hull of the vectors v_i). The third claim follows from the first claim: $\langle \theta, u(p, \ell) \rangle = \langle \theta, V \tilde{u}(p, \ell) \rangle = \langle V^T \theta, \tilde{u}(p, \ell) \rangle = \langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle$. Finally, the fourth claim follows directly from the third claim. \blacksquare

Now, fix a regret minimization problem (specified by \mathcal{P} , \mathcal{L} , and u) and consider the execution of algorithm \mathcal{A} on some specific loss sequence $\ell = (\ell_1, \ell_2, \dots, \ell_T)$. Each round t , \mathcal{A} runs the entropy-regularized FTRL algorithm \mathcal{F} to generate a $\theta_t \in \Delta_d$ (as a function of the actions and losses up until round t) and then uses θ_t to select a p_t that satisfies $\langle \theta_t, u(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$. If we execute algorithm $\tilde{\mathcal{A}}$ for the same regret minimization problem on the same loss sequence, each round t , $\tilde{\mathcal{A}}$ runs some (to be determined) FTRL algorithm $\tilde{\mathcal{F}}$ to generate a $\tilde{\theta}_t \in \mathcal{J}_f^*$, and then uses $\tilde{\theta}_t$ to select a p_t that satisfies $\langle \tilde{\theta}_t, \tilde{u}(p_t, \ell) \rangle$ for all $\ell \in \mathcal{L}$. Lemma 20 shows that if $\tilde{\theta}_t = V^T \theta_t$ for each t , then both algorithms will generate *exactly the same* sequence of actions in response to this loss sequence⁹, and hence the same regret.

The question then becomes: how do we design an OLO algorithm $\tilde{\mathcal{F}}$ that outputs $\tilde{\theta}_t = V^T \theta_t$ each round that \mathcal{F} would output θ_t ? Recall that if \mathcal{F} is an FTRL algorithm with regularizer $R : \Delta_d \rightarrow \mathbb{R}$, then

$$\theta_t = \operatorname{argmin}_{\theta \in \Delta_d} \left(R(\theta) + \sum_{s=1}^{t-1} \langle \theta, u(p_s, \ell_s) \rangle \right).$$

Define $\tilde{R} : \mathcal{J}_f^* \rightarrow \mathbb{R}$ via

$$\tilde{R}(\tilde{\theta}) = \min_{\substack{\theta \in \Delta_d \\ V^T \theta = \tilde{\theta}}} R(\theta). \quad (7)$$

We claim that if we let $\tilde{\mathcal{F}}$ be the FTRL algorithm that uses regularizer \tilde{R} , then $\tilde{\mathcal{F}}$ will output our desired sequence of $\tilde{\theta}_t$.

Lemma 21 *The following equality holds for the output $\tilde{\theta}_t$:*

$$\tilde{\theta}_t = \operatorname{argmin}_{\tilde{\theta} \in \mathcal{J}_f^*} \left(\tilde{R}(\tilde{\theta}) + \sum_{s=1}^{t-1} \langle \tilde{\theta}, \tilde{u}(p_s, \ell_s) \rangle \right) = V^T \theta_t.$$

Proof Note that we can write

9. Technically, there may be some leeway in terms of which p_t to choose that e.g. satisfies $\langle \theta_t, u(p_t, \ell) \rangle \leq 0$ for all $\ell \in \mathcal{L}$, and the two different procedures could result in different choices of p_t . But if we break ties consistently (e.g. add the additional constraint of choosing the p_t that maximizes the inner product of p_t with some generic vector), then both procedures will produce the same value of p_t .

$$\begin{aligned}
 \tilde{\theta}_t &= \operatorname{argmin}_{\tilde{\theta} \in \mathcal{T}_f^*} \left(\tilde{R}(\tilde{\theta}) + \sum_{s=1}^{t-1} \langle \tilde{\theta}, \tilde{u}(p_t, \ell_t) \rangle \right) \\
 &= \operatorname{argmin}_{\tilde{\theta} \in \mathcal{T}_f^*} \left(\left(\min_{\substack{\theta \in \Delta_d \\ V^T \theta = \tilde{\theta}}} R(\theta) \right) + \sum_{s=1}^{t-1} \langle \tilde{\theta}, \tilde{u}(p_t, \ell_t) \rangle \right) \\
 &= \operatorname{argmin}_{\tilde{\theta} \in \mathcal{T}_f^*} \left(\min_{\substack{\theta \in \Delta_d \\ V^T \theta = \tilde{\theta}}} \left(R(\theta) + \sum_{s=1}^{t-1} \langle \theta, u(p_t, \ell_t) \rangle \right) \right) \\
 &= V^T \operatorname{argmin}_{\theta \in \Delta_d} \left(R(\theta) + \sum_{s=1}^{t-1} \langle \theta, u(p_t, \ell_t) \rangle \right) \\
 &= V^T \theta_t.
 \end{aligned}$$

The third equality follows from the fact that if $\tilde{\theta} = V^T \theta$, then $\langle \theta, u(p, \ell) \rangle = \langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle$ (by Lemma 20). \blacksquare

Corollary 22 *Let \mathcal{F} be an FTRL algorithm over Δ_d with regularizer R , and let $\tilde{\mathcal{F}}$ be an FTRL algorithm over \mathcal{T}_f^* with regularizer \tilde{R} . Then $\operatorname{Reg}(\mathcal{A}) = \operatorname{Reg}(\tilde{\mathcal{A}})$.*

We now consider the specific case where R is the negentropy function; i.e., for $\theta \in \Delta_d$, $R(\theta) = -H(\theta) = \sum_{i=1}^d \theta_i \log \theta_i$. For this choice of R , \tilde{R} becomes

$$\tilde{R}(\tilde{\theta}) = - \max_{\substack{\theta \in \Delta_d \\ V^T \theta = \tilde{\theta}}} H(\theta). \tag{8}$$

In other words, $-\tilde{R}(\tilde{\theta})$ is the maximum entropy of any distribution θ in Δ_d that satisfies the nm linear constraints imposed by $V^T \theta = \tilde{\theta}$. This is exactly an instance of the Maxent problem studied in (Berger et al., 1996; Rosenfeld, 1996; Pietra et al., 1997; Dudík et al., 2007) and (Mohri et al., 2018)[chapter 12].

It is known (Pietra et al., 1997; Dudík et al., 2007; Mohri et al., 2018) that the entropy maximizing distribution is a *Gibbs distribution*. In particular, the θ maximizing the expression in (8) satisfies (for some real constants λ_{jk} for $j \in [n]$, $k \in [m]$)

$$\theta_i = \frac{\exp\left(\sum_{j=1}^n \sum_{k=1}^m \lambda_{jk} v_{ijk}\right)}{Z(\lambda)}, \tag{9}$$

where $Z(\lambda)$ (the ‘‘partition function’’) is defined via

$$Z(\lambda) = \sum_{i=1}^d \exp\left(\sum_{j=1}^n \sum_{k=1}^m \lambda_{jk} v_{ijk}\right). \tag{10}$$

Generally, there is exactly one choice of λ_{jk} which results in $V^T \theta = \tilde{\theta}$ (since there are mn free variables and mn linear constraints). For this optimal λ , it is known that the maximum entropy is given by $\langle \lambda, \tilde{\theta} \rangle - \log Z(\lambda)$. If it is possible to solve this system for λ_{jk} and evaluate $Z(\lambda)$ efficiently,

we can then evaluate \tilde{R} efficiently. In Section 4.1 we will see how to do this for the specific case of swap regret (where we are helped by the fact that (9) guarantees that θ is a product distribution).

Regardless of how we compute the maxent regularizer \tilde{R} , efficient computation leads to an efficient $O(\sqrt{T \log d})$ regret minimization algorithm.

Corollary 23 *If there exists an efficient (poly(n, m) time) algorithm for computing the maxent regularizer $\tilde{R}(\tilde{\theta})$, then there exists a regret minimization algorithm $\tilde{\mathcal{A}}$ for u with regret $O(\sqrt{T \log d})$ and that can be implemented in poly(n, m) time per round.*

4. Applications

4.1. Swap regret

Recall that in the setting of swap regret, the action set is $\mathcal{P} = \Delta_K$ (distributions over K actions), the loss set is $\mathcal{L} = [0, 1]^K$, and the *swap regret* of playing a sequence of actions \mathbf{p} on a sequence of losses ℓ is given by

$$\text{SwapReg}(\mathbf{p}, \ell) = \max_{\pi: [K] \rightarrow [K]} \sum_{t=1}^T \sum_{i=1}^K (p_{ti} \ell_{ti} - p_{ti} \ell_{t\pi(i)}).$$

In words, swap regret compares the total loss achieved by this sequence of actions with the loss achieved by any transformed action sequence formed by applying an arbitrary “swap function” to the actions (i.e., always playing action $\pi(i)$ instead of action i). Swap regret minimization can be directly written as an ℓ_∞ -approachability problem for the bilinear function $u: \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}^{K^K}$ where

$$u_\pi(p, \ell) = \sum_{i=1}^K (p_i \ell_i - p_i \ell_{\pi(i)}).$$

(here we index u by the K^K functions $\pi: [K] \rightarrow [K]$). Note that the negative orthant is indeed separable with respect to u since if we let $p^*(\ell) = \operatorname{argmin}_{p \in \mathcal{P}} \langle p, \ell \rangle$, $u_\pi(p, \ell) \leq 0$ for any swap function π .

We can now apply the theory developed in Section 3. First, since $m = n = K$ and the maximum absolute value of any coefficient in u is 1, Theorem 13 immediately results in a swap regret algorithm with regret $O(K^2 \sqrt{T})$. Moreover, since we can write

$$\max_{\pi: [K] \rightarrow [K]} \sum_{t=1}^T \sum_{i=1}^K (p_{ti} \ell_{ti} - p_{ti} \ell_{t\pi(i)}) = \sum_{i=1}^K \max_{j \in [K]} \sum_{t=1}^T (p_{ti} \ell_{ti} - p_{ti} \ell_{tj}),$$

we can compute SwapReg efficiently. By Theorem 19, we can therefore implement this $O(K^2 \sqrt{T})$ -regret algorithm efficiently (in poly(K) time per round). We can improve upon this regret bound by noting that $\operatorname{diam}(\mathcal{P}) = O(1)$, $\operatorname{diam}(\mathcal{L}) = O(\sqrt{K})$, and $D_z = O(\sqrt{K})$ (the coefficient vector v_π corresponding to $u_\pi(p, \ell)$ contains at most $2K$ coefficients that are ± 1). It follows that $D_z(\operatorname{diam} \mathcal{P})(\operatorname{diam} \mathcal{L}) = O(K)$, and it follows from the first part of Theorem 13 that the regret of the aforementioned algorithm is actually only $O(K \sqrt{T})$.

This is still a factor of approximately $O(\sqrt{K})$ larger than the optimal $O(\sqrt{TK \log K})$ bound. To achieve the optimal regret bound, we will show how to compute the maxent regularizer for swap regret (and hence can efficiently implement an $O(\sqrt{T \log d}) = O(\sqrt{TK \log K})$ algorithm via

Corollary 23). Recall that the maxent regularizer $\tilde{R}(\tilde{\theta})$ is defined for $\tilde{\theta} \in V^T \Delta_d$ is the negative of the maximum entropy of a distribution $\theta \in \Delta_d$ that satisfies $\tilde{\theta} = V^T \theta$. For our problem, $\tilde{\theta}$ is (K^2) -dimensional, and this imposes the following linear constraints on θ (which we view as a distribution over swap functions π):

$$\begin{aligned}\tilde{\theta}_{ij} &= - \sum_{\pi|\pi(i)=j} \theta_\pi && \text{for } j \neq i \\ \tilde{\theta}_{ii} &= 1 - \sum_{\pi|\pi(i)=i} \theta_\pi.\end{aligned}$$

Now, by the characterization presented in (9)¹⁰, we know the entropy maximizing θ satisfies (for some K^2 constants λ_{ij})

$$\theta_\pi = \frac{\exp(\sum_{i=1}^K \lambda_{i\pi(i)})}{\sum_{\pi'} \exp(\sum_{i=1}^K \lambda_{i\pi'(i)})} = \frac{\prod_{i=1}^K \exp(\lambda_{i\pi(i)})}{\sum_{\pi'} \prod_{i=1}^K \exp(\lambda_{i\pi'(i)})} = \prod_{i=1}^K \frac{\exp(\lambda_{i\pi(i)})}{\sum_{j=1}^K \exp(\lambda_{ij})}.$$

In particular, this shows that the entropy maximizing distribution θ is a product distribution over the set of swap functions π where for each $i \in [K]$ the value of $\pi(i)$ is chosen independently. Moreover, from $\tilde{\theta}$ we can recover the overall marginal probability $q_{ij} = \mathbb{P}_{\pi \sim \theta}[\pi(i) = j]$ (it is $-\tilde{\theta}_{ij}$ if $j \neq i$ and $1 - \tilde{\theta}_{ii}$ if $j = i$). The entropy of this product distribution can therefore be written as:

$$H(\theta) = \sum_{i=1}^K H(\pi(i)) = - \sum_{i=1}^K \sum_{j=1}^K q_{ij} \log q_{ij}.$$

Our regularizer $\tilde{R}(\tilde{\theta})$ is simply $-H(\theta)$ and can clearly be efficiently computed as a function of $\tilde{\theta}$. It follows from Corollary 23 that there exists an efficient (poly(K) time per round) regret minimization algorithm for swap regret that incurs $O(\sqrt{T \log d}) = O(\sqrt{TK \log K})$ worst-case regret.

Finally, we briefly remark that the pseudonorm f we construct here is closely related to the $\ell_{1,\infty}$ group norm defined over K -by- K square matrices as the ℓ_1 norm of vector formed by the ℓ_∞ norms of the rows (i.e., $\|M\|_{1,\infty} = \sum_{i=1}^K \max_{j \in [K]} |M_{ij}|$). This is not unique to swap regret; in many of our applications, the relevant pseudonorm can be thought of as a composition of multiple smaller norms (often ℓ_1 or ℓ_∞ norms).

4.2. Procrustean swap regret

To illustrate the power of Theorem 19, we present a toy variant of swap regret where the learner must compete against an *infinite* set of swap functions (in particular, all orthogonal linear transformations of their sequence of actions) and yet can do this efficiently while incurring low (polynomial in the dimension of their action set) regret.

In this problem, the action set $\mathcal{P} = \{x \in \mathbb{R}^n; \|x\|_2 \leq 1\}$ is the set of unit vectors with norm at most 1 and the loss set $\mathcal{L} = [-1, 1]^n$. The learner would like to minimize the following notion of regret:

$$\text{ProcrustesReg}(\mathbf{p}, \ell) = \max_{Q \in O(n)} \sum_{t=1}^T (\langle p_t, \ell_t \rangle - \langle Q p_t, \ell_t \rangle). \quad (11)$$

10. In particular, in the case of swap regret we have that $v_{\pi,i,j} = \mathbf{1}(k = j) - \mathbf{1}(k = \pi(j))$. Since the first term ($\mathbf{1}(k = j)$) does not depend on π , we can ignore its contribution to θ_π (it cancels from the numerator and denominator).

Here, $O(n)$ is the set of all orthogonal n -by- n matrices. We call this notion of regret *Procrustean swap regret* due to its similarity with the *orthogonal Procrustes problem* from linear algebra, which (loosely) asks for the orthogonal matrix which most closely maps one sequence of points x_1 onto another sequence of points x_2 (in our setting, we intuitively want to map x onto $-\ell$ to minimize the loss of our benchmark). See [Gower and Dijkstra \(2004\)](#) for a more detailed discussion of the Procrustes problem. Regardless, note that we can compute $\text{ProcrustesReg}(p, \ell)$ efficiently, since we have an efficient membership oracle for the convex hull $\text{conv}(O(n))$ of the set of orthogonal matrices (specifically, an n -by- n matrix M belongs to $\text{conv}(O(n))$ iff all its singular values are at most 1 in absolute value).

In our approachability framework, we can capture this notion of regret with the bilinear function $u(p, \ell)$ with coordinates indexed by $Q \in O(n)$ with $u(p, \ell)_Q = \langle p, \ell \rangle - \langle Qp, \ell \rangle$ (which is separable with respect to the negative orthant, since for $p = -\ell/\|\ell\|_2$, $u(p, \ell)_Q \leq 0$). Since all the conditions of [Theorem 19](#) hold, there is an efficient learning algorithm which incurs at most $O(\sqrt{nT})$ Procrustean swap regret (in particular, $\text{diam } \mathcal{P} = O(1)$, $\text{diam } \mathcal{L} = O(\sqrt{n})$, and $D_z = O(1)$).

4.3. Converging to Bayesian correlated equilibria

Swap regret has the nice property that if in a repeated n -player normal-form game, all players run a low-swap regret algorithm to select their actions, their time-averaged strategy profile will converge to a correlated equilibrium (indeed, this is one of the major motivations for studying swap regret).

In repeated Bayesian games (games where each player has private information drawn independently from some distribution each round) the analogue of correlated equilibria is *Bayesian correlated equilibria*. Playing a repeated Bayesian game requires a contextual learning algorithm, which can observe the private information of the player (the ‘‘context’’) and select an action based on this. [Mansour et al. \(2022\)](#) show that there is a notion of regret (that we call BayesSwapReg) such that if all learners are playing an algorithm with low BayesSwapReg , then over time they converge on average to a Bayesian correlated equilibrium. However, while [Mansour et al. \(2022\)](#) provide a low BayesSwapReg algorithm, their algorithm is not provably efficient (it requires finding the fixed point of a system of quadratic equations); by applying our framework, we show that it is possible to obtain a polynomial-time, low- BayesSwapReg algorithm for this problem.

Formally, we study the following full-information contextual online learning setting. As before, there are K actions, but there are now C different contexts (‘‘types’’). Every round t , the adversary specifies a loss function $\ell_t \in \mathcal{L} = [0, 1]^{CK}$, where $\ell_{t,i,c}$ represents the loss from playing action i in context c . Simultaneously, the learner specifies an action $p_t \in \mathcal{P} = \Delta([K])^C \subset \mathbb{R}^{CK}$ which we view as a function mapping each context to a distribution over actions. Overall, the learner receives expected utility $\sum_{c=1}^C \mathbb{P}[c] \sum_{i=1}^K p_t(c)_i \ell_{t,i,c}$ this round (the learner’s context is drawn iid from the publicly known distribution \mathcal{C} each round). In this formulation BayesSwapReg can be written in the form

$$\text{BayesSwapReg} = \max_{\kappa, \pi_c} \sum_{t=1}^T \sum_{c=1}^C \mathbb{P}[c] \sum_{i=1}^K (p_t(c)_i \ell_{t,i,c} - p_t(\kappa(c))_i \ell_{t,\pi_c(i),c}), \quad (12)$$

where the maximum is over all ‘‘type swap functions’’ $\kappa: [C] \rightarrow [C]$ and C -tuples of ‘‘action-deviation swap functions’’ $\pi_c: [K] \rightarrow [K]$. It is straightforward to verify that BayesSwapReg (as written in (12)) can be written as an ℓ_∞ -approachability problem for a bilinear function $u: \mathcal{P} \times \mathcal{L} \rightarrow \mathbb{R}^d$ for $d = C^C K^{KC}$. The theory of ℓ_∞ -approachability guarantees the existence of an algorithm

with $O(\sqrt{T \log d}) = O(\sqrt{T(C \log C + KC \log K)})$ regret, but this algorithm has time/space complexity $O(d)$ and is very inefficient for even moderate values of C or K .

Instead, in Appendix C, we show that BayesSwapReg can be written in the form

$$\text{BayesSwapReg} = \sum_{c=1}^C \max_{c' \in [C]} \sum_{i=1}^K \max_{j \in [K]} \sum_{t=1}^T \mathbb{P}[c] (p_t(c) i_{\ell_t, i, c} - p_t(c') i_{\ell_t, j, c}).$$

This allows us to evaluate BayesSwapReg in $\text{poly}(C, K, T)$ time and apply our pseudonorm approachability framework. Directly from Theorems 13 and 19, we know that there exists an efficient ($\text{poly}(C, K)$ time per round) learning algorithm that incurs at most $O(K^2 C^2 \sqrt{T})$ swap regret. As with swap regret, we can tighten this bound somewhat by examining the values of $\text{diam}(\mathcal{X})$ and $\text{diam}(\mathcal{Y})$ and show that this algorithm actually incurs at most $O(KC\sqrt{T})$ regret (details left to Appendix C).

This is within approximately an $O(\sqrt{KC})$ factor of optimal. Interestingly, unlike with swap regret, it is unclear if it is possible to efficiently solve the relevant entropy maximization problem for Bayesian swap regret (and hence achieve the optimal regret bound). We pose this as an open question.

Open Question 1 *Is it possible to efficiently (in $\text{poly}(K, C)$ time) evaluate the maximum entropy regularizer $\tilde{R}(\tilde{\theta})$ for the problem of Bayesian swap regret?*

4.4. Reinforcement learning in constrained MDPs

We consider episodic reinforcement learning in constrained Markov decision processes (MDPs). Here, the agent receives a vectorial reward (loss) in each time step and aims to find a policy that achieves a certain minimum total reward (maximum total loss) in each dimension. Approachability has been used to derive reinforcement learning algorithms for constrained MDPs before (Miryoosefi et al., 2019; Yu et al., 2021; Miryoosefi and Jin, 2021), however, exclusively using ℓ_2 geometry. As a result, these methods aim to bound the ℓ_2 distance to the feasible set and the bounds scale with the ℓ_2 norm of the reward vector. This deviates from the more common, and perhaps more natural, formulation for constrained MDPs studied in other works (Efroni et al., 2020; Brantley et al., 2020; Ding et al., 2021). Here, each component of the loss vector is within a given range (e.g. $[0, 1]$) and the goal is to minimize the largest constraint violation among all components. We will show that ℓ_∞ -approachability is the natural approach for this problem and yields algorithms that avoid the \sqrt{d} factor in the regret, where d is the number of constraints, that algorithms based on ℓ_2 approachability suffer. While the number of constraints d can sometimes be small, there are many applications where d is large and a $\text{poly}(d)$ dependency in the regret is undesirable, even when a computational dependency on d is manageable. For example, constraints may arise from fairness considerations like demographic disparity that ensure that the policy behaves similarly across all protected groups. This would require a constraint for each pair of groups which could be very many.

The formal problem setup is as follows. We consider an MDP $(\mathcal{X}, \mathcal{A}, P, \{\ell\}_{t \in [T]})$ defined by a state space \mathcal{X} , an action set \mathcal{A} , a transition function $P: \mathcal{X} \times \mathcal{A} \times \mathcal{X} \rightarrow [0, 1]$, where $P(x'|x, a)$ is the probability of reaching state x' when choosing action a at state x , and $\ell: \mathcal{X} \times \mathcal{A} \rightarrow [0, 1]^d$ the loss vector function with $\ell = (\ell^1, \dots, \ell^d)$. We work with losses for consistency with the other sections but our results also readily apply to rewards. To simplify the presentation, we will assume a *layered MDP* with $L + 1$ layers with $\mathcal{X} = \bigcup_{l=0}^L \mathcal{X}_l$, $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset$ for $i \neq j$, and with $\mathcal{X}_0 = \{x_0\}$ and $\mathcal{X}_L = \{x_L\}$.

We define a (stochastic) policy π as mapping from $\mathcal{A} \times \mathcal{X} \rightarrow [0, 1]$ where $\pi(a|x)$ represents the probability of action a in state x . Given a policy π and the transition probability P , we define the *occupancy measure* $\mathbf{q}^{P,\pi}$ as the probability of visiting state-action pair (x, a) when following policy P (Altman, 1999; Neu et al., 2012): $\mathbf{q}^{P,\pi}(x, a) = \mathbb{P}[x, a | P, \pi]$. We will denote by $\Delta(P)$ the set of all occupancy measures, obtained by varying the policy π . It is known that $\Delta(P)$ forms a polytope.

We consider the feasibility problem in CMDPs with stochastic rewards and unknown dynamics. The loss vector $\ell: \mathcal{X} \times \mathcal{A} \rightarrow [0, 1]^d$ is the same in all episodes, $\mathcal{L} = \{\ell\}$. Our goal is to learn a policy $\pi \in \Pi$ such that $\langle \mathbf{q}^{P,\pi}, \ell \rangle \leq \mathbf{c}$ for a given threshold vector $\mathbf{c} \in [0, L]^d$. The payoff function $u: \Delta(P) \times \mathcal{L} \rightarrow \mathbb{R}^d$ is defined as: $u(\mathbf{p}, \ell) = \begin{bmatrix} \mathbf{p} \cdot \ell^1 - c_1 \\ \vdots \\ \mathbf{p} \cdot \ell^d - c_d \end{bmatrix}$. The set $\mathcal{S} = (-\infty, 0]^d$ is separable as long as there is a policy that satisfies all constraints. Although we define the payoff function u in terms of occupancy measures $\Delta(P)$, they will be implicit in the algorithm.

To aide the comparison of our approach to existing work in this setting, we omit the dimensionality reduction with pseudonorms in this application and directly work in the d -dimensional space. We will analyze Algorithm 1 for ℓ_∞ approachability, which can be implemented in MDPs by adopting the following oracles from prior work (Miryoosefi et al., 2019) on CMDPs:

- **BESTRESPONSE-Oracle** For a given $\theta \in \mathbb{R}^d$, this oracle returns a policy π that is ϵ_0 -optimal with respect to the reward function $(s, a) \mapsto -\langle \ell(s, a), \theta \rangle$.
- **EST-Oracle** For a given policy π , this oracle returns a vector $z \in [0, L]^d$ such that $\|z - \langle \mathbf{q}^{P,\pi}, \ell \rangle\|_\infty \leq \epsilon_1$.

Consider first the case without approximation errors, $\epsilon_0 = \epsilon_1 = 0$, for illustration. For a vector $\theta_t \in \Delta_{d+1}$, let $\theta'_t \in \mathbb{R}^d$ be the vector that contains only the first d dimensions of θ_t . When we call the BESTRESPONSE oracle with vector θ'_t , it returns a policy π_t such that its occupancy measure satisfies $\langle \mathbf{q}^{P,\pi_t}, -\theta'_t \cdot \ell \rangle \geq \max_{\mathbf{q}^* \in \Delta(P)} \langle \mathbf{q}^*, -\theta'_t \cdot \ell \rangle$. We can use this to show:

$$\theta_t \cdot [u(\mathbf{q}^{P,\pi_t}, \ell), 0] = \theta'_t \cdot u(\mathbf{q}^{P,\pi_t}, \ell) = \langle \mathbf{q}^{P,\pi_t}, \theta'_t \cdot \ell \rangle - c \leq \langle \mathbf{q}^*, \theta'_t \cdot \ell \rangle - c \leq 0$$

where \mathbf{q}^* is the occupancy measure of a policy that satisfies all constraints.

Thus \mathbf{q}^{P,π_t} is a valid choice for p_t in Line 3 of Algorithm 1. Passing the policy π_t to the EST yields $\hat{z}_t = u(\mathbf{q}^{P,\pi_t}, \ell) + c$; this is enough to compute

$$y_t = -[\hat{z}_t - c, 0] = -[u(p_t, \ell), 0]$$

in Line 5 of Algorithm 1¹¹. Finally, we obtain $\theta_{t+1} \in \mathbb{R}^{d+1}$ by passing y_1, y_2, \dots, y_t to a negative entropy FTRL algorithm as the OLO algorithm \mathcal{F} (Line 6 of Alg. 1).

Using similar steps as for Theorem 3 and relying on the regret bound for \mathcal{F} in Lemma 2, we can show the following guarantee:

Proposition 24 *Consider a constrained episodic MDP with horizon L , fixed loss vectors ℓ with $\ell(s, a) \in [0, 1]^d$ for all state-action pairs $(s, a) \in \mathcal{X} \times \mathcal{A}$ and a constraint threshold vector $c \in [0, L]^d$. Assume that there exists a feasible policy that satisfies all constraints and let $\bar{\pi}$ be the mixture policy*

11. Since FTRL with negative entropy regularizer operates on the simplex with $\|\theta_t\| = 1$, we pad the inputs with a zero dimension obtain an OLO algorithm on the interior of Δ_d .

of π_1, \dots, π_T generated by the approach described above. Then the maximum constraint violation of $\bar{\pi}$ satisfies

$$\max_{i \in [d]} \{ \mathbf{q}^{P, \bar{\pi}} \cdot \ell^i - c_i \} = d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(\mathbf{q}^{P, \pi_t}, \ell), \mathcal{S} \right) \leq O \left(L \sqrt{\frac{\log(d)}{T}} \right) + \epsilon_0 + 2\epsilon_1.$$

Applying the results from prior work (Miryoosefi et al., 2019) based on ℓ_2 approachability would yield a bound of $(L\sqrt{d} + \epsilon_1)T^{-1/2} + \epsilon_0 + 2\epsilon_1$ in our setting with an additional \sqrt{d} and ϵ_1 factor in front of $T^{-1/2}$. For the sake of exposition, we illustrated the benefit of ℓ_∞ approachability using the oracles adopted by Miryoosefi et al. (2019), but our approach can also be applied with similar advantages to other works that make oracle calls explicit (Yu et al., 2021; Miryoosefi and Jin, 2021).

5. Conclusion

We presented a new algorithmic framework for ℓ_∞ -approachability, which we argued is the most suitable notion of approachability for a variety of applications such as regret minimization. Our algorithms leverage a key dimensionality reduction and a reduction to an online linear optimization. These ideas can be similarly used to derive useful algorithms for approachability with alternative distance metrics. In fact, as already pointed out, some of our algorithms can be similarly viewed as a reduction to an online linear optimization of an equivalent group-norm approachability.

References

- Jacob D. Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *21st Annual Conference on Learning Theory - COLT 2008, Helsinki, Finland, July 9-12, 2008*, pages 263–274. Omnipress, 2008.
- Jacob D. Abernethy, Peter L. Bartlett, and Elad Hazan. Blackwell approachability and no-regret learning are equivalent. In *Proceedings of COLT*, volume 19 of *JMLR Proceedings*, pages 27–46, 2011.
- Eitan Altman. *Constrained Markov Decision Processes*. Chapman and Hall, 1999.
- Dirk Bergemann and Stephen Morris. Bayes correlated equilibrium and the comparison of information structures in games. *Theoretical Economics*, 11(2):487–522, 2016.
- Adam L. Berger, Stephen Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Comp. Linguistics*, 22(1), 1996.
- Andrey Bernstein and Nahum Shimkin. Response-based approachability with applications to generalized no-regret problems. *J. Mach. Learn. Res.*, 16:747–773, 2015.
- David Blackwell. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1–8, 1956.
- Avrim Blum and Yishay Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8:1307–1324, 2007.

- Jonathan Borwein and Qiji Zhu. *Techniques of Variational Analysis*. Springer, 2005.
- Stephen P. Boyd and Lieven Vandenbergh. *Convex Optimization*. Cambridge University Press, 2014.
- Kianté Brantley, Miro Dudik, Thodoris Lykouris, Sobhan Miryoosefi, Max Simchowitz, Aleksandrs Slivkins, and Wen Sun. Constrained episodic reinforcement learning in concave-convex and knapsack settings. *Proceedings of NIPS*, 33:16315–16326, 2020.
- Evgenii Chzhen, Christophe Giraud, and Gilles Stoltz. A unified approach to fair online learning via blackwell approachability. In *Proceedings of NeurIPS*, pages 18280–18292, 2021.
- A. P. Dawid. The well-calibrated bayesian. *Journal of the American Statistical Association*, 77(379):605–610, 1982. doi: 10.1080/01621459.1982.10477856. URL <https://www.tandfonline.com/doi/abs/10.1080/01621459.1982.10477856>.
- Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo Jovanovic. Provably efficient safe exploration via primal-dual policy optimization. In *Proceedings of AISTATS*, pages 3304–3312. PMLR, 2021.
- Miroslav Dudík, Steven J. Phillips, and Robert E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, 8, 2007.
- Yonathan Efroni, Shie Mannor, and Matteo Pirotta. Exploration-exploitation in constrained MDPs. *arXiv preprint arXiv:2003.02189*, 2020.
- Françoise Forges. Five legitimate definitions of correlated equilibrium in games with incomplete information. *Theory and decision*, 35(3):277–310, 1993.
- Françoise Forges et al. Correlated equilibrium in games with incomplete information revisited. *Theory and decision*, 61(4):329–344, 2006.
- Dean P. Foster and Sergiu Hart. Smooth calibration, leaky forecasts, finite recall, and nash dynamics. *Games and Economic Behavior*, 109:271–293, 2018. ISSN 0899-8256. doi: <https://doi.org/10.1016/j.geb.2017.12.022>. URL <https://www.sciencedirect.com/science/article/pii/S0899825618300113>.
- Dean P Foster and Rakesh V Vohra. Regret in the on-line decision problem. *Games and Economic Behavior*, 29(1-2):7–35, 1999.
- Geoffrey J. Gordon, Amy Greenwald, and Casey Marks. No-regret learning in convex games. In *Proceedings of ICML*, volume 307, pages 360–367. ACM, 2008.
- John C Gower and Garnt B Dijksterhuis. *Procrustes problems*, volume 30. OUP Oxford, 2004.
- Sergiu Hart and Andreu Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150, 2000.
- Sergiu Hart and Andreu Mas-Colell. A general class of adaptive strategies. *Journal of Economic Theory*, 98(5):26–54, 2001.

- Jason Hartline, Vasilis Syrgkanis, and Eva Tardos. No-regret learning in bayesian games. *Advances in Neural Information Processing Systems*, 28, 2015.
- Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.
- Elad Hazan and Satyen Kale. Computational equivalence of fixed points and no regret algorithms, and convergence to equilibria. In *NIPS*, pages 625–632, 2008.
- Dileep M. Kalathil, Vivek S. Borkar, and Rahul Jain. Blackwell’s approachability in Stackelberg stochastic games: A learning version. In *Proceedings of CDC*, pages 4467–4472. IEEE, 2014.
- Jyrki Kivinen and Manfred K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. In *Proceedings of STOC*, pages 209–218. ACM, 1995.
- Joon Kwon. *Mirror descent strategies for regret minimization and approachability*. PhD thesis, Université Pierre et Marie Curie, Paris 6, 2016.
- Joon Kwon. Refined approachability algorithms and application to regret minimization with global costs. *The Journal of Machine Learning Research*, 22:200–1, 2021.
- Joon Kwon and Vianney Perchet. Online learning and blackwell approachability with partial monitoring: Optimal convergence rates. In Aarti Singh and Xiaojin (Jerry) Zhu, editors, *Proceedings of AISTATS*, volume 54, pages 604–613. PMLR, 2017.
- Yin Tat Lee, Aaron Sidford, and Santosh S Vempala. Efficient convex optimization with membership oracles. In *Conference On Learning Theory*, pages 1292–1294. PMLR, 2018.
- Ehud Lehrer. Approachability in infinite dimensional spaces. *Int. J. Game Theory*, 31(2):253–268, 2003.
- Shie Mannor and Nahum Shimkin. The empirical bayes envelope and regret minimization in competitive markov decision processes. *Math. Oper. Res.*, 28(2):327–345, 2003.
- Shie Mannor and Nahum Shimkin. A geometric approach to multi-criterion reinforcement learning. *J. Mach. Learn. Res.*, 5:325–360, 2004.
- Shie Mannor, Vianney Perchet, and Gilles Stoltz. Set-valued approachability and online learning with partial monitoring. *J. Mach. Learn. Res.*, 15(1):3247–3295, 2014a.
- Shie Mannor, Vianney Perchet, and Gilles Stoltz. Approachability in unknown games: Online learning meets multi-objective optimization. In *Proceedings of COLT*, volume 35, pages 339–355, 2014b.
- Yishay Mansour, Mehryar Mohri, Jon Schneider, and Balasubramanian Sivan. Strategizing against learners in Bayesian games, 2022. URL <https://arxiv.org/abs/2205.08562>.
- Sobhan Miryoosefi and Chi Jin. A simple reward-free approach to constrained reinforcement learning. *CoRR*, abs/2107.05216, 2021.

- Sobhan Miryoosefi, Kianté Brantley, Hal Daumé III, Miroslav Dudík, and Robert E. Schapire. Reinforcement learning with convex constraints. In *Proceedings of NIPS*, pages 14070–14079, 2019.
- Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive computation and machine learning. MIT Press, second edition, 2018.
- Gergely Neu, András György, and Csaba Szepesvári. The adversarial stochastic shortest path problem with unknown transition probabilities. In *Proceedings of AISTATS*, volume 22 of *JMLR Proceedings*, pages 805–813, 2012.
- Vianney Perchet. *Approchabilité, Calibration et Regret dans les Jeux à Observations Partielles*. PhD thesis, Université Pierre et Marie Curie - Paris VI, 2010.
- Vianney Perchet. Exponential weight approachability, applications to calibration and regret minimization. *Dynamic Games and Applications*, 5(1):136–153, 2015.
- Vianney Perchet and Marc Quincampoix. On a unified framework for approachability with full or partial monitoring. *Mathematics of Operations Research*, 40(3):596–610, 2015.
- Vianney Perchet and Marc Quincampoix. A differential game on Wasserstein space. Application to weak approachability with partial monitoring. *CoRR*, abs/1811.04575, 2018.
- Stephen Della Pietra, Vincent J. Della Pietra, and John D. Lafferty. Inducing features of random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(4), 1997.
- R. Tyrrell Rockafellar. *Convex analysis*. Princeton Mathematical Series. Princeton University Press, Princeton, NJ, 1970.
- Ronald Rosenfeld. A maximum entropy approach to adaptive statistical language modelling. *Computer Speech & Language*, 10(3):187–228, 1996.
- Shai Shalev-Shwartz. *Online Learning: Theory, Algorithms, and Applications*. PhD thesis, The Hebrew University of Jerusalem, 2007.
- Nahum Shimkin. An online convex optimization approach to blackwell’s approachability. *The Journal of Machine Learning Research*, 17(1):4434–4456, 2016.
- Xavier Spinat. A Necessary and Sufficient Condition for Approachability. *Mathematics of Operations Research*, 27(1):31–44, 2002.
- Nicolas Vieille. Weak approachability. *Math. Oper. Res.*, 17(4):781–791, 1992.
- John von Neumann. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.
- Tiancheng Yu, Yi Tian, Jingzhao Zhang, and Suvrit Sra. Provably efficient algorithms for multi-objective competitive rl. In *International Conference on Machine Learning*, pages 12167–12176. PMLR, 2021.
- Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*, pages 928–936. AAAI Press, 2003.

Contents of Appendix

A	Omitted proofs	30
A.1	Proof of Theorem 3	30
A.2	Proof of Corollary 6	30
A.3	Proof of Theorem 7	30
A.4	Proof of Lemma 8	31
A.5	Proof of Lemma 11	32
A.6	Proof of Lemma 14	32
A.7	Proof of Lemma 15	32
A.8	Proof of Lemma 16	32
A.9	Proof of Lemma 18	33
A.10	Proof of Proposition 24	34
B	General theorems from convex optimization	34
C	Bayesian correlated equilibria	35

Appendix A. Omitted proofs

A.1. Proof of Theorem 3

Proof [Proof of Theorem 3] Similar results appear in e.g. [Kwon \(2021\)](#). For completeness, we include a proof here.

We will need the following fact: for any $x \in \mathbb{R}^d$, $d_\infty(x, (\infty, 0]^d) = \sup_{\theta \in \overline{\Delta}_d} \langle \theta, x \rangle$. This is easy to verify (both sides are equal to $\max(\max_{i \in [d]} x_i, 0)$), but is also a consequence of Fenchel duality (see e.g. [Appendix B](#) and the proof of [Theorem 5](#)).

Armed with this fact, note that

$$\begin{aligned} d_\infty\left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d\right) &= \sup_{\theta \in \overline{\Delta}_d} \left\langle \theta, \frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t) \right\rangle \\ &= -\inf_{\theta \in \mathcal{X}} \left\langle \theta, \frac{1}{T} \sum_{t=1}^T (-u(p_t, \ell_t)) \right\rangle \\ &= \frac{1}{T} \text{Reg}(\mathcal{F}) - \frac{1}{T} \sum_{t=1}^T \langle \theta_t, -u(p_t, \ell_t) \rangle \\ &\leq \frac{1}{T} \text{Reg}(\mathcal{F}). \end{aligned}$$

■

A.2. Proof of Corollary 6

Proof [Proof of Corollary 6] Note that if $\theta \notin \mathcal{C}^\circ$, then there exists a $z \in \mathcal{C}$ such that $\langle \theta, z \rangle > 0$. Therefore, if $\theta \notin \mathcal{C}^\circ$, then the supremum $\sup_{s \in \mathcal{C}} \theta \cdot s = \infty$ (we can take s to be a large multiple of z). On the other hand, if $\theta \in \mathcal{C}^\circ$, then the supremum $\sup_{s \in \mathcal{C}} \theta \cdot s = 0$ (taking $s = 0$). It follows that the supremum $\sup_{\theta \in \mathcal{T}_f^*} \{\theta \cdot z - \sup_{s \in \mathcal{C}} \theta \cdot s\}$ in [Theorem 5](#) must be achieved for a $\theta \in \mathcal{C}^\circ$ (we know that $d_f(z, \mathcal{C})$ is not infinite since $0 \in \mathcal{C}$, so $d_f(z, \mathcal{C}) \leq f(z)$). For such θ , the $\sup_{s \in \mathcal{C}} \theta \cdot s$ term vanishes, and we are left with the statement of this corollary. ■

A.3. Proof of Theorem 7

Proof [Proof of Theorem 7] Note that for any $\rho \geq 0$ we have the following chain of equivalences:

$$\begin{aligned}
 & d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(p_t, \ell_t), (-\infty, 0]^d \right) \leq \rho \\
 \Leftrightarrow & \frac{1}{T} \sum_{t=1}^T u_i(p_t, \ell_t) \leq \rho \quad \forall i \in [d] \\
 \Leftrightarrow & \frac{1}{T} \sum_{t=1}^T \langle \tilde{u}(p_t, \ell_t), v_i \rangle \leq \rho \quad \forall i \in [d] \\
 \Leftrightarrow & \left\langle \frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), v_i \right\rangle \leq \rho \quad \forall i \in [d] \\
 \Leftrightarrow & f \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t) \right) \leq \rho. \tag{13}
 \end{aligned}$$

We will now prove that this final inequality on the norm f (13) implies the desired inequality on d_f :

$$d_f \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), \mathcal{C} \right) \leq \rho. \tag{14}$$

Note that since $0 \in \mathcal{C}$, (14) directly implies 13. To show that (13) implies (14), we must show that

$$f \left(\left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t) \right) - c \right) \leq \rho.$$

for any $c \in \mathcal{C}$. This in turn is equivalent to proving that:

$$\left\langle \left(\frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t) \right) - c, v_i \right\rangle \leq \rho \quad \forall i \in [d].$$

But since $c \in \mathcal{C}$, $\langle c, v_i \rangle \geq 0$ for all $i \in [d]$. Therefore, as long as

$$\left\langle \frac{1}{T} \sum_{t=1}^T \tilde{u}(p_t, \ell_t), v_i \right\rangle \leq \rho \quad \forall i \in [d],$$

(which is true by (13)), (14) will be true as well, as desired. ■

A.4. Proof of Lemma 8

Proof [Proof of Lemma 8] Let \mathcal{H} denote $\text{conv}\{v_1, \dots, v_d\}$. If $\tilde{\theta}$ is in \mathcal{H} , then we can write $\tilde{\theta} = \sum_{i=1}^d \alpha_i v_i$ for some $\alpha_i \geq 0$, $i \in [d]$, with $\sum_{i=1}^d \alpha_i = 1$. Thus, for any $x \in \mathbb{R}^{mn}$, we have

$$\langle \tilde{\theta}, x \rangle = \sum_{i=1}^d \alpha_i \langle v_i, x \rangle \leq \sum_{i=1}^d \alpha_i \max_{j \in [d]} \langle v_j, x \rangle = \max_{j \in [d]} \langle v_j, x \rangle,$$

which implies that $\tilde{\theta}$ is in \mathcal{T}_f^* . Conversely, if $\tilde{\theta}$ is not in \mathcal{H} , since \mathcal{H} is a non-empty closed convex set, $\tilde{\theta}$ can be separated from \mathcal{H} , that is there exists $x \in \mathbb{R}^d$ such that

$$\langle \tilde{\theta}, x \rangle > \sup_{v \in \mathcal{H}} \langle v, x \rangle \geq \max_{i \in [d]} \langle v_i, x \rangle,$$

which implies that $\tilde{\theta}$ is not in \mathcal{T}_f^* . This completes the proof. \blacksquare

A.5. Proof of Lemma 11

Proof [Proof of Lemma 11] We begin with the first statement. Note that since the negative orthant $(-\infty, 0]^d$ is separable with respect to u , for any $\ell \in \mathcal{L}$, there exists a $p \in \mathcal{P}$ such that $u_i(p, \ell) \leq 0$ for all $i \in [d]$. Now, recall that we can write $u_i(p, \ell) = \langle \tilde{u}(p, \ell), v_i \rangle$, so it follows that $\langle \tilde{u}(p, \ell), v_i \rangle \leq 0$ for all $i \in [d]$. This implies $\tilde{u}(p, \ell) \in \mathcal{C}$, as desired.

We next prove the second statement. Note that since $\tilde{\theta} \in \mathcal{T}_f^* \subseteq \mathcal{C}^\circ$, this implies that if $\tilde{u}(p, \ell) \in \mathcal{C}$, then $\langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle \leq 0$. By the first statement, this means that for any $\ell \in \mathcal{L}$, there exists a $p \in \mathcal{P}$ such that $\langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle \leq 0$. By the minimax theorem (since \mathcal{P} and \mathcal{L} are both convex sets and $\langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle$ is a bilinear function of p and ℓ), this implies that there exists a $p \in \mathcal{P}$ such that for all $\ell \in \mathcal{L}$, $\langle \tilde{\theta}, \tilde{u}(p, \ell) \rangle \leq 0$, as desired. \blacksquare

A.6. Proof of Lemma 14

Proof [Proof of Lemma 14] Note that by the definition of $\mathcal{T}_f^*(\mathcal{Z})$, it must be the case that $\sup_{\theta \in \mathcal{T}_f^*(\mathcal{Z})} \langle \theta, z \rangle \leq f(z)$ (this is one of the constraints defining $\mathcal{T}_f^*(\mathcal{Z})$). Since $\mathcal{T}_f^* \subseteq \mathcal{T}_f^*(\mathcal{Z})$, if we show that $\sup_{\theta \in \mathcal{T}_f^*} \langle \theta, z \rangle = f(z)$ this proves the theorem. But this follows directly from Theorem 5 applied to the closed convex set $\mathcal{S} = \{0\}$. \blacksquare

A.7. Proof of Lemma 15

Proof [Proof of Lemma 15] Consider the element

$$\tilde{\theta}' = \operatorname{argmin}_{\tilde{\theta} \in \mathcal{T}_f^*} -\langle \tilde{\theta}, z \rangle$$

which minimizes the unregularized objective over the smaller set \mathcal{T}_f^* . By Lemma 14, we must have that $\langle \tilde{\theta}', z \rangle \geq \langle \tilde{\theta}_{opt}, z \rangle$. But also, by definition of $\tilde{\theta}_{opt}$, we must have that $\eta R(\tilde{\theta}_{opt}) - \langle \tilde{\theta}_{opt}, z \rangle \leq \eta R(\tilde{\theta}') - \langle \tilde{\theta}', z \rangle$. It follows that $R(\tilde{\theta}_{opt}) \leq R(\tilde{\theta}')$, and thus $\|\tilde{\theta}_{opt}\| \leq \|\tilde{\theta}'\| \leq \operatorname{diam}(\mathcal{T}_f^*)$. \blacksquare

A.8. Proof of Lemma 16

Proof [Proof of Lemma 16] Extend the domain of \tilde{u} as a function to $\mathbb{R}^n \times \mathbb{R}^m$, and note that there is a unique way to write $z = \sum_{k=1}^m \tilde{u}(z_k, e_k)$, where for each $k \in [m]$, z_k is an element of \mathbb{R}^n and e_k is the k th unit vector in \mathbb{R}^m . We first claim $z \in \mathcal{Z} = \operatorname{cone}(\tilde{u}(\mathcal{P}, \mathcal{L}))$ iff each $z_k \in \operatorname{cone}(\mathcal{P})$.

To see this, first note that since \mathcal{L} is orthant-generating, there exist a sequence of λ_k such that $\lambda_k e_k \in \mathcal{L}$ for each $k \in [m]$. Now, if each $z_k \in \operatorname{cone}(\mathcal{P})$, then $\tilde{u}(z_k, e_k) \in \operatorname{cone}(\tilde{u}(\mathcal{P}, \mathcal{L}))$ (since

$e_k \in \text{cone}(\mathcal{L})$), so $z \in \mathcal{Z}$. Conversely, if $z \in \mathcal{Z}$, then we can write $z = \sum \alpha_r^R \tilde{u}(p_r, \ell_r)$ for some $R > 0$ and $\alpha_r \geq 0$. Expanding each ℓ_r in the $\{\lambda_k e_k\}$ basis, we find that each z_k must be a positive linear combination of the values p_r and therefore $z_k \in \text{cone}(\mathcal{P})$.

Therefore, to check whether $z \in \mathcal{Z}$, it suffices to check whether each component z_k of z belongs to $\text{cone}(\mathcal{P})$. This is possible to do efficiently given an efficient separation oracle for \mathcal{P} (we can write the convex program $\beta z_k \in \mathcal{P}$ for $\beta > 0$). Finally, if each $z_k \in \text{cone}(\mathcal{P})$ we can also recover a value β_k for each k such that $\beta_k z_k \in \mathcal{P}$ (via the same convex program). This allows us to explicitly write $z = \sum_{k=1}^m \alpha_k \tilde{u}(p_k, \ell_k)$ with $p_k = \beta_k z_k$, $\ell_k = \lambda_k e_k$, and $\alpha_k = 1/(\beta_k \lambda_k)$. ■

A.9. Proof of Lemma 18

Proof [Proof of Lemma 18] Checking for membership in the ball of radius ρ is straightforward, so it suffices to exhibit a membership oracle for the set $\mathcal{T}_f^*(\mathcal{Z})$. Fix a $\tilde{\theta}$ and consider the convex function $h(z) = \max_{i \in [d]} \langle v_i, z \rangle - \langle \tilde{\theta}, z \rangle$. Note that by the definition of $\tilde{\theta} \in \mathcal{T}_f^*(\mathcal{Z})$ iff $\min_{z \in \mathcal{Z}} h(z) \leq 0$, so it suffices to compute the minimum of $h(z)$ over the convex set \mathcal{Z} .

As mentioned, to do this it suffices to exhibit a membership oracle for \mathcal{Z} and an evaluation oracle for $h(z)$. But now, Lemma 16 provides a membership oracle for \mathcal{Z} and Corollary 17 allows us to efficiently evaluate $h(z)$ for $z \in \mathcal{Z}$. ■

A.10. Proof of Proposition 24

Proof [Proof of Proposition 24] We define $y_t = -[u(p_t, \ell), 0]$ the noiseless and $\tilde{y}_t = -[\hat{z}_t - c, 0]$ the actual loss passed to the OLO algorithm. We bound

$$\begin{aligned}
 \max_{i \in [k]} \{q^{P, \tilde{\pi}} \cdot \ell - c\} &= d_\infty \left(\frac{1}{T} \sum_{t=1}^T u(q^{P, \pi_t}, \ell), \mathcal{S} \right) \\
 &= \max_{\theta \in \Delta_{d+1}} \theta \cdot \begin{pmatrix} \frac{1}{T} \sum_{t=1}^T u(q^{P, \pi_t}, \ell) \\ 0 \end{pmatrix} \\
 &= \max_{\theta \in \Delta_{d+1}} \frac{1}{T} \sum_{t=1}^T \theta \cdot \begin{pmatrix} u(q^{P, \pi_t}, \ell) \\ 0 \end{pmatrix} \\
 &\leq \max_{\theta \in \Delta_{d+1}} \left\{ -\frac{1}{T} \sum_{t=1}^T \langle \theta, \tilde{y}_t \rangle + \epsilon_1 \right\} && \text{(def. } \tilde{y}_t \text{ and EST oracle)} \\
 &= -\min_{\theta \in \Delta_{d+1}} \left\{ \frac{1}{T} \sum_{t=1}^T \langle \theta, \tilde{y}_t \rangle \right\} + \epsilon_1 \\
 &\leq \frac{\text{Reg}(\mathcal{F})}{T} - \frac{1}{T} \sum_{t=1}^T \langle \theta_t, \tilde{y}_t \rangle + \epsilon_1 \\
 &\leq \frac{\text{Reg}(\mathcal{F})}{T} - \frac{1}{T} \sum_{t=1}^T \langle \theta_t, y_t \rangle + \epsilon_1 && \text{(EST oracle)} \\
 &\leq \frac{\text{Reg}(\mathcal{F})}{T} + \frac{1}{T} \sum_{t=1}^T \left\langle \theta_t, \begin{pmatrix} u(q^{P, \pi_t}, \ell) \\ 0 \end{pmatrix} \right\rangle + 2\epsilon_1 && \text{(definition } y_t) \\
 &\leq \frac{\text{Reg}(\mathcal{F})}{T} + \frac{1}{T} \sum_{t=1}^T \left\langle \theta_t \cdot \begin{pmatrix} u(q^{P, \pi_t}, \ell) \\ 0 \end{pmatrix} \right\rangle + 2\epsilon_1 && \text{(definition } y_t) \\
 &\leq \frac{\text{Reg}(\mathcal{F})}{T} + \epsilon_0 + 2\epsilon_1 && \text{(BESTRESPONSE oracle)} \\
 &\leq 2L \sqrt{\frac{\log(d+1)}{T}} + \epsilon_0 + 2\epsilon_1 && \text{(Lemma 2)}
 \end{aligned}$$

■

Appendix B. General theorems from convex optimization

We will use the following Fenchel duality theorem (Borwein and Zhu, 2005; Rockafellar, 1970), see also (Mohri et al., 2018).

Theorem 25 (Fenchel duality) *Let \mathcal{X} and \mathcal{Y} be Banach spaces, $f: \mathcal{X} \rightarrow \mathbb{R} \cup \{+\infty\}$ and $g: \mathcal{Y} \rightarrow \mathbb{R} \cup \{+\infty\}$ convex functions and $A: \mathcal{X} \rightarrow \mathcal{Y}$ a bounded linear map. Assume that f , g and A satisfy one of the following conditions:*

- f and g are lower semi-continuous and $0 \in \text{core}(\text{dom}(g) - A \text{dom}(f))$;
- $A \text{dom}(f) \cap \text{cont}(g) \neq \emptyset$;

then $p = d$ for the dual optimization problems

$$p = \inf_{x \in \mathcal{X}} \{f(x) + g(Ax)\}$$

$$d = \sup_{x^* \in \mathcal{Y}^*} \{-f^*(A^*x^*) - g^*(-x^*)\}$$

and the supremum in the second problem is attained if finite.

When constructing efficient algorithms, we will need a efficient method for minimizing a convex function over a convex set given only a membership oracle to the set and an evaluation oracle to the function. This is provided by the following lemma.

Lemma 26 *Let K be a bounded convex subset of \mathbb{R}^d and $f : K \rightarrow [0, 1]$ a convex function over K . Then given access to a membership oracle for K (along with an interior point $x_0 \in K$ satisfying $B(x_0, r) \subseteq K \subseteq B(x_0, R)$ for some given radii $r, R > 0$) and an evaluation oracle for f , there exists an algorithm which computes the minimum value over f (to within precision ε) using $\text{poly}(d, \log(R/r), \log(1/\varepsilon))$ time and queries to these oracles.*

Proof See [Lee et al. \(2018\)](#). ■

Appendix C. Bayesian correlated equilibria

Here we provide another application of the ℓ_∞ approachability framework, to the problem of constructing learning algorithms that converge to correlated equilibria in Bayesian games. Correlated equilibria in Bayesian games (alternatively, “games with incomplete information”) are well-studied throughout the economics and game theory literature; see e.g. ([Forges, 1993](#); [Forges et al., 2006](#); [Bergemann and Morris, 2016](#)). Unlike ordinary correlated equilibria, which are also well-studied from a learning perspective, relatively little is known about algorithms that converge to correlated equilibria in Bayesian games. [Hartline et al. \(2015\)](#) study no-regret learning in Bayesian games, showing that no-regret algorithms converge to a Bayesian coarse correlated equilibrium. More recently, [Mansour et al. \(2022\)](#) introduce a notion of Bayesian swap regret with the property that learners with sublinear Bayesian swap regret converge to correlated equilibria in Bayesian games. [Mansour et al. \(2022\)](#) construct a learning algorithm that achieves low Bayesian swap regret, albeit not a provably efficient one. In this section, we apply our approachability framework to develop the first efficient low-regret algorithm for Bayesian swap regret.

We begin with some preliminaries about standard (non-Bayesian) normal-form games. In a normal form game G with N players, each player n must choose a mixed action $\mathbf{x}_n \in \Delta([K])$ (for simplicity we will assume each player has the same number of pure actions). We call the collection of mixed strategies $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ played by all players a *strategy profile*. We will let $U_i(\mathbf{x})$ denote the utility of player n under strategy profile \mathbf{x} (and insist that U_i is linear in each player’s strategy).

Given a function $\pi : [K] \rightarrow [K]$ and a mixed action $\mathbf{x}_n \in \Delta([K])$, let $\pi(\mathbf{x}_n)$ be the mixed action in $\Delta([K])$ formed by sampling an action from \mathbf{x}_n and then applying the function π (i.e., $\pi(\mathbf{x}_n)_i = \sum_{i' | \pi(i')=i} \mathbf{x}_{n,i'}$). A *correlated equilibrium* of G is a distribution \mathcal{D} over strategy profiles \mathbf{x} such that for any player n and function $\pi : [K] \rightarrow [K]$, it is the case that

$$\mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [U_n(\mathbf{x}')] \leq \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [U_n(\mathbf{x})],$$

where $\mathbf{x}' = (\mathbf{x}_1, \mathbf{x}_2, \dots, \pi(\mathbf{x}_n), \dots, \mathbf{x}_N)$. Similarly, an ε -correlated equilibrium of G is a distribution \mathcal{D} with the property that

$$\mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [U_n(\mathbf{x}')] \leq \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [U_n(\mathbf{x})] + \varepsilon,$$

for any $n \in [N]$ and $\pi: [K] \rightarrow [K]$. Correlated equilibria have the following natural interpretation: a mediator samples a strategy profile \mathbf{x} from \mathcal{D} and tells to each player n a pure action a_n randomly sampled from \mathbf{x}_n . If each player is incentivized to play the action a_n they are told, then \mathcal{D} is a correlated equilibrium.

We define Bayesian games similarly to standard games, with the modification that now each player n also has some private information $c_n \in [C]$ drawn from some public distribution \mathcal{C}_n . We call the vector of realized types $\mathbf{c} = (c_1, c_2, \dots, c_N)$ the *type profile* of the players (drawn randomly from $\mathcal{C} = \mathcal{C}_1 \times \dots \times \mathcal{C}_N$), and now let the utility $U_n(\mathbf{x}, \mathbf{c})$ of player n depend on both the strategy profile and type profile of the players. Note that we can alternately think of the strategy of player n as a function $\mathbf{x}_n: [C] \rightarrow \Delta([K])$ mapping contexts to mixed actions; in this case we can again treat the expected utility for player n (with expectation taken over the random type profile) as a multilinear function $U_n(\mathbf{x})$ function of the strategy profile $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$.

As with regular correlated equilibria, we can motivate the definition of Bayesian correlated equilibria via the introduction of a mediator. In the Bayesian case, all players begin by revealing their private types to the mediator, and the mediator observes a type profile \mathbf{c} . The mediator then samples an joint action profile \mathbf{x} from a distribution $\mathcal{D}(\mathbf{c})$ that *depends* on the observed type profile. Finally, for each player n , the mediator samples a pure action $a_n \sim \mathbf{x}_n$ from the mixed strategy for n , and relays a_n to player n (which they should follow). In order for this to be a valid correlated equilibrium, the following incentive compatibility constraints must be met:

- Players must have no incentive to deviate from the strategy \mathbf{x}_n relayed to them. As in correlated equilibria, this includes deviations of the form “if I am told to play action i , I will instead play action j ”.
- Players also must have no incentive to misreport their type (thus affecting the distribution $\mathcal{D}(\mathbf{c})$ over joint strategy profiles).
- Moreover, no combination of the above two deviations should result in improved utility for a player.

Formally, we define a *Bayesian correlated equilibrium* for a Bayesian game as follows. The distributions $\mathcal{D}(\mathbf{c})$ form a Bayesian correlated equilibrium if, for any player $n \in [N]$, any “type deviation” $\kappa: [C] \rightarrow [C]$, and any collection of “action deviations” $\pi_c: [K] \rightarrow [K]$ (for each $c \in [C]$),

$$\mathbb{E}_{\mathbf{x} \sim \mathcal{D}(\mathbf{c}), \mathbf{c} \sim \mathcal{C}} [U_n(\mathbf{x}')] \leq \mathbb{E}_{\mathbf{x} \sim \mathcal{D}(\mathbf{c}), \mathbf{c} \sim \mathcal{C}} [U_n(\mathbf{x})]$$

where \mathbf{x}' is derived from \mathbf{x} by deviating from \mathbf{x}_n in the following way: when the player n has type c , they first report type $c' = \kappa(c)$ to the mediator; if the mediator then tells them to play action a , they

instead play $\pi_c(a)$. No such deviation should improve the utility of an agent in a Bayesian correlated equilibrium. Likewise, an ε -Bayesian correlated equilibrium is a collection of distributions $\mathcal{D}(\mathbf{c})$ where no deviation increases the utility of a player by more than ε .

In order to play a Bayesian game, a learning algorithm must be contextual (using the agent's private information to decide what action to play). We study the following setting of full-information contextual online learning. As before, there are K actions, but there are now C different contexts (/types). Every round t , the adversary specifies a loss function $\ell_t \in \mathcal{L} = [0, 1]^{CK}$, where $\ell_{t,i,c}$ represents the loss from playing action i in context c . Simultaneously, the learner specifies an action $p_t: [C] \rightarrow \Delta([K])$ mapping each context to a distribution over actions. Overall, the learner receives expected utility $\sum_{c=1}^C \mathbb{P}_{\mathcal{C}}[c] \sum_{i=1}^K p_t(c)_i \ell_{t,i,c}$ this round (here \mathcal{C} is a distribution over contexts; we assume that the learner's context is drawn i.i.d. from \mathcal{C} each round, and that this distribution \mathcal{C} is publicly known).

Motivated by the deviations considered in Bayesian correlated equilibria, we can define the following notion of swap regret in Bayesian games ("Bayesian swap regret"):

$$\text{BayesSwapReg} = \max_{\kappa, \pi_c} \sum_{t=1}^T \sum_{c=1}^C \mathbb{P}[c] \sum_{i=1}^K (p_t(c)_i \ell_{t,i,c} - p_t(\kappa(c))_i \ell_{t,\pi_c(i),c}). \quad (15)$$

Lemma 27 *Let \mathcal{A} be an algorithm with $\text{BayesSwapReg}(\mathcal{A}) = o(T)$. Assume each player i in a repeated Bayesian game G (over T rounds) runs a copy of algorithm \mathcal{A} , and let $p_t = (p_t^{(1)}, p_t^{(2)}, \dots, p_t^{(N)})$ be the strategy profile at time t . Then the time-averaged strategy profile $\mathcal{D}(\mathbf{c}): [C]^N \rightarrow \Delta([K]^N)$, defined by sampling a t uniformly at random from $[T]$ and returning p_t is an ε -Bayesian correlated equilibrium with $\varepsilon = o(1)$.*

Proof We will show that there exists no deviation for player n which increases their utility by more than $\text{BayesSwapReg}(\mathcal{A})/T = o(1)$.

Fix a type deviation $\kappa: [C] \rightarrow [C]$ and set of action deviations $\pi_c: [K] \rightarrow [K]$. This collection of deviations transforms an arbitrary strategy $p \in \Delta([K])^C$ into the strategy p' satisfying $p'(c)_i = \sum_{i'|\pi_c(i')=i} p(\kappa(c))_{i'}$.

For each $t \in [T]$, with probability $1/T$ the mediator will return the strategy profiles $p_t = (p_t^{(1)}, p_t^{(2)}, \dots, p_t^{(N)})$. Now, since U_n is multilinear in the strategies of each player, there exists some vector ℓ_t such that the utility of player n if they defect to a some strategy p' is given by the inner product $\langle p, \ell_t \rangle$.

In particular, conditioned on the mediator returning p_t , the difference in utility for player n between playing $p^{(n)}$ and the strategy p' formed by applying the above deviations to $p^{(n)}$ is exactly

$$\sum_{c=1}^C \mathbb{P}_{\mathcal{C}}[c] \sum_{i=1}^K (p_t^{(n)}(c)_i \ell_{t,i,c} - p_t^{(n)}(\kappa(c))_i \ell_{t,\pi_c(i),c}).$$

Taking expectations over all t , we have that the expected difference in utility by deviating is

$$\frac{1}{T} \sum_{t=1}^T \sum_{c=1}^C \mathbb{P}_{\mathcal{C}}[c] \sum_{i=1}^K (p_t^{(n)}(c)_i \ell_{t,i,c} - p_t^{(n)}(\kappa(c))_i \ell_{t,\pi_c(i),c}).$$

But since player n selected their strategies by playing \mathcal{A} , this is at most $\text{BayesSwapReg}(\mathcal{A})/T = o(1)$, as desired. \blacksquare

It is possible to phrase (15) in the language of ℓ_∞ -approachability by considering the $C^C \cdot K^{KC}$ dimensional vectorial payoff $u(p, \ell)$ given by:

$$u(p, \ell)_{\kappa, \pi_c} = \sum_{c=1}^C \mathbb{P}[c] \sum_{i=1}^K (p(c)_i \ell_{i,c} - p(\kappa(c))_i \ell_{\pi_c(i),c}).$$

A straightforward computation shows that the negative orthant is separable with respect to u .

Lemma 28 *The set $\mathcal{S} = (-\infty, 0]^d$ is separable with respect to the vectorial payoff u .*

Proof Fix an $\ell \in [0, 1]^{CK}$. Then note that if we let $p(c) = e_i$, where $i = \operatorname{argmin}_{j \in [K]} \ell_{t,j,c}$ (i.e., $p(c)$ is entirely supported on the best fixed action to play in context c), it follows that $p(c)_i \ell_{t,i,c} \leq p(c')_i \ell_{t,j,c}$ for all $c' \in [C]$ and $j \in [K]$, and therefore that $u(p, \ell) \in \mathcal{S}$. \blacksquare

This in turn leads (via Theorem 3) to a low-regret (albeit computationally inefficient) algorithm for Bayesian swap regret. Instead, as in the case of swap regret, we will apply our pseudonorm approachability framework. First, we will show that we can rewrite (15) in such a way that allows us to easily evaluate BayesSwapReg.

Lemma 29 *We have that*

$$\text{BayesSwapReg} = \sum_{c=1}^C \max_{c' \in [C]} \sum_{i=1}^K \max_{j \in [K]} \sum_{t=1}^T \mathbb{P}[c] (p_t(c)_i \ell_{t,i,c} - p_t(c')_i \ell_{t,j,c}). \quad (16)$$

Proof From (15), we have that

$$\begin{aligned} \text{BayesSwapReg} &= \max_{\kappa, \pi_c} \sum_{t=1}^T \sum_{c=1}^C \mathbb{P}[c] \sum_{i=1}^K (p_t(c)_i \ell_{t,i,c} - p_t(\kappa(c))_i \ell_{\pi_c(i),c}) \\ &= \max_{\kappa, \pi_c} \sum_{c=1}^C \sum_{i=1}^K \sum_{t=1}^T \mathbb{P}[c] (p_t(c)_i \ell_{t,i,c} - p_t(\kappa(c))_i \ell_{\pi_c(i),c}) \\ &= \max_{\kappa: [C] \rightarrow [C]} \sum_{c=1}^C \max_{\pi_c: [K] \rightarrow [K]} \sum_{i=1}^K \sum_{t=1}^T \mathbb{P}[c] (p_t(c)_i \ell_{t,i,c} - p_t(\kappa(c))_i \ell_{\pi_c(i),c}) \\ &= \sum_{c=1}^C \max_{c' \in [C]} \sum_{i=1}^K \max_{j \in [K]} \sum_{t=1}^T \mathbb{P}[c] (p_t(c)_i \ell_{t,i,c} - p_t(c')_i \ell_{t,j,c}). \end{aligned}$$

Note that (16) allows us to efficiently (in $\text{poly}(K, C, T)$ time) evaluate BayesSwapReg. As mentioned in the main text, directly from Theorems 13 and 19 this gives us an efficient ($\text{poly}(C, K)$ time per round) learning algorithm that incurs at most $O(K^2 C^2 \sqrt{T})$ swap regret. We will now examine the values of $\text{diam}(\mathcal{P})$, $\text{diam}(\mathcal{L})$ and D_z and show that this algorithm actually incurs at most $O(KC\sqrt{T})$ regret.

First, note that since $\mathcal{P} = \Delta([K])^C$, elements $p \in \mathcal{P}$ can be thought of as C K -tuples of positive numbers that add to 1. Each such K -tuple has squared distance at most 1, so $\text{diam}(\mathcal{P}) \leq \sqrt{C}$. Second, since $\mathcal{L} = [0, 1]^{KC}$, $\text{diam}(\mathcal{L}) = \sqrt{KC}$. Finally, the KC coefficients of each z_{κ, π_c} consist of $2K$ copies of the distribution $\mathbb{P}[c]$; this has ℓ_2 norm at most $\sqrt{2K}$, so $D_z = O(\sqrt{2K})$. Combining these three quantities according to Theorem 13, we obtain the following corollary.

Corollary 30 *There exists an efficient contextual learning algorithm with $\text{BayesSwapReg} = O(CK\sqrt{T})$.*