Lecture 8
Predictive Blackwell approachability

Instructor: Gabriele Farina∗

In Lecture 4 we constructed a regret minimizer, called Regret Matching, by solving a suitable Blackwell approachability game. In this lecture, we will do the opposite: we will investigate how regret minimization algorithm can give rise to Blackwell approachability algorithms. From there, we use predictive regret minimization algorithms to arrive at predictive Blackwell approachability algorithms.

1 Using regret minimization to solve Blackwell approachability games

Recall that a Blackwell approachability game is a tuple \((X, \mathcal{Y}, u, S)\), where \(X, \mathcal{Y}\) are closed convex sets, \(u : X \times \mathcal{Y} \rightarrow \mathbb{R}^d\) is a biaffine function, and \(S \subseteq \mathbb{R}^d\) is a closed and convex target set. A Blackwell approachability game represents a vector-valued repeated game between two players. At each time \(t\), the two players interact in this order:

- first, Player 1 selects an action \(x^t \in X\);
- then, Player 2 selects an action \(y^t \in \mathcal{Y}\), which can depend adversarially on all the \(x^t\) output so far;
- finally, Player 1 incurs the vector-valued payoff \(u(x^t, y^t) \in \mathbb{R}^d\), where \(u\) is a biaffine function.

Player 1’s objective is to guarantee that the average payoff converges to the target set \(S\). Formally, given target set \(S \subseteq \mathbb{R}^d\), Player 1’s goal is to pick actions \(x^1, x^2, \ldots \in X\) such that no matter the actions \(y^1, y^2, \ldots \in \mathcal{Y}\) played by Player 2,

\[
\min_{\hat{s} \in S} \left\| \hat{s} - \frac{1}{T} \sum_{t=1}^{T} u(x^t, y^t) \right\|_2 \to 0 \quad \text{as} \quad T \to \infty.
\]

As we discussed in Lecture 4, Blackwell’s theorem states that goal (1) can be attained if and only if any halfspace \(H \supseteq S\) is forceable, where forceability is recalled next.

\begin{definition}[Forceable halfspace] Let \((X, \mathcal{Y}, u, S)\) be a Blackwell approachability game and let \(H \subseteq \mathbb{R}^d\) be a halfspace, that is, a set of the form \(H = \{x \in \mathbb{R}^d : \mathbf{a}^\top x \leq b\}\) for some \(\mathbf{a} \in \mathbb{R}^d, b \in \mathbb{R}\). The halfspace \(H\) is said to be forceable if there exists a strategy of Player 1 that guarantees that the payoff is in \(H\) no matter the actions played by Player 2, that is, if there exists \(x^* \in X\) such that

\[u(x^*, y) \in H \quad \forall y \in \mathcal{Y}.\]

When that is the case, we call action \(x^*\) a forcing action for \(H\).
\end{definition}

Abernethy et al. [2011] showed that it is always possible to convert a regret minimizer into an algorithm for a Blackwell approachability game (that is, an algorithm that chooses actions \(x^t\) at all times \(t\) in such a way that goal (1) holds no matter the actions \(y^1, y^2, \ldots\) played by the opponent). (Gordon’s Lagrangian Hedging [Gordon, 2005, 2006] partially overlaps with the construction by Abernethy et al. [2011].)

∗Computer Science Department, Carnegie Mellon University. gfarina@cs.cmu.edu.
1.1 A couple preliminaries on convex cones

For simplicity, we will only be interested in Blackwell games whose target sets are (nonempty) closed convex cones $S \subseteq \mathbb{R}^n$.

**Definition 1.2.** A cone is a set such that for any point $s \in S$, the rescaled point $\lambda s$ belongs to $S$ for any $\lambda \in \mathbb{R}_{\geq 0}$. In particular, $0 \in S$ for any nonempty cone.

Cones have a very regular geometry that will make constructing approachability algorithms simpler. This simplicity actually doesn’t come at a generality cost: one of the contributions of Abernethy et al. [2011] is to show that any Blackwell approachability game with non-conic target set can be studied and solved by first transforming the problem into a slightly larger Blackwell approachability game with conic target set.

A standard concept in conic geometry is that of the polar cone, which we now define.

**Definition 1.3.** The polar of cone $S$, denotes $S^\circ$, is defined as the set of all vectors that form an obtuse angle with the cone $S$, that is,

$$S^\circ := \{ w \in \mathbb{R}^n : w^\top s \leq 0 \ \forall s \in S \}.$$

The polar $S^\circ$ is itself a closed and convex cone provided that $S$ is a closed and convex cone.

The reason we care about the polar of $S$ is that it gives a characterization of important halfspaces $H \supseteq S$ (see also Figure 1).

**Lemma 1.1.** Let $\theta \in S^\circ$ and consider the halfspace $H_\theta := \{ x \in \mathbb{R}^n : \theta^\top x \leq 0 \}$. Then, $H_\theta \supseteq S$.

**Proof.** Take any $s' \in S$; we will show that $s' \in H_\theta$. Since $\theta \in S^\circ$, by definition of polar cone we have that $\theta^\top s \leq 0$ for all $s \in S$, including in particular $s = s'$. So, $s' \in H_\theta$ as we wanted to show. □

1.2 Abernethy et al. [2011]’s idea

Blackwell’s algorithm described in Lecture 4 worked by playing, at every time $t$, a forcing actions for the halfspace tangent to $S$ at the projection point $\psi_t \in S$ of the current average payoff $\hat{\phi}_t := \frac{1}{T} \sum_{r=1}^{t-1} u(x^r, y^r)$. Abernethy et al. [2011]’s idea is to generalize this construction by letting a regret minimizer decide which halfspace to force.

Specifically, let $R_S$ be a regret minimizer that outputs strategies $\theta_t \in S^\circ$ that observes as utilities the Blackwell payoffs $\ell_t := u(x^t, y^t)$. At every time $t$, we will force the halfspace

$$H_{\theta_t} := \{ x \in \mathbb{R}^n : (\theta_t)^\top x \leq 0 \},$$

which, as we discussed in Lemma 1.1, is a superset of the target set $S$ (see also Figure 1).

The proof of correctness for Algorithm 1 relies on this lemma that shows that the problem of minimizing distance to a cone is equivalent to the problem of maximizing the inner product on the polar of the cone.

**Lemma 1.2.** Let $S \subseteq \mathbb{R}^n$ be a cone and $z$ be a generic point in $\mathbb{R}^n$. Then,

$$\min_{s \in S} \| s - z \|_2 = \max_{\theta \in S^\circ \cap \mathbb{B}_2^n} z^\top \theta,$$

where $\mathbb{B}_2^n := \{ x \in \mathbb{R}^n : \| x \|_2 \leq 1 \}$ denotes the unit ball in $\mathbb{R}^n$ with respect to the Euclidean norm.
Algorithm 1: From regret minimization to Blackwell approachability

Data: \( R \) regret minimizer for \( S \)

1. function NextStrategy()
   2. \( \theta' \leftarrow R.S.\text{NextStrategy}() \)
   3. return \( x' \) forcing action for \( H_{\theta'} := \{ x : (\theta')^\top x \leq 0 \} \)

4. function ReceivePayoff(\( u(x', y') \))
   5. \( R.S.\text{ObserveLoss}(\ell_t := u(x', y')) \)

Figure 1: Pictorial depiction of Algorithm 1’s inner working: at all times \( t \), the algorithm plays a forcing action for the halfspace \( H_{\theta'} \) induced by the last decision output by \( L \).

Proposition 1.1. Denote the regret of \( R.S \) compared to any \( \bar{\theta} \) as

\[
R^T_S(\bar{\theta}) := \sum_{t=1}^{T} (\ell_t^\top \bar{\theta} - \sum_{t=1}^{T} (\ell_t^\top \theta_t).
\]

Then, at all times \( T \), the distance between the average payoff cumulated by Algorithm 1 and the target cone \( S \) is upper bounded as

\[
\min_{\hat{s} \in S} \left\| \hat{s} - \frac{1}{T} \sum_{t=1}^{T} u(x_t, y_t) \right\|_2 \leq \frac{1}{T} \max_{\hat{\theta} \in S^\circ \cap B^n_2} R^T_S(\hat{\theta}),
\]

where \( B^n_2 \) denotes the unit ball in \( \mathbb{R}^n \) with respect to the Euclidean norm, just like in Lemma 1.2.

Proof. Using Lemma 1.2,

\[
\min_{\hat{s} \in S} \left\| \hat{s} - \frac{1}{T} \sum_{t=1}^{T} u(x_t, y_t) \right\|_2 = \max_{\theta \in S^\circ \cap B^n_2} \left( \frac{1}{T} \sum_{t=1}^{T} u(x_t, y_t) \right)^\top \hat{\theta} = \max_{\hat{\theta} \in S^\circ \cap B^n_2} \left( \frac{1}{T} \sum_{t=1}^{T} \ell_t \right)^\top \hat{\theta}
\]

where the second step uses \( \ell_t := u(x_t, y_t) \). By substituting the definition \( R^T_S(\hat{\theta}) \) into (2), we then find

\[
\min_{\hat{s} \in S} \left\| \hat{s} - \frac{1}{T} \sum_{t=1}^{T} u(x_t, y_t) \right\|_2 = \frac{1}{T} \max_{\hat{\theta} \in S^\circ \cap B^n_2} \left\{ R^T_S(\hat{\theta}) + \frac{1}{T} \sum_{t=1}^{T} (\ell_t)^\top \theta_t \right\}
\]

Now, by construction \( x' \) is a forcing action for the halfspace \( H_{\theta'} = \{ x \in \mathbb{R}^n : (\theta')^\top x \leq 0 \} \), and so \((\theta')^\top u(x', y') = (\ell')^\top \theta' \leq 0\). Hence,

\[
\frac{1}{T} \sum_{t=1}^{T} (\ell_t^\top \theta') \leq 0.
\]
Proposition 1.1 immediately implies that if the regret minimizer $R_S$ is able to guarantee that the regret on the subset $S^0 \cap B^2_n$ of its domain $S^0$ grows sublinearly, then goal (1) can be attained.

Algorithms that are able to guarantee that $\max_{\theta \in S^0 \cap B^2_n} R_S^T(\theta) = o(T)$ exist. For example, if $R_S$ is set to OMD or FTRL with Euclidean regularization, then it can be shown that

$$\max_{\theta \in S^0 \cap B^2_n} R_S^T(\theta) \leq \sqrt{2 \left( \sum_{t=1}^{T} \|\ell^t\|^2 \right)},$$

which clearly grows at a sublinear rate of $O(\sqrt{T})$.

## 2 Predictive Blackwell Approachability

**Predictive** Blackwell approachability is a natural extension of Blackwell approachability [Farina et al., 2021]. Similarly to how we defined **predictive** regret minimization, in predictive Blackwell approachability the environment provides Player 1 with a prediction $v^t$ of the next utility $u^t(x^t, y^t)$.

It is immediate to extend the construction of Abernethy et al. [2011] (Algorithm 1) to take into account predictions: since the utility observed by $R_S$ (Line 5) is exactly $u^t(x^t, y^t)$, we can simply use a predictive regret minimization algorithm $R_S$ and provide $v^t$ as the prediction of the next utility. The predictive version of Algorithm 1 is given in Algorithm 2.

The analysis in Proposition 1.1 holds verbatim. In fact, it can be shown that when $R_S$ is set to **predictive** OMD or FTRL with Euclidean regularization, then

$$\max_{\theta \in S^0 \cap B^2_n} R^T_S(\theta) \leq \sqrt{2 \left( \sum_{t=1}^{T} \|\ell^t - v^t\|^2 \right)},$$

which clearly grows at a sublinear rate of $O(\sqrt{T})$ and can be very small if the predictions $v^t$ are accurate.

### Algorithm 2: Predictive Blackwell approachability algorithm

**Data:** $R_S$ **predictive** regret minimizer for $S^0$

1. function NextStrategy($v^t$)
   - $v^t$ is the prediction of the next Blackwell payoff $u(x^t, y^t) \in \mathbb{R}^n$
   - $\theta^t \leftarrow R_S$.NextStrategy($v^t$)
   - return $x^t$ forcing action for $\mathcal{H}_{\theta^t} := \{ x : (\theta^t)\top x \leq 0 \}$

2. function ReceivePayoff($u(x^t, y^t)$)
   - $R_S$.ObserveLoss($\ell^t := u(x^t, y^t)$)

### References

