Lecture 18

Swap Regret and the Minimax Theorem

Lecture 17 proves that the coarse correlated equilibrium concept is tractable in a satisfying sense: there are simple and computationally efficient learning procedures that converge quickly to the set of coarse correlated equilibria in every finite game. What can we say if we zoom in to one of the smaller sets in our hierarchy of equilibrium concepts (Figure 13.1)? Sections 18.1 and 18.2 present a second and more stringent notion of regret, and use it to prove that the correlated equilibrium concept is tractable in a similar sense. Sections 18.3 and 18.4 zoom in further to the mixed Nash equilibrium concept, and prove its tractability in the special case of two-player zero-sum games.

18.1 Swap Regret and Correlated Equilibria

This lecture works with the definition of a correlated equilibrium given in Exercise 13.3, which is equivalent to Definition 13.4.

Definition 18.1 (Correlated Equilibrium) A distribution σ on the set $S_1 \times \cdots \times S_k$ of outcomes of a cost-minimization game is a *correlated equilibrium* if for every agent $i \in \{1, 2, \ldots, k\}$ and swapping function $\delta : S_i \to S_i$,

$$\mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s})] \leq \mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\delta(s_i), \mathbf{s}_{-i})] .$$

Every correlated equilibrium is a coarse correlated equilibrium, and the converse does not generally hold (Section 13.1.6).

Is there an analog of no-regret dynamics (Section 17.4) that converges to the set of correlated equilibria in the sense of Proposition 17.9? For an affirmative answer, the key is to define the appropriate more stringent notion of regret, which compares the cost of an online decision-making algorithm to that of the best swapping function in hindsight. This is a stronger benchmark than the best fixed action in hindsight, since fixed actions correspond to the special case of constant swapping functions.

Recall the model of online decision-making problems introduced in Section 17.1. At each time step $t = 1, 2, \ldots, T$, a decision maker commits to a distribution p^t over her n actions A, then an adversary chooses a cost function $c^t : A \to [-1, 1]$, and finally an action a^t is chosen according to p^t, resulting in cost $c^t(a^t)$ to the decision maker.

Definition 18.2 (Swap Regret) Fix cost vectors $c^1, \ldots, c^T$. The *swap regret* of the action sequence $a^1, \ldots, a^T$ is

$$\frac{1}{T}\left[\sum_{t=1}^{T} c^t(a^t) - \min_{\delta:A\to A} \sum_{t=1}^{T} c^t(\delta(a^t))\right], \tag{18.1}$$

where the minimum ranges over all swapping functions δ.[1]

Definition 18.3 (No-Swap-Regret Algorithm) An online decision-making algorithm $\mathcal{A}$ has *no swap regret* if for every $\epsilon > 0$ there exists a sufficiently large time horizon $T = T(\epsilon)$ such that, for every adversary for $\mathcal{A}$, the expected swap regret is at most ϵ.

As with Definition 17.3, we think of the number n of actions as fixed and the time horizon T tending to infinity, and we allow $\mathcal{A}$ to depend on T.

In every time step t of *no-swap-regret dynamics*, every agent i independently chooses a mixed strategy p_i^t using a no-swap-regret algorithm. Cost vectors are defined as in no-regret dynamics, with $c_i^t(s_i)$ the expected cost of the pure strategy $s_i \in S_i$, given that every other agent j plays the mixed strategy p_j^t. The connection between correlated equilibria and no-swap-regret dynamics is the same as that between coarse correlated equilibria and no-(external-)regret dynamics.

[1] *Internal regret* is a closely related notion, and is defined using the best single swap from one action to another in hindsight, rather than the best swapping function. The swap and internal regret of an action sequence differ by at most a factor of n.

Proposition 18.4 (No-Swap-Regret Dynamics and CE) *Suppose that after T iterations of no-swap-regret dynamics, each agent $i = 1, 2, \dots, k$ of a cost-minimization game has expected swap regret at most ϵ. Let $\sigma^t = \prod_{i=1}^k p_i^t$ denote the outcome distribution at iteration t and $\sigma = \frac{1}{T}\sum_{t=1}^T \sigma^t$ the time-averaged history of these distributions. Then σ is an approximate correlated equilibrium, in the sense that*

$$\mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s})] \le \mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\delta(s_i), \mathbf{s}_{-i})] + \epsilon$$

for every agent i and swapping function $\delta : S_i \to S_i$.

Definitions 18.2–18.3 and Proposition 18.4 are all fine and good, but do any no-swap-regret algorithms exist? The next result is a "black-box reduction" from the problem of designing a no-swap-regret algorithm to that of designing a no-external-regret algorithm.

Theorem 18.5 (Black-Box Reduction) *If there is a no-external-regret algorithm, then there is a no-swap-regret algorithm.*

Combining Theorems 17.6 and 18.5, we conclude that no-swap-regret algorithms exist. For example, plugging the multiplicative weights algorithm (Section 17.2) into this reduction yields a no-swap-regret algorithm that is also computationally efficient. We conclude that correlated equilibria are tractable in the same strong sense as coarse correlated equilibria.

*18.2 Proof of Theorem 18.5

The reduction is very natural, one that you'd hope would work. It requires one clever trick at the end of the proof.

Fix a set $A = \{1, 2, \dots, n\}$ of actions. Let $M_1, \dots, M_n$ denote n different no-(external)-regret algorithms, such as n instantiations of the multiplicative weights algorithm. Each of these algorithms is poised to produce probability distributions over the actions A and receive cost vectors as feedback. Roughly, we can think of algorithm M_j as responsible for protecting against profitable deviations from action j to other actions. Assume for simplicity that, as with the multiplicative weights algorithm, the probability distribution produced by each algorithm M_j at a time step t depends only on the cost vectors $c^1, \ldots, c^{t-1}$ of previous time steps, and not on the realized actions $a^1, \ldots, a^{t-1}$. This assumption lets us restrict attention to oblivious adversaries (Section 17.3.1), or equivalently to cost vector sequences $c^1, \ldots, c^T$ that are fixed a priori.

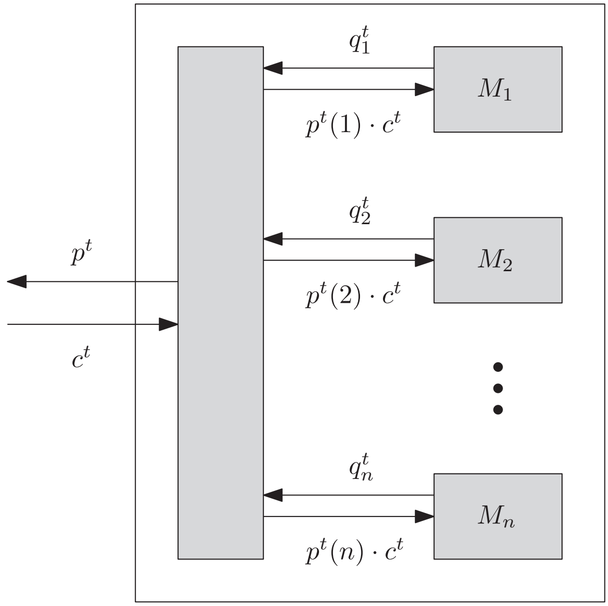

Figure 18.1: Black-box reduction from swap-regret-minimization to external-regret-minimization.

The following "master algorithm" M coordinates $M_1, \ldots, M_n$; see also Figure 18.1.

The Master Algorithm

for each time step $t = 1, 2, \ldots, T$ **do**
  receive distributions $q_1^t, \ldots, q_n^t$ over the actions A from the algorithms $M_1, \ldots, M_n$
  compute and output a consensus distribution p^t
  receive a cost vector c^t from the adversary
  give each algorithm M_j the cost vector $p^t(j) \cdot c^t$

We discuss how to compute the consensus distribution p^t from the distributions $q_1^t, \ldots, q_n^t$ at the end of the proof; this is the clever trick in the reduction. At the end of a time step, the true cost vector c^t is parceled out to the no-regret algorithms, scaled according to the current relevance (i.e., $p^t(j)$) of the algorithm.

We hope to piggyback on the no-external-regret guarantee provided by each algorithm M_j and conclude a no-swap-regret guarantee for the master algorithm M. Let's take stock of what we've got and what we want, parameterized by the consensus distributions $p^1, \ldots, p^T$.

Fix a cost vector sequence $c^1, \ldots, c^T$. The time-averaged expected cost of the master algorithm is

$$\frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n} p^t(i) \cdot c^t(i). \tag{18.2}$$

The time-averaged expected cost under a fixed swapping function $\delta : A \to A$ is

$$\frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n} p^t(i) \cdot c^t(\delta(i)). \tag{18.3}$$

Our goal is to prove that (18.2) is at most (18.3), plus a term that goes to 0 as T tends to infinity, for every swapping function δ.

Adopt the perspective of an algorithm M_j. This algorithm believes that actions are being chosen according to its recommended distributions $q_j^1, \ldots, q_j^T$ and that the true cost vectors are $p^1(j) \cdot c^1, \ldots, p^T(j) \cdot c^T$. Thus, algorithm M_j perceives its time-averaged expected cost as

$$\frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{n} q_j^t(i) \left(p^t(j) c^t(i) \right). \tag{18.4}$$

Since M_j is a no-regret algorithm, its perceived cost (18.4) is, up to a term R_j that tends to 0 as T tends to infinity, at most that of every fixed action $k \in A$. That is, the quantity (18.4) is bounded above by

$$\frac{1}{T} \sum_{t=1}^{T} p^t(j) c^t(k) + R_j. \tag{18.5}$$

Now fix a swapping function δ. Summing the inequality between (18.4) and (18.5) over all $j = 1, 2, \ldots, n$, with k instantiated as $\delta(j)$

in (18.5), proves that

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{n} q_j^t(i)p^t(j)c^t(i) \tag{18.6}$$

is at most

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{n} p^t(j)c^t(\delta(j)) + \sum_{j=1}^{n} R_j. \tag{18.7}$$

The expression (18.7) is equivalent to (18.3), up to a term $\sum_{j=1}^{n} R_j$ that goes to 0 as T goes to infinity. Indeed, we chose the splitting of the cost vector c^t among the no-external-regret algorithms $M_1, \ldots, M_n$ to guarantee this property.

We complete the reduction by showing how to choose the consensus distributions $p^1, \ldots, p^T$ so that (18.2) and (18.6) coincide. For each $t = 1, 2, \ldots, T$, we show how to choose the consensus distribution p^t so that, for each $i \in A$,

$$p^t(i) = \sum_{j=1}^{n} q_j^t(i)p^t(j). \tag{18.8}$$

The left- and right-hand sides of (18.8) are the coefficients of $c^t(i)$ in (18.2) and in (18.6), respectively.

The key trick in the reduction is to recognize the equations (18.8) as those defining the stationary distribution of a Markov chain. Precisely, given distributions $q_1^t, \ldots, q_n^t$ from the algorithms $M_1, \ldots, M_n$ at time step t, form the following Markov chain (Figure 18.2): the set of states is $A = \{1, 2, \ldots, n\}$, and for every $i, j \in A$, the transition probability from j to i is $q_j^t(i)$. That is, the distribution q_j^t specifies the transition probabilities out of state j. A probability distribution p^t satisfies (18.8) if and only if it is a stationary distribution of this Markov chain. At least one such distribution exists, and one can be computed efficiently using an eigenvector computation (see the Notes). This completes the proof of Theorem 18.5.

Remark 18.6 (Interpretation of Consensus Distributions) The choice of the consensus distribution p^t given the no-regret algorithms' suggestions $q_1^t, \ldots, q_n^t$ follows from the proof approach, but it also has a natural interpretation as the limit of an iterative

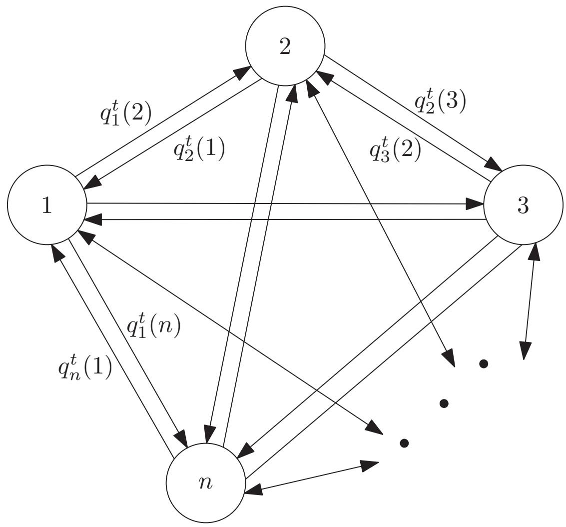

Figure 18.2: Markov chain used to compute consensus distributions.

decision-making process. Consider asking some algorithm M_{j_1} for a recommended strategy. It gives a recommendation j_2 drawn from its distribution $q_{j_1}^t$. Then ask algorithm M_{j_2} for a recommendation j_3, which it draws from its distribution $q_{j_2}^t$, and so on. This random process is effectively trying to converge to a stationary distribution p^t of the Markov chain defined in the proof of Theorem 18.5.

18.3 The Minimax Theorem for Zero-Sum Games

The rest of this lecture restricts attention to games with two agents. As per convention, we call each agent a *player*, and use the payoff-maximization formalism of games (Remark 13.1).

18.3.1 Two-Player Zero-Sum Games

A two-player game is *zero-sum* if, in every outcome, the payoff of each player is the negative of the other. These are games of pure competition, with one player's gain the other player's loss. A two-player zero-sum game can be specified by a single matrix $\mathbf{A}$, with the two strategy sets corresponding to the rows and columns. The entry

a_{ij} specifies the payoff of the row player in the outcome (i, j) and the negative payoff of the column player in this outcome. Thus, the row and column players prefer bigger and smaller numbers, respectively. We can assume that all payoffs lie between -1 and 1, scaling the payoffs if necessary.

For example, the following matrix describes the payoffs in the Rock-Paper-Scissors game (Section 1.3) in our current language.

	Rock	Paper	Scissors
Rock	0	-1	1
Paper	1	0	-1
Scissors	-1	1	0

Pure Nash equilibria (Definition 13.2) generally don't exist in two-player zero-sum games, so the analysis of such games focuses squarely on mixed Nash equilibria (Definition 13.3), with each player randomizing independently according to a mixed strategy. We use $\mathbf{x}$ and $\mathbf{y}$ to denote mixed strategies over the rows and columns, respectively.

When payoffs are given by an $m \times n$ matrix $\mathbf{A}$, the row strategy is $\mathbf{x}$, and the column strategy is $\mathbf{y}$, we can write the expected payoff of the row player as

$$\sum_{i=1}^{m} \sum_{j=1}^{n} x_i \cdot y_j \cdot a_{ij} = \mathbf{x}^\top \mathbf{A} \mathbf{y}.^2$$

The column player's expected payoff is the negative of this. Thus, the mixed Nash equilibria are precisely the pairs $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ such that

$$\hat{\mathbf{x}}^\top \mathbf{A} \hat{\mathbf{y}} \geq \mathbf{x}^\top \mathbf{A} \hat{\mathbf{y}} \qquad \text{for all mixed strategies } \mathbf{x} \text{ over rows}$$

and

$$\hat{\mathbf{x}}^\top \mathbf{A} \hat{\mathbf{y}} \leq \hat{\mathbf{x}}^\top \mathbf{A} \mathbf{y} \qquad \text{for all mixed strategies } \mathbf{y} \text{ over columns.}$$

18.3.2 The Minimax Theorem

In a two-player zero-sum game, would you prefer to commit to a mixed strategy before or after the other player commits to hers? Intuitively, there is only a first-mover disadvantage, since the second

[2]The symbol "$\top$" denotes vector or matrix transpose.

player can adapt to the first player's strategy. The Minimax theorem is the amazing statement that *it doesn't matter*.

Theorem 18.7 (Minimax Theorem) *For every two-player zero-sum game* $\mathbf{A}$*,*

$$\max_{\mathbf{x}} \left(\min_{\mathbf{y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right) = \min_{\mathbf{y}} \left(\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right). \tag{18.9}$$

On the left-hand side of (18.9), the row player moves first and the column player second. The column player plays optimally given the strategy chosen by the row player, and the row player plays optimally anticipating the column player's response. On the right-hand side of (18.9), the roles of the two players are reversed. The Minimax theorem asserts that, under optimal play, the expected payoff of each player is the same in the two scenarios. The quantity (18.9) is called the *value* of the game $\mathbf{A}$.

The Minimax theorem is equivalent to the statement that every two-player zero-sum game has at least one mixed Nash equilibrium (Exercise 18.3). It also implies the following "mix and match" property (Exercise 18.4): if $(\mathbf{x}^1, \mathbf{y}^1)$ and $(\mathbf{x}^2, \mathbf{y}^2)$ are mixed Nash equilibria of the same two-player zero-sum game, then so are $(\mathbf{x}^1, \mathbf{y}^2)$ and $(\mathbf{x}^2, \mathbf{y}^1)$.

*18.4 Proof of Theorem 18.7

In a two-player zero-sum game, it's only worse to go first: if $\hat{\mathbf{x}}$ is an optimal mixed strategy for the row player when she plays first, she always has the option of playing $\hat{\mathbf{x}}$ when she plays second. Thus the left-hand side of (18.9) is at most the right-hand side. We turn our attention to the reverse inequality.

Fix a two-player zero-sum game $\mathbf{A}$ with payoffs in $[-1, 1]$ and a parameter $\epsilon \in (0, 1]$. Suppose we run no-regret dynamics (Section 17.4) for enough iterations T that both players have expected (external) regret at most ϵ. For example, if both players use the multiplicative weights algorithm, then $T = (4 \ln(\max\{m, n\}))/\epsilon^2$ iterations are enough, where m and n are the dimensions of $\mathbf{A}$ (Corollary 17.7).[3]

[3]In Lecture 17, the multiplicative weights algorithm and its guarantee are stated for cost-minimization problems. Viewing payoffs as negative costs, they carry over immediately to the present setting.

Let $\mathbf{p}^1, \ldots, \mathbf{p}^T$ and $\mathbf{q}^1, \ldots, \mathbf{q}^T$ be the mixed strategies played by the row and column players, respectively, as advised by their no-regret algorithms. The payoff vector revealed to each no-regret algorithm after iteration t is the expected payoff of each strategy, given the mixed strategy played by the other player in iteration t. This translates to the payoff vectors $\mathbf{A}\mathbf{q}^t$ and $(\mathbf{p}^t)^\top \mathbf{A}$ for the row and column player, respectively.

Let

$$\hat{\mathbf{x}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{p}^t$$

be the time-averaged mixed strategy of the row player,

$$\hat{\mathbf{y}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{q}^t$$

the time-averaged mixed strategy of the column player, and

$$v = \frac{1}{T} \sum_{t=1}^{T} (\mathbf{p}^t)^\top \mathbf{A}\mathbf{q}^t$$

the time-averaged expected payoff of the row player.

Adopt the row player's perspective. Since her expected regret is at most ϵ, for every vector e_i corresponding to a fixed pure strategy i, we have

$$e_i^\top \mathbf{A}\hat{\mathbf{y}} = \frac{1}{T} \sum_{t=1}^{T} e_i^\top \mathbf{A}\mathbf{q}^t \le \frac{1}{T} \sum_{t=1}^{T} (\mathbf{p}^t)^\top \mathbf{A}\mathbf{q}^t + \epsilon = v + \epsilon. \tag{18.10}$$

Since an arbitrary mixed strategy $\mathbf{x}$ over the rows is just a probability distribution over the e_i's, inequality (18.10) and linearity imply that

$$\mathbf{x}^\top \mathbf{A}\hat{\mathbf{y}} \le v + \epsilon \tag{18.11}$$

for every mixed strategy $\mathbf{x}$.

A symmetric argument from the column player's perspective, using that her expected regret is also at most ϵ, shows that

$$\hat{\mathbf{x}}^\top \mathbf{A}\mathbf{y} \ge v - \epsilon \tag{18.12}$$

for every mixed strategy $\mathbf{y}$ over the columns. Thus

$$
\begin{aligned}
\max_{\mathbf{x}}\left(\min_{\mathbf{y}} \mathbf{x}^\top \mathbf{A}\mathbf{y}\right) &\geq \min_{\mathbf{y}} \hat{\mathbf{x}}^\top \mathbf{A}\mathbf{y} \\
&\geq v - \epsilon \qquad (18.13) \\
&\geq \max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A}\hat{\mathbf{y}} - 2\epsilon \qquad (18.14) \\
&\geq \min_{\mathbf{y}}\left(\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A}\mathbf{y}\right) - 2\epsilon,
\end{aligned}
$$

where (18.13) and (18.14) follow from (18.12) and (18.11), respectively. Taking the limit as $\epsilon \to 0$ (and $T \to \infty$) completes the proof of the Minimax theorem.

The Upshot

☆ The swap regret of an action sequence is the difference between the time-averaged costs of the sequence and of the best swapping function in hindsight.

☆ A no-swap-regret algorithm guarantees expected swap regret tending to 0 as the time horizon tends to infinity.

☆ There is a black-box reduction from the problem of no-swap-regret algorithm design to that of no-(external)-regret algorithm design.

☆ The time-averaged history of joint play in no-swap-regret dynamics converges to the set of correlated equilibria.

☆ A two-player game is zero-sum if, in every outcome, the payoff of each player is the negative of the other.

☆ The Minimax theorem states that, under optimal play in a two-player zero-sum game, the expected payoff of a player is the same whether she commits to a mixed strategy before or after the other player.

Notes

The close connection between no-swap-regret algorithms and correlated equilibria is developed in Foster and Vohra (1997) and Hart and Mas-Colell (2000). Theorem 18.5 is due to Blum and Mansour (2007a). Background on Markov chains is in Karlin and Taylor (1975), for example. The first proof of the Minimax theorem (Theorem 18.7) is due to von Neumann (1928). von Neumann and Morgenstern (1944), inspired by Ville (1938), give a more elementary proof. Dantzig (1951), Gale et al. (1951), and Adler (2013) make explicit the close connection between the Minimax theorem and linear programming duality, following the original suggestion of von Neumann (see Dantzig (1982)). Our proof of the Minimax theorem, using no-regret algorithms, follows Freund and Schapire (1999); a similar result is implicit in Hannan (1957). Cai et al. (2016) investigate generalizations of the Minimax theorem to wider classes of games. Problems 18.2 and Problem 18.3 are from Freund and Schapire (1999) and Gilboa and Zemel (1989), respectively.

Exercises

Exercise 18.1 *(H)* Prove that, for arbitrarily large T, the swap regret of an action sequence of length T can exceed its external regret by at least T.

Exercise 18.2 In the black-box reduction in Theorem 18.5, suppose we take each of the no-regret algorithms $M_1, \ldots, M_n$ to be the multiplicative weights algorithm (Section 17.2), where n denotes the number of actions. What is the swap regret of the resulting master algorithm, as a function of n and T?

Exercise 18.3 Let $\mathbf{A}$ denote the matrix of row player payoffs of a two-player zero-sum game. Prove that a pair $\hat{\mathbf{x}}, \hat{\mathbf{y}}$ of mixed strategies forms a mixed Nash equilibrium of the game if and only if it is a *minimax pair*, meaning

$$\hat{\mathbf{x}} \in \operatorname{argmax}_{\mathbf{x}} \left(\min_{\mathbf{y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right)$$

and

$$\hat{\mathbf{y}} \in \operatorname{argmin}_{\mathbf{y}} \left(\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right).$$

Exercise 18.4 *(H)* Prove that if $(\mathbf{x}_1, \mathbf{y}_1)$ and $(\mathbf{x}_2, \mathbf{y}_2)$ are mixed Nash equilibria of a two-player zero-sum game, then so are $(\mathbf{x}_1, \mathbf{y}_2)$ and $(\mathbf{x}_2, \mathbf{y}_1)$.

Exercise 18.5 A two-player game is *constant-sum* if there is a constant a such that, in every outcome, the sum of the players' payoffs equals a. Does the Minimax theorem (Theorem 18.7) hold in all constant-sum games?

Exercise 18.6 *(H)* Call a game with three players *zero-sum* if, in every outcome, the payoffs of the three players sum to zero. Prove that, in a natural sense, three-player zero-sum games include arbitrary two-player games as a special case.

Problems

Problem 18.1 *(H)* Exhibit a (non-zero-sum) two-player game in which the time-averaged history of joint play generated by no-regret dynamics need not converge to a mixed Nash equilibrium.

Problem 18.2 Fix a two-player zero-sum game $\mathbf{A}$ with payoffs in $[-1, 1]$ and a parameter $\epsilon \in (0, 1]$. Suppose that, at each time step $t = 1, 2 \ldots, T$, the row player moves first and uses the multiplicative weights algorithm to choose a mixed strategy $\mathbf{p}^t$, and the column moves second and chooses a best response $\mathbf{q}^t$ to $\mathbf{p}^t$. Assume that $T \geq (4 \ln m)/\epsilon^2$, where m is the number of rows of $\mathbf{A}$.

(a) Adopt the row player's perspective to prove that the time-averaged expected payoff $\frac{1}{T}\sum_{t=1}^{T} (\mathbf{p}^t)^\top \mathbf{A} \mathbf{q}^t$ of the row player is at least

$$\min_{\mathbf{y}} \left(\max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right) - \epsilon.$$

(b) Adopt the column player's perspective to prove that the time-averaged expected payoff of the row player is at most

$$\max_{\mathbf{x}} \left(\min_{\mathbf{y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \right).$$

(c) Use (a) and (b) to give an alternative proof of Theorem 18.7.

Problem 18.3 This problem and the next assume familiarity with linear programming, and show that all of our computationally tractable equilibrium concepts can be characterized by linear programs.

Consider a cost-minimization game with k agents that each have at most m strategies. We can view a probability distribution over the outcomes O of the game as a point $\mathbf{z} \in \mathbb{R}^O$ for which $z_{\mathbf{s}} \geq 0$ for every outcome $\mathbf{s}$ and $\sum_{\mathbf{s} \in O} z_{\mathbf{s}} = 1$.

(a) *(H)* Exhibit a system of at most km additional inequalities, each linear in $\mathbf{z}$, such that the coarse correlated equilibria of the game are precisely the distributions that satisfy all of the inequalities.

(b) Exhibit a system of at most km^2 additional inequalities, each linear in $\mathbf{z}$, such that the correlated equilibria of the game are precisely the distributions that satisfy all of the inequalities.

Problem 18.4 *(H)* Prove that the mixed Nash equilibria of a two-player zero-sum game can be characterized as the optimal solutions to a pair of linear programs.