

## Lecture 4: 2-Player Zero Sum Games

Lecturer: Yishay Mansour

Scribe: Yair Halevi, Daniel Deutch

## 4.1 2-Player Zero Sum Games

In this lecture we will discuss 2-player zero sum games. Such games are completely competitive, where whatever one player wins, the other must lose. Examples of such games include chess, checkers, backgammon, etc. We will show that in such games:

- An equilibrium always exists;
- All equilibrium points yield the same payoff for all players;
- The set of equilibrium points is actually the cartesian product of independent sets of equilibrium strategies per player.

We will also show applications of this theory.

**Definition** Let  $G$  be the game defined by  $\langle N, (A_i), (u_i) \rangle$  where  $N$  is the number of players,  $A_i$  is the set of possible pure strategies for player  $i$ , and  $u_i$  is the payoff function for player  $i$ . Let  $A$  be the cartesian product  $A = \prod_{i=1}^n A_i$ . Then  $G$  is a *zero sum game* if and only if:

$$\forall \vec{a} \in A, \sum_{i=1}^n u_i(\vec{a}) = 0 \quad (4.1)$$

In other words, a *zero sum game* is a game in which, for any outcome (any combination of pure strategies, one per player), the sum of payoffs for all players is zero.

We naturally extend the definition of  $u_i$  to any probability distribution  $\vec{p}$  over  $A$  by  $u_i(\vec{p}) = E_{\vec{a} \sim \vec{p}}(u_i(\vec{a}))$ . The following is immediate due to the linearity of the expectation and the zero sum constraint:

**Corollary 4.1** Let  $G$  be a zero sum game, and  $\Delta$  the set of probability distributions over  $A$ . Then

$$\forall \vec{p} \in \Delta, \sum_{i=1}^n u_i(\vec{p}) = 0 \quad (4.2)$$

Specifically, this will also hold for any probability distribution that is the product of  $N$  independent distributions, one per player, which applies to our normal mixed strategies game.

A *2-player zero sum game* is a zero sum game with  $N = 2$ . In this case, 4.1 may be written as

$$\forall a_1 \in A_1, a_2 \in A_2, \quad u_1(a_1, a_2) = -u_2(a_1, a_2) \quad (4.3)$$

Such a game is completely competitive. There is no motivation for cooperation between the players.

A two person zero sum game may also be described by a single function  $\pi : A_1 \times A_2 \rightarrow R$  describing the payoff value for player I, or the loss value for player II. The goal of player I is to maximize  $\pi$ , while the goal of player II is to minimize  $\pi$ . We say that  $\pi(i, j)$  is the *value of the game for strategies  $i$  and  $j$*  or simply the *payoff for  $i$  and  $j$* .

Given a certain ordering of the pure strategies of both players, we can also represent a finite 2-player zero sum game using a real matrix  $A_{m \times n}$  (the *payoff matrix*), where  $m$  is the number of pure strategies for player I and  $n$  is the number of pure strategies for player II. The element  $a_{ij}$  in the  $i$ th row and  $j$ th column of  $A$  is the payoff (for player I) assuming player I chooses his  $i$ th strategy and player II chooses his  $j$ th strategy.

## 4.2 Nash Equilibria

The Nash equilibria of a 2-player zero sum game have several interesting properties. First, they all exhibit the same value. Second, they are *interchangeable*, meaning that given 2 Nash equilibrium points, it is possible to replace a strategy for one of the players in the first point by the strategy of the same player in the second point and obtain another Nash equilibrium. Formally:

**Theorem 4.2** *Let  $G$  be a 2-player zero sum game defined by  $\langle (A_1, A_2), \pi \rangle$ . Let  $(\tau_1, \tau_2)$  and  $(\sigma_1, \sigma_2)$  be two Nash equilibria for  $G$ . Then*

1. Both  $(\sigma_1, \tau_2)$  and  $(\tau_1, \sigma_2)$  are Nash equilibria of  $G$ .
2.  $\pi(\tau_1, \tau_2) = \pi(\tau_1, \sigma_2) = \pi(\sigma_1, \tau_2) = \pi(\sigma_1, \sigma_2)$

**Proof:**  $(\sigma_1, \sigma_2)$  is a Nash equilibrium. Therefore, for the first player (who plays to maximize  $\pi$ ), we have

$$\pi(\sigma_1, \sigma_2) \geq \pi(\tau_1, \sigma_2)$$

However,  $(\tau_1, \tau_2)$  is a Nash equilibrium as well. Therefore, for the second player (who plays to minimize  $\pi$ ) we have

$$\pi(\tau_1, \sigma_2) \geq \pi(\tau_1, \tau_2)$$

Combining these two inequalities we get

$$\pi(\sigma_1, \sigma_2) \geq \pi(\tau_1, \sigma_2) \geq \pi(\tau_1, \tau_2)$$

Similarly,

$$\pi(\sigma_1, \sigma_2) \leq \pi(\sigma_1, \tau_2) \leq \pi(\tau_1, \tau_2)$$

From the last two inequalities we obtain

$$\pi(\sigma_1, \sigma_2) = \pi(\tau_1, \tau_2) = \pi(\sigma_1, \tau_2) = \pi(\tau_1, \sigma_2)$$

Which proves part 2 of the theorem. To prove part 1 we observe that because  $(\sigma_1, \sigma_2)$  is a Nash equilibrium for player I,

$$\forall \alpha'_1 \in A_1, \quad \pi(\alpha'_1, \sigma_2) \leq \pi(\sigma_1, \sigma_2) = \pi(\tau_1, \sigma_2)$$

Where the right-hand equation is due to part 2 of the theorem which has already been proven. Similarly, because  $(\tau_1, \tau_2)$  is a Nash equilibrium for player II,

$$\forall \alpha'_2 \in A_2, \quad \pi(\tau_1, \alpha'_2) \geq \pi(\tau_1, \tau_2) = \pi(\tau_1, \sigma_2)$$

Which means that  $(\tau_1, \sigma_2)$  is a Nash equilibrium as well. The proof is similar for  $(\sigma_1, \tau_2)$ .  $\square$

Theorem 4.2 holds with the same proof for both the deterministic and the nondeterministic case.

We define the *equilibrium strategies* of a player as the set of all strategies played by the player in any equilibrium point. For player I, this is given by

$$\{ \sigma_1 \in A_1 \mid \exists \sigma_2 \in A_2, (\sigma_1, \sigma_2) \text{ is an eq. pt. } \}$$

**Corollary 4.3** *The set of Nash equilibrium points of a 2-player zero sum game is the cartesian product of the equilibrium strategies of each player.*

When a 2-player zero sum game is represented as a matrix A, a deterministic Nash equilibrium for the game is a saddle point of A, or a pair of strategies  $i, j$  so that

$$a_{ij} = \max_k a_{kj}$$

$$a_{ij} = \min_l a_{il}$$

Such an equilibrium does not necessarily exist.

### 4.3 Payoff Bounds

For a deterministic game, player I can guarantee a payoff lower bound by choosing a pure strategy for which the minimal payoff is maximized. This assumes player II is able to know player I's choice and will play the worst possible strategy for player I (note that in a 2-player zero sum game this is also player II's best response to player I's chosen strategy).

We denote this "gain-floor" by  $V'_I$ :

$$V'_I = \max_i \min_j a_{ij}$$

Similarly, player II can guarantee a loss upper bound by choosing the pure strategy for which the maximal payoff is minimal. We denote this "loss-ceiling" by  $V'_{II}$ :

$$V'_{II} = \min_j \max_i a_{ij}$$

**Lemma 4.4** For any function  $F : X \times Y \rightarrow R$ , for which all the relevant minima and maxima exist:

1.  $\max_{x \in X} \min_{y \in Y} F(x, y) \leq \min_{y \in Y} \max_{x \in X} F(x, y)$
2. Equality holds iff:

$$\exists x_0 \in X, y_0 \in Y, F(x_0, y_0) = \min_{y \in Y} F(x_0, y) = \max_{x \in X} F(x, y_0)$$

**Proof:** The proof of this lemma is trivial and is not shown here. □

Applying Lemma 4.4 to our case proves the intuitive fact that player I's gain-floor cannot be greater than player II's loss-ceiling,

$$V'_I \leq V'_{II}$$

and that equality holds iff we have a saddle point and thus an equilibrium.

### 4.4 Mixed Strategies

For a finite 2-player zero sum game denoted as a matrix  $A_{m \times n}$ , we denote a mixed strategy for a player I (II) by a stochastic vector of length  $m$  ( $n$ ), where the  $i$ th element in the vector is the probability for the  $i$ th pure strategy of this player (using the same order used to generate the payoff matrix).

Vectors in this text are always row vectors. We will typically use  $x$  for player I mixed strategies, and  $y$  for player II mixed strategies. We shall denote by  $\Delta_d$  the set of stochastic vectors in  $R^d$ .

For a 2-player zero sum game given by matrix  $A_{m \times n}$ , and given mixed strategies  $x$  for player I and  $y$  for player II, the expected payoff is given by

$$A(x, y) = \sum_{i=1}^m \sum_{j=1}^n x_i a_{ij} y_j = xAy^T \quad (4.4)$$

Once again, if player I chose strategy  $x$ , the minimum gain (which is also player II's best response loss) is

$$v_{II}(x) = \min_{y \in \Delta_n} xAy^T \quad (4.5)$$

Assuming player II knows what player I has played before selecting a strategy. The minimum exists because  $\Delta_n$  is compact and  $xAy^T$  is continuous in  $y$ . It is easily shown that this minimum must be reachable in at least one pure strategy of player II.

**Lemma 4.5**

$$\forall x \in \Delta_m, \quad v_{II}(x) = \min_{y \in \Delta_n} xAy^T = \min_{1 \leq j \leq n} xAe_j^T$$

**Proof:** The proof is trivial given the fact that  $xAy^T$  is a stochastic combination of  $xAe_j^T$ , so  $xAy^T$  can never be less than all of  $xAe_j^T$ , and on the other hand,  $e_j$  is also in  $\Delta_n$ , so  $v_{II}(x) \leq xAe_j^T$ . □

Therefore we can write 4.5 as

$$v_{II}(x) = \min_{1 \leq j \leq n} xAe_j^T = \min_{1 \leq j \leq n} \sum_{i=1}^m x_i a_{ij} \quad (4.6)$$

Which means that player I can guarantee the following lower bound on his payoff (gain-floor)

$$V_I = \max_{x \in \Delta_m} \min_{y \in \Delta_n} xAy^T = \max_{x \in \Delta_m} \min_{1 \leq j \leq n} xAe_j^T = \max_{x \in \Delta_m} \min_{1 \leq j \leq n} \sum_{i=1}^m x_i a_{ij} \quad (4.7)$$

Such a mixed strategy  $x$  that maximizes  $v_{II}(x)$  is a *maximin* strategy for player I. Once again, this maximum exists due to compactness and continuity.

We define  $v_I(y)$  in a similar fashion as player I's most harmful response (to player II) to strategy  $y$  of player II (this is also player I's best response to  $y$ ). Then, player II can guarantee the following upper bound on his loss (loss-ceiling)

$$V_{II} = \min_{y \in \Delta_n} \max_{x \in \Delta_m} xAy^T = \min_{y \in \Delta_n} \max_{1 \leq i \leq m} e_i Ay^T = \min_{y \in \Delta_n} \max_{1 \leq i \leq m} \sum_{j=1}^n y_j a_{ij} \quad (4.8)$$

Such a mixed strategy  $y$  that maximizes  $v_I(y)$  is a *minimax* strategy for player II.

$V_I$  and  $V_{II}$  are called the *values* of the game for players I and II, respectively.

## 4.5 The Minimax Theorem

Applying Lemma 4.4 to the maximin and minimax values of the game we obtain

$$V_I \leq V_{II} \quad (4.9)$$

In fact, we will show the following fundamental property of 2-player zero sum games

**Theorem 4.6** (*The Minimax Theorem*)

$$V_I = V_{II}$$

We start by proving two lemmas.

**Lemma 4.7** (*Supporting Hyperplane Theorem*) Let  $B \subseteq \mathbb{R}^d$  be a closed convex set and  $\vec{x} \notin B$  then  $\vec{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_d)$  and  $\alpha_{d+1}$  exist such that

$$\vec{\alpha} \cdot \vec{x} = \sum_{i=1}^d \alpha_i x_i = \alpha_{d+1} \quad (4.10)$$

$$\forall y \in B, \quad \vec{\alpha} \cdot \vec{y} = \sum_{i=1}^d \alpha_i y_i > \alpha_{d+1} \quad (4.11)$$

In other words, given a convex closed set  $B$  and a point outside the set  $\vec{x}$ , the lemma claims that we can pass a hyperplane through  $\vec{x}$  such that  $B$  lies entirely on one side of the hyperplane. This lemma and its proof are schematically shown in figure 4.1.

**Proof:** Let  $\vec{z} \in B$  be the point in  $B$  nearest to  $\vec{x}$ . Such a point exists because  $B$  is closed, and the distance function is both continuous and bounded from below by 0. We define

$$\begin{aligned} \vec{\alpha} &= \vec{z} - \vec{x} \\ \alpha_{d+1} &= \vec{\alpha} \cdot \vec{x} \end{aligned}$$

4.10 holds immediately. We shall prove 4.11. Note that  $\vec{\alpha} \neq 0$  because  $\vec{z} \in B$  and  $\vec{x} \notin B$ . Thus,

$$\vec{\alpha} \cdot \vec{z} - \alpha_{d+1} = \vec{\alpha} \cdot \vec{z} - \vec{\alpha} \cdot \vec{x} = \vec{\alpha} \cdot (\vec{z} - \vec{x}) = \vec{\alpha} \cdot \vec{\alpha} = |\vec{\alpha}|^2 > 0$$

Therefore,

$$\vec{\alpha} \cdot \vec{z} > \alpha_{d+1}$$

Now, assume that there exists  $\vec{y} \in B$  such that

$$\vec{\alpha} \cdot \vec{y} \leq \alpha_{d+1}$$

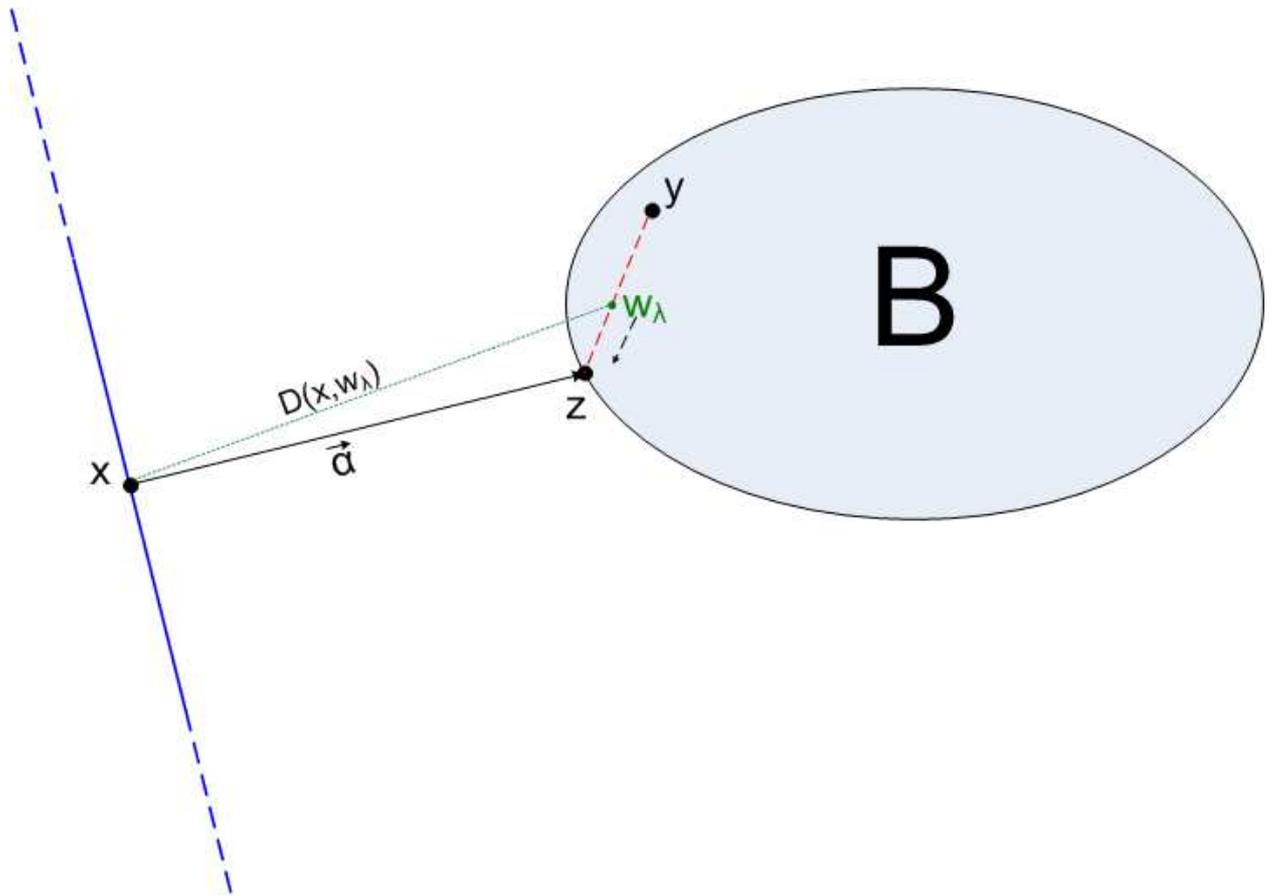


Figure 4.1: Supporting Hyperplane

As  $B$  is convex, for any  $0 \leq \lambda \leq 1$ ,

$$\vec{w}_\lambda = \lambda \vec{y} + (1 - \lambda) \vec{z} \in B$$

The square of the distance between  $\vec{x}$  and  $\vec{w}_\lambda$  is given by

$$D^2(\vec{x}, \vec{w}_\lambda) = |\vec{x} - \lambda \vec{y} - (1 - \lambda) \vec{z}|^2 = \sum_{i=1}^d (x_i - \lambda y_i - (1 - \lambda) z_i)^2$$

Deriving by  $\lambda$  we obtain

$$\begin{aligned} \frac{\partial D^2}{\partial \lambda} &= 2(\vec{x} - \lambda \vec{y} - (1 - \lambda) \vec{z}) \cdot (\vec{z} - \vec{y}) \\ &= 2(\vec{z} - \vec{x}) \cdot \vec{y} - 2(\vec{z} - \vec{x}) \cdot \vec{z} + 2\lambda(\vec{z} - \vec{y})^2 \\ &= 2\vec{\alpha} \cdot \vec{y} - 2\vec{\alpha} \cdot \vec{z} + 2\lambda(\vec{z} - \vec{y})^2 \end{aligned}$$

Evaluating for  $\lambda = 0$  we get

$$\frac{\partial D^2}{\partial \lambda} = 2\vec{\alpha} \cdot \vec{y} - 2\vec{\alpha} \cdot \vec{z}$$

But according to our assumption the first term  $\vec{\alpha} \cdot \vec{y} \leq \alpha_{d+1}$  and we have shown that the second term  $\vec{\alpha} \cdot \vec{z} > \alpha_{d+1}$ , and therefore

$$\left. \frac{\partial D^2}{\partial \lambda} \right|_{\lambda=0} < 0$$

Hence, for  $\lambda$  close enough to 0 we must have

$$D^2(\vec{x}, \vec{w}_\lambda) < D^2(\vec{x}, \vec{z})$$

But  $\vec{z}$  was chosen to minimize the distance to  $\vec{x}$ , so we have a contradiction. Therefore for all  $\vec{y} \in B$ , 4.11 must hold. □

**Lemma 4.8** (*Theorem of the Alternative for Matrices*). Let  $A = (a_{ij})$  be an  $m \times n$  real matrix. Let  $\{\vec{a}_i\}_{i=1}^m = (a_{i1}, a_{i2}, \dots, a_{in})$  be the rows of the matrix. Then one of the following must hold:

1. The point  $\vec{0}$  in  $R^n$  is in the convex hull of the  $m + n$  points  $\{\vec{a}_i\}_{i=1}^m \cup \{\vec{e}_i\}_{i=1}^n$  where  $\vec{e}_i$  is the  $i$ th elementary vector in  $R^n$ .
2. There exists a stochastic vector  $\vec{x} = (x_1, \dots, x_n) \in R^n$  satisfying

$$\begin{aligned} \sum_{j=1}^n x_j &= 1 \\ \forall 1 \leq j \leq n, \quad x_j &> 0 \\ \forall 1 \leq i \leq m, \quad \vec{a}_i \cdot \vec{x} = \sum_{j=1}^n a_{ij} x_j &> 0 \end{aligned}$$

**Proof:** Suppose 1 does not hold. Denote the convex hull mentioned in 1 by  $C$ . If we apply Lemma 4.7, there exist  $\vec{\alpha} \in R^n$  and  $\alpha_{n+1}$  such that

$$\vec{\alpha} \cdot \vec{0} = \alpha_{n+1}$$

(which means that  $\alpha_{n+1} = 0$ , of course) and

$$\forall \vec{y} \in C, \quad \vec{\alpha} \cdot \vec{y} > 0$$

In particular, this will hold if  $\vec{y}$  is any of the vectors  $\vec{a}_i$  or  $\vec{e}_i$ . Thus

$$\begin{aligned} \vec{a}_i \cdot \vec{\alpha} &> 0 \quad \text{for all } 1 \leq i \leq m, \\ \alpha_j &> 0 \quad \text{for all } 1 \leq j \leq n. \end{aligned}$$

Since  $\forall 1 \leq j \leq n$ ,  $\alpha_j > 0$  we have  $\sum_{j=1}^n \alpha_j > 0$ , so we can scale by the sum and define

$$x_j = \alpha_j / \sum_{j=1}^n \alpha_j$$

Therefore

$$\begin{aligned} \sum_{j=1}^n x_j &= 1 \\ \forall 1 \leq j \leq n, \quad x_j &> 0 \\ \forall 1 \leq i \leq m, \quad \vec{a}_i \cdot \vec{x} = \sum_{j=1}^n a_{ij} x_j &> 0 \end{aligned}$$

□

**Proof of the Minimax Theorem:** Let  $A_{m \times n}$  be a payoff matrix for a 2-player zero sum game. Applying Lemma 4.8 to  $A^T$ , either 1 or 2 must hold. If 1 holds, then  $\vec{0}$  is in the convex hull of the columns of  $A$  and the elementary vectors in  $R^m$ . Thus, there exist  $s_1, \dots, s_{n+m}$  such that

$$\begin{aligned} \sum_{j=1}^n a_{ij} s_j + s_{n+i} &= 0 \quad \forall 1 \leq i \leq m \\ s_i &\geq 0 \quad \forall 1 \leq i \leq n+m \\ \sum_{i=1}^{n+m} s_i &= 1 \end{aligned}$$

Now, it is impossible for all of  $s_1, \dots, s_n$  to be equal to 0, because the first equation would mean that all  $s_i$  are 0, and then equation 3 cannot hold (in other words, the vector  $\vec{0}$  cannot be a convex combination of  $\vec{e}_i$  alone, because they are linearly independent). Therefore at least one of  $s_1, \dots, s_n$  is positive, and  $\sum_{k=1}^n s_k > 0$ . We can therefore define a mixed strategy  $y$  for player II by

$$\forall 1 \leq j \leq n, \quad y_j = s_j / \sum_{k=1}^n s_k$$

And we have:

$$\begin{aligned} y_j &\geq 0 \quad \forall 1 \leq j \leq n \\ \sum_{j=1}^n y_j &= 1 \\ \sum_{j=1}^n a_{ij} y_j = -s_{n+i} / \sum_{k=1}^n s_k &\leq 0 \quad \forall 1 \leq i \leq m \end{aligned}$$

Therefore

$$v_I(y) = \max_{1 \leq i \leq m} e_i A y^T \leq 0$$

Thus  $V_{II} \leq 0$ .

If 2 holds, then we have a stochastic vector  $x \in R^m$ , which we will view as a mixed strategy for player I, that satisfies

$$\forall 1 \leq j \leq n, \sum_{i=1}^m a_{ij} x_i > 0$$

so  $v_{II}(x) = \min_{1 \leq j \leq n} x A e_j^T > 0$  and therefore  $V_I > 0$ . Because one of the two must hold, we see that it is impossible to have  $V_I \leq 0 < V_{II}$ .

Now, for any real  $M$ , let us look at the 2-player zero sum game defined by the payoff matrix  $B = (b_{ij})$  where

$$b_{ij} = a_{ij} - M$$

For any  $x, y$ ,

$$x B y^T = x A y^T - M$$

Hence,

$$\begin{aligned} V_I(B) &= V_I(A) - M \\ V_{II}(B) &= V_{II}(A) - M \end{aligned}$$

And since it is impossible that  $V_I(B) \leq 0 < V_{II}(B)$ , it is also impossible that

$$V_I(A) \leq M < V_{II}(A)$$

But this is true for any real  $M$ , thus it is impossible that

$$V_I < V_{II}$$

And we have shown that  $V_I \leq V_{II}$ , therefore

$$V_I = V_{II}$$

□

## 4.6 Results

We have shown that in a 2-player zero sum game the gain-ceiling for player I is equal to the loss-floor for player II. We denote this value simply by  $V$  and call it the *value of the game*.

Part 2 of Lemma 4.4 tells us that  $V_I = V_{II}$  means that we have a Nash equilibrium point. It is easy to see that the payoff in this equilibrium is exactly the value of the game. Theorem 4.2 tells us that all Nash equilibria will have this value, and that the set of all Nash equilibria is actually a cartesian product of the equilibrium strategies of each player.

A strategy  $x$  for player I satisfying

$$\forall 1 \leq j \leq n, \quad \sum_{i=1}^m x_i a_{ij} \geq V \quad (4.12)$$

is optimal for player I in the sense that this strategy guarantees a payoff of  $V$  against every strategy of player II, and there is no strategy that guarantees a higher payoff against every strategy of player II. Similarly, a strategy  $y$  for player II satisfying

$$\forall 1 \leq i \leq m, \quad \sum_{j=1}^n y_j a_{ij} \leq V \quad (4.13)$$

is optimal for player II. It is clear that

$$xAy^T = V$$

otherwise one of 4.12 or 4.13 will not hold. It is easy to see that  $(x, y)$  is a Nash equilibrium. Also, any Nash equilibrium must satisfy 4.12 and 4.13.

To summarize

**Theorem 4.9** *Let  $G$  be a 2-player zero sum game. Then*

1. *The gain-floor for player I and loss-ceiling for player II are equal (the value of the game,  $V$ ).*
2. *There is at least one Nash equilibrium.*
3. *The set of equilibrium points for the game is the cartesian product of the sets of equilibrium strategies for each player.*
4. *The value of the game in all equilibrium points is  $V$*
5. *The set of equilibrium strategies for each player is equal to the set of optimal strategies for the player*
6. *The set of optimal strategies for player I is the solution of the following linear program in variables  $x_1, \dots, x_m, V$*

$$\begin{aligned} \forall 1 \leq j \leq n, \quad & \sum_{i=1}^m x_i a_{ij} - V && \geq 0 \\ \forall 1 \leq i \leq m & x_i && \geq 0 \\ & \sum_{i=1}^m x_i && = 1 \end{aligned}$$

*Maximize target function  $V$*

and the dual program gives the optimal strategies for player II.

The problem of finding Nash equilibria for a 2-player zero sum game is therefore in  $\text{NP} \cap \text{co-NP}$ , and is solvable in polynomial time by linear programming.

## 4.7 Application of Zero Sum Games in Computer Science

### 4.7.1 Deterministic vs. Random Algorithms

In this example  $\Omega = \{A_i\}$  is a finite set of deterministic algorithms that can take as input any element of the finite input set  $\Lambda = \{x_j\}$ . We will denote by  $\Delta_S$  the set of probability distributions over the set  $S$ , for any set  $S$ .

**Definition**  $\text{Time}(\mathbf{A}, \mathbf{x})$  is the time complexity (measured, as usual in complexity, in the means of number of commands) of running the deterministic algorithm  $A$  with the input  $x$ . Also denoted  $T(A, x)$ .

**Definition** A **Random Algorithm** is a probability distribution over the deterministic algorithms,  $\vec{p} \in \Delta_\Omega$ . We denote the probability for algorithm  $A_i$  by  $p_i$ .

**Definition**  $\text{RTime}(\vec{p}, \mathbf{x})$  is time complexity of the random algorithm defined by distribution  $\vec{p}$  for fixed input  $x$ . It is defined as the expected deterministic time complexity for the fixed input  $x$ :

$$\text{RTime}(\vec{p}, x) = \sum_i p_i \cdot T(A_i, x)$$

**Definition**  $\text{AvgTime}(\mathbf{A}, \vec{q})$  is the time complexity of deterministic algorithm  $A$  given distribution  $\vec{q}$  over inputs. This is in essence an average-case complexity analysis for  $A$ . It is defined as the expected time complexity for the deterministic algorithm  $A$  with input distributed according to  $\vec{q}$ :

$$\text{AvgTime}(A, \vec{q}) = \sum_j q_j \cdot T(A, x_j)$$

### Complexity Analysis

**Corollary 4.10** *Deterministic worst-case time complexity is  $\min_i \max_j T(A_i, x_j)$ .*

**Proof:** The complexity of the problem is the minimum complexity over all relevant algorithms ( $\Omega$ ). We must choose the deterministic algorithm before knowing the input. Thus, the complexity of deterministic algorithm  $A_i$  is analyzed for the worst input, which yields complexity  $\max_j T(A, x_j)$ , and then the complexity of the problem is the complexity of the best algorithm, which results in complexity  $\min_i \max_j T(A_i, x_j)$ .  $\square$

**Corollary 4.11** *Non-deterministic worst-case time complexity is  $\max_j \min_i T(A_i, x_j)$*

**Proof:** For non-deterministic algorithms we can guess the best deterministic algorithm given the input. Thus, for input  $x_j$ , the complexity is  $\min_i T(A_i, x_j)$ . We now analyze for the worst case input, which yields complexity  $\max_j \min_i T(A_i, x_j)$ .  $\square$

**Corollary 4.12** *Random worst-case time complexity is*

$$\min_{\vec{p} \in \Delta_\Omega} \max_j RTime(\vec{p}, x_j)$$

**Theorem 4.13 (Yao's Lemma)** *For any distribution  $\vec{p}$  on the algorithms and  $\vec{q}$  on inputs*

$$\max_j E_{i \sim \vec{p}} [T(A_i, x_j)] \geq \min_i E_{j \sim \vec{q}} [T(A_i, x_j)]$$

**Proof:** We can view the complexity analysis as a 2-player zero sum game in the following way. The max. player pure strategies are the possible inputs,  $\Lambda$ . The min. player pure strategies are the deterministic algorithms  $\Omega$ . The payoff is the time complexity  $T(A_i, x_j)$ .

Given such a game, we can see that

$$\min_{\vec{p} \in \Delta_\Omega} \max_{\vec{q} \in \Delta_\Lambda} E_{i \sim \vec{p}, j \sim \vec{q}} [T(A_i, x_j)] = \max_{\vec{q} \in \Delta_\Lambda} \min_{\vec{p} \in \Delta_\Omega} E_{i \sim \vec{p}, j \sim \vec{q}} [T(A_i, x_j)] \quad (4.14)$$

As in the previous game analysis, it is easily shown that the internal maximum and minimum are obtained in deterministic points:

$$\max_{\vec{q} \in \Delta_\Lambda} E_{i \sim \vec{p}, j \sim \vec{q}} [T(A_i, x_j)] = \max_j E_{i \sim \vec{p}} [T(A_i, x_j)] \quad (4.15)$$

$$\min_{\vec{p} \in \Delta_\Omega} E_{i \sim \vec{p}, j \sim \vec{q}} [T(A_i, x_j)] = \min_i E_{j \sim \vec{q}} [T(A_i, x_j)] \quad (4.16)$$

Using only the  $\geq$  part of 4.14, and substituting using 4.15 and 4.16 we obtain

$$\min_{\vec{p} \in \Delta_\Omega} \max_j E_{i \sim \vec{p}} [T(A_i, x_j)] \geq \max_{\vec{q} \in \Delta_\Lambda} \min_i E_{j \sim \vec{q}} [T(A_i, x_j)] \quad (4.17)$$

Hence for any  $\vec{p} \in \Delta_\Omega$

$$\max_j E_{i \sim \vec{p}} [T(A_i, x_j)] \geq \max_{\vec{q} \in \Delta_\Lambda} \min_i E_{j \sim \vec{q}} [T(A_i, x_j)]$$

Thus for any  $\vec{p} \in \Delta_\Omega$  and  $\vec{q} \in \Delta_\Lambda$

$$\max_j E_{i \sim \vec{p}} [T(A_i, x_j)] \geq \min_i E_{j \sim \vec{q}} [T(A_i, x_j)]$$

□

Note that Yao's Lemma is actually a result of the weaker inequality established in Lemma 4.4.

**Corollary 4.14** *In order to prove a lower bound for the worst-case complexity of any random algorithm for a given problem, it is sufficient to prove a lower bound for any deterministic algorithm on some distribution of the input.*

**Proof:** Using

$$\begin{aligned} E_{i \sim \vec{p}} [T(A_i, x_j)] &= RTime(\vec{p}, x_j) \\ E_{j \sim \vec{q}} [T(A_i, x_j)] &= AvgTime(A_i, \vec{q}) \end{aligned}$$

we can write Yao's Lemma as

$$\max_j RTime(\vec{p}, x_j) \geq \min_i AvgTime(A_i, \vec{q})$$

so given a lower bound  $B$  on the complexity of any deterministic algorithm on some input distribution  $\vec{q}$ , we obtain

$$B \leq \min_i AvgTime(A_i, \vec{q}) \leq \max_j RTime(\vec{p}, x_j)$$

So  $B$  is a lower bound on the worst-case complexity of any random algorithm. □

### Example - Sorting a List of Numbers

We wish to bound the complexity of a random algorithm for sorting  $n$  numbers (comparison based sort). We can describe any deterministic comparison based sort algorithm as a decision tree, where each internal node corresponds to a comparison the algorithm performs, with 2 possible outcomes (we assume all elements are different). For a specific input, the execution of the algorithm corresponds to a path from the root to a leaf. It is impossible for 2 different permutations to result in the same path. The running time for the algorithm over an input is the length of the path.

Therefore, the decision tree must have at least  $n!$  leaves. Thus the depth of the tree is at least  $\log(n!) = O(n \log n)$  nodes. The number of leaves whose depth is not greater than  $l$  is  $\leq 2^{l+1}$ .

Thus, for any deterministic algorithm  $A$ , at least one half of the permutations are in depth greater than  $l$ , where  $l + 1 = \log(n!/2)$  (since then the number of leaves whose depth is less than  $l$  is  $\leq 2^{\log(n!/2)} = n!/2$ ).  $l + 1 = \log(n!/2) \implies l = \log(n!) - 2 = O(n \log n)$ .

We shall choose a uniform distribution  $\vec{q}$  over the possible inputs (all permutations of  $n$  numbers), and fix a deterministic algorithm  $A$ . The running time of  $A$  over this distribution is simply the average of the depths of the leaves for all possible inputs. But at least  $n!/2$  inputs are of depth at least  $\log(n!) - 1$ , so the average running time will be at least

$$\frac{\frac{n!}{2} \cdot (\log(n!) - 1)}{n!} = \Omega(n \log n)$$

And using Yao's lemma, the complexity of any random algorithm is also  $\Omega(n \log n)$ .

### 4.7.2 Weak vs. Strong Learning

Given a weak learner for a binary classification problem we will show that strong learning is possible.

**The model:**  $f$  is the target function,  $H$  a function family.

$$f : X \longrightarrow \{0, 1\}$$

$$\forall h \in H, \quad h : X \longrightarrow \{0, 1\}$$

$X$  is finite, and as a consequence  $H$  is finite ( $|H| \leq 2^{|X|}$ )

**The WL (weak learning) assumption:** For every distribution  $D$  on  $X$  there exists  $h \in H$  and  $\epsilon > 0$  such that

$$[Pr_D[h(x) = f(x)] \geq 1/2 + \epsilon] \tag{4.18}$$

**Question:** can  $f$  be approximated by functions in  $H$ ?

We represent the problem as a 2-player zero sum game as follows. The max. player pure strategies are the inputs  $X$ . The min. player pure strategies are the functions  $H$ . The payoff is an error indicator:

$$M(h, x) = \begin{cases} 0 & \text{if } f(x) = h(x) \text{ (no error)} \\ 1 & \text{if } f(x) \neq h(x) \text{ (error)} \end{cases}$$

Note that  $M(h, x) = |(f(x) - h(x))|$ . The max. player is trying to select a distribution over  $X$  that will maximize the expected error, while the min. player is trying to select a distribution over  $H$  that will minimize it.

The WL proposition implies that for each  $D$  there exists an  $\epsilon > 0$  and  $h \in H$  such that

$$1/2 - \epsilon \geq Pr_D[h(x) \neq f(x)] = Pr_D[M(h, x) = 1] = E_{x \sim D} D[M(h, x)]$$

Thus there exists an  $\epsilon > 0$  so that

$$\min_h E_{x \sim D}[M(h, x)] \leq 1/2 - \epsilon$$

Which means that

$$V_{X \text{ Player}} \leq 1/2 - \epsilon$$

Since in a zero sum game, the values of both players are equal, we conclude that

$$V_{H \text{ Player}} < 1/2 - \epsilon$$

hence

$$\min_q \max_x E_{h \sim q}[M(h, x)] \leq 1/2 - \epsilon$$

therefore there exists a distribution  $q$  (the one in which the minimum is obtained) such that

$$\max_x E_{h \sim q}[M(h, x)] \leq 1/2 - \epsilon$$

thus

$$\forall x \in X, \quad E_{h \sim q}[M(h, x)] \leq 1/2 - \epsilon$$

In other words, for this  $q$ , and all  $x \in X$ ,

$$1/2 - \epsilon \geq \sum_{h \in H} q(h)M(h, x) = \sum_{h \in H} q(h) |f(x) - h(x)|$$

We define an approximation  $G(x) = \sum_{h \in H} q(h) \cdot h(x)$ . Now, for all  $x \in X$ ,

$$|f(x) - G(x)| = \left| \sum_{h \in H} q(h)[f(x) - h(x)] \right| \leq \sum_{h \in H} q(h) |f(x) - h(x)| < 1/2$$

So  $G(x)$  is a strong approximation for  $f(x)$  (by rounding  $G(x)$  to 0 or 1 we obtain  $f(x)$  for all  $x \in X$ ).

### 4.7.3 Playing a Zero-Sum Game

Let  $A$  be the game matrix. In each time  $t$ :

- The rows player chooses a distribution  $p_t$ .
- The columns player chooses a distribution  $q_t$  (the player may or may not know  $p_t$  before choosing  $q_t$ ).
- The rows player loses  $p_t A q_t^T$ . His goal is to minimize his total loss,  $\sum_t p_t A q_t^T$
- After playing time  $t$ , the rows player is also given the vector  $A q_t^T$ , so he knows what his loss would have been in time  $t$  for any strategy he would have played.

**KL distance**

**Definition** KL(Kullback-Leibler) distance is defined as follows

$$KL(P_1||P_2) = \sum_x P_1(x) \ln \left( \frac{P_1(x)}{P_2(x)} \right)$$

Characteristics:  $KL(P_1||P_1) = 0$ , and for every  $P_1 \neq P_2$ ,  $KL(P_1||P_2) > 0$ .

**An algorithm**

We attach a weight for each action, at each time. The weights are updated as follows:

$$w_{t+1}(i) = w_t(i) \beta^{L_t(i)}$$

where  $\beta \in (0, 1)$  is a parameter,

$$L_t(i) = (Aq_t^T)_i = e_i A q_t^T$$

is the loss caused by strategy  $i$  at time  $t$ , and

$$w_1(i) = 1$$

At time  $t$  the algorithm chooses a distribution  $p_t$  such that

$$p_t(i) = w_t(i) / \sum_{j=1}^N w_t(j)$$

Denoting

$$z_t = \sum_{j=1}^N w_t(j)$$

**Theorem 4.15** For any game matrix  $A$  and for any  $q_1, \dots, q_t$  the  $p_1, \dots, p_t$  that the algorithm chooses satisfy:

$$Loss = \sum_t p_t A q_t^T \leq \min_p \left[ a_\beta \sum_t p(A q_t^T) + c_\beta KL(p||p_1) \right]$$

where

$$a_\beta = \frac{\ln(1/\beta)}{1-\beta}$$

$$c_\beta = \frac{1}{1-\beta}$$

**Lemma 4.16** For every iteration  $t$  and for every  $\tilde{p}$

$$KL(\tilde{p}\|p_{t+1}) - KL(\tilde{p}\|p_t) \leq \ln\left(\frac{1}{\beta}\right) \tilde{p}Aq_t^T + \ln(1 - (1 - \beta)p_tAq_t^T)$$

**Proof:**

$$\begin{aligned} KL(\tilde{p}\|p_{t+1}) - KL(\tilde{p}\|p_t) &= \sum_i \tilde{p}(i) \ln\left(\frac{p_t(i)}{p_{t+1}(i)}\right) = \sum_i \tilde{p}(i) \ln\left(\frac{z_{t+1}}{\beta^{L_t(i)} z_t}\right) = \\ &= \ln\left(\frac{z_{t+1}}{z_t}\right) + \sum_i \tilde{p}(i) \ln\left(\frac{1}{\beta}\right)^{L_t(i)} = \ln\left(\frac{z_{t+1}}{z_t}\right) + \ln\left(\frac{1}{\beta}\right) \sum_i \tilde{p}(i) L_t(i) = \\ &= \ln\left(\frac{z_{t+1}}{z_t}\right) + \ln\left(\frac{1}{\beta}\right) \tilde{p}Aq_t^T \end{aligned} \quad (4.19)$$

But

$$\ln\frac{z_{t+1}}{z_t} = \ln\frac{\sum_i w_{t+1}(i)}{\sum_i w_t(i)} = \ln\sum_i \frac{w_t(i)}{\sum_j w_t(j)} \cdot \beta^{L_t(i)} = \ln\sum_i p_t(i) \beta^{L_t(i)}$$

Using  $\beta^x \leq 1 - (1 - \beta)x$  for  $x \in [0, 1]$  we obtain

$$\begin{aligned} \ln\frac{z_{t+1}}{z_t} &\leq \ln\left(\sum_i p_t(i) (1 - (1 - \beta)L_t(i))\right) = \ln\left(1 - (1 - \beta) \sum_{i=1}^N p_t(i) L_t(i)\right) = \\ &= \ln(1 - (1 - \beta)p_tAq_t^T) \end{aligned} \quad (4.20)$$

Combining 4.19 and 4.20

$$KL(\tilde{p}\|p_{t+1}) - KL(\tilde{p}\|p_t) \leq \ln(1 - (1 - \beta)p_tAq_t^T) + \ln\left(\frac{1}{\beta}\right) \tilde{p}Aq_t^T \quad (4.21)$$

□

**Proof of the Theorem:**

Let  $\tilde{p}$  be some distribution over the rows. Since  $\ln(1 - x) \leq -x$  for  $x < 1$  we have,

$$\ln(1 - (1 - \beta)p_tAq_t^T) \leq -(1 - \beta)p_tAq_t^T$$

Hence from Lemma 4.16,

$$KL(\tilde{p}\|p_{t+1}) - KL(\tilde{p}\|p_t) \leq -(1 - \beta)p_tAq_t^T + \ln\left(\frac{1}{\beta}\right) \tilde{p}Aq_t^T \quad (4.22)$$

Summing 4.22 for all  $t = 1, \dots, T$  yields a telescope sum, resulting in

$$KL(\tilde{p} \| p_{T+1}) - KL(\tilde{p} \| p_1) \leq -(1 - \beta) \sum_{t=1}^T p_t A q_t^T + \ln\left(\frac{1}{\beta}\right) \sum_{t=1}^T \tilde{p} A q_t^T$$

But since  $KL(\tilde{p} \| p_{T+1}) \geq 0$  we obtain

$$(1 - \beta) \sum_{t=1}^T p_t A q_t^T \leq KL(\tilde{p} \| p_1) + \ln\left(\frac{1}{\beta}\right) \sum_{t=1}^T \tilde{p} A q_t^T$$

Thus

$$Loss = \sum_{t=1}^T p_t A q_t^T \leq c_\beta KL(\tilde{p} \| p_1) + a_\beta \sum_{t=1}^T \tilde{p} A q_t^T$$

For any distribution  $\tilde{p}$ , which proves the theorem. □

We can now use the theorem to bound the average loss per step for our algorithm. Substituting  $p_1$  to be uniform distribution ( $w_1(i) = 1$ ) means  $KL(\tilde{p} \| p_1) \leq \ln(N)$ , because

$$KL(\tilde{p} \| p_1) = \sum_x \tilde{p}(x) \ln\left(\frac{\tilde{p}(x)}{1/N}\right) = \sum_x \tilde{p}(x) \ln(\tilde{p}(x) \cdot N) \leq \sum_x \tilde{p}(x) \ln(N) \leq \ln(N)$$

So now we have

$$\sum_{t=1}^T p_t A q_t^T \leq c_\beta \ln N + a_\beta \sum_{t=1}^T \tilde{p} A q_t^T \tag{4.23}$$

Choosing  $\beta = \frac{1}{1 + \sqrt{\frac{2 \ln(N)}{T}}} \sim 1 - \Theta\left(\sqrt{\ln \frac{N}{T}}\right)$  we get

$$\frac{1}{T} Loss = \frac{1}{T} \sum_{t=1}^T p_t A q_t^T \leq \min_p \frac{1}{T} \sum_{t=1}^T p A q_t^T + \Delta$$

Where  $\Delta = \sqrt{\frac{2 \ln N}{T}} + \frac{\ln N}{T}$ . This is achieved by substituting our choice of  $\beta$  in 4.23 and using the approximation  $\ln(1/\beta) \leq (1 - \beta^2)/(2\beta)$  for  $\beta \in (0, 1]$ .

The meaning of this is that the difference between average loss per step to that of the optimal fixed strategy is bounded by a value that be made arbitrarily small for large  $T$ .