

Lecture 4

Monday, 29 August 2022 10:59 AM

Lemke-Howson: An algorithm for finding a NE in bimatrix games.

Step 1: Show reduction to symm. games

Step 2: Write down LP for which "special" vertices correspond to NE in the symm. game

Step 3: Find a "special" vtx. in the polytope given by the LP.

Symmetric Games:

Defn: A bimatrix game (R, C) is called **symmetric** if $R = C$

Thus for a symmetric game, row player's payoff for the mixed strategy $(x, y) =$ col player's payoff for the strategy (y, x) . NOT THE SAME AS IDENTICAL GAMES!

Also, both $x, y \in \Delta_n$

Given a bimatrix game $(R, C) \in \mathbb{R}^{n \times m} \times \mathbb{R}^{m \times n}$, $R, C \geq 0$, Consider the symmetric game

$$n \begin{bmatrix} 0 & C \\ R & 0 \end{bmatrix} = R' = C'$$

Lemma: Let $(w^1, x^1), (z^1, y^1)$ be a NE for R' , where $w, z \in \mathbb{R}^n, x, y \in \mathbb{R}^m$

ⓐ If $z \neq 0$, then $(\frac{x}{\|x\|_1}, \frac{z}{\|z\|_1})$ is a NE for (R, C)

ⓑ Else, $(\frac{y}{\|y\|_1}, \frac{w}{\|w\|_1})$ is a NE for (R, C)

Proof: Assume $z \neq 0$.

$$\text{Consider } R' \begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} 0 & C \\ R & 0 \end{bmatrix} \begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} Cy \\ Rz \end{bmatrix}$$

Since $(w^1, x^1) \in BR$, $x_i > 0 \Rightarrow (Rz)_i = \max_{k \in [m]} (Rz)_k$

Hence for (R, C) , x is BR to z

$$\text{Also, } R' \begin{bmatrix} w \\ x \end{bmatrix} = \begin{bmatrix} Cx \\ R w \end{bmatrix}$$

And since (z^1, y^1) is BR, $z_j > 0 \Rightarrow (Cx)_j = \max_{k \in [n]} (Cx)_k$

Hence for (R, C) , z is BR to x .

Thus if $z \neq 0$, then (x, z) is a NE for (R, C)

(note: if $x=0$, then $w \neq 0$, and hence $Cx=0, R w \neq 0$, & $\exists i: (R w)_i > 0$. Then $z \neq 0$ cannot be a BR)

Similarly, if $z=0$, then $y \neq 0$, & following above, $w \neq 0$. And we can show $(\frac{y}{\|y\|_1}, \frac{w}{\|w\|_1})$ is a NE

for (R, C) . □

Hence, to find a NE for (R, C) , suffices to find eq. in symmetric games.

Let $(R, R) \in \mathbb{R}^{m \times m}$ be a symmetric game. Consider the polytope:

$$P: \forall i \in [m], x_i \geq 0 \\ \forall i \in [m], (Rx)_i \leq 1$$

(As before, assume $R \geq 0$. Further, assume P is nondegenerate).

$x \in P$, we say coordinate i is **represented** if either $x_i = 0$ or $(Rx)_i = 1$.

And $x \in P$ is a **democracy** if every coordinate is represented at x .

Note that 0^m is a democracy.

Finally, let $Q = \{x \in P: \text{coordinates } 1 \dots m-1 \text{ are represented, coordinate } m \text{ is NOT represented, some coordinate is represented twice}\}$

Q: Show that if $x \in Q$, then x must be a vertex of P .

Claim: If $v \neq 0$ is a democracy, then $(\frac{v}{\|v\|_1}, \frac{v}{\|v\|_1})$ is a NE of (R, R)

Proof: Consider Rv , and some $i \in [m]$ s.t. $v_i > 0$. Then since v is a democracy, then $(Rv)_i \leq 1$ & $(Rv)_i = 1$, hence v is supported on max-payoff strategies & hence is a BR.

Lemke-Howson Algorithm:

- Start at vertex $v_0 = (0, 0, \dots, 0)$ w/ m tight constraints (since P is nondegenerate).
- "Untighten" the constraint $x_m = 0$. The $(m-1)$ tight constraints that define an edge of the polytope, one end of which is $v = 0$. Find the other end, say v^1 (How?)
- At the next vtx v^1 , the following holds:
 - Either v^1 is a democracy, or every coordinate $1 \dots m-1$ is represented, and exactly one coordinate k is represented twice, i.e., $x_k = 0$ & $(Rv^1)_k = 1$
 - Exactly one of these two equalities was not tight (i.e., was a strict inequality at the previous vertex. All the other tight inequalities were tight at the previous vtx.
 - Coordinate m is not represented. Hence, $v^1 \in Q$
- Untighten the inequality for coordinate k that was tight at the earlier vertex, and proceed along the resulting edge for the next vertex.
- Stop when you reach a democracy.

Let $v^0, v^1, \dots, v^{t-1}, v^t, \dots$ be the sequence of vertices visited.

Claim: Except the first & last vertex, all vertices v^1, v^2, \dots are in Q .

Proof: Since each v^i is a vertex, exactly m tight constraints at each.

Claim is true of v^1 . Assume true for v^{t-1} , will show it is true for v^t .

- By algo, for coordinates $1 \dots m-1$, a tight constraint continues to be tight at v^t .

- Since v^t is a vertex, P is nondegenerate, exactly one more constraint will be tight.

If this is a $x_m \geq 0$ or $(Rv^t)_m \leq 1$, then v^t is a democracy & algo ends.

Else, for $k \in [m-1]$, one additional constraint is tight. Hence $v^t \in Q$.

Lastly, we need to show that the algo doesn't cycle.

Lemma: The algorithm stops at a democracy that is not 0^m .

Proof: Consider the graph $G = (V, E)$ where:

- each vertex $w \in P$ is a vertex $w \in V$ if either:
 - w is a democracy, or
 - $w \in Q$
- each democratic $v \in V$ has an edge to $w \in V$, where w is the vertex reached by untightening $x_m = 0$ or $(Rv)_m = 1$
- each $v \in V$ that corresponds to vertex $v \in Q$ has 2 neighbours: if coordinate k is represented twice at v , then the neighbours are the two vertices obtained by untightening the 2 constraints for coordinate k .

Then each vertex has degree 1 or 2 in the graph, & hence the graph consists of paths & cycles.

Further, v has degree 1 in the graph iff v is a democracy.

Hence by the L-H algorithm, we will reach the other end of the path that starts at 0^m , which must be another democracy. □

Untightening a tight constraint: