Admin Issues

Email me if you still have issues accessing canvas

Please complete **demographic survey** if you haven't

- Canvas →Quizzes, not graded

Office hours **5-6pm Monday**, starting the **following week**

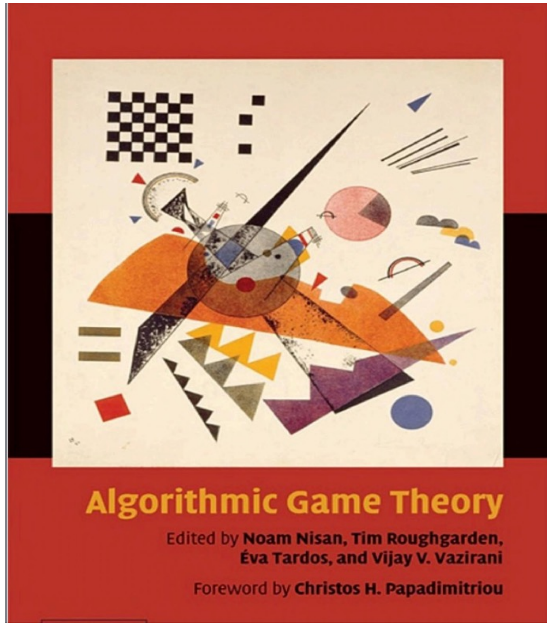
Can ask questions on canvas too

- Refrain from emails for content related questions unless necessary

Related reference material on canvas

Remind me if I haven't responded to your email

Lecture 2: Nash Equilibrium



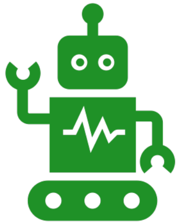
Lecture 2: 20 Aug 2025

CS6208 Fall 2025: Computational Game Theory

Recall: Need to be strategic

Should I fold, raise, or call?

The 7-2 offsuit is the ”worst” possible hand…
…so maybe we should fold…
…but I could possibly win big if I were to pretend to have a strong hand…
…but if I pretend all the time, it may become easy for my opponent to call my bluff…
…also, what cards could my opponent hold?…
…, …, …



This lecture: the simultaneous move case

Normal Form Games

General-Sum Matrix Games

Matrix games AKA Normal-form, Strategic-form

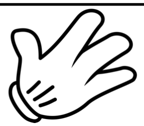
- Players (>1) This class focuses on 2 players
- Actions (per player, finite)
- Payoffs (per strategy profile)

Payoffs are represented by finite matrix (or “tensor”)

Column player aka P2

P2’s actions $\mathcal{A}_2$

Row player aka P1

			
	0,0	−1,1	1, −1
	1, −1	0,0	−1,1
	−1,1	1, −1	0,0

P1’s actions $\mathcal{A}_1$

P1’s payoff, P2’s payoff $u_1(a_1, a_2), u_2(a_1, a_2)$

Rock-Paper-Scissors

Some classic 2x2 games

2,2	0,0
0,0	2,2

Coordination game

8,8	0,7
7,0	5,5

Stag hunt

8,4	0,0
0,0	4,8

Battle of the Sexes

6,6	1,7
7,1	0,0

Chicken Game

10,10	-1,12
12,-1	0,0

Prisoner's Dilemma

Analyze them on the board

Important Classes of Bimatrix Games

Symmetric games

- Game "looks" the same whether you are P1 or P2

Zero-sum Games

- Purely competitive, $u_1 = -u_2$

Cooperative Games

- Purely cooperative, $u_1 = u_2$

Which of the previous games are any of these?

A Gentle Introduction to Nash Equilibrium

Best responses (intuitive)

If other player **fixes** choice of action (possibly randomized), how should I play?

Best responses are typically **set valued**

	0,0	$-1,1$	$1,-1$
	$1,-1$	0,0	$-1,1$
	$-1,1$	$1,-1$	0,0

$BR_1(\ ) = $ $BR_1(\ ) = \{\ \}$

$BR_2(\ ) = $ $BR_2(\ ) = \{\ \}$

$BR_2(½\ ½\ ) = $ $BR_2(½\ ½\ ) = \{\ \}$

$BR_2(^1/_3\ ^1/_3\ ^1/_3\ ) = \{\ \ \ (½\ ½\ )\}$

Contains all **convex combinations**!

Exercise: BR for Battle of the Sexes

	Soccer	Shopping
Soccer	**8,4**	**0,0**
Shopping	**0,0**	**4,8**

Battle of the Sexes

Find the following:

- $BR_1(soccer)$
- $BR_1(shopping)$
- $BR_1(0.5\ shopping, 0.5\ soccer)$
- $BR_1\left(\frac{1}{3}\ soccer, \frac{2}{3}\ shopping\right)$

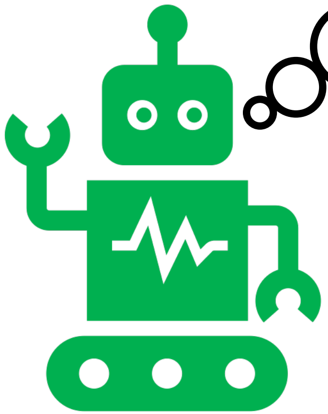
Unilateral Deviations

$a_1 \rightarrow a_1{}'$

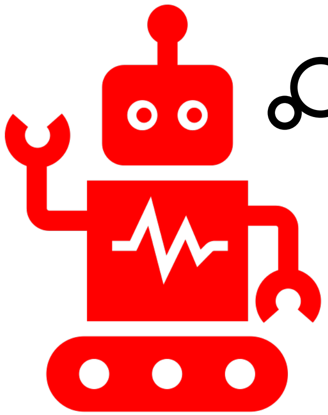
Can I do better than $u_1(a_1, a_2)$ if I play $a_1{}'$ instead?

$a_2 \rightarrow a_2{}'$

Can I do better than $u_2(a_1, a_2)$ if I play $a_2{}'$ instead?



Currently playing a_1



Currently playing a_2

Expected payoff $= u_i(a_1, a_2)$

Pure Nash equilibrium (NE)

A pure strategy NE is a pair of actions (a_1, a_2) such that neither player is incentivized to *unilaterally deviate.*

deterministic

$$a_1 \in BR_1(a_2) \text{ AND } a_2 \in BR_2(a_1)$$

	Soccer	Shopping
Soccer	**8,4**	**0,0**
Shopping	**0,0**	**4,8**

Battle of the Sexes

(Soccer, Soccer) is NE

- If P1 plays shopping instead, 8 → 0
- If P2 plays shopping instead, 4 → 0

Why is (Soccer, Shopping) not NE?

NE captures idea of **stability**

Pure NE: “locally optimal”

Exercise: Finding Pure NE

Find **all the pure** NE in the Prisoner's Dilemma & Chicken Game

	Cooperate	Defect
Cooperate	**10,10**	**-1,12**
Defect	**12,-1**	**0,0**

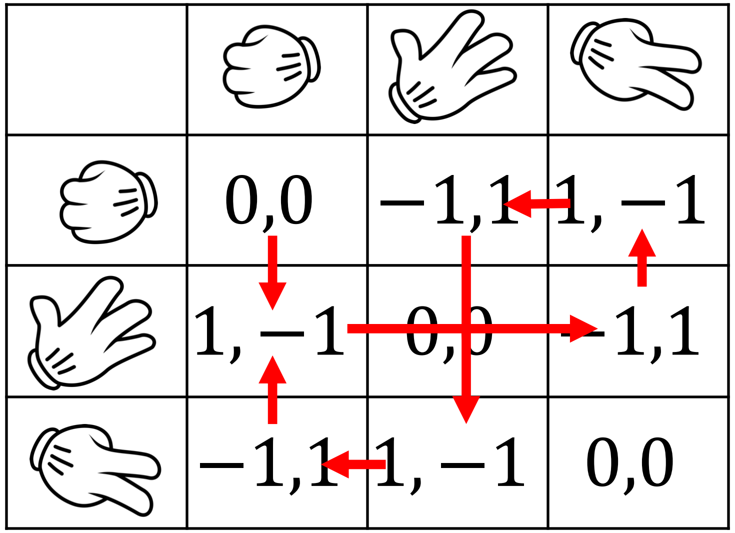
Prisoner's Dilemma

	Chicken	Dare
Chicken	**6,6**	**1,7**
Dare	**7,1**	**0,0**

Chicken Game

Pure NE may not exist!

No matter where you start, cycles occur



Mixed Strategy NE

A mixed strategy NE is a pair of distributions over actions (x, y) such that $x \in \Delta_1, y \in \Delta_2$ and neither player is incentivized to *unilaterally deviate.*

Definition extends to >2 players

Probability Simplex

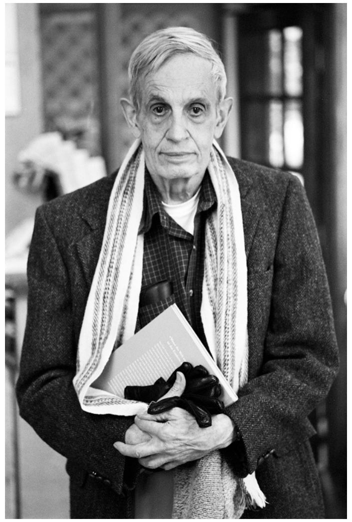
$$\Delta_i = \left\{ x \in \mathbb{R}_+^{|\mathcal{A}_i|} \,\middle|\, \sum_j^{|\mathcal{A}_i|} x_j = 1 \right\}$$

$$x \in BR_1(y) \text{ AND } y \in BR_2(x)$$

Includes pure strategy NE as a special case. Why?

Theorem: A mixed NE **always** exists!

Theorem by John Nash *Nash won the Nobel prize for this (amongst other) results



[Optional] Proof uses **Brouwer's Fixed Point Theorem**

compact: closed and bounded

If C is compact, convex and $f : C \to C$ is continuous then there exists c such that $f(c) = c$.

Sperner's Lemma $\xrightarrow{\text{Repeated divisions, Bolzano Weierstrass}}$ Brouwer's FP Theorem $\xrightarrow{\text{some transformations}}$ Nash's Theorem

Beautiful combinatorial theorem

Kakutani's FP Theorem $\xrightarrow{\text{follows directly}}$ Nash's Theorem

allows set functions

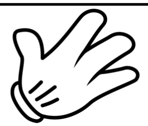
Sperner's Lemma + Brouwer's FP theorem are closely related to complexity of finding NE

Example of Mixed NE

Show that playing uniformly at random is a NE

- What is the payoff if both players play uniformly?
- After one player deviates (other frozen), what is the new payoff?

Show that playing uniformly at random is **the only** NE

			
	$0,0$	$-1,1$	$1,-1$
	$1,-1$	$0,0$	$-1,1$
	$-1,1$	$1,-1$	$0,0$

More Examples of NE

Two Pure NE. There is a 3rd mixed NE!

- #NE is almost always odd (Wilson's oddness theorem)

Can you find it?

	Soccer	
Soccer	**8,4**	**0,0**
Shopping	**0,0**	**4,8**

Battle of the Sexes

Also applies to Chicken Game.

- Are there other NE for Prisoner's dilemma?

Idea: opponent is *indifferent* to their actions

- Need to be careful, has caveats (later in this lecture)

Visualize NE on the board

Computational Complexity



Belongs to complexity class **PPAD**
- Polynomial Parity Arguments on Directed Graphs
- Daskalakis, Goldberg, Papadimitriou
- Chen and Deng

Not quite the same as NP-hard
- PPAD reductions are a bit different
- Deciding if *Nash exists* is doable in constant time (how?)
- Most decision variants are NP-hard (Conitzer & Sandholm, 2003)
  - Whether there exists Nash that contains a given action in support
  - Whether there exists Nash with or exceeding given social welfare

A common criticism of Nash, especially when used for modeling
- If people truly behave like NE, we could use it to compute "hard" problems
- "If your computer can't compute it, then humans shouldn't behave it"

[Optional] Elimination of Dominated Strategies

Sometimes, we get lucky and can trim the game by throwing away "obviously bad actions"

	Soccer	Shopping
Soccer	**8,4**	**0,0**
Shopping	**0,0**	**4,8**
Attend class	**-1,(.)**	**-1,(.)**

Battle of the Nerdy Sexes

Cooperating is a dominated strategy!

10,10	**-1,12**
12,-1	**0,0**

Prisoner's Dilemma

Will you ever attend classes?

- Playing soccer is always better for P1, regardless of what P2 chooses
- P2's payoff **doesn't matter**

Weak dominance: equality sometimes

Dominance by mixed strategies

Methods to find NE in general matrix games

assuming 2 players

Brute force

- **Support enumeration**
- Vertex enumeration

Can obtain **all** NE

Homotopy type methods

Interesting but beyond scope of this class

Lemke Howson Algorithm

Linear Complementary Programs/Integer Programming

My preferred method for quick hacks, allows to select Nash based on social welfare

Remarks about NE

NE in general-sum games always come in **pairs** (for 2p games)

Not exchangeable

- $(x, y), (x', y')$ are NE does not imply $(x, y'), (x', y)$ are NE

Multiple NE can occur. Payoffs can vary wildly.

- Choosing which NE is "preferred" is known as **equilibrium selection**

Price of Anarchy

- Compared to a dictator who could "force" players behavior, how badly does this "free market" perform in terms of welfare?

Many special cases: potential games, graphical games, routing games etc. which can be solved more easily, have nice properties

[Optional] Bounded Rationality

Quantal Response Equilibrium

- Recall for NE: $x \in BR_1(y)$ AND $y \in BR_2(x)$
  - Can be a bit brittle!
  - x, y is some kind of argmax over expected utilities (more on this later)
- Replace argmax by **softmax** for both players
- Equivalent to adding Gumbel noise to all utilities and playing according to the probability that an action is the best
- Always exists, converges to a NE as temperature → 0

Level-k

- NE is some kind of “infinite” nesting of beliefs
- E.g.: population contains different types of people, some think many “steps” ahead, others few. Common belief over distribution over levels
- http://www.columbia.edu/~md3405/Behave_Bounded_6_15.pdf

Representing NE in Matrix form

Preliminaries and Definitions

$$n = |\mathcal{A}_1|, m = |\mathcal{A}_2|$$

$$A, B \in \mathbb{R}^{n\times m}$$

A_{ij}, B_{ij}: payoff given actions i, j

8,4	**0,0**
0,0	**4,8**

Battle of the Sexes

$$\longrightarrow \underset{A}{\begin{bmatrix} 8 & 0 \\ 0 & 4 \end{bmatrix}} \qquad \underset{B}{\begin{bmatrix} 4 & 0 \\ 0 & 8 \end{bmatrix}}$$

P1's strategy $x \in \Delta_n$

$$x \in \mathbb{R}^n, \;\; x \geq 0, \;\; \sum_i x_i = 1$$

P2's strategy $y \in \Delta_m$

$$y \in \mathbb{R}^m, \;\; y \geq 0, \;\; \sum_j y_j = 1$$

1. [Soccer = 0.4, Shopping=0.6] ✓
2. [Soccer=-0.1, Shopping=1.1] ✗
3. [Soccer=0.3, Shopping=0.6] ✗

Some Useful Terms

$$A, B \in \mathbb{R}^{n \times m}$$

*Slight abuse of notation: I am treating x, y as column vectors

$$\underbrace{x^T A y = \sum_i \sum_j \overbrace{x_i \cdot y_j}^{\text{Probability that outcome } (i,j) \text{ occurs}} \cdot A_{ij}}_{\text{Row player's expected payoff under } x, y}$$

Row player's expected payoff under x, y

$$\underbrace{Ay}_{\in \mathbb{R}^{n \times 1}} = \begin{bmatrix} (Ay)_1 \\ (Ay)_2 \\ \dots \\ (Ay)_n \end{bmatrix} = \begin{bmatrix} \sum_j A_{1j} y_j \\ \sum_j A_{2j} y_j \\ \dots \\ \sum_j A_{nj} y_j \end{bmatrix}$$

i-th row: How much row player would have gotten if by playing the i-th action assuming other player plays y

*transposing gives similar interpretation for the other player $B^T x$

Best response condition

If x, y are mixed strategies, then $x \in BR_1(y)$ **if and only if**

$$x_i > 0 \Rightarrow (Ay)_i = L = \max\{Ay\}$$

Why?

- $x^T Ay = \sum_{i\in[m]} x_i \cdot (Ay)_i = \sum_{i\in[m]} x_i \cdot (L - (L - (Ay)_i)) = L - \sum_{i\in[m]} \underbrace{x_i \cdot (L - (Ay)_i)}$
- $x^T Ay \leq L$ if and only if $x_i \geq 0$ and $(L - (Ay)_i \geq 0$ for all $i \in [m]$
- $x^T Ay = L$ if and only if $x_i > 0 \Rightarrow L - (Ay)_i$

*both terms are nonnegative

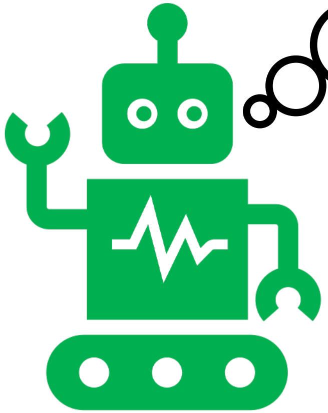
Back to Unilateral Deviations

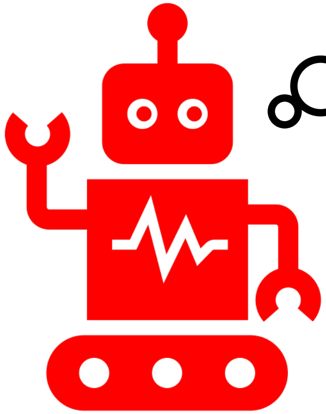
$x \to x'$

Can I do better than $x^T A y$ if I play x' instead?

$y \to y'$

Can I do better than $x^T B\ y$ if I play y' instead?





Currently playing x

Currently playing y

Expected payoff $= x^T\ A\ y, x^T B y$

Nash Equilibrium written explicitly

NE is a pair $x \in \Delta_n,\ y \in \Delta_m$ such that

*Expected payoffs given strategies x, y

*Infinitely many constraints!

$x^T A\, y \geq x'^T A\, y \qquad \forall x' \in \Delta_n$ P1 cannot do any better by changing strategies

$x^T B\, y \geq x^T B\, y' \qquad \forall y' \in \Delta_m$ P2 cannot do any better by changing strategies

$\Updownarrow$ Players cannot unilaterally deviate and perform better

* Best response condition

$x^T A\, y \geq e_i A\, y \qquad \forall i \in [n]$

$x^T B\, y \geq x^T B\, e_j \qquad \forall j \in [m]$

AND $\quad x \in \Delta_n,\ y \in \Delta_m$

elementary basis vector

Feasibility problem. Can we just plug this into some generic solver?

Solving NE via Integer Programming

How does a NE look like?

Suppose x^*, y^* is a NE. Recall best-response condition

There exists some L such that

$$Ay^* = \begin{bmatrix} < L \\ L \\ L \\ < L \\ \cdots \\ < L \\ L \end{bmatrix}$$

not in support of x^*

In support of x^*

Writing this in matrix form

$$L \geq e_i A y^* \qquad \forall i \in [n]$$

$$x_i^* \cdot (L - (Ay^*)_i) = 0 \qquad \forall i \in [n]$$

$$\Updownarrow$$

$${x^*}^T (L - Ay^*) = 0$$

Same holds for $B^T x^*$ and support of y^*

Rewriting as an LCP

$$x^T A\, y \geq e_i A\, y \qquad \forall i \in [n]$$
$$x^T B\, y \geq x^T B\, e_j \qquad \forall j \in [m]$$

AND $\quad x \in \Delta_n,\ y \in \Delta_m$

$$\Updownarrow$$

Find $L, K \in \mathbb{R}, x \in \Delta_n, y \in \Delta_m$ such that

$$L \geq e_i^T A y \quad \forall i \in [n]$$
$$K \geq x^T B e_j \quad \forall j \in [m]$$
$$\left.\begin{aligned} x^T(L - Ay) &= 0 \\ y^T(K - B^T x) &= 0 \end{aligned}\right\}$$

Complementary conditions: at most one term in product strictly positive

Not quite accessible yet

A simple integer program formulation

$$\text{Find } \overbrace{L, K \in \mathbb{R}}^{\text{scalar payoffs to each player at equilibrium}}, x \in \Delta_n, y \in \Delta_m, \overbrace{w \in \{0,1\}^n, z \in \{0,1\}^m}^{\text{binary variables indicating where support is}} \text{ such that}$$

$$L \geq e_i^T A y \quad \forall i \in [n]$$

$$K \geq x^T B e_j \quad \forall j \in [m]$$

$$L - Ay \leq M \cdot (1 - w)$$

$$K - B^T x \leq M \cdot (1 - z)$$

"big-M" constraint

$$\left.\begin{aligned} w &\geq x \\ z &\geq y \end{aligned}\right\} \; x_i \text{ positive only where } w_i = 1$$

for some big enough M.

Feasibility problem: can optimize for "good" equilibrium in objective!

Sandholm, Gilpin and Conitzer (https://www.cs.cmu.edu/~sandholm/MIPNash.aaai05.pdf)

Support Enumeration

"When in doubt, use brute force"

-Ken Thompson

Support Enumeration

Assume that game is non-degenerate

- Every mixture of k-strategies can only have at most k pure best-responses
- Reasonable for randomly generated games
  - Adding noise to payoffs makes games non-degenerate
  - Not reasonable for certain types of structured games!

If (x^*, y^*) is Nash, then x^*, y^* have equal sized supports

- Follows from best-response condition

Algorithm Sketch:

- Iterate for all $k = 1, \ldots \min(n, m)$
  - Enumerate all k-sized subsets I of [n] and J of [m]
  - Find some mixture in I such that P2 is indifferent to their actions in J
  - Find some mixture in J such that P1 is indifferent to their actions in I
  - Both above are obtained by **solving system of linear equations**

Example: Chicken Game

	Chicken	Dare
Chicken	**6,6**	**1,7**
Dare	**7,1**	**0,0**

For $k = 1$

- Trivial, joint supports correspond to pure strategy profiles
- Only Dare-Chicken and Chicken-Dare are equilibria here

For $k = 2$ (full support)

- Player 1 needs to be indifferent to both of player 2's actions
- If P1 chickens, get $6 \cdot y_{chicken} + 1 \cdot y_{dare}$, if P1 dares, get $7 \cdot y_{chicken} + 0 \cdot y_{dare}$
- These must be equal (why?). Also, y needs to be a probability distribution
- $6 \cdot y_{chicken} + 1 \cdot y_{dare} = 7 \cdot y_{chicken} + 0 \cdot y_{dare}$ AND $y_{chicken} + y_{dare} = 1$
- Solve system of 2 equations and 2 unknowns, unique solution (non-degenerate)
  - $y = [0.5, 0.5]$
- Repeat the same for player 2 indifference to player 1's action to get x
- CHECK that candidate Nash's are both valid distributions and satisfy Nash
  - In the case of full support this has already been done "automatically" (why?)

Remarks

Common Mistakes

- Assuming that P1's mixture is indifferent to **all** actions of P2
  - Guessing the right support is important!
  - Remember how Ay^* and $A^T x^*$ look like: not necessarily all same values
  - Counterexample: dominated actions
- Not verifying if solutions to linear system are NE of original game
  - valid distributions alone **does not imply** NE
  - Counterexample: choose support size of $k = 1$ for any reasonable game e.g., rock paper scissors

Support enumeration gets all Nash for nondegenerate games

- Common theme to "convert" linear inequalities to equalities, become system of linear equations

Downside: exponentially many supports to guess

Lemke-Howson Algorithm

Worst-case brute force can still be very insightful…

Outline

Algorithm to find **one** NE, not all, or not special ones
- Key idea: try to exploit combinatorial/geometric structure of NE

Assumptions:
- A, B are strictly positive
- Use symmetric variant
  - Consider symmetric game with payoff matrix $\begin{bmatrix} 0 & A \\ B^T & 0 \end{bmatrix}$
  - Symmetric games have at least one symmetric equilibrium. Prove using fixed point theorem.
  - Let symmetric eqm of new game be of the form $[x,^* \ y^*]$. Then (x^*, y^*) is unnormalized NE to original game
    - Prove it! Be careful, need to prove x^*, y^* in new game are not 0
- Game is non-degenerate (in some sense)

All these assumptions can be relaxed in practice

Best response polytope

Let A be the symmetric matrix (from previous slide)

- Let n be the size of the symmetric matrix (n+m from the original game)

Consider $\mathrm{P} = \{\mathrm{z} | Az \leq 1, z \geq 0\}$ for some $z \in R^n$

- 2n inequalities, n from $Az \leq 1$ and n from $z \geq 0$
- An inequality is tight if it holds with **equality**
- For given z, we say that an action i is represented if either $(Az)_i = 1$ OR $z_i = 0$

If $z \neq 0$ is such that every action i is represented, then

- $x \in R^n$ where $x_i = z_i / \sum_i z_i$ is a symmetric NE
- Why? Represented ⇒ best response condition

Common trick to normalize (remove value of the game)

- Unnormalized version $\mathrm{P} = \{\mathrm{z}, \mathrm{L} | Az \leq L, z \geq 0, |z|_1 = 1\}$

Now, problem is to find such z where all actions are represented

Vertices of P

Type equation here. Assume system of inequalities is nondegenerate

- Vertices in polytope are intersection of exactly n hyperplanes, i.e., n out of $2n$ inequalities hold with **equality**

n equalities (✓)→single point!

	1	**2**	**3**	**4**	**5**	**…**	**n**
$(Az)_i = 1$	✓				✓		✓
$z_i = 0$		✓	✓	✓			

NE since every action is represented

	1	**2**	**3**	**4**	**5**	**…**	**n**
$(Az)_i = 1$	✓			✓			
$z_i = 0$		✓	✓	✓			✓

Not NE, not every action is represented

Finding NE by pivoting

Trick: incrementally "improve" support set by pivoting

- Maintain set of n tight inequalities (out of $2n$)
- Stick to *almost-Nash set*, all actions except possibly one distinguished one (say the n-th one*)* are represented
- Non-Nash Almost-Nash sets have exactly one "doubly represented" action

	1	**2**	**3**	**4**	**5**	**…**	**n**
$(Az)_i = 1$	✓			✓	✓	✓	
$z_i = 0$		✓	✓	✓			

Start from almost-Nash set. Take doubly represented action, remove one equality→system of equations is now a line

- Line is part of an edge in P, walk along that line to a new vertex, keep repeating until all actions are represented→Nash!

Need to start at an almost-Nash set that is a vertex in P. How?

- Use the all 0's vector! (artificial equilibrium)

Illustration

	1	2	3	4	5	…	n
$(Az)_i = 1$							
$z_i = 0$	✓	✓	✓	✓	✓	✓	✓

Not NE, 0 is not a valid unnormalized strategy

	1	2	3	4	5	…	n
$(Az)_i = 1$							
$z_i = 0$	✓	✓	✓	✓	✓	✓	

Remove equality involving action n

	1	2	3	4	5	…	n
$(Az)_i = 1$				✓			
$z_i = 0$	✓	✓	✓	✓	✓	✓	

"walk" along edge to get new vertex

Repeat until all actions represented

Visualization

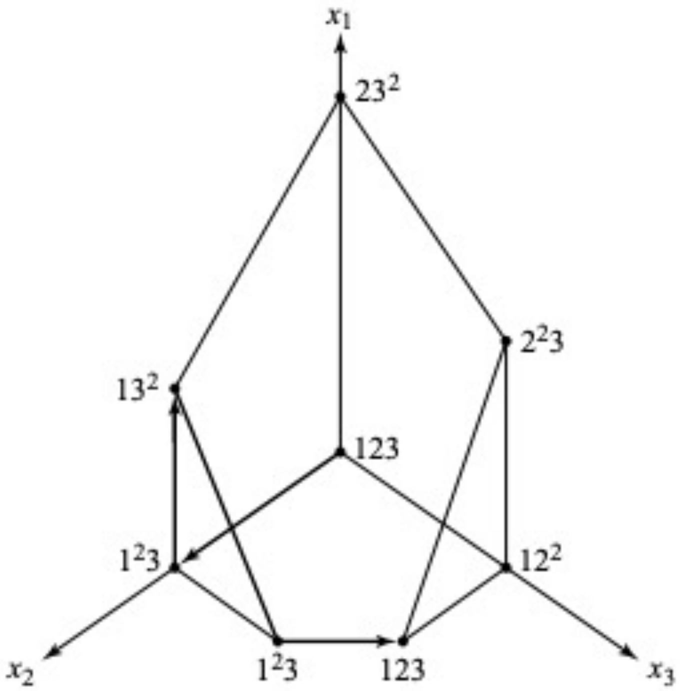


Figure 2.1. The Lemke–Howson algorithm can be thought of as following a directed path in a graph.

Source: Algorithmic Game Theory

Remarks

Only one "option" of equality to add in

Walking determistically along a path!

- Cannot have loops, can never return to 0
- Paths cannot cross itself (why?)

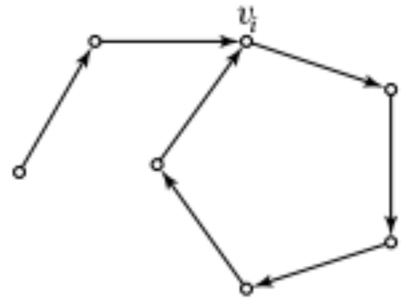


Figure 2.2. The path cannot cross itself.

Always finds solution since finite (large) #vertices

- But can take worst case exponential time

Gives a **constructive proof** of existence of equilibrium (recall Nash's theorem is existential)

Source: Algorithmic Game Theory

Some additional structure

Almost-Nash vertices form forest of paths and cycles

- Source and sinks are Nash, including the all-0 artificial Nash (standard source)
- Implies Wilson's theorem for symmetric games! (assuming nondegeneracy)

Lemke Howson is the starting point for proving PPAD-hardness

- Closely linked to Brouwer's fixed point theorem and reminiscent of Sperner's Lemma (see me after class if you want to learn more)

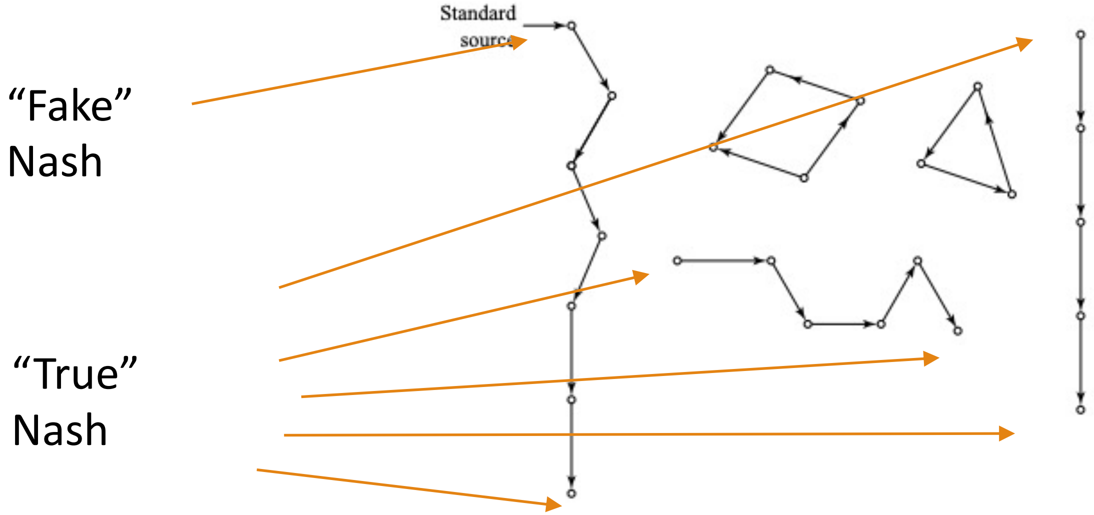


Figure 2.3. A typical problem in PPAD.

Source: Algorithmic Game Theory

Tools to compute NE

Gambit: https://www.gambit-project.org/

- Mainly useful for academic reasons, implements most classical algorithms

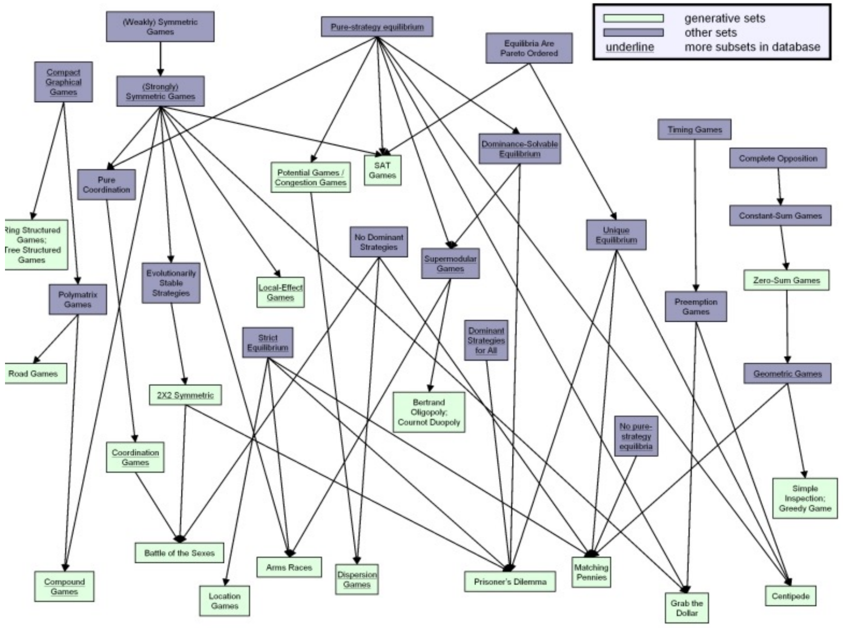
Gamut: http://gamut.stanford.edu/

- Suit of game generators for testing algorithms

Nashpy: https://nashpy.readthedocs.io/en/stable/

Some other academic libraries out there

- Free online solver by Rahul Savani: https://cgi.csc.liv.ac.uk/~rahul/bimatrix_solver/



End of Lecture