ECONOMICS & COMPUTATION

Spring 2025 | Lecture 2

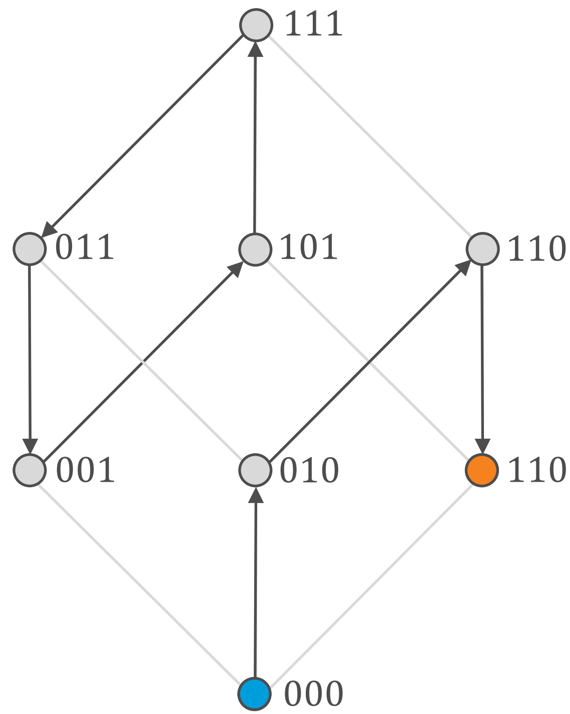
Equilibrium Computation

Ariel Procaccia | Harvard University

END OF THE LINE

- In the End of the Line problem, the input is a directed graph $G = (V, E)$ with $V = \{0,1\}^n$, where every vertex has at most one predecessor and at most one successor
- The edges E are implicitly given by a polynomial-time-computable functions $f_p: \{0,1\}^n \to \{0,1\}^n$ and $f_s: \{0,1\}^n \to \{0,1\}^n$ that return the predecessor and successor of a given vertex (if they exist)
- Given a source vertex (no predecessor), the task is to find a sink (no successor)

END OF THE LINE



For any input to END OF THE LINE, the existence of a sink vertex is guaranteed — but how do you find it?

THE PPAD CLASS

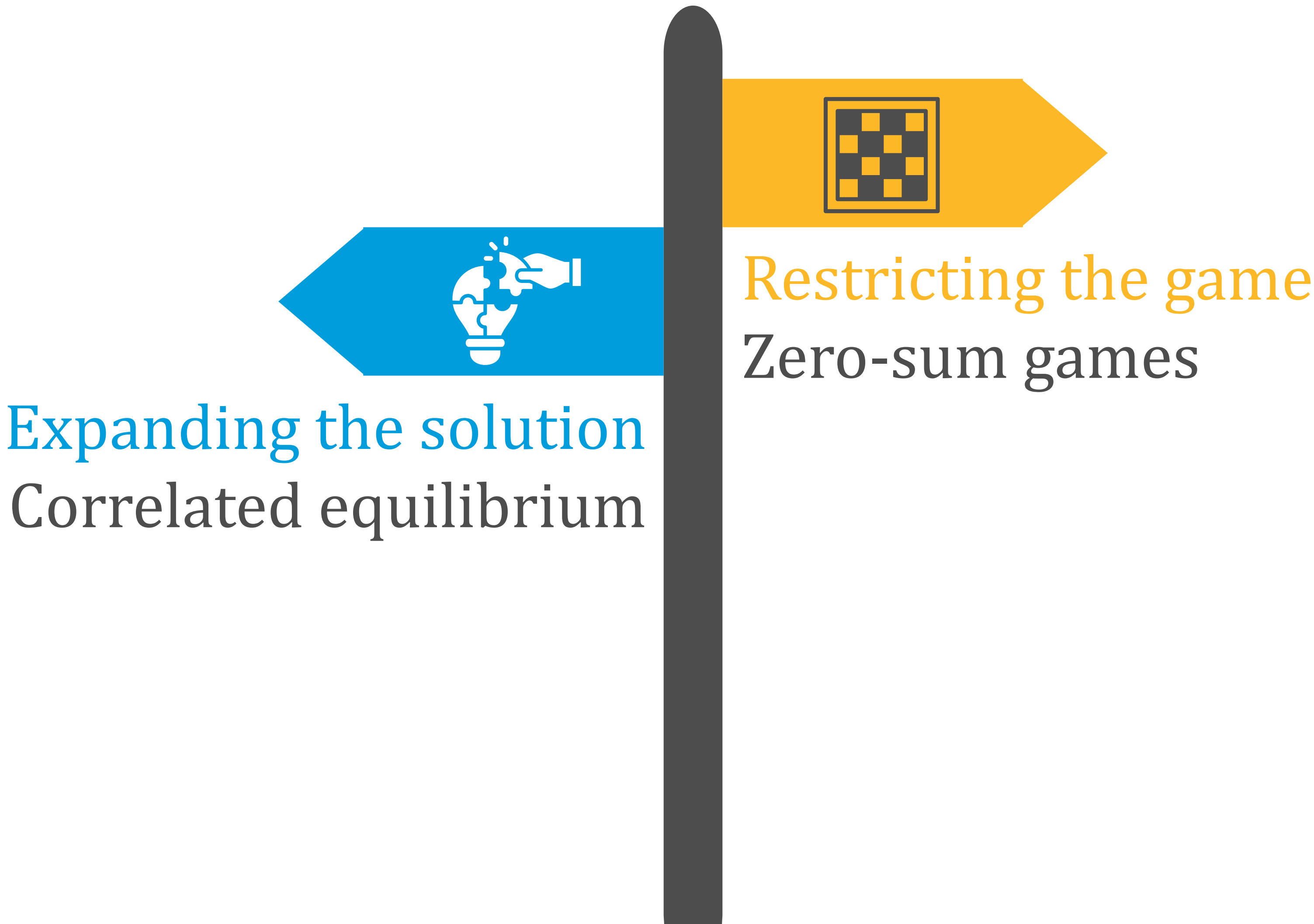
- The complexity class TFNP (total function NP) includes problems that are guaranteed to have a solution, and this solution can be checked in polynomial time
- The complexity class PPAD (polynomial parity arguments on directed graphs) includes all problems in TFNP that have polynomial-time reductions to END OF THE LINE
- Theorem: For all $n \geq 2$, computing an (approximate) Nash equilibrium in an n-player normal-form game is PPAD-complete

Christos Papadimitriou

1949–

Influential theoretical computer scientist and a founder of algorithmic game theory. Also known for not naming PPAD after himself.

WHERE TO GO FROM HERE?



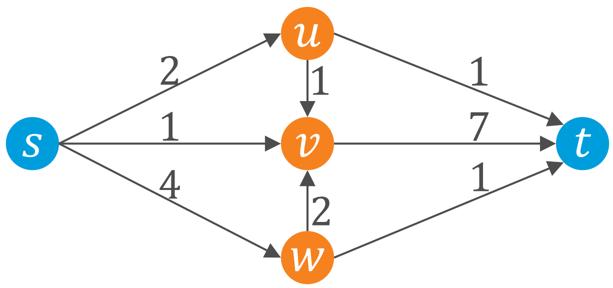
INTERLUDE: LINEAR PROGRAMMING

- Linear programming:

$$\min_{\boldsymbol{x}} \boldsymbol{c}^T \boldsymbol{x}$$
$$\text{s.t. } A\boldsymbol{x} = \boldsymbol{a}$$
$$B\boldsymbol{x} \leq \boldsymbol{b}$$

  where $\boldsymbol{x} \in \mathbb{R}^n$ is the optimization variable, and $\boldsymbol{c} \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, \boldsymbol{a} \in \mathbb{R}^m, B \in \mathbb{R}^{k \times n}, \boldsymbol{b} \in \mathbb{R}^k$ are the problem data
- Linear programs can be solved in polynomial time using interior-point methods

INTERLUDE: LINEAR PROGRAMMING



- In the max flow problem, we are given a directed graph $G = (V, E)$ with a source s and a sink t, and a capacity α_{xy} for each $(x, y) \in E$
- A flow is a function $f: E \to \mathbb{R}^+$ that satisfies $f_{xy} \leq \alpha_{xy}$ for all $(x, y) \in E$, and for all $x \in V \setminus \{s, t\}$, $\sum_{(y,x) \in E} f_{yx} = \sum_{(x,z) \in E} f_{xz}$
- The value of a flow is $\sum_{(s,x) \in E} f_{sx}$
- In the above example, the value of the max flow is 6

INTERLUDE: LINEAR PROGRAMMING

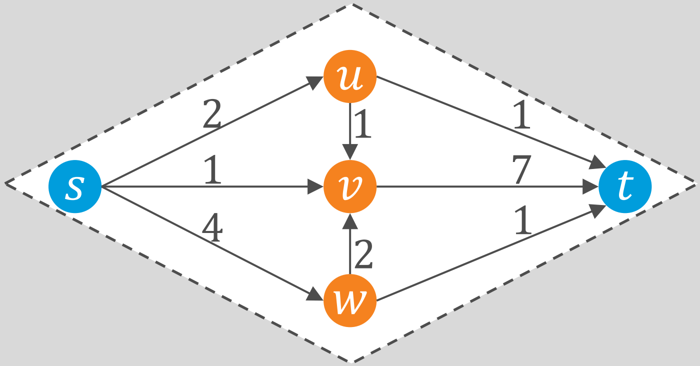
$$\min_{\boldsymbol{x}} \boldsymbol{c}^T \boldsymbol{x}$$
$$\text{s.t. } A\boldsymbol{x} = \boldsymbol{a}$$
$$B\boldsymbol{x} \leq \boldsymbol{b}$$

How does the canonical LP form fit with the max flow example?

$$\max f_{su} + f_{sv} + f_{sw}$$

s.t.

$f_{s,u} \leq 2$	$f_{s,u} \geq 0$	$f_{s,u} = f_{uv} + f_{ut}$
$f_{s,v} \leq 1$	$f_{s,v} \geq 0$	$f_{s,v} + f_{uv} + f_{wv} = f_{vt}$
$f_{s,w} \leq 4$	$f_{s,w} \geq 0$	$f_{sw} = f_{wv} + f_{wt}$
$f_{uv} \leq 1$	$f_{uv} \geq 0$	
$f_{wv} \leq 2$	$f_{wv} \geq 0$	
$f_{u,t} \leq 1$	$f_{u,t} \geq 0$	
$f_{v,t} \leq 7$	$f_{v,t} \geq 0$	
$f_{w,t} \leq 1$	$f_{w,t} \geq 0$	



TWO-PLAYER ZERO-SUM GAMES

- In two-player zero-sum games, it holds that for every strategy profile $\boldsymbol{s}$,
$$u_1(\boldsymbol{s}) = -u_2(\boldsymbol{s})$$
- Maximin (randomized) strategy of player 1 is $x_1^\star \in \arg\max_{x_1 \in \Delta(S)} \min_{s_2 \in S} u_1(x_1, s_2)$
- Minimax (randomized) strategy of player 2 is $x_2^\star \in \arg\min_{x_2 \in \Delta(S)} \max_{s_1 \in S} u_1(s_1, x_2)$

ZERO-SUM GAMES

$-1, 1$	$2, -2$
$2, -2$	$-2, 2$

Poll 1

Denote $x_1^\star = (p, 1-p)$. What is p?

- ○ 4/7
- ○ 3/5
- ○ 5/8
- ○ 8/9

MAXIMIN AS LP

Maximin strategy is computed via LP (and minimax strategy is computed analogously):

$$\max w$$

$$\text{s.t.} \quad \forall s_2 \in S, \sum_{s_1 \in S} x(s_1) u_1(s_1, s_2) \geq w$$

$$\sum_{s_1 \in S} x(s_1) = 1$$

$$\forall s_1 \in S, x(s_1) \geq 0$$

John von Neumann

1903–1957

A founder of game theory. Also known for revolutionary contributions to mathematics, physics, computer science and the Manhattan Project.

THE MINIMAX THEOREM

- Theorem [von Neumann 1928]: Every 2-player zero-sum game has a unique value v such that:
  - Player 1 can guarantee utility at least v
  - Player 2 can guarantee utility at least $-v$
- Proof (via Nash's Theorem):
  - Let (x_1, x_2) be a Nash equilibrium and denote $v = u_1(x_1, x_2)$
  - For every $s_2 \in S_2$, $u_1(x_1, s_2) \geq v$, so player 1 can guarantee utility at least v by playing x_1
  - For every $s_1 \in S_1$, $u_2(s_1, x_2) \geq -v$, so player 2 can guarantee utility at least $-v$ by playing x_2 ■
- We will prove the theorem from scratch later in the course

CORRELATED EQUILIBRIUM

- Let $N = \{1,2\}$ for simplicity
- A mediator chooses a pair of strategies (s_1, s_2) according to a distribution p over S^2
- Reveals s_1 to player 1 and s_2 to player 2
- When player 1 gets $s_1 \in S$, they know that the distribution over strategies of 2 is

$$\Pr[s_2 | s_1] = \frac{\Pr[s_1 \wedge s_2]}{\Pr[s_1]} = \frac{p(s_1, s_2\ )}{\sum_{s_2' \in S} p(s_1, s_2')}$$

CORRELATED EQUILIBRIUM

- Player 1 is best responding if for all $s_1' \in S$

$$\sum_{s_2 \in S} \Pr[s_2 | s_1]\, u_1(s_1, s_2) \geq \sum_{s_2 \in S} \Pr[s_2 | s_1]\, u_1(s_1', s_2)$$

- Equivalently,

$$\sum_{s_2 \in S} p(s_1, s_2) u_1(s_1, s_2) \geq \sum_{s_2 \in S} p(s_1, s_2) u_1(s_1', s_2)$$

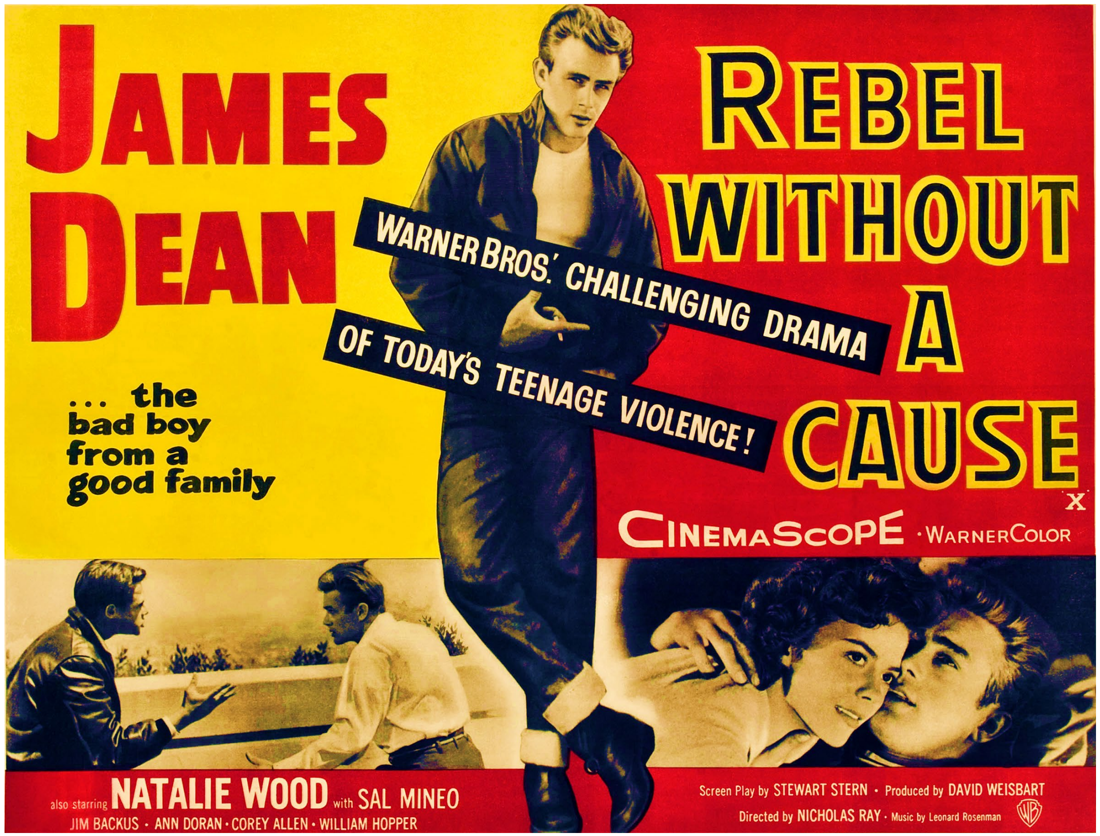
- p is a correlated equilibrium (CE) if both players are best responding

Poll 2

What is the relation between correlated equilibrium and Nash equilibrium?

- ○ CE ⊆ NE
- ○ NE ⊆ CE
- ○ Neither

GAME OF CHICKEN



http://youtu.be/u7hZ9jKrwvo

GAME OF CHICKEN

	Dare	Chicken
Dare	0,0	4,1
Chicken	1,4	3,3

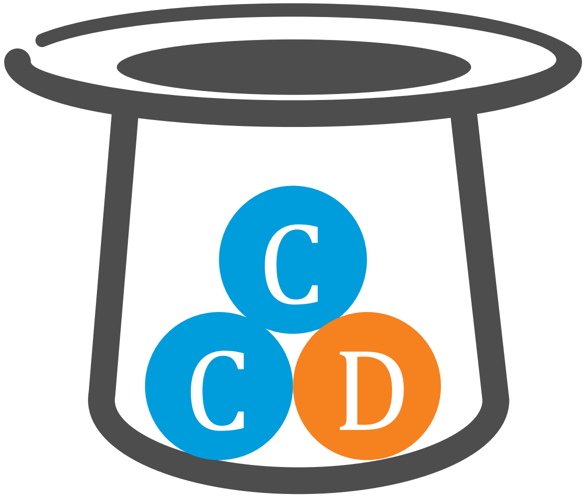
- Social welfare is the sum of utilities
- Pure NE: (C,D) and (D,C), social welfare = 5
- Mixed NE: both (1/2,1/2), social welfare = 4
- Optimal social welfare = 6

GAME OF CHICKEN

	Dare	Chicken
Dare	0,0	4,1
Chicken	1,4	3,3

- Correlated equilibrium: (D,D) played with probability 0, (D,C) with probability 1/3, (C,D) with probability 1/3, and (C,C) with probability 1/3
- Social welfare of CE = 16/3

IMPLEMENTATION OF CE



To implement the mediator, simply put two "chicken" balls and one "dare" ball in a hat, and have each blindfolded player pick a ball

CE AS LP

Can compute CE via linear programming in polynomial time!

find $\forall s_1, s_2 \in S, p(s_1, s_2)$

s.t. $$\forall s_1, s_1' \in S, \sum_{s_2 \in S} p(s_1, s_2) u_1(s_1, s_2) \geq \sum_{s_2 \in S} p(s_1, s_2) u_1(s_1', s_2)$$

$$\forall s_2, s_2' \in S, \sum_{s_1 \in S} p(s_1, s_2) u_2(s_1, s_2) \geq \sum_{s_1 \in S} p(s_1, s_2) u_2(s_1, s_2')$$

$$\sum_{s_1, s_2 \in S} p(s_1, s_2) = 1$$

$$\forall s_1, s_2 \in S, p(s_1, s_2) \in [0,1]$$