Game Theory

Enrico Franchi (efranchi@ce.unipr.it)

Game Theory by Enrico Franchi is licensed under aCreative Commons Attribution-ShareAlike 3.0 Unported License.
Based on a work at https://github.com/rik0/game-theory-slides.

Introduction

Game theory can be defined as the study of mathematical models of conflict and cooperation between intelligent rational decision-makers.

R. Myerson

Rationality and Intelligence

- A decision-maker is rational if he makes decisions consistently in pursuit of his own objectives
- A player is intelligent if he knows everything that we know about the game and he can make inferences about the situation that we can make

Outcomes

- Let O be a finite set of outcomes
- A *lottery* is a probability distribution over O

$$l = [p_1 : o_1, \ldots, p_k : o_k]$$

$$o_i \in O \quad p_i \in [0,1]$$

$$\sum_{i=1}^{k} p_i = 1$$

- We assume agents can rank outcomes and lotteries with a *utility function*

TCP Game

	C	D
C	-1,-1	-4,0
D	0,-4	-3,-3

D: defective implementation

C: correct implementation

TCP Game

Prisoner's Dilemma

	C	D
C	-1,-1	-4,0
D	0,-4	-3,-3

D: defective implementation

C: correct implementation

Game in Normal Form

- A finite n-person *normal form* game is a tuple (N, A, u) where
  - N is a finite set of n players
  - $A = A_1 \times \ldots \times A_n$, where A_i is a finite set of actions available to player i
  - $u = (u_1, \ldots, u_n)$, where $u_i : A \mapsto \mathbf{R}$ is a real valued utility function
- $a = (a_1, \ldots, a_n)$ is an *action profile*

TCP Game (again)

	C	D
C	-1,-1	-4,0
D	0,-4	-3,-3

- $N=\{1,2\}$
- $A=\{C,D\}\times\{C,D\}$

A_1	A_2	u_1	u_2
C	C	-1	-1
C	D	-4	0
D	C	0	-4
D	D	-3	-3

TCP Game (again)

A_1	C	D
C	-1,-1	-4,0
D	0,-4	-3,-3

- $N=\{1,2\}$
- $A=\{C,D\}\times\{C,D\}$

A_1	A_2	u_1	u_2
C	C	-1	-1
C	D	-4	0
D	C	0	-4
D	D	-3	-3

TCP Game (again)

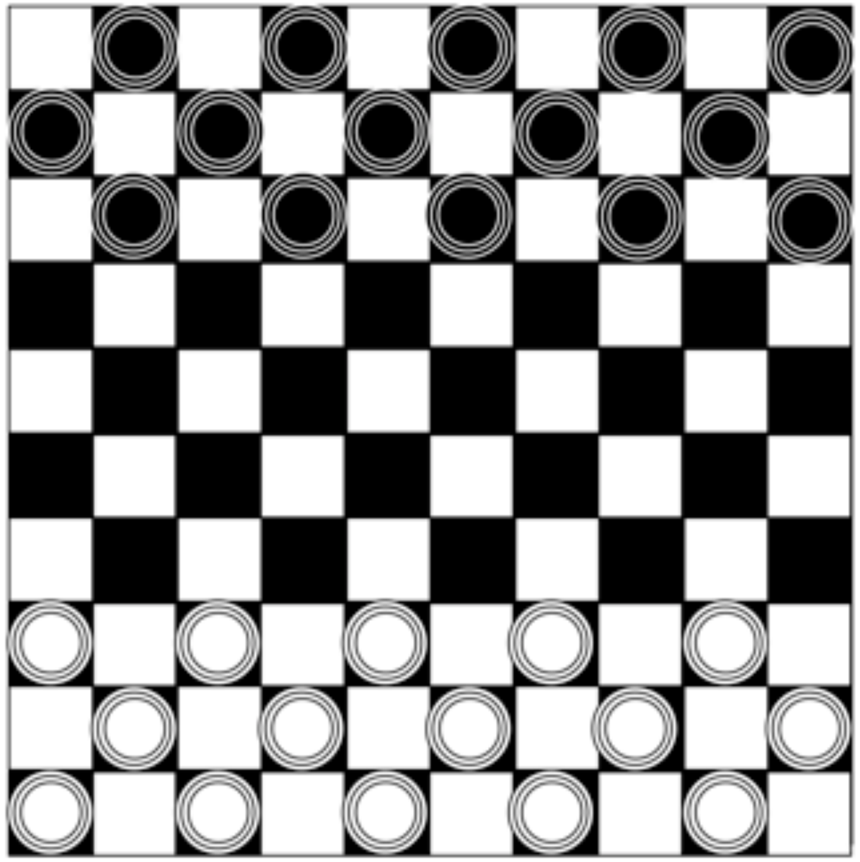
		A_2: C	D
A_1	C	-1,-1	-4,0
	D	0,-4	-3,-3

- $N=\{1,2\}$
- $A=\{C,D\}\times\{C,D\}$

A_1	A_2	u_1	u_2
C	C	-1	-1
C	D	-4	0
D	C	0	-4
D	D	-3	-3

Actions

- Actions can be "arbitrarily complex"
- Ex.: international draughts
  - an action is not a *move*
  - an action *maps* every possible board configuration to the move to be played if the configuration occurs



$2 \cdot 10^{22}$

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

Rock-Paper-Scissor

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

Strategy

- A *pure strategy* is selecting an action and playing it
- A *mixed strategy* for player i is an element of the set $S_i = \Pi(A_i)$ of probability distributions over A_i
- The *support* of a mixed strategy is the set of pure strategies $\{a_i \mid s_i(a_i) > 0\}$
- The set of *mixed-strategy profiles* is $S_1 \times \ldots \times S_n$ and a *mixed strategy profile* is a tuple $(s_1, \ldots, s_n)$
- The *utility* of a mixed strategy profile is

$$u_i(s) = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j)$$

Solution Concept

- Games are complex, the environment can be stochastic, other player's choices affect the outcome
- Game theorists study certain subsets of outcomes that are interesting in one sense or another which are called **solution concepts**

Pareto Efficiency

- The strategy profile s *Pareto dominates* the strategy profile s' if for some players the utility for s is strictly higher and for the others is not worse
- A strategy profile is *Pareto optimal* if there is no other strategy profile dominating it

Nash Equilibrium

- Player's i *best response* to the strategy profile s_{-i} is a mixed strategy $s^*_i \in S_i$ such that $u_i(s^*_i, s_{-i}) \geq u_i(s_i, s_{-i})$ for every $s_i \in S_i$
- A strategy profile $s=(s_1,\ldots,s_n)$ is a Nash equilibrium if, for all agents , s_i is a best response to s_{-i}
- Weak Nash (≥), Strong Nash (>)

Battle of the Sexes

	LW	WL
LW	2,1	0,0
WL	0,0	1,2

- Both pure strategies are Nash Equilibria
- Are there any other Nash Equilibria?
- There is at least another mixed-strategy equilibrium (usually very tricky to compute, but can be done with simple examples)

Battle of the Sexes

	LW	WL
LW	2,1	0,0
WL	0,0	1,2

Being "indifferent" means obtaining the same utility, not playing indifferently

- Suppose the husband chooses LW with probability p and WL with probability p-1
- The wife should be indifferent between her available options, otherwise she would be better off choosing a pure strategy
- What is the p which allows the wife to be really indifferent?
- Please notice that the pure strategies are Pareto optimal

Battle of the Sexes

p: probability that husband plays LW

$$U_{\text{wife}}(\mathrm{LW}) = U_{\text{wife}}(\mathrm{WL})$$

$$1 \cdot p + 0 \cdot (1-p) = 0 \cdot p + 2 \cdot (1-p)$$

$$p = 2 - 2p \qquad p = \frac{2}{3}$$

r: probability that wife plays LW

$$U_{\text{husband}}(\mathrm{LW}) = U_{\text{husband}}(\mathrm{WL})$$

$$2 \cdot r + 0 \cdot (1-r) = 0 \cdot r + 1 \cdot (1-r)$$

$$2r = 1 - r \qquad r = \frac{1}{3}$$

$$U_w(s) = 2(1-p)(1-r) + pr$$

$$U_w\left(s_w(r), \frac{2}{3}\mathrm{LW} + \frac{1}{3}\mathrm{WL}\right) = \frac{2}{3}(1-r) + \frac{2}{3}r = \frac{2}{3}$$

$$U_h(s) = (1-p)(1-r) + 2pr$$

$$U_h\left(s_h(p), \frac{1}{3}\mathrm{LW} + \frac{2}{3}\mathrm{WL}\right) = \frac{2}{3}(1-p) + \frac{2}{3}p = \frac{2}{3}$$

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

- Do we have any pure strategies?

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

- Do we have any pure strategies?
    - No

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

- Do we have any pure strategies?
    - No
- Do we have mixed strategies?

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

- Do we have any pure strategies?
  - No
- Do we have mixed strategies?
  - Yes

Matching Pennies

	Heads	Tails
Heads	1,-1	-1,1
Tails	-1,1	1,-1

- Do we have any pure strategies?
  - No
- Do we have mixed strategies?
  - Yes

$$U_1(\mathrm{H}) = U_1(\mathrm{T})$$

$$1 \cdot p + (-1) \cdot (1-p) = -1 \cdot p + 1 \cdot (1-p)$$

$$2p - 1 = 1 - 2p \qquad p = \frac{1}{2}$$

Rock-Paper-Scissor

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

Rock-Paper-Scissor

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

- Do we have any pure strategies?

Rock-Paper-Scissor

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

- Do we have any pure strategies?
  - No

Rock-Paper-Scissor

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

- Do we have any pure strategies?
  - No
- Do we have mixed strategies?

Rock-Paper-Scissor

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

- Do we have any pure strategies?
  - No
- Do we have mixed strategies?
  - Yes

Rock-Paper-Scissor

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

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

- Do we have any pure strategies?
  - No
- Do we have mixed strategies?
  - Yes

Rock-Paper-Scissor

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

	R	P	S
R	0,0	-1,1	1,-1
P	1,-1	0,0	-1,1
S	-1,1	1,-1	0,0

Rock-Paper-Scissor

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

$$\begin{cases} 0p_r + (-1)p_p + 1p_s = 1p_r + 0p_p + (-1)p_s \\ 1p_r + 0p_p + (-1)p_s = (-1)p_r + 1p_p + 0p_s \end{cases}$$

	R	P	S
R	0,0	-1,1	1,-1
P	1,-1	0,0	-1,1
S	-1,1	1,-1	0,0

Rock-Paper-Scissor

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

	R	P	S
R	0,0	-1,1	1,-1
P	1,-1	0,0	-1,1
S	-1,1	1,-1	0,0

$$\begin{cases} 0p_r + (-1)p_p + 1p_s = 1p_r + 0p_p + (-1)p_s \\ 1p_r + 0p_p + (-1)p_s = (-1)p_r + 1p_p + 0p_s \end{cases}$$

$$\begin{cases} 2p_s = p_r + p_p \\ 2p_r = p_s + p_p \end{cases}$$

$$2p_s = \frac{p_s + p_p}{2} + p_p$$

$$4p_s = p_s + 3p_p$$

Rock-Paper-Scissor

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

	R	P	S
R	0,0	-1,1	1,-1
P	1,-1	0,0	-1,1
S	-1,1	1,-1	0,0

$$\begin{cases} 0p_r + (-1)p_p + 1p_s = 1p_r + 0p_p + (-1)p_s \\ 1p_r + 0p_p + (-1)p_s = (-1)p_r + 1p_p + 0p_s \end{cases}$$

$$\begin{cases} 2p_s = p_r + p_p \\ 2p_r = p_s + p_p \end{cases} \qquad 2p_s = \frac{p_s + p_p}{2} + p_p$$

$$4p_s = p_s + 3p_p$$

$$p_s = p_p = p_r$$

Rock-Paper-Scissor

$$p_r + p_p + p_s = 1$$

$$U_1(\mathrm{R}) = U_1(\mathrm{P}) = U_1(S)$$

	R	P	S
R	0,0	-1,1	1,-1
P	1,-1	0,0	-1,1
S	-1,1	1,-1	0,0

$$\begin{cases} 0p_r + (-1)p_p + 1p_s = 1p_r + 0p_p + (-1)p_s \\ 1p_r + 0p_p + (-1)p_s = (-1)p_r + 1p_p + 0p_s \end{cases}$$

$$\begin{cases} 2p_s = p_r + p_p \\ 2p_r = p_s + p_p \end{cases} \qquad 2p_s = \frac{p_s + p_p}{2} + p_p$$

$$4p_s = p_s + 3p_p$$

$$p_s = p_p = p_r$$

$$\begin{cases} p_s = \dfrac{1}{3} \\ p_r = \dfrac{1}{3} \\ p_p = \dfrac{1}{3} \end{cases}$$

Existence of Nash Equilibria

- We have seen that not every game has a pure strategy Nash equilibrium
- Does every game have a Nash equilibrium (random or pure)?

Existence of Nash Equilibria

- We have seen that not every game has a pure strategy Nash equilibrium
- Does every game have a Nash equilibrium (random or pure)?

Theorem (Nash, 1951) Every game with a finite number of players and action profiles has at least one Nash equilibrium

Computing Nash Equilibria

- There are algorithms which compute Nash equilibria, but they are exponential in the size of the game
- It is not known if there are polynomial algorithms (but the consensus is that there are none)

Dominated Strategies Definitions

- Let s_i and s_i' be two strategies of player i and S_{-i} the set of all strategy profiles of the remaining players
  - s_i **strictly** dominates s_i' if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i,s_{-i})>u_i(s_i',s_{-i})$
  - s_i **weakly** dominates s_i' if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i,s_{-i})\geq u_i(s_i',s_{-i})$ and for at least one $s_{-i} \in S_{-i}$ it is the case that $u_i(s_i,s_{-i})>u_i(s_i',s_{-i})$
  - s_i **very weakly** dominates s_i' if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i,s_{-i})\geq u_i(s_i',s_{-i})$

Dominated Strategies
Example

	L	C	R
U	3,1	0,1	0,0
M	1,1	1,1	5,0
D	0,1	4,1	0,0

Dominated Strategies Example

	L	C	~~R~~
U	3,1	0,1	~~0,0~~
M	1,1	1,1	~~5,0~~
D	0,1	4,1	~~0,0~~

Dominated Strategies
Example

	L	C	~~R~~
U	3,1	0,1	~~0,0~~
M	1,1	1,1	~~5,0~~
D	0,1	4,1	~~0,0~~

	L	C
U	3,1	0,1
M	1,1	1,1
D	0,1	4,1

Dominated Strategies Example

	L	C	R
U	3,1	0,1	~~0,0~~
M	1,1	1,1	~~5,0~~
D	0,1	4,1	~~0,0~~

	L	C
U	3,1	0,1
M	1,1	1,1
D	0,1	4,1

Dominated Strategies
Example

	L	C	R
U	3,1	0,1	~~0,0~~
M	1,1	1,1	~~5,0~~
D	0,1	4,1	~~0,0~~

	L	C
U	3,1	0,1
M	~~1,1~~	~~1,1~~
D	0,1	4,1

Dominated Strategies
Example

	L	C	R
U	3,1	0,1	0,0
M	1,1	1,1	5,0
D	0,1	4,1	0,0

	L	C
U	3,1	0,1
M	1,1	1,1
D	0,1	4,1

	L	C
U	3,1	0,1
D	0,1	4,1

Dominated Strategies
Prisoner's Dilemma

	C	D
C	-1,-1	-4,0
D	0,-4	-3,-3

Dominated Strategies
Prisoner's Dilemma

	C	D
C	~~-1,-1~~	~~-4,0~~
D	0,-4	-3,-3

Dominated Strategies
Prisoner's Dilemma

	C	D
~~C~~	~~-1,-1~~	~~-4,0~~
D	0,-4	-3,-3

	C	D
D	0,-4	-3,-3

Dominated Strategies
Prisoner's Dilemma

	C	D
C	~~-1,-1~~	~~-4,0~~
D	0,-4	-3,-3

	C	D
D	~~0,-4~~	-3,-3

Dominated Strategies
Prisoner's Dilemma

	C	D
C	~~-1,-1~~	~~-4,0~~
D	0,-4	-3,-3

	C	D
D	~~0,-4~~	-3,-3

	D
D	-3,-3

Dominated Strategies as solution concepts

- The set of all strategy profiles that assign 0 probability to playing any action that would be removed through iterated removal of strictly dominated strategies is a **solution concept**
- Sometimes, no action can be “removed”, sometimes we can solve the game (we say the game is *solvable* by iterated elimination)

Dominated Strategies
Costs

- Iterated elimination ends after a finite number of steps
- Iterated elimination preserves Nash equilibria
  - We can use it to reduce the size of the game
- Iterated elimination of *strictly* dominated strategies can occur in any order without changing the results
- Checking if a (possibly mixed) strategy is dominated can be done in polynomial time
  - Domination by pure strategies can be checked with a very simple iterative algorithm
  - Domination by mixed strategies can be checked solving a linear problem
  - Iterative elimination needs only to check pure strategies

Dominated Strategies (domination by pure-strategies)

```
forall pure strategies aᵢ∈Aᵢ for player
i where aᵢ≠sᵢ do
 dom ← true
 forall pure-strategy profiles a₋ᵢ∈A₋ᵢ do
   if uᵢ(sᵢ,a₋ᵢ)≥uᵢ(aᵢ,a₋ᵢ) then
     dom ← false
     break
 if dom = true then
   return true
return false
```

Other Solution Concepts: Maxmin & Minmax

- The **maxmin strategy** for player *i* in an *n* player, general sum game is a not necessarily unique (mixed) strategy that maximizes *i*'s worst case payoff
- The **maxmin value** (or **security level**) is the minimum payoff level guaranteed by a maxmin strategy

$$\arg\max_{s_i} \min_{s_{-i}} u_i(s_i, s_{-i})$$

- In *two player general* sum games the **minmax strategy** for player *i* against player –*i* is the strategy that keeps the maximum payoff for –*i* at minimum
- It is a punishment

$$\arg\min_{s_i} \max_{s_{-i}} u_i(s_i, s_{-i})$$

Other Solution Concepts: Minmax, *n*-player

- In an n-player game, the **minmax strategy** for player i against player $j \neq i$ is i's component of the mixed-strategy profile s_{-j} in the expression:

$$\arg\min_{s_{-j}} \max_{s_j} u_j(s_j, s_{-j})$$

where –j denotes the set of players other that j.

- Player i receives his minmax value if players $-i$ choose their strategies to minimize i utility "after" he chose strategy s_i
- A player maxmin value is always less than or equal to his minmax value

Maxmin & Minmax Examples

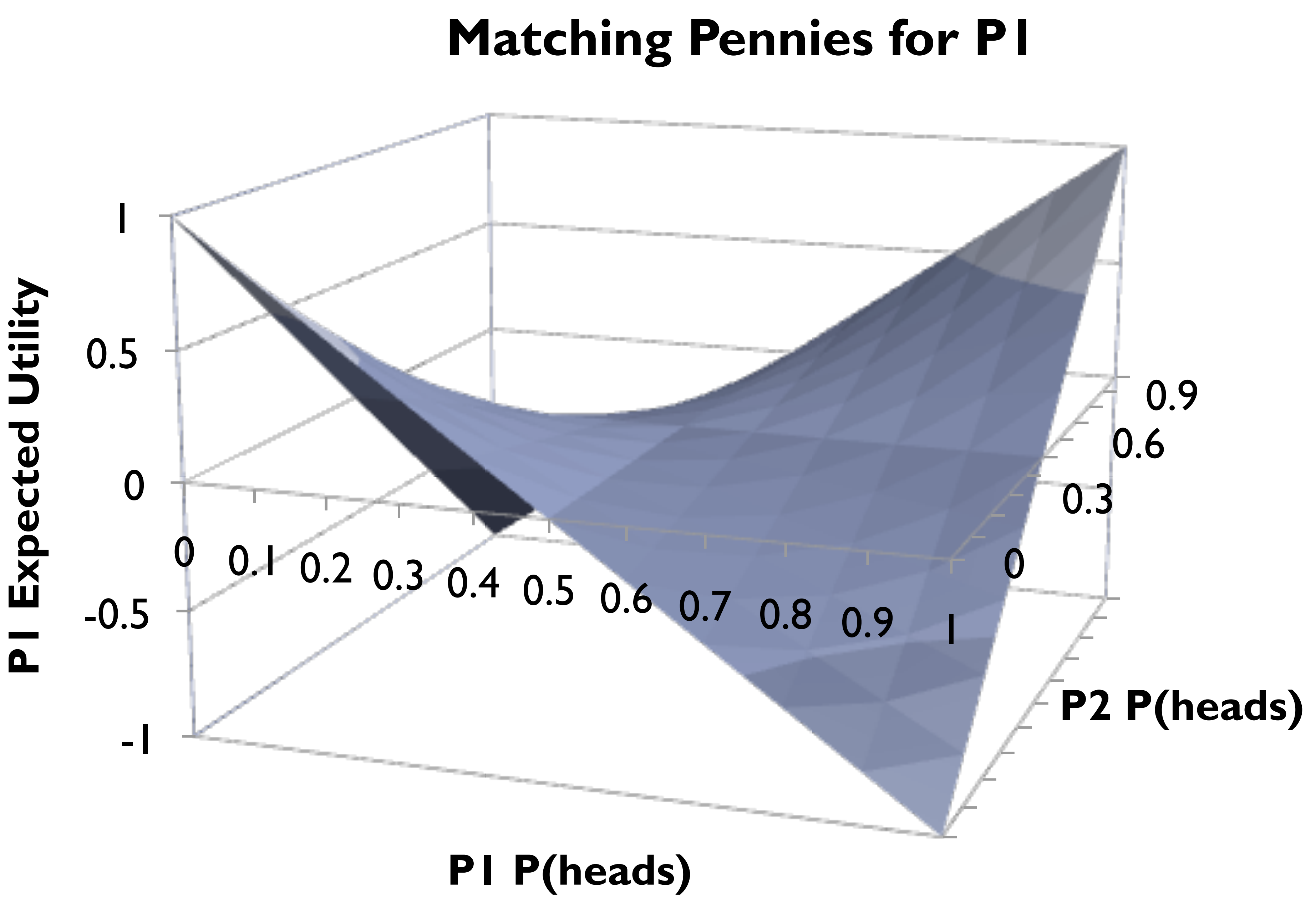
- Matching Pennies
  - Maxmin: 0.5 T + 0.5 H
  - Minmax: 0.5 T + 0.5 H
- Battle of Sexes
  - Maxmin: H→0.66 LW + 0.33 WL
    W→0.33 LW + 0.66 WL
  - Minmax: H→0.66 LW + 0.33 WL
    W→0.33 LW + 0.66 WL

Minmax Theorem

Theorem (von Neumann, 1928) In any finite, two-player, zero-sum game, in any Nash equilibrium each player receives a payoff that is equal to both his maxmin value and his minmax value

- Each player's maxmin equals his minmax (value of the game)
- Maxmin strategies coincide with minmax strategies
- Any maxmin strategy profile is a Nash equilibrium and any Nash equilibrium is a maxmin strategy profile

Matching Pennies



Minimax Regret

- An agent i's **regret** for playing an action a_i if other agents adopt action profile a_{-i} is defined as:

$$\left[\max_{a_{i'} \in A_i} u_i(a_{i'}, a_{-i})\right] - u_i(a_i, a_{-i})$$

- An agent i's **maximum regret** for playing an action a_i is defined as:

$$\max_{a_{-i} \in A_{-i}} \left(\left[\max_{a_{i'} \in A_i} u_i(a_{i'}, a_{-i})\right] - u_i(a_i, a_{-i})\right)$$

- **Minimax regret** actions for agent i are defined as:

$$\arg\min_{a_i \in A_i} \left[\max_{a_{-i} \in A_{-i}} \left(\left[\max_{a_{i'} \in A_i} u_i(a_{i'}, a_{-i})\right] - u_i(a_i, a_{-i})\right)\right]$$

Maxmin vs. Minmax Regret

	L	R
T	100, a	1-ε, b
B	2, c	1, d

P_1 Maxmin is B (why?), his Minimax Regret strategy is T

Maxmin vs. Minmax Regret

$$\text{regret}(T,[R]) = 1 - 1 + \epsilon = \epsilon$$
$$\text{regret}(B,[R]) = 1 - 1 = 0$$
$$\text{regret}(T,[L]) = 100 - 100 = 0$$
$$\text{regret}(B,[L]) = 100 - 2 = 98$$

	L	R
T	100, a	1-ε, b
B	2, c	1, d

P_1 Maxmin is B (why?), his Minimax Regret strategy is T

Maxmin vs. Minmax Regret

$$\text{regret}(T,[R]) = 1 - 1 + \epsilon = \epsilon$$
$$\text{regret}(B,[R]) = 1 - 1 = 0$$
$$\text{regret}(T,[L]) = 100 - 100 = 0$$
$$\text{regret}(B,[L]) = 100 - 2 = 98$$

$$\max \text{regret}(T) = \max\{\epsilon, 0\} = \epsilon$$
$$\max \text{regret}(B) = \max\{98, 0\} = 98$$

	L	R
T	100, a	1-ε, b
B	2, c	1, d

P_1 Maxmin is B (why?), his Minimax Regret strategy is T

Computing Equilibria

Computing Nash-Equilibrium for two-players zero-sum games

- Consider the class of two-player zero-sum games

$$\Gamma = \left(\{1,2\}, A_1 \times A_2, (u_1, u_2)\right)$$

- U_i^* is the expected utility for player i in equilibrium
- In the next slide, the LP for computing player 2 and player 1 strategies are given
- Linear Programs are rather inexpensive to compute

Computing Nash-Equilibrium for two-players zero-sum games

$$
\begin{aligned}
&\text{minimize} && U_1^* \\
&\text{subject to} && \sum_{k \in A_2} u_1\left(a_1^j, a_2^k\right) \cdot s_2^k \le U_1^* && \forall j \in A_1 \\
& && \sum_{k \in A_2} s_2^k = 1 \\
& && s_2^k \ge 0 && \forall k \in A_2
\end{aligned}
$$

- Constants: $u_1(\ldots)$
- Variables: s_2, U_1^*

U_1^* is a maxmin value!

Computing Nash-Equilibrium for two-players zero-sum games

$$
\begin{aligned}
\text{maximize} \quad & U_1^* \\
\text{subject to} \quad & \sum_{j \in A_1} u_1\left(a_1^j, a_2^k\right) \cdot s_1^j \geq U_1^* && \forall k \in A_2 \\
& \sum_{j \in A_1} s_1^j = 1 \\
& s_1^j \geq 0 && \forall j \in A_1
\end{aligned}
$$

- Constants: $u_1(\ldots)$
- Variables: s_1, U_1^*

U_1^* is a maxmin value!

Computing Nash-Equilibrium for two-players zero-sum games

$$
\begin{aligned}
&\text{minimize} && U_1^* \\
&\text{subject to} && \sum_{k \in A_2} u_1\left(a_1^j, a_2^k\right) \cdot s_2^k + r_1^j = U_1^* && \forall j \in A_1 \\
& && \sum_{k \in A_2} s_2^k = 1 \\
& && s_2^k \geq 0 && \forall k \in A_2 \\
& && r_1^j \geq 0 && \forall j \in A_1
\end{aligned}
$$

- Constants: $u_1(\ldots)$
- Variables: s_2, U_1^*

U_1^* is a maxmin value!

Computing Nash-Equilibrium for two-players zero-sum games

$$
\begin{aligned}
&\text{minimize} && U_1^* \\
&\text{subject to} && \sum_{j \in A_1} u_1\left(a_1^j, a_2^k\right) \cdot s_1^j + r_2^k = U_1^* && \forall k \in A_2 \\
& && \sum_{j \in A_1} s_1^j = 1 \\
& && s_1^j \geq 0 && \forall j \in A_1 \\
& && r_1^k \geq 0 && \forall k \in A_2
\end{aligned}
$$

- Constants: $u_1(\ldots)$
- Variables: s_2, U_1^*

U_1^* is a maxmin value!

Computing maxmin & minmax for two-players general-sum games

- We know how to compute minmax & maxmin strategies for two-players 0-sum games
- It is sufficient to transform the general-sum game in a 0-sum game
- Let G be an arbitrary two-player game G=({1,2}, $A_1 \times A_2$,(u_1,u_2)); we define G′= ({1,2}, $A_1 \times A_2$,(u_1,-u_1))
  - G′ is 0-sum: a strategy that is part of a Nash equilibrium for G′ is a maxmin strategy for 1 in G′
  - Player 1 maxmin strategy is independent of u_2
  - Thus player’s 1 maxmin strategy is the same in G and in G′
  - A minmax strategy for Player 2 in G′ is a minmax strategy for 2 in G as well (for the same reasons)

Two Players General sum Games

$$\sum_{k\in A_2} u_1\left(a_1^j, a_2^k\right)\cdot s_2^k + r_1^j = U_1^* \qquad \forall j\in A_1$$

$$\sum_{j\in A_1} u_1\left(a_1^j, a_2^k\right)\cdot s_1^j + r_2^k = U_1^* \qquad \forall k\in A_2$$

$$\sum_{j\in A_1} s_1^j = 1 \qquad \sum_{k\in A_2} s_2^k = 1$$

$$s_1^j \geq 0, \quad s_2^k \geq 0 \qquad \forall j\in A_1, \forall k\in A_2$$

$$r_1^j \geq 0, \quad r_2^k \geq 0 \qquad \forall j\in A_1, \forall k\in A_2$$

$$r_1^j \cdot s_1^j = 0, \quad r_2^k \cdot s_2^k = 0 \qquad \forall j\in A_1, \forall k\in A_2$$

- We can formulate the game as a *linear complimentarity problem* (LCP)
- This is a constraint satisfaction problem (feasibility, not optimization)
- The Lemke-Howson algorithm is the best suited to solve this kind of problems

Computing *n*-players Nash

- Could be formulated as a *nonlinear complementarity problem* (NLCP), thus it would not be easily solvable
- A sequence of linear complementarity problems (SLCP) can be used; it is not *always* convergent, but if we're lucky it's fast
- It is possible to formulate as the computation of the minimum of a specific function, both with or without constraints

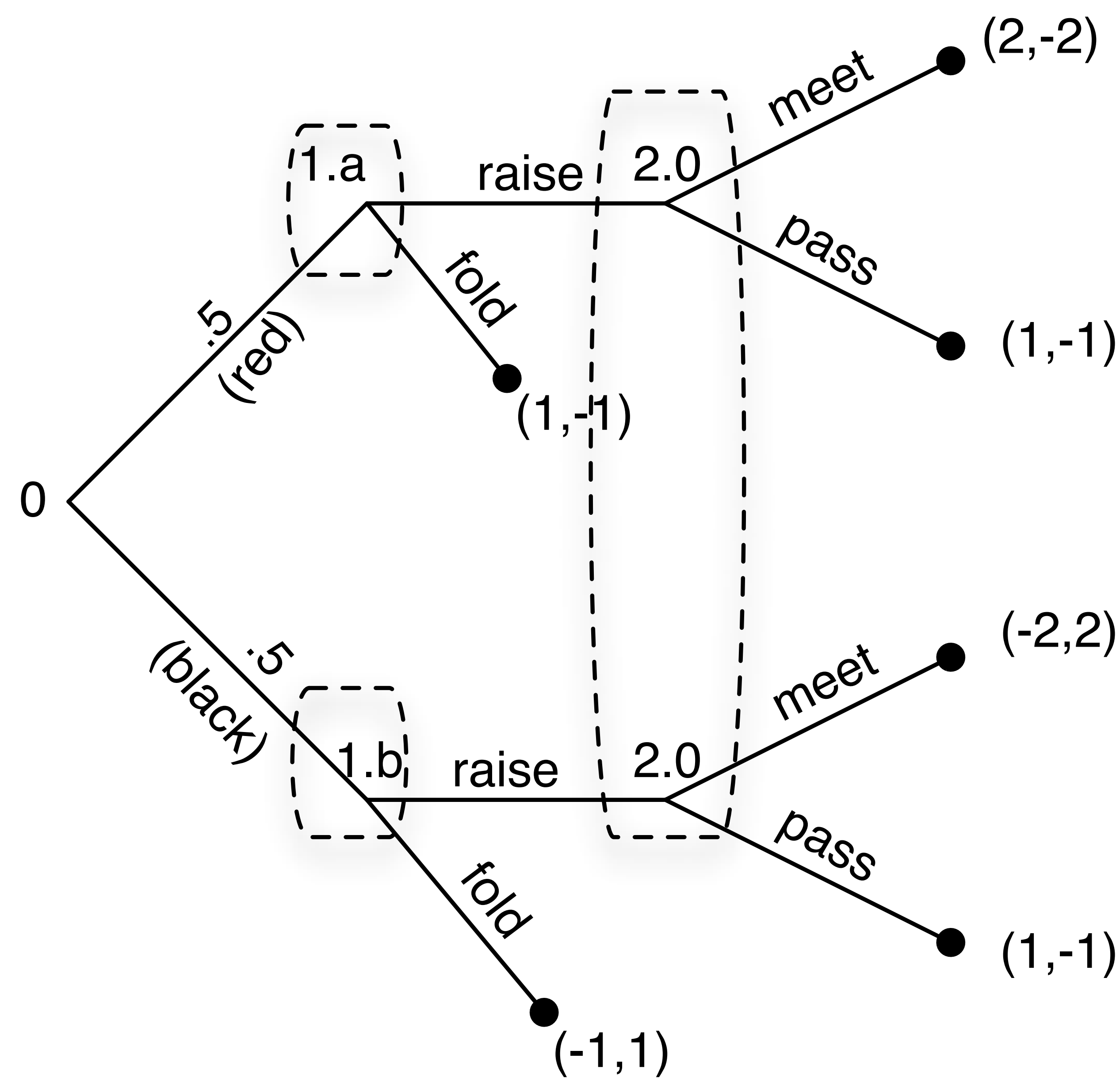
Software Tools

McKelvey, Richard D., McLennan, Andrew M., and Turocy, Theodore L. (2010). Gambit: Software Tools for Game Theory, Version 0.2010.09.01. http://www.gambit-project.org.

Other Games

Extensive Form Game Example

- Each player bets a coin
- Player 1 draw a card and is the only one to see it
- Player 1 also plays before Player 2



Repeated Games

- What happens if the same NF game is repeated:
  - An infinite number of times?
  - A finite number of times?
  - A finite but unknown number of times?

Bayesian Games

- Represent uncertainty about the *game* being played; there is a set of possible games
  - with the same number of agents and same strategy spaces but different payoffs
  - Agents beliefs are posteriors obtained conditioning a common prior on individual private signals