The joker's dilemma

Your suggestions ??

1

Pareto efficiency

Pareto optimality is a measure of efficiency.

"An outcome of a game is Pareto optimal if there is no other outcome that makes every player at least as well off and at least one player strictly better off. That is, a Pareto Optimal outcome cannot be improved upon without hurting at least one player."

NE and PO

	whale	fish
whale	2 / 2	1 / 0
fish	0 / 1	1 / 1

NE

PO

	Split	Steal
Split	3 / 3	7 / 0
Steal	0 / 7	1 / 1

NE

Shor, Mikhael, "Pareto Optimal," Dictionary of Game Theory Terms, Game Theory .net, <http://www.gametheory.net/dictionary/ParetoOptimal.html> Web accessed: 11/09/2012

2

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0 / 0	0 / 5
Refrain	5 / 0	0 / 0

3-1

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0 / 0	0 / 5
Refrain	5 / 0	0 / 0

3-2

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0, 0	5, 0
Refrain	0, 5	0, 0

3-3

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0, 0	5, 0
Refrain	0, 5	0, 0

Nash equilibrium

3-4

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0, 0	5, 0
Refrain	0, 5	0, 0

Nash equilibrium   Pareto optimal

3-5

Game theory in popular culture

When the joker can execute his plan

	Blow up	Refrain
Blow up	0, 0	5, 0
Refrain	0, 5	0, 0

Nash equilibrium   Pareto optimal

Blow up weakly dominates Refrain

3-6

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 0	0 5
Refrain	5 0	10 10

4-1

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 0	0 5
Refrain	5 0	10 10

4-2

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 0	0 5
Refrain	5 0	10 10

4-3

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 0	0 5
Refrain	5 0	10 10

Nash equilibrium

4-4

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 / 0	0 / 5
Refrain	5 / 0	10 / 10

Nash equilibrium   Pareto optimal

Game theory in popular culture

When the joker cannot execute his plan

	Blow up	Refrain
Blow up	0 / 0	0 / 5
Refrain	5 / 0	10 / 10

Nash equilibrium   Pareto optimal

Refrain weakly dominates Blow up

INFO-F-409
Learning dynamics

Mixed strategies and Nash Algorithms

T. Lenaerts and A. Nowé
MLG, Université Libre de Bruxelles and
AI-lab, Vrije Universiteit Brussel

Previous session

- What is Game Theory?
- Why do we study it in the context of computational intelligence
- Some history
- Theory of rational choice
- Defining strategic games
- Examples
- Symmetricalization
- Nash equilibrium and how to detect it
- steady state description
- Best response, strict and weak dominance
- Pareto optimality

Matching pennies

Best response analysis

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-1

Matching pennies

Best response analysis

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-2

Matching pennies

Best response analysis

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-3

Matching pennies

Best response analysis

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-4

Matching pennies

Best response analysis

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-5

Matching pennies

Best response analysis

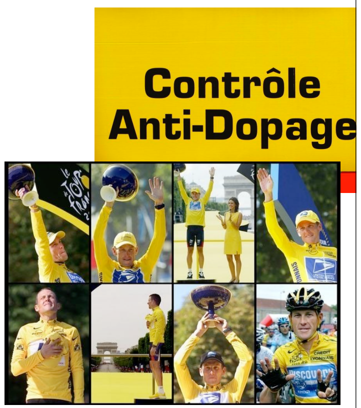
Nash Equilibrium?

	head	tail
head	-1 +1	+1 -1
tail	+1 -1	-1 +1

7-6

Inspection game

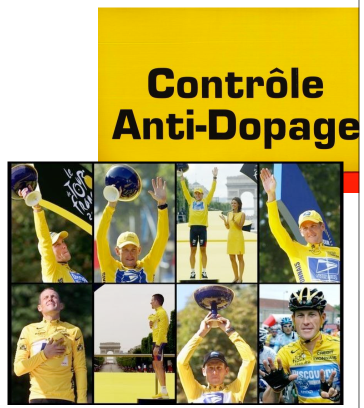
	Comply	Cheat
Don't Inspect	25 60	40 0
Inspect	25 52	20 12



8-1

Inspection game

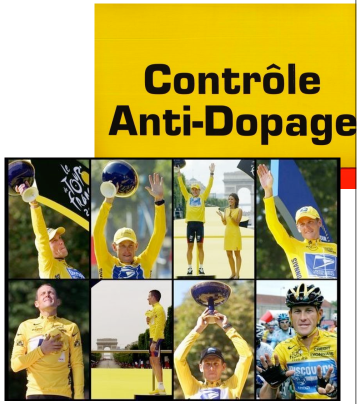
	Comply	Cheat
Don't Inspect	25 60	40 0
Inspect	25 52	20 12



8-2

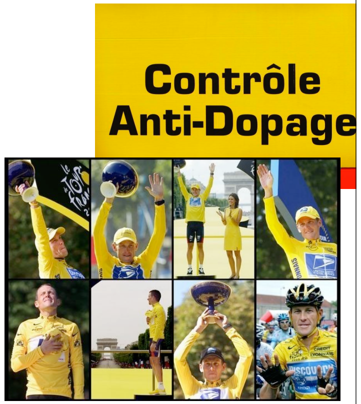
Inspection game

	Comply	Cheat
Don't Inspect	60, 25	0, 40
Inspect	52, 25	12, 20



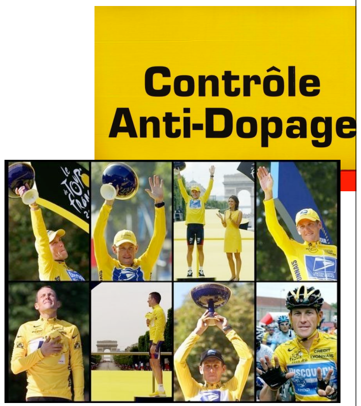
Inspection game

	Comply	Cheat
Don't Inspect	60, 25	0, 40
Inspect	52, 25	12, 20



Inspection game

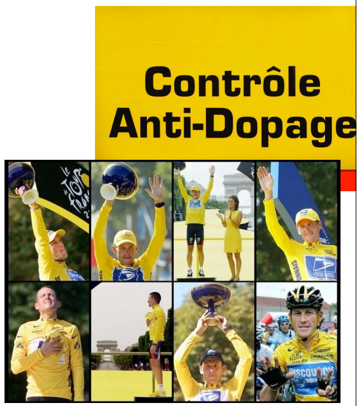
	Comply	Cheat
Don't Inspect	60, 25	0, 40
Inspect	52, 25	12, 20



Inspection game

	Comply	Cheat
Don't Inspect	60, 25	0, 40
Inspect	52, 25	12, 20

Nash Equilibrium?



Best response

	C	H
D	60, 25	0, 40
I	52, 25	12, 20

	DC	DH	IC	IH
DC	42.5, 42.5	12.5, 50	42.5, 38.5	12.5, 46
DH	50, 12.5	20, 20	40, 18.5	10, 26
IC	38.5, 42.5	18.5, 40	38.5, 38.5	18.5, 36
IH	46, 12.5	26, 10	36, 18.5	16, 16

	D	I	C	H
D	0, 0	0, 0	60, 25	0, 40
I	0, 0	0, 0	52, 25	12, 20
C	25, 60	25, 52	0, 0	0, 0
H	40, 0	20, 12	0, 0	0, 0

9

Mixed strategies

	Head	Tail
Head	+1, -1	-1, +1
Tail	-1, +1	+1, -1

Is there an equilibrium when we allow players to randomize over their actions ?

10-1

Mixed strategies

P(Head) $= p$ and
P(Tail) $= (1-p)$

	Head	Tail
Head	+1, -1	-1, +1
Tail	-1, +1	+1, -1

10-2

Mixed strategies

P(Head) $= q$ and
P(Tail) $= (1-q)$

P(Head) $= p$ and
P(Tail) $= (1-p)$

	Head	Tail
Head	+1, -1	-1, +1
Tail	-1, +1	+1, -1

10-3

Mixed strategies

P(Head) $= q$ and
P(Tail) $= (1-q)$

P(Head) $= p$ and
P(Tail) $= (1-p)$

	Head	Tail
Head	+1, -1	-1, +1
Tail	-1, +1	+1, -1

strategy profile : $\{\{($Head$, p);($Tail$, 1\text{-}p)\}; \{($Head$, q);($Tail$, 1\text{-}q)\}$

Mixed strategies

Definition :

A **mixed strategy** of a player in a strategic game is a probability distribution over the player's actions

We denote a **mixed strategy profile** by α,

$\alpha_i(a_i)$ is the probability assigned by player i's mixed strategy α_i to her action a_i

Example:

$\alpha_1($Head$) = p$ $\alpha_2($Head$) = q$

$\alpha_1($Tail$) = 1\text{-}p$ $\alpha_2($Tail$) = 1\text{-}q$

Note the when $\alpha_1($Head$) = 1$, the mixed strategy $(1,0)$ is a **pure** strategy

Strategic games with vNM preferences

von Neumann-Morgenstern (vNM) preferences are preferences regarding lotteries (probability distribution, mixed strategies)

They are represented by the **expected value** of a payoff function over the deterministic outcomes

$$U(p_1,\dots, p_K)) = \sum_{k=1}^{K} p_k\, u(a_k)$$

the payoff function u is called a **Bernouilli payoff function**

Strategic games with vNM preferences

There is a Bernouilli payoff function u over deterministic outcomes such that the decision-makers preferences over lotteries represented by this function

$$U(p_1,\dots, p_K)) = \sum_{k=1}^{K} p_k\, u(a_k)$$

allows one to conclude :

$$\sum_{k=1}^{K} p_k\, u(a_k) > \sum_{k=1}^{K} p'_k\, u(a_k)$$

if and only if the decision-maker prefers the lottery $(p_1, \dots, p_K)$ over the lottery $(p_1', \dots, p_K')$

Example

Assume a game for which the outcomes are, A, B or C and naturally she prefers C over B over A

Assume also that that she prefers mixed strategy ($1/2$, 0, $1/2$) over (0, $3/4$, $1/4$)

Then the payoff function $u(A)=0$, $u(B)=1$ and $u(C)=4$ makes these preferences consistent since

$$(1/2*0+1/2*4) > (3/4*1+1/4*4)$$

Suppose that she on the other hand prefers (0, $3/4$, $1/4$) over ($1/2$, 0, $1/2$), then the payoff function $u(A)=0$, $u(B)=\mathbf{3}$ and $u(C)=4$ makes these preferences consistent since

$$(1/2*0+1/2*4) < (3/4*\mathbf{3}+1/4*4)$$

Strategic games with vNM preferences

A **strategic game** consists of :
- a set of players
- for each player a set of actions
- for each player, a *Bernouilli payoff function* over action profiles, representing the preferences of the player

Mixed Nash Equilibrium

Assume that (α_i',α_{-i}) is the **mixed** strategy profile in which every player j **except *i*** chooses her mixed strategy α_j as specified by α, whereas player i deviates to α_i'

Definition :

The mixed strategy profile α^* in a strategic game is a **mixed strategy Nash Equilibrium** if for every player i and for every mixed strategy α_i of player i, the expected payoff to i in α^* is at least as large as the expected payoff to i in $(\alpha_i,\alpha_{-i}*)$ according to a payoff function that represents player i's preferences over lotteries.

Mixed Nash Equilibrium

Definition :

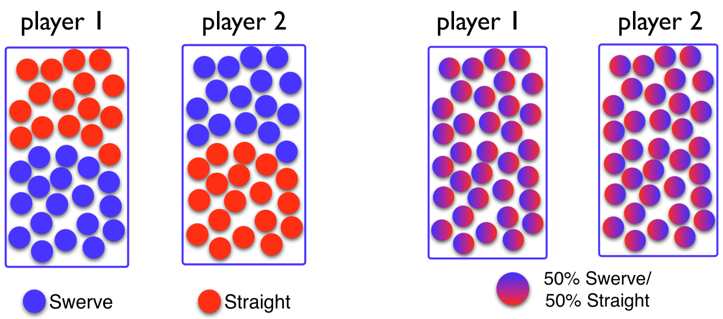
Equivalently, for every player i,

$U_i(\alpha^*) \geq U_i(\alpha_i,\alpha_{-i}*)$ for every mixed strategy α_i of player i

where $U_i(\alpha)$ is the player's i expected payoff to the mixed strategy profile α

Stochastic steady state

Again this NE can be interpreted as an steady state of an interaction between the members of several populations, one for each player in the game



Best-response

To find the mixed strategy NE, we can again make use of the notion of a Best-response.

Definition :

The mixed strategy profile α^* in a strategic game is a mixed strategy Nash Equilibrium if and only if α_i^* is in $B_i(\alpha_{-i}^*)$ for every player i

$B_i(\alpha_{-i})$ is the set of all player i's best mixed strategies when the list of the other players' mixed strategy is α_{-i}

In two-player/two-action games

What is the set of best responses of player 1 to a mixed strategy of player 2?

	L (q)	R (1-q)
T (p)	pq	$p(1-q)$
B (1-p)	$(1-p)q$	$(1-p)(1-q)$

$$E_1(\alpha) = p[q*u_1(T,L) + (1-q)*u_1(T,R)] + (1-p)[q*u_1(B,L) + (1-q)*u_1(B,R)]$$

$$E_1(\alpha) = p*E_1(T, \alpha_{-1}) + (1-p)*E_1(B, \alpha_{-1})$$

the **expected payoff** E_1 of player 1, given player 2's mixed strategy is a linear function of p

In two-player/two-action games

The linearity implies 3 possible outcomes :

1. player 1's unique best response is the pure strategy T (when $E_1(T, \alpha_{-1}) > E_1(B, \alpha_{-1})$)

2. player 1's unique best response is the pure strategy B (when $E_1(T, \alpha_{-1}) < E_1(B, \alpha_{-1})$)

3. all player 1's mixed strategies are all best responses (when $E_1(T, \alpha_{-1}) = E_1(B, \alpha_{-1})$)

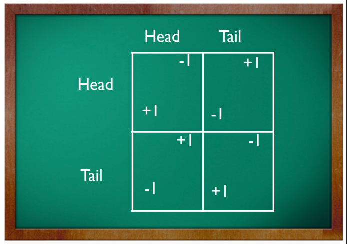
Matching pennies

1. player 1's expected payoff for the pure strategy *Head* (p) is

$$q*1+(1-q)*(-1) = 2q-1$$

2. player 1's expected payoff for the pure strategy *Tail* ($1-p$) is

$$q*(-1)+(1-q)*1 = 1-2q$$



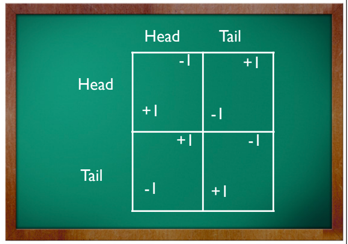
Matching pennies

1. player 1's expected payoff for the pure strategy *Head* (p) is

$$q*1+(1-q)*(-1) = 2q-1$$

2. player 1's expected payoff for the pure strategy *Tail* ($1-p$) is

$$q*(-1)+(1-q)*1 = 1-2q$$



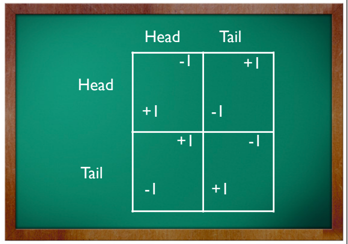
Matching pennies

1. player 1's expected payoff for the pure strategy *Head* (p) is

$$q*1+(1-q)*(-1) = 2q-1$$

2. player 1's expected payoff for the pure strategy *Tail* ($1-p$) is

$$q*(-1)+(1-q)*1 = 1-2q$$



$2q-1 < 1-2q$ when ***$q < 1/2$***
Thus best response set is {Tail} or $p=0$

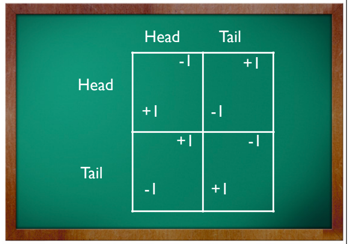
Matching pennies

1. player 1's expected payoff for the pure strategy *Head* (p) is

$$q*1+(1-q)*(-1) = 2q-1$$

2. player 1's expected payoff for the pure strategy *Tail* ($1-p$) is

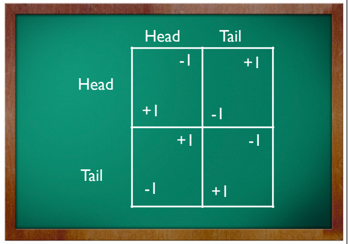
$$q*(-1)+(1-q)*1 = 1-2q$$



$2q-1 < 1-2q$ when ***$q < 1/2$***
Thus best response set is {Tail} or $p=0$

$2q-1 > 1-2q$ when ***$q > 1/2$***
Thus best response set is {Head} or $p=1$

Matching pennies



1. player 1's expected payoff for the pure strategy *Head* (p) is

   $q *1 +(1-q)*(-1) = 2q-1$

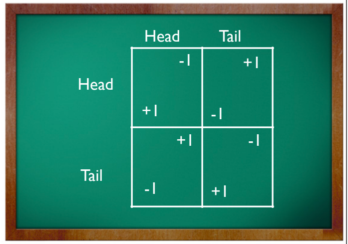
2. player 1's expected payoff for the pure strategy *Tail* ($1-p$) is

   $q *(-1) +(1-q) *1 = 1-2q$

$\boldsymbol{2q-1 = 1-2q}$ when $\boldsymbol{q = 1/2}$
Thus best response set is the set of all mixed strategies

Matching pennies

And for player 2 ...



1. player 2's expected payoff for the pure strategy Head (q) is

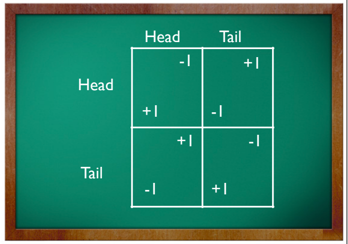
   $p *-1 +(1-p)*1 = 1-2p$

2. player 2's expected payoff for the pure strategy Tail (1-q) is

   $p *1 +(1-p) *(-1) = 2p-1$

Matching pennies

And for player 2 ...



1. player 2's expected payoff for the pure strategy Head (q) is

   $p *-1 +(1-p)*1 = 1-2p$

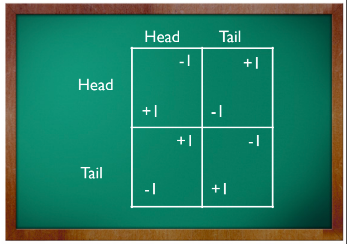
2. player 2's expected payoff for the pure strategy Tail (1-q) is

   $p *1 +(1-p) *(-1) = 2p-1$

$\boldsymbol{1-2p > 2p-1}$ when $\boldsymbol{p < 1/2}$ thus best response set is {Head} or $q=1$

Matching pennies

And for player 2 ...



1. player 2's expected payoff for the pure strategy Head (q) is

   $p *-1 +(1-p)*1 = 1-2p$

2. player 2's expected payoff for the pure strategy Tail (1-q) is

   $p *1 +(1-p) *(-1) = 2p-1$

$\boldsymbol{1-2p > 2p-1}$ when $\boldsymbol{p < 1/2}$ thus best response set is {Head} or $q=1$

$\boldsymbol{1-2p < 2p-1}$ when $\boldsymbol{p > 1/2}$ thus best response set is {Tail} or $q=0$

Matching pennies

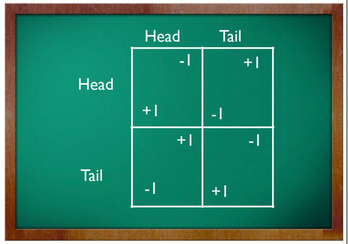
And for player 2 ...

1. player 2's expected payoff for the pure strategy Head (q) is

$$p *-1 +(1-p)*1 = 1-2p$$

2. player 2's expected payoff for the pure strategy Tail (1-q) is

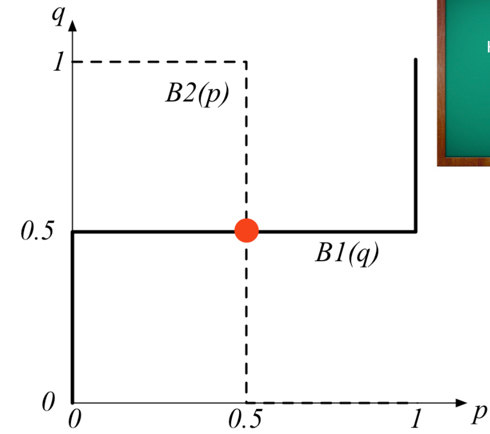
$$p *1 +(1-p) *(-1) = 2p-1$$

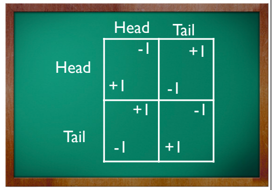


$2p\text{-}1 = 1\text{-}2p$ **when** $p = 1/2$ for any mixed strategy

Thus best response set is the set of all mixed strategies

Matching pennies





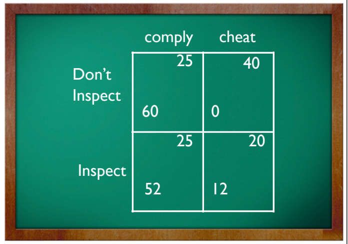
Inspection game

1. player 1's expected payoff for the pure strategy *Don't Inspect* (p) is

$$q *60 +(1-q)*0 = 60q$$

2. player 1's expected payoff for the pure strategy *Inspect* ($1-p$) is

$$q *52 +(1-q) *12 = 40q+12$$



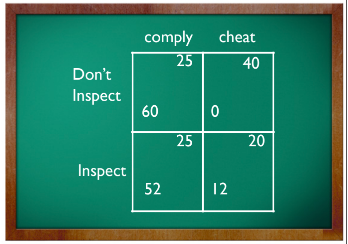
Inspection game

1. player 1's expected payoff for the pure strategy *Don't Inspect* (p) is

$$q *60 +(1-q)*0 = 60q$$

2. player 1's expected payoff for the pure strategy *Inspect* ($1-p$) is

$$q *52 +(1-q) *12 = 40q+12$$



When $q > 3/5$ then $60q > 40q+12$

Thus best response set is {Don't Inspect} or $p=1$

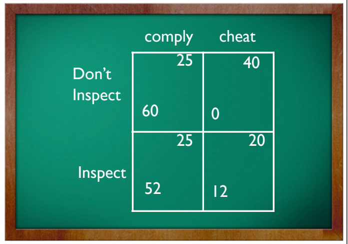
Inspection game

1. player 1's expected payoff for the pure strategy *Don't Inspect* (p) is

$$q *60 +(1-q)*0 = 60q$$

2. player 1's expected payoff for the pure strategy *Inspect* ($1-p$) is

$$q *52 +(1-q) *12 = 40q+12$$



When $q > 3/5$ then $60q > 40q+12$
Thus best response set is {Don't Inspect} or $p=1$

When $q < 3/5$ then $60q < 40q+12$
Thus best response set is {Inspect} or $p=0$

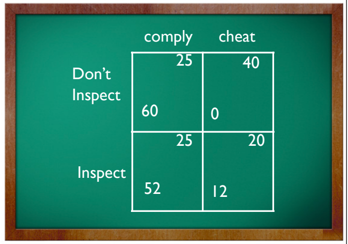
Inspection game

1. player 1's expected payoff for the pure strategy *Don't Inspect* (p) is

$$q *60 +(1-q)*0 = 60q$$

2. player 1's expected payoff for the pure strategy *Inspect* ($1-p$) is

$$q *52 +(1-q) *12 = 40q+12$$



When $q = 3/5$ then $60q = 40q+12$
Thus best response set is the set of all p values in $[0\ 1]$

Inspection game

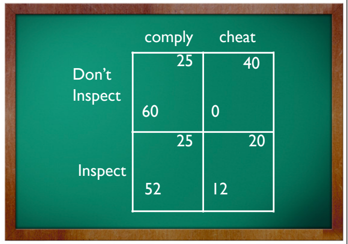
And for player 2 ...

1. player 2's expected payoff for the pure strategy *Comply* (q) is

$$p *25 +(1-p)25 = 25$$

2. player 2's expected payoff for the pure strategy *Cheat* (1-q) is

$$p *40 +(1-p) *20 = 20p+20$$



Inspection game

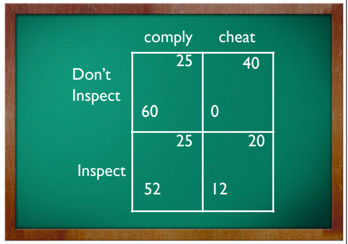
And for player 2 ...

1. player 2's expected payoff for the pure strategy *Comply* (q) is

$$p *25 +(1-p)25 = 25$$

2. player 2's expected payoff for the pure strategy *Cheat* (1-q) is

$$p *40 +(1-p) *20 = 20p+20$$



When $p < 1/4$ then $25 > 20p+20$ thus best response set is {*Comply*} or $q=1$

Inspection game

And for player 2 ...

1. player 2's expected payoff for the pure strategy *Comply* (q) is

$$p *25 +(1-p)25 = 25$$

2. player 2's expected payoff for the pure strategy *Cheat* (1-q) is

$$p *40 +(1-p) *20 = 20p+20$$

	comply	cheat
Don't Inspect	25 60	40 0
Inspect	25 52	20 12

When $p < 1/4$ then $25 > 20p+20$ thus best response set is {*Comply*} or $q=1$

When $p > 1/4$ then $25 < 20p+20$ thus best response set is {*Cheat*} or $q=0$

Inspection game

And for player 2 ...

1. player 2's expected payoff for the pure strategy *Comply* (q) is

$$p *25 +(1-p)25 = 25$$

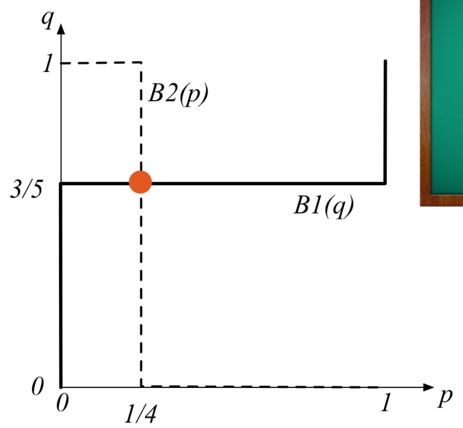
2. player 2's expected payoff for the pure strategy *Cheat* (1-q) is

$$p *40 +(1-p) *20 = 20p+20$$

	comply	cheat
Don't Inspect	25 60	40 0
Inspect	25 52	20 12

When $p = 1/4$ then $25 = 20p+20$ for any mixed strategy.
Thus best response set is the set of all values for q in $[0 .. 1]$

Inspection game



	comply	cheat
Don't Inspect	25 60	40 0
Inspect	25 52	20 12

Battle of the sexes

1. player 1's expected payoff for the pure strategy Bach is

$$q *2 +(1-q)*0 = 2q$$

2. player 1's expected payoff for the pure strategy Stravinsky is

$$q *0 +(1-q) *1 = 1-q$$

	Bach	Strav.
Bach	1 2	0 0
Strav.	0 0	2 1

Battle of the sexes

1. player 1's expected payoff for the pure strategy Bach is

$q *2 +(1\text{-}q)*0 = 2q$

2. player 1's expected payoff for the pure strategy Stravinsky is

$q *0 +(1\text{-}q) *1 = 1\text{-}q$

	Bach	Strav.
Bach	1 2	0 0
Strav.	0 0	2 1

$2q < 1\text{-}q$ or $q < 1/3$ then the best response set is {Strav.}

Battle of the sexes

1. player 1's expected payoff for the pure strategy Bach is

$q *2 +(1\text{-}q)*0 = 2q$

2. player 1's expected payoff for the pure strategy Stravinsky is

$q *0 +(1\text{-}q) *1 = 1\text{-}q$

	Bach	Strav.
Bach	1 2	0 0
Strav.	0 0	2 1

$2q < 1\text{-}q$ or $q < 1/3$ then the best response set is {Strav.}

$2q > 1\text{-}q$ or $q > 1/3$ then the best response set is {Bach}

Battle of the sexes

1. player 1's expected payoff for the pure strategy Bach is

$q *2 +(1\text{-}q)*0 = 2q$

2. player 1's expected payoff for the pure strategy Stravinsky is

$q *0 +(1\text{-}q) *1 = 1\text{-}q$

	Bach	Strav.
Bach	1 2	0 0
Strav.	0 0	2 1

$2q < 1\text{-}q$ or $q < 1/3$ then the best response set is {Strav.}

$2q > 1\text{-}q$ or $q > 1/3$ then the best response set is {Bach}

$2q = 1\text{-}q$ or $q = 1/3$ then all the players mixed strategies are best responses

Battle of the sexes

1. player 2's expected payoff for the pure strategy Bach is

$p *1 +(1\text{-}p)*0 = p$

2. player 2's expected payoff for the pure strategy Stravinsky is

$p *0 +(1\text{-}p) *2 = 2(1\text{-}p)$

	Bach	Strav.
Bach	1 2	0 0
Strav.	0 0	2 1

Battle of the sexes

1. player 2's expected payoff for the pure strategy Bach is

$$p *1 +(1-p)*0 = p$$

2. player 2's expected payoff for the pure strategy Stravinsky is

$$p *0 +(1-p) *2 = 2(1-p)$$

	Bach	Strav.
Bach	2, 1	0, 0
Strav.	0, 0	1, 2

$p < 2(1-p)$ or $p < 2/3$ then the best response set is {Strav.}

33-2

Battle of the sexes

1. player 2's expected payoff for the pure strategy Bach is

$$p *1 +(1-p)*0 = p$$

2. player 2's expected payoff for the pure strategy Stravinsky is

$$p *0 +(1-p) *2 = 2(1-p)$$

	Bach	Strav.
Bach	2, 1	0, 0
Strav.	0, 0	1, 2

$p < 2(1-p)$ or $p < 2/3$ then the best response set is {Strav.}

$p > 2(1-p)$ or $p > 2/3$ then the best response set is {Bach}

33-3

Battle of the sexes

1. player 2's expected payoff for the pure strategy Bach is

$$p *1 +(1-p)*0 = p$$

2. player 2's expected payoff for the pure strategy Stravinsky is

$$p *0 +(1-p) *2 = 2(1-p)$$

	Bach	Strav.
Bach	2, 1	0, 0
Strav.	0, 0	1, 2

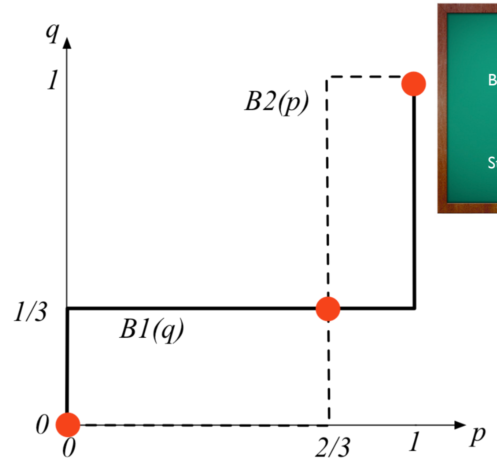
$p < 2(1-p)$ or $p < 2/3$ then the best response set is {Strav.}

$p > 2(1-p)$ or $p > 2/3$ then the best response set is {Bach}

$p = 2(1-p)$ or $p = 2/3$ then all the players mixed strategies are best responses

33-4

Battle of the sexes

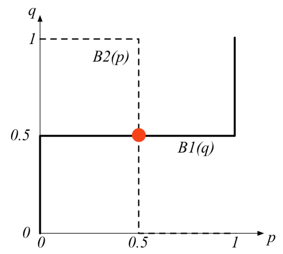


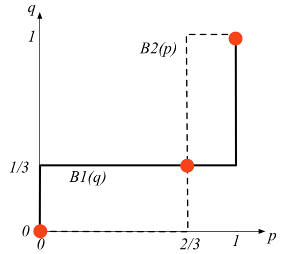
	Bach	Strav.
Bach	2, 1	0, 0
Strav.	0, 0	1, 2

34

Existence

Every strategic game with vNM preferences in which each player has a finite number of actions has a mixed strategy Nash equilibrium





Equilibrium test

How can we verify in more advanced game if a mixed strategy profile is a mixed Nash Equilibrium?

A player's expected payoff to the mixed strategy profile α is a weighted average of her expected payoffs to all mixed strategy profiles of the type (a_i, α_{-i}) where the weight attached to (a_i, α_{-i}) is the probability $\alpha_i(a_i)$ assigned to a_i by player i's mixed strategy α_i

$$U_i(\alpha) = \sum_{a_i \in A_i} \alpha_i(a_i) E_i(a_i, \alpha_{-1})$$

Equilibrium test

The previous property leads to an equilibrium test :

A mixed strategy profile α^* in a strategic game with vNM preferences in which each player has a finitely many actions is a mixed strategy Nash equilibrium if and only if for each player i,

(1) the expected payoff, given α_{-i}^*, of every action a_i in α_i that has $\alpha_i(a_i)>0$, is the same

(2) the expected payoff, given α_{-i}^*, of every action a_i in α_i that has a $\alpha_i(a_i)=0$, has at most the payoff of **(1)**

The expected payoff in equilibrium is the expected payoff of **(1)**

Example

Take for instance the Battle of the Sexes :

We have three possible mixed strategy Nash equilibria :
{(1,0);(1,0)}, {(0,1),(0,1)} and {(2/3,1/3);(1/3,2/3)}

	Bach (1)	Strav. (0)
Bach (1)	2, 1	0, 0
Strav. (0)	0, 0	1, 2

expected payoff for actions

$Bach, (P_{Bach} = 1) \rightarrow 1*2 + 0*0 = 2$ **(1)**

$Strav., (P_{Strav} = 0) \rightarrow 1*0 + 0*1 = 0$ **(2)**

Example

Take for instance the Battle of the Sexes :

We have three possible mixed strategy Nash equilibria :
{(1,0);(1,0)}, {(0,1),(0,1)} and {(2/3,1/3);(1/3,2/3)}

expected payoff for actions

	Bach (0)	Strav. (1)
Bach (0)	2, 1	0, 0
Strav. (1)	0, 0	1, 2

Bach, $(P_{Bach} = 0) \longrightarrow 0*2 + 1*0 = 0$ **(2)**

Strav., $(P_{Strav} = 1) \longrightarrow 0*0 + 1*1 = 1$ **(1)**

38-2

Example

Take for instance the Battle of the Sexes :

We have three possible mixed strategy Nash equilibria :
{(1,0);(1,0)}, {(0,1),(0,1)} and {(2/3,1/3);(1/3,2/3)}

expected payoff for actions

	Bach (1/3)	Strav. (2/3)
Bach (2/3)	2, 1	0, 0
Strav. (1/3)	0, 0	1, 2

Bach, $(P_{Bach} = 2/3) \longrightarrow 1/3*2 + 2/3*0 = 2/3$ **(1)**

Strav., $(P_{Strav} = 1/3) \longrightarrow 1/3*0 + 2/3*1 = 2/3$ **(1)**

38-3

Example

Take for instance the Battle of the Sexes :

We have three possible mixed strategy Nash equilibria :
{(1,0);(1,0)}, {(0,1),(0,1)} and {(2/3,1/3);(1/3,2/3)}

expected payoff for actions

	Bach (1/2)	Strav. (1/2)
Bach (1/2)	2, 1	0, 0
Strav. (1/2)	0, 0	1, 2

Bach, $(P_{Bach} = 1/2) \longrightarrow 1/2*2 + 1/2*0 = 1$ **(1)**

Strav., $(P_{Strav} = 1/2) \longrightarrow 1/2*0 + 1/2*1 = 1/2$ **(1)**

38-4

Example

Take for instance the Battle of the Sexes :

We have three possible mixed strategy Nash equilibria :
{(1,0);(1,0)}, {(0,1),(0,1)} and {(2/3,1/3);(1/3,2/3)}

expected payoff for actions

	Bach (1/2)	Strav. (1/2)
Bach (1/2)	2, 1	0, 0
Strav. (1/2)	0, 0	1, 2

Bach, $(P_{Bach} = 1/2) \longrightarrow 1/2*2 + 1/2*0 = 1$ **(1)**

$\neq$

Strav., $(P_{Strav} = 1/2) \longrightarrow 1/2*0 + 1/2*1 = 1/2$ **(1)**

38-5

Support

Remember

The mixed strategy profile α^* in a strategic game is a mixed strategy Nash Equilibrium if and only if α_i^* is in $B_i(\alpha_{-i}^*)$ for every player i (it is a best-response to the rest)

Now (Best Response Condition)

A mixed strategy is a best response if and only if all pure strategies in its ***support*** are best responses

The support of a mixed strategy is the set of all pure strategies with non-zero probability

Thus players combine pure best response strategies (proof see Algorithmic Game Theory p. 55)

Support

Take for instance the following symmetric game:

	a	b	c
a	0, 0	3, 0	0, 2
b	0, 3	0, 0	3, 2
c	2, 0	2, 3	2, 2

Consider the following equilibrium for both players

$(0,1/3,2/3)$ support S= {b,c}

We can verify whether it is an equilibrium by calculating the utility of each action (assuming that the opponent plays the same mixed strategy)

$u_a = 0*0+3*(1/3)+0*(2/3) = 1$

$u_b = 0*0+0*(1/3)+3*(2/3) = 2$

$u_c = 2*0+2*(1/3)+2*(2/3) = (6/3)=2$

both are best responses

Support

Take for instance the following symmetric game:

	a	b	c
a	0, 0	3, 0	0, 2
b	0, 3	0, 0	3, 2
c	2, 0	2, 3	2, 2

All pure strategies in the support must have maximum and equal payoff

From the perspective of the row player, **playing just b or c or some mixture of b and c, is equally beneficial to the equilibrium mixed strategy**

The only benefit of playing the NE is that it motivates the other player to do the same!

Support

Thus finding the Nash equilibrium comes down to finding the right support.

Hence finding the Nash equilibrium is a combinatorial problem

Once found the precise mixed strategy can be computed by solving a system of algebraic equations (see Algorithmic Game Theory book p. 31)

Finding the supports

Assume the following game

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

(see Equilibrium Computation for Two-Player Games in Strategic and Extensive form (Chapter 3) by B.Von Stengel in Algorithmic Game Theory p. 53-78)

43-1

Finding the supports

Assume the following game

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

The game has already 1 pure NE

Best response indicates (a,d) or ((1,0,0),(1,0))

(see Equilibrium Computation for Two-Player Games in Strategic and Extensive form (Chapter 3) by B.Von Stengel in Algorithmic Game Theory p. 53-78)

43-2

Finding the supports

Assume the following game

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

The game has already 1 pure NE

Best response indicates (a,d) or ((1,0,0),(1,0))

mixed equilibria contain at least 2 pure strategies in their support

Possible support are : {{a,b}{d,e}}
{{a,c}{d,e}}
{{b,c}{d,e}}

(see Equilibrium Computation for Two-Player Games in Strategic and Extensive form (Chapter 3) by B.Von Stengel in Algorithmic Game Theory p. 53-78)

43-3

Finding the supports

Take first the support {{a,b},{d,e}}

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

player 2 has to be indifferent between action **d** and **e** to make them a best response to the actions of player 1 (and vice versa)

Solve: player 1

$x_a + x_b = 1$
$3x_a + 2x_b = 2x_a + 6x_b$
$x_a = 4/5$
$x_b = 1/5$

exp. payoffs for player 2 $(14/5,\ 14/5)$

Solve: player 2

$y_d + y_e = 1$
$3y_d + 3y_e = 2y_d + 5y_e$
$y_d = 2/3$
$y_e = 1/3$

exp. payoffs for player 1 $(3,3,2)$

44

Finding the supports

Take another support {{b,c},{d,e}}

	d	e
a	3 3	2 3
b	2 2	6 5
c	3 0	1 6

player 2 has to be indifferent between action **d** and **e** to make them a best response to the actions of player 1 (and vice versa)

Solve: player 1

$x_b + x_c = 1$

$2x_b + 3x_c = 6x_b + 1x_c$

$x_b = 1/3$

$x_c = 2/3$

exp. payoffs for player 2 $(8/3,\ 8/3)$

Solve: player 2

$y_d + y_e = 1$

$2y_d + 5y_e = 0y_d + 6y_e$

$y_d = 1/3$

$y_e = 2/3$

exp. payoffs for player 1 $(3,4,4)$

Finding the supports

Take another support {{a,c},{d,e}}

	d	e
a	3 3	2 3
b	2 2	6 5
c	3 0	1 6

player 2 has to be indifferent between action **d** and **e** to make them a best response to the actions of player 1 (and vice versa)

Solve: player 1

$x_a + x_c = 1$

$3x_a + 3x_c = 2x_a + 1x_c$

$x_a = 2$

$x_c = -1$

$\boldsymbol{x}$ is no longer a vector of probabilities

Finding the supports

What about the support {{a,b,c},{d,e}} ?

	d	e
a	3 3	2 3
b	2 2	6 5
c	3 0	1 6

In any mixed-strategy Nash Equilibrium α^* of a *non-degenerate* game, the supports for both players are of equal size.

A two-player game is non-degenerate when no mixed strategy of support size k has more than k pure best responses

Finding the supports

Dickhaut-Kaplan algorithm (1991)

Input : a non-degenerate bi-matrix game, with M and N strategy sets for player 1 and player 2 respectively

Output : All Nash equilibria of the game

For each $k = 1 \dots \min\{m,n\}$

For each pair (I,J) a k-sized subset of M and N

Solve $\sum_{i \in I} x_i b_{ij} = v$ for $j \in J$, $\sum_{i \in I} x_i = 1$ and

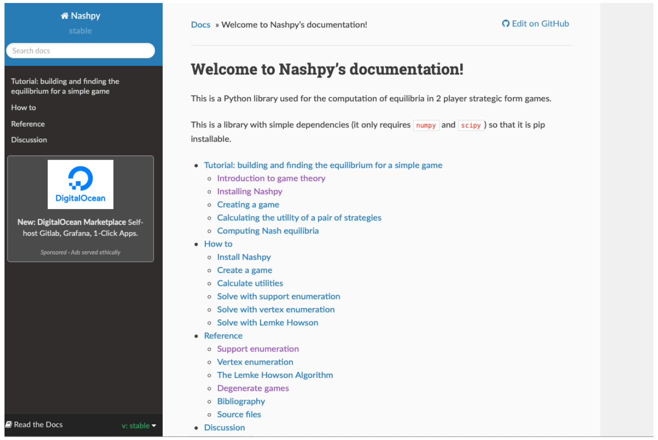
$\sum_{j \in J} a_{ij} y_j = u$ for $i \in I$, $\sum_{j \in J} y_j = 1$

and check that $\boldsymbol{x} \geq 0$, $\boldsymbol{y} \geq 0$ and that no mixed

strategy of support size k has more than

k pure best responses

Nashpy



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

49

```
(base) Toms-MacBook-Pro-2:~ tlenaert$ python
Python 3.7.1 (default, Dec 14 2018, 13:28:58)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import nashpy as nash
>>> import numpy as np
>>> A=np.array([[3,3],[2,5],[0,6]])
>>> B=np.array([[3,2],[2,6],[3,1]])
>>> mygame=nash.Game(A,B)
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
```

50

```
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
>>> equilibria=mygame.support_enumeration()
>>> for eq in equilibria:
...     print(eq)
...
(array([1., 0., 0.]), array([1., 0.]))
(array([0.8, 0.2, 0. ]), array([0.66666667, 0.33333333]))
(array([0.        , 0.33333333, 0.66666667]), array([0.33333333, 0.66666667]))
>>>
```

51

Vertex enumeration

Uses a best-response polyhedron (BRP) to identify the supports of the equilibrium strategies

$\tilde{N} = \{(x,v) \in \boldsymbol{R}^M \times \boldsymbol{R} \mid B^T x \leq \boldsymbol{1} v,\ x \geq \boldsymbol{0},\ \boldsymbol{1}^T x = 1\}$ row player

$\tilde{O} = \{(y,u) \in \boldsymbol{R}^N \times \boldsymbol{R} \mid Ay \leq \boldsymbol{1} u,\ y \geq \boldsymbol{0},\ \boldsymbol{1}^T y = 1\}$ column player

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

The BRP $\tilde{O}$ consists of triplets (y_d, y_e, u) that meet the following conditions:

$$3y_d + 3y_e \leq u \qquad y_d + y_e = 1$$
$$2y_d + 5y_e \leq u \qquad y_d \geq 0,\ y_e \geq 0$$
$$0y_d + 6y_e \leq u$$

52

Vertex enumeration

The BRP $\tilde{O}$ consists of triplets (y_d, y_e, u) that meet the following conditions:

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

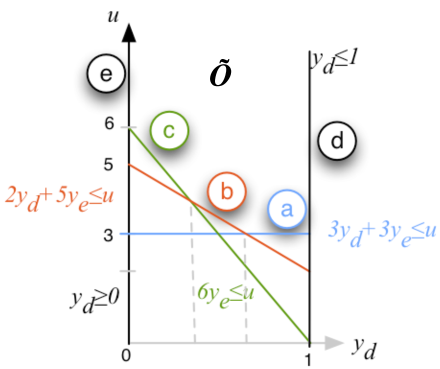
$3y_d + 3y_e \leq u$

$2y_d + 5y_e \leq u$

$0y_d + 6y_e \leq u$

$y_d \geq 0,\ y_e \geq 0$

$y_d + y_e = 1$



The BRP shows which strategy is a best response for player 1 to a mixed profile of player 2

and

when the strategies of player 2 have zero probability

Vertex enumeration

The BRP $\tilde{O}$ consists of triplets (y_d, y_e, u) that meet the following conditions:

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

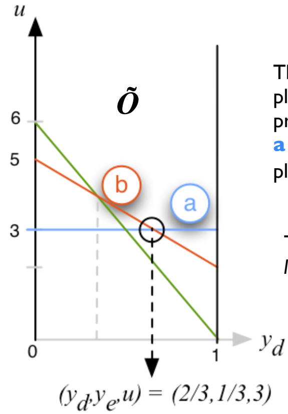
$3y_d + 3y_e \leq u$

$2y_d + 5y_e \leq u$

$0y_d + 6y_e \leq u$

$y_d \geq 0,\ y_e \geq 0$

$y_d + y_e = 1$



The best response for row player to the column player profile (2/3,1/3) are the actions **a** and **b**, which give the row player a payoff of 3

This point is said to be *labelled* by **a** and **b**

The polyhedron $\tilde{N}$ can be produced in a similar manner for the row player.

Vertex enumeration

An equilibrium is pair (x, y) of mixed strategies so that with the corresponding expected payoffs u and v, the pair $((x,v)(y,u))$ in $\tilde{N} \times \tilde{O}$ is completely labelled, meaning that every pure strategy $k \in M \times N$ appears as a label either in (x,v) or in (y,u)

This is equivalent to the best-response condition mentioned earlier

Vertex enumeration

The best-response polyhedron $\tilde{N}$ ($\tilde{O}$) can be simplified by eliminating the payoff value v (u), which can be achieved by dividing the inequalities in $\tilde{N}$ ($\tilde{O}$) by v (u)

$N = \{x \in \mathbf{R}^M \mid B^T x \leq \mathbf{1},\ x \geq \mathbf{0}\}$ row player

$O = \{y \in \mathbf{R}^N \mid Ay \leq \mathbf{1},\ y \geq \mathbf{0}\}$ column player

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$3y_d + 3y_e \leq 1 \qquad 6y_e \leq 1$

$2y_d + 5y_e \leq 1 \qquad y_d \geq 0,\ y_e \geq 0$

Vertex enumeration

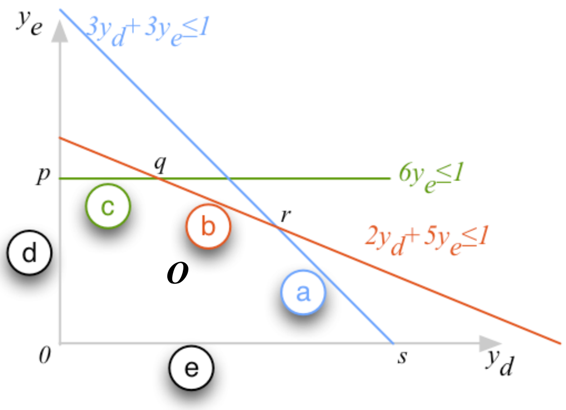
	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$3y_d + 3y_e \leq 1$

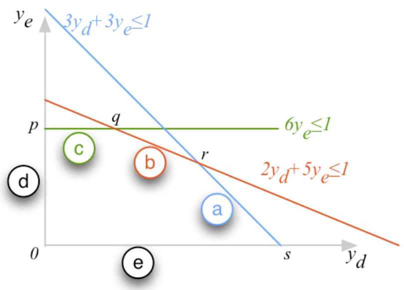
$2y_d + 5y_e \leq 1$

$6y_e \leq 1$

$y_d \geq 0,\ y_e \geq 0$



Vertex enumeration



vertex 0 (0,0) has labels e,d

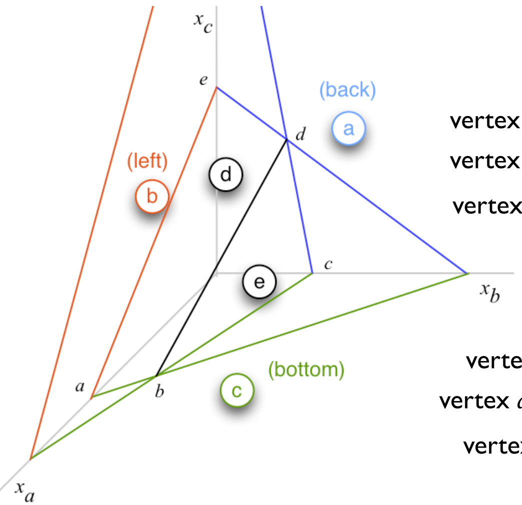
vertex p (0,1/6) has labels c,d

vertex q (1/12,1/6) has labels b,c

vertex r (2/9,1/9) has labels a,b

vertex s (1/3,0) has labels a,e

Vertex enumeration



vertex 0 (0,0,0) has labels a,b,c

vertex a (1/3,0,0) has labels b,c,d

vertex b (2/7,1/14,0) has labels c,d,e

vertex c (0,1/6,0) has labels a,c,e

vertex d (0,1/8,1/4) has labels a,d,e

vertex e (0,0,1/3) has labels a,b,d

Vertex enumeration

Remember :

An equilibrium is pair (x, y) that is completely labelled

row player

0	(0,0,0)	a,b,c
a	(1/3,0,0)	b,c,d
b	(2/7,1/14,0)	c,d,e
c	(0,1/6,0)	a,c,e
d	(0,1/8,1/4)	a,d,e
e	(0,1/3,1/3)	a,b,d

column player

0	(0,0)	e,d
p	(0,1/6)	c,d
q	(1/12,1/6)	b,c
r	(2/9,1/9)	a,b
s	(1/3,0)	a,e

Vertex enumeration

Remember :

An equilibrium is pair *(x, y)* that is completely labelled

But first we need to normalize the values of each vertex to obtain the actual mixed strategies

(a,s)	((1/3,0,0),(1/3,0))	((1,0,0),(1,0))
(b,r)	((2/7,1/14,0),(2/9,1/9))	((4/5,1/5,0),(2/3,1/3))
(d,q)	((0,1/8,1/4),(1/12,1/6))	((0,1/3,2/3),(1/3,2/3))

mixed strategy Nash Equilibria

61

Vertex enumeration

Input : a non-degenerate bi-matrix game, with M and N strategy sets for player 1 and player 2 respectively

Output : All Nash equilibria of the game

```
For each vertex x of N
 For each vertex y of O
   if(x,y) is completely labelled
     store this pair as a Nash equilibrium
     determine mixed strategy by normalization of (x,y)
```

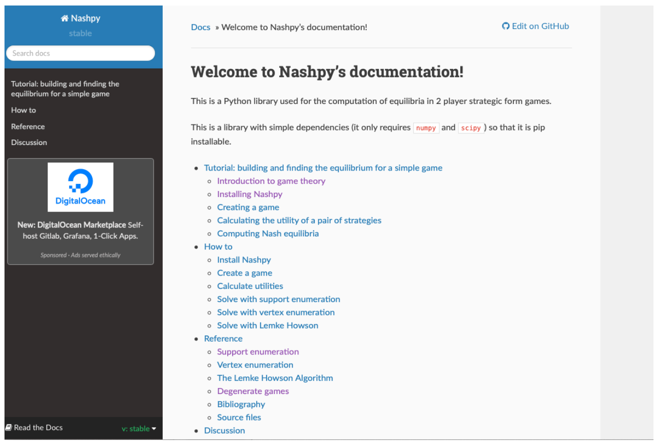
Approach is more efficient than support enumeration

Implement using lexicographic reverse search¶

(¶Corneil, Derek G. (2004), "Lexicographic breadth first search – a survey", Graph-Theoretic Methods in Computer Science, Lecture Notes in Computer Science, 3353, Springer-Verlag, pp. 1–19 and Rose, D. J.; Tarjan, R. E.; Lueker, G. S. (1976), "Algorithmic aspects of vertex elimination on graphs", SIAM Journal on Computing 5 (2): 266–283)

62

Nashpy



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

63

```
(base) Toms-MacBook-Pro-2:~ tlenaert$ python
Python 3.7.1 (default, Dec 14 2018, 13:28:58)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import nashpy as nash
>>> import numpy as np
>>> A=np.array([[3,3],[2,5],[0,6]])
>>> B=np.array([[3,2],[2,6],[3,1]])
>>> mygame=nash.Game(A,B)
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
```

64

```
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
>>> equilibria=mygame.vertex_enumeration()
>>> for eq in equilibria:
...     print(eq)
...
(array([1., 0., 0.]), array([1., 0.]))
(array([3.70074342e-17, 3.33333333e-01, 6.66666667e-01]), array([0.33333333,
0.66666667]))
(array([ 8.00000000e-01,  2.00000000e-01, -7.77156117e-17]), array([0.66666667,
0.33333333]))
```


Other approaches

Two-player games

Lemke-Howson algorithm (1964)
Pivoting

Porter-Nudelman-Shoham algorithm (2004)
Support enumeration

Sandholm-Gilpin-Conitzer algorithm (2005)
Mixed integer-programming approach

Lemke-Howson Algorithm

This algorithm uses the polyhedron approach discussed earlier by following a path (*LH path*) of vertex pairs starting at the *artificial equilibrium* $(0,0)$ and ending at a Nash equilibrium

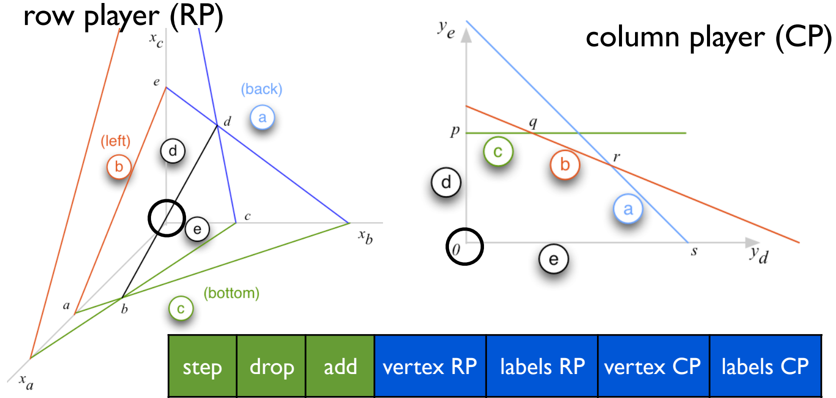
Each vertex in the polyhedra N and O has a number of labels equal to the number of actions (in case of non-degenerate games)

going from one vertex to the next corresponds to dropping one label and picking up another one

as long as there are duplicated labels, this process is continued

Once no labels are duplicated, a Nash Equilibrium is found

Lemke-Howson Algorithm

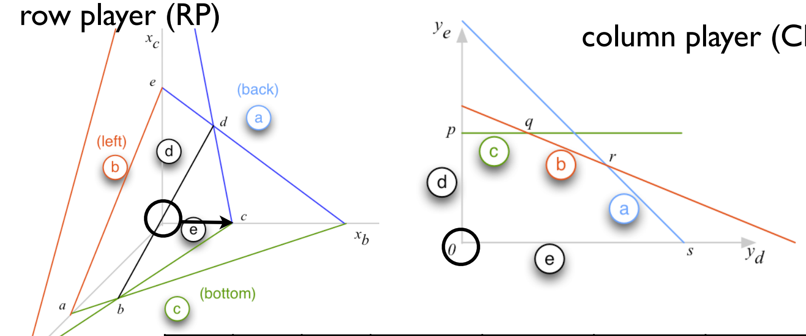


step	drop	add	vertex RP	labels RP	vertex CP	labels CP
0	-	-	0	a,b,c	0	e,d

Lemke-Howson Algorithm

row player (RP)

column player (CP)



x_a

initial free choice

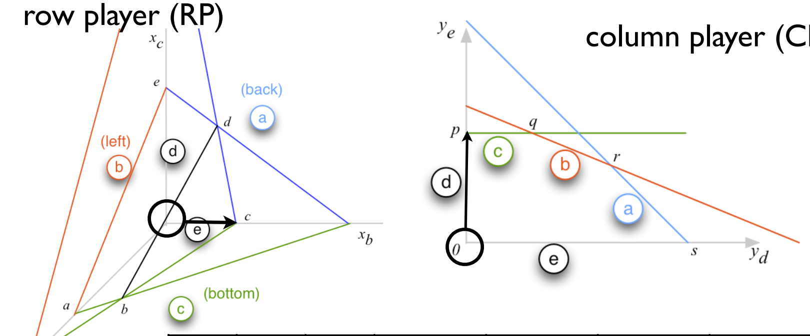
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
0	-	-	*0*	*a,b,c*	*0*	*e,d*
1	*b*	*e*	*c*	*a,c,**e***	*0*	***e**,d*

69

Lemke-Howson Algorithm

row player (RP)

column player (CP)



x_a

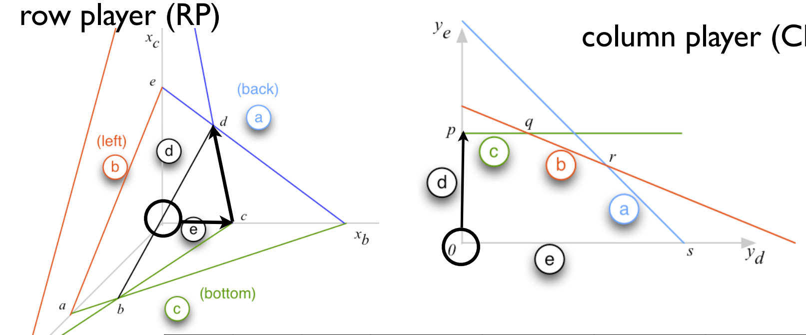
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
1	*b*	*e*	*c*	*a,c,**e***	*0*	***e**,d*
2	*e*	*c*	*c*	*a,**c**,e*	*p*	***c**,d*

70

Lemke-Howson Algorithm

row player (RP)

column player (CP)



x_a

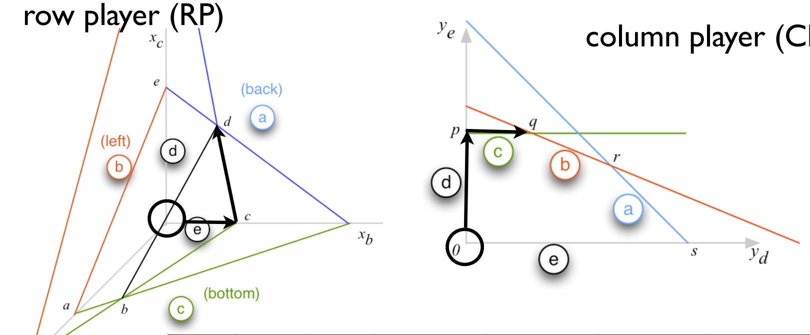
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
2	*e*	*c*	*c*	*a,**c**,e*	*p*	***c**,d*
3	*c*	*d*	*d*	*a,**d**,e*	*p*	*c,**d***

71

Lemke-Howson Algorithm

row player (RP)

column player (CP)



x_a

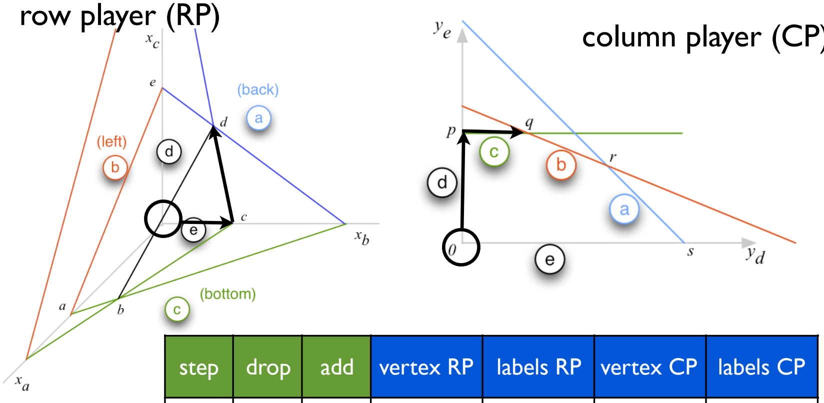
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
3	*c*	*d*	*d*	*a,**d**,e*	*p*	*c,**d***
4	*d*	*b*	*d*	*a,d,e*	*q*	*b,c*

72-1

Lemke-Howson Algorithm

row player (RP)

column player (CP)



step	drop	add	vertex RP	labels RP	vertex CP	labels CP
3	c	d	d	a,**d**,e	p	c,**d**
4	d	b	d	a,d,e	q	b,c

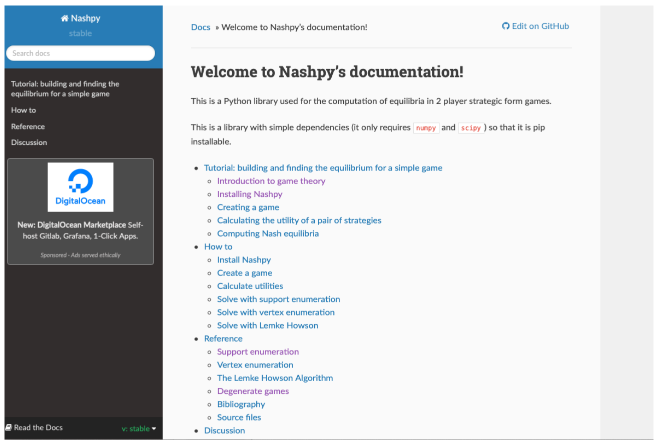
Lemke-Howson Algorithm

Input : a non-degenerate bi-matrix game, with M and N strategy sets for player 1 and player 2 respectively

Output : **One** Nash equilibrium of the game

```
Choose k ∈ M ∪ N, called missing label
Let (x,y) = (0,0)∈ N×O
Drop label k (from x in N if k∈M, from y in M if k∈N)
Loop {
  Call the new vertex pair (x,y)
  l is the label that is picked up
  if (l=k), break loop
  drop l in the other polytope
} //end loop
report nash (x,y), once rescaled to mixed strategy
```


Nashpy



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


```
(base) Toms-MacBook-Pro-2:~ tlenaert$ python
Python 3.7.1 (default, Dec 14 2018, 13:28:58)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import nashpy as nash
>>> import numpy as np
>>> A=np.array([[3,3],[2,5],[0,6]])
>>> B=np.array([[3,2],[2,6],[3,1]])
>>> mygame=nash.Game(A,B)
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
```

```
>>> mygame
Bi matrix game with payoff matrices:

Row player:
[[3 3]
 [2 5]
 [0 6]]

Column player:
[[3 2]
 [2 6]
 [3 1]]
>>> equilibria=mygame.lemke_howson_enumeration()
>>> for eq in equilibria:
...     print(eq)
...
(array([1., 0., 0.]), array([1., 0.]))
(array([0.        , 0.33333333, 0.66666667]), array([0.33333333, 0.66666667]))
(array([1., 0., 0.]), array([1., 0.]))
(array([1., 0., 0.]), array([1., 0.]))
(array([0.        , 0.33333333, 0.66666667]), array([0.33333333, 0.66666667]))
```

?

Lemke-Howson Algorithm

Note that the algorithms always terminates, given that there are only finitely many vertex pairs

The path can start at any Nash equilibrium !!
Hence one can use this approach to find all Nash Equilibria

An efficient implementation of this algorithm uses pivoting as used by the simplex algorithm for solving a linear program.

Pivoting

The previous polyhedron constraints are now represented as linear equations with non-negative slack variables ($s \in \boldsymbol{R}^N$ and $r \in \boldsymbol{R}^M$) redefining them as follows:

$$N = \{x \in \boldsymbol{R}^M \mid B^T x + s = \boldsymbol{1},\ x \geq \boldsymbol{0},\ s \geq \boldsymbol{0}\}$$

$$O = \{y \in \boldsymbol{R}^N \mid r + Ay = \boldsymbol{1},\ y \geq \boldsymbol{0},\ r \geq \boldsymbol{0}\}$$

A basic solution is given by n *basic columns* of $B^T x + s = 1$ and m *basic rows* of $r + Ay = 1$

A feasible solution is a basic solution that also meets $x \geq 0$, $s \geq 0$, $y \geq 0$ and $r \geq 0$, and defines a vertex x of N and y of O. The labels are given by the *non-basic columns*

Pivoting

Visualizing basic and non-basic columns in the example

row player (RP)

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$$\begin{array}{lllllll} 3x_a + & 2x_b + & 3x_c + & s_d & & = 1 \\ 2x_a + & 6x_b + & x_c + & & s_e & = 1 \end{array}$$

column player (CP)

$$\begin{array}{lllllll} r_a + & & & 3y_d + & 3y_e & = 1 \\ & r_b + & & 2y_d + & 5y_e & = 1 \\ & & r_c + & 0y_d + & 6y_e & = 1 \end{array}$$

Pivoting

Visualizing basic and non-basic columns in the example

row player (RP)

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$$\begin{array}{llllll} 3x_a + & 2x_b + & 3x_c + & s_d & & =1 \\ 2x_a + & 6x_b + & x_c + & & s_e & =1 \end{array}$$

column player (CP)

$$\begin{array}{llllll} r_a + & & & 3y_d + & 3y_e & =1 \\ & r_b + & & 2y_d + & 5y_e & =1 \\ & & r_c + & 0y_d + & 6y_e & =1 \end{array}$$

79-2

Pivoting

Visualizing basic and non-basic columns in the example

row player (RP)

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$$\begin{array}{llllll} 3x_a + & 2x_b + & 3x_c + & s_d & & =1 \\ 2x_a + & 6x_b + & x_c + & & s_e & =1 \end{array}$$

column player (CP)

$$\begin{array}{llllll} r_a + & & & 3y_d + & 3y_e & =1 \\ & r_b + & & 2y_d + & 5y_e & =1 \\ & & r_c + & 0y_d + & 6y_e & =1 \end{array}$$

79-3

Pivoting

Visualizing basic and non-basic columns in the example

row player (RP)

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$$\begin{array}{llllll} 3x_a + & 2x_b + & 3x_c + & s_d & & =1 \\ 2x_a + & 6x_b + & x_c + & & s_e & =1 \end{array}$$

column player (CP)

$$\begin{array}{llllll} r_a + & & & 3y_d + & 3y_e & =1 \\ & r_b + & & 2y_d + & 5y_e & =1 \\ & & r_c + & 0y_d + & 6y_e & =1 \end{array}$$

79-4

Pivoting

Visualizing basic and non-basic columns in the example

row player (RP)

	d	e
a	3, 3	3, 2
b	2, 2	5, 6
c	0, 3	6, 1

$$\begin{array}{llllll} 3x_a + & 2x_b + & 3x_c + & s_d & & =1 \\ 2x_a + & 6x_b + & x_c + & & s_e & =1 \end{array}$$

column player (CP)

$$\begin{array}{llllll} r_a + & & & 3y_d + & 3y_e & =1 \\ & r_b + & & 2y_d + & 5y_e & =1 \\ & & r_c + & 0y_d + & 6y_e & =1 \end{array}$$

79-5

Pivoting

Pivoting is a change of the basis, where a non-basic variable *enters* (pick-up) and a basic variables *leaves* (drop) the set of basic variables, while making sure that the solution remains feasible.

Let's illustrate the LH path to (d,q). The initial variable we want to pick-up is x_b.

Step 1: select the pivot element in x_b the column

Determine the minimum ratio; $x_b \leq 1/2$ or $x_b \leq 1/6$

pivot column ↓

$$
\begin{array}{llllll}
3x_a + & 2x_b + & 3x_c + & s_d & & =1 \\
2x_a + & \boxed{6x_b} + & x_c + & & \boxed{s_e} & =1
\end{array}
$$

pivot element enters the basis

leaves the basis

pivot?
$s_d = 1\text{-}2x_b$
$s_e = 1\text{-}6x_b$
we require that $s_d \geq 0$, $s_e \geq 0$, $x_b \geq 0$

Pivoting

Step 2: multiply other rows by pivot element

$$
\begin{array}{llllll}
18x_a + & 12x_b + & 18x_c + & 6s_d & & =6 \\
2x_a + & \boxed{6x_b} + & x_c + & & \boxed{s_e} & =1
\end{array}
$$

pivot row→

pivot element enters the basis

leaves the basis

Step 3: subtract multiples of the pivot row from the other rows to obtain zero entries in the pivot column

$$
\begin{array}{llllll}
14x_a + & & 16x_c + & 6s_d - & 2s_e & =4 \\
2x_a + & 6x_b + & x_c + & & s_e & =1
\end{array}
$$

pivot row→

Pivoting

The process shown here corresponds to Integer Pivoting (all coefficients are kept integers)

The process finishes when we try to remove a non-basic column which was already removed before

The pivoting of N removes s_e from the basis so now we need to examine O to see which other variable leaves the basis

Step 1: select the pivot element in y_e the column

pivot column ↓

$$
\begin{array}{llllll}
r_a + & & & 3y_d + & 3y_e & =1 \\
& r_b + & & 2y_d + & 5y_e & =1 \\
& & \boxed{r_c +} & & \boxed{6y_e} & =1
\end{array}
$$

leaves the basis

pivot element enters the basis

pivot?
$r_a = 1\text{-}3y_e$
$r_b = 1\text{-}5y_e$
$r_c = 1\text{-}6y_e$
we require that $r_a \geq 0$, $r_b \geq 0$, $r_c \geq 0$, $y_e \geq 0$

Pivoting

Step 2: multiply other rows by pivot element

$$
\begin{array}{llllll}
6r_a + & & & 18y_d + & 18y_e & =6 \\
& 6r_b + & & 12y_d + & 30y_e & =6 \\
& & \boxed{r_c +} & & \boxed{6y_e} & =1
\end{array}
$$

pivot row→

leaves the basis

enters the basis

Step 3: subtract multiples of the pivot row from the other rows to obtain zero entries in the pivot column

$$
\begin{array}{llllll}
6r_a - & & 3r_c + & 18y_d & & =3 \\
& 6r_b - & 5r_c + & 12y_d & & =1 \\
& & r_c + & & 6y_e & =1
\end{array}
$$

pivot row→

Pivoting

The pivoting of O removes r_c (which was not removed before) from the basis so now we need to examine N to see which other variable leaves the basis

Step 1: select the pivot element in x_c the column

enters the basis
pivot element
leaves the basis

$$14x_a + \boxed{16x_c} + \boxed{6s_d} - 2s_e = 4$$

$$2x_a + 6x_b + x_c + s_e = 1$$

pivot column ↑

pivot?
$6s_d = 4 - 16x_c$
$6x_b = 1 - x_c$
we require that $s_d \geq 0$, $x_b \geq 0$, $x_c \geq 0$

Step 2: multiply other rows by pivot element

enters the basis leaves the basis

pivot row→ $14x_a + \boxed{16x_c} + \boxed{6s_d} - 2s_e = 4$

$$32x_a + 96x_b + 16x_c + 16s_e = 16$$

Pivoting

Step 3: subtract multiples of the pivot row from the other rows to obtain zero entries in the pivot column

enters the basis leaves the basis

pivot row→ $14x_a + \boxed{16x_c} + \boxed{6s_d} - 2s_e = 4$

$$18x_a + 96x_b - 6s_d + 18s_e = 12$$

Step 4: reduce coefficients, divide by previous pivot (6)

pivot row→ $14x_a + 16x_c + 6s_d - 2s_e = 4$

$$3x_a + 16x_b - s_d + 3s_e = 2$$

Pivoting

The pivoting of N removes s_d (which was not removed before) from the basis so now we need to examine O to see which other variable leaves the basis

Step 1: select the pivot element in x_c the column

$$6r_a - 3r_c + 18y_d = 3$$

$$\boxed{6r_b} - 5r_c + \boxed{12y_d} = 1$$

$$r_c + 6y_e = 1$$

pivot element
leaves the basis
enters the basis
pivot column ↑

pivot?
$6r_a = 3 - 18y_d$
$6r_b = 1 - 12y_d$
$6y_e = 1$
we require that $r_a \geq 0$, $r_b \geq 0$, $y_d \geq 0$, $y_e \geq 0$

Step 2: multiply other rows by pivot element

$$72r_a - 36r_c + 216y_d = 36$$

pivot row→ $\boxed{6r_b} - 5r_c + \boxed{12y_d} = 1$

$$12r_c + 6y_e = 1$$

Pivoting

Step 3: subtract multiples of the pivot row from the other rows to obtain zero entries in the pivot column

$$72r_a - 108r_b + 54r_c + \quad = 18$$

pivot row→ $\boxed{6r_b} - 5r_c + \boxed{12y_d} = 1$

$$12r_c + 6y_e = 1$$

Step 4: reduce coefficients, divide by previous pivot (6)

$$12r_a - 18r_b + 9r_c + \quad = 3$$

pivot row→ $6r_b - 5r_c + 12y_d = 1$

$$12r_c + 6y_e = 1$$

Pivoting

So r_b is leaving the basis now ... but this is the column we started with!

row player (RP)

$$14x_a + 16x_c + 6s_d - 2s_e = 4$$
$$3x_a + 16x_b - s_d + 3s_e = 2$$

so x_b and x_c are part of the equilibrium with values $x_b=1/8$ and $x_c=1/4$

the labels are a,d,e

column player (CP)

$$12r_a - 18r_b + 9r_c + = 3$$
$$6r_b - 5r_c + 12y_d = 1$$
$$12r_c + 6y_e = 1$$

so y_d and y_e are part of the equilibrium with values $y_d=1/12$ and $y_e=1/6$

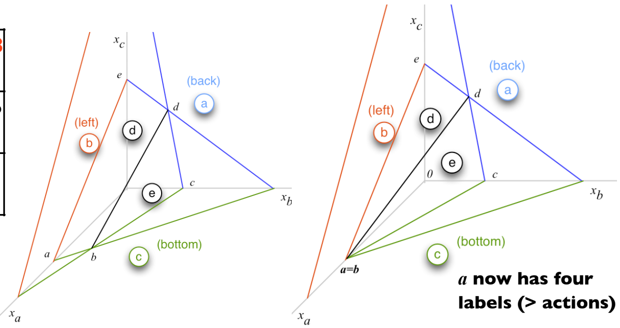
the labels are b,c

This solution corresponds to vertex pair (d,q)

degenerate games

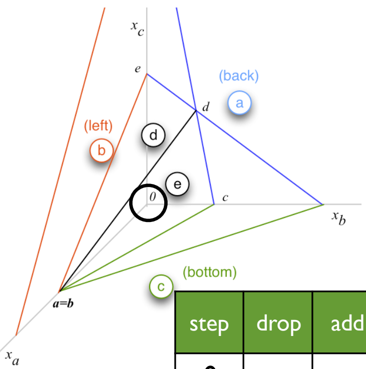
If the game is degenerate then the LH path is no longer unique, since a vertex may have more than the allowed number of labels (the number of actions)

	d	e
a	3, 3	3, 3
b	2, 2	5, 6
c	0, 3	6, 1

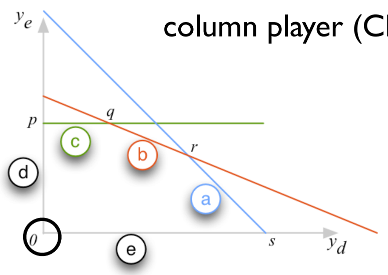


Lemke-Howson Algorithm

row player (RP)



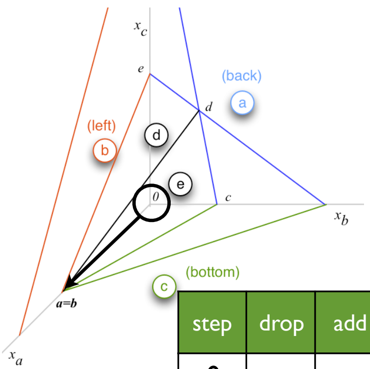
column player (CP)



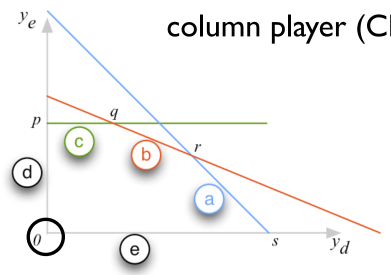
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
0	-	-	0	a,b,c	0	e,d

Lemke-Howson Algorithm

row player (RP)



column player (CP)

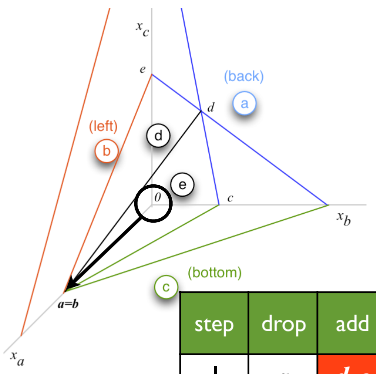


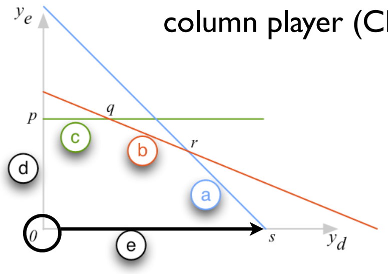
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
0	-	-	0	a,b,c	0	e,d
1	a	d,e	c	b,c,d,e	0	e,d

Lemke-Howson Algorithm

row player (RP)

column player (CP)





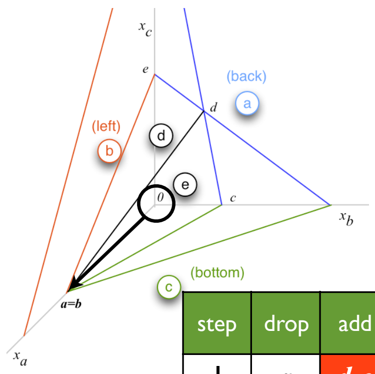
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
1	*a*	***d,e***	*a*	*b,c,**d,e***	*0*	***e,d***
2	*d*	***a***	*a*	*b,c,**d,e***	*s*	*a,**e***

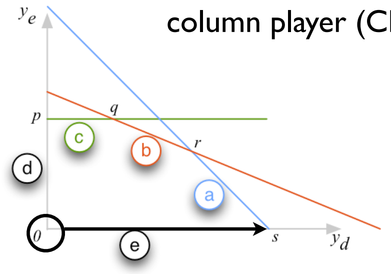
92-1

Lemke-Howson Algorithm

row player (RP)

column player (CP)





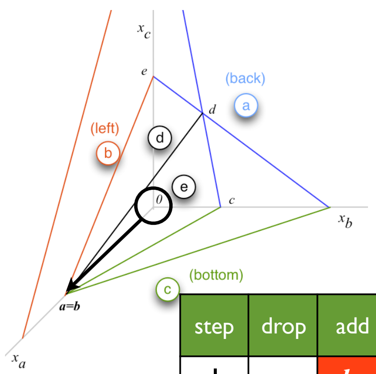
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
1	*a*	***d,e***	*a*	*b,c,**d,e***	*0*	***e,d***
2	*d*	***a***	*a*	*b,c,**d,e***	*s*	*a,**e***

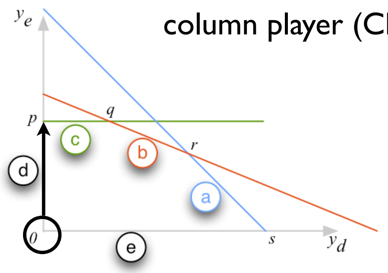
92-2

Lemke-Howson Algorithm

row player (RP)

column player (CP)





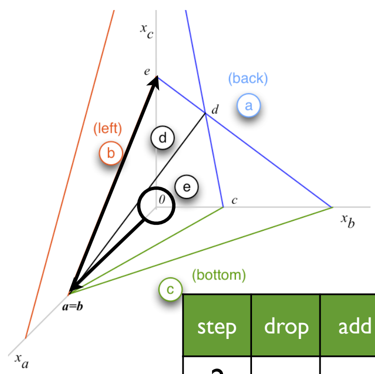
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
1	*a*	***d,e***	*a*	*b,c,**d,e***	*0*	***e,d***
2	*e*	*c*	*a*	*b,**c,d**,e*	*p*	***c,d***

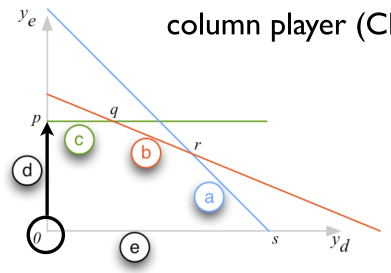
93

Lemke-Howson Algorithm

row player (RP)

column player (CP)





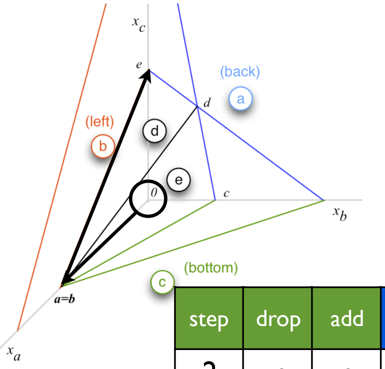
step	drop	add	vertex RP	labels RP	vertex CP	labels CP
2	*e*	*c*	*a*	*b,**c,d**,e*	*p*	***c,d***
3	*c*	***a***	*e*	*a,b,**d***	*p*	*c,**d***

NO *e*

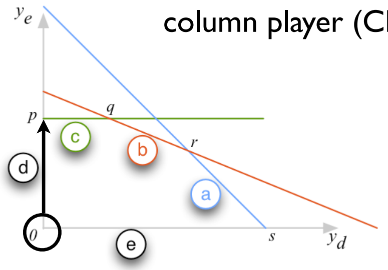
94-1

Lemke-Howson Algorithm

row player (RP)



column player (CP)

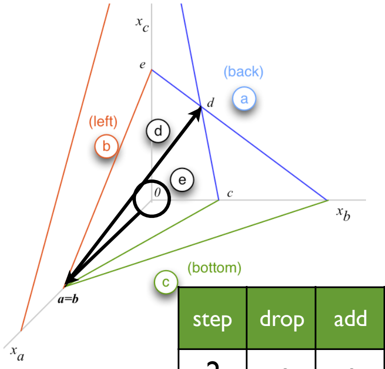


	step	drop	add	vertex RP	labels RP	vertex CP	labels CP
	2	*e*	*c*	*a*	*b*,***c***,***d***,*e*	*p*	***c***,***d***
NO *e*	3	*c*	***a***	*e*	*a*,*b*,***d***	*p*	*c*,***d***

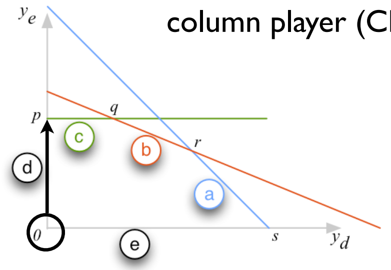
94-2

Lemke-Howson Algorithm

row player (RP)



column player (CP)

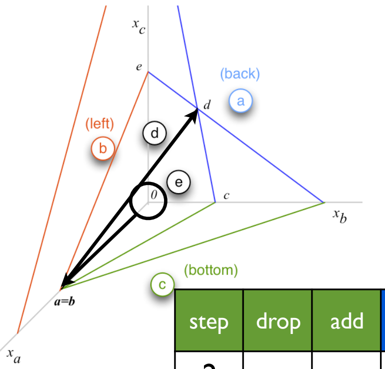


	step	drop	add	vertex RP	labels RP	vertex CP	labels CP
	2	*e*	*c*	*a*	*b*,***c***,***d***,*e*	*p*	***c***,***d***
NO *b*	3	*c*	***a***	*d*	*a*,***d***,*e*	*p*	*c*,***d***

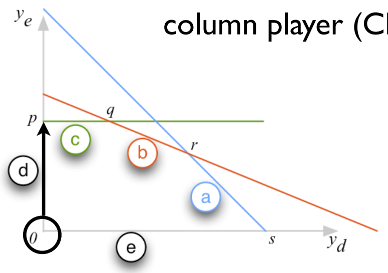
95-1

Lemke-Howson Algorithm

row player (RP)



column player (CP)



	step	drop	add	vertex RP	labels RP	vertex CP	labels CP
	2	*e*	*c*	*a*	*b*,***c***,***d***,*e*	*p*	***c***,***d***
NO *b*	3	*c*	***a***	*d*	*a*,***d***,*e*	*p*	*c*,***d***

95-2

degenerate games

Degeneracy can be resolved by perturbing the system lexicographically

$N = \{x \in \boldsymbol{R}^M |\ B^T x \leq \boldsymbol{1} + \varepsilon,\ x \geq \boldsymbol{0},\ \varepsilon \geq \boldsymbol{0}\}$ row player

$O = \{y \in \boldsymbol{R}^N |\ Ay \leq \boldsymbol{1} + \varepsilon,\ y \geq \boldsymbol{0},\ \varepsilon \geq \boldsymbol{0}\}$ column player

see Codenotti B, De Rossi S and Pagan M (2008) An experimental analysis of Lemke-Howson Algorithm. (arXiv: 0811.3247v1) for an in depth description on how to implement the algorithm

96