451: Game Theory

G. Miller, K. Sutner
Carnegie Mellon University
2020/11/12

Game theory is the mathematical study of the strategic behaviour of multiple, rational agents (the rationality requirement seems to leave out humans).

Developed initially by economists (much like LP), but now has solid mathematical foundations; applications in computer science.

Players participants in the game.

Actions that are available to the players.
All players make their move simultaneously.

Payoff the resulting "value" of actions taken.

Alas, this has nothing to do with interactive/recreational games.

- There are n players.
- Player i has a finite set of possible actions A_i.
- A play is a vector $a \in \times A_i$.
- A payoff function determines a value for each play and player.

We will focus on single-round games with just 2 players. Moreover, we consider matrix games, where the payoff function is given by a matrix.

Question: Which strategy should each player use to maximize their payoff?

For pure strategies (always pick the same action) this is just combinatorics. But for mixed strategies (pick a probability distribution over the actions) things become interesting.

- A profile is vector of n strategies, either pure or mixed.

There are two players that we will call the row player and the column player. Each player has two actions available, $\uparrow$ and $\downarrow$. The payoff is determined by the following matrix (actually, tensor):

$$M = \begin{array}{c|cc} & \downarrow & \uparrow \\ \hline \downarrow & (1,-1) & (-1,1) \\ \uparrow & (-1,1) & (1,-1) \end{array}$$

If both players choose the same action, row wins, column loses. If they pick different actions, it's the other way around. Of course, neither one knows what the other will do.

$M_{ij} = (\alpha, \beta)$ means: if choices $i \in A_r$ and $j \in A_c$ occur, the payoff for the row player is α, the payoff for the column player is β.

If you like semantic sugar, think of the contest between a shooter and a goalie during a penalty kick.

The goalie example has the important property that $\alpha = -\beta$ everywhere. This is called a zero-sum game.

For 2 players, we can think of M as being composed of two matrices R and C, so in zero-sum game $R = -C$.

By contrast, in a general-sum game we drop that condition. Example:

$$M = \begin{pmatrix} (1,1) & (-1,-1) \\ (-1,-1) & (1,1) \end{pmatrix}$$

This is a variant of the well-know "game of chicken," with potentially fatal consequences: here both players swerve either left or right (similar to the goalie game).

Note that one can turn a non-zero sum game of n players into a zero-sum game of $n+1$ players.

A well-known example where both players have 3 available actions is rock-paper-scissors. The R matrix here looks like so:

$$\begin{pmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix}$$

Clearly any player adopting a pure strategy is doomed in this game (at least in the iterated version): the other player can always insure payoff 1.

It is far less clear what happens if both players adopt a mixed strategy.

So the row-player picks some probability distribution p over the rows (corresponding to the actions A_r), and the column-player picks another probability distribution q over the columns (the actions A_c)

What is the expected payoff for each player?

$$V_R(p,q) = \sum_{ij} \text{Pr[play } ij] R_{ij} = \sum_{ij} p_i q_j R_{ij}$$

Similarly $V_C(p,q) = \sum_{ij} p_i q_j C_{ij}$.

For example, for the Goalie game we have $V_R((1/3, 2/3), (2/3, 1/3)) = 1/9$.

This notation is annoyingly redundant, it is better to streamline things.

For simplicity, parametrize the payoff function like so:

$$V(x, y) = V_R((x, 1 - x), (y, 1 - y))$$

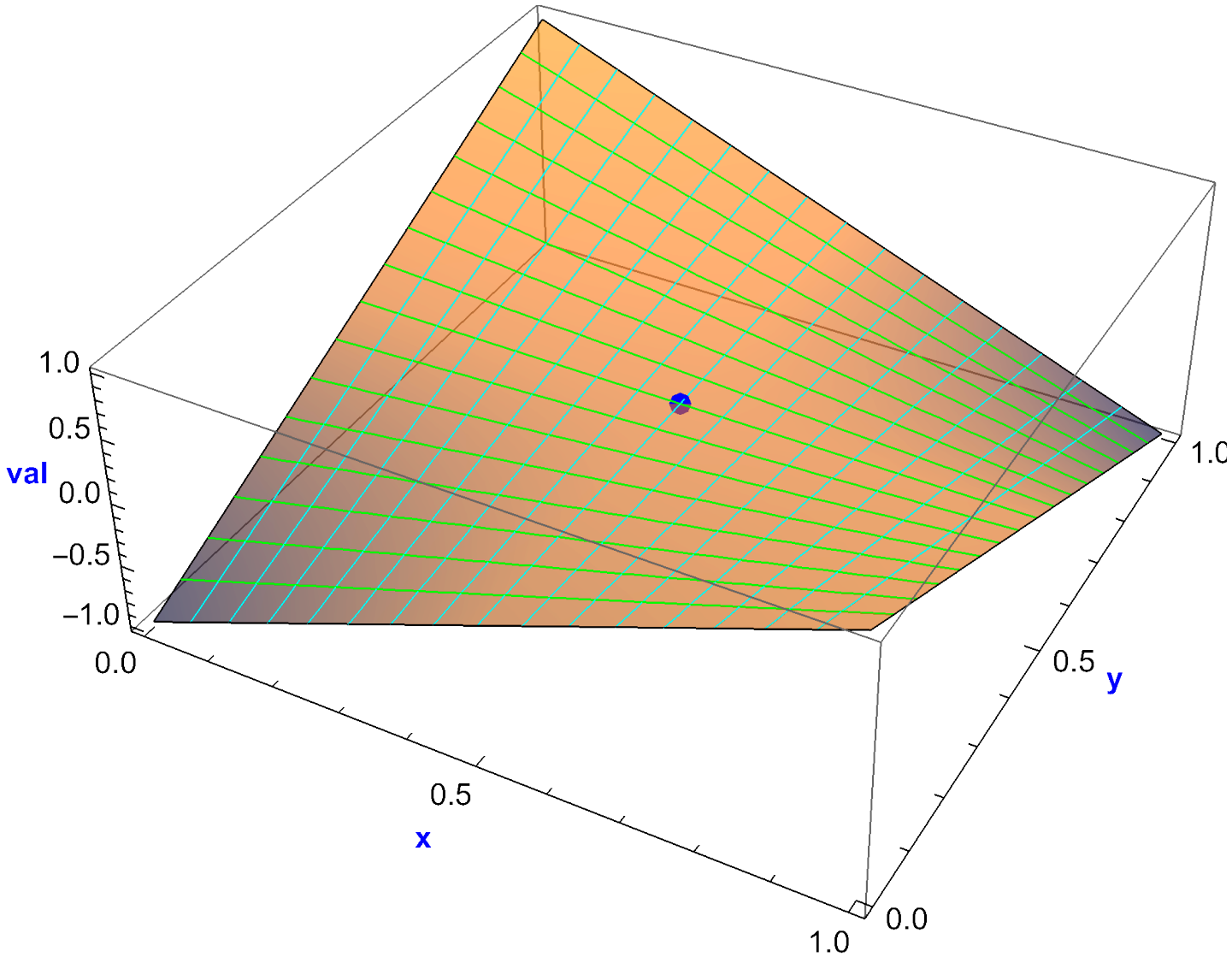
where $0 \leq x, y \leq 1$.

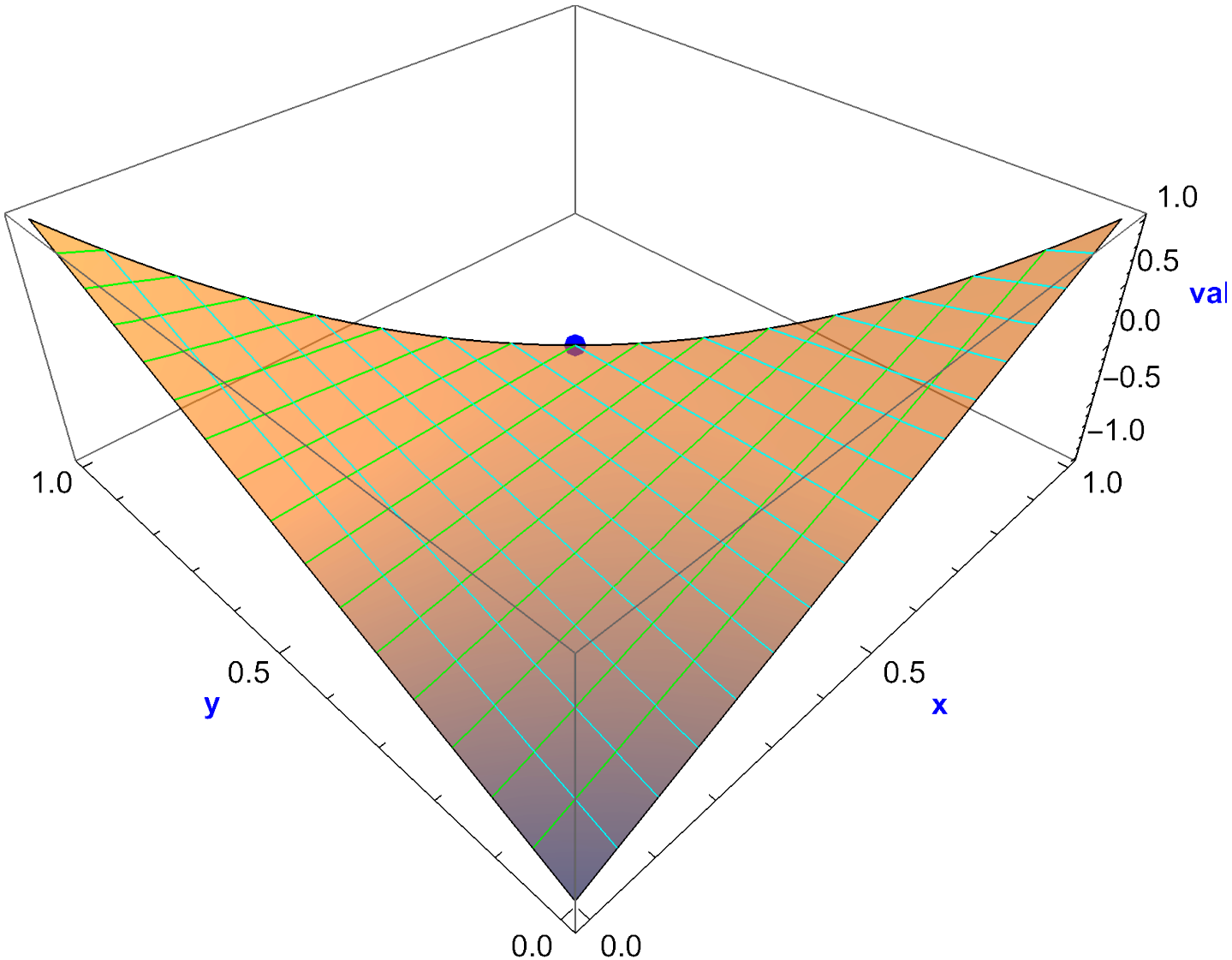
Note that $V(x, y)$ is just a polynomial in x and y, so we can easily plot pictures of the value surface.

For example, for Goalie, we have the quadratic polynomial

$$V(x, y) = -1 + 2x + 2y - 4xy$$

It's easy to see that $V([0, 1], [0, 1]) = [-1, 1]$.


1.0
0.5
val
0.0
−0.5
−1.0
0.0
0.5
x
1.0
0.0
0.5
y
1.0


1.0
0.5
val
0.0
−0.5
−1.0
1.0
1.0
0.5
y
0.5
x
0.0
0.0

Given a strategy q by the column-player, the row-player can always try to maximize the payoff. Hence we have the following lower bound on the row payoff:

$$\text{lb} = \max_p \min_q V_R(p, q)$$

An analogous argument works for the column-player. Since we are zero-sum, we also get an upper bound:

$$\begin{aligned} \max_q \min_p V_C(p, q) &= \max_q \min_p -V_R(p, q) \\ &= -\min_q \max_p V_R(p, q) \end{aligned}$$

Hence for the row-player we get

$$\text{ub} = \min_q \max_p V_R(p, q)$$

Recall that for Goalie we have a value polynomial

$$\begin{aligned} V(x,y) &= -4xy + 2x + 2y - 1 \\ &= (-1 + 2y) + (2 - 4y)x \\ &= (-1 + 2x) + (2 - 4x)y \end{aligned}$$

Right choice for row player, depending on y:

val	x	y	
1	1	$< 1/2$	
0	$*$	$= 1/2$	don't care
1	0	$> 1/2$	

Therefore lb $= 0$. By symmetry, ub $= 0$. The value of the game is 0.

OK, but what if the game matrix is less symmetric?

Let

$$R = \begin{pmatrix} -1/2 & 1 \\ 1 & -1 \end{pmatrix}$$

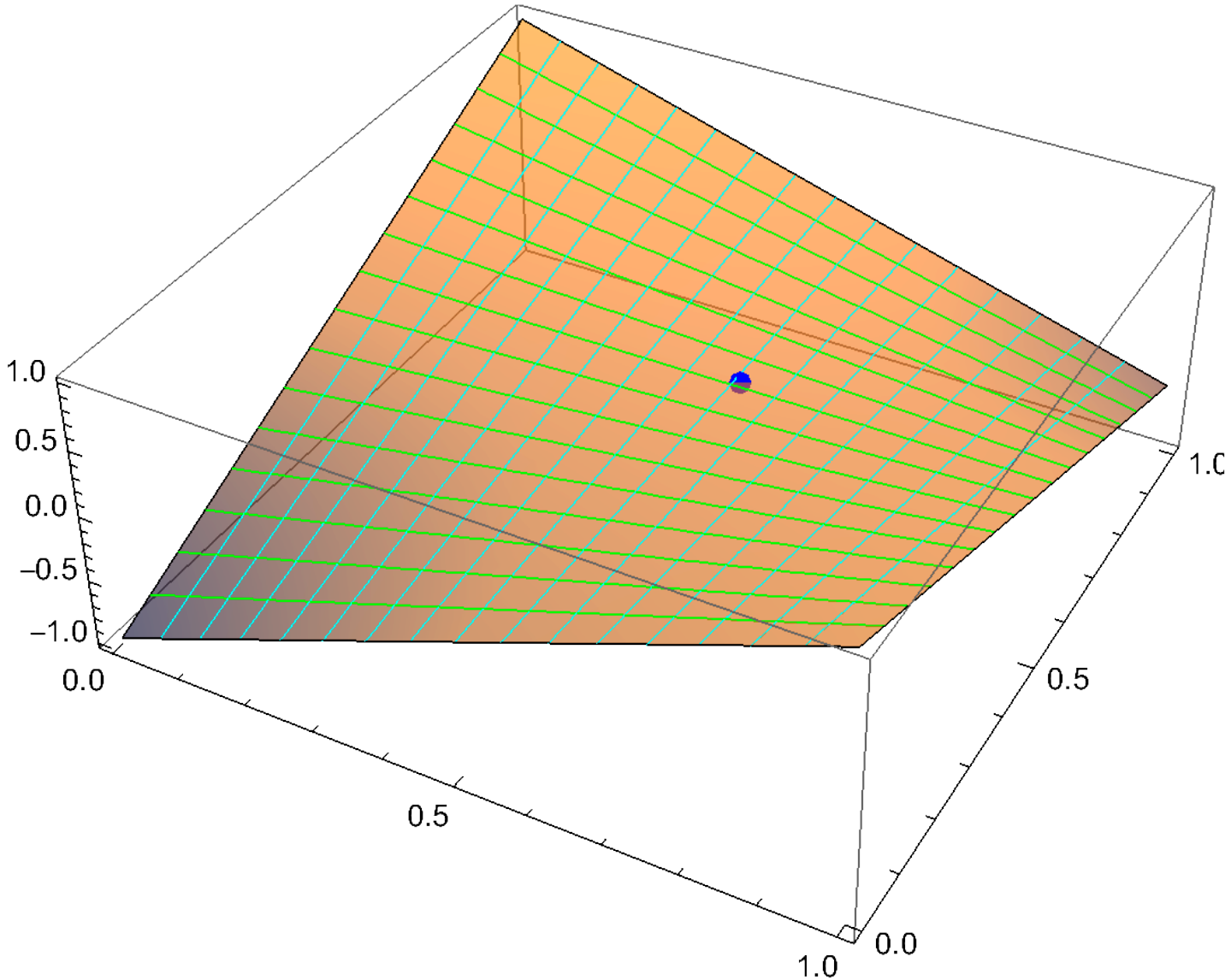
Here

$$V(x, y) = -7/2\, xy + 2x + 2y - 1$$

Still good, the polynomial is symmetric and the value of the game is lb = ub = $1/7$.

Exercise

Check the arithmetic.


1.0
0.5
0.0
−0.5
−1.0
0.0
0.5
1.0
0.0
0.5
1.0

We can push a bit further by keeping two actions for the row-player, but allowing more for the column-player, so $R \in \mathbb{R}^{2\times n}$, say

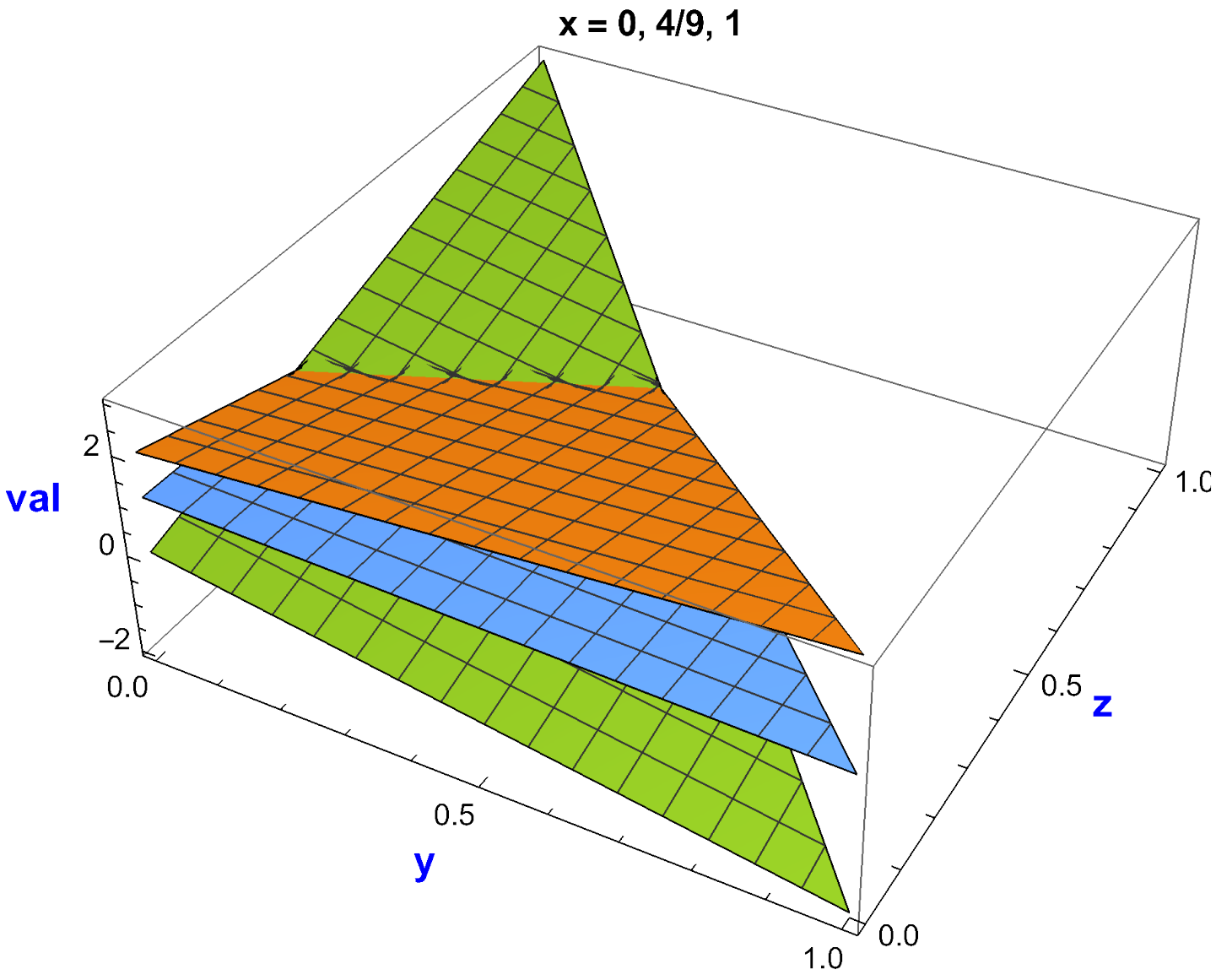
$$R = \begin{pmatrix} -2 & 3 & 0 \\ 3 & -1 & 2 \end{pmatrix}$$

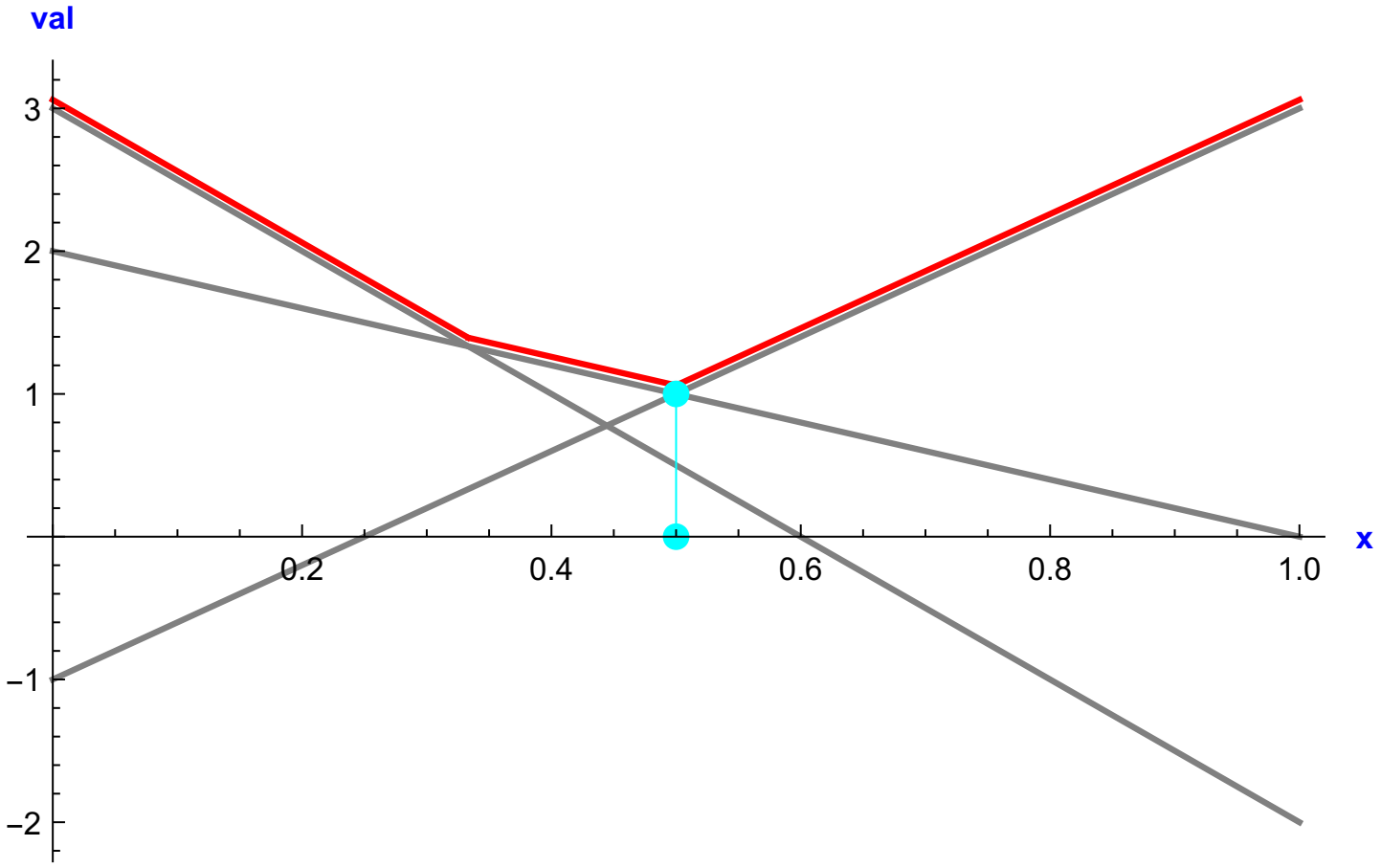
Here the value polynomial takes the form

$$\begin{aligned} V(x,y,z) &= 2 - 2x + y - 3xy - 3z + 6xz \\ &= (2-2x) + (1-3x)y + (-3+6x)z \\ &= (2-y-3z) + (-2-3y+6z)x \end{aligned}$$

We need to determine the optimal strategies for the row and column player.

For example, for the row player the sign of $-2-3y+6z$ is critical.


x = 0, 4/9, 1
val
2
0
−2
0.0
0.5
1.0
y
0.0
0.5
1.0
z



The three pure strategies of the column player: $3 - 5x$, $-1 + 4x$, $2 - 2x$.

$$V(x, y, z) = 2 - 2x + y - 3xy - 3z + 6xz$$

Row Player

val	x	y, z	
$2y - 3z$	0	$z < 1/3 + 1/2\, y$	row 2
$1 - y/2$	$*$	$z = 1/3 + 1/2\, y$	don't care
$-2y + 3z$	1	$z > 1/3 + 1/2\, y$	row 1

Column Player

val	y	z	x
$3 - 5x$	1	0	$x < 1/3$
$4/3 - z$	$*$	0	$x = 1/3$
$2 - 2x$	0	0	$x < 1/2$
$-y/2$	0	$*$	$x = 1/2$
$-1 + 4x$	0	1	$x > 1/2$

Supposedly, von Neumann immediately proposed the concept of duality for LP in a conversation with Dantzig in 1947. This may not be as surprising considering the following result.

Theorem (Von Neumann, 1928)

In any zero-sum game we have lb $=$ ub*.*

We'll have more to say about the connection between linear programming and 2-person games in a while.

The players (crooks undergoing separate interrogations) have a choice to remain silent or talk (rat out the other), with payoff matrix

	s	t
s	$(3,3)$	$(0,5)$
t	$(5,0)$	$(1,1)$

This shows utility: $u = 5 -$ years in jail. More abstractly, the payoff is

$$\begin{array}{cc} (R,R) & (S,T) \\ (T,S) & (P,P) \end{array}$$

with the meaning R reward, P punishment, S sucker, T temptation.

Things get interesting when

$$T > R > P > S$$

Now suppose you have n players that pair off against each other (round robin) multiple times. They all can remember their history, so they can adjust their strategy in the next encounter according to past behaviour of the opponent.

Political scientist R. Axelrod conducted such tournaments in 1979, inviting submissions of strategies.

Silent	always remain silent.
Talk	always rat out the other.
Tit-for-Tat	start silent, then mimic the opponent.
Joss	silent with probability 0.9 after silence, talk after talk.
Grofman	silent unless last time actions were different; then silent with probability $2/7$.

Surprising Fact: unsophisticated Tit-for-Tat tends to do very well in IPD.

Suppose you have a collection A_j, $j = 1, \ldots, n$, of deterministic algorithms for some task, as well as a list I_i, $i = 1, \ldots, m$, of inputs (of some fixed size). Define

$$R_{ij} = \text{running time of } A_j \text{ on } I_i$$

For example, the A_j could be sorting algorithms, and the I_i all permutations of $[k]$ (so $m = k!$).

Thinking of (R_{ij}) as a game matrix is mildly helpful.

We can think of column j (algorithm A_j) does well against all rows (all inputs) iff algorithm A_j has good worst-case behavior.

A randomized algorithm can be viewed as a probability distribution over the columns (given a suitable selection of the algorithms).

- The expected running time of a randomized algorithm is good if the mixed column strategy does well against all row strategies.
- To find the best randomized algorithm we need to find the best colum strategy.
- To obtain lower bounds for randomized algorithms, we can look for a row strategy such that the expected cost of any column strategy is high.

It is well-known that comparison-based sorting requires $\Omega(n \log n)$ steps on some inputs. Here is a generalization.

Lemma

Every randomized comparison-based sorting algorithm takes $\Omega(n \log n)$ steps in expectation.

Proof.

Fix n and enumerate all deterministic algorithms that sort length n lists, represented by a decision tree (leaves labeled by permutations of $[n]$). Form a matrix R whose columns are these algorithms, and whose rows are the permutations. Let R_{ij} be the number of comparisons of algorithm j on input i. Take the uniform distribution over the rows.

Each decision tree has at least $n!$ leaves leading to a depth of $d = \Omega(n \log n)$ since $\log n! = n \log n - n + O(\log n)$. But then the fraction of leaves at depth at most $d - c$ is at most 2^{-c}.

□

Question: How does one solve a game?

We are supposedly dealing with rational agents, so everybody would prefer a stable situation: given a profile, none of the players sees any advantage in changing their strategy (while everybody else keeps theirs).

In other words, no player is motivated to make unilateral changes. The current strategies are mutual best responses. Eternal bliss ensues.

This is called a (pure/mixed) Nash equilibrium.

So now there are two questions:

Existence When does a pure/mixed Nash equilibrium exist?

Computation How do we compute one (if it exists)?

Let's focus on 2-person games. In a Nash equilibrium, there are no alternative strategies p' or q' such that

$$V_R(p', q) > V_R(p, q) \quad \text{or} \quad V_C(p, q') > V_C(p, q)$$

Here are some examples:

Goalie all probabilities $1/2$

Chicken $(1, 0)(0, 1)$; $(0, 1)(1, 0)$; all probabilities $1/2$

Prisoner rat/rat

R-P-S all probabilities $1/3$

Suppose we have a 2-player zero-sum game R. We can write

$$V_R(p, q) = p^T R\, q$$

which makes it tempting to bring linear programming to bear on the problem of finding a Nash equilibrium. Think of the row player as picking an optimal strategy if forced to announce in advance.

$$\begin{aligned} \max\ & z \\ p^T R - z\,\mathbf{1} &\geq 0 \\ p \circ \mathbf{1} &= 1 \\ p &\geq 0 \end{aligned}$$

Here $\mathbf{1}$ is the all-ones vector. The optimal value z_0 of this LP is none other than $\max_p \min_q p^T R\, q$.

Using $C = -R$ we can write the dual in the form

$$\begin{aligned} \max\ & z' \\ Cq - z'\mathbf{1} &\geq 0 \\ q \circ \mathbf{1} &= 1 \\ q &\geq 0 \end{aligned}$$

Note that the dual describes the strategy chosen by the column player if forced to announce in advance.

The solution z_0' of the dual is $\max_q \min_p p^T C\, q = -\min_q \max_p p^T R\, q$.

Using the strong duality theorem for LP, one can now show

Lemma

There exists a mixed Nash equilibrium for each 2-person zero-sum game, and it can be computed by linear programming.

Consider again the zero-sum game given by

$$R = \begin{pmatrix} -2 & 3 & 0 \\ 3 & -1 & 2 \end{pmatrix}$$

The primal and dual problems have solutions

$$\begin{aligned} (p_1, p_2, z) &= (4/9, 5/9, 7/9) \\ (q_1, q_2, q_3, z) &= (4/9, 5/9, 0, -7/9) \end{aligned}$$

and we have

$$\begin{aligned} V(p_1, 4/9, 5/9) &= 7/9 \\ V(4/9, q_1, q_2) &= 10/9 - q_1/3 - q_2/3 = (10 - 3(q_1 + q_2))/9 \end{aligned}$$

We duly have a Nash equilibrium.

The last result can be pushed a whole lot further.

Theorem (Nash 1950)

Each finite game has at least one mixed Nash equilibrium.

Finite here means that there are finitely many players, and, as usual, each player has finitely many actions to choose from.

But note that the corresponding claim for pure strategies is false.

Suppose we have a non-empty, compact, convex set $S \subseteq \mathbb{R}^n$, and a function $f : S \to \mathfrak{P}(S)$ such that $f(x)$ is likewise non-empty, compact and convex.

Let $\mathfrak{N}(x)$ denote the collection of open neighborhoods of x. Then f is upper hemicontinuous if

$$\forall\, x \in S, V \in \mathfrak{N}(f(x))\, \exists\, U \in \mathfrak{N}(x)\, \Big(\bigcup f(U) \subseteq V\Big)$$

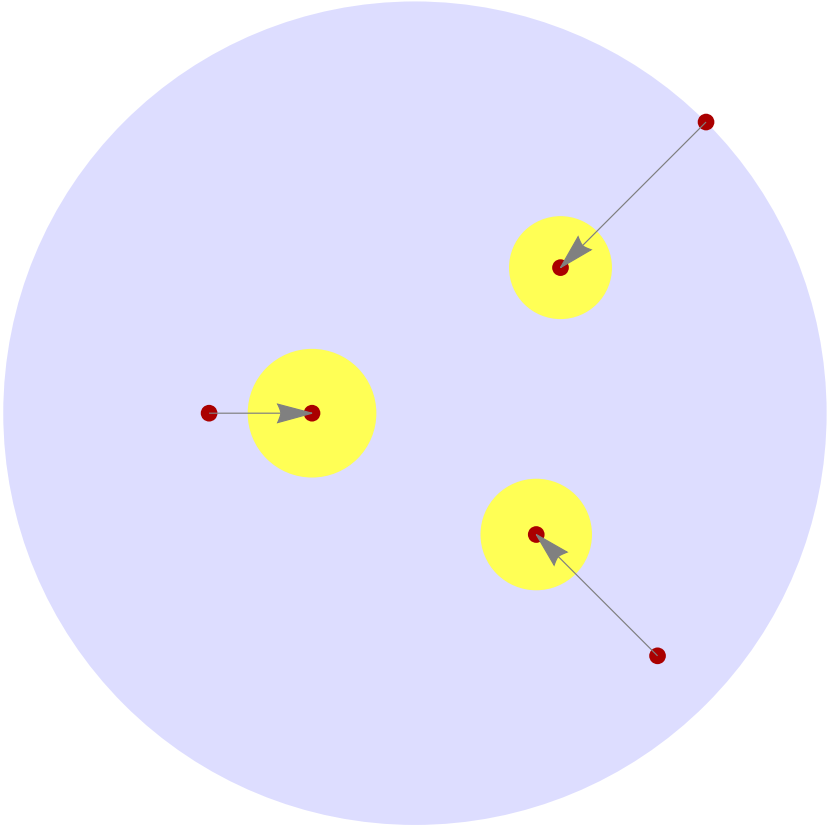
Buring Question: Why on earth should we care?

Because plain continuous functions are not quite good enough in our situation. Recall from the examples that sometimes the optimal response by a player is unconstrained. Any choice of a parameter (in a reasonable range) is fine. So we are dealing with a relation, not a function.

Here is a cheap example. Let $B(x, r) \subseteq \mathbb{R}^n$ be the closed ball of radius r around x. Then the following map is upper hemicontinuous:

$$S = B(0, 1)$$

$$f(x) = B(x/2, \cos(\|x\|)/4)$$



Theorem (Kakutani 1941)

Let f by upper hemicontinuous. Then f has a fixed point $a \in S$: $a \in f(a)$.

Since our action sets $A_i = (a_{ij})$ are finite, a probability distribution is simply a vector p of non-negative reals p_j such that $\sum p_j = 1$. A profile consists of a vector $p^{(i)}$ of such distributions. Write $\mathcal{D} = \times D_i$ for the space of all profiles.

Note that even though a game is finite, $\mathcal{D}$ is a bounded polytope in Euclidean space.

It is not hard to see that $\mathcal{D}$ is closed. Since strategies can be combined as in $\lambda p + (1 - \lambda)p'$, $\mathcal{D}$ is also convex.

In other words, we have a nice simplex, a convex, compact polytope in $(\mathbb{R}^n)^n$.

This is in strong contrast to the situation where we only consider pure strategies. We have to use analysis rather than combinatorics.

For a vector v, write $v{\uparrow}i{:}x$ for the result of replacing v_i by x. Here x may be a point or a set. For example, $(5,1,1,2){\uparrow}3{:}\{a,b\} = \{(5,1,a,2),(5,1,b,2)\}$.

Define a best-response function $\beta : \mathcal{D} \to \mathcal{D}$ by $\beta = (\beta_1, \ldots, \beta_n)$ where

$$\beta_i : \bigtimes_{j \neq i} D_j \longrightarrow \mathfrak{P}(D_i)$$

$$\beta_i(d) = d{\uparrow}i{:}\{\, p \in D_i \mid \text{preferable over } d_i \,\}$$

We will not axiomatize the notion of "preferable" here: intuitively it just means that the payoff would be at least as good or better than the current strategy, given that none of the opponents change theirs.

We need some $d^\star$ such that $d^\star \in \beta(d^\star)$.

One can check that Kakutani's theorem applies under these circumstances.

□

When von Neumann heard about Nash's result, he supposedly commented thus:

That's trivial, you know. That's just a fixed-point theorem.

Exercise

Von Neumann's comment nonwithstanding, check all the details.

Consider the 2×3 game (slightly modified from above to keep things simple)

$$R = \begin{pmatrix} -2 & 3 & 2 \\ 3 & -1 & 2 \end{pmatrix}$$

With the usual parametrization we have

$$D_1 = [0,1] \qquad D_2 = \{\, (q_1, q_2) \in [0,1]^2 \mid q_1 + q_2 \leq 1 \,\}$$

$$\beta_1(p_1, q_1, q_2) = \begin{cases} [0, p_1] & \text{if } q_2 < 5/4 q_1, \\ D_1 & \text{if } q_2 = 5/4 q_1, \\ [p_1, 1] & \text{if } q_2 > 5/4 q_1. \end{cases}$$

$$\beta_2(p_1, q_1, q_2) = \begin{cases} [q_1, 1] \times [0, q_2] \cap D_2 & \text{if } p_1 < 4/9, \\ D_2 & \text{if } p_1 = 4/9, \\ [0, q_1] \times [q_2, 1] \cap D_2 & \text{if } p_1 > 4/9. \end{cases}$$

Nash equilibria: $(1, 0, 1)$, $(0, 1, 0)$

One can turn computation of a Nash equilibrium into a variety of decision problems. There is some heuristic evidence that the most natural translations are not $\mathbb{NP}$-complete (though some are). For example, the decision question whether there exists a *second* Nash equilibrium is $\mathbb{NP}$-complete.

If one tries to deal with the actual function problem, not some artificial decision problem, things get messy (as usual). It is known that one can express the computation of a Nash equilibrium in terms of a mixed real/integer linear program.

The problem is known to be $\mathbf{PPDG}$-complete where $\mathbf{PPDG}$ is slightly messy complexity class: in essence, functions that can be proved to be total by a parity argument on directed graphs There is some evidence that there are hard problems in $\mathbf{PPDG}$.