Computational Game Theory

Lecture 1: Introduction & Motivation
Part I

Orientation
What is Game Theory?

- The mathematical theory of interaction between self-interested agents (“players”).
- Self-interest: Players assumed to act in their own interests, in pursuit of their preferences.
- Focus on decision-making where each player’s decision can influence the outcomes (and hence well-being) of other players.
- Each player must consider how each other player will act in order to make its optimal choice: hence strategic considerations.
- If a system has a single designer/owner, then game theoretic analysis is probably inappropriate.
- If all players have same preferences, then the problem is coordination: ensuring that all players “pull in the same direction”.
A “game” in the sense of game theory is an abstract model of a particular scenario in which self-interested players interact.

Abstract in the sense that we only include detail relevant to the decisions that players make:

- leads to claims that game theoretic models are “toy”
- aim is to isolate issues that are central to decision making.

Game theory origins: study of parlor games (e.g., chess)

- such games are useful for highlighting key concepts
- but the term “game” conveys something trivial :-(

An Example Game
The Brexit Game

- The UK must choose to negotiate hard or soft for brexit.
- But the best choice depends on whether the EU choose to negotiate hard or soft. . .
- The worst outcome for both parties is that both parties choose to negotiate hard.
- Otherwise, best to choose the opposite of what your counterpart does:

  a game of chicken
An Example Game
Issues or Insults?

• Trump and Clinton meet in a presidential debate
• They must each choose between **debating issues** or **making insults**
• What should Clinton do...?
• How well she is perceived to do will depend in part on the choice Trump makes
• What are the possible outcomes here? How do the candidates rank them?
Key concern in game theory is to understand what the outcomes of a game will be, under the assumption that players act rationally (in their best interests).

But it is often not clear what the best thing to do is.

Solution concepts attempt to characterise rational outcomes of games

For every game, a solution concept identifies a subset of the outcomes of the game – those that would occur if players acted according to the corresponding model of rational choice.
Evaluating Solution Concepts

**Existence:** Does a solution concept **guarantee that the game has a rational outcome?**

**Uniqueness:** Does a solution concept guarantee that the rational outcome is **unique?**

**Tractability:** Is it computationally easy to verify that an outcome is rational? Is it easy to compute a rational outcome?

**Comprehensibility:** Is it easy for people to understand why an outcome is/is not rational?

**Invariance:** Do small changes in game setup lead to small changed in the outcome?

⇒ All solution concepts fail on one or more of these criteria. ⇐
• Under a **descriptive** interpretation, we take game theory as predicting how people will act in strategic settings, and explaining why they acted the way they did.

• A major area of research to determine the extent to which game theoretic solution concepts predict human choices (somewhat controversial)
In real life settings, social norms (and in particular, norms of cooperation) often play a part in how people make decisions. However, if the incentives are sufficiently large, then these can override such norms.

For incentives (such as payments) to influence behavior, they must be adequate.

For players to make rational choices, the game they are playing must be sufficiently simple.

Players will adapt their behavior over time towards more rational outcomes, if they are given sufficient opportunity for trial-and-error learning.

---

Interpreting Game Theory

Normative Interpretations

- Under a **normative** interpretation, we take game theory as **giving us advice**: telling us what we **ought to do** in a real-world situation.

- Whether the advice is useful depends on whether the game model used was appropriate, and whether the **assumptions** on which the model depends are satisfied. (Typical assumptions: everybody knows everybody’s preferences, actions, and their consequences, everybody acts rationally, . . . )

- Game theory can be used to **design** interaction scenarios: (**mechanism design**).
  
  **EXAMPLE 1.** 3G spectrum auctions in 2000 yielded $35 billion for UK government.
  
  **EXAMPLE 2.** “security games” paradigm (Milind Tambe)
Non-Cooperative *versus* Cooperative Games

- Game theory is usually sub-divided into *non-cooperative* and *cooperative* versions.
- Non-cooperative game theory is bigger and better-known: it concerns settings where *players must act alone*. Solution concepts in non-cooperative game theory relate to *individual* action.
- Cooperative game theory is concerned with settings where players can make *binding agreements* to work together, allowing for teamwork, cooperation, joint action.
Part II

Why is game theory relevant to computer science?
Mechanisms and Protocols

- Distributed systems research has focussed on **protocols** (TCP/IP, leader election, bluetooth, ...)
  Typical issues: deadlock, mutual exclusion...

- In computational game theory, we study **mechanisms**:
  
  \[
  \text{mechanism} = \text{protocol} + \text{preferences} + \text{rational choice}
  \]

- Mechanisms take into account the fact that protocol participants are not benevolent entities – they are self-interested.

  **strategic considerations come to the fore.**

- Treating mechanisms as if they were simply protocols misses a big part of the story.

  **Example: sniping on eBay**

- In **multi-agent systems**, mechanism participants are software agents.
Two perspectives

- **Algorithmic mechanism design**: take economic factors (in particular: self interested behaviour) into consideration when designing computational systems.
- **Electronic market design**: Use computer science techniques in the design of economic systems.
• Let $\Gamma$ be a class of games. (It doesn’t matter exactly what the games $G \in \Gamma$ are.)

• Associated with $\Gamma$ is a set $\Omega$ of outcomes.

• Where $G \in \Gamma$ is a specific game, let $\Omega_G$ denote the possible outcomes of $G$.

• A solution concept $f$ for a class of games $\Gamma$ with outcomes $\Omega$ is a function:

$$f : \Gamma \rightarrow 2^\Omega$$

such that $f(G) \subseteq \Omega_G$. 
Non-emptiness: Given $G \in \Gamma$, is it the case that $f(G) \neq \emptyset$? Does the game have any rational outcome according to the solution concept $f$?

Membership: Given $G \in \Gamma$ and $\omega \in \Omega_G$, is it the case that $\omega \in f(G)$? Is a given outcome rational according to $f$?

Computation: Given $G \in \Gamma$, output some $\omega$ such that $\omega \in f(G)$. Here, we actually want to compute a rational outcome of the game.
Part III

Further reading
Further Reading

General Game Theory References

  (IMHO, the best contemporary reference for game theory: rigorous but very readable.)

  (Until Maschler *et al* came along, this was my favourite. Available free (legally!) from: [http://tinyurl.com/gtbook](http://tinyurl.com/gtbook))

  (A useful companion for bedtime reading. Full of razor sharp opinions and insight from a master of the art.)
Further Reading

Game Theory and Computer Science


- **Computational Aspects of Cooperative Game Theory** by Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. Morgan-Claypool, 2011. (As the name suggests, studies cooperative game theory from the point of view of computer science.)

Part IV

History of game theory
History of Game Theory
Phase One: 1928–54

- Originated in current form in early part of 20th century.
- Original focus: parlor games such as poker, chess (e.g., Zermelo on game of chess)
- First milestone: the **minimax theorem** proved in 1928 by Hungarian polymath John von Neumann (1903–57), leading to...
- Initial scope of game theoretic techniques very limited (typically “2 person zero sum games”)

History of Game Theory
Phase Two: 1954–1980

• Scope of game theory **hugely** extended in 1950s with work of John Forbes Nash, Jr (1928–2014), and the concept of **Nash equilibrium** (NE)
  (NE remains to this day the chief analytical concept in game theory)
• A flurry of activity in 1950s, with other key results by Selten, Aumann, Shapley, Harsanyi and others
• But activity began to peter out as limitations to applicability of NE make themselves felt.
In late 1970s/early 1980s, focus shifted to how societies converge on strategies.

John Maynard Smith (1920–2004) and George Price (1922–75) laid foundations of evolutionary game theory, which refines NE and shows how societies can converge on equilibria through purely evolutionary processes.

Explain many biological questions, but also turn out to have direct relevance to economics.

Robert Axelrod (1943–) hosts Prisoner’s Dilemma competition, to much acclaim.
• **Auction design** raises much interest in game theoretic mechanism design

• Links between computer science & game theory: Christos Papadimitriou et al

• Four Nobel prizes for game theory:
  - 1994: John Harsanyi, John Forbes Nash, Reinhard Selten
  - 2005: Robert Aumann, Thomas Schelling
  - 2007: Leonid Hurwicz, Eric Maskin, Roger Myerson
  - 2012: Al Roth, Lloyd Shapley
Lecture 2: Preferences, Utilities, and Decisions
• Preferences are what give games their strategic character.
• In multi-agent systems, we delegate our preferences to a software agent, who then acts on our behalf in pursuit of them.
• A problem: people find it hard to formulate our preferences, and may not act rationally wrt them.
• Preference elicitation is the process of extracting preferences from principals
• Utilities are a numeric representation of preferences: allows us to reduce decision-making to optimisation
• $\Omega = \{\omega_1, \ldots, \omega_k\}$ is the set of “outcomes”. In this lecture we assume $\Omega$ is finite.

• These are the consequences of player’s choices.

• $\Omega$ may be:
  - all the possible outcomes of a game of chess
  - the possible outcomes of negotiations between nations
  - the possible outcomes of an eBay auction
  - ... and so on.
We use slightly different interpretations of preferences, depending on whether the decision-making setting is one of **certainty** or **uncertainty**.

In **decision-making under certainty**, we know exactly what the consequences of our choices will be.

In **decision-making under uncertainty**, we don’t know exactly what the consequences of our choices will be: for every possible choice, there are multiple possible consequences, each with an attached probability.
Part V

Decision Making Under Certainty
A preference relation is a binary relation $\succeq \subseteq \Omega \times \Omega$, which is required to satisfy:

- **Reflexivity:**
  \[ \omega \succeq \omega \text{ for all } \omega \in \Omega \]
  
- **Totality:**
  for all $\omega, \omega' \in \Omega$ we have either $\omega \succeq \omega'$ or $\omega' \succeq \omega$

- **Transitivity:**
  for all $\omega, \omega', \omega'' \in \Omega$, if $\omega \succeq \omega'$ and $\omega' \succeq \omega''$ then $\omega \succeq \omega''$
Indifference

If both

\[ \omega \succeq \omega' \quad \text{and} \quad \omega' \succeq \omega \]

then we say

you are indifferent between \( \omega \) and \( \omega' \)

and we write

\[ \omega \sim \omega' \]
Strict Preference

If

\[ \omega \succeq \omega' \quad \text{but not} \quad \omega' \succeq \omega \]

then we say

\text{you strictly prefer } \omega \text{ over } \omega' \]

and we write

\[ \omega \succ \omega' \]
Interpreting Preferences (IMPORTANT)

Revealed Preferences

- $\omega \succ \omega'$ means that:
  - if you have a choice between $\omega$ and $\omega'$, you will choose $\omega$
  - if you have a choice between two options, one of which will result in $\omega$, the other of which will result in $\omega'$, you will choose the option resulting in $\omega$

- Your preference relation must capture everything about the game that you care about.

- If you care about other people, then this is reflected in your preferences.
  (Many arguments in game theory would be avoided if everybody understood this!)
Utility functions

• It is useful to represent preference relations by attaching numbers to outcomes: higher numbers indicate more preferred.

• The numbers are called **utility values** or **utilities**.

• A **utility function** \( u : \Omega \rightarrow \mathbb{R} \) is said to represent a preference relation \( \succeq \) iff we have:

\[
  u(\omega) \geq u(\omega') \quad \text{iff} \quad \omega \succeq \omega'
\]

**Theorem**

*For every preference relation \( \succeq \subseteq \Omega \times \Omega \) there is a utility function \( u : \Omega \rightarrow \mathbb{R} \) that represents \( \succeq \).*

Proof: exercise.
What is Utility?

- We use numeric utilities because it allows us to use numeric techniques for solving games.
- Utilities are selected to represent preferences $\succeq$.
- It is a fallacy to claim you choose $\omega$ over $\omega'$ because $u(\omega) > u(\omega')$.
  - You make this choice because $\omega \succ \omega'$.
  - The $u(\cdots)$ values were chosen to reflect this.
- But, if we picked the numbers right, then you behave as though you were maximising utility.
- Utility values in decision-making under certainty don’t represent intensity: they are ordinal values, which indicate relative rankings.
- Interpersonal comparisons of utility are difficult. “One util” for me is not the same as “one util” for you.
Utility is not money!

• Much misunderstanding caused by people interpreting utility as money, leading to the implication that game theory is about “greed”…

• Utility as money is often a useful analogy.

• Typical relationship between utility & money:

```
utility

money
```
• Let $\Sigma$ be the set of **strategies** (choices, actions, alternatives...) available to our decision maker.

• An **outcome function** (a.k.a. consequences function) is

$$g : \Sigma \rightarrow \Omega$$

• The **feasible outcomes** are those that could be obtained through the performance of an appropriate strategy. Formally, the feasible outcomes are the range of $g$, i.e., $\text{ran } g$.

• If $\text{ran } g \subset \Omega$ then some outcomes cannot be obtained.
Decision Making Under Certainty

• A problem of **decision making under certainty** is given by a quad

\[ \langle \Omega, \ u : \Omega \rightarrow \mathbb{R}, \ \Sigma, \ g : \Sigma \rightarrow \Omega \rangle \]

• The task of our decision maker is to select a strategy \( \sigma^* \) that leads to an outcome which maximises utility:

\[ \sigma^* \in \arg \max_{\sigma \in \Sigma} u(g(\sigma)) \]

• Notice that using numeric utilities allows us to express decision problems as optimisation problems.
Part VI

Preference Relations with Structure
Preference Relations with Structure

• Preference relations often have some structure.
• Here we look at 3 important classes of preference relation:
  1. single-peaked preferences
  2. dichotomous preferences
  3. lexicographic preferences
A preference relation is **single-peaked** with respect to a fixed ordering $\omega_1 > \omega_2 > \ldots > \omega_k$ of the alternatives (the axis) iff

1. there is a most preferred candidate $\omega^*$ and;
2. candidates closer to $\omega^*$ are preferred over those that are further away:
   - if $\omega^* > \omega_1 > \omega_2$ then $\omega_1 \succ \omega_2$
   - if $\omega_1 > \omega_2 > \omega^*$ then $\omega_2 \succ \omega_1$

**Example**

Suppose we can order electoral candidates according to on the left-right spectrum. It is natural for us to identify a single point on this spectrum representing our personal political preferences, and we prefer candidates closer to this ideal.

Single-peaked preferences are important in social choice theory, where they play a key role in the **median voter theorem**.
Dichotomous Preferences

• A preference relation is **dichotomous** if it classifies all outcomes as either **win** or **lose**.
• Formally, there exist $W \subseteq \Omega$ and $L \subseteq \Omega$ such that:
  - $W \cup L = \Omega$
  - $W \cap L = \emptyset$
  - $\forall \omega_1, \omega_2 \in W, \omega_1 \sim \omega_2$
  - $\forall \omega_1, \omega_2 \in L, \omega_1 \sim \omega_2$
  - $\forall \omega_1 \in W, \forall \omega_2 \in L, \omega_1 \succ \omega_2$
• The set $W$ can be interpreted as a **goal**.
• Dichotomous preferences naturally specified with logical formulae.
Preferences are lexicographic if outcomes can be characterised by an ordered set of attributes, where each attribute has its own ordering.

**Example**

Let $\Omega = \text{all words in the Oxford English dictionary}$. Suppose I prefer words occurring earlier in the dictionary. Hence I prefer all words starting with “a” over those starting with “b”, and all words starting “aa” over those starting “ab”, and so on.

$\Rightarrow$ why such preferences are called **lexicographic**.
Lexicographic Preferences

Example

With respect to cars, the attributes I use to order are:

\[ \text{colour} \succ \text{engine} \succ \text{nationality} \]

The ordering for each of these attributes is:

1. **colour**: red \( \succ \) blue \( \succ \) green
2. **engine type**: electric \( \succ \) petrol \( \succ \) diesel
3. **nationality**: German \( \succ \) French \( \succ \) UK

So, I rank all red cars above all other colours. I rank all red electric cars above all red petrol cars. I rank all red electric German cars above all red electric UK cars.
Part VII

Decision Making Under Uncertainty
Motivation

• In many settings, we don’t know exactly what outcome will result by performing a strategy.
• For every strategy $\sigma \in \Sigma$, there will typically be a range of possible outcomes, with differing probabilities of occurring.
• Such settings require more complex machinery for preferences and utilities.
• In particular, preference relations $\succeq \subseteq \Omega \times \Omega$ are not enough: we need preferences over lotteries.
• A **probability distribution** over a non-empty set $S$ is a function

$$f : S \rightarrow [0, 1]$$

which must satisfy the constraint that

$$\sum_{s \in S} f(s) = 1$$

• So $f(s)$ is the probability of $s$ given distribution $f$

• Let $\Delta(S)$ denote the set of all probability distributions over $S$

• Where $s \in S$ and $f \in \Delta(S)$, we sometimes write $P(s, f)$ to denote $f(s)$
Lotteries

- A **lottery** over $S$ is a probability distribution over $S$, i.e., an element $\ell \in \Delta(S)$.
- We denote individual lotteries by $\ell, \ell', \ell_1$, etc, and denote the set of lotteries over set $S$ by $\text{lott}(S)$.

**Example**

Suppose $\Omega = \{\text{whisky}, \text{gin}, \text{brandy}\}$. Then

$$\ell_1 = \frac{1}{10} \text{ whisky} + \frac{2}{10} \text{ gin} + \frac{7}{10} \text{ brandy}$$

means whisky with probability 0.1, gin with probability 0.2, brandy with probability 0.7.

- We work with preference relations over lotteries:

$$\succeq \subseteq \text{lott}(\Omega) \times \text{lott}(\Omega)$$
Example

Suppose

\[ \ell_1 = \frac{1}{10} \text{gin} + \frac{9}{10} \text{brandy} \quad \text{and} \quad \ell_2 = \text{brandy} \]

Here, \( \ell_2 \) is a **degenerate** lottery (brandy with certainty!)

If you prefer brandy over gin, what is an appropriate preference relation over these lotteries?

Suppose you prefer gin over brandy?
A **compound lottery** is a lottery over lotteries – an element of the set $lott(lott(\Omega))$.

**Example**

Recall

$$l_1 = \frac{1}{10} \text{gin} + \frac{9}{10} \text{brandy} \quad \text{and} \quad l_2 = 1 \text{brandy}$$

Now suppose:

$$l_3 = \frac{9}{10} l_1 + \frac{1}{10} l_2 \quad \text{and} \quad l_4 = \frac{1}{100} l_1 + \frac{99}{100} l_2$$

We obtain a simple lottery by multiplying out probabilities. In what follows, when we talk about lotteries, assume we are talking about compound lotteries.
How do we Measure the Utility of a Lottery?

Expected Utility

• How can we use a utility function $u : \Omega \rightarrow \mathbb{R}$ to measure the utility of a lottery?
• We use expected utility – intuitively, the “average” utility that we could expect from the lottery.
• More precisely, the expected value of the function $u$.
• Given a utility function $u : \Omega \rightarrow \mathbb{R}$, the expected utility $EU(\ell)$ of lottery $\ell$ is defined by:

$$EU(\ell) = \sum_{\omega \in \Omega} u(\omega) P(\omega, \ell)$$
We now have three preference indicators for decision-making under uncertainty:
- preference relations \(\preceq \subseteq \text{lott}(\Omega) \times \text{lott}(\Omega)\) over (compound) lotteries
- utility functions over outcomes \(u : \Omega \rightarrow \mathbb{R}\)
- expected utility \(EU(\ell) = \sum_{\omega \in \Omega} u(\omega) P(\omega, \ell)\)

John von Neumann and Oskar Morgenstern constructed the mathematical theory that links these coherently, via the von Neumann and Morgenstern axioms.

A preference relation \(\succeq\) satisfying these properties is sufficient to induce a utility function \(u : \text{lott}(\Omega)\) over lotteries that corresponds to (expected) utility.

**Important:** Utility in decision making under uncertainty must capture the intensity of preferences.
• Suppose $\Omega = \{\mathcal{W}, \mathcal{L}\}$ with $\mathcal{W} = \text{“win”}$ and $\mathcal{L} = \text{“lose”}$, with $\mathcal{W} \succ \mathcal{L}$.

• Then we only need one additional axiom, **continuity**, which says that you prefer lotteries that increase the probability of a good outcome:

For all lotteries $\ell_1, \ell_2$, we have

$$\ell_1 \succeq \ell_2 \iff P(\mathcal{W}, \ell_1) \geq P(\mathcal{W}, \ell_2)$$
A Simple Case

**Theorem**

If a preference relation $\succeq \subseteq \text{lott}(\{\mathcal{W}, \mathcal{L}\}) \times \text{lott}(\{\mathcal{W}, \mathcal{L}\})$ over win-lose lotteries satisfies totality, reflexivity, transitivity, and continuity, then there exists a utility function

$$u : \{\mathcal{W}, \mathcal{L}\} \rightarrow \mathbb{R}$$

such that

$$\ell_1 \succeq \ell_2 \text{ iff } EU(\ell_1) \geq EU(\ell_2)$$

where

$$EU(\ell) = \sum_{\omega \in \{\mathcal{W}, \mathcal{L}\}} u(\omega) P(\omega, \ell)$$
In addition to **totality**, **reflexivity**, and **transitivity**, Von Neumann and Morgenstern introduced:

1. the **Equivalence** axiom
2. the **Monotonicity** axiom
3. the **Archimedean** axiom
4. the **Independence/Substitution** axiom
The **structure** of a lottery is irrelevant — all that matters is the probability distribution over outcomes that the lottery defines.

*Every compound lottery is equivalent to the simple lottery with the same probability distribution over outcomes.*
If you prefer $\ell_1$ over $\ell_2$ then you will prefer to maximise the probability of getting $\ell_1$ over $\ell_2$.

If

\[ \ell_1 \succeq \ell_2 \]

and $p \geq q$

\[ p\ell_1 + (1 - p)\ell_2 \succeq q\ell_1 + (1 - q)\ell_2 \]
Essentially, this says we can quantify our preferences over lotteries.

If

\[ \ell_1 \succeq \ell_2 \succeq \ell_3 \]

then there is some \( p \in [0, 1] \) such that

\[ \ell_2 \sim p\ell_1 + (1 - p)\ell_3 \]
If we have preferences \( \ell_1 \succ \ell_2 \), then mixing these with an independent alternative preserves our preferences.

\[
\ell_1 \succeq \ell_2 \text{ iff for all lotteries } \ell_3 \text{ and probabilities } p \in (0, 1) \text{ we have }
\]

\[
p\ell_1 + (1 - p)\ell_3 \succeq p\ell_2 + (1 - p)\ell_3
\]

This axiom implies we can substitute lotteries that we are indifferent between.
Von Neumann and Morgenstern’s Theorem

**Theorem**

If a preference relation \(\succeq \subseteq \text{lott}(\Omega) \times \text{lott}(\Omega)\) satisfies the von Neumann and Morgenstern axioms, then there exists a function \(u : \Omega \rightarrow \mathbb{R}\) such that:

\[
\ell_1 \succeq \ell_2 \iff EU(\ell_1) \geq EU(\ell_2)
\]

where

\[
EU(\ell) = \sum_{\omega \in \Omega} u(\omega)P(\omega, \ell)
\]
Von Neumann and Morgenstern’s Theorem
Proof Overview

1. Identify **best and worst outcomes** – call them \( \mathcal{W} \) and \( \mathcal{L} \)
2. Use \( \mathcal{W} \) and \( \mathcal{L} \) to establish a **scale** with \( \mathcal{L} \) valued 0 and \( \mathcal{W} \) valued 1.
3. Use the Archimedean axiom to place outcomes \( \omega \) on this scale.
Von Neumann and Morgenstern’s Theorem
Proof Step 1: Dealing with the trivial case

- If $\omega_1 \sim \omega_2 \sim \cdots \sim \omega_k$

then we are indifferent between all outcomes.
- In this case define $u(\omega) = 1$ for all $\omega \in \Omega$
- \ldots and we are done.
Von Neumann and Morgenstern’s Theorem
Proof Step 2: Establish a scale from worst to best

- Otherwise, order the alternatives worst up to best. (Assume for simplicity the ordering is strict: no indifference.)
- Such an ordering exists by the **totality** axiom.
- Pick the lowest ranked outcome; call it \( \mathcal{L} \) (“lose”). Let \( u(\mathcal{L}) = 0 \).
- Pick the highest ranked outcome; call it \( \mathcal{W} \) (“win”). Let \( u(\mathcal{W}) = 1 \).
- \( \mathcal{L} \) and \( \mathcal{W} \) define our scale, within which we fit other outcomes.
Von Neumann and Morgenstern’s Theorem
Proof Step 3: Ordering the alternatives

• Where \( p \in [0, 1] \), we let \( \ell^* (p) \) denote the following lottery:

\[
\ell^* (p) = pW + (1 - p)L
\]

• By the **Archimedean axiom**, for each outcome \( \omega \) there is a \( p_\omega \in [0, 1] \) such that \( \omega \sim \ell^* (p_\omega) \), i.e.,

\[
\omega \sim p_\omega W + (1 - p_\omega)L
\]

• Define \( u(\omega) = p_\omega \)

• The probability \( p_\omega \) places \( \omega \) on the scale between \( L \) and \( W \)
Von Neumann and Morgenstern’s Theorem
Proof Step 4: Correctness of the construction

- Take two lotteries \( \ell_1 \succeq \ell_2 \), where
  \[
  \ell_1 = p_1 \omega_1 + \cdots + p_k \omega_k \\
  \ell_2 = q_1 \omega_1 + \cdots + q_k \omega_k
  \]
- Replace each occurrence of \( \omega_i \) in \( \ell_1, \ell_2 \) by \( \ell^*(p_{\omega_i}) \).
  So, for example, we have:
  \[
  \ell_1 = p_1 \ell^*(p_{\omega_1}) + \cdots + p_k \ell^*(p_{\omega_k})
  \]
- We now have \( \ell_1 \) and \( \ell_2 \) expressed in terms of \( \mathcal{W} \) and \( \mathcal{L} \)
- Collect \( \mathcal{W} \) and \( \mathcal{L} \) terms and simplify. Looking at \( \ell_1 \):
  \[
  \ell_1 = (\sum_{i=1}^k p_i p_{\omega_i}) \mathcal{W} + (1 - (\sum_{i=1}^k p_i p_{\omega_i})) \mathcal{L}
  \]
  \[
  = (\sum_{i=1}^k p_i u(\omega_i)) \mathcal{W} + (1 - (\sum_{i=1}^k p_i u(\omega_i))) \mathcal{L}
  \]
  Similarly:
  \[
  \ell_2 = (\sum_{i=1}^k q_i u(\omega_i)) \mathcal{W} + (1 - (\sum_{i=1}^k q_i u(\omega_i))) \mathcal{L}
  \]
Since \( \mathcal{W} \succ \mathcal{L} \), by monotonicity it must be that

\[
\sum_{i=1}^{k} p_i u(\omega_i) \geq \sum_{i=1}^{k} q_i u(\omega_i)
\]

Since \( u(\mathcal{W}) = 1 \) and \( u(\mathcal{L}) = 0 \), then

\[
EU(\ell_1) = \sum_{i=1}^{k} p_i u(\omega_i)
\]
\[
EU(\ell_2) = \sum_{i=1}^{k} q_i u(\omega_i)
\]

Therefore

\[
EU(\ell_1) \geq EU(\ell_2)
\]
A problem of **decision making under uncertainty** is given by a quad

\[ \langle \Omega, \ u : \Omega \rightarrow \mathbb{R}, \ \Sigma, \ g : \Sigma \rightarrow \text{lott}(\Omega) \rangle \]

The task of our decision maker is to select a strategy \( \sigma^* \) that maximises expected utility:

\[
\sigma^* \in \arg \max_{\sigma \in \Sigma} EU(g(\sigma)) \\
\in \arg \max_{\sigma \in \Sigma} \sum_{\omega \in \Omega} u(\omega) P(\omega, g(\sigma))
\]
Part VIII

Paradoxes of Expected Utility Theory
The MEU paradigm is so entrenched that it is commonplace to define **rational agents** as those that act so as to maximise expected utility.

However, it is not hard to find examples in which either maximising expected utility seems to be the wrong thing to do, or where the advice offered by the theory is counter to strong intuitions.
Paradoxes of Expected Utility Theory

Paradox 1: A Simple Example

Example

Suppose

\[ \ell_1 = \$50 \quad \ell_2 = \frac{1}{2} \$101 + \frac{1}{2} \$0 \]

Many people prefer \( \ell_1 \), and the certain \( \$50 \).

An example of **risk aversity**, and an illustration that expected monetary reward does not equate to utility.
Paradoxes of Expected Utility Theory

Paradox 2: The Allais Paradox

Example

Consider following lotteries:

\[ \ell_A = \$2m \]

\[ \ell_B = \frac{89}{100} \$2m + \frac{1}{10} \$10m + \frac{1}{100} \$0 \]

Most people prefer \( \ell_A \) over \( \ell_B \).

Most people prefer \( \ell_D \) over \( \ell_C \).

Example

Consider following lotteries:

\[ \ell_A = 2m \]
\[ \ell_B = \frac{89}{100} 2m + \frac{1}{10} 10m + \frac{1}{100} 0 \]

Most people prefer \( \ell_A \) over \( \ell_B \).

Paradoxes of Expected Utility Theory
Paradox 2: The Allais Paradox

Example

Consider the following lotteries:

\[ \ell_A = 2m \]
\[ \ell_B = \frac{89}{100}2m + \frac{1}{10}10m + \frac{1}{100}0 \]

Most people prefer \( \ell_A \) over \( \ell_B \).

\[ \ell_C = \frac{11}{100}2m + \frac{89}{100}0 \]
\[ \ell_D = \frac{1}{10}10m + \frac{9}{10}0 \]

Paradoxes of Expected Utility Theory
Paradox 2: The Allais Paradox

Example

Consider following lotteries:

\[ \ell_A = \$2m \]
\[ \ell_B = \frac{89}{100}\$2m + \frac{1}{10}\$10m + \frac{1}{100}\$0 \]

Most people prefer \( \ell_A \) over \( \ell_B \).

\[ \ell_C = \frac{11}{100}\$2m + \frac{89}{100}\$0 \]
\[ \ell_D = \frac{1}{10}\$10m + \frac{9}{10}\$0 \]

Most people prefer \( \ell_D \) over \( \ell_C \).

Lemma

If you have preferences $\ell_A \succ \ell_B$ and $\ell_D \succ \ell_C$ then you do not satisfy the Von Neumann and Morgenstern axioms.

Proof: Let $u : \Omega \to \mathbb{R}$ represent $\succ$, and let $x = u(0)$, $y = u(2m)$, and $z = u(10m)$. 

$EU(\ell_A) > EU(\ell_B)$ (1)

$y > 0.1z + 0.89y + 0.01x$ (2)

$EU(\ell_D) > EU(\ell_C)$ (3)

$0.1z + 0.9x > 0.11y + 0.89x$ (4)

But add 0.89($x - y$) to each side of (2):

$0.11y + 0.89x > 0.1z + 0.9x$ (5)

$EU(\ell_C) > EU(\ell_D)$ (6)

Contradiction.
Lemma

If you have preferences $\ell_A \succ \ell_B$ and $\ell_D \succ \ell_C$ then you do not satisfy the Von Neumann and Morgenstern axioms.

Proof: Let $u : \Omega \to \mathbb{R}$ represent $\succ$, and let $x = u(0)$, $y = u(2m)$, and $z = u(10m)$.

\[
EU(\ell_A) > EU(\ell_B) \quad (1)
\]
\[
y > 0.1z + 0.89y + 0.01x \quad (2)
\]
Paradoxes of Expected Utility Theory
The Allais Paradox

Lemma

If you have preferences $\ell_A \succ \ell_B$ and $\ell_D \succ \ell_C$ then you do not satisfy the Von Neumann and Morgenstern axioms.

Proof: Let $u : \Omega \rightarrow \mathbb{R}$ represent $\succ$, and let $x = u(0)$, $y = u(2m)$, and $z = u(10m)$.

\[
EU(\ell_A) > EU(\ell_B) \tag{1}
\]

\[
y > 0.1z + 0.89y + 0.01x \tag{2}
\]

\[
EU(\ell_D) > EU(\ell_C) \tag{3}
\]

\[
0.1z + 0.9x > 0.11y + 0.89x \tag{4}
\]

But add $0.89(x - y)$ to each side of (2):

\[
y + 0.89(x - y) > 0.11y + 0.89x \tag{5}
\]

Contradiction.
Paradoxes of Expected Utility Theory
The Allais Paradox

Lemma

If you have preferences $\ell_A \succ \ell_B$ and $\ell_D \succ \ell_C$ then you do not satisfy the Von Neumann and Morgenstern axioms.

Proof: Let $u : \Omega \rightarrow \mathbb{R}$ represent $\succ$, and let $x = u(0)$, $y = u(2m)$, and $z = u(10m)$.

\[
EU(\ell_A) > EU(\ell_B) \tag{1} \\
y > 0.1z + 0.89y + 0.01x \tag{2} \\
EU(\ell_D) > EU(\ell_C) \tag{3} \\
0.1z + 0.9x > 0.11y + 0.89x \tag{4}
\]

But add $0.89(x - y)$ to each side of (2):

\[
0.11y + 0.89x > 0.1z + 0.9x \tag{5} \\
EU(\ell_C) > EU(\ell_D) \tag{6}
\]

Contradiction.
Paradoxes of Expected Utility Theory
Paradox 3: Kahneman and Tversky’s Framing Effects

Example

A deadly strain of flu has been detected in the USA, which is expected to kill 600 people. There are only 2 treatments:

(A) 200 people will be saved
(B) there is a \(\frac{1}{3}\) probability that 600 will be saved, and a \(\frac{2}{3}\) probability that nobody will be saved

\[72\% \text{ of people had } A \succ B.\]

Now consider the following.

(C) 400 people will die
(D) there is a \(\frac{1}{3}\) probability that nobody will die, and a \(\frac{2}{3}\) probability that 600 people will die

\[78\% \text{ of people had } D \succ C.\]

But A and C are identical, as are B and D.

---

Paradoxes of Expected Utility Theory

Paradox 3: Kahneman and Tversky’s Framing Effects

Example
A deadly strain of flu has been detected in the USA, which is expected to kill 600 people. There are only 2 treatments:

(A) 200 people will be saved
(B) there is a \(\frac{1}{3}\) probability that 600 will be saved, and a \(\frac{2}{3}\) probability that nobody will be saved

72% of people had \(A \succ B\).

Paradoxes of Expected Utility Theory
Paradox 3: Kahneman and Tversky’s Framing Effects

Example
A deadly strain of flu has been detected in the USA, which is expected to kill 600 people. There are only 2 treatments:

(A) 200 people will be saved
(B) there is a $\frac{1}{3}$ probability that 600 will be saved, and a $\frac{2}{3}$ probability that nobody will be saved

72% of people had $A \succ B$. Now consider the following.

(C) 400 people will die
(D) there is a $\frac{1}{3}$ probability that nobody will die, and a $\frac{2}{3}$ probability that 600 people will die

---

Example

A deadly strain of flu has been detected in the USA, which is expected to kill 600 people. There are only 2 treatments:

(A) 200 people will be saved
(B) there is a $\frac{1}{3}$ probability that 600 will be saved, and a $\frac{2}{3}$ probability that nobody will be saved

72% of people had $A \succ B$. Now consider the following.

(C) 400 people will die
(D) there is a $\frac{1}{3}$ probability that nobody will die, and a $\frac{2}{3}$ probability that 600 people will die

78% of people had $D \succ C$.

---

Paradoxes of Expected Utility Theory

Paradox 3: Kahneman and Tversky’s Framing Effects

Example

A deadly strain of flu has been detected in the USA, which is expected to kill 600 people. There are only 2 treatments:

- **(A)** 200 people will be saved
- **(B)** there is a \(\frac{1}{3}\) probability that 600 will be saved, and a \(\frac{2}{3}\) probability that nobody will be saved

72% of people had \(A \succ B\). Now consider the following.

- **(C)** 400 people will die
- **(D)** there is a \(\frac{1}{3}\) probability that nobody will die, and a \(\frac{2}{3}\) probability that 600 people will die

78% of people had \(D \succ C\).

But A and C are identical, as are B and D...

---

The issue here is that people are affected by the way a decision problem is “framed”.

In this example, we prefer to choose “saving lives”.

The study of **how people make economic decisions** is the domain of **behavioural economics**.

Fun reading on this subject:

Part IX

Compact Representations
The Need for Compact Representations

- Often, the set $\Omega$ is too large to enumerate preference relations explicitly
  $\Rightarrow$ we need a **compact** and **tractable** representation for preferences

**Example**

Suppose you are in a class with $n$ other people, and you must form a team with some subset of them. Your preferences must order $2^n$ possible teams...

- But compact representations raise **computational** problems: decision problems start to get hard!
- **Compact but tractable representations** of utilities/preferences is a major area of research.
Many domains can be represented by a finite set of variables $\Phi = \{x_1, \ldots, x_l\}$, where each variable takes value $\top$ ("true") or $\bot$ (false).

**Example**

Recall the class team example. Let $N = \{1, \ldots, n\}$ be the class members. For each class member $i \in N$ define a Boolean variable $x_i$, with

- $x_i = \top$ means "$i$ is in the team"
- $x_i = \bot$ means "$i$ is not in the team"

Any valuation $\nu : \Phi \rightarrow \{\top, \bot\}$ defines a team.
Dichotomous Boolean Preferences

Example
Continuing the team example. Suppose you have dichotomous preferences: you divide the teams into $\mathcal{W} \subseteq 2^N$ and $\mathcal{L} \subseteq 2^N$, such that $\mathcal{W} \cup \mathcal{L} = 2^N$ and $\mathcal{W} \cap \mathcal{L} = \emptyset$.

- We can specify dichotomous such preference relations via propositional formulae, $\gamma$.
- Each $i \in N$ corresponds to a Boolean variable $x_i$.
- The set of satisfying assignments for $\gamma$ are the “winning” teams.
- We can define a utility function:

\[
\begin{aligned}
\quad u(v) &= \begin{cases} 
1 & \text{if } v \models \gamma \\
0 & \text{otherwise}
\end{cases} 
\end{aligned}
\]
Dichotomous Preferences
The Fab Four

Example
Suppose

\[ \gamma = John \lor Paul \land (George \land Ringo) \land \neg (John \land Paul) \]

Which teams satisfy this goal?
Let the **naive representation** for dichotomous Boolean preferences be the representation in which we **explicitly list all winning sets** $\mathcal{W} \subseteq 2^N$.

**Theorem**

1. *The propositional formula representation for dichotomous Boolean preferences is complete:* any dichotomous Boolean preference relation can be represented by a propositional formula.

2. *The propositional formula representation can be exponentially more compact* than the naive representation.

3. *There exist dichotomous preference relations for which the smallest propositional representation is of size exponential in $|N|$.*
Weighted Formula Representations

• What about utility functions \( u : 2^N \rightarrow \mathbb{R} \)?
  The naive representation here involves listing all \( 2^{|N|} \) input/output pairs of \( u \)

• We can use weighted formula representation.

• A weighted formula, or rule is a pair \((\varphi, x)\) where \( \varphi \) is a propositional formula and \( x \in \mathbb{R} \).
  We sometimes write \( \varphi \rightarrow x \)

• We use rule bases, \( \mathcal{R} \), to define utility functions:

\[
\mathcal{R} = \{(\varphi_1, x_1), \ldots, (\varphi_k, x_k)\}
\]

• The utility function \( u \) associated with \( \mathcal{R} \) is defined:

\[
u_{\mathcal{R}}(v) = \sum_{(\varphi_i, x_i) \in \mathcal{R} : v \models \varphi_i} x_i
\]
Weighted Formula Representations

Theorem

1. The weighted formula representation is a complete representation for utility functions $u : 2^N \rightarrow \mathbb{R}$.

2. The weighted formula representation can be exponentially more compact than the naive representation.

3. There exist utility functions $u : 2^N \rightarrow \mathbb{R}$ for which the smallest weighted formula representation requires exponentially many rules.
Weighted Formula Representations

Theorem

1. Given a target value $k \in \mathbb{R}$ and rulebase $\mathcal{R}$, the problem of determining whether there exists a valuation $v$ such that $u_\mathcal{R}(v) \geq k$ is NP-complete.

2. The problem of finding an optimal valuation $v^*$ satisfying

$$v^* \in \arg \max_v u_\mathcal{R}(v)$$

is FP$^{NP}$-complete.

(This means it is as hard as the travelling salesman problem: to solve it requires a polynomial number of queries to an NP oracle.)
Preferences for Combinatorial Auctions

- Auctions for bundles of goods.
- A good example of bundles of good are spectrum licences.
- For the 1.7 to 1.72 GHz band for Brooklyn to be useful, you need a license for Manhattan, Queens, Staten Island.
- Most valuable are the licenses for the same bandwidth.
- But a different bandwidth licence is more valuable than no license.
• Let $\mathcal{Z} = \{z_1, \ldots, z_m\}$ be a set of items to be auctioned.

• We capture preferences of agent $i$ with a 

**valuation** function:

$$v_i : 2^\mathcal{Z} \rightarrow \mathbb{R}$$

• Thus, for every possible bundle of goods $Z \subseteq \mathcal{Z}$, $v_i(Z)$ says how much $Z$ is worth to $i$. 
Properties of Valuation Functions

- If 
  \[ v_i(\emptyset) = 0 \]
  then we say that the valuation function for \( i \) is \textit{normalised}.

- A common assumption is \textit{free disposal}:
  \[ Z_1 \subseteq Z_2 \quad \text{implies} \quad v_i(Z_1) \leq v_i(Z_2) \]

- Free disposal means an agent is never worse off having more stuff.
Rather than exhaustive evaluations, allow bidders to construct valuations from the bids they want to mention.

**Atomic bids** take the form

\[(Z, p)\]

where

- \(Z \subseteq \mathcal{Z}\)
- \(p \in \mathbb{R}_+\)

A bundle \(Z'\) **satisfies** a bid \((Z, p)\) if \(Z \subseteq Z'\).

In other words a bundle satisfies a bid if it contains at least the things in the bid.
Atomic Bids

- Atomic bids define valuations

\[
\nu_\beta(Z') = \begin{cases} 
    p & \text{if } Z' \text{ satisfies } (Z, p) \\
    0 & \text{otherwise}
\end{cases}
\]

- Atomic bids alone don’t allow us to construct very interesting valuations.
XOR Bids

• To construct more complex valuations, atomic bids can be combined into more complex bids.
• One approach is XOR bids

$$\beta_1 = ([a, b], 3) \text{ XOR } ([c, d], 5)$$

• XOR because we will pay for at most one.
• We read the bid to mean:

I would pay 3 for a bundle that contains a and b but not c and d. I will pay 5 for a bundle that contains c and d but not a and b, and I will pay 5 for a bundle that contains a, b, c and d.

• From this we can construct a valuation.
The valuation function corresponding to

\[ \beta_1 = (\{a, b\}, 3) \text{ XOR } (\{c, d\}, 5) \]

is thus:

\[
\begin{align*}
\nu_{\beta_1}(\{a\}) &= 0 \\
\nu_{\beta_1}(\{b\}) &= 0 \\
\nu_{\beta_1}(\{a, b\}) &= 3 \\
\nu_{\beta_1}(\{c, d\}) &= 5 \\
\nu_{\beta_1}(\{a, b, c, d\}) &= 5
\end{align*}
\]
More formally, the following XOR bid:

$$\beta = (Z_1, p_1) \text{ XOR } \cdots \text{ XOR } (Z_k, p_k)$$

defines a valuation $v_\beta$ as follows:

$$v_\beta(Z') = \begin{cases} 
0 & \text{if } Z' \text{ doesn’t satisfy any } (Z_i, p_i) \\
\max \{ p_i \mid Z_i \subseteq Z' \} & \text{otherwise}
\end{cases}$$
• XOR bids are **fully expressive**, that is they can express any valuation function over a set of goods.

• To do that, we may need an exponentially large number of atomic bids.

• However, the valuation of a bundle can be computed in polynomial time.
Lecture 3: Strategic Form Games
Part X

Game Forms and Games
In this lecture we study strategic form non-cooperative games and their solution concepts.

This is the best-known class of games.

Recall that in non-cooperative games, players must act alone – joint decisions are not possible.
• Let $N = \{1, \ldots, n\}$ be the set of **players**.
• Players simultaneously choose a **strategy**, and as a result of the combination of strategies selected, an outcome in $\Omega$ will result.
• Player $i$’s strategies are given in set $\Sigma_i$, with members $\sigma_i$ etc.
• Environment behaviour defined by **outcome function**:

$$g : \Sigma_1 \times \cdots \times \Sigma_n \to \Omega$$

• A **game form** is a structure:

$$\langle N, \Omega, \Sigma_1, \ldots, \Sigma_n, g \rangle$$
An Example Game Form

• Suppose we have \( N = \{1, 2\}, \Omega = \{\omega_1, \ldots, \omega_4\}, \) and \( \Sigma_1 = \Sigma_2 = \{C, D\}. \)

• Here is an outcome function:

\[
g(D, D) = \omega_1 \quad g(D, C) = \omega_2 \quad g(C, D) = \omega_3 \quad g(C, C) = \omega_4
\]

• This game form is sensitive to actions of both agents.
Another Game Form

\[ g(D, D) = \omega_1 \quad g(D, C) = \omega_1 \quad g(C, D) = \omega_1 \quad g(C, C) = \omega_1 \]

Neither agent has any influence in this environment.
Yet Another Game Form

\[ g(D, D) = \omega_1 \quad g(D, C) = \omega_2 \quad g(C, D) = \omega_1 \quad g(C, C) = \omega_2 \]

This environment is controlled by player 2.
Adding preferences

• Suppose we have the first case, where both agents can influence the outcome.
• Now suppose players have utility functions as follows:

\[
\begin{align*}
  u_1(\omega_1) &= 1 & u_1(\omega_2) &= 1 & u_1(\omega_3) &= 4 & u_1(\omega_4) &= 4 \\
  u_2(\omega_1) &= 1 & u_2(\omega_2) &= 4 & u_2(\omega_3) &= 1 & u_2(\omega_4) &= 4
\end{align*}
\]

• With a bit of abuse of notation:

\[
\begin{align*}
  u_1(D, D) &= 1 & u_1(D, C) &= 1 & u_1(C, D) &= 4 & u_1(C, C) &= 4 \\
  u_2(D, D) &= 1 & u_2(D, C) &= 4 & u_2(C, D) &= 1 & u_2(C, C) &= 4
\end{align*}
\]

• Agent 1’s preferences are:

\[
(C, C) \sim_1 (C, D) \succ_1 D, C \sim_1 D, D
\]

• Informally, “C” is the **rational choice** for 1. (Why?)
• In what follows, we drop the outcome function \( g \) and assume utility functions are of the form:

\[
u_i : \Sigma_1 \times \cdots \times \Sigma_n \rightarrow \mathbb{R}\]
A normal form game is a structure:

$$\langle N, \Sigma_1, \ldots, \Sigma_n, u_1, \ldots, u_n \rangle$$

where:

- $N = \{1, \ldots, n\}$ is the set players;
- $\Sigma_i$ is a set of possible strategies for player $i \in N$;
- $u_i : \Sigma_1 \times \cdots \times \Sigma_n \to \mathbb{R}$ is the utility function for agent $i \in N$.

Notice that the utility $i$ gets depends not on only her actions, but on the actions of others, and similarly for other agents. For $i$ to find the best action involves deliberating about what others will do, taking into account the fact that they will also try to maximise their utility taking into account how $i$ will act.
We can neatly summarise the game in a **payoff matrix**

<table>
<thead>
<tr>
<th></th>
<th>D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 D</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>1 C</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

- Agent **1** is the **row player**: strategies for this player correspond to rows.
- Agent **2** is the **column player**: strategies for this player correspond to columns.
- Each cell lists utilities from the corresponding outcome.
Part XI

Solution Concepts
If players act rationally, what will the outcome of the game be?

Answered by solution concepts

Key solution concepts for strategic games:
- dominant strategies
- Nash equilibria
- iterated elimination equilibrium

An key concept to understand these is the notion of best response.
A **strategy profile**, $\vec{\sigma}$, is a tuple of strategies, one for each player:

$$\vec{\sigma} = (\sigma_1, \ldots, \sigma_i, \ldots, \sigma_n) \in \Sigma_1 \times \cdots \times \Sigma_i \times \cdots \times \Sigma_n$$

We denote the strategy profile obtained by replacing the $i$ component of $\vec{\sigma}$ with $\sigma'_i$ by

$$(\vec{\sigma}_{-i}, \sigma'_i)$$

And so:

$$(\vec{\sigma}_{-i}, \sigma'_i) = (\sigma_1, \ldots, \sigma'_i, \ldots, \sigma_n)$$

We sometimes refer to $\Sigma_{-i}$, with obvious interpretation
Dominant Strategies

• Suppose you have a strategy $\sigma$, with the following property:

  no matter what choice you made, my best response to that choice would be to choose $\sigma$.

• Strategies that have this property are called dominant strategies.

• The fact that a strategy is dominant is a pretty compelling argument for choosing it: it is never a sub-optimal decision.
Dominant Strategies

Weak and Strong Varieties

Strategy $\sigma_i \in \Sigma_i$ is **weakly dominant** for $i$ if:

1. for all $\bar{\sigma}$ and for all $\sigma'_i \in \Sigma_i$, we have

$$u_i(\bar{\sigma}_{-i}, \sigma_i) \geq u_i(\bar{\sigma}_{-i}, \sigma'_i)$$

2. for some $\bar{\sigma}$ we have

$$u_i(\bar{\sigma}_{-i}, \sigma_i) > u_i(\bar{\sigma}_{-i}, \sigma'_i)$$

for all $\sigma'_i \in \Sigma_i$

Strategy $\sigma_i \in \Sigma_i$ is **strongly dominant** if

for all $\bar{\sigma}$ and for all $\sigma'_i \in \Sigma_i$, we have

$$u_i(\bar{\sigma}_{-i}, \sigma_i) > u_i(\bar{\sigma}_{-i}, \sigma'_i)$$
A dominant strategy equilibrium is a strategy profile in which every player chooses a dominant strategy.

A strong solution concept... but unfortunately, there isn’t always a dominant strategy.
• A strategy profile $\bar{\sigma}$ is a **Nash equilibrium** if no player would rather have done something else, assuming the other players stuck with their strategies.

• Formally, $\bar{\sigma}$ is a NE if there is no player $i \in N$ and strategy $\sigma'_i \in \Sigma_i$ such that

$$u_i(\bar{\sigma}_{-i}, \sigma'_i) > u_i(\bar{\sigma}).$$

• Nobody can benefit by deviating from a Nash equilibrium.
Problems with Nash Equilibrium

1. Not every game has a (pure) NE.
2. Some games have more than one NE ($\Rightarrow$ equilibrium selection problem)
3. Some NE are bad (have undesirable social properties)
Part XII

The Concept of “Best Response”
Best Responses

- A important way of understanding solution concepts is through the idea of a **best response** function.
- A player’s best response to a strategy profile $\vec{\sigma}$ is the choice that would give that player highest utility, assuming the other players made choices as defined in $\vec{\sigma}$. 
Best Responses

For each player $i$ we define a function

$$BR_i : \Sigma_1 \times \cdots \times \Sigma_n \rightarrow 2^{\Sigma_i}$$

as follows:

$$BR_i(\bar{\sigma}) = \arg\max_{\sigma_i \in \Sigma_i} u_i(\bar{\sigma}_{-i}, \sigma_i)$$

We define the grand best response as follows:

$$BR(\bar{\sigma}) = BR_1(\bar{\sigma}) \times \cdots \times BR_n(\bar{\sigma})$$
NE as a Fixed Point of the Best Response Function

- A value $s \in S$ is a **fixed point** of a function $f : S \rightarrow 2^S$ if $s \in f(s)$.
- NE can naturally be characterised in terms of best responses.

**Lemma**

\[ \bar{\sigma} \in NE(G) \]

iff

\[ \bar{\sigma} \in BR(\bar{\sigma}) \]
Lemma

If $\sigma_i \in \Sigma_i$ is a dominant strategy for player $i$ then

$$\sigma_i \in \bigcap_{\vec{\sigma} \in \Sigma} BR_i(\vec{\sigma})$$
Part XIII

Social Welfare
What would a Benevolent God choose?

• Suppose an omniscient, impartial, benevolent external entity was able to choose the outcome of the game.
• What would they choose?
• Intuitively, the outcome that is best for the society
• This is the realm of social welfare
• The answer is not obvious, because interpersonal comparisons of utility are very difficult
• Key notions:
  • Pareto optimality
  • utilitarian social welfare
  • egalitarian social welfare
A strategy profile is **Pareto optimal** (a.k.a. **Pareto efficient**)
if there is no other outcome that makes one agent **better off**
without making another agent **worse off**.

- If $\vec{\sigma}$ **is** Pareto optimal, then at least one agent will be reluctant
to move away from it (because this agent will be worse off).

- If $\vec{\sigma}$ **is not** Pareto optimal, then $\vec{\sigma}$ is **inefficient**: it is “wasting”
  utility.

- Pareto optimality is probably the least contentious notion of
  social welfare.
Utilitarian Social Welfare

• The utilitarian social welfare of $\vec{\sigma}$ is the **sum of utilities** that each agent gets from $\vec{\sigma}$

• An outcome $\vec{\sigma}^*$ that maximises utilitarian social welfare thus satisfies:

$$\vec{\sigma}^* \in \arg \max_{\vec{\sigma}} \sum_{i \in N} u_i(\vec{\sigma})$$

• Intuitively the “total amount of wealth that $\vec{\sigma}$ creates”.

• Problem: it doesn’t look at the **distribution** of utility

• Appropriate when the whole system (all agents) has a single owner (then overall benefit of the system is important, not individuals).
Egalitarian Social Welfare

- Egalitarian social welfare says that we should **try to make the worst off member of society as well off as possible**.
- A strategy profile $\vec{\sigma}^*$ that maximises egalitarian social welfare will satisfy:

$$\vec{\sigma}^* \in \arg \max_{\vec{\sigma}'} \min_{i \in N} \{ u_i(\vec{\sigma}') \}$$
## Relationship Between Solution Concepts

<table>
<thead>
<tr>
<th>Theorem</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Every dominant strategy equilibrium is a Nash equilibrium, but the converse need not be the case.</td>
</tr>
<tr>
<td>2 Nash equilibria and dominant strategy need not be Pareto efficient, nor need they maximise utilitarian/egalitarian social welfare.</td>
</tr>
<tr>
<td>3 Any outcome that maximises utilitarian social welfare is Pareto efficient, but the converse need not be the case.</td>
</tr>
</tbody>
</table>
Part XIV

Some Important Games
The Prisoner’s Dilemma

“Two men are collectively charged with a crime and held in separate cells, with no way of meeting or communicating. They are told that:

- if one confesses and the other does not, the confessor will be freed, and the other will be jailed for three years;
- if both confess, then each will be jailed for two years.

Both prisoners know that if neither confesses, then they will each be jailed for one year.”
### Payoff matrix for the Prisoner’s Dilemma

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>defect</td>
<td>−2</td>
<td>−3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>−1</td>
</tr>
<tr>
<td>coop</td>
<td>−3</td>
<td>−1</td>
</tr>
</tbody>
</table>
Payoff matrix for the Prisoner’s Dilemma

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>defect</td>
<td>coop</td>
</tr>
<tr>
<td>defect</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

- Top left: If both defect, then both get punishment for mutual defection: two years in jail.
- Top right: If 2 cooperates and 1 defects, 2 gets sucker’s payoff (3 yrs jail) while 1 goes free.
- Bottom left: If 1 cooperates and 2 defects, 1 gets sucker’s payoff, 2 goes free.
- Bottom right: Reward for mutual cooperation, 1 year in jail.
Consider player 1’s analysis:

- Suppose 2 defects: my best response is to defect.
- Suppose 2 cooperates: my best response is to defect.

Defection is a best response to all of 2’s actions.

$\Rightarrow$ defection is a dominant strategy for 1.

- The game is symmetric: defection is also a dominant strategy for player 2.

- Mutual defection is a dominant strategy equilibrium $\Rightarrow$ each serve two years in jail.

- But intuition says this is not the best outcome:
  - Surely they should both cooperate – then they each serve just one year in jail!
Consider player 1’s analysis:

- Suppose 2 defects: my best response is to defect.

Defection is a best response to all of 2's actions. 

⇒ defection is a dominant strategy for 1.

- The game is symmetric: defection is also a dominant strategy for player 2.

- Mutual defection is a dominant strategy equilibrium ⇒ each serve two years in jail.

But intuition says this is not the best outcome: Surely they should both cooperate – then they each serve just one year in jail!
Dominant Strategy Analysis

- Consider player 1’s analysis:
  - Suppose 2 defects: my best response is to defect.
  - Suppose 2 cooperates: my best response is to defect.

\[\Rightarrow \text{defection is a dominant strategy for 1.}\]

- The game is symmetric: defection is also a dominant strategy for player 2.

- Mutual defection is a dominant strategy equilibrium \[\Rightarrow \text{each serve two years in jail}\]

- But intuition says this is not the best outcome: surely they should both cooperate – then they each serve just one year in jail!
Dominant Strategy Analysis

- Consider player 1’s analysis:
  - Suppose 2 defects: my best response is to defect.
  - Suppose 2 cooperates: my best response is to defect.
  - Defection is a best response to all of 2’s actions.

- The game is symmetric: defection is also a dominant strategy for player 2.

- Mutual defection is a dominant strategy equilibrium ⇒ each serve two years in jail.

- But intuition says this is not the best outcome: Surely they should both cooperate – then they each serve just one year in jail!
Dominant Strategy Analysis

• Consider player 1’s analysis:
  - Suppose 2 defects: my best response is to defect.
  - Suppose 2 cooperates: my best response is to defect.
  - Defection is a best response to all of 2’s actions.
  - ⇒ defection is a dominant strategy for 1.

• The game is symmetric: defection is also a dominant strategy for player 2.
• Mutual defection is a dominant strategy equilibrium ⇒ each serve two years in jail
• But intuition says this is not the best outcome:
  Surely they should both cooperate – then they each serve just one year in jail!
Solution Concepts

- $(D, D)$ is a dominant strategy equilibrium.
- $(D, D)$ is the only Nash equilibrium.
- All outcomes except $(D, D)$ are Pareto optimal.
- $(C, C)$ maximises social welfare.
The Dilemma!

• This apparent paradox has been described as “a fundamental problem of multi-agent interactions”.
• Real world examples:
  • nuclear arms reduction (“why don’t I keep mine. . .”)
  • free rider systems — public transport;
  • in the UK — television licenses.
• The prisoner’s dilemma is ubiquitous.
• Can we recover cooperation?
Arguments for Recovering Cooperation

• Conclusions that some have drawn from this analysis:
  • the game theory notion of rational action is wrong!
  • somehow the dilemma is being formulated wrongly

• Arguments to recover cooperation:
  • We are not all machiavelli!
  • The other prisoner is my twin!
  • Program equilibria and mediators
  • The shadow of the future...
The Game of Chicken

• Think of James Dean in **Rebel without a Cause**: swerving = coop, driving straight = defect.

• Difference to prisoner’s dilemma:

  **Mutual defection is most feared outcome.**

  (Whereas sucker’s payoff is most feared in prisoner’s dilemma.)
• There is no dominant strategy.
• Strategy pairs \((C, D)\) and \((D, C)\) are pure NE.
• All outcomes except \((D, D)\) are Pareto optimal.
• All outcomes except \((D, D)\) maximise social welfare.
• An **anti-coordination game**: players should choose **different** strategies.
A Coordination Game
How to choose between multiple similar equilibria?

Here \((C, C)\) and \((D, D)\) are pure NE, but how do the players independently choose which to select?

A coordination game, because the problem faced by players is how to coordinate.
Focal points:
Sometimes outcomes in games have features that make them stand out, independently of the utility structure in games⁴.
Example: Suppose we are visiting Paris for a day, and get separated. Where do we meet up? In terms of utility, any place would do, but likely to pick a “landmark” → Eiffel Tower.

Evolutionary approaches:
If we have time, we learn to coordinate (cf. ESS).

---

If it was a matter of hunting a deer, everyone well realised that he must remain faithful to his post; but if a hare happened to pass within reach of one of them, we cannot doubt that he would have gone off in pursuit of it without scruple.”

Rousseau (A Discourse on Inequality)
The Stag Hunt

• Another social dilemma, but less painful than the prisoner’s dilemma.
• Here there are two pure NE.

<table>
<thead>
<tr>
<th></th>
<th>deer</th>
<th>hare</th>
</tr>
</thead>
<tbody>
<tr>
<td>deer</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>hare</td>
<td>0</td>
<td>7</td>
</tr>
</tbody>
</table>

1 2

150
Players 1 and 2 each have a $1 coin

They simultaneously show one face of their coin (either “heads” or “tails”)

If they show the same face, then 1 takes 2’s coin.

If they show different faces, then 2 takes 1’s coin.
Matching Pennies Payoff Matrix
Where preferences of agents are diametrically opposed we have **strictly competitive** scenarios.

Zero-sum encounters are those where utilities sum to zero:

$$\sum_{i \in N} u_i(\omega) = 0 \quad \text{for all } \omega \in \Omega.$$ 

Zero sum encounters are bad news: for me to get +ve utility **you have to get negative utility**! The best outcome for me is the **worst** for you!

Zero sum encounters in real life are very rare . . . but people frequently act as if they were in a zero sum game.
Part XV

Eliminating Dominated Strategies
Given $\sigma_1, \sigma_2 \in \Sigma_i$, we say that $\sigma_1$ strictly dominates $\sigma_2$ if

$$\forall \sigma_{-i} \in \Sigma_{-i} \quad u_i(\sigma_{-i}, \sigma_1) > u_i(\sigma_{-i}, \sigma_2)$$

Thus, $\sigma_1$ is always a better choice than $\sigma_2$, no matter what the others choose... so:

a rational agent will never play
a strictly dominated strategy.
We can thus “simplify” games by deleting dominated strategies.

Suppose we are given a game $G$. Then:

1. Let $G_0 = G$ and let $t = 0$
2. Does $G_t$ contain any strictly dominated strategies? If not, terminate.
3. Delete any strictly dominated strategies from $G_t$ to obtain a new (simpler) game $G_{t+1}$, set $t = t + 1$, and go to step (1).
• Suppose that after IEDS, we have a **single** outcome remaining. . .

• . . . then we say the game is **dominance-solvable**.

**Lemma**

*If a game* $G$ *is dominance-solvable, then the unique outcome of the game according to IEDS is the unique pure-strategy Nash equilibrium of* $G$.

Proof: exercise.
Observe that C is dominated by R, so delete it.
Observe that C is dominated by R, so delete it
An Example

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>M</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
An Example

Here, both M and B are dominated by T
In this game, R is dominated by L, and with one final elimination, the outcome of the game is \((T, L)\).
Evaluating IEDS

- IEDS guarantees a solution exists
- Does not guarantee solution is unique
- Often fails to make any useful predictions (all outcomes survive)
- Powerful when it can be applied
- Hinges on common knowledge of rationality
Part XVI

Computing Pure NE
An Algorithm for Computing Pure NE

- In each **column**, underline the utilities of the **row player** corresponding to the best choice for that player (i.e., underline the largest **blue** number(s) in each column)
- In each **row**, underline the utilities of the **column player** corresponding to the best choice for that player (i.e., underline the largest **red** number(s) in each row)
- Any cell with both payoffs underlined is a NE
- Any row with all one player’s payoffs underlined is a DS; similarly for columns
Consider the **defect** column:

<table>
<thead>
<tr>
<th></th>
<th>defect</th>
<th>coop</th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

**Table:**

- Defect:
  - Nash Equilibrium -2
- Cooperate:
  - Nash Equilibrium -3

**Matrix:**

- \( \begin{array}{c}
  \text{defect} \\
  \text{coop}
  \end{array} \) = \( \begin{array}{cc}
  2 & -2 \\
  -2 & 0
  \end{array} \)
Consider the *cooperate* column:

<table>
<thead>
<tr>
<th></th>
<th>defect</th>
<th>coop</th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

Nash Equilibrium Analysis
Consider the **defect** row:

<table>
<thead>
<tr>
<th></th>
<th>defect</th>
<th>coop</th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>1</td>
<td>-3</td>
<td></td>
</tr>
</tbody>
</table>
Consider the cooperate row:

<table>
<thead>
<tr>
<th></th>
<th>defect</th>
<th>coop</th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>
• Simplest approach to computing pure NE is **exhaustive search**.
• Assume we can compute the value $u_i(\bar{\sigma})$ in unit time.
• The exhaustive search will take time $O(|\Sigma_1 \times \cdots \times \Sigma_n|)$, i.e., exponential in number of agents.
Exhaustive Search for Pure Nash Equilibrium
(Only for finite games!)

\[
\text{for } \vec{\sigma} \in \Sigma_1 \times \cdots \times \Sigma_n \text{ do}
\]
\[
\text{found} = \top
\]
\[
\text{for } i \in N \text{ do}
\]
\[
\text{for } \sigma'_i \in \Sigma_i \text{ do}
\]
\[
\text{if } u_i(\vec{\sigma}_{-i}, \sigma'_i) > u_i(\vec{\sigma}) \text{ then}
\]
\[
\text{found} \leftarrow \bot
\]
\[
\text{end if}
\]
\[
\text{end for}
\]
\[
\text{end for}
\]
\[
\text{if } \text{found} \text{ then}
\]
\[
\text{return } \vec{\sigma}
\]
\[
\text{end if}
\]
\[
\text{end for}
\]
\[
\text{return } \text{“no pure NE found”}
\]
An alternative can be more efficient in some cases.

In **myopic best response** we search for a solution with unhappy players flipping their strategies to a best responses.

In some cases, MBR works well; but not guaranteed.

If it terminates, it gives a pure NE; but it is not guaranteed to terminate.
\( \bar{\sigma} \leftarrow \text{random element of } \Sigma_1 \times \cdots \times \Sigma_n \)

while exists player \( i \) who is not playing best response in \( \bar{\sigma} \) do

\( \sigma'_i \leftarrow \text{an element of } BR_i(\bar{\sigma}) \)

\( \bar{\sigma} \leftarrow (\bar{\sigma}_{-i}, \sigma'_i) \)

end while

return \( \bar{\sigma} \)
An Example where Myopic Best Response Fails

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>1</td>
<td>-1</td>
<td>-2</td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td>1</td>
<td>-2</td>
</tr>
<tr>
<td>M</td>
<td>-1</td>
<td>1</td>
<td>-2</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-1</td>
<td>-2</td>
</tr>
<tr>
<td>B</td>
<td>-2</td>
<td>-2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>-2</td>
<td>-2</td>
<td>2</td>
</tr>
</tbody>
</table>

What happens if we start myopic best response at \((T, L)\)?
A natural question: are there classes of games in which pure NE are guaranteed to exist?

One natural class of games satisfying this property is potential games.

An important class of potential games is congestion games.
Potential Games

A game

\[ \langle N, \Sigma_1, \ldots, \Sigma_n, u_1, \ldots, u_n \rangle \]

is said to be a **potential game** if there is a function

\[ P : \Sigma_1 \times \cdots \times \Sigma_n \rightarrow \mathbb{R} \]

such that

1. for all players \( i \in N \),
2. for all strategy profiles \( \bar{\sigma} \),
3. for all strategies \( \sigma_i \in \Sigma_i \) and \( \sigma'_i \in \Sigma_i \)

we have

\[ u_i(\bar{\sigma}_{-i}, \sigma_i) - u_i(\bar{\sigma}_{-i}, \sigma'_i) = P(\bar{\sigma}_{-i}, \sigma_i) - P(\bar{\sigma}_{-i}, \sigma'_i) \]
Theorem

Every (finite) potential game has a pure strategy Nash equilibrium.

Proof:

1. Find a \( \bar{\sigma}^* \) that maximises the value of \( P \).
   (Since the games are finite, such a \( \bar{\sigma}^* \) is guaranteed to exist, though need not be unique.)

2. Claim: \( \bar{\sigma}^* \) is a pure NE.
Part XVII

Computational Considerations
Computational considerations

- **Issues of representation:**
  In a game with $n$ players, where each player has $m$ strategies, there are $m^n$ possible outcomes: how do we represent utility functions $u_i(\cdots)$ in this case?

- **Complexity issues:**
  NE, PO, etc involve **quantifying over strategies**.
  Checking whether a game has a pure NE is NP-hard, even under very restrictive assumptions\(^5\)

---

A Boolean game consists of:

- \( N = \{1, \ldots, n\} \) (the players)
- \( \Phi = \{p, q, \ldots\} \) (a finite set of Boolean variables)
- \( \Phi_i \) for each \( i \in N \) (the set of variables under the control of \( i \))
- The assignments that \( i \) can make to \( \Phi_i \) are the actions available to \( i \).
- \( \gamma_i \) (goal of agent \( i \) – the specification for \( i \) – propositional logic formula over \( \Phi \))
Outcomes

- A **choice** for agent \(i\) is an assignment
  \[ \nu_i : \Phi_i \rightarrow \mathbb{B} \]
  Agent \(i\) chooses a value for all its variables.

- An **outcome** is a **collection of choices**, one for each agent:
  \[ (\nu_1, \ldots, \nu_n) \]
The utility of outcome \((v_1, \ldots, v_n)\) to player \(i\) is:

\[
    u_i(v_1, \ldots, v_n) = \begin{cases} 
        1 & \text{if } (v_1, \ldots, v_n) \models \gamma_i \\
        0 & \text{otherwise.}
    \end{cases}
\]

We can then define NE in the standard way.
An Example

Suppose:

\[ \Phi_1 = \{p\} \]
\[ \Phi_2 = \{q, r\} \]
\[ \gamma_1 = q \]
\[ \gamma_2 = q \lor r \]

Then \( \gamma_1 \land \gamma_2 \) is satisfied in NE.
Another Example
Matching pennies as a Boolean game

Suppose:

\[ \Phi_1 = \{ p \} \]
\[ \Phi_2 = \{ q \} \]
\[ \gamma_1 = p \leftrightarrow q \]
\[ \gamma_2 = \neg (p \leftrightarrow q) \]

There is no NE in this game.
Complexity of Boolean Games

Theorem

It is co-NP-complete to check whether an outcome forms a NE in a Boolean game.

It is $\Sigma^p_2$-complete to check whether a Boolean game has a NE.
Lecture 4: Mixed Strategies and Nash’s Theorem
Recall the game of **matching pennies**: 

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>T</td>
<td>−1</td>
<td>−1</td>
</tr>
</tbody>
</table>

• No pair of strategies forms a pure NE in matching pennies: whatever pair of strategies is chosen, somebody wishes they had done something else.

• The solution is to allow **mixed strategies**:
  - play “heads” with probability 0.5
  - play “tails” with probability 0.5

• If both players do this, we have a NE strategy profile.
Nash Equilibria in Mixed Strategies

In a mixed strategy NE, both players:

- play “heads” with probability 0.5
- play “tails” with probability 0.5

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>T</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>-1</td>
</tr>
</tbody>
</table>

- play “heads” with probability 0.5
- play “tails” with probability 0.5
Mixed Strategies

A mixed strategy $ms$ for $i$ is a probability distribution over the pure strategies $\Sigma_i$, hence has the form

- play $\sigma_1$ with probability $p_1$
- play $\sigma_2$ with probability $p_2$
- ... 
- play $\sigma_k$ with probability $p_k$.

which must satisfy probability constraints:

$$p_1 + p_2 + \cdots + p_k = 1$$
$$p_i \in [0, 1] \text{ for all } 1 \leq i \leq k$$

Let $MS_i = \Delta \Sigma_i$ be the set of all mixed strategies for player $i$.

$\Rightarrow$ We are in the domain of expected utility. $\Leftarrow$
A game is finite if $\Sigma_i$ is finite for all $i \in N$.

**Theorem (Nash, 1950)**

*Every finite game has a Nash equilibrium in mixed strategies.*

- Guarantees the existence of NE
- But what about **computing** NE...?
Part XVIII

Best Response Functions
• A important way of understanding solution concepts – and the key to Nash’s theorem – is the idea of a best response function.

• A player’s best response to a mixed strategy profile $\vec{m}$ is the choice that would give that player highest expected utility, assuming the other players made choices as defined in $\vec{m}$. 
For each player $i$ we define a function

$$BR_i : MS_1 \times \cdots \times MS_n \rightarrow 2^{MS_i}$$

as follows:

$$BR_i(m\vec{s}) = \arg\max_{m_{s_i} \in MS_i} EU_i(m\vec{s}_{-i}, ms_i)$$

We define the grand best response as follows:

$$BR(m\vec{s}) = BR_1(m\vec{s}) \times \cdots \times BR_n(m\vec{s})$$
• A value $s \in S$ is a fixed point of a function $f : S \rightarrow 2^S$ if $s \in f(s)$.
• A value $s \in S$ is a fixed point of $f : S \rightarrow 2^S$ if $s \in f(s)$.
• The following characterises NE in terms of the grand best response function.

**Lemma**

\[ \vec{\sigma} \in NE(G) \]  
iff  
\[ \vec{\sigma} \in BR(\vec{\sigma}) \]
Part XIX

The Indifference Principle Part 1:
2 × 2 Games
Two Player Games

• Mixed Nash equilibria for $2 \times 2$ games can easily be computed
• We need some more definitions and results to get there...
• The technique we use is called the **indifference principle**
The support of a mixed strategy

$$ms_i : \Sigma_i \rightarrow [0, 1]$$

is the set of pure strategies played with +ve probability in $$ms_i$$:

$$supp(ms_i) = \{ \sigma \mid ms_i(\sigma) > 0 \}$$
In what follows, we will work with the following generic $2 \times 2$ game.

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>T</strong></td>
<td>$v_1^1$</td>
<td>$v_2^2$</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>$v_3^1$</td>
<td>$v_3^2$</td>
</tr>
</tbody>
</table>

- The *superscript* identifies the player (1 or 2)
- The *subscript* identifies the cell (1 to 4)
Mixed Strategies in $2 \times 2$ Games

Represent a mixed strategy for player 1 as a value $p \in [0, 1]$:

- play $T$ with probability $p$
- play $B$ with probability $1 - p$

Represent a mixed strategy for player 2 as a value $q \in [0, 1]$:

- play $L$ with probability $q$
- play $R$ with probability $1 - q$

A mixed strategy profile is a pair $(p, q)$.
Suppose \((p, q)\) is a pair of mixed strategies. Define:

\[
EU_1(T, q) = (v_1^1 \times q) + (v_2^1 \times (1 - q))
\]
\[
EU_1(B, q) = (v_1^3 \times q) + (v_4^1 \times (1 - q))
\]
\[
EU_2(L, p) = (v_1^2 \times p) + (v_3^2 \times (1 - p))
\]
\[
EU_2(R, p) = (v_2^2 \times p) + (v_4^2 \times (1 - p))
\]
Indifference Principle for 2x2 Games

**Theorem**

A pair of probabilities \((p, q)\) is a mixed strategy Nash equilibrium in the generic \(2 \times 2\) game iff:

\[
EU_1(T, q) = EU_1(B, q) \quad \text{and} \\
EU_2(L, p) = EU_2(R, p)
\]

This is a **special case** of a **general result**, called the **indifference principle**: The expected payoff you would get from all pure strategies in the support of a NE is the same.
Algorithm for Computing Mixed NE in 2x2 Games

1. Check for **pure** NE. (If you find any, you are done.)
2. Otherwise: consider the following equality:

\[
EU_1(T, q) = EU_1(B, q)
\]

Find solutions for \(q\).
3. Then consider the following equality:

\[
EU_2(L, p) = EU_2(R, p)
\]

Find solutions for \(p\)
4. Any pair of solutions \((p, q)\) defines a mixed NE

Since we are dealing with **linear equalities**, we can solve them in polynomial time.
First we find $q$:

\[
EU_1(T, q) = EU_1(B, q)
\]

\[
(v_1^1 \times q) + (v_2^1 \times (1 - q)) = (v_3^1 \times q) + (v_4^1 \times (1 - q))
\]

\[
(1 \times q) + (-1 \times (1 - q)) = (-1 \times q) + (1 \times (1 - q))
\]

\[
2q - 1 = 1 - 2q
\]

\[
4q = 2
\]

\[
q = 0.5
\]
Now we find $p$:

\[
\begin{align*}
EU_2(L, p) &= EU_2(R, p) \\
(v_1^2 \times p) + (v_3^2 \times (1 - p)) &= (v_2^2 \times p) + (v_4^2 \times (1 - p)) \\
(-1 \times p) + (1 \times (1 - p)) &= (1 \times p) + (-1 \times (1 - p)) \\
1 - 2p &= 2p - 1 \\
2 &= 4p \\
p &= 0.5
\end{align*}
\]

Hence $(0.5, 0.5)$ is a mixed NE in matching pennies.
The Indifference Principle

The $2 \times 2$ instance is a special case of a general result.

Theorem (Indifference Principle)

A mixed strategy profile $\vec{ms} = (ms_1, \ldots, ms_n)$ is a NE iff

1. for all $i \in N$, and
2. for all $\sigma_1, \sigma_2 \in supp(ms_i)$, we have

$$EU_i(\sigma_1, \vec{ms}_{-i}) = EU_i(\sigma_2, \vec{ms}_{-i})$$
Part XX

Illustrating Nash’s Theorem in the $2 \times 2$ Case
We can “prove” Nash’s theorem in the $2 \times 2$ case by plotting the best response functions $BR_1$ and $BR_2$ against each other.

The best response functions can be derived from the equations we saw earlier:

$BR_1(q) = \begin{cases} 
\{0\} & \text{if } q < 0.5 \\
[0, 1] & \text{if } q = 0.5 \\
\{1\} & \text{if } q > 0.5 
\end{cases}$

$BR_2(p) = \begin{cases} 
\{1\} & \text{if } p < 0.5 \\
[0, 1] & \text{if } p = 0.5 \\
\{0\} & \text{if } p > 0.5 
\end{cases}$
The Best Response Function $BR_1(q)$

$$BR_1(q) = \begin{cases} 
\{0\} & \text{if } q < 0.5 \\
[0, 1] & \text{if } q = 0.5 \\
\{1\} & \text{if } q > 0.5 
\end{cases}$$
The Best Response Function \( BR_2(p) \)

\[
BR_2(p) = \begin{cases} 
\{1\} & \text{if } p < 0.5 \\
[0, 1] & \text{if } p = 0.5 \\
\{0\} & \text{if } p > 0.5 
\end{cases}
\]
Plotting $BR_1(q)$ against $BR_2(p)$

(Apologies for the deeply distasteful shape of the graph.)
Part XXI

Nash’s Theorem
The key to Nash’s result are a class of results in algebraic topology, known as fixed point theorems.

Recall that a function \( f : S \rightarrow S \) has a fixed point if there is some value \( x \in S \) such that \( f(x) = x \).

Fixed point theorems characterise the existence of fixed point in functions with respect to their properties.

Nash’s theorem can be proved via Brouwer’s fixed point theorem or Kakutani’s fixed point theorem.
Brouwer’s Fixed Point Theorem

Theorem (Brouwer, 1909)

Let \( S \) be a convex, bounded, closed set and let \( f : S \rightarrow S \) be a continuous function from \( S \) to itself. Then \( f \) has a fixed point.

- **convex**: \( S \) does not contain “holes”
- **bounded**: every element is within a “fixed distance” of every other element
- **closed**: contains its own end points
- **continuous**: you can plot the function “without lifting pen from paper”.

We will now prove these conditions are **necessary** for the existence of a fixed point; the proof that they are **sufficient** is substantially more involved.
A set $S \subseteq \mathbb{R}^k$ is convex if it “contains no holes”.

For any two elements $A, B \in S$, convexity requires that all points on the straight line connecting $A$ to $B$ are contained in $S$.

**Example**

Let $S \subseteq \mathbb{R}^2$ define a circle and let $f : S \rightarrow S$ map every point on the circle to the point $90^\circ$ anticlockwise. Clearly $f$ is continuous. No fixed point!
Brouwer’s Theorem: Boundedness

• A set \( S \subseteq \mathbb{R} \) is **bounded** if it has upper and lower bounds, i.e., values \( x \) and \( y \) such that for all values \( z \in \mathbb{R} \), we have \( x \leq z \) and \( y \geq z \).

• For multiple dimensions, boundedness generalises: all points are within a fixed distance of each other.

**Example**

Consider \( S = \mathbb{R}_+ \), and define \( f : S \to S \) by \( f(x) = x + 1 \). No fixed point!
Brouwer’s Theorem: Continuity

- Intuitively a function $f$ is continuous if it

Example

Let $S = [0, 1]$, and define $f : S \rightarrow S$ by

$$f(x) = \begin{cases} 
0.7 & \text{if } x \leq 0.5 \\
0.3 & \text{otherwise.}
\end{cases}$$

Clearly no fixed point.
Brouwer’s Theorem: Closed Set

- Recall that a set is closed if it contains its own end points.
- The set \([0, 1]\) is closed, while \((0, 1)\) is not.

**Example**

Let \(S = [0, 1)\), and define

\[
f(x) = \frac{x + 1}{2}
\]

This function shifts every point to the right, and while \(f(x) \to 1\) as \(x \to 1\), it does not have a fixed point.

If \(S = [0, 1]\), however, the function **does** have a fixed point: \(f(1) = 1\).
Theorem

Every continuous function \( f : [0, 1] \rightarrow [0, 1] \) has a fixed point.

The proof is easy:

1. Plot the function \( f(x) \) for \( x \) from 0 to 1 inclusive.
2. At some point, the line \textbf{must} cross the diagonal line \( y = x \).
   (Now see the need for \( f \) to be continuous, and to include end points 0 and 1.)
3. At the point \((x, y)\) where crosses the diagonal, we have \( y = f(x) = x \), i.e., a fixed point.
Theorem (Kakutani, 1940.)

Suppose $S$ is a non-empty, convex, closed, bounded, non-empty subset of $\mathbb{R}^n$, and suppose $f : S \rightarrow 2^S$ is such that $f(s)$ is non-empty and convex, and $f$ has a closed graph. Then $f$ has a fixed point.

(The conditions are essentially generalisations of the conditions for Brouwer.)
Nash’s Theorem

**Theorem (Nash, 1950)**

*Every finite game G has a Nash equilibrium in mixed strategies.*

**Proof.**

1. The Nash equilibria of G are precisely the fixed points of the grand best response function $BR$.
2. The grand best response function satisfies the conditions of Kakutani’s fixed point theorem.

All the work of the proof of Nash’s theorem is therefore in showing that the grand best response function satisfies Kakutani’s conditions.
Computational Game Theory

Lecture 5: Dynamic Games
Part XXII

Introduction
• Strategic form games assume that **players make just one move**, and that **this move is made ignorance of the moves of others**

• (“Simultaneous moves” is really an **informational** assumption, not a **temporal** one!)

• In many settings, games have a much richer dynamic and informational structure.

• In this lecture study three classes of dynamic games:
  - extensive form games
  - iterated games
  - evolutionary games
Part XXIII

Extensive Form Games
Extensive Form Games

- Many games of interest involve **multiple moves**.
- Information may or may not be available about previous moves.
- **Extensive form games** model scenarios with this structure:
  - **games of perfect information**: players know exactly how the current state of the game was reached.
  - **games of imperfect information**: players may be uncertain about previous moves, may not know how the reached the current game state.
  - **games of imperfect recall**: players may forget things they knew previously.
- (In this lecture, we restrict ourselves to games with **no chance moves**.)
Extensive form games are usually modelled as game trees.

- A finite tree structure $T$, with vertices $V$, edges $E \subseteq V \times V$, and root $v_0$
- The leaves of $T$, denoted $\text{leaves}(T)$ are end games, and are labelled with payoffs for each player:
  \[ u_i : \text{leaves}(T) \to \mathbb{R} \]
- Interior nodes of $T$ are decision nodes, and are labelled with the player who makes a move at that point
- Each edge corresponds to a move or action that can be made by that player.
- The player at the root of the tree moves first.
Game Trees

Let $V_i$ denote the decision nodes for player $i$. We require:

- $V_i \cap V_j = \emptyset$ for $i \neq j$
- $V_i \cap \text{leaves}(T) = \emptyset$
- $V = V_1 \cup \cdots \cup V_n \cup \text{leaves}(T)$

(What do these conditions mean?)
Game Trees

Let $V_i$ denote the decision nodes for player $i$. We require:

- $V_i \cap V_j = \emptyset$ for $i \neq j$
- $V_i \cap \text{leaves}(T) = \emptyset$
- $V = V_1 \cup \cdots \cup V_n \cup \text{leaves}(T)$

(What do these conditions mean?)

- Each edge $(v, v') \in E$ is labelled with an action $a(v, v')$
- Let $A(v) = \{ a(v, v') | (v, v') \in E \}$ be the actions available at vertex $v$
- We require that $a(v, v') = a(v, v'')$ implies $v' = v''$

(What does this condition mean?)
Let $V_i$ denote the decision nodes for player $i$. We require:

$V_i \cap V_j = \emptyset$ for $i \neq j$

$V_i \cap \text{leaves}(T) = \emptyset$

$V = V_1 \cup \cdots \cup V_n \cup \text{leaves}(T)$

(What do these conditions mean?)

- Each edge $(v, v') \in E$ is labelled with an action $a(v, v')$
- Let $A(v) = \{a(v, v') \mid (v, v') \in E\}$ be the actions available at vertex $v$
- We require that $a(v, v') = a(v, v'')$ implies $v' = v''$
  (What does this condition mean?)
- Let $A_i$ be the total set of actions available to $i$ in the game:

$$A_i = \bigcup_{v \in V_i} A(v)$$
A (pure) strategy, $\sigma_i$, for player $i \in N$ is a function that selects a move for every decision node labelled with $i$, i.e.,

$$\sigma_i : V_i \rightarrow A_i$$

such that

$$\sigma_i(v) \in A(v)$$

(What does this condition mean?)
Strategies in Extensive Form Games

- **A (pure) strategy**, $\sigma_i$, for player $i \in N$ is a function that selects a move for every decision node labelled with $i$, i.e.,

$$\sigma_i : V_i \rightarrow A_i$$

such that

$$\sigma_i(v) \in A(v)$$

(What does this condition mean?)

- Let $\Sigma_i$ be the set of pure strategies for $i \in N$

- A strategy profile $\bar{\sigma} = (\sigma_1, \ldots, \sigma_n)$ induces a unique path in the tree, leading to an leaf node labelled with payoffs for players

- **IMPORTANT**: A strategy for $i$ defines a choice for all decision nodes $V_i$
Two players: $N = \{E, A\}$.
First player to move is $E$; he can perform either $L$ or $R$ moves.
• Use backward induction to label every node with payoff profile that would be achieved in equilibrium (dynamic programming).

• Repeat the following:
  • For each decision node $v \in V$:
    • If all the children of $v$ have been labelled with a payoff profile, then label $v$ with a payoff profile from a child that maximises the payoff of the player making the decision at that node. (If there is a choice, choose arbitrarily.)

until all vertices have been labelled with payoff profiles.
Recall our game:

To illustrate the algorithm, we delete parts of the game tree that we have already “processed”.
Initially, start with A’s bottom left choice: given a choice between 1 and 2, he will choose 2, i.e., move “r”.
Now consider A’s bottom right choice: given a choice between 1 and 2, he will choose 2, i.e., move “r”.

Illustrating Zermelo’s Algorithm
Now consider $E$’s choice: he has a choice between 0 and 1 so will choose 1.
So, player $E$ receives 1 in equilibrium, while player $A$ receives 2. We write an equilibrium in an extensive form game by listing the actions for each player in turn. In this case: $(R, r)$. 

1, 2
Properties of Zermelo’s Algorithm

**Theorem**

Zermelo’s algorithm terminates, leaving the root labelled with a payoff profile that would be obtained by a NE strategy profile.

The algorithm runs in time polynomial in the size of the game tree.
Properties of Extensive Form Games

**Theorem**

1. Every extensive form game (with perfect information and no chance moves) has a NE in pure strategies.
2. Pure strategy NE in extensive form games can be computed in polynomial time with Zermelo’s algorithm.
3. If no two leaf nodes have the same utility for any player, then the NE is unique.

Proof: Zermelo’s algorithm.
Zermelo’s Algorithm in Computer Science

- One of the most phenomenally useful algorithms in computer science.
- Classic example of **dynamic programming**.
- Same algorithm is used in:
  - CTL model checking\(^6\)
  - Computing optimal policies in Markov decision processes via “value iteration”\(^7\)

---


Part XXIV

Subgame Perfect Nash Equilibrium
An Extensive Form Game with a Paradoxical NE

What does Zermelo give when applied to this?
An Extensive Form Game with a Paradoxical NE

Zermelo tells us that $(R, r)$ is a NE, which makes sense.
But \((L, l)\) is also a NE:

- if \(E\) plays \(L\) then \(A\) will get 0 whatever she does.  
  \(\Rightarrow\) \(l\) is a best response to \(L\)

- if \(A\) chooses \(l\) then \(E\) has a choice of choosing \(L\) and receiving 0, or playing \(R\) and receiving 0. 
  \(\Rightarrow\) \(L\) is a best response to \(l\)
An Extensive Form Game with a Paradoxical NE

Here, $A$ is **threatening** to play $l$.

... but is this threat **credible**? If $A$ is ever called on to make a choice, we would be irrational to choose $l$!

This is a weakness of NE in extensive form games.

We need a refinement of NE, due to Reinhard Selten, called **subgame perfect Nash equilibrium (SPNE)**.
• To define SPNE, we need the notion of a subgame
• The subgames of an extensive form game $G$ are the games induced by each decision node of $G$ (with strategies, etc, restricted appropriately)
• (Remember that $G$ is a subgame of itself.)
In this example, the game has just two sub-games.
A strategy profile $\vec{\sigma} = (\sigma_1, \ldots, \sigma_n)$ is a subgame perfect Nash equilibrium of a game $G$ if it is a Nash equilibrium in each subgame $G'$ of $G$.

- Observe that $(L, l)$ is not a SPNE, because $l$ is not a NE of the subgame induced by $A$'s decision node.
- However, $(R, r)$ is a SPNE.
Theorem

1. Every extensive form game has a SPNE
2. SPNE for extensive form games can be computed in polynomial time using Zermelo’s algorithm.
What does Zermelo say?
What does Zermelo say? SPNE says that first player moves $R$ and game ends immediately.
The Centipede Game
A Game With a Counterintuitive SPNE

- In practice, people manage to move $D$ for a few rounds before someone moves $R$, leaving them both better off
- A SPNE with poor social welfare
We can represent extensive form games as strategic form games, and solve them using techniques that we use for these.

Recall that a strategy profile $\bar{\sigma}$ in an extensive form game uniquely determines a leaf node, with payoffs for each player.

So, for each player $i$, define $u_i$

Let $\Sigma_i = \{\sigma_i^1, \ldots, \sigma_i^k\}$ be the pure strategies for player $i \in N$

This defines a strategic form game

Note that we get an **exponential blowup**: $\Sigma_i$ may be exponentially larger than the original game tree
Part XXV

Imperfect Information in Extensive Form Games
What do player’s know?

- In the extensive form game model we have looked at so far, all players have **perfect information** about the game. In particular, they know all the moves that have been made to date.
- This is often unrealistic!
- A variation of extensive form games allows us to capture **imperfect information**
Illustration: A Failed Attempt to Represent Matching Pennies

Suppose we try to capture matching pennies as an extensive form game...

- This doesn’t work, because when it comes to his move, A will know whether E has shown heads or tails! Consider:

\[ \sigma_A(v) = \begin{cases} 
  t & \text{if } v = v_1 \\
  h & \text{if } v = v_2 
\end{cases} \]
Modelling Imperfect Information

- Partition each players decision nodes into **information sets**
- Let $\mathcal{I}_i$ denote player $i$’s information sets
- If $v \in V_i$ then denote by $[v]$ the information set containing $v$ (note $v \in [v]$)
- Intuition:
  - if $[v] = [v']$ then the decision player does not know whether
    he is in $v$ or $v'$
  - she **cannot distinguish** these nodes
- We require that if $[v] = [v']$ then $A(v) = A(v')$
- A strategy in an imperfect information game is then a function that assigns an action to each information set

$$\sigma_i : \mathcal{I}_i \rightarrow A_i$$

(We are glossing over some technicalities here...)
Matching Pennies as an Imperfect Information Game

- Information sets indicated with a dotted line (but don’t draw singletons)

\[
\mathcal{I}_A = \{ \{v_1, v_2\} \} \quad \mathcal{I}_E = \{ \{v_0\} \}
\]

- Thus, when \( A \) makes his move, he doesn’t know whether \( E \) chose \( H \) or \( T \)
Part XXVI

Randomized Strategies in Extensive Form Games
• Recall that in strategic games, a mixed (randomized) strategy $ms_i$ for player $i$ is a probability distribution over player $i$’s pure strategies $\Sigma_i$.

• In extensive form games, we have two ways in which we can randomize:
  - mixed strategies
  - behavioural strategies
Mixed Strategies

- As in strategic form games, a mixed strategy in an extensive form game is a probability distribution over pure strategies i.e., a probability distribution.
- So, denote by $MS_i = \Delta \Sigma_i$ the set of mixed strategies for $i$. 
Behavioural Strategies

• An alternative formulation of randomized strategies has players randomizing at each decision node
• A behavioural strategy $\beta$ is then a function

$$\beta : V_i \rightarrow \Delta A_i$$

such that

$$\text{supp}(\beta_i(v)) \subseteq A(v)$$

• These are called behavioural strategies

• An obvious question:

  How are mixed and behavioural strategies related? Is one kind more “expressive” than the other?

• The answer is that, under certain conditions, they are equivalent.
Recall a player $i$ has **perfect recall** if she knows all her previous decisions.

**Theorem**

*In extensive form games with perfect recall*

1. for every *mixed strategy* there exists a *behavioural strategy* that yields the same probability distribution over outcomes

2. for every *behavioural strategy* there exists a *mixed strategy* that yields the same probability distribution over outcomes

Since perfect information implies perfect recall, the result holds for games of perfect information.
The Forgetful Driver

• An absent minded professor is driving home. It is foggy and hard to see much. The road has two exits, A, and B, which appear after each other.

• Exit A involves a long drive through poor country roads, yielding the driver a utility of 0.

• Exit B is the best: it goes home directly on good roads, yielding a utility of 4.

• If the driver does not exit at B, then he has to drive a fair distance to get home, but not so far as if he exited at A, yielding a utility of 1.

• However, the professor is absent minded, and when he reaches an exit, in the fog he cannot tell whether it is exit A or exit B.
The Forgetful Driver

- Exit straight
- Straight
- Exit
- 0
- 4
- 1
The Forgetful Driver

- Any pure strategy will yield payoff 0 or 1 (why?)
- Since mixed strategies randomize over pure strategies, any mixed strategy will either exit immediately or drive straight to the end.
- **The only chance to get payoff 4 is to randomize at decision nodes.**
Part XXVII

Iterated Games
Recall the Prisoner’s Dilemma... 
An Equilibrium with Undesirable Social Properties

<table>
<thead>
<tr>
<th></th>
<th>defect</th>
<th>coop</th>
</tr>
</thead>
<tbody>
<tr>
<td>defect</td>
<td>−2</td>
<td>−3</td>
</tr>
<tr>
<td>coop</td>
<td>0</td>
<td>−1</td>
</tr>
</tbody>
</table>

Mutual defection (2 years in jail each) is a **dominant strategy equilibrium**
The Iterated Prisoner’s Dilemma

• One answer: **play the game more than once**.
• If you know you will be meeting your opponent again, then perhaps the incentive to defect evaporates...?
• But... suppose you both know that you will play the game exactly $n$ times.

• What should you do? Imagine yourself playing the final round.

• On round $n$, you have an incentive to defect, to gain that extra bit of payoff. . .

• But this makes round $n - 1$ the last “real” round. . . but you have an incentive to defect there, too.

• This analysis technique is known as backwards induction.

**Theorem**

*Playing the iterated Prisoner’s Dilemma with a fixed, finite, pre-determined, commonly known number of rounds, mutual defection at every step is a dominant strategy equilibrium.*
Suppose you play the game an infinite number of rounds?

Two issues:

- How to measure utility over infinite plays?
  Summing utilities doesn’t work – sums to infinity.
- How to model strategies for infinite plays?
  Strategies are not just “C” or “D”
Utility functions for infinite runs

• Common approach: use a discount factor, $0 < \delta \leq 1$, to discount the value of future rounds – gives a finite value to infinite sum

• The value of the infinite run

$$\omega_0 \omega_1 \omega_2 \omega_3 \cdots \omega_k \cdots$$

to player $i$ is then

$$\sum_{k \in \mathbb{N}} \delta^k u_i(\omega_k)$$

• Alternative: compute average over all rounds. If players use automata strategies this is easy!
Strategies for infinite plays
Strategies as automata strategies

- We represent strategies as **finite automata** – technically, **Moore machines** (“transducers”)
- Here is an automaton strategy called “ALLD”, which always defects:

```
C

D
```

- Value inside a state is the action selected; outgoing arrows are actions of counterpart.
The ALLC strategy

Simply cooperates forever.
The GRIM strategy

I cooperate until you defect, at which point I flip to punishment mode: I defect forever after.
The TIT-FOR-TAT strategy

What does this strategy do?
Theorem

Finite machine strategies playing against each other will eventually enter a finite repeating sequence of outcomes.

They generate a run of the form

$$\alpha \cdot \beta^\omega$$

where $\alpha$ and $\beta$ are regular expressions and $\omega$ is the infinite iteration operator.

The average utility of an infinite run is then simply the average utility over just the finite sequence $\beta$. 

ALLC against ALLC

round: 0 1 2 3 4 ⋯
ALLC: C C C C C ⋯ average utility = −1
ALLC: C C C C C C ⋯ average utility = −1

This is **not** a NE: either player would do better to choose another strategy (e.g., ALLD)
ALLC against ALLD

round: 0 1 2 3 4 …
ALLC: C C C C C C … average utility = −3
ALLD: D D D D D D … average utility = 0

This is not a NE: ALLC would do better to choose another strategy (e.g., ALLD)
ALLD against ALLD

round: 0 1 2 3 4 ⋯

ALLD: D D D D D ⋯ average utility = −2

ALLD: D D D D D ⋯ average utility = −2

This is a NE (basically same as in one-shot case).
But it is not very desirable!
Notice that GRIM tries to cooperate but then goes into punishment mode: on average, it doesn’t do worse than if it had been ALLD. This is **not** a NE: ALLD can beneficially deviate, as next slide shows.
GRIM against GRIM

round: 0 1 2 3 4 ...  
GRIM: C C C C C C C ... average utility = \(-1\)  
GRIM: C C C C C C C ... average utility = \(-1\)

This **is** a NE! **Rationally sustained cooperation.**

The **threat of punishment keeps players in line.**
In a game $G$, let player $i$’s **security value** be the best utility that it can guarantee for itself, no matter what the other players do (i.e., even if they “gang up on it”).

**Theorem (Nash Folk Theorem)**

*In an infinitely repeated game, every outcome in which every player gets at least their security value can be sustained as a Nash equilibrium.*

*In the infinitely repeated Prisoner’s Dilemma, this means mutual cooperation can be sustained as an equilibrium.*

Proof: use GRIM strategies. If any player deviates from required profile, other players punish him, ensuring he gets his reservation value.
Consider the following stage game:

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>−3</td>
<td>0</td>
</tr>
<tr>
<td>M</td>
<td>−1</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>−2</td>
</tr>
</tbody>
</table>

What outcomes can be sustained as NE in the infinitely repeated game?

We first need to identify the security value for each player.
A Worked Example

Consider the following stage game:

\[
\begin{array}{c|cc}
 & L & R \\
\hline
T & -3 & 0 \\
 & -1 & 1 \\
M & -1 & 1 \\
 & 2 & 0 \\
B & 1 & -2 \\
\end{array}
\]

Define:

- \( \bar{u}_i(\sigma_j) = \max \{ u_i(\sigma_i, \sigma_j) \mid \sigma_i \in \Sigma_i \} \)
  - the largest utility that \( i \) could get if \( j \) plays \( \sigma_j \)
- \( u_i(\sigma_j) = \min \{ u_i(\sigma_i, \sigma_j) \mid \sigma_i \in \Sigma_i \} \)
  - the smallest utility that \( i \) could get if \( j \) plays \( \sigma_j \)
A Worked Example

So...

- $\bar{u}_1(L) = \max\{-1, 2, 1\} = 2$
- $\bar{u}_1(R) = \max\{1, 0, -2\} = 1$
- $\bar{u}_2(T) = \max\{-3, 0\} = 0$
- $\bar{u}_2(M) = \max\{-1, 1\} = 1$
- $\bar{u}_2(B) = \max\{1, 1\} = 1$
• Player 2’s punishment strategy against 1 would be to choose a strategy $\sigma_j$ that minimizes $\bar{u}_1(\sigma_j)$, i.e., $R$
• Player 1’s punishment strategy against 2 would be to choose a strategy $\sigma_j$ that minimizes $\bar{u}_2(\sigma_j)$, i.e., $T$

So the reservation values are $(1, 0)$: every outcome in which the respective players get at least these values can be sustained as an NE.
The outcomes are: (T,R), (B,L).
A Worked Example

Sanity check:

Draw finite state machine strategies for both players to sustain (T,R) as an NE.
Part XXVIII

Iterated Boolean Games
Iterated Boolean Games (iBG)

- A model of multi-agent systems in which players repeatedly choose truth values for Boolean variables under their control.
- Players behave selfishly in order to achieve individual goals.
- Goals expressed as **Linear Temporal Logic** (LTL) formulae.
Propositional Linear Temporal Logic (LTL)

A standard language for talking about \textbf{infinite state sequences}.

\begin{align*}
\top & \quad \text{truth constant} \\
p & \quad \text{primitive propositions } (\in \Phi) \\
\neg \varphi & \quad \text{classical negation} \\
\varphi \lor \psi & \quad \text{classical disjunction} \\
X \varphi & \quad \text{in the next state} \ldots \\
F \varphi & \quad \text{will eventually be the case that } \varphi \\
G \varphi & \quad \text{is always the case that } \varphi \\
\varphi U \psi & \quad \varphi \text{ until } \psi
\end{align*}
Example LTL formulae

\[ F \rightarrow jetlag \]

eventually I will not have jetlag (a \textit{liveness} property)
Example LTL formulae

\( G \neg \text{crash} \)

the plane will never crash (a \textit{safety} property)
Example LTL formulae

\[ \mathsf{G} \mathsf{F} \text{drinkBeer} \]
Example LTL formulae

\[ G F \text{drink} \text{Beer} \]

I will drink beer \textit{infinitely often}
Example LTL formulae

\[ F \, G \, \text{dead} \]
Example LTL formulae

\( F \bigwedge G \text{dead} \)

Eventually will come a time at which I am dead forever after.
Example LTL formulae

$$(\neg \text{friends}) \mathbf{U} \text{youApologise}$$
Example LTL formulae

\((\neg \text{friends}) \mathbf{U} \text{youApologise}\)

we are not friends \textbf{until} you apologise
An iBG is a structure

\[ G = (N, \Phi, \Phi_1, \ldots, \Phi_n, \gamma_1, \ldots, \gamma_n) \]

where

- \( N = \{1, \ldots, n\} \) is a set of **agents** (the players of the game),
- \( \Phi = \{p, q, \ldots\} \) is a finite set of **Boolean variables**,  
- \( \Phi_i \subseteq \Phi \) is the set of variables controlled by player \( i \),
- \( \gamma_i \) is the **LTL goal** of player \( i \).
• Let $V$ be the set of **valuations** of Boolean variables $\Phi$.
• Let $V_i$ be the valuations for the variables $\Phi_i$ controlled by player $i$.
• Models of LTL formulae $\varphi$ are **runs** $\rho$: infinite sequences in $V^\omega$.
• We write $\rho \models \varphi$ to mean $\rho$ satisfies LTL formula $\varphi$. 
Playing an iBG

• Players play an infinite number of rounds, where on each round each player chooses values for their variables.

• The sequence of valuations traced out in this way forms a run, which either satisfies or doesn’t satisfy a player’s goal.

• A **strategy** for \( i \) is thus abstractly a function

\[
f : V^* \rightarrow V_i
\]

…but this isn’t a **practicable** representation.

• So we model strategies as **finite state machines (FSM)** with **output** (transducers).
A machine strategy for $i$ is a structure:

$$\sigma_i = (Q_i, q_i^0, \delta_i, \tau_i)$$

where:

- $Q_i$ is a finite, non-empty set of states,
- $q_i^0$ is the initial state,
- $\delta_i : Q_i \times V \rightarrow Q_i$ is a state transition function,
- $\tau_i : Q_i \rightarrow V_i$ is a choice function.
A strategy profile $\vec{\sigma}$ is an $n$-tuple of machine strategies, one for each player $i$:

$$\vec{\sigma} = (\sigma_1, \ldots, \sigma_n).$$

As strategies are deterministic, each strategy profile $\vec{\sigma}$ induces a unique run: $\rho(\vec{\sigma})$. 
Strategy profile $\vec{\sigma} = (\sigma_1, \ldots, \sigma_i, \ldots, \sigma_n)$ is a (pure strategy) **Nash equilibrium** if for all players $i \in N$, if $\rho(\vec{\sigma}) \not\models \gamma_i$ then for all $\sigma'_i$ we have

$$\rho(\sigma_1, \ldots, \sigma'_i, \ldots, \sigma_n) \not\models \gamma_i$$

Let $NE(G)$ denote the Nash equilibria of a given iBG $G$. 


An Example

• \( N = \{1, 2\} \),
• \( \Phi_1 = \{p\} \)
• \( \Phi_2 = \{q\} \)
• \( \gamma_1 = GF(p \leftrightarrow q) \)
• \( \gamma_2 = GF\neg(p \leftrightarrow q) \)

These strategies form a NE.
**MODEL CHECKING:**

*Given*: Game $G$, strategy profile $\overline{\sigma}$, and LTL formula $\varphi$.
*Question*: Is it the case that $\rho(\overline{\sigma}) \models \varphi$?

**MEMBERSHIP:**

*Given*: Game $G$, strategy profile $\overline{\sigma}$.
*Question*: Is it the case that $\overline{\sigma} \in \text{NE}(G)$?
**Model Checking:**

**Given:** Game $G$, strategy profile $\bar{\sigma}$, and LTL formula $\varphi$.

**Question:** Is it the case that $\rho(\bar{\sigma}) \models \varphi$?

**Membership:**

**Given:** Game $G$, strategy profile $\bar{\sigma}$.

**Question:** Is it the case that $\bar{\sigma} \in NE(G)$?

**Theorem**

*The Model Checking and Membership problems are PSPACE-complete.*

Proof: follow from the fact that we can encode FSM strategies as LTL formulae.
Decision problems

**E-NASH:**
*Given:* Game $G$, LTL formula $\varphi$.
*Question:* $\exists \vec{\sigma} \in NE(G). \rho(\vec{\sigma}) \models \varphi$?

**A-NASH:**
*Given:* Game $G$, LTL formula $\varphi$.
*Question:* $\forall \vec{\sigma} \in NE(G). \rho(\vec{\sigma}) \models \varphi$?

**NON-EMPTINESS:**
*Given:* Game $G$.
*Question:* Is it the case that $NE(G) \neq \emptyset$?
Decision problems

**E-NASH:**
*Given:* Game $G$, LTL formula $\varphi$.
*Question:* $\exists \vec{\sigma} \in NE(G). \rho(\vec{\sigma}) \models \varphi$?

**A-NASH:**
*Given:* Game $G$, LTL formula $\varphi$.
*Question:* $\forall \vec{\sigma} \in NE(G). \rho(\vec{\sigma}) \models \varphi$?

**NON-EMPTINESS:**
*Given:* Game $G$.
*Question:* Is it the case that $NE(G) \neq \emptyset$?

**Theorem**

*The E-NASH, A-NASH, and NON-EMPTINESS problems are 2EXPTIME-complete.*

Proof: we can reduce **LTL synthesis** (Pnueli & Rosner, 1989)
For iBGs, the Folk (Nash) Theorems for iBG answer the question:

*Which LTL properties are satisfied in the Nash equilibria of a given iterated Boolean game?*

In other words, which LTL formulae will be true if everyone acts rationally?
Punishable players and safety goals

- Player $i$ is **punishable** if (at any point of time) $i$’s opponents can jointly find values for the propositional variables under their control that guarantee $\gamma_i$ to be false no matter which values $i$ chooses for its variables.
- Say $\gamma_i$ is a **safety** goal if $\gamma_i \equiv G \varphi_i$ for some LTL formula $\varphi_i$. 
Theorem

If all players have propositional safety goals and are punishable, then every satisfiable LTL formula is satisfied in some equilibrium.

Formally, let $G = (N, \Phi, \Phi_1, \ldots, \Phi_n, \gamma_1, \ldots, \gamma_n)$ be an iBG in which each player $i$ has a propositional safety goal $\gamma_i$. Then, the following two statements are equivalent:

- all players are punishable
- for all satisfiable $\psi$, there is a $\vec{\sigma} \in \text{NE}(G)$: $\rho(\vec{\sigma}) |\models \psi$. 
Second folk theorem

**Theorem**

We can obtain $\psi$ as an equilibrium in any game as long as $\psi$ implies all non-punishable players get their goal achieved.

Let $G = (N, \Phi, \Phi_1, \ldots, \Phi_n, \gamma_1, \ldots, \gamma_n)$ be an iBG in which each player $i$ has a propositional safety goal $\gamma_i$. Then, for all $\psi$ such that

$$\psi \land \bigwedge \{ \gamma_i \mid \gamma_i \text{ is not punishable} \}$$

is satisfiable, there is $\bar{\sigma} \in \text{NE}(G)$ such that $\rho(\bar{\sigma}) \models \psi$. 


Third folk theorem

**Theorem**

Let \( G = (N, \Phi, \Phi_1, \ldots, \Phi_n, \gamma_1, \ldots, \gamma_n) \) be an iBG where each player \( i \) has a safety goal \( \gamma_i \). Then, if all players are punishable, for all satisfiable \( \psi \), there is Nash equilibrium \( \vec{\sigma} \in NE(G) \) such that \( \rho(\vec{\sigma}) \models \psi \).
The strategy you really want to play in the prisoner’s dilemma is:

I’ll cooperate if he will.

Program equilibria provide one way of enabling this.

Each agent submits a program strategy to a mediator which jointly executes the strategies. Crucially, strategies can be conditioned on the strategies of the others.

---

Consider the following program:

```
IF HisProgram == ThisProgram THEN
  DO(C);
ELSE
  DO(D);
END-IF.
```

“==” is **string comparison**: comparing program texts.

(Compare this with GRIM in iterated games.)

The best response to this program is to **submit the same program**, giving an outcome of \((C, C)\)!

This is a **program equilibrium**.
Theorem (Tennenholtz)

In any one shot game, every outcome in which every player gets at least their reservation value can be obtained as the outcome of a program equilibrium.

For the Prisoner’s Dilemma, this means mutual cooperation can be obtained as the outcome of a program equilibrium.
Part XXIX

Evolutionary Games
• Suppose you play iterated prisoner’s dilemma against a range of opponents . . .
  What strategy should you choose, so as to maximise your overall payoff?

• Axelrod (1984) investigated this problem, with a computer tournament for programs playing the prisoner’s dilemma\(^9\).

Some strategies from Axelrod’s Tournament

- **ALLD:**
  “Always defect” — the **hawk** strategy;

- **TIT-FOR-TAT:**
  1. On round $u = 0$, cooperate.
  2. On round $u > 0$, do what your opponent did on round $u - 1$.

- **TESTER:**
  On 1st round, defect. If the opponent retaliated, then play TIT-FOR-TAT. Otherwise intersperse cooperation & defection.

- **JOSS:**
  As TIT-FOR-TAT, except periodically defect.

Of the 63 strategies entered, he found TIT-FOR-TAT did best.
Why did TIT-FOR-TAT do well?

Perhaps surprising that TIT-FOR-TAT do so well...

Proposition

In all 2 player finitely repeated prisoner’s dilemma games, TIT-FOR-TAT does no better (and possibly worse) than all other strategies: in a one-to-one competition, it does no better than any possible strategy.

So what is the explanation?
Axelrod suggests the following rules for succeeding in his tournament:

- **Don’t be envious:**
  Don’t play as if it were zero sum!

- **Be nice:**
  Start by cooperating, and reciprocate cooperation.

- **Retaliate appropriately:**
  Always punish defection immediately, but use “measured” force — don’t overdo it.

- **Don’t hold grudges:**
  Always reciprocate cooperation immediately.

This is not mathematically robust advice – somewhat controversial amongst game theorists.
So, why does TIT-FOR-TAT do so well?

- If TIT-FOR-TAT was in a population of ALLD, it would suffer.
- **But it isn’t.** It is in a population that contains **cooperative** agents.
- TIT-FOR-TAT does well because it gets to play against **other cooperative strategies**: the “strategy population” consisted of other cooperative strategies.
- When cooperative strategies meet, they can **share** the benefits of mutual cooperation, while strategies that immediately defect can get bogged down in conflict.
Axelrod’s evolutionary tournament

- Axelrod then suggested interpreting performance in his tournament as a measure of **evolutionary fitness**, and repeated the tournament over hundreds of generations.
- Strategies with higher relative fitness **increased their presence in the strategy population** compared to others.
- Notice that how well a strategy does **depends on what other strategies are present in the population**.
- Just assuming evolutionary forces, what will a population of strategies evolve to?
- Again, TIT-FOR-TAT did very well.
“The first thing that happens is that the lowest-ranking eleven entries fall to half their initial size by the fifth generation while the middle-ranking entries tend to hold their own and the top-ranking entries gradually grow in size. By the fiftieth generation, the [strategies] that ranked in the bottom third of the tournament have virtually disappeared, while most of those the middle third have started to shrink, and those in the top third are continuing to grow. The process simulates survival of the fittest. A [strategy] that is successful on average with the current distribution of [strategies] in the population will become an even larger proportion of the environment ... in the next generation. At first, a rule that is successful with all sorts of rules will proliferate, but later as the unsuccessful rules disappear, success requires success with other successful rules.”

(Axelrod 1984)
For Axelrod, the exciting thing was that TIT-FOR-TAT, and mutually sustained cooperation, could arise merely through blind evolutionary processes: cooperation through evolution.

There is no “thinking” about what strategy to choose

Strategies are chosen through natural selection

In evolutionary game theory, we have:

\[ \text{evolutionary fitness} = \text{utility} \]

The Hawk-Dove Game

- Suppose have a very large population of individuals, which come in two variants: **Hawks** and **Doves**
- These variants play role of strategies in conventional game theory.
- Individual don’t “decide” whether to be a Hawk or a Dove: variants are **genetically hardwired**
- Individuals reproduce over time, but reproduction is **asexual**: individual doesn’t need a partner to reproduce, and if an individual reproduces, it begets offspring of the same type.
- The key attribute of an individual that determines how likely they are to reproduce is a numeric value that we’ll call their **fitness**, which measures **how likely that individual is to be able to reproduce and pass on their genes**
In the Hawk-Dove Game, individuals increase their fitness by obtaining a particular resource (e.g., food) from the environment. Individuals are in competition with others to obtain resources. Hawks are fierce; Doves are timid.

1. When a Hawk competes with a Dove, the Hawk takes the whole of the resource.
2. When a Dove competes with a Dove, they share the resource equally.
3. When a Hawk competes with a Hawk, they fight, and have an equal chance of obtaining the resource or being injured.
The Hawk-Dove Game

Let... 

- $V$ denote the value of the resource
  This is the **increase in fitness** that an individual would gain by obtaining the resource.

- $C$ denote the **cost of injury**
  This is the amount by which fitness would **decrease** if an individual fought for the resource and lost.
Rules of the Hawk-Dove Game

1. When a Hawk meets a Hawk: they fight, and have an equal chance of increasing their fitness by $V$ or decreasing their fitness by $C$; on average, this will result in an increase of fitness by $(V - C)/2$.

2. When a Dove meets a Dove, they share the resource equally, each obtaining an increase of fitness of $V/2$.

3. When a Hawk meets a Dove, the Hawk takes the whole of the resource, giving $V$ to the Hawk, while the Dove gets no benefit.

Can you write down the payoff matrix?
Fitness

- Let $p$ denote **proportion of Hawks** in population.
- Let $W(H)$ and $W(D)$ denote **average fitness** of Hawks and Doves:

\[
W(H) = p((V - C)/2) + (1 - p) V \\
W(D) = (1 - p)(V/2)
\]

- The **expected utility** of playing the Hawk-Dove game.
• Let $A$ denote the **average fitness of the population**:

$$A = pW(H) + (1 - p)W(D)$$

i.e., the expected fitness of an individual drawn uniformly at random from the population

• The new frequency $p'$ of Hawks in the next generation is then:

$$p' = p \frac{W(H)}{A}$$

• Thus if $W(H) > A$ then Hawks will **increase** in next generation, those whose fitness is **less** will decrease
Replicator Dynamics in the Hawk-Dove Game
Evolutionarily stable strategies (ESS)

Formally, strategy $\sigma$ is an ESS iff:

1. It is a best response to itself. (Otherwise, other strategies could “prey” on it.)

2. For any strategy $\sigma'$ that does as well against $\sigma$ as $\sigma$ does, $\sigma$ does better against $\sigma'$ than $\sigma'$ does against itself. (So other strategies can’t benefit against $\sigma$ by playing against themselves.)

(TIT-FOR-TAT is not, in fact, an ESS.)
Lecture 6: Zero Sum Games
• Recall that zero sum games are games in which for every outcome \( \omega \in \Omega \) we have

\[
\sum_{i \in N} u_i(\omega) = 0
\]

• Zero sum games are strictly competitive: best outcome for me is worst for you, and vice versa.

• The symmetry of preferences in zero sum games means we can use different (simpler!) techniques to analyse them.

\( \Rightarrow \) Zero sum games are different! \( \Leftarrow \)

• Zero sum games also play an important role in complexity analysis
Part XXX

Normal Form Zero Sum Games
How should you play a zero sum game?

(We only list utilities of row player in zero sum payoff matrices.)

Imagine you are the row player and you make a choice first; column player then gets to respond.
How should you play a zero sum game?

(We only list utilities of row player in zero sum payoff matrices.)

Imagine you are the row player and you make a choice first; column player then gets to respond. You know that whichever row you choose, column player will pick your smallest utility in that row.
How should you play a zero sum game?

(We only list utilities of row player in zero sum payoff matrices.)

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>−2</td>
</tr>
</tbody>
</table>

Imagine you are the row player and you make a choice first; column player then gets to respond. You know that whichever row you choose, column player will pick your smallest utility in that row. So you should choose to maximise that minimum.
### The Maximin Value of a Zero Sum Game

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>min = 2</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
<td>min = 5.6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
<td>min = 1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>−2</td>
<td>min = −2</td>
</tr>
</tbody>
</table>

Take the minimum of each row.
The Maximin Value of a Zero Sum Game

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
<td>5.6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>−2</td>
<td>−2</td>
</tr>
</tbody>
</table>

Take the minimum of each row.
The **maximin value**, \( \bar{v} \), is then the maximum of these.
Take the minimum of each row.

The maximin value, $\bar{\nu}$, is then the maximum of these.

$$
\bar{\nu} = \max_{\sigma_1 \in \Sigma_1} \min_{\sigma_2 \in \Sigma_2} u_1(\sigma_1, \sigma_2)
$$
Safety Strategies
Life is Not a Zero-Sum Game

- Strategies that yield maximin outcomes are sometimes called **safety strategies**
- The maximin value for a player is sometimes called the **safety level** or **reservation value**
- This is “worst case scenario” thinking
- Is it ever really justified?
  - zero-sum games are rare
  - maybe makes sense against **irrational** players...?
- Unfortunately, in practice, people often interact as though in a zero-sum game, and miss out on benefits of common interest
The Minimax Value of a Zero Sum Game

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>min = 2</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
<td>min = 5.6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
<td>min = 1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>−2</td>
<td>min = −2</td>
</tr>
</tbody>
</table>

Take the maximum of each column.

max = 10  max = 5.6  max = 13
The Minimax Value of a Zero Sum Game

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>min = 2</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
<td>min = 5.6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
<td>min = 1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>-2</td>
<td>min = -2</td>
</tr>
</tbody>
</table>

Take the maximum of each column.
The **minimax value**, $v$, is then the minimum of these.
The Minimax Value of a Zero Sum Game

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>5.6</td>
<td>10.5</td>
<td>5.6</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>4.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3</td>
<td>-2</td>
<td>-2</td>
</tr>
</tbody>
</table>

Take the maximum of each column.
The minimax value, $v$, is then the minimum of these.

$$v = \min_{\sigma_2 \in \Sigma_2} \max_{\sigma_1 \in \Sigma_1} u_1(\sigma_1, \sigma_2)$$
The Minimax Theorem (Pure Strategies)

Theorem (John von Neumann, 1928.)

Suppose we have a two player zero-sum game, in which $(\sigma_1, \sigma_2)$ is a Nash equilibrium.

$$u_1(\sigma_1, \sigma_2) = \bar{v} = v$$

Thus, in zero-sum games, Nash equilibria and maximin/minimax outcomes coincide.

The maximin value for player 1 is called the value of the game.

(Player 2 gets $-\bar{v}$.)

(Von Neumann proved it for mixed strategies; a much more complex and significant result.)
The symmetry in two player zero-sum games means that computing mixed NE is an optimization problem.

This optimization problem can be solved via linear programming.
Minimax Linear Program for Player 2

minimize $U_1^*$ subject to:

$$\sum_{\sigma_2^k \in \Sigma_2} u_1(\sigma_1, \sigma_2^k) \cdot p_2^k \leq U_1^*$$

for all $\sigma_1 \in \Sigma_1$

$$\sum_{\sigma_2^k \in \Sigma_2} p_2^k = 1$$

$$p_2^k \geq 0$$

for all $\sigma_2^k \in \Sigma_2$

Here the unknowns are $U_1^*, p_2^1, \ldots, p_2^l$

Values $p_2^k$ give probability of player 2 choosing $\sigma_2^k$
Maximin Linear Program for Player 1

maximize $U_1^*$ subject to:

$$\sum_{\sigma^j_1 \in \Sigma_1} u_1(\sigma^j_1, \sigma_2) \cdot p^j_1 \geq U_1^* \quad \text{for all } \sigma_2 \in \Sigma_2$$

$$\sum_{\sigma^j_1 \in \Sigma_1} p^j_1 = 1$$

$$p^j_1 \geq 0 \quad \text{for all } \sigma^j_1 \in \Sigma_1$$

Values $p^j_1$ give probability of player 1 choosing $\sigma^j_1$
Part XXXI

Extensive Form Zero Sum Games
We now focus on 2 player games, with players $E$ and $A$.

We are interested in two-player zero-sum games (i.e., strictly competitive games).

Assume leaf nodes are labelled with either 1 or $-1$, indicating payoff for player $E$:
- payoff = 1 means “player $E$ wins”
- payoff = $-1$ means “player $E$ loses”
Determinacy

- Key concept in win-lose games is whether games are determined: whether some player can force a win.
- You forcing a win means that you have a strategy such that all outcomes possible by playing that are strategy result in a win for you:

  \[
  \text{there exists a choice for you}
  \]

  \[
  \text{such that for all choices of your counterpart}
  \]

  \[
  \text{there exists a choice for you}
  \]

  \[
  \text{such that for all choices of your counterpart}
  \]

  \[
  \ldots
  \]

  \[
  \text{you win.}
  \]

- Hence we talk of **winning strategies** for players.
As usual in zero-sum games, we only list payoffs for one player, in this case the blue player.
Zermelo’s algorithm works fine for such games: player $E$ can force a win ($=\text{payoff 1}$) by choosing $L$. 
In extensive form win-lose games, Zermelo’s algorithm tells us which player can force a win; the strategies computed by Zermelo’s algorithm are the optimal strategies for all players, and in particular, give us a winning strategy for the relevant player.

As a corollary, (finite) extensive form win-lose games are determined: one of the players can force a win.
Theorem

In extensive form win-lose games, Zermelo’s algorithm tells us which player can force a win; the strategies computed by Zermelo’s algorithm are the optimal strategies for all players, and in particular, give us a winning strategy for the relevant player.

As a corollary, (finite) extensive form win-lose games are determined: one of the players can force a win.

Theorem

Determining whether a given player has a winning strategy in a finite win-lose extensive form game is P-complete.
Two-player extensive form win-lose games have a particularly important role in computer science, although game-theoretically, they are quite simple.

Many computer science decision problems can be formulated as extensive-form win-lose games.

Typical formulation:

\[ x \text{ is a positive instance of the decision problem } \Pi \text{ iff player 1 can force a win in the game } G_{(\Pi,x)} . \]

Different types of extensive form win-lose games neatly characterise various complexity classes (P, PSPACE, EXPTIME)
Observe that there is a close correspondence between the game...

\[ (v_3 \land v_4) \lor (v_5 \land v_6) \]

Letting true = 1 and false = −1, the formula is true under a given valuation iff player \( E \) has a winning strategy in the game.
Since checking whether a player has a winning strategy in a two-person finite extensive form zero sum can be formulated as evaluating propositional formulae, we have:

**Theorem**

The problem of determining whether a given propositional formula is true under a given valuation is P-complete.

... it can be solved in polynomial time, but is as hard as any problem that can be solved in polynomial time.
• Zermelo’s magic algorithm finds winning strategies in polynomial time – yay for Zermelo!
• However, this assumes that the entire game tree is given as an input: running time is polynomial in the size of the tree.
• In many cases, we look for compact representations of game trees.
• Where the game tree is represented in a compact way, we might expect the complexity to increase... and it does.
Compactly Represented Games

We consider three types of compactly represented games:

1. games guaranteed to end after a “small” number of moves
2. games that can go on for exponentially many rounds
3. games that can go on for a long time and require memory.
The formula game is played by two players, and is defined by variables

\[ \vec{x} = x_1, \ldots, x_k \]
\[ \vec{y} = y_1, \ldots, y_k \]

and propositional logic formula

\[ \varphi(\vec{x}, \vec{y}). \]

Player one picks a value (\( \top \) or \( \bot \)) for \( x_1 \), then 2 picks a value for \( y_1 \), and so on, until all variables have a value.

Player 1 wins if \( \varphi \) is made true under the valuation they define in this way.
• Observe that the formula game is a **compactly specified extensive form game**, with a small number of moves ($k$ moves for each player).

• We can solve it by generating the corresponding game tree and applying Zermelo . . .

• . . . but the game tree will be of size exponential in the number of variables.
Notice that game is a win for player 1 iff:

there exists a value for $x_1$
such that for all values of $y_1$
there exists a value for $x_2$
such that for values of $y_2$
\ldots

$\varphi$ is true under the resulting valuation.

But this is just the following **Quantified Boolean Formula**:

$$\exists x_1 \, \forall y_1 \, \exists x_2 \, \forall y_2 \, \ldots \, \varphi(\bar{x}, \bar{y}).$$

**Theorem**

Checking whether a given player can force a win in the formula game is PSPACE-complete.
The Game of Geography

- Start by naming a city.
- Players alternate to name another city that starts with the last letter of the previous city name.
- Not allowed to name same city twice.
- Player loses if can’t move.
• Compactly specified extensive form win-lose games that have a small number of moves tend to be PSPACE-complete.
• … and Geography is no exception

**Theorem**

*Checking whether a given player can force a win in the game of Geography is PSPACE-complete.*
Games with Exponentially Many Moves

The Game \textsc{PEEK-}\text{-}\textit{G}_4

An instance of \textsc{PEEK-}\text{-}\textit{G}_4 is a quad:

\[
\langle X_1, X_2, X_3, \varphi \rangle
\]

where:

- \(X_1\) and \(X_2\) are disjoint, finite sets of Boolean variables, with the intended interpretation that the variables in \(X_1\) are under the control of agent 1, and \(X_2\) are under the control of agent 2;
- \(X_3 \subseteq (X_1 \cup X_2)\) are the variables deemed to be true in the initial state of the game; and
- \(\varphi\) is a propositional logic formula over the variables \(X_1 \cup X_2\), representing the winning condition.

Game starts from the initial assignment \(X_3\)

Players alternate (1 moving first) to select a value for one of their variables.

Player wins if it makes \(\varphi\) true.
• If you can force a win, you can do so in at most exponential number of moves (why?)
• This is just a (big) extensive form game!

**Theorem**

*Checking whether a given player has a winning strategy in an instance of $\text{PEEK-} G_4$ is EXPTIME-complete.*

Can you give an algorithm that solves in EXPTIME?
Consider an instance of $\text{PEEK-}G_4$ where you are not allowed to revisit the same valuation of variables.

**Theorem**

$\text{PEEK-}G_4$ without repetitions is EXPSPACE-complete.