**Time:** Wednesday, Thursday: 10–12

**Place:** Am Regenbogen

**Website**

http://www.in.tu-clausthal.de/abteilungen/cig/cigroot/teaching

Visit regularly!

There you will find important information about the lecture, documents, labs et cetera.

**Lecture:** Prof. Dix, Dr. N. Bulling

**Exam:** tba
About this lecture (1)

This MSc course is about Decision Making in Multi Agent Systems. We look at decision making mainly from a game-theoretical perspective. The lecture can be roughly divided into three parts:

- **Classical game theory**: We take the perspective of a single agent participating in a game. How should she behave? Complete/incomplete information games (Chapters 1 and 5), repeated games (Chapter 2), mechanism design (Chapter 6), cooperative (or coalitional) game theory (Chapter 3).

- **Voting and auctions**: Here we take the perspective of a designer (or principal). Agents will follow the rules that she defines (but will act selfishly). Social choice, ranking systems, various types of auctions (Chapter 6).
Formalizing solution concepts in ATLP: We define an extension of the well-known logic ATL, namely ATLP, that allows us to formalize various solution concepts within this logic. This makes it possible to reason about it and do model checking (Chapter 8).

Temporal logics LTL, CTL: In order to better understand the logic ATL, we briefly introduce linear temporal logic (LTL), and branching time logic (CTL) (Chapter 7).

My thanks go to Dr. Nils Bulling who helped to transform my former MAS course into this new and advanced format.
Main References (1)

*Chapter 13: Programming Multi Agent Systems.*

*Reasoning about Temporal Properties of Rational Play,*

Model checking logics of strategic ability: Complexity.

*Chapter 14: Verifying Multi Agent Systems.*
In G. Weiß (Editor), *MultiAgent Systems*, pages 641–693. MIT Press.

*Reasoning about Interaction: From Game Theory to Logic and Back.*
Dagstuhl Reports, 1(3)*1–18.
http://dx.doi.org/10.4230/DagRep.1.3.1
Main References (2)


Lecture Overview

**Classical Game Theory:** Complete/incomplete information games, repeated/coalitional games, mechanism design.

Chapters 1–3, 5–6: (5+2+3+1+1=) 12 lectures.

**Voting and auctions:** Social choice and auctions.

Chapter 4: 5 lectures.

**LTL and CTL:** Temporal logics to motivate the logic ATL.

Chapter 7: 2 lectures.

**ATLP:** An extension of ATL, a logic handling strategies, to express game-theoretical solution concepts.

Chapter 8: 3 lectures.

**Exercises:** 6 exercise classes (roughly fortnightly).
Outline

1. Complete Information Games
2. Repeated Games
3. Coalitional Games
4. Social Choice and Auctions
5. Incomplete Information Games
6. Mechanism Design
7. From Classical to Temporal Logics
8. Strategic Logics
1. Complete Information Games

- Examples and Terminology
- Normal Form Games
- Extensive Form Games
- An Example from Economics
Outline (1)

We illustrate the difference between classical AI and MAS. We present several **evaluation criteria** for comparing protocols.

We then introduce the formal machinery of game theory assuming we have **complete information**:

- **normal form (NF) games**, where players play simultaneously,
- **extensive form (tree form) games**, where players play one after another. Here the history plays a role and players come up with strategies depending on the past.

We also distinguish between **perfect** and **imperfect recall** and discuss various notions of equilibria.

Finally we consider the existence of equilibria for **market mechanisms**.
**Classical DAI:** System Designer fixes Interaction-Protocol which is uniform for all agents. The designer also fixes a strategy for each agent.

**Outcome**

What is the outcome, assuming that the protocol is followed and the agents follow the strategies?
**MAI:** Interaction-Protocol is given. Each agent determines its own strategy (maximising its own good, via a utility function, without looking at the global task).

### Global optimum

What is the outcome, given a **protocol** that guarantees that each agent’s desired local strategy is the best one (and is therefore chosen by the agent)?
1.1 Examples and Terminology
We need to **compare protocols**. Each such protocol leads to a solution. So we determine how good these solutions are.

**Social Welfare**: Sum of all utilities

**Pareto Efficiency**: A solution \( x \) is Pareto-optimal, if

<table>
<thead>
<tr>
<th>there is no solution ( x' ) with:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) ( \exists ) agent ( ag : ut_{ag}(x') &gt; ut_{ag}(x) )</td>
</tr>
<tr>
<td>(2) ( \forall ) agents ( ag' : ut_{ag'}(x') \geq ut_{ag'}(x) ).</td>
</tr>
</tbody>
</table>

**Individual rational**: The payoff should be higher than not participating at all.
Stability:

Case 1: Strategy of an agent depends on the others.
The profile \( s^*_A = \langle s^*_1, s^*_2, \ldots, s^*|A| \rangle \) is called a Nash-equilibrium, iff \( \forall i : s^*_i \) is the best strategy for agent \( i \) if all the others choose
\[
\langle s^*_1, s^*_2, \ldots, s^*_{i-1}, s^*_{i+1}, \ldots, s^*_{|A|} \rangle.
\]

Case 2: Strategy of an agent does not depend on the others.
Such strategies are called dominant.
Example 1.1 (Prisoners Dilemma, Type 1)

Two prisoners are suspected of a crime (which they both committed). They can choose to (1) cooperate with each other (not confessing to the crime) or (2) defect (giving evidence that the other was involved). Both cooperating (not confessing) gives them a shorter prison term than both defecting. But if only one of them defects (the betrayer), the other gets maximal prison term. The betrayer then has maximal payoff.

<table>
<thead>
<tr>
<th>Prisoner 1</th>
<th>Prisoner 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cooperate</td>
<td>cooperate</td>
<td>(3, 3)</td>
</tr>
<tr>
<td>defect</td>
<td>defect</td>
<td>(5, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0, 5)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1, 1)</td>
</tr>
</tbody>
</table>
- **Social Welfare**: Both cooperate,
- **Pareto-Efficiency**: All are Pareto optimal, except when both defect.
- **Dominant Strategy**: Both defect.
- **Nash Equilibrium**: Both defect.
Prisoners dilemma revisited: $c \not\geq a \not\geq d \not\geq b$

<table>
<thead>
<tr>
<th>Prisoner 1</th>
<th>Prisoner 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>cooperate</strong></td>
<td>cooperate</td>
</tr>
<tr>
<td><strong>defect</strong></td>
<td>(a,a)</td>
</tr>
<tr>
<td></td>
<td>(b,c)</td>
</tr>
</tbody>
</table>
**Example 1.2 (Trivial mixed-motive, Type 0)**

<table>
<thead>
<tr>
<th>Player 1</th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C</td>
</tr>
<tr>
<td>C</td>
<td>(4, 4)</td>
</tr>
<tr>
<td>D</td>
<td>(3, 2)</td>
</tr>
</tbody>
</table>
Example 1.3 (Battle of the Bismarck Sea)

In 1943 the northern half of New Guinea was controlled by the Japanese, the southern half by the allies. The Japanese wanted to reinforce their troops. This could happen using two different routes: (1) north (rain and bad visibility) or (2) south (weather ok). Trip should take 3 days.

The allies want to bomb the convoy as long as possible. If they search north, they can bomb 2 days (independently of the route taken by the Japanese). If they go south, they can bomb 3 days if the Japanese go south too, and only 1 day, if the Japanese go north.
### 1 Complete Information Games

#### 1.1 Examples and Terminology

<table>
<thead>
<tr>
<th></th>
<th>Japanese</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Allies</td>
<td>Sail North</td>
<td>Sail South</td>
</tr>
<tr>
<td></td>
<td>Search North</td>
<td>2 days</td>
</tr>
<tr>
<td></td>
<td>Search South</td>
<td>1 day</td>
</tr>
</tbody>
</table>

**Allies:** What is the largest of all row minima?  
**Japanese:** What is smallest of the column maxima?

**Battle of the Bismarck sea:**  
largest row minimum = smallest column maximum.

This is called a **saddle point**.
1.2 Normal Form Games
Definition 1.4 \((n\)-Person Normal Form Game\)

A finite \(n\)-person normal form game is a tuple \(\langle \mathcal{A}, \text{Act}, O, \varrho, \mu \rangle\), where

- \(\mathcal{A} = \{1, \ldots, i, \ldots, n\}\) is a finite set of players or agents.
- \(\text{Act} = A_1 \times \cdots \times A_i \times \cdots \times A_n\) where \(A_i\) is the finite set of actions available to player \(i\). \(\vec{a} \in \text{Act}\) is called an action profile. Elements of \(A_i\) are called pure strategies.
- \(O\) is the set of outcomes.
- \(\varrho : \text{Act} \rightarrow O\) assigns each action profile an outcome.
- \(\mu = \langle \mu_1, \ldots, \mu_i, \ldots, \mu_n \rangle\) where \(\mu_i : O \rightarrow \mathbb{R}\) is a real-valued utility (payoff) function for player \(i\).
Note that we distinguish between **outcomes** and **utilities** assigned to them. Often, one assigns utilities directly to actions.

Games can be represented graphically using an $n$-dimensional payoff matrix. Here is a generic picture for 2-player, 2-strategy games:

<table>
<thead>
<tr>
<th>Player 1</th>
<th>$a_1^1$</th>
<th>$a_2^1$</th>
<th>$a_1^2$</th>
<th>$a_2^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1^1$</td>
<td>$(\mu_1(a_1^1, a_1^2), \mu_2(a_1^1, a_1^2))$</td>
<td>$(\mu_1(a_1^1, a_1^2), \mu_2(a_1^1, a_2^2))$</td>
<td>$(\mu_1(a_1^1, a_1^2), \mu_2(a_1^1, a_2^2))$</td>
<td>$(\mu_1(a_1^1, a_1^2), \mu_2(a_1^1, a_2^2))$</td>
</tr>
<tr>
<td>$a_1^2$</td>
<td>$(\mu_1(a_1^2, a_2^2), \mu_2(a_1^2, a_2^2))$</td>
<td>$(\mu_1(a_1^2, a_1^2), \mu_2(a_1^2, a_2^2))$</td>
<td>$(\mu_1(a_1^2, a_1^2), \mu_2(a_1^2, a_2^2))$</td>
<td>$(\mu_1(a_1^2, a_1^2), \mu_2(a_1^2, a_2^2))$</td>
</tr>
</tbody>
</table>
We often forget about \( \varrho \) (thus we are making no distinction between actions and outcomes). Thus we simply write \( \mu_1(a_1^1, a_1^2) \) instead of the more precise \( \mu_1(\varrho(a_1^1, a_1^2)) \). However, there are situations where we need to distinguish between the two, in particular when talking about mechanism design (in Chapter 6, and auctions (in Chapter 4, Section 7).
Definition 1.5 (Common Payoff Game)

A common payoff game (team game) is a game in which for all action profiles \( \vec{a} \in A_1 \times \ldots \times A_n \) and any two agents \( i, j \) the following holds: \( \mu_i(\vec{a}) = \mu_j(\vec{a}) \).

In such games agents have no conflicting interests. Their graphical depiction is simpler than above (the second component is not needed).
While a team game is on one side of the spectrum, there is another type of games which is on the opposite side:

**Definition 1.6 (Constant Sum Game)**

A 2-player \( n \)-strategy normal form game is called constant sum game, if there exists a constant \( c \) such that for each action profile \( \vec{a} \in A_1 \times A_2 \):

\[
\mu_1(\vec{a}) + \mu_2(\vec{a}) = c.
\]

We usually set wlog \( c = 0 \) (zero sum games).
Constant sum games can also be visualised with a simpler matrix, missing the second component (like common payoff games) \( \mu_2(\alpha_1^2, \alpha_2^2) \) in each entry:

<table>
<thead>
<tr>
<th></th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C</td>
</tr>
<tr>
<td>Player 1</td>
<td>( \frac{C}{D} )</td>
</tr>
</tbody>
</table>

Of course, we then have to state whether it is a **common payoff** or a **zero-sum** game (they are completely different).
What we are really after are strategies.

**Definition 1.7 (Pure strategy)**

A **pure strategy** for a player is a particular action that is chosen and then played constantly. A **pure strategy profile** is just an action profile \( \vec{a} = \langle a_1, \ldots, a_n \rangle \).

Are pure strategy profiles sufficient?
Example 1.8 (Rochambeau Game)

Also known as paper, rock and scissors: paper covers rock, rock smashes scissors, scissors cut paper.

<table>
<thead>
<tr>
<th></th>
<th>P</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Max</strong></td>
<td>P</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

What about pure vs mixed strategies?
Definition 1.9 (Mixed Strategy for NF Games)

Let \( \langle \mathbf{A}, \text{Act}, O, \varrho, u \rangle \) be normal form game. For a set \( X \) let \( \Pi(X) \) be the set of all \textit{probability distributions} over \( X \). The set of \textit{mixed strategies} for player \( i \) is the set \( S_i = \Pi(A_i) \). The set of mixed strategy profiles is \( S_1 \times \ldots \times S_n \). This is also called the \textit{strategy space} of the game.

Note: Some books use

- \( S_i \), with elements \( s_i \), to denote the set of \textit{pure} strategies,
- \( \Sigma_i \) with elements \( \sigma \) to denote the set of mixed strategies,
- \( u \) to denote utilities, and
- \( N \) to denote the set of agents.
The **support** of a mixed strategy is the **set of actions** that are assigned non-zero probabilities.

What is the payoff of such strategies? We have to take into account the probability with which an action is chosen. This leads to the expected utility $\mu_{\text{expected}}$. 
Definition 1.10 (Expected Utility for player \( i \))

The **expected utility** for player \( i \) of the mixed strategy profile \((s_1, \ldots, s_n)\) is defined as

\[
\mu_{\text{expected}}(s_1, \ldots, s_n) = \sum_{\tilde{a} \in \text{Act}} \mu_i(q(\tilde{a})) \prod_{j=1}^{n} s_j(a_j).
\]

What is the optimal strategy (maximising the expected payoff) for an agent in an 2-agent setting?
Example 1.11 (Fighters and Bombers)

Consider fighter pilots in WW II. A good strategy to attack bombers is to swoop down from the sun: Hun-in-the-sun strategy. But the bomber pilots can put on their sunglasses and stare into the sun to watch the fighters. So another strategy is to attack them from below Ezak-Imak strategy: if they are not spotted, it is fine, if they are, it is fatal for them (they are much slower when climbing). The table contains the survival probabilities of the fighter pilot.

<table>
<thead>
<tr>
<th>Fighter Pilots</th>
<th>Bomber Crew</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hun-in-the-Sun</td>
<td>Look Up</td>
</tr>
<tr>
<td>Ezak-Imak</td>
<td>Look Down</td>
</tr>
<tr>
<td></td>
<td>0.95</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
Example 1.12 (Battle of the Sexes, Type 2)

Married couple looks for evening entertainment. They prefer to go out together, but have different views about what to do (say going to the theatre and eating in a gourmet restaurant).

- **Husband**
  - Theatre: (4,3)
  - Restaurant: (1,1)

- **Wife**
  - Theatre: (2,2)
  - Restaurant: (3,4)
Example 1.13 (Leader Game, Type 3)

Two drivers attempt to enter a busy stream of traffic. When the cross traffic clears, each one has to decide whether to concede the right of way of the other (C) or drive into the gap (D). If both decide for C, they are delayed. If both decide for D there may be a collision.

<table>
<thead>
<tr>
<th></th>
<th>Driver 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C</td>
</tr>
<tr>
<td>Driver 1</td>
<td>(2,2)</td>
</tr>
<tr>
<td>C</td>
<td>(4,3)</td>
</tr>
<tr>
<td>D</td>
<td></td>
</tr>
</tbody>
</table>
Example 1.14 (Matching Pennies Game)

Two players display one side of a penny (head or tails). Player 1 wins the penny if they display the same, player 2 wins otherwise.

<table>
<thead>
<tr>
<th>Player 1</th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>(1, -1)</td>
</tr>
<tr>
<td>Tails</td>
<td>(-1, 1)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-1, 1)</td>
</tr>
<tr>
<td></td>
<td>(1, -1)</td>
</tr>
</tbody>
</table>
Definition 1.15 (Maxmin strategy)

Given a game \( \langle \{1, 2\}, \{A_1, A_2\}, \{\mu_1, \mu_2\} \rangle \), the maxmin strategy of player \( i \) is a mixed strategy that maximises the guaranteed payoff of player \( i \), no matter what the other player \( -i \) does:

\[
\text{arg max}_{s_i} \min_{s_{-i}} \mu_i^\text{expected}(s_i, s_{-i})
\]

The maxmin value for player \( i \) is \( \max_{s_i} \min_{s_{-i}} \mu_i^\text{expected}(s_i, s_{-i}) \).

The minmax strategy for player \( i \) is

\[
\text{arg min}_{s_i} \max_{s_{-i}} \mu_{-i}^\text{expected}(s_i, s_{-i})
\]

and its minmax value is \( \min_{s_i} \max_{s_{-i}} \mu_{-i}^\text{expected}(s_i, s_{-i}) \).
Lemma 1.16

In each finite normal form 2-person game (not necessarily constant sum), the maxmin value of one player is never strictly greater than the minmax value for the other.
We illustrate the maxmin strategy using a 2-person 3-strategy constant sum game:

<table>
<thead>
<tr>
<th>Player A</th>
<th>Player B</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-I</td>
<td>0</td>
</tr>
<tr>
<td>A-II</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B-I</th>
<th>B-II</th>
<th>B-III</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5/6</td>
<td>1/2</td>
</tr>
<tr>
<td>1</td>
<td>1/2</td>
<td>3/4</td>
</tr>
</tbody>
</table>

We assume Player A’s optimal strategy is to play strategy
- A-I with probability $x$ and
- A-II with probability $1 - x$.

In the following we want to determine $x$. 
Thus Player A’s expected utility is as follows:

1. when playing against B-I: \( 0x + 1(1 - x) = 1 - x \),
2. when playing against B-II: \( \frac{5}{6}x + \frac{1}{2}(1 - x) = \frac{1}{2} + \frac{1}{3}x \),
3. when playing against B-III: \( \frac{1}{2}x + \frac{3}{4}(1 - x) = \frac{3}{4} - \frac{1}{4}x \).

This can be illustrated with the following picture (see blackboard). Thus B-III does not play any role.

Thus the maxmin point is determined by setting

\[
1 - x = \frac{1}{2} + \frac{1}{3}x,
\]

which gives \( x = \frac{3}{8} \). The value of the game is \( \frac{5}{8} \).

The strategy for Player B is to choose B-I with probability \( \frac{1}{4} \) and B-II with probability \( \frac{3}{4} \).
More in accordance with the minmax strategy let us compute

\[
\arg \max_{s_i} \min_{s_{-i}} \mu_i^{\text{expected}}(s_1, s_2)
\]

We assume Player A plays (as above) A-I with probability \(x\) and A-II with probability \(1 - x\) (strategy \(s_1\). Similarly, Player B plays B-I with probability \(y\) and B-II with probability \(1 - y\) (strategy \(s_2\)).

We compute \(
\mu_1^{\text{expected}}(s_1, s_2)
\)

\[
0 \cdot x \cdot y + \frac{5}{6} x (1 - y) + 1 \cdot (1 - x) y + \frac{1}{2} (1 - x)(1 - y)
\]

thus

\[
\mu_1^{\text{expected}}(s_1, s_2) = y\left(\left(-\frac{4}{3} x\right) + \frac{1}{2}\right) + \frac{1}{3} x + \frac{1}{2}
\]
According to the minmax strategy, we have to choose $x$ such that the minimal values of the above term are maximal. For each value of $x$ the above is a straight line with some gradient. Thus we get the maximum when the line does not slope at all!

Thus $x = \frac{3}{8}$. A similar reasoning gives $y = \frac{1}{4}$. 
### Theorem 1.17 (von Neumann (1928))

In any finite 2-person constant-sum game the following holds:

1. The **maxmin value** for one player is equal to the **minmax value** for the other. The maxmin of player 1 is usually called **value of the game**.

2. For each player, the set of maxmin strategies coincides with the set of minmax strategies.

3. The maxmin strategies are **optimal**: if one player does not play a maxmin strategy, then its payoff goes down.
From now on we use just $\mu_1(s_1, s_2)$ instead of the more precise $\mu_1^{expected}(s_1, s_2)$. It will be clear from context whether the argument is a profile (and thus it is the expected utility $\mu^{expected}$) or it is the utility of an outcome (and thus it is defined in the underlying game with $\mu$).

What is the optimal strategy (maximising the expected payoff) for an agent in an $n$-agent setting?
Figure 1: A saddle.
Definition 1.18 (Notation $s_{\overrightarrow{i}}, S_{\overrightarrow{i}}$)

Note that from now on, for $\vec{s} = \langle s_1, s_2, \ldots, s_n \rangle$ we use the notation $s_{\overrightarrow{i}}$ to denote the strategy profile $\langle s_1, s_2, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n \rangle$: the strategies of all opponents of agent $i$ are fixed. For a set of strategies $S$, we denote by $S_{\overrightarrow{i}} = \{ s_{\overrightarrow{i}} | \vec{s} \in S \}$.

For ease of notation, we also use $\mu_i(s_i, s_{\overrightarrow{i}})$ to denote $\mu_i(\langle s_1, s_2, \ldots, s_n \rangle)$, thus

$$\langle s_1, s_2, \ldots, s_i, \ldots, s_n \rangle = \langle s_i, s_{\overrightarrow{i}} \rangle.$$

In the last vector, although the $s_i$ is written in the first entry, we mean it to be inserted at the $i$’th place.
Definition 1.19 (Best Response to a Profile)

Given a strategy profile

\[ \vec{s}_{-i} = \langle s_1, s_2, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n \rangle, \]

a best response of player \( i \) to \( \vec{s}_{-i} \) is any mixed strategy \( s_i^* \in S_i \) such that

\[ \mu_i(s_i^*, \vec{s}_{-i}) \geq \mu_i(s_i, \vec{s}_{-i}) \]

for all strategies \( s_i \in S_i \).

Is a best response unique?
Example 1.20 (Responses for Rochambeau)

How does the set of best responses look like?

1. Player 2 plays the pure strategy paper.
2. Player 2 plays paper with probability 0.5 and scissors with probability 0.5.
3. Player 2 plays paper with probability $\frac{1}{3}$ and scissors with probability $\frac{1}{3}$ and rock with probability $\frac{1}{3}$.

Observation

Is a non-pure strategy in the best response set (say a strategy $(a_1, a_2)$ with probabilities $\langle p, 1 - p \rangle$, $p \neq 0$), then so are all other mixed strategies with probabilities $\langle p', 1 - p' \rangle$ where $p \neq p' \neq 0$. 
Consider the set of best responses.

Either this set is a **singleton** (namely when it consists of a **pure strategy**), or

the set is **infinite**.
Definition 1.21 (Nash Equilibrium (NE))

A strategy profile \( \vec{s}^* = \langle s_1^*, s_2^*, \ldots, s_n^* \rangle \) is a Nash equilibrium if for any agent \( i \), \( s_i^* \) is a best response to \( \vec{s}_{-i}^* = \langle s_1^*, s_2^*, \ldots, s_{i-1}^*, s_{i+1}^*, \ldots, s_n^* \rangle \).

What are the Nash equilibria in the Battle of sexes? What about the matching pennies?
Example 1.22 (Cuban Missile Crisis, Type 4)

This relates to the well-known crisis in October 1962.

<table>
<thead>
<tr>
<th>U. S.</th>
<th>USSR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Withdrawal</td>
</tr>
<tr>
<td>Blockade</td>
<td>(3, 3)</td>
</tr>
<tr>
<td>Air strike</td>
<td>Compromise</td>
</tr>
<tr>
<td></td>
<td>(4, 2)</td>
</tr>
<tr>
<td></td>
<td>U.S. victory</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Theorem 1.23 (Nash (1950))

Every finite normal form game has a Nash equilibrium.

Corollary 1.24 (Nash implies maxmin)

In any finite normal form 2-person constant-sum game, the Nash equilibria are exactly all pairs \( \langle s_1, s_2 \rangle \) of maxmin strategies \((s_1 \text{ for player } 1, s_2 \text{ for player } 2)\). All Nash equilibria have the same payoff: the value of the game, that player 1 gets.
Proof

We use Kakatuni’s theorem: Let $X$ be a nonempty subset of $n$-dimensional Euclidean space, and $f : X \to 2^X$. The following are sufficient conditions for $f$ to have a fixed point (i.e. an $x^* \in X$ with $x^* \in f(x^*)$):

1. $X$ is compact: any sequence in $X$ has a limit in $X$.
2. $X$ is convex: $x, y \in X$, $\alpha \in [0, 1] \Rightarrow \alpha x + (1 - \alpha)y \in X$.
3. $\forall x : f(x)$ is nonempty and convex.
4. For any sequence of pairs $(x_i, x_i^*)$ such that $x_i, x_i^* \in X$ and $x_i^* \in f(x_i)$, if $\lim_{i \to \infty} (x_i, x_i^*) = (x, x^*)$ then $x^* \in f(x)$. 
Let $X$ consist of all mixed strategy profiles and let $f$ be the best response set: $f\left(\langle s_1, \ldots, s_n \rangle\right)$ is the set of all best responses $\langle s'_1, \ldots, s'_n \rangle$ (where $s'_i$ is players $i$ best response to $\langle s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n \rangle$).

**Why is that a subset of an $n$-dimensional Euclidean space?**

A mixed strategy over $k$ actions (pure strategies) is a $k-1$ dimensional simplex (namely the one satisfying $\sum_{i=1}^{k} p_i = 1$). Therefore $X$ is the cartesian product of $n$ simplices. $X$ is compact and convex (why?).

The function $f$ satisfies the remaining properties listed above. Thus there is a fixed point and this fixed point is a Nash equilibrium.
Theorem 1.25 (Brouwer’s Fixpoint Theorem)

Let \( D \) be the unit Euclidean ball in \( \mathbb{R}^n \) and let \( f : D \to D \) be a continuous mapping. Then there exists a fixed point of \( f \): there is a \( x \in D \) with \( f(x) = x \).
Proof

**Reduction to the $C^1$-differentiable case:** Let $r : \mathbb{R}^n \rightarrow \mathbb{R}$ be the function

$r(x) = r(x_1, x_2, \ldots, x_n) = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$.

Let $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ given by:

$$\phi(x) = a_n \left( 1/4r(x)^4 - 1/2r(x)^2 + 1/4 \right)$$

on $D$ and equal to 0 in the complement of $D$, where the constant $a_n$ is chosen such that the integral of $\phi$ over $\mathbb{R}^n$ equals 1.
Proof (cont.)

Let \( f : D \rightarrow D \) be continuous. Let \( F : \mathbb{R}^n \rightarrow D \) be the extension of \( f \) to \( \mathbb{R}^n \), for which we have

\[
F(x) = f\left(\frac{x}{||x||}\right) \text{ on } \mathbb{R}^n \setminus D.
\]

For \( k \in \mathbb{N}, k \neq 0 \), put \( f_k(x) := \int_{\mathbb{R}^n} k^n \phi(ky) F(x - y) dy \). Show that the restriction of \( f_k \) to \( D \) maps \( D \) into \( D \). Show that the mappings \( F_k \) are continuously differentiable and approximate in the topology of uniform convergence the mapping \( F \). Show that if there exists a continuous mapping \( f : D \rightarrow D \) without fixed points, then there will also exist a continuously differential mapping without fixed points. It follows, that it suffices to prove the Brouwer Fixpoint Theorem only for continuously differentiable mappings.
Proof (cont.)

**Proof for $C^1$-differentiable mappings:**

Assume, that the continuously differentiable mapping $f : D \to D$ has no fixed points. Let $g : D \to \partial D$ the mapping, such that for every point $x \in D$ the points $f(x), x, g(x)$ are in that order on a line of $\mathbb{R}^n$. The mapping $g$ is also continuously differentiable and satisfies $g(x) = x$ for $x \in \partial D$. We write $g(x) = (g_1(x), g_2(x), \ldots, g_n(x))$ and get (for $x \in \partial D$ and $i = 1 \ldots n$) $g_i(x_1, x_2, \ldots, x_n) = x_i$. Note $dg_1 \wedge dg_2 \wedge \cdots \wedge dg_n = 0$ since $g_1^2 + g_2^2 + \cdots + g_n^2 = 1$. Then:

$$0 \neq \int_D dx_1 \wedge dx_2 \wedge \cdots \wedge dx_n = \int_{\partial D} x_1 \wedge dx_2 \wedge \cdots \wedge dx_n$$

$$= \int_{\partial D} g_1 \wedge dg_2 \wedge \cdots \wedge dg_n = \int_D dg_1 \wedge dg_2 \wedge \cdots \wedge dg_n$$

$$= \int_D 0 = 0$$
Example 1.26 (Majority Voting)

Consider agents 1, 2, 3 and three outcomes A, B, C. Agents vote simultaneously for one outcome (no abstaining). The outcome with most votes wins. If there is no majority, then A is selected. The payoff functions are as follows:

\[\mu_1(A) = \mu_2(B) = \mu_3(C) = 2, \ \mu_1(B) = \mu_2(C) = \mu_3(A) = 1\text{ and } \mu_1(C) = \mu_2(A) = \mu_3(B) = 0\].

What are the Nash equilibria and what are their outcomes?
### Example 1.27 (Unique Equilibrium)

The following game has exactly one Nash equilibrium.

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>$\langle 1, -2 \rangle$</td>
<td>$\langle -2, 1 \rangle$</td>
<td>$\langle 0, 0 \rangle$</td>
</tr>
<tr>
<td>M</td>
<td>$\langle -2, 1 \rangle$</td>
<td>$\langle 1, -2 \rangle$</td>
<td>$\langle 0, 0 \rangle$</td>
</tr>
<tr>
<td>D</td>
<td>$\langle 0, 0 \rangle$</td>
<td>$\langle 0, 0 \rangle$</td>
<td>$\langle 1, 1 \rangle$</td>
</tr>
</tbody>
</table>
Minmax value and feasible payoffs

Definition 1.28 (Minmax value and feasible payoffs)

In a $n$-person normal form game $\langle \mathcal{A}, \text{Act}, O, \varphi, \mu \rangle$ we define the minmax value of player $i$ as follows

$$v_i = \min_s \max_{s_i} \mu_i^{\text{expected}}(s_i, s_{-i}) .$$

We call a payoff profile $\langle r_1, \ldots, r_i, \ldots, r_n \rangle$ feasible if there exist rational values $\alpha_{\tilde{a}} \geq 0$ such that for all $i$

$$r_i = \sum_{\tilde{a} \in \text{Act}} \alpha_{\tilde{a}} \mu_i(\tilde{a}) \text{ where } \sum_{\tilde{a} \in \text{Act}} \alpha_{\tilde{a}} = 1.$$ 

Geometrically speaking, this is the (rational) convex hull of all possible payoffs of pure action profiles.
Feasible payoffs

What is the idea behind feasible payoffs?

- The minmax value is important, as this is the minimal payoff in any Nash equilibrium.
- Not any feasible profile can be obtained as payoff. In the leader game in Example 1.13 on Slide 36, the profile \( \langle \frac{7}{2}, \frac{7}{2} \rangle \) is feasible but can not be realised in one game.

What if the game is played infinitely often (or just twice)?

- In general, convex combinations of pure-strategy payoffs can only be obtained by correlated strategies, not by independent randomizations.

In fact, feasible payoffs (and their convexity) will play a role later in Section 2 on Slide 168 (Theorem 2.10). They can be realized in repeated games.
It is obvious that **there is not always** a strategy that is **strictly dominating** all others (this is why the Nash equilibrium has been introduced).

**Reducing games**

However, often games can be **reduced** and the computation of the equilibrium considerably simplified.

**A rational player would never choose a strategy that is strictly dominated.**
A pure strategy $a_i$ is **strictly dominated** for an agent $i$, if there exists some other (mixed) strategy $s'_i$ that strictly dominates it, i.e. for all profiles $a_{-i} = \langle a_1, \ldots a_{i-1}, a_{i+1}, \ldots a_n \rangle$, we have

$$\mu_i(\langle s'_i, a_{-i} \rangle) > \mu_i(\langle a_i, a_{-i} \rangle).$$

We say that a pure strategy $a_i$ is **weakly dominated** for an agent $i$, if in the above inequality we have $\geq$ instead of $>$ and the inequality is strict for at least one of the other $a_{-i}$. 
Definition 1.30 (Reduced Sets $A^\infty_i$, $S^\infty_i$)

For an arbitrary normal form game with $A_i$ the set of pure strategies and $S_i$ the set of mixed strategies for agent $i$, we define ($A^0_i := A_i$, $S^0_i := S_i$)

$$A^n_i := \{a_i \in A^{n-1}_i \mid \text{there is no } s'_i \in S^{n-1}_i \text{ s.t.} \mu_i(s'_i, a^-_i) \not\geq \mu_i(a_i, a^-_i) \text{ for all } a^-_i \in A^{n-1}_{-i}\}$$

$$S^n_i := \{s' \in S_i \mid s'(a_i) \not\geq 0 \text{ only if } a_i \in A^n_i\}$$

Finally, $A^\infty_i := \bigcap_{n=0}^{\infty} A^n_i$ and $S^\infty_i$ is the set of all mixed strategies $s_i$, such that there is no $s'_i$ with $\mu_i(s'_i, a^-_i) \not\geq \mu_i(s_i, a^-_i)$ for all $a^-_i \in A^\infty_{-i}$.
Some Comments to $A_i^\infty$ and $S_i^\infty$

- What are the $S_i^n$? They are the sets of mixed strategies over only the pure strategies in $A_i^n$.
- The $A_i^n$ are sets of pure strategies from which we remove those, that are strictly dominated by certain other mixed strategies.
- Therefore only the strategies in $A_i^\infty$ are those that we have to keep.
- Note that $S_i^\infty$ is defined wrt. $A_i^\infty$, not wrt. $S_i^n$.
- $S_i^\infty$ is the set of mixed strategies that are not strictly dominated by pure action profiles from $A_i^\infty$.
- Note that $S_i^\infty$ can be strictly smaller than the set of all mixed strategies over $A_i^\infty$. 
### Theorem 1.31 (Solvable by Iterated Strict Dominance)

*If for a finite normal form game, the sets $A_i^\infty$ are all singletons (such a game is called *solvable by iterated strict dominance*), then this strategy profile is the unique Nash equilibrium.*
Lemma 1.32 (Church-Rosser)

Given a 2-person normal form game. All strictly dominated columns, as well as all strictly dominated rows can be eliminated without changing the Nash equilibria (or similar solution concepts). This results in a finite series of reduced games. The final result does not depend on the order of the eliminations.

Note: the last lemma is not true for weakly dominated strategies. There, the order does matter.
Note that we eliminate only pure strategies. Such a strategy might be dominated by a mixed strategy.

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>⟨3,2⟩</td>
<td>⟨2,1⟩</td>
<td>⟨3,1⟩</td>
</tr>
<tr>
<td>M</td>
<td>⟨1,1⟩</td>
<td>⟨1,1⟩</td>
<td>⟨2,2⟩</td>
</tr>
<tr>
<td>D</td>
<td>⟨0,1⟩</td>
<td>⟨4,2⟩</td>
<td>⟨0,1⟩</td>
</tr>
</tbody>
</table>

1. Eliminate row M.
2. Eliminate column R.
This leads to

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>U</strong></td>
<td>⟨3, 2⟩</td>
<td>⟨2, 1⟩</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>⟨0, 1⟩</td>
<td>⟨4, 2⟩</td>
</tr>
</tbody>
</table>
Elimination of Weakly Dominated actions

We consider the normal form game

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>⟨1,1⟩</td>
<td>⟨0,0⟩</td>
</tr>
<tr>
<td>M</td>
<td>⟨1,1⟩</td>
<td>⟨2,1⟩</td>
</tr>
<tr>
<td>B</td>
<td>⟨0,0⟩</td>
<td>⟨2,1⟩</td>
</tr>
</tbody>
</table>

1. If we first eliminate T, and then L we get the outcome ⟨2, 1⟩.

2. If we first eliminate B, and then R we get the outcome ⟨1, 1⟩.
1.3 Extensive Form Games
We have previously introduced **normal form games** (Definition 1.4 on Slide 23). This notion does not allow to deal with sequences of actions that are **reactions to actions** of the opponent.

**Extensive form (tree form) games**

Unlike games in normal form, those in **extensive form** do not assume that all moves between players are made simultaneously. This leads to a **tree form**, and allows to introduce **strategies**, that take into account the **history** of the game.

We distinguish between **perfect** and **imperfect** information games. While the former assume that the players have **complete** knowledge about the game, the latter do not: a player might not **know** exactly which node it is in.
The following definition covers a game as a tree:

**Definition 1.33 (Perfect Extensive Form Games)**

A finite perfect information game in extensive form is a tuple \( \Gamma = \langle \mathbf{A}, \text{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle \) where

- \( \mathbf{A} \) is a set of \( n \) players, \( \text{Act} \) is a set of actions
- \( H \) is a set of non-terminal nodes, \( Z \) a set of terminal nodes, \( H \cap Z = \emptyset \), \( H \cup Z \) form a tree,
- \( \alpha : H \to 2^{\text{Act}} \) assigns to each node a set of actions,
- \( \rho : H \to \mathbf{A} \) assigns to each non-terminal node a player who chooses an action at that node,
- \( \sigma : H \times A \to H \cup Z \) assigns to each \((\text{node}, \text{action})\) a successor node \( (h_1 \neq h_2 \text{ implies } \sigma(h_1, a_1) \neq \sigma(h_2, a_2)) \),
- \( \mu_i : Z \to \mathbb{R} \) are the utility functions.
Such games can be visualised as trees. Here is the famous “Sharing Game”.

**Example 1.34 (Sharing Game)**

The game consists of two rounds. In the first, player 1 offers a certain share (namely (1) 2 for player 1, 0 for player 2, (2) 1 for player 1, 1 for player 2, (3) 0 for player 1, 2 for player 2). Player 2 can only accept, or refuse. In the latter case, nobody gets anything.
Figure 2: The Sharing game.
Strategies in extensive form games

Definition 1.35 (Strategies in Extensive Form Games)

Let $\Gamma = \langle A, \text{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle$ be a finite perfect information game in extensive form. A strategy for player $i$ in $\Gamma$ is any function that assigns a legal move to each history owned by $i$. The pure strategies of player $i$ are the elements of $\prod_{h \in H, \rho(h)=i} \alpha(h)$. These are also functions: whenever player $i$ can do a move, it chooses one of the actions available. Thus we can write a pure strategy as a vector $\langle a_1, \ldots, a_r \rangle$, where $a_1, \ldots, a_r$ are $i$’s choices at the respective moves.
In the sharing game, a pure strategy for player 2 is \(\langle \text{no, yes, no} \rangle\). A (better) one is \(\langle \text{no, yes, yes} \rangle\).

Why don’t we introduce mixed strategies?

**Best response, Nash Equilibrium**

Note that the definitions of best response and Nash equilibrium carry over (literally) to games in extensive form.

Note that in the following we are talking only about pure strategy profiles.
What are the NE’s in the sharing game?

⟨1, ⟨y, y, y⟩⟩, ⟨1, ⟨n, n, n⟩⟩, ⟨1, ⟨n, n, y⟩⟩, ⟨1, ⟨y, n, n⟩⟩, ⟨1, ⟨y, n, y⟩⟩, ⟨1, ⟨y, y, n⟩⟩ are NE’s, ⟨1, ⟨n, y, y⟩⟩ is not. Also ⟨2, ⟨n, y, n⟩⟩, ⟨2, ⟨n, y, y⟩⟩ and ⟨3, ⟨n, n, y⟩⟩ are NE’s.

We claim that only ⟨1, ⟨y, y, y⟩⟩ and ⟨2, ⟨n, y, y⟩⟩ make sense.
Lemma 1.36 (Extensive form $\leftrightarrow$ Normal form)

Each game $\Gamma$ in perfect information extensive form can be transformed to a game $NF(\Gamma)$ in normal form (such that the pure strategy spaces correspond).
Proof.

A **strategy profile** determines a **unique** path from the root \( \emptyset \) of the game to one of the terminal nodes (and hence also a single profile of payoffs). Therefore one can construct the corresponding normal form game \( NF(\Gamma) \) by enumerating all strategy profiles and filling the payoff matrix with the resulting payoffs.
### Sharing Game in normal form

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>nnn</th>
<th>nny</th>
<th>nyn</th>
<th>nyy</th>
<th>ynn</th>
<th>yny</th>
<th>yyn</th>
<th>yyy</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2, 0)</td>
<td></td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(2, 0)</td>
<td>(2, 0)</td>
<td>(2, 0)</td>
<td>(2, 0)</td>
</tr>
<tr>
<td>(1, 1)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(1, 1)</td>
<td>(1, 1)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(1, 1)</td>
<td>(1, 1)</td>
<td></td>
</tr>
<tr>
<td>(0, 2)</td>
<td>(0, 0)</td>
<td>(0, 2)</td>
<td>(0, 0)</td>
<td>(0, 2)</td>
<td>(0, 0)</td>
<td>(0, 2)</td>
<td>(0, 0)</td>
<td>(0, 2)</td>
<td></td>
</tr>
</tbody>
</table>
Figure 3: A generic game.
Example 1.37 (Generic Game in normal form)

We consider the game in Figure 3. The pure strategies of player 1 are \{⟨A, E⟩, ⟨A, F⟩, ⟨B, E⟩, ⟨B, F⟩\}. The pure strategies of player 2 are \{C, D\}.

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>W</td>
<td>Y</td>
</tr>
<tr>
<td>AF</td>
<td>X</td>
<td>Y</td>
</tr>
<tr>
<td>BE</td>
<td>Z</td>
<td>Z</td>
</tr>
<tr>
<td>BF</td>
<td>Z</td>
<td>Z</td>
</tr>
</tbody>
</table>

Note that ⟨B, E⟩, ⟨B, F⟩ are pure strategies that have to be considered.
Is there a converse of Lemma 1.36?

We consider prisoner’s dilemma and try to model a game in extensive form with the same payoffs and strategy profiles.
In fact, it is not surprising that we do not succeed in the general case:

**Theorem 1.38 (Zermelo, 1913; Kuhn)**

*For each perfect information game in extensive form there exists a pure strategy Nash equilibrium. The theorem can be strengthened (Kuhn’s theorem): For each perfect information game in extensive form there exists a pure strategy subgame Nash equilibrium.*

In fact, this was the reason that we do not need mixed strategies for perfect information extensive games (question on Slide 79).

We will later introduce imperfect information games (in extensive form): Slide 99.
Proof.

By backward induction we construct a subgame perfect Nash equilibrium $s$:

- Let $h$ be a terminal history and $h'$ be the history with the last action removed, say $h = h'a$. Moreover, let it be player $i$’s move in $h'$. Then, we define $s_i(h')$ to be the action which maximizes $i$’s payoff. We proceed like this for all such histories $h'$.

- Suppose now that $s$ is a subgame perfect Nash equilibrium for all histories of a certain length, say $k$. Consider a history $h' = ha$ of length $k + 1$. As before the player whose move it is in $h'$ chooses an action which maximizes its payoff assuming that all other players follow $s$. We proceed like this for all histories $h'$.

- The constructed strategy $s$ is a subgame perfect Nash equilibrium.
Example 1.39 (Unintended Nash equilibria)

Consider the following game in extensive form.

Figure 4: Unintended Equilibrium.
The game depicted in Example 1.39 has two equilibria: \( \langle A, R \rangle \) and \( \langle B, L \rangle \). The latter one is not intuitive (while the first one is).

Can we refine the notion of NE and rule out this unintended equilibrium?
This leads to the notion of **subgame perfect Nash equilibria**:

**Definition 1.40 (Subgame Perfect NE (SPE))**

Let $\Gamma$ be a perfect information game in extensive form.

**Subgame:** A subgame of $G$ rooted at node $h$ is the restriction of $\Gamma$ to the descendants of $h$.

**SPE:** The subgame perfect Nash equilibria (SPE) of a perfect information game $\Gamma$ in extensive form are those Nash equilibria of $\Gamma$, that are also Nash equilibria for all subgames $\Gamma'$ of $\Gamma$. 
What are the SPE’s in the Sharing game (Example 1.34)?

What are the SPE’s in the following instance of the generic game:

<table>
<thead>
<tr>
<th></th>
<th>$C$</th>
<th>$D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$AE$</td>
<td>$\langle 2, 0 \rangle$</td>
<td>$\langle 1, 1 \rangle$</td>
</tr>
<tr>
<td>$AF$</td>
<td>$\langle 0, 2 \rangle$</td>
<td>$\langle 1, 1 \rangle$</td>
</tr>
<tr>
<td>$BE$</td>
<td>$\langle 3, 3 \rangle$</td>
<td>$\langle 3, 3 \rangle$</td>
</tr>
<tr>
<td>$BF$</td>
<td>$\langle 3, 3 \rangle$</td>
<td>$\langle 3, 3 \rangle$</td>
</tr>
</tbody>
</table>
Theorem 1.41 (Existence of SPE (Kuhn))

For each finite perfect information game in extensive form there exists a SPE.

The proof (on blackboard) is by induction on the length of histories. The SPE is therefore defined constructively.
Example 1.42 (Centipede Game)

This is a two person game which illustrates that even the notion of SPE can be critical.

\[
\begin{array}{cccccc}
1 & A & 2 & A & 1 & A \\
D & D & D & D & D \\
(1,0) & (0,2) & (3,1) & (2,4) & (4,3) \\
\end{array}
\]

(3,5)
The Centipede game has just one SPE: All players always choose $D$.

This is rational, but humans often do not behave like that.

Experiments show, that humans start with going across and do a down only towards the end of the game.
Imperfect Information

- In an extensive game with **perfect** information, the player does know all previous moves (and also the payoffs that result).
- In an extensive game with **imperfect** information, a player might not be completely informed about the past history.
- Or some moves in the past may have been **done randomly**, so in the future, even under the same circumstances, other action may be taken.
The Idea

- An **extensive game** is nothing else than a **tree**. Thus each node is unique and carries with it the path from the root (the history that lead to it).

- In order to model that a player does not perfectly know the past events, we introduce an **equivalence relation** on the nodes. That two nodes are **equivalent**, means that the **player cannot distinguish** between them.

- All nodes in one equivalence class must be assigned the same actions: otherwise the player could distinguish them.
Definition 1.43 (Information set $I_i$, Partition)

For a set $W$ (nodes, worlds, games) and a set of agents $\mathcal{A}$, we define a partition $I_i$ of agent $i \in \mathcal{A}$ over it (or the information set of $i$) as an equivalence relation over $W$. Its classes $I_{ij}$ are also called partition classes. Thus the following holds.

A partition $I_i$ is a set of subsets $W_{i1}, \ldots W_{is}$ of $W$ such that:
1. $\bigcup_i W_{ij} = W$
2. $W_{ij} \cap W_{ij'} = \emptyset$ for $j \neq j'$.
Definition 1.44 (Extensive Games, Imperfect Inf.)

A finite imperfect information game in extensive form is a tuple \( G = \langle \mathcal{A}, \text{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n, I_1, \ldots, I_n \rangle \) where

- \( \langle \mathcal{A}, \text{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle \) is a perfect information game in the sense of Definition 1.33 on Slide 75,

- \( I_i \) are partitions on \( \{ h \in H : \rho(h) = i \} \) such that \( h, h' \in I_{i,j} \) implies \( \alpha(h) = \alpha(h') \).
Example 1.45

(Utility of the rightmost leaf node is $\langle 2, 1 \rangle$.)
Player 1 can not distinguish between the nodes connected by a dotted line.

Therefore player 1 can not play the right move.

It could play a mixed strategy: with probability $\frac{1}{2}$ choose $l$.

Now we need mixed strategies, to deal with the uncertainty.
Definition 1.46 (Pure strategy in Extensive Form)

Given an imperfect information game in extensive form, a pure strategy for player $i$ is a vector $\langle a_1, \ldots, a_k \rangle$ with $a_j \in \alpha(I_{ij})$ where $I_{i1}, \ldots I_{ik}$ are the $k$ equivalence classes for agent $i$. Note that this vector is nothing than a function assigning an action to each node owned by player $i$.

Can we model prisoner’s dilemma as an extensive game with imperfect information?
There is a pure strategy Nash equilibrium.
But we could have chosen to switch player 1 with player 2.
NF game $\leftrightarrow$ Imperfect game

For pure strategies we have the following:

- Each *game in normal form* can be transformed into an *imperfect information game in extensive form* (but this is not one-to-one).

- Each imperfect information game in extensive form can be transformed into a game in normal form (this is one to one).

What are *mixed* strategies for an imperfect information game?
Mixed Strategy: First try.

- Given an imperfect information game in extensive form $\Gamma$.
- Assign the normal form game (for any $i$) as usual, by enumerating the pure strategies.
- Now we can take the usual set of mixed strategies in the normal form game as the set of mixed strategies of the original game $\Gamma$. 
Behavioral Strategy: Second try.

- We consider the game from Figure 3 on Slide 84.
- Consider the following strategy for player 1. $A$ is chosen with probability 0.7, $B$ with 0.3 and $E$ with probability 0.4 and $F$ with 0.6. Such strategies are called behavioral: at each node, the (probabilistic) choice is made independently from the other nodes.

- Consider the following mixed strategy for player 1. $\langle A, E \rangle$ is chosen with probability 0.6 and $\langle B, F \rangle$ with probability 0.4. Thus, here we have a strong correlation: $\langle A, F \rangle$ is not possible!
Mixed vs. Behavioral

Definition 1.47 (Mixed and Behavioral strategies)

Let $G = \langle A, \text{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n, I_1, \ldots, I_n \rangle$ be an imperfect information game in extensive form.

**Mixed:** A mixed strategy of player $i$ is one single probability distribution over $i$’s pure strategies.

**Behavioral:** A behavioral strategy of player $i$ is a vector of probability distributions $P(I_{ij})$ over the set of actions $\alpha(I_{ij})$ for $I_{ij} \in I_i$. We define by $P(h)(a)$ the probability $P(I_{ij})(a)$ for the action $a$ for player $i$ if $h \in I_{ij}$. 
Mixed vs. Behavioral (2)

The main difference is that for behavioral strategies, at each node the probability distribution is started freshly. Even if a player ends up in the same partition, she can choose independently of her previous choice. Whereas for mixed strategies, this choice is not independent: there is just one single distribution that relates the possible choices.
Are behavioral strategies more general?

We consider a one player game. At the start node, the player can choose $L$ or $R$. There result two nodes which can not be distinguished by the player. Again, $L$ or $R$ can be played and result in the four outcomes $o_1, o_2, o_3, o_4$.

- What is the outcome of the mixed strategy $\langle \frac{1}{2}LL, \frac{1}{2}RR \rangle$?
- It is $\langle \frac{1}{2}, 0, 0, \frac{1}{2} \rangle$
- **No behavioral strategy** results in this distribution.
- Therefore mixed strategies are **not necessarily** behavioral.
Example 1.48 (A game of imperfect recall)

We consider the following game

For mixed strategies, \( \langle R, D \rangle \) is the unique NE. But for behavioral strategies, the following mixed strategy is a better response of player 1 to \( D \): \( \left( \frac{98}{198} L, \frac{100}{198} R \right) \).
Are mixed strategies more general? (2)

- For mixed strategies, once decided, the pure strategy is consistently chosen. Therefore the outcome $\langle 100, 100 \rangle$ is not reachable.

- This is not true for behavioral strategies, where at each node, the probabilistic choice is done independently.

- **What is the best response of player 1 to $D$ in behavioral strategies?** Consider a mixed behavioral strategy: choose $L$ with probability $p$ and $R$ with $1 - p$. A little computation shows that the maximal payoff is obtained for $p = \frac{98}{198}$. 
Behavioral vs. mixed strategies?

- We have just seen that there are mixed strategies for which there are no behavioral strategies with the same outcome and vice versa.

- Therefore we introduce two concepts of Nash equilibria on the next page.

- Is there a class of games where both concepts are equivalent?
Definition 1.49 (NE for Mixed/Behavioral Strategies)

Let \( G \) be an extensive game. A **NE in mixed strategies** is a mixed strategy profile \( s^* = \langle s_1^*, s_2^*, \ldots, s_n^* \rangle \), s.t. for any agent \( i \):

\[
\mu_i(\langle s_i^*, s_{-i}^* \rangle) \geq \mu_i(\langle s_i, s_{-i}^* \rangle)
\]

for all mixed strategies of player \( i \).

A **NE in behavioral strategies** is a behavioral strategy profile \( s^* = \langle s_1^*, s_2^*, \ldots, s_n^* \rangle \), s.t. for any agent \( i \):

\[
\mu_i(\langle s_i^*, s_{-i}^* \rangle) \geq \mu_i(\langle s_i, s_{-i}^* \rangle)
\]

for all behavioral strategies of player \( i \).
Definition 1.50 (Perfect Recall)

Let $\Gamma$ be an imperfect information game in extensive form. We say that player $i$ has **perfect recall** in $\Gamma$, if the following holds. If $h, h'$ are two nodes in the same $I_{ij}$ (for a $j$), and $h_0, a_0, h_1, a_1, \ldots, h_n, a_n, h$ resp. $h'_0, a'_0, h'_1, a'_1, \ldots, h'_m, a'_m, h'$ are paths from the root of the tree to $h$ (resp. $h'$), then

1. $n = m$,
2. for all $0 \leq j \leq n$: $h_j$ and $h'_j$ are in the same partition class,
3. for all $0 \leq j \leq n$: if $\alpha(h_j) = i$ then $a_j = a'_j$.

$\Gamma$ is a **game of perfect recall**, if all players have perfect recall. Otherwise it is called of **imperfect recall**.
A few games: Which have perfect recall?
Do they model the same situation?
Perfect Recall: Behavioral strategies suffice?

**Theorem 1.51 (Behavioral = Mixed (Kuhn, 1953))**

Let \( \Gamma \) be a game of perfect recall (perfect or imperfect information). Then for any mixed strategy of agent \( i \) there is a behavioral one such that both strategies induce the same probabilities on outcomes for all fixed strategy profiles of the other agents.

**Corollary 1.52**

In a game of perfect recall, it suffices to compute the Nash equilibria based on behavioral strategies.
**SPE:** What about *subperfect* equilibria (analogue of Definition 1.40 on Slide 91 for imperfect games?

**First try:** In each information set, we have a set of subgames (a *forest*). Why not asking that a strategy should be a *best response in all subgames of that forest*?
Example 1.53 (A Game with no SPE’s)

```
1
   /\  \\
  /   \  \\
L     C   R
   |     |
   |     |
  /\   /\  \\
 /   /   /\  \\
(1,1) 2   2
   U   D   U   D
   (0,1000) (0,0) (1,0) (3,1)
```
Nash equilibria: \((L, U)\) and \((R, D)\). None of them is a subgame perfect Nash equilibrium.

In one subgame, \(U\) dominates \(D\), in the other \(D\) dominates \(U\).

But \((R, D)\) seems to be the unique choice: both players can put themselves into the others place and reason accordingly.

Requiring that a strategy is best response to all subgames is too strong.
There are two prominent refinements of SPE’s.

- One is the trembling hand perfect equilibrium. It is defined for normal form, and extensive form games.
- The other the sequential equilibrium is defined for extensive form games of perfect recall.
We use a **belief system** $\mu$: functions which assign to each information set $I_{i,j}$ a probability measure over nodes in $I_{i,j}$. $\mu(I_{ij})(h)$ is the probability that player $i$ assigns to history $h$, conditional on $I_{i,j}$ being reached.

So, a belief system captures the probability of being in a specific node of an information set.

Given a behavioral strategy profile $\beta$ which is completely mixed (i.e. assigns non-zero probability to all actions), a belief system $\mu$ can be **uniquely assigned to** $\beta$. 
1 Complete Information Games

1.3 Extensive Form Games

Definition 1.54 (Sequential Equilibrium)

A (behavioral) strategy profile $\beta^* = \langle \beta^*_1, \beta^*_2, \ldots, \beta^*_n \rangle$ is a **sequential equilibrium** of an extensive form game $\Gamma$ if there exist probability distributions $\mu(h)$ for each information set $I_i$ such that

1. $(\beta^*, \mu) = \lim_{n \to \infty} (\beta^n, \mu^n)$ for some sequence where $\beta^n$ is completely mixed (where $\mu^n$ is uniquely determined by $\beta^n$),

2. for any $I_i$ of agent $i$ and any alternative strategy $\beta'_i$ of agent $i$: $\mu_i(\beta^* \mid h, \mu(h)) \geq \mu_i((\beta'_i, \beta_{-i}) \mid h, \mu(h))$.

The first assumption (consistency) is of a rather technical nature to enable the consistent definition of expectations in case of behavioral strategies which assign probability 0 to actions.
Theorem 1.55 (Sequential Equilibrium)

For each imperfect information game in extensive form with perfect recall there exists a sequential equilibrium.

For perfect information games, each SPE is a sequential equilibrium but not vice versa.
1.4 An Example from Economics
A theory for efficiently allocating goods and resources among agents, based on market prices.

**Goods:** Given $n > 0$ goods $g_1, \ldots, g_n$ (coffee, mirror sites, parameters of an airplane design). We assume $g_i \neq g_j$ for $i \neq j$ but within each $g_i$ everything is indistinguishable.

**Prices:** The market has prices $p = [p_1, p_2, \ldots, p_n] \in \mathbb{R}^n$: $p_i$ is the price of the good $i$. 
**Consumers:** Consumer $i$ has $\mu_i(x)$ encoding its preferences over consumption bundles $x_i = [x_{i1}, \ldots, x_{in}]^t$, where $x_{ig} \in \mathbb{R}^+$ is consumer $i$’s allocation of good $g$. Each consumer also has an initial endowment $e_i = [e_{i1}, \ldots, e_{in}]^t \in \mathbb{R}$.

**Producers:** Use some commodities to produce others: $y_j = [y_{j1}, \ldots, y_{jn}]^t$, where $y_{jg} \in \mathbb{R}$ is the amount of good $g$ that producer $j$ produces. $Y_j$ is a set of such vectors $y$.

**Profit of producer $j$:** $p \times y_j$, where $y_j \in Y_j$. 
Profits: The profits are divided among the consumers (given predetermined proportions $\Delta_{ij}$): $\Delta_{ij}$ is the fraction of producer $j$ that consumer $i$ owns (stocks). Profits are divided according to $\Delta_{ij}$. 
Definition 1.56 (General Equilibrium)

\((p^*, x^*, y^*)\) is in general equilibrium, if the following holds:

I. The markets are in equilibrium:

\[
\sum_i x_i^* = \sum_i e_i + \sum_j y_j^*
\]

II. Producer \(j\) maximises profit wrt. the market

\[
y_i^* = \arg \max_{y_j \in Y_j} p^* \times y_j
\]
III. Consumer $i$ maximises preferences according to the prices

$$x_i^* = \arg \max \{x_i \in \mathbb{R}^n \mid \text{cond}_{i} \} \mu_i(x_i)$$

where $\text{cond}_{i}$ stands for

$$p^* \times x_i \leq p^* \times e_i + \sum_j \Delta_{ij} p^* \times y_j.$$
Theorem 1.57 (Pareto Efficiency)
Each general equilibrium is pareto efficient.

Theorem 1.58 (Coalition Stability)
Each general equilibrium with no producers is coalition-stable: no subgroup can increase their utilities by deviating from the equilibrium and building their own market.
Theorem 1.59 (Existence of an Equilibrium)

Let the sets $Y_j$ be closed, convex and bounded above. Let $\mu_i$ be continuous, strictly convex and strongly monotone. Assume further that at least one bundle $x_i$ is producible with only positive entries $x_{il}$.

Under these assumptions a general equilibrium exists.
Meaning of the assumptions

Formal definitions: \( \rightsquigarrow \) blackboard.

**Convexity of** \( Y_j \): Economies of scale in production do not satisfy it.

**Continuity of the** \( \mu_i \): Not satisfied in bandwidth allocation for video conferences.

**Strictly convex:** Not satisfied if preference increases when one gets more of this good (drugs, alcohol, dulce de leche).
In general, there exist more than one equilibrium.

**Theorem 1.60 (Uniqueness)**

*If the society-wide demand for each good is non-decreasing in the prices of the other goods, then a unique equilibrium exists.*

This condition is called *gross substitutes property* and comes in many variants.

**Positive example**: increasing price of meat forces people to eat potatoes (pasta).

**Negative example**: increasing price of bread implies that the butter consumption decreases.
How to find market equilibria?

We describe an algorithm using steepest descent.

Theorem 1.61

The price tâtonnement algorithm, explained on the next few pages, converges to a general equilibrium if for all $p$ that are not proportional to an equilibrium vector $p^*$, the following holds:

$$ \sum_i (x_i(p) - e_i) - \sum_j y_j(p) \geq 0 $$
Price tâtonnement process (1)

- This is a **decentralized** algorithm that performs steepest descent (can be improved by Newton-method).
- The main part is a **price adjustor**, that suggests a price and receives production plans from the producers and consumption plans from the consumers.
- Based on these plans, a new price is calculated and the cycle starts again.
- Producer $j$ takes the current price and develops a production plan maximizing its profit. This plan is sent to the adjustor.
- Consumer $i$ takes the current price and production plans from the producers and develops a consumption plan maximizing its utility given budget constraints. This plan is sent to the adjustor.
Price tâtonnement: The adjustor

for $g = 1$ to $n$ do
    $p_g \leftarrow 1$
end for

for $g = 1$ to $n$ do
    $\lambda_g \leftarrow$ a positive number
end for
repeat
    Broadcast $p$ to consumers and producers.
    Receive a production plan $y_j$ from each producer $j$.
    Broadcast the plans $y_j$ to consumers.
    Receive a consumption plan $x_i$ from each consumer $i$.
    for $g = 1$ to $n$ do
        $p_g \leftarrow p_g + \lambda_g (\sum_i (x_{ig} - e_{ig}) - \sum_j y_{jg})$
    end for
until $|\sum_i (x_{ig} - e_{ig}) - \sum_j y_{jg}| \leq \epsilon$ for all $1 \leq g \leq n$
Inform consumers and producers that an equilibrium has been reached.
repeat
  Receive $p$ from the adjustor.
  Receive a production plan $y_j$ for all $j$ from the adjustor.
  Announce to the adjustor a consumption plan $x_i$ that maximizes $i$’s utility given the budget constraint (see Condition III on Slide 130).
until Informed by adjustor that equilibrium has been reached.
Exchange and consume.
repeat
  Receive $p$ from the adjustor.
  Receive a production plan for all $j$ from the adjustor.
  Announce to the adjustor a production plan $y_j \in Y_j$
  that maximizes $p \times y_j$
until Informed by adjustor that equilibrium has been reached.
Exchange and produce.
1.5 References
1.5 References

*Game Theory.*
MIT Press.

Equilibrium points in n-person games.
*Proceedings of the National Academy of Sciences of the United States of America* 36, 48–49.

*A Course in Game Theory.*
MIT Press.
2. Repeated Games

- Examples and Motivation
- Finitely Repeated Games
- Infinite Horizon Games
Outline

What happens if a game is not played once, but several times or even infinitely often? We consider **repeated games with observed actions**. One of the famous results is Axelrod’s observations on the iterated prisoners dilemma.

- We first consider **finitely repeated** games. They do not behave intuitively, but can be treated with **backward induction**.
- **Infinite-horizon** games are much more intuitive, but require more technical details.
2.1 Examples and Motivation
Often games are not just played once. They are repeated finitely often (until consensus is reached).

Sometimes, infinitely repeated games can be used to define equilibria.

In repeated games, it makes sense to make its choices dependent on the previous game (or the whole history).
Bargaining

We assume two agents $1, 2$, each with a utility function $\mu_i : E \rightarrow \mathbb{R}$. If the agents do not agree on a result $e$ a fixed fallback $e_{\text{fallback}}$ is taken.

**Example 2.1 (Sharing 1 Pound)**

How to share 1 Pound? Let $e_{\text{fallback}} = 0$.

Agent 1 offers $\rho$ ($0 < \rho < 1$). Agent 2 agrees! Such deals are *individually rational* and each one is a *Nash equilibrium*!

How can we change this unwanted outcome? Either we *impose certain axioms* (see Slide 159) or we view it as a game: *strategic bargaining*. 
Re古代 Bargaining

Figure 6: The bargaining game.
Lemma 2.2 (Trivial Bargaining)

Each strategy profile

\[
\mathcal{s}^x : \begin{cases} 
    a_1 \text{ offers } \langle x, 1 - x \rangle, \text{ agrees to } \langle y, 1 - y \rangle \text{ for } y \geq x \\
    a_2 \text{ offers } \langle x, 1 - x \rangle, \text{ agrees to } \langle y, 1 - y \rangle \text{ iff } 1 - y \geq 1 - x 
\end{cases}
\]

is a NE: agreement is reached in the first round.
Strategic Bargaining

Example revisited: Sharing 1 Pound Sterling.

Protocol with finitely many steps: The last offerer just offers $\epsilon$. This should be accepted, so the last offerer gets $1 - \epsilon$ (see also the Centipede game on Slide 94).

This is again unsatisfiable.
Ways out:

1. Consider \textit{infinite} games ($\rightarrow$ Section 3).
2. Keep finite games, but:

   1. Add a discount factor $\delta$: in round $n$, only the $\delta^{n-1}$th part of the original value is available.
   2. Bargaining costs: bargaining is not for free—fees have to be paid.
Backward Induction

Example 2.3 (Fair offers)

There are four agents (ordered by 1, 2, 3 and 4) and 1 Mio Euros to be distributed between them. The protocol is as follows. Agent 1 makes an offer to all agents. Then either a strict (nonstrict) majority agrees and the game is over, or there is no agreement. In this latter case, agent 1 is out (she gets nothing), and the next agent makes an offer to the rest. And so on.

What should Agent 1 offer so that agreement is reached in the first round (assuming all agents act rational)?
2.2 Finitely Repeated Games
Model of finitely repeated games (1)

- A finite \(n\)-person normal form game \(\langle \mathcal{A}, \text{Act}, O, \varrho, \mu \rangle\), where all \(A_i\) are finite, is also called a stage game. It might be played finitely many times (finite horizon, which is known to all agents), or even infinitely many times (infinite horizon).

- We view such a repeated game as an imperfect information game in extensive form (see Figure 7 on Slide 155). Thus the strategy space of the finitely repeated game is defined.

- More formally, a mixed behavioral strategy in the finitely repeated game is a sequence of mappings from the set of all possible period-\(t\) histories to mixed actions \(s_i \in S_i\).

- These strategies cannot depend on the opponents’ randomizing probabilities \(s_{-i}\), but only on the past values of \(a_{-i}\): the probabilities are not known.
Model of finitely repeated games (2)

- The payoff function of the finite horizon game can be defined as the **sum of the payoffs of the stage games in each round**. Better: **average over time periods** or to introduce a **discount factor**: see Definition 2.12 on Slide 178.

- We can now consider NE’s of such a finitely repeated game.

- Note that we defined **player i’s minmax value** $v_i$:

  $$v_i = \min_{s_{-i}} \max_{s_i} \mu_i^{expected}(s_{-i}, s_i)$$

  in Definition 1.28 on Slide 62.
Example 2.4 (Iterated Prisoners Dilemma)

After each round, the players know what the other player played. So a strategy can take past plays into account.

Figure 7: Iterated Prisoners Dilemma.
Lemma 2.5 (NE for iterated prisoners dilemma)

In the finitely repeated version of the prisoners dilemma, the only Nash equilibrium is the subgame perfect equilibrium where both players always defect.
Axelrod’s tournament

- What is the **best** strategy when prisoner’s dilemma is repeatedly played?
- **Tit for tat**: Cooperate in the first step, and then do what the other player did in the previous step.
- It turned out, that **tit for tat** is not only simple and easy to calculate (only the last move is considered) but also extremely powerful.
- Several experiments have shown that.
- Even counter strategies to tit for tat are difficult to find.
Isn’t Lemma 2.5 contradicting Axelrod’s result?

What if the finite horizon is not known to the players, so the game could end at any round?
Axioms on Bargaining

We consider again Example 2.1 on Slide 146 and state the following axioms on the global solution
\[ \mu^* = \langle \mu_1(e^*), \mu_2(e^*) \rangle. \]

**Invariance:** Absolute values of the utility functions do not matter, only relative values.

**Symmetry:** Changing the agents does not influence the solution.

**Irrelevant Alternatives:** If \( E \) is made smaller but \( e^* \) still remains, then \( e^* \) remains the solution.

**Pareto:** The players can not get a higher utility than
\[ \mu^* = \langle \mu_1(e^*), \mu_2(e^*) \rangle. \]
Theorem 2.6 (Unique solution)

The four axioms above determine a unique solution. This solution is given by

\[ e^* = \arg \max_e \left\{ (\mu_1(e) - \mu_1(e_{\text{fallback}})) \times (\mu_2(e) - \mu_2(e_{\text{fallback}})) \right\}. \]
Strategic bargaining for finite games

Suppose $\delta = 0.9$. Then the outcome depends on the number of rounds.

<table>
<thead>
<tr>
<th>Round</th>
<th>1’s share</th>
<th>2’s share</th>
<th>Total value</th>
<th>Offerer</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n-3$</td>
<td>0.819</td>
<td>0.181</td>
<td>$0.9^{n-4}$</td>
<td>2</td>
</tr>
<tr>
<td>$n-2$</td>
<td>0.91</td>
<td>0.09</td>
<td>$0.9^{n-3}$</td>
<td>1</td>
</tr>
<tr>
<td>$n-1$</td>
<td>0.9</td>
<td>0.1</td>
<td>$0.9^{n-2}$</td>
<td>2</td>
</tr>
<tr>
<td>$n$</td>
<td>1</td>
<td>0</td>
<td>$0.9^{n-1}$</td>
<td>1</td>
</tr>
</tbody>
</table>
Bargaining Costs

Agent 1 pays $c_1$, agent 2 pays $c_2$.

**Protocol:** After each round, the roles change and the fee is subtracted ($c_1$ from agent 1, $c_2$ from agent 2). Therefore the game is finite.

**Theorem 2.7**

1. $c_1 = c_2$: Any split is a SPE.
2. $c_1 < c_2$: Only one SPE: Agent 1 gets all.
3. $c_1 > c_2$: Only one SPE: Agent 1 gets $c_2$, agent 2 gets $1 - c_2$. 
Proof.

(1) is obvious.

(2) Assume agent 2 offered \( \pi \) in round \( t \). Then in round \( t - 1 \) agent 1 had offered \( 1 - (\pi + c_2) \). Thus in round \( t - 2 \) agent 2 would have offered \( \pi + c_2 - c_1 \) and kept \( 1 - \pi - (c_2 - c_1) \). So in round \( t - 2k \), agent 2 would keep \( 1 - \pi - k(c_2 - c_1) \). But this would go to \(-\infty\), so agent 2 accepts 0 upfront.

(3) This follows from (2): After the first round, agent 2 is in the role of agent 1. But to reach the second round, agent 2 would have to pay \( c_2 \). So agent 2 is willing to pay agent 1 its (i.e. agent 2’s) fees. So agreement is reached in the first round (no bargaining fees).
Example 2.8 (Bargaining)

Two players, 1 and 2, bargain about how to split goods worth initially $w_0 = 1$ EUR. After each round without agreement, the worth of the goods reduces by discount rates $\delta_1$ (for player $a_1$) and $\delta_2$ (for player $a_2$). So, after $t$ rounds, the goods are worth only $\langle \delta_t^1, \delta_t^2 \rangle$. Subsequently, $a_1$ (if $t$ is even) or $a_2$ (if $t$ is odd) makes an offer to split the goods in proportions $\langle x, 1 - x \rangle$, and the other player accepts or rejects it. If the offer is accepted, then $a_1$ takes $x\delta_t^1$, and $a_2$ gets $(1 - x)\delta_t^2$; otherwise the game continues.
Finite Set of payoffs: Making the game finite.

- In order to obtain a finite set of payoffs, we assume that the goods are split with finite precision represented by a rounding function $r : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

- We assume from the rounding function $r$ that there is $\epsilon \geq 0$ such that the following holds: (1) $r(x) \leq x$, (2) $r(x) > x - \epsilon$, (3) $r$ is monotonically increasing, and (4) $r(\epsilon) = 0$.

- So, after $t$ rounds, the goods are in fact worth $\langle r(\delta_1^t), r(\delta_2^t) \rangle$, and if the offer is accepted, then $a_1$ takes $r(x\delta_1^t)$, and $a_2$ gets $r((1 - x)\delta_2^t)$. 
**Problem:** Nash equilibria in sequential games might exist in the first stages, but not later.

**Solution:** We consider SPE's: Nash equilibria that remain Nash equilibria in every possible subgame, see Definition 1.40 on Slide 91.
Bargaining Revisited

What about SPE’s in Example 2.8?
Because of the finite precision, there is a minimal round \( T \) with \( r(\delta_i^{T+1}) = 0 \) for \( i = 1 \) or \( i = 2 \). For simplicity, assume that \( i = 2 \) and agent 1 is the offerer in \( T \) (i.e., \( T \) is even).

Lemma 2.9 (Bargaining made finite)

For bargaining with a rounding function, there exists exactly one SPE. The goods are split \( \langle \kappa, 1 - \kappa \rangle \) and agreement is reached in the first round:

\[
\kappa = (1 - \delta_2) \frac{1 - (\delta_1 \delta_2) \frac{T}{2}}{1 - \delta_1 \delta_2} + (\delta_1 \delta_2) \frac{T}{2}.
\]

We assume also that the \( \epsilon \) in the rounding function satisfies \( \epsilon < |\delta_1 - \delta_2| \).
Theorem 2.10 (Benoit and Krishna, 1987)

Suppose that for each player $i$ there is a static equilibrium in which $i$ has a payoff that is strictly greater than its minmax value $\min_{s_{-i}} \max_{s_i} \mu_i^{\text{expected}}(s_{-i}, s_i)$. We are interested in the set of NE payoffs of the $T$-period game (averaged over $T$). What happens with this set, when $T$ goes to $\infty$?

The sets converge to the set of feasible, individually rational payoffs.

Note that we have defined and discussed the minmax-value and feasible payoffs on Slide 62. The theorem talks about payoffs, not about NE’s.
Proof.

- Clearly, if all players play their minmax strategies, they get their minmax value.
- In the following we construct strategy profiles where each agent gets \textit{strictly more} than its minmax value. By playing such a cycle many times, we can get any feasible payoff.
- In the first phase, agents play suboptimal, but in the terminal phase, they get their rewards.
- If one agent \textit{deviates}, then the others punish her by \textit{minmaxing} her for the rest of the game.
- In the second phase, the \textit{reward phase}, each agent gets strictly more than its minmax value for many rounds (this ensures that each feasible IR payoff will be reached).
Proof (cont.)

- Let $v$ be a feasible, strictly IR payoff. We want to find equilibria and payoffs that are “close” to $v$ (within $\epsilon$ range).

- Let $\alpha^*(i)$ be a static equilibrium for agent $i$ in which its payoff is strictly greater than its minmax value. We construct two phases. In the first phase all players will play suboptimal. The all get the reward for that in the second phase, the $R$-cycle.

- We construct first the reward phase where first player 1’s $\alpha^*(1)$ is played, then player 2’s $\alpha^*(2)$ etc. This cycle is repeated itself $R$-many times, where $R \in \mathbb{N}$ is arbitrary (to be adjusted later). We call it the $R$-cycle.

- Any terminal $R$-cycle is a NE path in any subgame of length $R \times n$. The average payoff for each player is strictly better than its minmax value in each such $R$-cycle.
Proof (cont.)

- How can we ensure, that a player is \textit{cooperating}, i.e. accepting a non-optimal payoff, namely the minmax value \( v_i \), in the first phase?

- If a player deviates, she is getting \textit{in that round} a better payoff, but will be punished by all other players: they simply \textit{minmax} her in all remaining rounds (and cooperate among them).

- So we choose \( R \) large enough, so that accepting just \( v_i \) in some rounds plus the terminal \( R \)-cycle gives a better payoff than \textit{deviating and being minmaxed} for the rest of the game.
Proof (cont.)

- We are now designing the first phase. Given $\epsilon \geq 0$, we choose $T$ large enough, so that there exist sets of pure actions $\{a_1(i), \ldots, a_{j_i}(i)\}$ for each player $i$ with the following property: the phase of length $T - R \times n$, where all agents play deterministically these actions, has average payoffs that are within $\epsilon$-range of $v$.

- Now we are ready to play both phases. Our strategy is to deterministically play, as long as there are still more than $R \times n$ rounds left, the pure actions, then the terminal $R$-cycle.
Proof (cont.)

Deviating in this first phase does not pay off (minmaxing the deviating player). So these strategies are a NE as long as \( T \geq R \times n \).

The average payoffs are within \( 2\epsilon \)-range of \( v \) for

\[
T \geq R \times n \times \frac{\max_{\vec{a}} \mu_i(\vec{a}) - v_i}{\epsilon}.
\]
2.3 Infinite Horizon Games
Strategic bargaining for infinite games

We introduce $\delta_1$ factor for agent 1, $\delta_2$ factor for agent 2.

Theorem 2.11 (Unique solution for infinite games)

In a discounted infinite round setting, there exists a unique subgame perfect Nash equilibrium:

1. Agent 1 gets $\frac{1-\delta_2}{1-\delta_1\delta_2}$.
2. Agent 2 gets the rest.
3. Agreement is reached in the first round.
Proof.

<table>
<thead>
<tr>
<th>Round</th>
<th>1’s share</th>
<th>2’s share</th>
<th>offerer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>$t - 2$</td>
<td>$1 - \delta_2(1 - \delta_1\bar{\pi}_1)$</td>
<td>:</td>
<td>1</td>
</tr>
<tr>
<td>$t - 1$</td>
<td>:</td>
<td>$1 - \delta_1\bar{\pi}_1$</td>
<td>2</td>
</tr>
<tr>
<td>$t$</td>
<td>$\bar{\pi}_1$</td>
<td>:</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

Let $\bar{\pi}_1$ the **maximal undiscounted share** that agent 1 can possibly get in any subgame perfect Nash equilibrium when she is offering. We can now get back two rounds and get

$$1 - \delta_2(1 - \delta_1\bar{\pi}_1).$$

Setting both equal gives us the result. $\square$
The last theorem is about infinite games.

On Slide 165, we have introduced a finite precision rounding function to get a finite set of payoffs.

Obviously, this assumption makes the entire game finite.

So we get an analogue of Theorem 2.11 for finite games.
Model of infinitely repeated games (1)

How to define an infinitely repeated game? Again, we consider a fixed stage game \( \langle A, \text{Act}, O, \varrho, \mu \rangle \).

**Definition 2.12 (Average and discounted reward)**

Let \( r^{(1)}_i, r^{(2)}_i, \ldots \) denote an infinite sequence of payoffs for player \( i \) and let \( \delta \) be a discount factor \( 0 \leq \delta \leq 1 \).

We define the *average reward* of player \( i \) as

\[
\limsup_{k \to \infty} \frac{1}{k} \sum_{j=1}^{k} r^{(j)}_i.
\]

We define the *future discounted reward* of player \( i \) as

\[
\sum_{j=1}^{\infty} \delta^j r^{(j)}_i.
\]
Model of infinitely repeated games (2)

In the following, we consider the discount-factor version. Given a fixed stage game $\langle A, Act, O, \varrho, \mu \rangle$ and a discount factor $\delta \leq 1$.

- In each period $t$ certain actions $a^t_1, \ldots, a^t_{l_t}$ are played: denoted by $a^t$.
- $h^t = (a^0, \ldots, a^{t-1})$ is the sequence of all realized actions until $t$.
- $H^t$ is the space of all possible period-$t$ histories.
- **Pure strategy of $i$ in the infinite game:** $a_i$ is a sequence of mappings $a^t_i$ (for each period $t$) from $H^t$ into $A_i$. In each period $t$ a pure strategy $a_i$ is chosen.
- **Mixed (behavioral) strategy in the infinite game:** $s_i$ is a sequence of mappings $s^t_i$ (for each period $t$) from $H^t$ into mixed actions $S_i$. In each period $t$ a mixed strategy $s_i$ is chosen.
Model of infinitely repeated games (3)

- We also use $\vec{a}$ and $\vec{s}$ for the pure action profile, resp. the mixed strategy profile. Similarly we use $\vec{a}^t$ and $\vec{s}^t$ for the $t$ period.
- The only proper subgames are the period-$t$ games.
- What is player $i$ trying to maximize? Its discounted payoff.
- But this depends on the possible histories $H^t$.
- And these histories depend on the strategy profile $\vec{s}$.
- So we need to know something about how the infinite histories, depending on $\vec{s}$, are distributed. So the expectation $E_{\vec{s}}$ will play a role.
Model of infinitely repeated games (4)

Definition 2.13 (The game $G(\delta)$)

Player $i$’s intention is to maximize the value

$$\mu_i^{\text{repeated}} = E_{\bar{s}} (1 - \delta) \sum_{t=0}^{\infty} \delta^t \mu_i(s_t(h_t))$$

- Why the factor $(1 - \delta)$?
- Because $\sum_{i=0}^{\infty} \delta^i = \frac{1}{1-\delta} \leadsto \text{normalization.}$
Model of infinitely repeated games (5)

We have defined the payoff of an infinitely repeated game. Therefore we can now reason about the strategy spaces of repeated games.

- Let $\alpha^*$ be a NE of the stage game (above we called this static equilibrium).
- Then the strategies “each player $i$ plays $\alpha_i^*$ from now on” are a subgame-perfect equilibrium.
- If there are $m$ static equilibria of the stage game, then any mapping from time periods $t$ to any such NE (meaning that this NE is played at time $t$) leads to a subgame-perfect equilibrium as well.
- Why not playing a static best response at all?
  $\leadsto$ the future!!
In Definition 1.28 on Slide 62 we defined the notion of minmax value and discussed it for the stage game. Its importance for repeated games is because of the following

**Lemma 2.14 (Minmax value is guaranteed)**

*In the infinitely repeated game $G(\delta)$, player $i$’s payoff is at least its minmax value in all Nash equilibria (regardless of the discount factor $\delta$).*

Feasible payoffs are convex outcomes of the stage game.
Question: Suppose we have a better payoff profile \( \langle r_1, \ldots, r_i, \ldots, r_n \rangle \) (i.e. all \( r_i \) are strictly greater than their minmax values) for a stage game \( \langle A, Act, O, \varrho, \mu \rangle \).

Is there an infinitely repeated game with a Nash equilibrium that has exactly this payoff profile?

The answer is yes, if the profile is feasible.

Any feasible, individually rational payoff can be obtained (if one is patient enough).
Folk Theorem for infinitely repeated games

**Theorem 2.15 (Folk Theorem)**

For every feasible payoff profile \( \vec{r} \) that is strictly individually rational for each player, there exists a \( \delta' \leq 1 \) such that for all \( \delta \in ]\delta', 1[ \) there is a Nash equilibrium of the infinitely repeated game \( G(\delta) \) with payoff profile \( \vec{r} \).
The last theorem does not speak about strategy profiles characterizing Nash equilibria. It only talks about the payoffs.

It says that the payoffs obtained in the infinitely repeated game are exactly those that can be obtained in the stage game using mixed strategies.
Proof.

- We have already seen in Lemma 2.14 that each player gets at least its minmax value,
- We construct an equilibrium that meets the given payoff profile by cycling the games accordingly.
- Any agent deviating (getting perhaps a better outcome in the current game) will be punished for this behavior by the other agents (which will play their minmax startegies against him.)

This is exactly mimicking the proof of Theorem 2.10 given on Slides 169 pp.
Proof (cont.)

- For simplicity, we assume that such a payoff profile $\vec{r}$ can be obtained by a pure action profile $\vec{a}$. If this is not the case, we take a public randomization with expected value $\vec{r}$. The rest of the proof can be easily adapted.

- Strategy for player $i$ is as follows.
  - Play $a_i$ in the beginning and continue until
  - either $\vec{a}$ was played in the last round,
  - or the profile played in the last round differed from $\vec{a}$ in at least two components.
  - If in some previous round, $i$ was the only deviating from $\vec{a}$, then each player plays $m^i_j$ for the remaining part of the game.
Proof (cont.)

Is it possible for player $i$ to **deviate** and gain a higher payoff?

- She can indeed gain a bit more in one round, namely $\max_a \mu_i(\bar{a})$.
- But because all others will minmax her for the rest of the game, she obtains only her minmax value (forever).
- Therefore, if $i$ deviates in round $t$, then she obtains at most

\[
(1 - \delta^t)v_i + \delta^t(1 - \delta) \max_{\bar{a}} \mu_i(\bar{a}) + \delta^{t+1}v_i
\]

- which is less than $v_i$ for $\delta \geq \delta_i$,
- where $\delta_i$ is defined by $(1 - \delta_i) \max_a \mu_i(\bar{a}) + \delta_i v_i = v_i$. ($\delta_i$ is smaller than 1 because $v_i \geq \underline{v}$.)
Is the NE constructed in the last proof subgame perfect?

No!

Punishments might also be costly (for the punishers).

Repeated quantity setting oligopoly: Too much output $\implies$ price drop.

Prices might be below punished players average costs, even below own costs (of punishers).
Perfect Folk Theorem for infinitely repeated games

**Theorem 2.16 (Perfect Folk Theorem (Fudenberg/Maskin 1986))**

*We assume that the dimension of the set $V$ of feasible payoffs equals the number of players. For every feasible payoff profile $\vec{r}$ that is strictly individually rational for each player, for any $v \in V$ with $v_i \geq v_{i'}$ for all agents $i$, there exists a $\delta' \leq 1$ such that for all $\delta \in ]\delta', 1[$ there is a subgame perfect Nash equilibrium of the infinitely repeated game $G(\delta)$ with payoff profile $\vec{r}$.***
3 Coalitional Games

<table>
<thead>
<tr>
<th>3</th>
<th>Coalitional Games</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coalition Formation in CFG’s</td>
</tr>
<tr>
<td></td>
<td>Classes of Games</td>
</tr>
<tr>
<td></td>
<td>The Core and its refinements</td>
</tr>
<tr>
<td></td>
<td>Payoff Division: Shapley value and Banzhaf Index</td>
</tr>
</tbody>
</table>
What happens if agents decide to team up and work together to solve a problem more efficiently? We consider

- abstract coalition formation for characteristic function games (CFG);
- algorithms for searching the coalition structure graph; and
- how to distribute the profit among the agents: core of a CFG and Shapley value.
3.1 Coalition Formation in CFG’s
Definition 3.1 (Characteristic Function Game (CFG))

A characteristic function game is a tuple \( \langle A, v \rangle \) where \( A \) is a finite set (of agents) and \( v : 2^A \to \mathbb{R}_0^+ ; S \mapsto v(S) \).

We assume \( v(\emptyset) = 0 \) and call \( v(S) \) the value of coalition \( S \).

Thus the value is independent of the nonmembers. But

1. **Positive Externalities**: Overlapping goals. Nonmembers perform actions and move the world closer to the coalition’s goal state.

2. **Negative Externalities**: Shared resources. Nonmembers may use up the resources.
Definition 3.2 (Coalition Structure CS)

A coalition structure $CS$ over the set $A$ is any partition \( \{C^1, \ldots, C^k\} \) of $A$, i.e. \( \bigcup_{j=1}^{k} C^j = A \) and $C^i \cap C^j = \emptyset$ for $i \neq j$.

We denote by $CS_M$ the set of all coalition structures $CS$ over the set $M \subseteq A$.

Finally, we define the social welfare of a coalition structure $CS$ by

$$v(CS) := \sum_{C \in CS} v(C).$$
Definition 3.3 (Coalition Formation in CFG’s)

**Coalition Formation** in CFG’s consists of:

**Forming CS:** formation of coalitions such that within each coalition agents coordinate their activities.

**Solving Optimisation Problem:** For each coalition in a CS the tasks and resources of the agents have to be pooled. *Maximise monetary value.*

**Payoff Division:** Divide the value of the generated solution among agents.
Maximising Social Welfare

Maximise the social welfare of the agents $A$ by finding a coalition structure

$$CS^* = \arg \max_{CS \in CS_A} v(CS),$$

where

$$v(CS) := \sum_{S \in CS} v(S).$$

How many coalition structures are there?
A lot: $\Omega\left(\frac{|A|^2}{2}\right)$. Enumerating is feasible for $|A| < 15$.

**Figure 8:** Number of Coalition (Structures).
Approximation of $\nu(\text{CS})$.

How can we approximate $\nu(\text{CS})$?

Choose set $N \subseteq \text{CS}_A$ and pick the best coalition seen so far:

$$\text{CS}_N^* = \arg \max_{\text{CS} \in N} \nu(\text{CS}).$$
Figure 9: Coalition Structure Graph $\mathcal{CS}_A$. 
We want our approximation as good as possible.

We want to find a small \( k \) and a small \( N \) such that

\[
\frac{v(CS^*)}{v(CS_N^*)} \leq k.
\]

\( k \) is the **bound** (best value would be 1) and \( N \) is the **part of the graph** that we have to search exhaustively.
We consider 3 search algorithms:

**MERGE:** Breadth-first search from the top.

**SPLIT:** Breadth first from the bottom.

**Coalition-Structure-Search (CSS1):** First the bottom 2 levels are searched, then a breadth-first search from the top.

MERGE might not even get a bound, without looking at all coalitions.

SPLIT gets a good bound ($k = |A|$) after searching the bottom 2 levels (see below). But then it can get slow.

CSS1 combines the good features of MERGE and SPLIT.
Theorem 3.4 (Minimal Search to get a bound)

To bound $k$, it suffices to search the lowest two levels of the CS-graph. Using this search, the bound $k = |A|$ can be taken. This bound is tight and the number of nodes searched is $2^{|A|} - 1$.

No other search algorithm can establish the bound $|A|$ while searching through less than $2^{|A|} - 1$ nodes.
Proof.

There are at most $|\mathcal{A}|$ coalitions included in $\mathcal{CS}^*$. Thus

$$v(\mathcal{CS}^*) \leq |\mathcal{A}| \max_S v(S) \leq |\mathcal{A}| \max_{\mathcal{CS} \in \mathcal{N}} v(\mathcal{CS}) = |\mathcal{A}|v(\mathcal{CS}^*_N)$$

Number of coalitions at the second lowest level: $2^\mathcal{A} - 2$.

Number of coalition structures at the second lowest level:

$$\frac{1}{2}(2^\mathcal{A} - 2) = 2^{\mathcal{A}-1} - 1.$$ 

Thus the number of nodes visited is: $2^{\mathcal{A}-1}$. 

$\Box$
What exactly does the last theorem mean? Let $n_{min}$ be the smallest size of $N$ such that a bound $k$ can be established.

Positive result: $\frac{n_{min}}{\text{partitions of } A}$ approaches 0 for $|A| \rightarrow \infty$.

Negative result: To determine a bound $k$, one needs to search through exponentially many coalition structures.
Algorithm (CS-Search-1)

The algorithm comes in 3 steps:

1. Search the bottom two levels of the CS-graph.
2. Do a breadth-first search from the top of the graph.
3. Return the CS with the highest value.

This is an anytime algorithm.
Theorem 3.5 (CS-Search-1 up to Layer l)

With the algorithm CS-Search-1 we get the following bound for $k$ after searching through layer $l$:

$$\begin{cases} \left\lceil \frac{|\mathcal{A}|}{h} \right\rceil & \text{if } |\mathcal{A}| \equiv h - 1 \mod h \text{ and } |\mathcal{A}| \equiv l \mod 2, \\ \left\lfloor \frac{|\mathcal{A}|}{h} \right\rfloor & \text{otherwise.} \end{cases}$$

where $h = \text{def } \left\lfloor \frac{|\mathcal{A}| - l}{2} \right\rfloor + 2$.

Thus, for $l = |\mathcal{A}|$ (check the top node), $k$ switches from $|\mathcal{A}|$ to $\frac{|\mathcal{A}|}{2}$.
Experiments

6-10 agents, values were assigned to each coalition using the following alternatives

1. values were uniformly distributed between 0 and 1;
2. values were uniformly distributed between 0 and $|\mathcal{A}|$;
3. values were superadditive;
4. values were subadditive.
7 agents

Figure 10: MERGE.

Figure 10: MERGE.
3.1 Coalition Formation in CFG’s

Figure 11: SPLIT.
Figure 12: CS-Search-1.
Figure 13: Subadditive Values.
Figure 14: Superadditive Values.
Figure 15: Coalition values chosen uniformly from \([0, 1]\).
Figure 16: Coalition values chosen uniformly from $[0, |S'|]$. 
Figure 17: Comparing CS-Search-1 with SPLIT.
1 Is CS-Search-1 the **best anytime algorithm**?

2 The search for best $k$ for $n' > n$ is perhaps not the same search to get best $k$ for $n$.

3 **CS-Search-1** does not use any information while searching. Perhaps $k$ can be made smaller by not only considering $v(CS)$ but also $v(S')$ in the searched CS$'$.
3.2 Classes of Games
Idea: Consider a protocol (to build coalitions) as a game and consider Nash-equilibrium.

Problem: Nash-Eq is too weak!

Definition 3.6 (Strong Nash Equilibrium)

A profile is in strong Nash-Eq if there is no subgroup that can deviate by changing strategies jointly in a manner that increases the payoff of all its members, given that nonmembers stick to their original choice.

This is often too strong and does not exist.
Definition 3.7 (Monotone Games)

A CFG $\langle A, v \rangle$ is called monotone, if

$$v(C) \leq v(D),$$

for every pair of coalitions $C, D \subseteq A$ such that $C \subseteq D$.

Many games have this property, but there may be communication/coordination costs. Or some players hate others and do not want to be in the same coalition. The next slide introduces a strictly stronger condition.
**Definition 3.8 (Superadditive Games)**

A CFG $\langle A, v \rangle$ is called **superadditive**, if

$$v(S \cup T) \geq v(S) + v(T),$$

where $S, T \subseteq A$ and $S \cap T = \emptyset$.

**Lemma 3.9**

Coalition formation for superadditive games is trivial.

**Conjecture**

All games are superadditive.
The conjecture is wrong, because the **coalition process** is not for free: **communication costs, penalties, time limits**.

**Definition 3.10 (Subadditive Games)**

A CFG $\langle \mathcal{A}, \nu \rangle$ is called **subadditive**, if

$$\nu(S \cup T) \not\leq \nu(S) + \nu(T),$$

where $S, T \subseteq \mathcal{A}$ and $S \cap T = \emptyset$.

**Coalition formation for subadditive games is trivial.**
**Definition 3.11 (Superadditive Cover)**

Given a game $G = \langle \mathcal{A}, v \rangle$ that is not superadditive, we can transform it to a superadditive game $G^* = \langle \mathcal{A}, v^* \rangle$ as follows

$$v^*(C) := \max_{CS \in CS_C} v(CS)$$

This game is called the **superadditive cover** of $G$. 

Convex Games

Definition 3.12 (Convex Game)

A CFG $\langle A, v \rangle$ is **convex**, if for all coalitions $T, S$ with $T \subseteq S$ and each player $i \in A \setminus S$:

$$v(S \cup \{i\}) - v(S) \geq v(T \cup \{i\}) - v(T)$$

- Convex games are superadditive.
- Superadditive games are monotone.
- The other directions do not hold.
Example 3.13 (Treasure of Sierra Madre Game)

There are $n$ people finding a treasure of many gold pieces in the Sierra Madre. Each piece can be carried by two people, not by a single person.

Example 3.14 ($3$-player majority Game)

There are three people that need to agree on something. If they all agree, there is a payoff of $1$. If just $2$ agree, they get a payoff of $\alpha$ ($0 \leq \alpha \leq 1$). The third player gets nothing.

How do the $v(S)$ look like?
Example 3.15 (Parliament)

Suppose there are four parties and the result of the elections is as follows:

1. $A$: 45 %,
2. $B$: 25 %,
3. $C$: 15 %,
4. $D$: 15 %.

We assume there is a 100 Mio Euro spending bill, to be controlled by (and distributed among) the parties that win.

**Version 1:** Simple majority wins ($\geq 50\%$).
**Version 2:** Any majority over 80% wins.
Definition 3.16 (Payoff Vector)

A payoff vector for a CFG and a coalition structure $CS$ is a tuple $\langle x_1, \ldots, x_n \rangle$ such that

1. $x_i \geq 0$ and $\sum_{i=1}^{\mid A \mid} x_i = v(A)$,
2. $\forall C \in CS : \sum_{i \in C} x_i \geq v(C)$.

Note that the last condition is only supposed to hold for all coalitions in the given coalition structure.
3.3 The Core and its refinements
Consider a CFG that is not necessarily superadditive (so the grand coalition does not necessarily form). Assume that a certain coalition structure $\mathcal{CS}$ forms.

**Definition 3.17 (Core of a CFG)**

The *core of a CFG* is the set of all pairs $\langle \mathcal{CS}, \langle x_1, \ldots, x_n \rangle \rangle$ of coalition structures ($\mathcal{CS} \in \mathcal{CS}_A$) and payoff vectors such that the following holds:

$$\forall S \subseteq A : \sum_{i \in S} x_i \geq v(S)$$

Here, the condition is supposed to hold for all $S$. We do not want any set of agents to form a new coalition. It ensures that only the grand coalition forms.
When the grand coalition forms, we can simplify the last definition.

**Definition 3.18 (Core of Superadditive Games)**

The *core of a superadditive CFG* is the set of all payoff vectors \( \langle x_1, \ldots, x_n \rangle \) such that the following holds:

\[
\forall S \subseteq A : \sum_{i \in S} x_i \geq v(S)
\]

Thus the *core* corresponds to the *strong Nash equilibrium* mentioned in the beginning.

**What about the core in the above examples?**
Lemma 3.19

If $\langle CS, \langle x_1, \ldots, x_n \rangle \rangle$ is in the core of a CFG $\langle A, v \rangle$, then $v(CS) \geq v(CS')$ for all coalition structures $CS' \in CS_A$.

Proof.

We can write $v(CS) = \sum_{i \in A} x_i = \sum_{C' \in CS'} x(C')$ and $v(CS') = \sum_{C' \in CS'} v(C')$.

Because of the definition of the core, $x(C') \geq v(C')$ for all $C'$ and therefore $v(CS) \geq v(C')$. 

$\square$
Theorem 3.20

Let a CFG $G = \langle A, v \rangle$ be given (not necessarily superadditive). Then $G$ has a non-empty core if and only if its superadditive cover $G^*$ has a non-empty core.

Proof $\Leftarrow$ exercise
Theorem 3.21

Each convex CFG $G = \langle A, v \rangle$ has a non-empty core.

Proof.

Let $\pi$ be a permutation of $A$ and let $S_\pi(i)$ be the set of all predecessors of $i$ wrt. $\pi$.

We claim that for $x_i := v(S_\pi(i) \cup \{i\}) - v(S_\pi(i))$, the core of $G$ contains $\langle x_1, \ldots, x_n \rangle$.

It is easy to show that all $x_i$ are greater or equal to 0 and that they all sum up to the value of the game ($\Rightarrow$ exercise).
(Proof of Theorem 3.21, cont.)

Assume there is a coalition \( C = \{i_1, \ldots, i_s\} \) such that \( v(C) \geq x(C) \). Wlog we assume \( \pi(i_1) \leq \ldots \leq \pi(i_s) \). Obviously

\[
v(C) = v(\{i_1\}) - v(\emptyset) + v(\{i_1, i_2\}) - v(\{i_1\}) + \ldots + v(C) - v(C \setminus \{i_s\})
\]

Because of convexity (apply convexity to \( T_j := \{i_1, \ldots, i_{j-1}\} \) and \( S_j := \{1, 2, \ldots, i_j - 1\} \)) for all \( j \):

\[
v(T_j \cup \{i_j\}) - v(T_j) \leq v(S_j \cup \{i_j\}) - v(S_j) = x_{i_j}
\]

Adding these pairs up, we get \( v(C) \leq x(C) \), which is a contradiction.
3.4 Payoff Division: Shapley value and Banzhaf Index
We now assume that the grand coalition forms. The payoff division should be fair between the agents, otherwise they would leave the coalition.

**Definition 3.22 (Dummies, Interchangeable)**

Agent $i$ is called a *dummy*, if for all coalitions $S$ with $i \not\in S$:

$$v(S \cup \{i\}) - v(S) = v(\{i\}).$$

Agents $i$ and $j$ are called *interchangeable*, if for all coalitions $S$ with $i \in S$ and $j \not\in S$:

$$v(S \setminus \{i\} \cup \{j\}) = v(S).$$
Marginal Contribution

The marginality axiom, introduced by Young in the 80’ies, concentrates on the marginal contributions of a player in two different games.

**Definition 3.23 (Marginal Contribution in two games)**

We consider two CFG games over the same coalition structures, with values $v$ and $w$. We say that agent $i$ is marginally indifferent between $v$ and $w$, if its marginal contributions in all coalitions is the same in both games: for all $S \subseteq A \setminus \{i\}$

$$v(S \cup \{i\}) - v(S) = w(S \cup \{i\}) - w(S).$$
Axioms for Payoff Division

**Efficiency:** \( \sum_{i \in A} x_i = v(A) \).

**Symmetry:** If \( i \) and \( j \) are interchangeable, then \( x_i = x_j \).

**Dummies:** For all dummies \( i \): \( x_i = v(\{i\}) \).

**Additivity:** For any two games \( v, w \):

\[
x_i^{v \oplus w} = x_i^v + x_i^w,
\]

where \( v \oplus w \) denotes the game defined by

\[
(v \oplus w)(S) = v(S) + w(S).
\]

**Marginality:** If an agent \( i \) is marginally indifferent between two games \( v, w \), then it should get the same payoff in both of them:

\[
x_i^v = x_i^w.
\]
Theorem 3.24 (Shapley-Value: 1st Characterisation)

For a CFG $G = \langle \mathcal{A}, v \rangle$ there is only one payoff division satisfying the first four axioms. It is called the Shapley value of agent $i$ and is defined by

$$
\phi_i(G) = \frac{1}{|\mathcal{A}|!} \sum_{S \subseteq \mathcal{A} \setminus \{i\}} (|\mathcal{A}| - |S| - 1)!|S|!(v(S \cup \{i\}) - v(S))
$$

Theorem 3.25 (Shapley-Value: 2nd Characterisation)

For a CFG $G = \langle \mathcal{A}, v \rangle$ there is only one payoff division satisfying the efficiency, symmetry and marginality. It is the Shapley value.
$(|\mathcal{A}| - |S|)!$ is the number of all possible joining orders of the agents (to form a coalition).

$(v(S \cup \{i\}) - v(S))$ is $i$’s marginal contribution when added to set $S$.

There are $|S|!$ ways for $S$ to be built before $i$’s joining. There are $(|\mathcal{A}| - |S| - 1)!$ ways for the remaining agents to form $S$ (after $i$).

The Shapley value sums up the marginal contributions of agent $i$ averaged over all joining orders.

An expected gain can be computed by taking a random joining order and computing the Shapley value.
We have shown in Theorem 3.21 that convex games have non-empty cores. In fact, we can show a stronger statement.

**Theorem 3.26**

*In each convex game, at least the Shapley value is contained in the core.*
**Definition 3.27 (Banzhaf Index)**

For a CFG $G = \langle A, v \rangle$ the Banzhaf Index of agent $i$ is

$$
\beta_i(G) = \frac{1}{2|A|-1} \sum_{S \subseteq A \setminus \{i\}} \left( v(S \cup \{i\}) - v(S) \right)
$$

The Banzhaf Index satisfies all axioms but efficiency. The following normalised Banzhaf index $\eta_i(G)$ is also often considered:

$$
\eta_i(G) := \frac{\beta_i(G)}{\sum_{i \in A} \beta_i(G)} v(A).
$$
4. Social Choice and Auctions

- Classical Voting Systems
- Formal model for social choice
- Social Choice Functions
- Social Choice Correspondences
- Social Welfare Functions
- Results based on partial orders
- Auctions
We deal with voting systems and discuss

- some classical approaches;
- an abstract framework to describe arbitrary voting mechanisms: social choice theory, and
- Arrow’s theorem and some variants thereof in this framework.

We also consider auctions (deals between two agents). They constitute one of the most important frameworks for resource allocation problems between selfish agents. They can be seen as an important application of mechanism design, dealt with in Chapter 6.
4.1 Classical Voting Systems
### Voting procedure

Agents give input to a **mechanism**: The outcome is taken as a **solution** for the agents.

#### Non-ranking voting:
Each agent **votes for exactly one candidate**. Winners are those with a majority of votes.

#### Approval voting:
Each agent can **cast a vote for as many candidates** as she wishes (at most one for each candidate). Winners are those with the highest number of approval votes.

#### Ranking voting:
Each agent expresses his **full preference over the candidates**. Computing the winning can be complicated.
From now on we assume there is a fixed set of alternatives (candidates, outcomes) $O$, in addition to the set of agents $A$, the elements of which we call now voters.

**Definition 4.1 (Beat and tie)**

We say that a candidate $o$ **beats** another candidate $o'$ (in direct comparison) if the number of voters that **strictly prefer** $o$ to $o'$ is **strictly greater** than the number of voters that **strictly prefer** $o'$ to $o$.

If both numbers are equal, we say that $o$ **ties** $o'$ (in direct comparison).

We note that the strict preference is not important: $o$ beats $o'$ iff the number of voters that **prefer** $o$ to $o'$ (or are indifferent) is **strictly greater** than the number of voters that **prefer** $o'$ to $o$ (or are indifferent).
Definition 4.2 (Condorcet-winner, -Set)

1. A candidate \( o \) is a **Condorcet winner** if \( o \) beats any other candidate \( o' \) (\( o' \neq o \)).

2. A candidate \( o \) is a **weak Condorcet winner** if \( o \) beats or ties any other candidate \( o' \) (\( o' \neq o \)).

3. The **Condorcet set** is the set of weak Condorcet winners.

Note that sometimes the Condorcet winner is called **strict Condorcet winner**.
### Figure 18: A Tie, but Condorcet helps.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w_1)</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>(w_2)</td>
<td>B</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>(w_3)</td>
<td>C</td>
<td>B</td>
<td>A</td>
</tr>
</tbody>
</table>
Comparing A and B: majority for A.
Comparing A and C: majority for C.
Comparing B and C: majority for B.
Desired Preference ordering: $A > B > C > A$
Another interesting set is the Smith set.

**Definition 4.3 (Smith set)**

The **Smith set** is the smallest, non-empty set $S \subseteq O$ of candidates such that for each candidate $o \in S$ and each candidate $o' \notin S$ the following holds: $o$ beats $o'$.

There is a strong relation between the **Condorcet set** and the **Smith set**. This will be treated in more detail in the exercise class.
We note that we only consider “truthful” approvals. So, \( o \) is strictly preferred over \( o' \) if \( o \) received a vote but \( o' \) did not.

**Lemma 4.4**

In **approval voting**, at least one of the winners is a Condorcet winner.

**Definition 4.5 (Borda Protocol)**

Each voter gives its best candidate \(|O|\) points, the second best gets \(|O| - 1\) points, etc. After all votes have been cast, they are **summed up, across all voters**. Winners are those with the **highest count**.
Winner turns loser and loser turns winner.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$a \succ b \succ c \succ d$</td>
</tr>
<tr>
<td>2</td>
<td>$b \succ c \succ d \succ a$</td>
</tr>
<tr>
<td>3</td>
<td>$c \succ d \succ a \succ b$</td>
</tr>
<tr>
<td>4</td>
<td>$a \succ b \succ c \succ d$</td>
</tr>
<tr>
<td>5</td>
<td>$b \succ c \succ d \succ a$</td>
</tr>
<tr>
<td>6</td>
<td>$c \succ d \succ a \succ b$</td>
</tr>
<tr>
<td>7</td>
<td>$a \succ b \succ c \succ d$</td>
</tr>
</tbody>
</table>

**Borda count**

- $c$ wins: 20, $b$: 19, $a$: 18, $d$ loses: 13

**Borda count without $d$**

- $a$ wins: 15, $b$: 14, $c$ loses: 13
Binary protocol: *Pairwise comparison.*

Take any two candidates and determine the winner. The winner enters the next round, where it is compared with one of the remaining candidates.

Which ordering should we use?
Figure 20: Four different orderings and four alternatives.

Last ordering:

\(d\) wins, but all agents prefer \(c\) over \(d\).
**Coomb’s method:** Each voter ranks all candidates in linear order. If there is no candidate ranked first by a majority of all voters, the candidate which is ranked last (by a majority) is eliminated. The last remaining candidate wins.

**d’Hondt’s method:** Each voter cast his votes. Seats are allocated according to the quotient $\frac{V}{s+1}$ ($V$ the number of votes received, $s$ the number of seats already allocated).
Nanson’s method: Compute the Borda scores of all candidates and eliminate the candidate with the lowest Borda score (using some tie breaking mechanism). Then, proceed in the same way with the remaining candidates, recomputing the Borda score.

Proportional Approving voting: Each voter gives points $1, \frac{1}{2}, \frac{1}{3}, \ldots, \frac{1}{n}$ for her candidates (she can choose as many or few as she likes). The winning candidates are those, where the sum of all points is maximal (across all voters).
4.2 Formal model for social choice
Ballots of the voters

How should a **general model for voting** look like? What are **fair** elections based on it?

Before determining such a model we need to answer the question **How should voters express their intentions?**

Voters often have preferences over candidates: “A” is better than “B”, but “C” is better than both of them. Only if “C” is not a candidate, I would vote for “B”. What are properties of such an ordering?

**First try:** Here are two “obvious” properties.

- **transitive:** If “A” is better than “B” and “B” is better than “C”, then “A” should be considered better than “C”.
- **no cycles:** A voter should not be able to express that “A” is a strictly better candidate than “B” and, at the same time, “B” is strictly better than “A”.
Ballots of the voters (2)

Thus we could express the ballot of an agent as a \textit{dag}, a directed acyclic graph.

\begin{figure}[h]
\centering
\begin{tikzpicture}
\node (A) at (0,0) {A};
\node (B) at (0,-1) {B};
\node (C) at (1,0) {C};
\node (D) at (2,0) {D};
\node (E) at (2,-1) {E};
\node (F) at (2,-2) {F};
\path[->, thick]
(A) edge (B)
(B) edge (C)
(C) edge (D)
(D) edge (E)
(E) edge (F)
\end{tikzpicture}
\caption{Two Examples of dags.}
\end{figure}

In all voting systems considered in Section 1, the voters’ ballot can be modelled with dags.

But finite dags are just finite \textit{strict partial orders}. In fact, most ballots are even based on \textit{linear orders}.

\textit{Cycles might make sense for non-strict orderings (a cycle might model that all elements in it are of equal standing).}
Ballots of the voters (3)

We use binary relations for the ballot of an agent.

- A set of agents, $O$ set of possible outcomes. ($O$ could be $A$, a set of laws, or a set of candidates).

Preferences based on binary orderings $\preceq_i$

The preference order or ranking of agent $i$ is described by a binary relation

$\preceq_i \subseteq O \times O$.

- Which properties should we assume from such a binary relation?
Ballots of the voters (4)

What are the right properties for $\prec_i$?

- To express **dags**: irreflexive, transitive and asymmetric (**strict partial orders** $\nleq$).

We could also use non-strict partial orders $\leq$, which are in one-to-one correspondence via:

- $\leq$ is the **reflexive closure** of $\nleq$,
- $\nleq$ is the **irreflexive kernel** of $\leq$.

How to express ties?

A partial order allows for **incomparability**: $a$ and $b$ might simply not be ordered at all (in the first graph on Slide 261 elements C and E are tied). **Voters have more freedom when they are allowed not to order candidates.**
**Second try:** We could also consider **total (or linear) orders:** they are transitive and strictly total (for all $a, b$ either $a \prec_i b$ or $b \prec_i a$). In that case, ties are not allowed: voters have to take a decision for each pair of candidates. So voters have less possibilities to express their ballot.
Ballots of the voters = Weak Orders

Orderings inbetween partial and total orders are the following:

Definition 4.6 (Weak Order, $L(O)$)

Any binary relation $\preceq$ satisfying transitivity and totality (for all $a, b$: $a \preceq b$ or $b \preceq a$) is called a weak order (total preorder). We denote by $L(O)$ the set of all such binary relations over $O$ (we omit $O$ if it is clear from context).

Weak orders allow ties: it is perfectly possible that there are pairs $a \neq b$ with $a \preceq b$ and $b \preceq a$. Thus $a$ and $b$ are indifferent: the weak order treats them as equivalent. This can not happen with linear orders (see previous slide).

Note the difference to partial orders, where such ties could be modelled as incomparable. However, also note the subtle differences of the two concepts.
Ballots of the voters = Weak Orders (cont.)

Again there is a variant of weak orders $\prec_i$ in which no voters are indifferent between options. They are simply total orders.

So we allow elements to be “equivalent” (or indifferent).

- $a \prec_i b$ iff $a \preceq_i b$ and not $b \preceq_i a$. “i strictly prefers $b$ over $a$”.
- $a \sim_i b$: $a \preceq_i b$ and $b \preceq_i a$. “i is indifferent between $a$ and $b$”.

Note that $\prec_i$ is not total anymore, just a transitive partial order. However, not any partial order can be obtained as a strict weak order.

Attention

It is important not to confuse $=$ and $\sim$. 
Dag, weak orders, strict weak order

Here we illustrate the use of dags compared with weak orders.

Figure 22: Equivalent modelings as dag (left), as weak order (middle), and as strict weak order.
Definition 4.7 (Preference Profile)

A **preference profile** for agents $1, \ldots, |\mathcal{A}|$ is a tuple

$$\langle \succeq_1, \ldots, \succeq_{|\mathcal{A}|} \rangle \in L^{|\mathcal{A}|} \quad (:= \prod_{i=1}^{|\mathcal{A}|} L)$$

We often write $\vec{\succeq}$ for $\langle \succeq_1, \ldots, \succeq_{|\mathcal{A}|} \rangle$ if the set of agents is clear from context.

So the weak orders $\succeq_i$ do allow for ties (although they are total). They are also called **preferences** or **rankings**.
Often, not all subsets of $O$ are votable, only a subset $V \subseteq 2^O \setminus \{\emptyset\}$. The simplest scenario is for $V = \{O\}$.

Each $v \in V$ represents a possible “set of candidates”. The voting model then has to select some of the elements of $v$.

Each agent votes independently of the others. But we also allow that only a subset is considered. Let therefore be

$$U \subseteq \prod_{i=1}^{\mid A \mid} L.$$ 

The set $U$ represents the set of agents (and their preferences over the candidates) participating at the election and casting their votes.
Election systems

In the following sections we introduce social choice functions, correspondences, or social welfare function. Depending on which setting we consider, the characterization of the voters preferences is different:

Social choice functions: Voters express their preferences by strict total orders $\prec$.

Social choice correspondences: Voters express their preferences by strict total orders $\prec$.

Social welfare functions: Voters express their preferences by weak orders $\sim$. 
Election Systems

We have now a good idea about how to model the voters and their preferences. **How about the outcome of an election?**

**Social Choice Function (SCF):** We define any

\[ C^* : V \times U \rightarrow O; \langle v, \prec \rangle \mapsto o \]

as an election. The outcome is **one single winner**.

**Social Choice Correspondence (SCC):** We define any

\[ W^* : V \times U \rightarrow 2^O; \langle v, \prec \rangle \mapsto v' \]

as an election. The outcome is a **set of winners**.

**Social Welfare Function (SWF):** This approach views a

\[ f^* : U \rightarrow L; \prec \mapsto \prec^* \]

as an election: The outcome is a **weak order**. This weak order then determines the winners of the election (the maximal (or top) elements, for example.)

We consider these approaches in the following sections.
Dictators

In the next sections we show that under reasonable conditions on the election systems, there simply does not exist a fair election. More specifically we show results of the form

Dictators always exist

For each of the election systems defined, under reasonable assumptions on the election process, there is only a dictatorship possible.

What are the most preferred elements of $\prec$?

- $\text{top}(\prec) = \{ o \in O \mid \forall o' \in O : o \prec o' \Rightarrow o' \prec o \}$
- $\text{bot}(\prec) = \{ o \in O \mid \forall o' \in O : o \prec o' \Rightarrow o \prec o' \}$

In the case of a strict total order $\prec$, the sets $\text{top}(\prec)$ and $\text{bot}(\prec)$ are singletons.
Dictators (2)

Informally, a dictator is an agent $i$, such that whatever the profile of all voters $\succsim$ looks like, the result of the election is always the one that agent $i$ puts forward.

Let now $U \subseteq \prod_{i=1}^{A} L$ be given.

**Social Choice Function:** A dictator is an agent $i$ if for all $\succsim$:  
$$\text{top}(\succsim_i|_v) = \{ C^*(v, \succsim) \}$$

**Social Choice Correspondence:** A dictator is an agent $i$ if for all $\succsim$:  
$$\text{top}(\succsim_i|_v) \subseteq W^*(v, \succsim)$$

**Social Welfare Function:** A dictator is an agent $i$ if for all $\succsim$:  
for all $o, o' \in O$, $o \prec_i o' \Rightarrow o \prec o'$

where $\prec$ is the **strict** version of the weak order $f^*(\langle \succsim_1, \ldots, \succsim_{|A|} \rangle)$
4.3 Social Choice Functions
In this section agents ballots are strict total orders $\prec$.

**Definition 4.8 (Social choice function (SCF))**

A social choice function is any function

$$C^* : V \times U \rightarrow O; (v, \prec) \mapsto o$$

where $o \in v$.

A SCF returns exactly one “winner”. E.g. plurality voting where ties are broken in a predefined way (e.g. lexicographic ordering).
Definition 4.9 (Unanimity, Monotonicity)

A SCF \( C^* \) satisfies

**surjectivity:** for any \( v \in V \) and outcome \( o \in v \) there is a profile \( \prec \) such that \( C^*(v, \prec) = o \).

**unanimity:** for any profile \( \prec \), \( v \in V \) and outcome \( o \in v \):
   If for all \( 1 \leq i \leq |A| : o = top(\prec_i|v) \) then \( C^*(v, \prec) = o \),

**strong monotonicity:** for any profile \( \prec \), \( v \in V \) and \( o = C^*(v, \prec) \in v \):
   if \( \prec' \) is a different profile such that for all \( o' \neq o \) and \( i \)
   \( o' \prec_i o \) implies \( o' \prec'_i o \),
   then \( C^*(v, \prec') = o \).

Intuitively, strong monotonicity means that any additional support for a winning alternative, should only benefit that alternative.
Example 4.10 (Plurality with runoff)

This is the system used in the french elections. Assume that

- 6 voters support $y ≺ z ≺ x$
- 5 voters support $z ≺ x ≺ y$
- 6 voters support $x ≺ y ≺ z$

Under plurality with runoff, $x$ and $z$ make it to the second round, where $x$ wins with 11 to 6.

Suppose 2 voters of the last group change their preferences and behave like the first group. Thus there is additional support for $x$.

But now, $x$ and $y$ make it to the second round, where $y$ beats $x$ with 9 to 8.
Theorem 4.11 (May (1952))

If there are only two candidates, there is a SCF which is not dictatorial but yet satisfies unanimity and strong monotonicity.

An example of such a SCF is simple majority voting, called plurality voting when there are more than two candidates.

In fact, we get a complete characterization if we assume two more, very natural properties expressing that a choice function should by symmetric wrt. (1) individuals (anonymity), and (2) alternatives (neutrality). As this result holds in the general case for correspondences, we refer to Slide 293.
Theorem 4.12 (Muller-Satterthwaite (1977))

If there are at least 3 candidates, then any SCF satisfying surjectivity and strong monotonicity must be dictatorial.

What about plurality voting?

Suppose we fix a $o_{\text{fix}} \in O$, and define a SCF by mapping any ranking profile to $o_{\text{fix}}$. Is that SCF strongly monotone? Is it non-dictatorial? Is it unanimous? What sort of properties does it not satisfy?
Decisive sets

An important proof technique is that of decisive sets. Intuitively, a set of agents $G$ is decisive on a pair $(p, o)$, if $p$ cannot be a winner, if all agents in $G$ put $p$ below $o$. More formally:

**Definition 4.13 (Decisive sets $G \subset A$)**

A set $G \subset A$ is called decisive for the pair $(p, o) \in O^2$, if for all $i \in G$: $p \prec_i o$ implies $C^*(O, \prec^i) \neq p$.

**Definition 4.14 (Independence)**

A SCF $C^*$ satisfies the Independence property, if the following holds: If $C^*(O, \prec) = o$, then $C^*(O, \prec') \neq p$ for all $p \neq o$ with "$p \prec_i o$ iff $p \prec'_i o$".
Contraction-Lemma

Two observations are important:

- The grand coalition \( \mathcal{A} \) is decisive on any pair \((p, o)\) (this follows from surjectivity and strong monotonicity).
- A singleton set \( \{d\} \) is decisive on any pair \((p, o)\) if and only if \(d\) is a dictator.

Contraction-Lemma

Let \( G \subseteq \mathcal{A} \) with \( |G| \geq 2 \) be decisive on all pairs \((p, o) \in O^2\) and \( G = G_1 \cup G_2 \) (meaning \( G_1 \cap G_2 = \emptyset \) and \( G = G_1 \cup G_2 \)). Then \( G_1 \) or \( G_2 \) is decisive on all pairs \((p, o) \in O^2\).

If we can prove the contraction lemma under some assumptions, then this implies that there is a dictator (using our two observations above).
Proof of Theorem 4.12.

- We first show that $G \subseteq A$ is decisive on any pair $(p, o) \in O^2$, if the following holds: “If $G$ consists of all $i \in A$ with $p \prec_i o$ then $C^*(O, \prec) \neq p$”. $\leadsto$ whiteboard.
- We then show that strong monotonicity implies independence

  Hint: Given a profile $\prec$ consider the following profile $\prec'$ and apply strong monotonicity (twice): “$o \prec_i o'$” iff: $o \prec_i o'$ and $o, o'$ are in each $\prec_i$ ranked among the two top places ($o, o'$ are the two outcomes in the independence property).
- Then we show the Contraction-Lemma (in the proof we need the independence property).
Theorem of Muller-Satterthwaite (1977) (2)

Proof (cont.)

We prove the Contraction lemma. So $G = G_1 \hat{\cup} G_2$ is given. We construct a ranking profile $\prec$ as follows.

- for all $i \in G_1$: $q \prec_i p \prec_i o$,
- for all $i \in G_2$: $o \prec_i q \prec_i p$,
- for all $i \in \mathcal{A} \setminus G$: $p \prec_i o \prec_i q$,
- all other $o' \in O$ are ranked below $o, p, q$ by all agents $i$.

Because $G$ is decisive on all pairs, $q$ can not be the winner. So either $o$ or $p$ is the winner.
Theorem of Muller-Satterthwaite (1977) (3)

Proof (cont.)

\( p \) is the winner: Only those in \( G_2 \) rank \( o \) below \( p \).
Therefore \( G_2 \) is decisive.

\( o \) is the winner: Only those in \( G_1 \) rank \( q \) below \( o \).
Independence implies that \( q \) looses against \( o \) in each profile in which exactly the agents from \( G_1 \) rank \( q \) below \( o \).
Therefore \( G_1 \) is decisive.
Manipulation

Can voters hide their true preferences and use another ballot to achieve better results?

Example 4.15 (Plurality Voting)

- 49%: Nader ≺≺≺ Gore ≺≺≺ Bush
- 20%: Bush ≺≺≺ Nader ≺≺≺ Gore
- 20%: Nader ≺≺≺ Bush ≺≺≺ Gore
- 11%: Bush ≺≺≺ Gore ≺≺≺ Nader

If the last group of voters change their preferences by putting their favorite at the end, then they achieve a better result overall!

Can we find choice functions where such manipulations are not possible?
Strategy-Proofness

Earlier, in Definition 1.18 on Slide 47 we have introduced the notation using $-i$ to denote a profile without agent $i$’s entry. We are using this notation here as well.

**Definition 4.16 (Strategy-Proofness)**

A social choice function $C^*$ is called **strategy-proof**, if there is no agent $i$ and profile $\langle \prec_{-i}, \prec'_i \rangle$ such that

$$C^*(O, \prec_{-i}) \prec_i C^*(O, \langle \prec_{-i}, \prec'_i \rangle)$$

We compare the profile $\langle \prec_1, \ldots, \prec_{i-1}, \prec'_i, \ldots, \prec_A \rangle$, in which agent $i$ misrepresents her true preference $\prec_i$, with the real profile $\langle \prec_1, \ldots, \prec_{i-1}, \prec_i, \ldots, \prec_A \rangle$. 
Theorem of Gibbard and Satterthwaite

Lemma 4.17

Strategy-proofness implies strong monotonicity.

Proof.

We assume that a SCF is not strongly monotone and show that it is not strategy-proof. So we assume there is $o \neq o'$ and $\vec{x}, \vec{x}'$ s.t.

- $C^*(O, \vec{x}) = o$, and $C^*(O, \vec{x}') = o'$, and
- for all $i \in A$, for all $p \in O \setminus \{o\}$: $p \prec_i o$ implies $p \prec_i' o$.

We are now modifying $\vec{x}$ as follows. For $i = 1, 2, \ldots, n$ we replace successively $\prec_i$ by $\prec_i'$, until the winner of the new action profile under $C^*$ is no more $o$ but somebody else (which must happen, because at the end it is $o'$). Let $j$ be this agent.

Because we can adapt our assumption to this new situation, we assume wlog that $\vec{x}, \vec{x}'$ differ only at the entry $j$. So wlog we can use the notation $o'$ as in our assumption.
Theorem of Gibbard and Satterthwaite (2)

Proof (cont.)

Case 1: “\( o' \prec_j o \)”. If \( j \)’s true preferences are as in \( \prec_j \), then it pays off for \( j \) to vote as in \( \prec_j \) (to ensure \( o \) is winning, and not \( o' \)). So it is not strategy-proof.

Case 2: “not \( o' \prec_j o \)”. Then “not \( o' \prec_j o \)”, therefore \( o \prec_j o' \). If \( j \)’s true preferences are as in \( \prec_j \), then it pays off for \( j \) to vote as in \( \prec_j \) (to ensure \( o' \) is winning, and not \( o \)). So it is not strategy-proof.
Theorem of Gibbard and Satterthwaite (3)

**Theorem 4.18 (Gibbard-Satterthwaite (1973/1975))**

*If there are at least 3 candidates, then any SCF satisfying surjectivity and strategy-proofness must be dictatorial.*

Using Lemma 4.17 this is just a corollary to Theorem 4.12.
4.4 Social Choice Correspondences
In this section agents ballots are strict total orders $\preceq$. We also assume $V = O$ in order not to simplify notation.

**Definition 4.19 (Social choice correspondence)**

A **social choice correspondence** is any function

$$W^* : U \to 2^O; \vec{\preceq} \mapsto v$$

where $v \neq \emptyset$.

A correspondence returns a **nonempty set** of winners.

The **Borda rule** is a typical example of a correspondence.

Impossibility results similar to Theorem 4.12 are rare. The most important one is due to **Duggan and Schwartz**: See Theorem 4.25 on Slide 299.
Definition 4.20 (Positive responsiveness)

A SCC $W^*$ satisfies

**positive responsiveness**: for any profile $\vec{\succ}$, $o \in O$:

if $o \in W^*(\vec{\succ})$ then $W^*(\vec{\succ}') = \{o\}$, provided that $\vec{\succ}'$ is a different profile such that for all $o' \neq o \neq o''$ and $i$ the following holds:

$$o'\prec_i o \text{ implies } o'\prec_i' o', \text{ and } o'\prec_i o'' \text{ iff } o'\prec_i' o''.$$  

Intuitively, this property means that if $o$ is among the winners and at least one voter raises $o$ up, then $o$ should become the sole winner.

Consider the SCC that always declares all candidates as winners, independently of the voters choice.
Anonymity and neutrality

Definition 4.21 (Anonymity and Neutrality)

\( W^* \) satisfies **anonymity** if for all permutations \( \pi \) on \( \{1, \ldots, |A|\} \):

\[
W^*(\langle \prec_1, \ldots, \prec_{|A|}\rangle) = W^*(\langle \prec_{\pi(1)}, \ldots, \prec_{\pi(|A|)}\rangle)
\]

\( W^* \) satisfies **neutrality** if for all permutations \( \pi \) on \( O \)

\[
\pi(W^*(\vec{\prec})) = W^*(\pi(\vec{\prec}))
\]

where \( \pi(\vec{\prec}) \) is defined componentwise and \( \pi(\prec_i) \) is defined in the obvious way: \( o \pi(\prec_i) o' \) iff: \( \pi(o) \prec_i \pi(o') \).
Only two candidates: $|O| = 2$

**Theorem 4.22 (May (1952))**

Assume there are only two candidates, $|O| = 2$. A SCC $W^*$ satisfies anonymity, neutrality, and positive responsiveness if and only if $W^*$ is simple majority voting.

Assume there are only 2 candidates and 2 voters, and we are considering a social choice function $C^*$ (not a correspondence). Then the choice function can not satisfy both anonymity and neutrality.
Weak Monotonicity

There is a weaker version of the responsiveness condition, namely *weak monotonicity*, when we replace

\[ W^*(\vec{z}') = \{ o \} \] by \( o \in W^*(\vec{z}') \)

- Does Theorem 4.22 hold for weak monotonicity instead of positive responsiveness?
Optimistic and pessimistic voters

**Strategy-proof** in Definitions 4.24 and 4.44 rules out manipulability by untruthful voting: by misrepresenting their true preferences, voters should not be able to get overall better results. Social choice correspondences determine sets of winners (not just single winners).

⇝ We need to rank such sets.
Optimistic and pessimistic voters (2)

Definition 4.23 (Optimist and pessimists)

An agent $i$ is an \textbf{optimist}, if she ranks $X$ higher than $Y$, whenever $\text{top}(\prec_i | Y) \prec_i \text{top}(\prec_i | X)$.

An agent $i$ is a \textbf{pessimist}, if she ranks $X$ higher than $Y$, whenever $\text{bot}(\prec_i | Y) \prec_i \text{bot}(\prec_i | X)$.

By slightly abusing notation, we also write $Y \prec_i X$.

As we are considering strict linear orders, top and bottom elements are always unique.
But even in the case of \textit{weak orders} all maximal elements (and all minimal elements) are indifferent to each other, so both definitions are well-defined (see Slide 266).
4 Social Choice and Auctions

4.4 Social Choice Correspondences

**Definition 4.24 (Strategy-Proofness)**

A social choice correspondence \( W^* \) is called \textbf{strategy-proof}, if there is no agent \( i \), profile \( \prec \) and ranking \( \prec' \) such that for all \( v \in V \)

\[
W^*(\prec) \prec_i W^*(\langle \prec_{-i}, \prec' \rangle)
\]

We compare the profile \( \langle \prec_1, \ldots, \prec_{i-1}, \prec' \ i, \ldots, \prec_A \rangle \), in which agent \( i \) misrepresents her true preference \( \prec_i \), with the real profile \( \langle \prec_1, \ldots, \prec_{i-1}, \prec_i, \ldots, \prec_A \rangle \).

Note that we consider the relation \( W^*(\langle \prec_{-i}, \prec_i \rangle) \prec_i W^*(\langle \prec_{-i}, \prec' \ i \rangle) \) only for optimistic or pessimistic voters \( i \) (see Definition 4.23).
Theorem 4.25 (Duggan/Schwartz (2000))

We assume that there are at least 3 candidates. Then any SCC that is nonimposed (for each \( o \in O \) there is \( \vec{\prec} \) s.t. \( W^*(\vec{\prec}) = \{o\} \)) and strategy-proof for both optimistic and pessimistic voters is dictatorial.

What is the importance of the nonimposed property?
The proof is based on several lemmas. It has striking similarity to the proof of Gibbard/Satterthwaite. Important concepts are down monotonicity and dictating sets (corresponding to decisive sets).

### Definition 4.26 (Down-monotonicity)

Suppose we have a profile $\prec$ with $W^*(\prec)$ is a singleton and the following holds: If we modify $\prec$ by letting one agent move a losing alternative down one spot (obtaining profile $\prec'$), then $W^*(\prec) = W^*(\prec')$. Then we call $W^*$ down-monotone for singleton winners.

### Lemma 4.27

A SCC $W^*$ satisfying strategy-proofness for both optimistic and pessimistic voters also satisfies down-monotonicity for singleton winners.
Definition 4.28 (Dictating Sets $G, pGo$)

Let $G \subseteq A$ a group of agents and $p, o \in O$. We denote by $pG^*o$ the fact that $W^*(\prec) \neq \{p\}$ for all profiles $\prec$ in which all agents from $G$ rank $o$ above $p$.

A set $G \subseteq A$ is called dictating for $W^*$, if $pG^*o$ holds for all pairs $(p, o)$.

Lemma 4.29

Suppose $G \subseteq A$, $p, o \in O$, and $pG^*o$. Let $o \neq o' \neq p$ and $G = G_1 \cup G_2$.

Then we have $o'G_1o$ or $pG_2o'$. 
Lemma 4.30

If $W^*$ is down-monotonic for singleton winners and nonimposed, then the set of all agents is a dictating set.

Lemma 4.31

We assume SCC $W^*$ satisfies strategy-proofness for both optimistic and pessimistic voters.

- Let $G$ be a dictating set. If it is the disjoint union of sets $G_1$ and $G_2$, then one of these sets is dictating too.

- There is an agent whose maximal element is the unique winner, whenever $W^*(\prec)$ is a singleton.
4.5 Social Welfare Functions
In this section agents ballots are weak orders $\prec$. We also use $o \prec_i o'$ to denote “$o \preceq o'$ and not $o' \preceq o$”: $o'$ is strictly greater than $o$.

**Definition 4.32 (Social welfare function)**

A social welfare function is any function

$$f^* : U \rightarrow L; (\preceq_1, \ldots, \preceq_{|A|}) \mapsto \preceq^*$$

For each $V \subseteq 2^O \setminus \{\emptyset\}$ the function $f^*$ w.r.t. $U$ induces a choice correspondence $C_{(\preceq_1, \ldots, \preceq_{|A|})}$ as follows:

$$C_{(\preceq_1, \ldots, \preceq_{|A|})} = \text{def } \left\{ \begin{array}{c} V \\ v \end{array} \mapsto \left\{ \begin{array}{c} V \\ \text{top}(\preceq^*|v) \end{array} \right. \right\}$$

Each tuple $\preceq$ determines the election for all possible $v \in V$. 

What are desirable properties for $f^*$?

- **Weak Pareto Efficiency:**
  for all $o, o' \in O$: $\left( \forall i \in A : o \prec_i o' \right)$ implies $o \prec^* o'$.

- **Independence of Irrelevant Alternatives (IIA):**
  for all $o, o' \in O$:
  
  - $\left( \forall i \in A : o \prec_i o' \right) \Rightarrow \left( o \prec^* o' \right)$,
  - $\left( \forall i \in A : o \preceq_i o' \right) \Rightarrow \left( o \preceq^* o' \right)$,
  - $\left( \forall i \in A : o \sim_i o' \right) \Rightarrow \left( o \sim^* o' \right)$

**IIA** expresses that the social ranking of two alternatives does only depend on the relative individual rankings of these alternatives.

Note that this implies in particular

$$\left( \forall i \in A : \preceq_i | v = \preceq_i' | v \right) \Rightarrow \forall o, o' \in v, \forall v' \in V \text{ s.t. } v \subseteq v' : \left( o \prec^* | v', o' \right) \iff o \prec{'}^* | v', o'$$
Shouldn’t we also ask in the pareto efficiency condition that

\[(\forall i \in A : o \preceq_i o') \text{ implies } o \preceq^* o'\]?

The answer is no, the stated condition, also called weak pareto efficiency, is sufficient for Arrows theorem. The stronger condition, usually called pareto efficiency, is not needed. Note that for strict linear orders, there is no such distinction.
Example 4.33 (Which champagne is the best?)

Suppose you go out for dinner and you want to start with a champagne.

- The waiter gives you the choice between a Blanc de Blancs or a Blanc de Noirs (both grands crus from respectable houses)
- You choose a Blanc de Blancs.
- Then the waiter returns and mentions that they also have a Rosé.
- “Oh, in that case, I take a Blanc de Noirs.”
Majority Vote

The simple **majority vote** protocol does not satisfy the **IIA**.

We consider 7 voters \((\mathcal{A} = \{w_1, w_2, \ldots, w_7\})\) and \(O = \{a, b, c, d\}\), \(V = \{\{a, b, c, d\}, \{a, b, c\}\}\). The columns in the following table represent two different preference orderings of the voters (black and red).

<table>
<thead>
<tr>
<th></th>
<th>(\prec_1) ((\prec_1))</th>
<th>(\prec_2) ((\prec_2))</th>
<th>(\prec_3) ((\prec_3))</th>
<th>(\prec_4) ((\prec_4))</th>
<th>(\prec_5) ((\prec_5))</th>
<th>(\prec_6) ((\prec_6))</th>
<th>(\prec_7) ((\prec_7))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(a)</td>
<td>((2))</td>
<td>((2))</td>
<td>((1))</td>
<td>((1))</td>
<td>((2))</td>
<td>((2))</td>
</tr>
<tr>
<td></td>
<td>(b)</td>
<td>((3))</td>
<td>((3))</td>
<td>((2))</td>
<td>((2))</td>
<td>((1))</td>
<td>((1))</td>
</tr>
<tr>
<td></td>
<td>(c)</td>
<td>((4))</td>
<td>((4))</td>
<td>((3))</td>
<td>((3))</td>
<td>((3))</td>
<td>((3))</td>
</tr>
<tr>
<td></td>
<td>(d)</td>
<td>((1))</td>
<td>((1))</td>
<td>((4))</td>
<td>((4))</td>
<td>((4))</td>
<td>((4))</td>
</tr>
</tbody>
</table>

Let \(\prec^*\) be the solution generated by \(\prec\) and \(\prec^*\) the solution generated by the \(\prec\). Then we have for \(i = 1, \ldots, 7\): \(b \prec_i a\) iff \(b \prec_1 a\), but \(b \prec^* a\) and \(a \prec^* b\). The latter holds because on the whole set \(O\), for \(\prec^*\) \(a\) gets selected 4 times and \(b\) only 3 times, while for \(\prec^*\) \(a\) gets selected only 2 times but \(b\) gets still selected 3 times. The former holds because we even have \(\prec_i\mid\{a, b, c\} = \prec_i\mid\{a, b, c\}\).

The introduction of the **irrelevant** (concerning the relative ordering of \(a\) and \(b\)) alternative \(d\) changes everything: the original majority of \(a\) is split and drops below one of the less preferred alternatives \((b)\).
We assume that there are at least two voters and three candidates ($|O| \geq 3$). If the SWF $f^*$ satisfies Weak Pareto Efficiency and Independence from Irrelevant Alternatives, then there always exists a dictator.
Proof (of Arrows theorem).

The proof is the third proof given by John Geanakoplos (1996) and based on the following

Lemma 4.35 (Strict Neutrality)

We assume Pareto Efficiency and IIA and consider two pairs of alternatives \( a, b \) and \( \alpha, \beta \). Suppose each voter strictly prefers \( a \) to \( b \) or \( b \) to \( a \), i.e. for all \( i \): \( a \prec_i b \) or \( b \prec_i a \). Suppose further that each voter has the same preference for \( \alpha, \beta \) as she has for \( a, b \).

Then either \( a \prec^* b \) and \( \alpha \prec^* \beta \) or \( b \prec^* a \) and \( \beta \prec^* \alpha \).
A simple corollary is the following:

**Corollary 4.36 (Extremal Lemma)**

*Let the social welfare function $f^*$ satisfy Pareto Efficiency and Independence of Irrelevant Alternatives. Let $o \in O$ and suppose each voter $i$ puts $o$ either on the very top (unique top element wrt. $\prec_i$) or to the very bottom (unique bottom element wrt. $\prec_i$). Then $o$ is either a unique bottom or a unique top element of $\preceq^*$.***
Proof (of the lemma).

We assume wlog that \((a, b)\) is distinct from \((\alpha, \beta)\) and that \(b \succ^* a\) (we have to show the preference is strict).

We construct a different profile \(\langle \succsim'_1, \ldots, \succsim'_{|A|} \rangle\) obtained as follows (for each \(i\)):

- If \(a \neq \alpha\), we change \(\succsim_i\) by moving \(\alpha\) just strictly above \(a\).
- If \(b \neq \beta\), we change \(\succsim_i\) by moving \(\beta\) just strictly below \(b\).

This can be done by maintaining the old preferences between \(\alpha\) and \(\beta\) (as preferences between \(a\) and \(b\) are strict).
By pareto efficiency, we have $a \prec'\!
\prec\!
\prec'\!
\alpha$ (for $\alpha \neq a$) and $\beta \prec'\!
\prec\!
\prec'\!
\beta \prec'\!
\prec\!
\prec'\!
b$ (for $\beta \neq b$).

By IIA, we have $b \preceq'\!
\preceq\!
\preceq'\!
\alpha$. Using transitivity, we get $\beta \prec'\!
\prec\!
\prec'\!
\alpha$.

By IIA again, we also get $\beta \prec'\!
\prec\!
\prec'\!
\alpha$ (because $\alpha \prec'\!
\prec\!
\prec'\!
i \beta$ iff $\alpha \prec'\!
\prec\!
\prec'\!
i \beta$).

We now reverse the roles of $(a, b)$ with $(\alpha, \beta)$ and apply IIA again to get $b \prec'\!
\prec\!
\prec'\!
a$. □
The proof of Arrows theorem is by considering two alternatives $a, b$ and the profile where $a \prec_i b$ for all agents $i$. By pareto efficiency, $a \prec^* b$. Note that this reasoning is true for all rankings with the same relative preference between $a$ and $b$.

We now consider a sequence of profiles $\langle \prec_1^i, \ldots, \prec_{|A|}^i \rangle$ (from $i = 0, \ldots, |A|$, starting with the one described above ($i = 0$)), where in step $i$, we let all agents numbered $\leq i$ change their profile by moving $a$ strictly above $b$ (leaving all other rankings untouched). In fact, only one agent, number $i$ changes its ranking in step $i$. 
We consider the rankings $\prec^i_*$ obtained from $\prec^i_1$. There must be one step, let’s call it $d$, where $a \prec^{d-1}_* b$ but $b \prec^{d}_* a$ (because of pareto efficiency and strict neutrality).

Again: this reasoning is true not just for one profile $\langle \prec^i_1, \ldots, \prec^i_{|A|} \rangle$, but for all such profiles with the same relative ranking of $a$ and $b$.

We claim that $d$ is a dictator.
Take any pair of alternatives \( \alpha, \beta \) and assume wlog \( \beta \prec_d \alpha \) (otherwise the following argument works as well for \( \alpha \prec_d \beta \)).

\( d \) is a dictator, when we can show \( \beta \prec_d^* \alpha \).

---

Take \( c \notin \{ \alpha, \beta \} \) (because \( |O| \geq 3 \)) and consider the new profile \( \langle \prec_1', \ldots, \prec_{|A|}' \rangle \) obtained as follows from \( \langle \prec_1^d, \ldots, \prec_{|A|}^d \rangle \):

- for \( 1 \leq i \leq d \): we put \( c \) on top of each \( \prec_i^d \).
- for \( d \): we put \( c \) inbetween \( \alpha \) and \( \beta \).
- for \( d \leq i \leq |A| \): we put \( c \) to the bottom of each \( \prec_i^d \).

We are changing the profile \( \langle \prec_1^d, \ldots, \prec_{|A|}^d \rangle \) by moving \( c \) around in a very particular way.

We apply strict neutrality to the pair \( \langle c, \beta \rangle \) and \( \langle a, b \rangle \). Because both pairs have the same relative ranking in \( \langle \prec_1', \ldots, \prec_{|A|}' \rangle \), we have \( \beta \prec_d^c \).
Now we consider the new profile $\langle \preceq''_1, \ldots, \preceq''_{|\mathcal{A}|} \rangle$ obtained as follows from $\langle \preceq'_1, \ldots, \preceq'_{|\mathcal{A}|} \rangle$:

- For $1 \leq i \leq d$: we put $c$ to the bottom of each $\preceq'_i$.
- For $d$: we put $c$ in between $\alpha$ and $\beta$.
- For $d \leq i \leq |\mathcal{A}|$: we put $c$ to the top of each $\preceq'_i$.

We now apply strict neutrality to the pair $(\alpha, c)$ and $(a, b)$. Because both pairs have the same relative ranking in $\langle \preceq''_1, \ldots, \preceq''_{|\mathcal{A}|} \rangle$, we have $c \preceq^d \alpha$.

By transitivity: $\beta \preceq^d \alpha$. □
Ways out (of Arrow's theorem):

1. Choice function is not always defined.
2. Independence of alternatives is dropped.
4.6 Results based on partial orders
Basic Definition: SCT on partial orders

Up to now we considered ballots consisting of **linear orders** (total orders) or **weak orders** (total preorders). There are also some results based on the more general **partial orders** (or dags).

**Definition 4.37 (Incomplete order)**

An **incomplete order** (IO) is a preorder: a binary relation which is **reflexive** and **transitive**.
As before, we can use an IO to define strict preference as well as indifference, but also incomparability (must not be confused with indifference!).

- \( a ≺_i b \) iff \( a ≲_i b \) and not \( b ≲_i a \) (\( i \) strictly prefers \( a \) over \( b \))
- \( a ∼_i b \) iff \( a ≲_i b \) and \( b ≲_i a \) (\( i \) is indifferent between \( a \) and \( b \))
- \( a ∆_i b \) iff not \( a ≲_i b \) and not \( b ≲_i a \) (\( i \) considers \( a \) and \( b \) incomparable)

Which axiomatic property distinguishes \( ∼ \) from \( ∆ \)?

**Definition 4.38 (Strict IO)**

A strict incomplete order (SIO) is a IO \( \preceq \) such that for all distinct alternatives \( o \) and \( o' \), we have that not \( o ∼ o' \).
What are the most preferred elements of ≼?

- \( \text{top}(≼) = \{ o \in O | \forall o' \in O : o \neq o' \Rightarrow \neg (o ≼ o') \} \)
- \( \text{bot}(≼) = \{ o \in O | \forall o' \in O : o \neq o' \Rightarrow \neg (o' ≼ o) \} \)

**Definition 4.39 (Restricted IO)**

A IO ≼ is called **restricted**, denoted by rIO, if all top elements or all bottom elements are indifferent. Formally,

- for all \( o, o' \in \text{top}(≼) \), \( o \sim o' \), or
- or all \( o, o' \in \text{bot}(≼) \), \( o \sim o' \).

**The sets IO(O), SIO(O), and rIO(O).**

Given a set of alternatives \( O \) we write \( IO(O), SIO(O), \) and \( rIO(O) \) to refer to the set of all **incomplete**, **strict incomplete**, and **restricted incomplete** orders over \( O \).
We now define SCF’s, SCC’s and SWF’s over IO. To simplify the presentation, we leave out the set $V$ of votable candidates, assuming that all alternatives from $O$ are also votable.

**Social Choice Function** over SIO:

$$C^* : SIO(O) \rightarrow O; \prec \rightarrow o$$

**Social Choice Correspondence:**

$$W^* : SIO(O) \rightarrow 2^O; \prec \rightarrow v'$$

**Social Welfare Function** over IO:

$$f^* : IO(O) \rightarrow IO(O); (\preceq_1, \ldots, \preceq_{|A|}) \mapsto \preceq^*$$
Social Welfare Functions on IO

**Unanimity:** for all outcomes $a, b \in O$ and all $\succeq \in IO(O)$, if $a \succeq_i b$ for all $i \in A$, then also $a \succeq^* b$.

**IIA:** if for any two profiles $\succeq, \succeq' \in IO(O)$ and all $a, b \in O$, if $\succeq_i$ and $\succeq'_i$ define the same ordering on $a$ and $b$ for each $i \in A$, then $\succeq^*$ and $(\succeq')^*$ define the same ordering on $a$ and $b$. 
Strong dictator: An agent $i$ is a strong dictator, if
$f^*(\lesssim) = \lesssim_i$, for every IO $\lesssim$.

Dictator: An agent $i$ is a dictator, if
$a \prec_i b$ implies $a \prec^* b$, for every IO $\lesssim$.

Weak dictator: An agent $i$ is a weak dictator, if
$a \prec_i b$ implies $a \prec^* b$ or $a \fork^* b$, for every IO $\lesssim$. 
In the following we consider variants of Arrow’s theorem. We need to distinguish between the different variants of dictators, as the next result shows:

**Proposition 4.40**

There is a SWF over IO with at least two agents and at least two outcomes which is IIA, unanimous, and has no dictator.

Exercise: Find such a SWF.
Theorem 4.41 (Arrow’s theorem for IO)

Suppose there are at least two agents and three outcomes. If the SWF $f^* : IO(O) \to rIO(O)$ is unanimous and satisfies IIA then there is weak dictator.

The proof is similar to the proof of Arrow’s theorem, adapted to the case of restricted incomplete orders.
Social Choice Correspondences on SIO

**Ontoness:** A SCF over SIO’s is onto if for any \( a \in O \) there is an \( \prec \in SIO(O) \) such that \( \{a\} = W^*(\prec) \).

**Unanimity:** A SCF over SIO’s is unanimous if for any \( a \in O \) and \( \prec \in SIO(O) \) with \( \{a\} = \text{top}(\prec_i) \) for each \( i \in A \), then \( W^*(\prec) = \{a\} \).

**Monotonicity:** A SCF over SIO’s is monotonic if for all pairs of orderings \( \prec, \prec' \in SIO(O) \) it holds that:

- if \( a \in W^*(\prec) \) and for every \( b \in O \) and \( i \in A \), \( (b \prec_i a \text{ or } b \succ_i a) \) implies \( (b \prec'_i a \text{ or } b \succ'_i a) \), then \( a \in W^*(\prec') \); and
- if \( A = W^*(\prec) \) and for every \( a \in A \), and any other \( b \in O \), and \( i \in A \), \( (b \prec_i a \text{ implies } b \prec'_i a) \) and \( (b \succ_i a \text{ implies } (b \prec'_i a \text{ or } b \succ'_i a)) \), then \( A = W^*(\prec') \).
Also for social choice correspondences we distinguish between three types of dictators.

**Strong dictator:** An agent $i$ is a **strong dictator** over SIO, if
\[ W^*(\prec) = \text{top}(\prec_i) \text{ for all profiles } \prec. \]

**Dictator:** An agent $i$ is a **dictator** over SIO, if
\[ W^*(\prec) \subseteq \text{top}(\prec_i) \text{ for all profiles } \prec. \]

**Weak dictator:** An agent $i$ is a **weak dictator** over SIO, if
\[ W^*(\prec) \cap \text{top}(\prec_i) \neq \emptyset \text{ for all profiles } \prec. \]
Again, in the case of (standard) dictators, we have the following positive result:

**Proposition 4.42**

There is a social choice correspondence over $SIO$ which is *monotonic*, *unanimous*, and has no *dictator*.

Exercise: Find such a SWF.
The generalization of Muller-Satterthweite’s theorem for weak dictators still holds:

**Theorem 4.43 (Generalization: M-S’s theorem for SIO)**

*Suppose there are at least two agents and three outcomes. If the social choice correspondence over SIO is unanimous and monotonic then there is at least one weak dictator.*
Strategy-Proofeness over SIO

We consider social choice correspondences over SIO.

Definition 4.44 (Strategy Proofness)

A SCC $W^*$ over SIOs is **strategy proof** (over SIO’s), if for all agents $i$ and all $\prec, \prec' \in SIO(O)$ with $\prec_i = \prec'_i$ and $W^*(\prec) \neq W^*(\prec')$ we have that:

1. for all $a \in W^*(\prec) \setminus W^*(\prec')$ and for all $b \in W^*(\prec) \cap W^*(\prec')$ we have $b \prec_i a$ or $a \not\succ_i b$, 
2. for all $b \in W^*(\prec') \setminus W^*(\prec)$ we have that
   1. for all $a \in W^*(\prec)$, $b \prec_i a$ or $a \not\succ_i b$, and
   2. there is an $a \in W^*(\prec)$ such that $b \prec_i a$. 

---

4 Social Choice and Auctions
4.6 Results based on partial orders
Lemma 4.45

If a SCC over SIO is strategy-proof and onto then it is unanimous and monotonic.
Theorem 4.46 (Generalization: G-S for SIO)

Suppose there are at least two agents and three outcomes. If a SCC over SIO is **strategy-proof** and **onto** then there is at least one weak dictator.

Proof.

By Lemma 4.45, a SCC which is strategy-proof and onto is also unanimous and monotonic. Thus, by Theorem 4.43 there is a weak dictator.

Note, over strict total orders, the Theorem is equivalent to Theorem 4.18.
We have presented the main impossibility results from social choice theory over incomplete orders:

- Arrow’s theorem,
- Muller-Satterthwaite’s theorem, and
- Gibbard-Satterthwaite’s theorem.

The main fundamental (negative) impossibility theorems remain also in the case of incomplete orders.
4.7 Auctions
While voting binds all agents, auctions are always deals between 2.

Types of auctions:

first-price open cry: (English, Japanese auction), as usual.
first-price sealed bid: bidding without knowing the other bids.

dutch auction: (descending auction) the seller lowers the price until it is taken (flower market).

second-price sealed bid: (Vickrey auction) Highest bidder wins, but the price is the second highest bid!
Three different auction settings:

**private value (IPV):** Value depends only on the bidder (cake).

**common value (CV):** Value depends only on other bidders (treasury bills).

**correlated value:** Partly on own’s values, partly on others.
What is the best strategy in Vickrey auctions?

Theorem 4.47 (Private-value Vickrey auctions)

The dominant strategy of a bidder in a Private-value Vickrey auction is to bid the true valuation.

Therefore it is equivalent to English auctions (which are equivalent to Dutch auctions).

Vickrey auctions are used to

- allocate computation resources in operating systems,
- allocate bandwidth in computer networks,
- control building heating.
Are first-price auctions better for the auctioneer than second-prize auctions?

**Theorem 4.48 (Expected Revenue)**

All 4 types of protocols produce the same expected revenue to the auctioneer (assuming (1) private value auctions, (2) values are independently distributed and (3) bidders are risk-neutral).
Expected revenue for agents taking risks.

For non risk-neutral agents, there is a difference in the expected revenue for second price versus first price auctions:

<table>
<thead>
<tr>
<th>risk-neutral</th>
<th>Jap.</th>
<th>Engl.</th>
<th>2nd</th>
<th>1st</th>
<th>Dutch</th>
</tr>
</thead>
<tbody>
<tr>
<td>risk-averse</td>
<td>=</td>
<td>=</td>
<td>&lt;</td>
<td>=</td>
<td></td>
</tr>
<tr>
<td>risk-seeking</td>
<td>=</td>
<td>=</td>
<td>&gt;</td>
<td>=</td>
<td></td>
</tr>
</tbody>
</table>
Why are second price auctions not so popular among humans?

1. Lying auctioneer.
2. When the results are published, subcontractors know the true valuations and what the winner saved. So they might want to share the profit.
Inefficient Allocation

Auctioning heterogeneous, *interdependent* items.

**Example 4.49 (Task Allocation)**

Two delivery tasks $t_1$, $t_2$. Two agents.
The global optimal solution is not reached by auctioning independently and truthful bidding.

\( t_1 \) goes to agent 2 (for a price of 2) and \( t_2 \) goes to agent 1 (for a price of 1.5).

Even if agent 2 considers (when bidding for \( t_2 \)) that she already got \( t_1 \) (so she bids
\[
\text{cost}(\{t_1, t_2\}) - \text{cost}(\{t_1\}) = 2.5 - 1.5 = 1
\]
) she will get it only with a probability of 0.5.
What about full lookahead?

⇝ blackboard.

Therefore:

- It pays off for agent 1 to bid more for \( t_1 \) (up to 1.5 more than truthful bidding).
- It does not pay off for agent 2, because agent 2 does not make a profit at \( t_2 \) anyway.
- Agent 1 bids 0.5 for \( t_1 \) (instead of 2), agent 2 bids 1.5. Therefore agent 1 gets it for 1.5. Agent 1 also gets \( t_2 \) for 1.5.
Lying at Vickrey

Does it make sense to counterspeculate at private value Vickrey auctions?

Vickrey auctions were invented to avoid counterspeculation.

But what if the private value for a bidder is uncertain?

The bidder might be able to determine it, but she needs to invest some costs.
Example 4.50 (Incentive to counterspeculate)

Suppose bidder 1 does not know the (private-) value $v_1$ of the item to be auctioned. To determine it, she needs to invest cost. We assume that $v_1$ is uniformly distributed: $v_1 \in [0, 1]$. 

For bidder 2, the private value $v_2$ of the item is fixed: $0 \leq v_2 < \frac{1}{2}$. So her dominant strategy is to bid $v_2$.

Should bidder 1 try to invest cost to determine her private value? How does this depend on knowing $v_2$?
\[ \text{Answer: Bidder } 1 \text{ should invest } \text{cost} \text{ if and only if } \]

\[ v_2 \geq (2 \text{cost})^{\frac{1}{2}}. \]
4.8 References

5. Incomplete Information Games

- Examples and Motivation
- Bayesian Games
- Bayes Nash equilibrium
- Bayesian equilibria and mixed equilibria
We consider **incomplete knowledge**, where players are not sure about which game they are actually playing: **Bayesian games**. The players do not know the payoffs of the other players, which makes it difficult to come up with strategies.

- We first motivate the (quite involved) definition of Bayesian games.
- Then we define the Bayes-Nash equilibrium.
- Finally we consider variants of the Bayes-Nash equilibrium (mixed equilibria).
5.1 Examples and Motivation
What if the players do not know the payoff?

What, if they do not even know the game they are playing?

It turns out that both cases above are essentially identical! Such games are called Bayesian games or incomplete knowledge games (do not mix up with imperfect knowledge).
Example 5.1 (Uncertainty about payoffs)

Agent 1 (firm) is about to decide whether to build a new plant. There are incurring costs for this. Agent 2 (opponent) is about to buy the firm of agent 1. But agent 2 is not sure about the incurring costs for agent 1. Buying is good for agent 2 if and only if agent 1 does not build. Agent 1 has a clear dominant strategy.

<table>
<thead>
<tr>
<th></th>
<th>Buy</th>
<th>Don’t</th>
<th>Buy</th>
<th>Don’t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build</td>
<td>⟨0, −1⟩</td>
<td>⟨2, 0⟩</td>
<td>⟨3, −1⟩</td>
<td>⟨5, 0⟩</td>
</tr>
<tr>
<td>Don’t</td>
<td>⟨2, 1⟩</td>
<td>⟨3, 0⟩</td>
<td>⟨2, 1⟩</td>
<td>⟨3, 0⟩</td>
</tr>
<tr>
<td>1’s costs are high</td>
<td>1’s costs are low</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Example 5.2**

Same example as before, but some payoffs are different.

<table>
<thead>
<tr>
<th></th>
<th>Buy</th>
<th>Don’t</th>
<th>Buy</th>
<th>Don’t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build</td>
<td>⟨0, −1⟩</td>
<td>⟨2, 0⟩</td>
<td>⟨1.5, −1⟩</td>
<td>⟨3.5, 0⟩</td>
</tr>
<tr>
<td>Don’t</td>
<td>⟨2, 1⟩</td>
<td>⟨3, 0⟩</td>
<td>⟨2, 1⟩</td>
<td>⟨3, 0⟩</td>
</tr>
</tbody>
</table>

1’s costs are high  1’s costs are low
**Player 1:** Let $x$ be the probability of player 1 for building (obviously, she builds only when her cost is low).

**Player 2:** Let $y$ be the probability of player 2 for buying (player 2 does know nothing about the actual costs incurring for player 1).

**Costs high or low:** Let $p$ be the probability that the building costs for player 1 are high.

**Equilibrium $\langle x, y \rangle$:** We need to find pairs $\langle x, y \rangle$ that are stable in the following sense: $x$ is best for player 1 and $y$ is best for player 2, given probability $p$. 
Analysis of the game

- \( \langle 0, 1 \rangle \) is an equilibrium, independently of \( p \).
- \( \langle 1, 0 \rangle \) is an equilibrium iff \( p \leq 0.5 \).
- \( \left\langle \frac{1}{2(1-p)}, \frac{1}{2} \right\rangle \) is a (mixed) equilibrium iff \( p \leq 0.5 \).
5.2 Bayesian Games
Assumptions

**Uncertainty:** Only about the *payoffs*, not about the strategy spaces, number of players, actions available etc.

**Common-prior:** Agents have all sorts of beliefs about other agents, about their beliefs about other agents etc. This can be very difficult to model. We make the *common-prior assumption*, explained on the next slide.
The common-prior assumption is the simplifying assumption, that the probability distribution is fixed and known to the agents in advance. In Example 5.1, the probability distribution underlying \( p \) (whether the building costs are high or not) is uniform and known to both players.
The uncertainty assumption is not restrictive.

- Suppose player 1 does not know whether her opponent has two or three actions available:

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>(1, 2)</td>
<td>(3, 4)</td>
</tr>
<tr>
<td>D</td>
<td>(5, 6)</td>
<td>(7, 8)</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>(1, 2)</td>
<td>(3, 4)</td>
<td>(9, 10)</td>
</tr>
<tr>
<td>D</td>
<td>(5, 6)</td>
<td>(7, 8)</td>
<td>(11, 12)</td>
</tr>
</tbody>
</table>

- Then we define the following *padded* game in such a way, that the Nash equilibria are the same and the uncertainty is only in the payoffs:

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>(1, 2)</td>
<td>(3, 4)</td>
<td>(9, −100)</td>
</tr>
<tr>
<td>D</td>
<td>(5, 6)</td>
<td>(7, 8)</td>
<td>(11, −100)</td>
</tr>
</tbody>
</table>
Bayesian Game: Informal

A **Bayesian game** for $n$ agents consists of (1) a set $W$ of $n$-person games that differ in their payoffs, (2) a probability distribution over these games (the **common-prior**), and (3) a set of $n$ **partitions** $I_1, \ldots, I_n$ of $W$. $I_i$ is the set of potential games that agent $i$ considers possible (can’t distinguish, are equivalent for her etc.).

A partition $I_i$ is a set of subsets $W_{i1}, \ldots, W_{is}$ of $W$ such that: (1) $\bigcup_j W_{ij} = W$ and (2) $W_{ij} \cap W_{ij'} = \emptyset$ for $j \neq j'$.

$I_i$ is the **information** that agent $i$ has about the game. $i$ does not know which game is being played, but it considers all games in one equivalence class (one of the $W_{ij}$) as **indistinguishable**. Therefore a strategy is determined **per partition class**.
Definition 5.3 (*n*-Person Bayesian Game)

A finite *n*-person Bayesian game is a tuple \( \langle \mathcal{A}, G, P, I \rangle \), where

- \( \mathcal{A} = \{1, \ldots, i, \ldots, n\} \) is a finite set of players.
- \( G \) is a set of *n*-person normal form games, each of which has the same action profiles \( A = \langle A_1, \ldots, A_i, \ldots A_n \rangle \) where \( A_i \) is the set of actions available to player \( i \).
- \( P \) is a probability distribution over the set \( G \) of all games (the set of all distributions is denoted by \( \Pi(G) \)).
- \( I = \{I_1, \ldots, I_n\} \) is a set of partitions of \( G \).
### 5 Incomplete Information Games

#### 5.2 Bayesian Games

<table>
<thead>
<tr>
<th></th>
<th>$I_{1,1}$</th>
<th>$I_{2,1}$</th>
<th>$I_{2,2}$</th>
<th>$I_{1,2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\begin{array}{c</td>
<td>c</td>
<td>c} 1 &amp; 0 &amp; \hline 0 &amp; 3 \end{array}$</td>
<td>$\begin{array}{c</td>
</tr>
<tr>
<td></td>
<td>$p = 0.2$</td>
<td>$p = 0.1$</td>
<td>$p = 0.1$</td>
<td>$p = 0$</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\begin{array}{c</td>
<td>c</td>
<td>c} 3 &amp; 2 &amp; \hline 0 &amp; 3 \end{array}$</td>
<td>$\begin{array}{c</td>
</tr>
<tr>
<td></td>
<td>$p = 0.1$</td>
<td>$p = 0.1$</td>
<td>$p = 0.25$</td>
<td>$p = 0.15$</td>
</tr>
</tbody>
</table>
5.3 Bayes Nash equilibrium
Strategies for Bayesian Games

What is a strategy for agent $i$ in a Bayesian game? Clearly, a strategy must be compatible with the information the agent has about the game, i.e. it is a function from the set of partition classes into the set of (mixed) strategies of the normal form games in $G$:

$$s_i : I_i \rightarrow \Pi(A_i).$$
Definition 5.4 (Bayes-Nash Equilibrium)

Given a Bayesian game \( \langle A, G, P, I \rangle \) a strategy profile \( s^* = \langle s_1^*, s_2^*, \ldots, s_n^* \rangle \) is a **Bayes-Nash equilibrium** if for each agent \( i \) the following holds

\[
    s_i^* \in \arg\max_{s_i \in S_i} \sum_{r \in I_i} \sum_{g \in G} u_{g,i}^{\text{expected}} (s_{-i}^*, s_i) P(g|r)
\]

\( u_{g,i}^{\text{expected}} \) is the expected utility function for agent \( i \) in game \( g \).
Figure 23: A Bayesian game consisting of zero-sum games
What are the equilibria in the game depicted in Figure 23?

- The players have 4 different strategies: two partition classes and again 2 for the $2 \times 2$ games. Let player 1’s strategies for the underlying games be $U$ and $D$ and players 2’s strategies $L$ and $R$.

- Then they have to choose which strategies to play in their two partition classes. Thus player 1 can play $UU$, $UD$, $DU$, or $DD$: the first symbol stands for $I_{11}$, the second for $I_{12}$.

- Analogously, we get $LL$, $LR$, $RL$, or $RR$ for player 2.

What is the reduced $4 \times 4$ normal form game?
## Associated normal form game

<table>
<thead>
<tr>
<th></th>
<th>LL</th>
<th>LR</th>
<th>RL</th>
<th>RR</th>
</tr>
</thead>
<tbody>
<tr>
<td>UU</td>
<td>1.3</td>
<td>1.45</td>
<td>1.1</td>
<td>1.25</td>
</tr>
<tr>
<td>UD</td>
<td>1.8</td>
<td>1.65</td>
<td>1.8</td>
<td>1.65</td>
</tr>
<tr>
<td>DU</td>
<td>1.1</td>
<td>0.7</td>
<td>2.0</td>
<td>1.95</td>
</tr>
<tr>
<td>DD</td>
<td>1.5</td>
<td>1.15</td>
<td>2.8</td>
<td>2.35</td>
</tr>
</tbody>
</table>

**Exercise:** Show how to obtain these values.
Reduced game: The unique NE in Figure 371 is $\langle UD, LR \rangle$.

Bayesian game: Therefore the Bayes-Nash Equilibrium of the original game is $\langle U, L \rangle$. 
Another Definition of Bayesian Game

- We can also define a Bayesian game by introducing an additional agent (God, Nature) that does all the probabilistic choices beforehand.
- God is the first player and then all the original players come in.
- We have thus reduced a Bayesian game to an extensive game with imperfect information: see Figure 24.
5 Incomplete Information Games

5.3 Bayes Nash equilibrium

Figure 24: Bayesian Game in Extensive Form with Nature-Agent
Figure 25: Another Bayesian Game
Figure 26: Bayesian Game with Nature-Agent
5.4 Bayesian equilibria and mixed equilibria
5.5 References
*Game Theory.*
MIT Press.

Equilibrium points in n-person games. 
*Proceedings of the National Academy of Sciences of the United States of America* 36, 48–49.

*A Course in Game Theory.*
MIT Press.
6. Mechanism Design

- Examples and Motivation
- MD for Voting procedures
In previous chapters we considered particular games and defined equilibria for them. We also defined voting mechanisms/auctions and noted that agents could lie in order to achieve their goals. In this chapter we consider the question how should a mechanism be designed so as to maximize the principal’s expected utility, even if the agents do not act truthfully? This area is called Mechanism Design (MD).

So we design the rules of the game to maximize our utility, even if all agents act in self-interest.

- Well-known examples are the various types of auctions.
- We first consider MD for voting procedures. The Gibbard/Satterthwaite theorem is similar to Arrow’s theorem and deals with non-manipulable voting systems.
- Then we consider MD for Bayessian games. But we only consider solutions for the case of a single agent.
6.1 Examples and Motivation
Lying and manipulation

- What if agents vote \textbf{tactically}? I.e. they know the voting design and do not vote \textbf{truthfully}, but such that their preferred choice is elected after all?
- Is it possible to come up with an \textbf{implementation} of a social choice function (or correspondence) that \textbf{can not be manipulated}?
Example 6.1 (Babysitting, Shoham)

You are babysitting 4 kids (Will, Liam, Vic and Ray). They can choose among (a: going to the video arcade, b: playing baseball, c: going for a leisurely car ride). The kids give their true preferences as follows:

<table>
<thead>
<tr>
<th></th>
<th>Will</th>
<th>Liam</th>
<th>Vic</th>
<th>Ray</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>b</td>
<td>b</td>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td>2</td>
<td>a</td>
<td>a</td>
<td>c</td>
<td>a</td>
</tr>
<tr>
<td>3</td>
<td>c</td>
<td>c</td>
<td>b</td>
<td>b</td>
</tr>
</tbody>
</table>

Majority voting, breaking ties alphabetically.
Suppose Ray hates playing basketball but knows that his fellows do like it. How can he vote to avoid ending up playing basketball?

Note that the other kids do not disclose their preferences, but Ray might know them. Other kids may have other preferences, so choosing “a” is not a dominant strategy.
Definition 6.2 (Mechanism)

A mechanism wrt. a set of agents $\mathcal{A}$ and a set of outcomes $O$ is a pair $\langle \text{Act}, \mathcal{M} \rangle$ where

1. $\text{Act} = A_1 \times \ldots \times A_{|\mathcal{A}|}$, where $A_i$ is the set of actions available to agent $i$.
2. $\mathcal{M} : \text{Act} \rightarrow \Pi(O)$, where $\Pi(O)$ is the set of all probability distributions over the set of outcomes.

This is a very general definition: it allows arbitrary actions. What about our babysitter example?
Relation between a mechanism and a game?

Remember Definition 1.4. For the case that $M(\langle a_1, \ldots, a_|\mathcal{A}| \rangle)(o) \in \{0, 1\}$ for all action profiles and outcomes $o$, we can view $M$ as a function $\varrho$ as in Definition 1.4: $M(\langle a_1, \ldots, a_|\mathcal{A}| \rangle)(o) = 1$ becomes $M(\langle a_1, \ldots, a_|\mathcal{A}| \rangle) = o$. Then

**Mechanisms and games**

If $\mu$ is a utility profile and $\langle \text{Act}, M \rangle$ is a mechanism wrt. $\mathcal{A}$ and $O$, then $\langle \mathcal{A}, \text{Act}, O, M, \mu \rangle$ is a $|\mathcal{A}|$-person normal form game.
6.2 MD for Voting procedures
Mechanism vs. Social Choice Function

What is the difference between a social choice function and a mechanism?

- Social choice function is often considered to be **truthful**: it is based on the real preferences.
- A mechanism is an **implementation** (or not!) of a social choice function.
- MD is also called **inverse game theory or incentive engineering**.
Definition 6.3 (Mechanism/Implementation)

Let $A$ and $O$ be fixed. Let a social choice function $C^*$ be given. We say that a mechanism $\langle Act, M \rangle$ implements the function $C^*$ in dominant strategies, if for all utility profiles $\mu$ the game $\langle A, A, O, M, \mu \rangle$ has an equilibrium in dominant strategies and for any equilibrium $\langle s_1^*, s_2^*, \ldots, s_n^* \rangle$ we have:

$$M(\langle s_1^*, s_2^*, \ldots, s_n^* \rangle) = C^*(\mu)$$
Implementation in dominant strategies

What about our example? Is there a mechanism implementing our social choice function in dominant strategies?

Exercise

What if we redefine the last definition in terms of Nash-equilibria? Is there a mechanism implementing our social choice function in Nash-equilibria?
Definition 6.4 (Bayes-Nash Implementation)

Let $\mathcal{A}$ and $O$ be fixed. Let $U$ be the set of all utility profiles $\mu$ over $O$. Let $G$ be the set of games $\langle \mathcal{A}, A, O, M, \mu \rangle$ for $\mu \in U$. Let $P$ be a probability distribution on $U$ and let $I = \{I_1, \ldots, I_n\}$ be a set of partitions, one for each agent. Finally, let a social choice function $C^*$ be given.

We say that a mechanism $\langle \text{Act}, M \rangle$ implements the function $C^*$ in Bayes-Nash equilibria wrt. $P$ and $I$, if there exists a Bayes-Nash equilibrium of the Bayesian game $\langle \mathcal{A}, G, P, I \rangle$, such that for each game $g \in G$ and each action profile $\langle a_1, \ldots, a_n \rangle \in \text{Act}$ that can arise in $g$, it holds that $M(\langle a_1, \ldots, a_n \rangle) = C^*(\mu)$. 
Lying

There are situations, where one does not want to reveal the true preferences. *Lying might pay off*: not only to get the desired result, but also to ensure that critical information is not disclosed.
Often, mechanisms are used that are much more restricted.

### Definition 6.5 (Direct Mechanism)

A **direct mechanism** wrt. a set of agents $A$ and a set of outcomes $O$ is a mechanism $\langle \text{Act}, M \rangle$ with

$$A_i = \{ \mu_i : \mu_i \text{ are utility functions} \}.$$ 

But: agents may lie and not reveal their true utilities.
Truthful or strategy-proof mechanism

A mechanism is **truthful** (or **strategy-proof**) in dominant strategies, if for any utility profile, in the resulting game it is a dominant strategy for each agent to announce its true utility function.

Theorem 6.6 (Revelation)

*If there exists an implementation of a social choice rule in dominant strategies, then there is also a direct and truthful mechanism implementing the same function.*

The same theorem is also true for implementation in Nash equilibria (with the same proof).
Proof.

The proof is simple: one simply builds-in the lying-part into the procedure. That is, one lets the procedure do what is best for oneself.
Truthful or strategy-proof mechanism

A mechanism is **truthful** (or **strategy-proof**) in dominant strategies, if for any utility profile, in the resulting game it is a **dominant strategy** for each agent to **announce its true utility function**.

Theorem 6.7 (Revelation)

*If there exists an implementation of a social choice rule in dominant strategies, then there is also a direct and truthful mechanism implementing the same function.*

The same theorem is also true for implementation in Nash equilibria (with the same proof).
Proof.

The proof is simple: one simply builds-in the lying-part into the procedure. That is, one lets the procedure do what is best for oneself.
7. From Classical to Temporal Logics

- Sentential Logic (SL)
- First-Order Logic (FOL)
- Linear Time Logic (LTL)
- Branching Time Logic (CTL)
- References
Outline

- We recapitulate very briefly *sentential* (also called *propositional*) (SL) and *first-order logic* (FOL).

- As an example of FOL, we consider $\text{FO}(\leq)$: *monadic* FOL of linear order.

- Then we present LTL, a logic to deal with linear time (no branching).

- While LTL is *equivalent* to $\text{FO}(\leq)$, LTL is a more compact formalism and can be easily extended.
Outline (cont.)

- **CTL** is an extension of **LTL** to branching time.
- **CTL** is an interesting fragment of **CTL***, incomparable with **LTL**, but with interesting computational properties.
- While **LTL** is defined over path formulae, **CTL** is defined over state formulae.
- **CTL*** is defined over both sorts of formulae.
- We present a criterion to decide whether a **CTL*** formula is equivalent to a **LTL** formula.
7.1 Sentential Logic (SL)
# Syntax of SL

**Definition 7.1 (Sentential Logic $L_{SL}$, Lang. $L \subseteq L_{SL}$)**

The *language $L_{SL}$ of propositional (or sentential) logic* consists of:

- $p, q, r, x_1, x_2, \ldots x_n, \ldots$: a countable set $AT$ of $SL$-constants,
- $\neg, \lor$: the sentential connective ($\neg$ is unary, $\lor$ is binary),
- $(, )$: the parentheses to help readability.

In most cases we consider only a finite set of $SL$-constants. They define a language $L \subseteq L_{SL}$. The set of $L$-formulae $Fml_L$ is defined inductively.
7.1 Sentential Logic (SL)

Macros

\[
\begin{align*}
\top &:= p \lor \neg p \\
\bot &:= \neg \top \\
\varphi \land \psi &:= \neg (\neg \varphi \lor \neg \psi) \\
\varphi \rightarrow \psi &:= \neg \varphi \lor \psi \\
\varphi \leftrightarrow \psi &:= (\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi)
\end{align*}
\]
Definition 7.2 (Semantics, Valuation, Model)

A valuation $v$ for a language $\mathcal{L} \subseteq \mathcal{L}_{\text{SL}}$ is a mapping from the set of $\text{SL}$-constants defined by $\mathcal{L}$ into the set $\{\text{true, false}\}$. Each valuation $v$ can be uniquely extended to a function $\bar{v} : Fml_\mathcal{L} \rightarrow \{\text{true, false}\}$ so that:

- $\bar{v}(\neg p) = \begin{cases} \text{true, if } \bar{v}(p) = \text{false}, \\ \text{false, if } \bar{v}(p) = \text{true}. \end{cases}$

- $\bar{v}(\varphi \lor \gamma) = \begin{cases} \text{true, if } \bar{v}(\varphi) = \text{t or } \bar{v}(\gamma) = \text{true}, \\ \text{false, else} \end{cases}$
Definition (continued)

Thus each valuation \( \nu \) uniquely defines a \( \nu \)-structure. We call a structure determines for each formula if it is true or false. If a formula \( \phi \) is true in structure \( \nu \) we also say \( A_\nu \) is a model of \( \phi \). From now on we will speak of models, structures and valuations synonymously.

Semantics

The process of mapping a set of \( \mathcal{L} \)-formulae into \{true, false\} is called semantics.
**Definition 7.3 (Model, Theory, Tautology (Valid))**

1. A formula $\varphi \in Fml_L$ holds under the valuation $v$ if \( \bar{v}(\varphi) = \text{true} \). We also write $\bar{v} \models \varphi$ or simply $v \models \varphi$. $\bar{v}$ is a model of $\varphi$.

2. A **theory** is a set of formulae: $T \subseteq Fml_L$. $v$ satisfies $T$ if $\bar{v}(\varphi) = \text{true}$ for all $\varphi \in T$. We write $v \models T$.

3. A $L$-formula $\varphi$ is called **$L$-tautology** (or simply called **valid**) if for all possible valuations $v$ in $L$ $v \models \varphi$ holds.

From now on we suppress the language $L$ when obvious from context.
**Truth Tables**

Truth tables are a conceptually simple way of working with PL (invented by Wittgenstein in 1918).

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$\neg p$</th>
<th>$p \lor q$</th>
<th>$p \land q$</th>
<th>$p \rightarrow q$</th>
<th>$p \leftrightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>t</td>
<td>t</td>
<td>f</td>
<td>t</td>
<td>t</td>
<td>t</td>
<td>t</td>
</tr>
<tr>
<td>f</td>
<td>t</td>
<td>t</td>
<td>t</td>
<td>f</td>
<td>t</td>
<td>f</td>
</tr>
<tr>
<td>t</td>
<td>f</td>
<td>f</td>
<td>t</td>
<td>f</td>
<td>f</td>
<td>f</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>t</td>
<td>f</td>
<td>f</td>
<td>t</td>
<td>t</td>
</tr>
</tbody>
</table>
Fundamental Semantical Concepts

- If it is possible to find some valuation \( v \) that makes \( \varphi \) true, then we say \( \varphi \) is **satisfiable**.
- If \( v \models \varphi \) for all valuations \( v \) then we say that \( \varphi \) is **valid** and write \( \models \varphi \). \( \varphi \) is also called **tautology**.
- A **theory** is a set of formulae: \( \Phi \subseteq \mathcal{L}_{PL} \).
- A theory \( \Phi \) is called **consistent** if there is a valuation \( v \) with \( v \models \Phi \).
- A theory \( \Phi \) is called **complete** if for each formula \( \varphi \) in the language, \( \varphi \in \Phi \) or \( \neg \varphi \in \Phi \).

**Two simple examples**

Consider the two formulae \( p \land \neg b \) and \( a \lor \neg a \).
- Are they **satisfiable** or **valid**?
- Are they both **consistent**? What if we add \( b \)?
Consequences

Given a theory $\Phi$ we are interested in the following question: **Which facts can be derived from $\Phi$?** We can distinguish two approaches:

1. **Semantical** consequences, and
2. **Syntactical** inference.

Let $\Phi$ be a theory and $\varphi$ be a formula. We say that $\varphi$ is a **semantical consequence of** $\Phi$ if for all valuations $v$:

$$ v \models \varphi \quad \text{if} \quad v \models \Phi \implies v \models \varphi. $$
7.2 First-Order Logic (FOL)
Predicate logic

In addition to the propositional language (on which the modal language is built as well), the first-order language (FOL) contains variables, function-, and predicate symbols.

**Definition 7.4 (Variable)**

A variable is a symbol of the set $\text{Var}$. Typically, we denote variables by $x_0, x_1, \ldots$

**Example 7.5**

$$\varphi := \exists x_0 \forall x_1 (P_0^2(f_0^1(x_0), x_1) \land P_2^1(f_1^0))$$
Definition 7.6 (Function symbols)

Let \( k \in \mathbb{N}_0 \). The set of \( k \)-ary function symbols is denoted by \( \text{Func}^k \). Elements of \( \text{Func}^k \) are given by \( f_1^k, f_2^k \ldots \). Such a symbol takes \( k \) arguments. The set of all function symbols is defined as

\[
\text{Func} := \bigcup_k \text{Func}^k
\]

A 0-ary function symbol is called constant.
Definition 7.7 (Predicate Symbols)

Let $k \in \mathbb{N}_0$. The set of $k$-ary predicate symbols (or relation symbols) is given by $\mathcal{P}red^k$. Elements of $\mathcal{P}red^k$ are denoted by $P^k_1, P^k_2, \ldots$. Such a symbol takes $k$ arguments. The set of predicate symbols is defined as

$$\mathcal{P}red := \bigcup_k \mathcal{P}red^k$$

A 0-ary predicate symbol is called (atomic) proposition.
Syntax

The **first-order language with equality** $\mathcal{L}_{\text{FOL}}$ is built from terms and formulae.

In the following we fix a set of variables, function-, and predicate symbols.

**Definition 7.8 (Term)**

A **term** over $\mathcal{F}_{\text{unc}}$ and $\mathcal{V}_{\text{ar}}$ is inductively defined as follows:

1. Each variable from $\mathcal{V}_{\text{ar}}$ is a term.
2. If $t_1, \ldots, t_k$ are terms then $f^k(t_1, \ldots, t_k)$ is a term as well, where $f^k$ is an $k$-ary function symbol from $\mathcal{F}_{\text{unc}}^k$. 
The first-order language with equality $L_{FOL}(\mathcal{Var}, \mathcal{Func}, \mathcal{Pred})$ is defined by the following grammar:

$$\varphi ::= P^k(t_1, \ldots, t_k) \mid \neg \varphi \mid \varphi \lor \varphi \mid \exists x(\varphi) \mid t \equiv r$$

where $P^k \in \mathcal{Pred}^k$ is a $k$-ary predicate symbol and $t_1, \ldots, t_k$ and $t, r$ are terms over $\mathcal{Var}$ and $\mathcal{Func}$. 
### Definition 7.10 (Macros)

We define the following syntactic constructs as macros ($P \in \text{Pred}^0$):

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bot$</td>
<td>$P \land \neg P$</td>
</tr>
<tr>
<td>$\top$</td>
<td>$\neg \bot$</td>
</tr>
<tr>
<td>$\varphi \land \psi$</td>
<td>$\neg(\neg \varphi \lor \neg \psi)$</td>
</tr>
<tr>
<td>$\varphi \rightarrow \psi$</td>
<td>$\neg \varphi \lor \psi$</td>
</tr>
<tr>
<td>$\varphi \leftrightarrow \psi$</td>
<td>$(\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi)$</td>
</tr>
<tr>
<td>$\forall x(\varphi)$</td>
<td>$\neg \exists x(\neg \varphi)$</td>
</tr>
</tbody>
</table>
Notation

- We will often leave out the index \( k \) in \( f_i^k \) and \( P_i^k \) indicating the arity and just write \( f_i \) and \( P_i \).
- Variables are also denoted by \( u, v, w, \ldots \)
- Function symbols are also denoted by \( f, g, h, \ldots \)
- Constants are also denoted by \( a, b, c, \ldots, c_0, c_1, \ldots \)
- Predicate symbols are also denoted by \( P, Q, R, \ldots \)
- We will use our standard notation \( p \) for 0-ary predicate symbols and also call them (atomic) propositions.

Attention

For linear temporal logic, we only need unary predicates (monadic logic) and we do not need any function symbols at all. So our terms are exactly the variables.
Definition 7.11 (Model, Structure)

A model or structure for FOL over $\text{Var}$, $\text{Func}$ and $\text{Pred}$ is given by $\mathcal{M} = (U, I)$ where

1. $U$ is a non-empty set of elements, called universe or domain and
2. $I$ is called interpretation. It assigns to each function symbol $f^k \in \mathcal{F}unc^k$ a function $I(f^k) : U^k \to U$, to each predicate symbol $P^k \in \mathcal{P}red^k$ a relation $I(P^k) \subseteq U^k$; and to each variable $x \in \text{Var}$ an element $I(x) \in U$.

We write:

1. $\mathcal{M}(P^k)$ for $I(P^k)$,
2. $\mathcal{M}(f^k)$ for $I(f^k)$, and
3. $\mathcal{M}(x)$ for $I(x)$. 
Note that a **structure** comes with an interpretation $I$, which is based on functions and predicate symbols and assignments of the variables. But these are also defined in the notion of a language. Thus we assume from now on that the structures are **compatible** with the underlying language: The arities of the functions and predicates must correspond to the associated symbols.
Example 7.12

\( \varphi := Q(x) \lor \forall z(P(x, g(z))) \lor \exists x(\forall y(P(f(x), y) \land Q(a))) \)

- \( U = \mathbb{R} \)
- \( I(a) : \{\emptyset\} \rightarrow \mathbb{R}, \emptyset \mapsto \pi \) constant functions,
- \( I(f) : I(f) = \sin : \mathbb{R} \rightarrow \mathbb{R} \) and \( I(g) = \cos : \mathbb{R} \rightarrow \mathbb{R} \),
- \( I(P) = \{(r, s) \in \mathbb{R}^2 : r \leq s\} \) and \( I(Q) = [3, \infty) \subseteq \mathbb{R} \),
- \( I(x) = \frac{\pi}{2}, I(y) = 1 \) and \( I(z) = 3 \).
Definition 7.13 (Value of a Term)

Let $t$ be a term and $\mathcal{M} = (U, I)$ be a model. We define inductively the **value of $t$ wrt $\mathcal{M}$**, written as $\mathcal{M}(t)$, as follows:

- $\mathcal{M}(x) := I(x)$ for a variable $t = x$,
- $\mathcal{M}(t) := I(f^k)(\mathcal{M}(t_1), \ldots, \mathcal{M}(t_k))$ if $t = f^k(t_1, \ldots, t_k)$. 

Definition 7.14 (Semantics)

Let $\mathcal{M} = (U, I)$ be a model and $\varphi \in \mathcal{L}_{FOL}$. $\varphi$ is said to be true in $\mathcal{M}$, written as $\mathcal{M} \models \varphi$, if the following holds:

- $\mathcal{M} \models P^k(t_1, \ldots t_k)$ iff $(\mathcal{M}(t_1), \ldots, \mathcal{M}(t_k)) \in \mathcal{M}(P^k)$
- $\mathcal{M} \models \neg \varphi$ iff not $\mathcal{M} \models \varphi$
- $\mathcal{M} \models \varphi \lor \psi$ iff $\mathcal{M} \models \varphi$ or $\mathcal{M} \models \psi$
- $\mathcal{M} \models \exists x(\varphi)$ iff $\mathcal{M}[x/a] \models \varphi$ for some $a \in U$ where $\mathcal{M}[x/a]$ denotes the model equal to $\mathcal{M}$ but $\mathcal{M}[x/a](x) = a$.
- $\mathcal{M} \models t \doteq r$ iff $\mathcal{M}(t) = \mathcal{M}(r)$

Given a set $\Sigma \subseteq \mathcal{L}_{FOL}$ we write $\mathcal{M} \models \Sigma$ iff $\mathcal{M} \models \varphi$ for all $\varphi \in \Sigma$. 
Example: FO(≤)

**Monadic** first-order logic of order, denoted by FO(≤), is first-order logic with the only binary symbol ≤ (except equality, which is also allowed) and, additionally, any number of **unary predicates**. The theory assumes that ≤ is a **linear order with least element**, but nothing else.

A typical model is given by

\[ \mathcal{N} = \langle \mathbb{N}_0, \leq_{\mathbb{N}_0}, P_1^\mathcal{N}, P_2^\mathcal{N}, \ldots, P_n^\mathcal{N} \rangle \]

where \( \leq_{\mathbb{N}_0} \) is the usual ordering on the natural numbers and \( P_i^\mathcal{N} \subseteq \mathbb{N}_0 \).

The sets \( P_i^\mathcal{N} \) determine the **timepoints** where the property \( P_i \) holds.
What can we express in FO(≤)?

Can we find formulae expressing that:

- a property $r$ is true **infinitely often**?
- whenever $r$ is true, then $s$ is true in the **next timepoint**?
- $r$ is true at all **even timepoints** and $\neg r$ at all **odd timepoints**?
7.3 Linear Time Logic (LTL)
Reasoning about Time

- The accessibility relation represents time.
- Time: linear vs. branching.
- Reasoning about a particular computation of a system.
- Models: paths (e.g. obtained from Kripke structures)
Temporal logic was originally developed in order to represent tense in natural language.

Within Computer Science, it has achieved a significant role in the formal specification and verification of concurrent and distributed systems.

Much of this popularity has been achieved because a number of useful concepts can be formally, and concisely, specified using temporal logics, e.g.

- safety properties
- liveness properties
- fairness properties
## Typical temporal operators

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathbf{X} \varphi$</td>
<td>$\varphi$ is true in the next moment in time</td>
</tr>
<tr>
<td>$\mathbf{G} \varphi$</td>
<td>$\varphi$ is true globally: in all future moments</td>
</tr>
<tr>
<td>$\mathbf{F} \varphi$</td>
<td>$\varphi$ is true finally: eventually (in the future)</td>
</tr>
<tr>
<td>$\varphi U \psi$</td>
<td>$\varphi$ is true until at least the moment when $\psi$ becomes true (and this eventually happens)</td>
</tr>
</tbody>
</table>

- $\mathbf{G}((\neg \text{passport} \lor \neg \text{ticket}) \rightarrow \mathbf{X}(\neg \text{board\_flight}))$
- $\mathbf{send}(msg,r\text{cvr}) \rightarrow \mathbf{F} \mathbf{receive}(msg,r\text{cvr})$
Safety Properties

“something bad will not happen”
“something good will always hold”

Typical examples:

\( \mathsf{G} \neg \text{bankrupt} \)
\( \mathsf{G} \text{fuelOK} \)
and so on . . .

Usually: \( \mathsf{G} \neg . . . \)
Liveness Properties

“something good will happen”

Typical examples:

Frich

power_on → Fonline

and so on ... 

Usually: F...
Fairness Properties

Combinations of safety and liveness possible:

\[ FG \neg \text{dead} \]
\[ G(\text{request\_taxi} \rightarrow F\text{arrive\_taxi}) \rightarrow \text{fairness} \]

Strong fairness

“If something is requested then it will be allocated”:

\[ G(\text{attempt} \rightarrow F\text{success}), \]
\[ GF\text{attempt} \rightarrow GF\text{success}. \]

- Scheduling processes, responding to messages, etc.
- No process is blocked forever, etc.
Definition 7.15 (Language $\mathcal{L}_{\text{LTL}}$ [Pnueli, 1977])

The language $\mathcal{L}_{\text{LTL}}(\text{Prop})$ is given by all formulae generated by the following grammar, where $p \in \text{Prop}$ is a proposition:

$$\varphi ::= p \mid \neg \varphi \mid \varphi \lor \varphi \mid \varphi U \varphi \mid X \varphi.$$ 

The additional operators

- $\mathbf{F}$ (eventually in the future) and
- $\mathbf{G}$ (always from now on)

can be defined as macros:

$$F\varphi \equiv \top U \varphi \quad \text{and} \quad G\varphi \equiv \neg F\neg \varphi$$

The standard Boolean connectives $\top, \bot, \land, \rightarrow, \leftrightarrow$ are defined in their usual way as macros.
Models of LTL

The semantics is given over paths, which are infinite sequences of states from $Q$, and a standard labelling function $\pi : Q \rightarrow \mathcal{P}(\text{Prop})$ that determines which propositions are true at which states.

**Definition 7.16 (Path $\lambda = q_1q_2q_3 \ldots$)**

- A path $\lambda$ over a set of states $Q$ is an denumerable infinite sequence from $Q$ into $\omega$. We also identify it with a mapping from $\mathbb{N}_0$ into $Q$.

- $\lambda[i]$ denotes the $i$th position on path $\lambda$ (starting from $i = 0$) and

- $\lambda[i, \infty]$ denotes the subpath of $\lambda$ starting from $i$ ($\lambda[i, \infty] = \lambda[i] \lambda[i + 1] \ldots$).
\[ \lambda = q_1 q_2 q_3 \ldots \in Q^\omega \]

**Definition 7.17 (Semantics of LTL)**

Let \( \lambda \) be a **path** and \( \pi \) be a **labelling function** over \( Q \). The semantics of LTL, \( \models_{LTL} \), is defined as follows:

- \( \lambda, \pi \models_{LTL} p \) iff \( p \in \pi(\lambda[0]) \) and \( p \in \text{Prop} \);
- \( \lambda, \pi \models_{LTL} \neg \varphi \) iff not \( \lambda, \pi \models_{LTL} \varphi \) (we also write \( \lambda, \pi \not\models_{LTL} \varphi \));
- \( \lambda, \pi \models_{LTL} \varphi \lor \psi \) iff \( \lambda, \pi \models_{LTL} \varphi \) or \( \lambda, \pi \models_{LTL} \psi \);
- \( \lambda, \pi \models_{LTL} X \varphi \) iff \( \lambda[1, \infty], \pi \models_{LTL} \varphi \); and
- \( \lambda, \pi \models_{LTL} \varphi U \psi \) iff there is an \( i \in \mathbb{N}_0 \) such that \( \lambda[i, \infty], \pi \models \psi \) and \( \lambda[j, \infty], \pi \models_{LTL} \varphi \) for all \( 0 \leq j < i \).
Other temporal operators

\[ \lambda, \pi \models F \varphi \iff \lambda[i, \infty], \pi \models \varphi \text{ for some } i \in \mathbb{N}_0 ; \]
\[ \lambda, \pi \models G \varphi \iff \lambda[i, \infty], \pi \models \varphi \text{ for all } i \in \mathbb{N}_0 ; \]

Exercise

Prove that the semantics does indeed match the definitions \( F \varphi \equiv T \varphi U \varphi \) and \( G \varphi \equiv \neg F \neg \varphi \).
\[ \lambda, \pi \models F_{\text{pos}_1} \]

\[ \lambda' = \lambda[1, \infty], \pi \models \text{pos}_1 \]

\[ \text{pos}_1 \in \pi(\lambda'[0]) \]
\[ \lambda, \pi \models GF_{pos_1} \iff \begin{align*}
\lambda[0, \infty], \pi & \models F_{pos_1} \text{ and } \\
\lambda[1, \infty], \pi & \models F_{pos_1} \text{ and } \\
\lambda[2, \infty], \pi & \models F_{pos_1} \text{ and } \\
& \ldots \end{align*} \]
Representation of paths

- Paths are **infinite entities**.
- They are theoretical constructs.
- We need a **finite representation**!
- Such a finite representation is given by a transition system or a pointed Kripke structure.
Computational vs. behavioral structure

System

Computational str.

pos_0

pos_1

pos_2

q_0

q_1

q_2
7 From Classical to Temporal Logics
7.3 Linear Time Logic (LTL)

Important!
The behavioral structure is usually infinite! Here, it is an infinite tree. We say it is the $q_0$-unfolding of the model.
Some Exercises

Example 7.18

Formalise the following as LTL formulae:

1. r should never occur.
2. r should occur exactly once.
3. At least once r should directly be followed by s.
Example 7.19

Formalise the following as LTL formulae:

1. \( r \) is true at exactly all even states.
2. \( r \) is true at all even state (the odd states do not matter). Does \( r \land G(r \rightarrow XXr) \) work?
Relation to first-order logic (1)

The monadic first-order theory of (linear) order, \( \text{FO}(\leq) \) (see Slide 424) is equivalent to LTL:

There is a translation from sentences of LTL to sentences of FO(\(\leq\)) and vice versa, such that the LTL sentence is true in \(\lambda, \pi\) iff its translation is true in the associated first-order structure.
Relation to first-order logic (2)

1 More precisely: an infinite path $\lambda$ is described as a first-order structure with domain $\mathbb{N}$ and predicates $P_p$ for $p \in Prop$. The predicates stand for the set of timepoints where $p$ is true. So each path $\lambda$ can be represented as a structure $\mathcal{N}_\lambda = \langle \mathbb{N}, \leq^{\mathbb{N}}, P_1^{\mathbb{N}}, P_2^{\mathbb{N}}, \ldots P_n^{\mathbb{N}} \rangle$.

Then each LTL formula $\phi$ translates to a first-order formula $\alpha_\phi(x)$ with one free variable s.t.

$\phi$ is true in $\lambda[n, \infty]$ iff $\alpha_\phi(n)$ is true in $\mathcal{N}_\lambda$.

And conversely: for each first-order formula with a free variable there is a corresponding LTL formula s.t. the same condition holds.

Does that mean that LTL is useless?
The formulae $\text{GF}_p$, $\text{FG}_p$

1. What are their counterparts in $\text{FO}(\leq)$?
2. We will see later that $\text{FG}_p$ does not even belong to $\text{CTL}$, but to $\text{CTL}^*$. It is not even equivalent to a $\text{CTL}$ formula.
3. However, $\text{GF}_p$ is equivalent to a $\text{CTL}$ formula: $\text{AGAF}_p$
Some Remarks

1. A particular logic LTL is determined by the number $n$ of propositional variables. Strictly speaking, this number should be a parameter of the logic. This also applies to the logics CTL and ATL (to be introduced in the next section).

2. While both $F$ and $G$ can be expressed using $U$, the converse is not true: $U$ cannot be expressed with $F$ and $G$. 
Satisfiability of LTL formulae

A formula is satisfiable, if there is a path where it is true. Can we restrict the structure of such paths? I.e. can we restrict to simple paths, for example paths that are periodic?

- If this is the case, then we might be able to construct counterexamples more easily, as we need only check very specific paths.

- It would be also useful to know how large the period is and within which initial segment of the path it starts, depending on the size of the formula $\phi$. 
Theorem 7.20 (Periodic model theorem [Sistla and Clarke, 1985])

A formula \( \varphi \in \mathcal{L}_{LTL} \) is **satisfiable** iff there is a path \( \lambda \) which is **ultimately periodic**, and the period starts within \( 2^{1+|\varphi|} \) steps and has a length which is \( \leq 4^{1+|\varphi|} \).
7.4 Branching Time Logic (CTL)
Branching Time

- **CTL, CTL**: Computation Tree Logics.
- Reasoning about possible computations of a system.
- Time is **branching**: We want all possible computations included!
- **Models**: states (time points, situations), transitions (changes). (⇝ Kripke models).
- **Paths**: courses of action, computations. (⇝ LTL)
Path quantifiers: $A$ (for all paths), $E$ (there is a path);

Temporal operators: $X$ (nexttime), $F$ (finally), $G$ (globally) and $U$ (until);

**CTL**: each temporal operator must be immediately preceded by exactly one path quantifier;

**CTL**: no syntactic restrictions.
Example 7.21 (Branching Time)

Whenever $p$ holds at some timepoint, then there is a path where $q$ holds in the next step and there is (another) path where $\neg q$ holds in the next step. And this holds along all paths (there are three infinite paths).
Definition 7.22 (\(\mathcal{L}_{\text{CTL}^*}\) [Emerson and Halpern, 1986])

The language \(\mathcal{L}_{\text{CTL}^*}(\text{Prop})\) is given by all formulae generated by the following grammar:

\[
\phi ::= p \mid \neg \phi \mid \phi \lor \phi \mid E \gamma
\]

where

\[
\gamma ::= \phi \mid \neg \gamma \mid \gamma \lor \gamma \mid \gamma U \gamma \mid X \gamma
\]

and \(p \in \text{Prop}\). Formulae \(\phi\) (resp. \(\gamma\)) are called state (resp. path) formulae.

We use the same abbreviations as for \(\mathcal{L}_{\text{LTL}}\):

\[
\lambda, \pi \models F \phi \text{ iff } \lambda[i, \infty], \pi \models \phi \text{ for some } i \in \mathbb{N}_0;
\]

\[
\lambda, \pi \models G \phi \text{ iff } \lambda[i, \infty], \pi \models \phi \text{ for all } i \in \mathbb{N}_0;
\]
The $\mathcal{L}_{CTL^*}$-formula $\text{EF}\varphi$, for instance, ensures that there is at least one path on which $\varphi$ holds at some (future) time moment.

The formula $\text{AFG}\varphi$ states that $\varphi$ holds almost everywhere. More precisely, on all paths it always holds from some future time moment.

$\mathcal{L}_{CTL^*}$-formulae do not only talk about temporal patterns on a given path, they also quantify (existentially or universally) over such paths.

The logic is complex! For practical purposes, a fragment with better computational properties is often sufficient.
Definition 7.23 ($\mathcal{L}_{\text{CTL}}$ [Clarke and Emerson, 1981])

The language $\mathcal{L}_{\text{CTL}}$ is given by all formulae generated by the following grammar, where $p \in \text{Prop}$ is a proposition:

$$\varphi ::= p \mid \neg \varphi \mid \varphi \lor \varphi \mid \mathcal{E}(\varphi \mathcal{U} \varphi) \mid \mathcal{E}\mathcal{X}\varphi \mid \mathcal{E}\mathcal{G}\varphi.$$

We introduce the following macros:

- $F\varphi \equiv T \mathcal{U} \varphi,$
- $A\mathcal{X}\varphi \equiv \neg \mathcal{E}\mathcal{X}\neg \varphi,$
- $A\mathcal{G}\varphi \equiv \neg \mathcal{E}F\neg \varphi,$ and
- $A\varphi \mathcal{U} \psi \equiv \ldots \text{ Exercise!}$
For example, $\text{AGEX}_p$ is a $\mathcal{L}_{\text{CTL}}$-formula whereas $\text{AGF}_p$ is not.

### Example 7.24 (CTL* or CTL?)

Are the following CTL* or CTL formulae? What do they express?

1. $\text{EFAX}_{\text{shutdown}}$
2. $\text{EFX}_{\text{shutdown}}$
3. $\text{AGF}_{\text{rain}}$
4. $\text{AGAF}_{\text{rain}}$ (Is it different from (3)?)
5. $\text{EFG}_{\text{broken}}$
6. $\text{AG}(p \rightarrow (\text{EX}q \land \text{EX}\neg q))$
The precise definition of Kripke structures is given in the next section. To understand the following definitions it suffices to note that:

- Given a set of states $Q$ (each is a propositional model), a **Kripke model** $M$ is simply a tuple $(Q, \mathcal{R})$ where $\mathcal{R} \subseteq Q \times Q$ is a binary relation.

- $q_1 \mathcal{R} q_2$ (also written $(q_1, q_2) \in \mathcal{R}$ or $\mathcal{R}(q_1, q_2)$) means that state $q_2$ is reachable from state $q_1$ (by executing certain actions).

- The relation $\mathcal{R}$ is **serial**: for all $q$ there is a $q'$ such that $q \mathcal{R} q'$. This ensures that our paths are infinite.

- Given a state $q$ in a Kripke model, by $\Lambda(q)$ we mean the set of all **paths** determined by the relation $\mathcal{R}$ starting in $q$: $q, q_1, q_2, \ldots, q_i, \ldots$ where $q \mathcal{R} q_1, \ldots, q_i \mathcal{R} q_{i+1}, \ldots$
Definition 7.25 (Semantics $\models_{\text{CTL}^*}$)

Let $M$ be a Kripke model, $q \in Q$ and $\lambda \in \Lambda$. The semantics of $L_{\text{CTL}^*}$- and $L_{\text{CTL}}$-formulae is given by the satisfaction relation $\models_{\text{CTL}^*}$ for state formulae by:

- $M, q \models_{\text{CTL}^*} p$ iff $\lambda[0] \in \pi(p)$ and $p \in \text{Prop}$;
- $M, q \models_{\text{CTL}^*} \neg \varphi$ iff $M, q \not\models_{\text{CTL}^*} \varphi$;
- $M, q \models_{\text{CTL}^*} \varphi \lor \psi$ iff $M, q \models_{\text{CTL}^*} \varphi$ or $M, q \models_{\text{CTL}^*} \psi$;
- $M, q \models_{\text{CTL}^*} \exists \varphi$ iff there is a path $\lambda \in \Lambda(q)$ such that $M, \lambda \models_{\text{CTL}^*} \varphi$. 


and for path formulae by:

- $\mathcal{M}, \lambda \models_{\text{CTL}^*} \varphi$ iff $\mathcal{M}, \lambda[0] \models_{\text{CTL}^*} \varphi$;
- $\mathcal{M}, \lambda \models_{\text{CTL}^*} \neg \gamma$ iff $\mathcal{M}, \lambda \not\models_{\text{CTL}^*} \gamma$;
- $\mathcal{M}, \lambda \models_{\text{CTL}^*} \gamma \lor \delta$ iff $\mathcal{M}, \lambda \models_{\text{CTL}^*} \gamma$ or $\mathcal{M}, \lambda \models_{\text{CTL}^*} \delta$;
- $\mathcal{M}, \lambda \models_{\text{CTL}^*} \Box \gamma$ iff $\lambda[1, \infty], \pi \models_{\text{CTL}^*} \gamma$; and
- $\mathcal{M}, \lambda \models_{\text{CTL}^*} \gamma U \delta$ iff there is an $i \in \mathbb{N}_0$ such that $\mathcal{M}, \lambda[i, \infty] \models_{\text{CTL}^*} \delta$ and $\mathcal{M}, \lambda[j, \infty] \models_{\text{CTL}^*} \gamma$ for all $0 \leq j < i$.

Is this complicated semantics over paths really necessary for CTL?
State-based semantics for CTL

- \( M, q \models_{\text{CTL}} p \) iff \( q \in \pi(p) \);
- \( M, q \models_{\text{CTL}} \neg \varphi \) iff \( M, q \not\models_{\text{CTL}} \varphi \);
- \( M, q \models_{\text{CTL}} \varphi \lor \psi \) iff \( M, q \models_{\text{CTL}} \varphi \) or \( M, q \models_{\text{CTL}} \psi \);
- \( M, q \models_{\text{CTL}} \text{EX} \varphi \) iff there is a path \( \lambda \in \Lambda(q) \) such that \( M, \lambda[1] \models_{\text{CTL}} \varphi \);
- \( M, q \models_{\text{CTL}} \text{EG} \varphi \) iff there is a path \( \lambda \in \Lambda(q) \) such that \( M, \lambda[i] \models_{\text{CTL}} \varphi \) for every \( i \geq 0 \);
- \( M, q \models_{\text{CTL}} \text{EU} \varphi \) iff there is a path \( \lambda \in \Lambda(q) \) such that \( M, \lambda[i] \models_{\text{CTL}} \psi \) for some \( i \geq 0 \), and \( M, \lambda[j] \models_{\text{CTL}} \varphi \) for all \( 0 \leq j < i \).
LTL as subset of CTL*

LTL is interpreted over *infinite chains* (infinite words), but not over (serial) Kripke structures (which are branching).

- To consider LTL as a subset of CTL*, one can just **add the quantifier $A$ in front of a LTL formula** and use the semantics of CTL*. For infinite chains, this semantics coincides with the LTL semantics.

- The theorem of *Clarke und Draghiescu* gives a nice characterization of those CTL* formulae that are **equivalent to LTL formulae**. Given a CTL* formula $\varphi$, we construct $\varphi'$ by just **removing all path operators** and putting A in front of it. Then

  $$\varphi \text{ is equivalent to a LTL formula iff }$$

  $$\varphi \text{ and } \varphi' \text{ are equivalent under the semantics of CTL*}.$$
Theorem 7.26 (Clarke und Draghiescu)

Given a $\text{CTL}^*$ formula $\varphi$, we construct $\hat{\varphi}$ by simply removing all path operators. Then the following are equivalent:

- $\varphi$ is equivalent to a $\text{LTL}$ formula,
- $\varphi$ and $A\hat{\varphi}$ are equivalent in $\text{CTL}^*$. 
Application of Clarke and Draghiescu

We consider the LTL formula $\text{GF}_p$. Viewed as a $\text{CTL}^*$ formula it becomes $\text{AGF}_p$. But this is equivalent (in $\text{CTL}^*$) to $\text{AGAF}_p$, a CTL formula.

Now we consider the CTL formula $\text{EGEF}_p$. It is not equivalent to any LTL formula. This is because $\text{EGEF}_p$ and $\text{AGF}_p$ are not equivalent in $\text{CTL}^*$: a counterexample is

![Diagram](image)

The first formula holds, the second does not.
LTL as subset of CTL* (2)

- **How do LTL and CTL compare?**

- The **CTL** formula $\text{AG}(p \rightarrow (\text{EX}q \land \text{EX}\neg q))$ describes Kripke structures of the form in Example 7.21. **No LTL formula** can describe this class of Kripke structures.

- The **LTL** formula $\text{AF}(p \land \text{X}p)$ **can not be expressed** by a **CTL** formula. Check why neither $\text{AF}(p \land \text{AX}p)$ nor $\text{AF}(p \land \text{EX}p)$ are equivalent. Similarly, the **LTL** formula $\text{AFG}p$ **can not be expressed** by a **CTL** formula.

- There is a syntactic characterisation of formulae expressible in both **CTL** and **LTL**. Model checking in this class can be done more efficiently. We refer to [Maidl 2000].
Example 7.27 (Robots and Carriage)

- **Two robots** push a carriage from opposite sides.
- Carriage can **move clockwise or anticlockwise**, or it can remain in the same place.
- **3 positions** of the carriage.
- We label the states with **propositions** $\text{pos}_0$, $\text{pos}_1$, $\text{pos}_2$, respectively, to allow for referring to the current position of the carriage in the object language.

Figure 27: Two robots and a carriage.
Figure 28: Two robots and a carriage: A schematic view (left) and a transition system $\mathcal{M}_0$ that models the scenario (right).
\[ M_0, q_0 \models^{CTL} EF_{pos_1} : \text{In state } q_0, \text{ there is a path such that the carriage will reach position 1 sometime in the future.} \]

\[ M_0, q_0 \not\models^{CTL} AF_{pos_1} : \text{The same is not true for all paths, so we also have:} \]

It becomes more interesting if abilities of agents are considered \(\rightsquigarrow\) ATL.
Example: Rocket and Cargo

Example 7.28 (Rocket and Cargo)

A cargo is moved between destinations by a rocket.

- The rocket can be moved between London (proposition roL) and Paris (proposition roP).
- The cargo can be in London (caL), Paris (caP), or inside the rocket (caR).
- The rocket can be moved only if it has its fuel tank full (fuelOK).
- When it moves, it consumes fuel, andnofuel holds after each flight.
Example: Rocket and Cargo (cont.)

\[
\text{roL} \rightarrow \text{EF} \text{roP}
\]

\[
\text{AG} (\text{roL} \lor \text{roP})
\]

\[
\text{roL} \rightarrow \text{AX} (\text{roP} \rightarrow \text{nofuel})
\]
Example: Rocket and Cargo (cont.)
In our logics, we assumed a **serial** accessibility relation: no **deadlocks** are possible.

One can also allow states with no outgoing transitions. In that case, in the semantical definition of $E$ on Slide 459 one has to replace “there is a path” by “**there is an infinite path or one which can not be extended**”.

Similar modifications are needed in the definition of $\text{CTL}$. One can also add to each state with no outgoing transitions a special transition leading to a new state that loops into itself.

### Expressibility

How to express that there is no possibility of a deadlock?

\[
\forall x (\Box x \rightarrow (\Diamond \top) \iff \text{CTL}\star)
\]

\[
\forall x (\Box x \lor (\Diamond \top) \iff \text{CTL})
\]
A Venn diagram showing typical formulae in the respective areas.
7.5 References
Design and synthesis of synchronization skeletons using branching time temporal logic.

Chapter 14: Specification and Verification of Multi-agent Systems.
In G. Weiss (Ed.), *Multiagent Systems*, MIT Press.

Temporal and modal logic.

“Sometimes” and “Not Never” Revisited: On Branching versus Linear Time Temporal Logic.


8. Strategic Logics

- Alternating-Time Temporal Logic (ATL)
- Perfect vs. imperfect recall
- Imperfect Information
- Defining Equilibria in ATLP
We introduce **ATL**, *Alternating Time Temporal Logic*: a blend of *temporal logic* and *game theory*.

Like **CTL**, **ATL** comes in two variants: **ATL** and **ATL**$^*$. Appropriate models for **ATL** are *concurrent game structures*.

We introduce four variants of **ATL** along two different axes:  
- **perfect vs imperfect information**, and  
- **perfect vs imperfect recall**.
8.1 Alternating-Time Temporal Logic (ATL)
Alternating-time Temporal Logics

- **ATL, ATL**\* [Alur et al. 1997]
- **Temporal logic** meets **game theory**
- Modeling abilities of **multiple agents**
- Main idea: **cooperation modalities**

\[\langle A \rangle \varphi: \text{coalition } A \text{ has a collective strategy to bring about } \varphi\]

**Bringing about** is understood in the game-theoretical sense: There is a **winning strategy**.
The syntax is given as for the computation-tree logics.

**Definition 8.1 (Language \( \mathcal{L}_{\text{ATL}}^* \) [Alur et al., 1997])**

The language \( \mathcal{L}_{\text{ATL}}^* \) is given by all formulae generated by the following grammar:

\[
\phi ::= p \mid \neg \phi \mid \phi \lor \phi \mid \langle A \rangle \gamma \quad \text{where}
\]

\[
\gamma ::= \phi \mid \neg \gamma \mid \gamma \lor \gamma \mid \gamma U \gamma \mid \Diamond \gamma,
\]

\( A \subseteq \text{Agt} \), and \( p \in \text{Prop} \). Formulae \( \phi \) (resp. \( \gamma \)) are called **state** (resp. **path**) formulae.

**Notation:** \( \Diamond \) instead of \( X \)

Note that we are now using the symbol “\( \Diamond \)” instead of “\( X \)” as it is more custom when dealing with \( \text{ATL} \).
The language $\mathcal{L}_{\text{ATL}}$ restricts $\mathcal{L}_{\text{ATL}*}$ in the same way as $\mathcal{L}_{\text{CTL}}$ restricts $\mathcal{L}_{\text{CTL}*}$:

Each temporal operator must be directly preceded by a cooperation modality.

**Definition 8.2** (Language $\mathcal{L}_{\text{ATL}}$ [Alur et al., 1997])

The language $\mathcal{L}_{\text{ATL}}$ is given by all formulae generated by the following grammar:

$$\varphi ::= p \mid \neg \varphi \mid \varphi \lor \varphi \mid \langle\langle A\rangle\rangle \lozenge \varphi \mid \langle\langle A\rangle\rangle \square \varphi \mid \langle\langle A\rangle\rangle \varphi U \varphi$$

where $A \subseteq \text{Agt}$ and $p \in \mathcal{P}_{\text{rop}}$.

**Notation:** $\square$ instead of $\Box$

Note that we are using now the symbol “$\square$” instead of “$\Box$” as it is more custom when dealing with ATL.
The language $\mathcal{L}_{\text{ATL}^+}$ restricts $\mathcal{L}_{\text{ATL}^*}$ but extends $\mathcal{L}_{\text{ATL}}$. It allows for Boolean combinations of path formulae.

**Definition 8.3 (Language $\mathcal{L}_{\text{ATL}^+}$)**

The language $\mathcal{L}_{\text{ATL}^+}$ is given by all formulae generated by the following grammar:

$$
\varphi ::= p \mid \neg \varphi \mid \varphi \lor \varphi \mid \langle A \rangle \gamma, \quad \gamma ::= \neg \gamma \mid \gamma \lor \gamma \mid \Box \varphi \mid \varphi U \varphi.
$$

where $A \subseteq \text{Agt}$ and $p \in \text{Prop}$. 
ATL Models: Concurrent Game Structures

- **Agents, actions, transitions, atomic propositions**
- Atomic propositions + **interpretation**
- **Actions are abstract**
Definition 8.4 (Concurrent Game Structure)

A concurrent game structure is a tuple \( M = \langle \text{Agt}, Q, \pi, \text{Act}, d, o \rangle \), where:

- \( \text{Agt} \): a finite set of all agents;
- \( Q \): a set of states;
- \( \pi : Q \rightarrow \mathcal{P}(\text{Prop}) \): a valuation of propositions;
- \( \text{Act} \): a finite set of (atomic) actions;
- \( d : \text{Agt} \times Q \rightarrow \mathcal{P}(\text{Act}) \) defines actions available to an agent in a state;
- \( o \): a deterministic transition function that assigns outcome states \( q' = o(q, \alpha_1, \ldots, \alpha_k) \) to states and tuples of actions.
Recall and information

A **strategy** of agent \( a \) is a **conditional plan** that specifies what \( a \) is going to do in each situation.

Two types of “situations”: Decisions are based on

- the **current state** only (\( \leadsto \) **memoryless strategies**)
  \[
  s_a : Q \rightarrow \text{Act}.
  \]
- on the **whole history** of events that have happened (\( \leadsto \) **perfect recall strategies**)
  \[
  s_a : Q^+ \rightarrow \text{Act}.
  \]
We also distinguish between agents with

- **perfect information** (all states are distinguishable).
- **imperfect information** (some states are indistinguishable).
Perfect Information Strategies

Definition 8.5 (IR- and Ir-strategies)

- A **perfect information perfect recall strategy** for agent $a$ (IR-strategy for short) is a function
  
  $$s_a : Q^+ \rightarrow \text{Act such that } s_a(q_0q_1 \ldots q_n) \in d_a(q_n).$$

  The set of such strategies is denoted by $\Sigma_a^{IR}$.

- A **perfect information memoryless strategy** for agent $a$ (Ir-strategy for short) is given by a function
  
  $$s_a : Q \rightarrow \text{Act where } s_a(q) \in d_a(q).$$

  The set of such strategies is denoted by $\Sigma_a^{Ir}$.

$i$ (resp. $I$) stands for **imperfect** (resp. **perfect** information) and $r$ (resp. $R$) for **imperfect** (resp. **perfect** recall). [Schobbens, 2004]
Some Notation

The following holds for all kind of strategies:

- A **collective strategy** for a group of agents $A = \{a_1, \ldots, a_r\} \subseteq \text{Agt}$ is a set $s_A = \{s_a \mid a \in A\}$ of strategies, one per agent from $A$.
- $s_A|_a$, we denote agent $a$’s part of the collective strategy $s_A$, $s_A|_a = s_A \cap \Sigma_a$.
- $s_\emptyset = \emptyset$ denotes the strategy of the empty coalition.
- $\Sigma_A$ denotes the **set of all collective strategies** of $A$.
- $\Sigma = \Sigma_{\text{Agt}}$
Outcome of a strategy

\[ \text{out}(q, s_A) = \text{set of all paths that may occur} \]
when agents \( A \) execute \( s_A \) from state \( q \) onward.

**Definition 8.6 (Outcome)**

\[ \lambda = q_0 q_1 \ldots \in Q \in \text{out}(q, s_A) \subseteq Q^\omega \text{ iff} \]

1. \( q_0 = q \)
2. for each \( i = 1, \ldots \) there is a tuple \( (\alpha^{i-1}_1, \ldots, \alpha^{i-1}_k) \in Act^k \) such that
   - \( \alpha^{i-1}_a \in d_a(q_{i-1}) \) for each \( a \in \text{Ag} \),
   - \( \alpha^{i-1}_a = s_A|_a(q_0 q_1 \ldots q_{i-1}) \) for each \( a \in A \), and
   - \( o(q_{i-1}, \alpha^{i-1}_1, \ldots, \alpha^{i-1}_k) = q_i \).

For an \( Ir \)-strategy replace “\( s_A|_a(q_0 q_1 \ldots q_{i-1}) \)” by “\( s_A|_a(q_{i-1}) \)”.
Definition 8.7 (Perfect information semantics)

\[ M, q \models_{Ix} p \] iff \( p \) is in \( \pi(q) \);

\[ M, q \models_{Ix} \varphi \lor \psi \] iff \( M, q \models_{Ix} \varphi \) or \( M, q \models_{Ix} \psi \);

\[ M, q \models_{Ix} \langle A \rangle \Phi \] iff there is a collective \( Ix \)-strategy \( s_A \) such that, for each path \( \lambda \in \text{out}(q, s_A) \), we have \( M, \lambda \models_{Ix} \Phi \).

\[ M, \lambda \models_{Ix} \Box \varphi \] iff \( M, \lambda[1, \infty] \models_{Ix} \varphi \);

\[ M, \lambda \models_{Ix} \Diamond \varphi \] iff \( M, \lambda[i, \infty] \models_{Ix} \varphi \) for some \( i \geq 0 \);

\[ M, \lambda \models_{Ix} \square \varphi \] iff \( M, \lambda[i, \infty] \models_{Ix} \varphi \) for all \( i \geq 0 \);

\[ M, \lambda \models_{Ix} \varphi U \psi \] iff \( M, \lambda[i, \infty] \models_{Ix} \psi \) for some \( i \geq 0 \), and \( M, \lambda[j, \infty] \models_{Ix} \varphi \) for all \( 0 \leq j \leq i \).

Note that temporal formulae and Boolean connectives are handled as before.
Example: Robots and Carriage

\[ \text{pos}_0 \rightarrow \langle 1 \rangle [\square \neg \text{pos}_1] \]
Definition 8.8 (ATL\(_{Ix}\), ATL\(_{Ix}^+\), ATL\(_{Ix}^*\), ATL, ATL\(^*\))

We define ATL\(_{Ix}\), ATL\(_{Ix}^+\), and ATL\(_{Ix}^*\) as the logics \((\mathcal{L}_{ATL}, \models_Ix)\), \((\mathcal{L}_{ATL^+}, \models_Ix)\) and \((\mathcal{L}_{ATL^*}, \models_Ix)\) where \(x \in \{r, R\}\), respectively. Moreover, we use ATL (resp. ATL\(^*\)) as an abbreviation for ATL\(_{IR}\) (resp. ATL\(_{IR}^*\)).

As usual, a logic is given by the set of all valid formulae.
8.2 Perfect vs. imperfect recall
Theorem 8.9

For $\mathcal{L}_{\text{ATL}}$, the perfect recall semantics is equivalent to the memoryless semantics under perfect information, i.e., $M, q \models_{\text{IR}} \varphi$ iff $M, q \models_{\text{Ir}} \varphi$. Both semantics are different for $\mathcal{L}_{\text{ATL}}^\ast$. That is

$$\text{ATL} = \text{ATL}_{\text{Ir}} = \text{ATL}_{\text{IR}}.$$ 

Proof idea.

The first “non-looping part” of each path has to satisfy a formula.

Exercise

The property has been first observed in [Schobbens, 2004] but it follows from [Alur et al. 2002] in a straightforward way.
What about $\langle 1, 2 \rangle (\Diamond \text{pos}_1 \land \Diamond \text{halt})$?

$\mathcal{M}, q_0 \models IR \langle 1, 2 \rangle (\Diamond \text{pos}_1 \land \Diamond \text{halt})$

$\mathcal{M}, q_0 \not\models IR \langle 1, 2 \rangle (\Diamond \text{pos}_1 \land \Diamond \text{halt})$
Example 8.10 \((\text{ATL}^*_{IR} \neq \text{ATL}^*_{lr})\)

\[
\varphi = \langle a \rangle (\lozenge p \land \lozenge \lozenge \lozenge \neg p)
\]
An interesting point is the comparison between memory and no memory.

Can agents really achieve more (in terms of validities) if they have memory available?

Suppose we want to show that $\text{ATL}_{Ir}^* \subseteq \text{ATL}_{IR}^*$; i.e., more properties of games are valid if perfect recall strategies are considered.

For this purpose, we show that every $\text{IR}$-satisfiable formula is also $\text{Ir}$-satisfiable.

Then, the claim follows: Suppose $\varphi \in \text{ATL}_{Ir}$ and $\varphi \notin \text{ATL}_{IR}$. By the latter, $\neg \varphi$ is $\text{IR}$-satisfiable hence also $\text{Ir}$-satisfiable.

Contradiction!

How can we show that $\text{IR}$-satisfiability implies $\text{Ir}$-satisfiability?
Suppose \((M, q)\) \(IR\)-satisfies \(\varphi\). Then, we show that there is a pointed model \((M', q)\) which satisfies the same formulae and in which memoryless and perfect-recall strategies coincide.

Which properties must \(M'\) have so that both kind of strategies have the same expressive power?

**Definition 8.11 (Tree-like CGS)**

Let \(M\) be a CGS. \(M\) is called tree-like iff, by definition, there is a state \(q_0\) (the root) such that for every \(q\) there is a unique history leading from \(q_0\) to \(q\).

**Proposition 8.12 (Recall invariance for tree-like CGS)**

For every tree-like CGS \(M\), state \(q\) in \(M\), and \(ATL^*\)-formula \(\varphi\), we have: \(M, q \models IR \varphi\) iff \(M, q \models IR \varphi\).

Can we always obtain such a tree-like “version” of a model?
For each model, we can construct an equivalent tree-like model: Fix a state and unfold the model to an infinite tree.

Note: states correspond to finite histories.
**Definition 8.13 (Perfect information tree unfolding)**

Let $\mathcal{M} = (\text{Agt}, Q, \Pi, \pi, \text{Act}, d, o)$ be a CGS and $q$ be a state in it.

The (perfect information) tree unfolding of the pointed model $(\mathcal{M}, q)$ denoted $T(\mathcal{M}, q)$ is defined as $(\text{Agt}, Q', \text{Prop}, \pi', \text{Act}, d', o')$ where

- $Q' := \Lambda_{\mathcal{M}}^{\text{fin}}(q)$,
- $d'(a, h) := d(a, \text{last}(h))$,
- $o'(h, \vec{\alpha}) := h \circ o(\text{last}(h), \vec{\alpha})$, and
- $\pi'(h) := \pi(\text{last}(h))$.

The node $q$ in the unfolding is called root of $T(\mathcal{M}, q)$.

**Theorem 8.14**

For every CGS $\mathcal{M}$, state $q$ in $\mathcal{M}$, and ATL*-formula $\varphi$ we have:

$$\mathcal{M}, q \models_{IR} \varphi \iff T(\mathcal{M}, q), q \models_{IR} \varphi \iff T(\mathcal{M}, q), q \models_{Ir} \varphi.$$
We now compare **perfect vs. imperfect memory**.

**Proposition 8.15**

\[ \text{ATL}_{Ir}^* \subsetneq \text{ATL}_{IR}^* \]  
*(In fact: \( \text{ATL}_{Ir}^+ \subsetneq \text{ATL}_{IR}^+ \))*

**Membership:** If \( \models_{Ir} \varphi \) then \( \text{Treemodels} \models_{Ir} \varphi \) then \( \text{Treemodels} \models_{IR} \varphi \) then \( \models_{IR} \varphi \)

**Strict inclusion:**

\[ M, q_0 \not\models_{Ir} \langle a \rangle (\lozenge p_1 \land \lozenge p_2) \iff \langle a \rangle (\lozenge (p_1 \land \langle a \rangle \lozenge p_2) \lor (p_2 \land \langle a \rangle \lozenge p_1)). \]

\( p_1 = \text{clean} \)

\( p_2 = \text{delivered} \)
8.3 Imperfect Information
Imperfect information

How can we reason about agents/extensive games with imperfect information?

We combine ATL* and epistemic logic.

- We extend CGSs with indistinguishability relations $\sim_a \subseteq Q \times Q$, one per agent. The relations are assumed to be equivalence relations.

- We interpret $\langle A \rangle$ epistemically ($\models_i R$ and $\models_i r$)
Definition 8.16 (CEGS)

A concurrent epistemic game structure (CEGS) is a tuple

\[ \mathcal{M} = (\text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \{\sim_a \mid a \in \text{Agt}\}) \]

with

- \((\text{Agt}, Q, \Pi, \pi, \text{Act}, d, o)\) a CGS and
- \(\sim_a \subseteq Q \times Q\) equivalence relations (indistinguishability relations).
Example: Robots and Carriage

What about $\langle\text{Agt}\rangle \bigcirc pos_1$ in $q_0$?

$\mathcal{M}, q_0 \models lr\langle\text{Agt}\rangle \bigcirc pos_1$

$\mathcal{M}, q_0 \not\models lr\langle\text{Agt}\rangle \bigcirc pos_1$
The last example shows that although there is a strategy, the agents do not know it (because of imperfect information).

**Problem:** Strategic and epistemic abilities are not independent!

\[ \langle A \rangle \Phi = A \text{ can bring about } \Phi \]

It should at least mean that \( A \) are able to **identify** and **execute** the right strategy!

Executable strategies = *uniform strategies*
Definition 8.17 (Uniform strategy)

Strategy \( s_a \) is **uniform** iff it specifies the **same choices for indistinguishable situations**:

- **Memoryless strategies:**
  \[
  \text{if } q \sim_a q' \text{ then } s_a(q) = s_a(q').
  \]

- **Perfect recall:**
  \[
  \text{if } \lambda \approx_a \lambda' \text{ then } \Rightarrow s_a(\lambda) = s_a(\lambda'),
  \]
  where \( \lambda \approx_a \lambda' \text{ iff } \lambda[i] \sim_a \lambda'[i] \text{ for every } i. \)

A **collective strategy** is uniform iff it consists only of uniform individual strategies.
Imperfect Information Strategies

Definition 8.18 (IR- and Ir-strategies)

- A imperfect information perfect recall strategy for agent $a$ (iR-strategy for short) is a uniform IR-strategy.
- A imperfect information memoryless strategy for agent $a$ (ir-strategy for short) is a uniform Ir-strategy.

The outcome is defined as before.
Imperfect Information Semantics

The *imperfect information semantics* is defined as before, only the clause for

\[ M, q \models_{ix} \langle A \rangle \varphi \text{ iff there is a collective } \text{Ix-strategy} \ s_A \text{ such that, for each path } \lambda \in \text{out}(q, s_A), \text{ we have } M, \lambda \models_{ix} \varphi. \]

is replaced by \((x \in \{r, R\} \text{ and } \sim_A := \bigcup_{a \in A} \sim_a.\)

\[ M, q \models_{ix} \langle A \rangle \varphi \text{ iff there is a uniform } \text{ix-strategy} \ s_A \text{ such that, for each path } \lambda \in \bigcup_{q' : q \sim_A q'} \text{out}(q', s_A), \text{ we have } M, \lambda \models_{ix} \varphi \]
Remark 8.19

This definition models that “everybody in A knows that $\varphi$”.

The fixed-point characterization does not hold anymore!

Theorem 8.20

The following formulae are not valid for $\text{ATL}_{ir}$:

- $\langle\langle A \rangle\rangle \Box \varphi \leftrightarrow \varphi \land \langle\langle A \rangle\rangle \bigcirc \langle\langle A \rangle\rangle \Box \varphi$
- $\langle\langle A \rangle\rangle \varphi_1 U \varphi_2 \leftrightarrow \varphi_2 \lor (\varphi_1 \land \langle\langle A \rangle\rangle \bigcirc \langle\langle A \rangle\rangle \varphi_1 U \varphi_2)$.

Proof: $\leadsto$ Exercise.
Proof idea

We construct a counterexample for

\[ \langle 1 \rangle \diamond p \leftrightarrow p \lor \langle 1 \rangle \circ \langle 1 \rangle \diamond p \]

\[ \mathcal{M}, q_1 \not\models_{ir} \langle 1 \rangle \diamond p \text{ iff } \]
\[ \neg (\exists s \in \Sigma_u \forall \lambda \in \bigcup_{q \in \{q_1, q_2\}} \text{out}(q, s) \exists i \in \mathbb{N}_0 : \mathcal{M}, \lambda[i] \models_{ir} p) \]

\[ \mathcal{M}, q_1 \models_{ir} p \lor \langle 1 \rangle \circ \langle 1 \rangle \diamond p \]
Comparing $\text{ATL}_{ir}$ vs. $\text{ATL}_{Ir}$

Incomplete vs. perfect information.

**Proposition 8.21**

$\text{ATL}_{ir} \subsetneq \text{ATL}_{Ir}$

**Inclusion:** Every CGS can be seen as a special CEGS

$\mathcal{M}, q_0 \not\models_{ir} (\text{shot} \lor \langle a \rangle \circ \langle a \rangle \lozenge \text{shot}) \rightarrow \langle a \rangle \lozenge \text{shot}$
Proposition 8.22

\[ \text{ATL}_{iR} \subset \text{ATL}_{IR} \]

\[ M, q_4 \not\models_i R \langle a \rangle \lozenge \text{shot} \rightarrow (\text{shot} \lor \langle a \rangle \bigcirc \langle a \rangle \lozenge \text{shot}) \]
The tree unfolding for the $i$-semantics is more sophisticated. Consider the following model and the formula $\langle a \rangle \circ \langle a \rangle \circ \langle a \rangle \circ \text{shot}$. How can an $iR$-tree unfolding look like?
A first approach is to connect separate unfoldings of the indistinguishable states by epistemic links.

Figure 29: Two separate unfoldings connected by an epistemic link. We use number $i_1i_2 \ldots$ to refer to the history $q_{i_1}q_{i_2} \ldots$.

What about the formula $\langle \langle a \rangle \rangle \bigcirc \langle \langle a \rangle \rangle \bigcirc \langle \langle a \rangle \rangle \bigcirc$ shot? The $iR$-tree unfoldings is shown on the next slide.
8 Strategic Logics

8.3 Imperfect Information

To \((M_1, q_0)\) \(\sim_a\) \(\vdash a\) \(\sim_a\) To \((M_1, q_1)\)

\(\vdash a\)
Now we can state our main result for $iR$-tree unfoldings.

**Theorem 8.23**

For every CEGS $M$, state $q$ in $M$, and ATL$^*$-formula $\varphi$, it holds that

$$M, q \models _{iR} \varphi \text{ iff } Ts(M, q), q \models _{iR} \varphi \text{ iff } Ts(M, q), q \models _{ir} \varphi.$$ 

**Summary**

If a formula is $IR$-, or $iR$-satisfiable then it also is $Ir$-, or $ir$-satisfiable, respectively.
Remark 8.24 (Important Validities and Invalidities)

1. \( \langle a \rangle \diamond p \leftrightarrow p \lor \langle a \rangle \circlearrowleft \langle a \rangle \diamond p \)
   - Invalid in all variants with imperfect information.
   - Valid for perfect information.

2. \( \langle a \rangle (\diamond p_1 \land \diamond p_2) \leftrightarrow \langle a \rangle \diamond ((p_1 \land \langle a \rangle \diamond p_2) \lor (p_2 \land \langle a \rangle \diamond p_1)) \)
   - Invalid for imperfect information
   - Valid for perfect information and perfect recall

3. \( \neg \langle \emptyset \rangle \neg p \leftrightarrow \langle \text{Agt} \rangle \square p \)
   - Invalid for imperfect information
   - Valid for perfect information.
Overview of the Results

- “All” semantic variants are different on the level of general properties; before our study, it was by no means obvious.

- Strong pattern of subsumption (memory and information)

- Very natural (not obvious before).

- Non-validities are interesting.

\[ \text{ATL}_{IR} \uparrow \]

\[ \text{ATL}_{iR} \]

\[ \text{ATL}_{isR} \]

\[ \text{ATL}_{ioR} \]

\[ \text{incomparable} \]

\[ \text{ATL}_{IR} = \text{ATL}_{iR} \]
8.4 Defining Equilibria in ATLP
Aim

We would like to

... reason about rational behavior of agents.

... have a logic that can use & describe solution concepts.

... compare different game theoretic solution concepts.

For this section we refer to [Bulling et al., 2008] for further details.
Plausibility concept

**ATL**: Reasoning about *all* possible behaviors.

\[ \lll A \rrr \varphi : \text{Agents } A \text{ have a collective strategy to enforce } \varphi \text{ against any response of their opponents.} \]

**ATLP**: Reasoning about *plausible* behaviors.

\[ \lpl \lll A \rrr \varphi : \text{Agents } A \text{ have a plausible collective strategy to enforce } \varphi \text{ against any plausible response of their opponents.} \]

Playing *undominated* strategies is plausible,...
The Base Logic: $\mathcal{L}_\text{ATLP}^{\text{base}}$

**Definition 8.25 ($\mathcal{L}_\text{ATLP}^{\text{base}}$)**

The language $\mathcal{L}_\text{ATLP}^{\text{base}}$ is defined over nonempty sets:

- $\text{Prop}$ of propositions, $p \in \text{Prop}$,
- $\text{Agt}$ of agents, $a \in \text{Agt}$, $A \subseteq \text{Agt}$, and
- $\Omega$ of basic plausibility terms, $\omega \in \Omega$.

$$\varphi ::= p \mid \neg \varphi \mid \varphi \land \varphi \mid \langle\langle A \rangle\rangle \Box \varphi \mid \langle\langle A \rangle\rangle \varphi U \varphi \mid \Pi_A \varphi \mid (\text{set-pl} \ \omega) \varphi$$
Semantics

$\text{Pl}_B : \text{Assuming plausible play of } B$

$M, q \models \text{Pl}_B \langle A \rangle \gamma$

is true iff

1. $A$ can enforce $\gamma$ if
2. agents in $B$ play only plausible strategies

Which strategies are plausible?
Plausibility Terms

Ω: Set of basic plausibility terms, \( \omega \in \Omega \)

**Hard-wired** sets of strategies:

- \( \omega_{NE} \): Nash equilibria
- \( \omega_{PO} \): Pareto optimal strategies

How to activate them?

\[(\text{set-pl } \omega) : \text{Sets plausible strategies to } [\omega] \subseteq \Sigma\]

And where do the terms come from?
Concurrent game structures with plausibility

\[ \mathcal{M} = (\text{Ag}, Q, \text{Prop}, \pi, \text{Act}, d, \delta, \mathcal{Y}, \Omega, [\cdot]) \]

- \( \mathcal{Y} \subseteq \Sigma \): set of (plausible) strategy profiles

  Example: \( \mathcal{Y} = \{ (\text{head}, \text{head}) \} \)

- \( \Omega = \{ \omega_1, \omega_2, \ldots \} \): set of plausibility terms

  Example: \( \omega_{NE} \) stands for all Nash equilibria

- \( [\cdot] : Q \rightarrow (\Omega \rightarrow \mathcal{P}(\Sigma)) \): plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: \( [\omega_{NE}]_q = \{ (\text{head}, \text{head}), (\text{tail}, \text{tail}) \} \)
Semantics of $\mathcal{L}_{ATLP}$

Let $P \subseteq \Sigma$ be a set of strategy profiles.

$\Sigma_A(P)$: strategy profiles of $A$ that are consistent with $P$.

Restricting $A$’s strategies wrt $P$

$\Sigma_A(P) := \{ s_A \in \Sigma_A \mid \exists t \in P \ (t[A] = s_A) \}$

$P(s_A)$: plausible strategy profiles of $A$ that agree on $s_A$.

Restricting $A$’s opponents strategies wrt $P$

$P(s_A) := \{ t \in P \mid t[A] = s_A \}$

t$[A]$: restriction of $t \in \Sigma$ to the strategy profile of $A$. 
Outcome of a strategy

Outcome = **Paths** that may occur when agents $A$ perform $s_A$

$$\text{out}(q, s_A, P) =$$

$$\{q_0q_1 \ldots \in Q^+ \mid q = q_0 \land \exists t \in P(s_A) \forall i \in \mathbb{N} (q_{i+1} = \delta(q_i, t(q_i)))\}$$

The outcome is given wrt to a set of (plausible) strategy profiles $P$, **restricting the opponents choices**!
8 Strategic Logics

8.4 Defining Equilibria in ATLP

\[ \text{out}(q_0, \text{head}_1, \Sigma) = \{q_0q_2q_0q_2 \ldots, q_0q_1q_0q_1 \ldots \} \]
8 Strategic Logics

8.4 Defining Equilibria in ATLP

Semantics of $\mathcal{L}_{ATLP}$

We use a satisfaction relation $\models_P$ annotated with a set of strategy profiles.

$P$: strategies currently considered available

$M, q \models_P \langle A \rangle \circ \varphi$ iff $\exists s_A \in \Sigma_A(P) \quad \forall \lambda \in \text{out}(q, s_A, P)$

$M, \lambda[1] \models_P \varphi$

$M, q \models_P \text{Pl} \varphi$ iff $M, q \models_\Upsilon \varphi$

$M, q \models_P \text{Ph} \varphi$ iff $M, q \models_\Sigma \varphi$

$M, q \models_P (\text{set-pl} \omega) \varphi$ iff $M^\omega, q \models_P \varphi$ where the new model $M^\omega$ is equal to $M$ but the “new” set $\Upsilon^\omega$ of plausible strategy profiles is set to $[\omega]_q$. 

Example: A Penny Game

The diagram represents a game with five states: q0, q1, q2, q3, q4, and q5. The states are connected by transitions labeled with outcomes of coin flips (Hh, Th, Ht, Tt). The game starts at state q0 and proceeds through the states based on the outcomes of the coin flips. The states q1, q2, q3, q4, and q5 are labeled with money amounts (money1, money2). The transitions show the possible paths and outcomes of the game.
How to describe strategies?

Plausibility terms: *abstract labels, no structure*!

Idea: Formulas that describe plausible strategies!

Select all $s$ st $s$ is better than any other strategy $s'$

Complex plausibility terms $\omega$:

\[
\sigma \land \forall \sigma_1 \exists \sigma_2 \ldots \forall \sigma_n \varphi(\sigma, \sigma_1, \ldots, \sigma_n)
\]

Property that $\sigma$ should fulfill $\in \mathcal{L}_{\text{base}}^{\text{ATLP}}(\Omega \cup \{\sigma, \sigma_1, \ldots, \sigma_n\})$

Example: $\omega_{\text{DOM}} = \sigma \land \forall \sigma' (\sigma \text{ better than } \sigma')$

How to determine whether a strategy is good?
General Solution Concepts

Agents have preferences: \( \tilde{\eta} = \langle \eta_1, \ldots, \eta_k \rangle \)
\( \eta_i \): ATL path formulas

Example: \( \eta_2 = \Diamond \text{money}_2 \)

No payoffs needed as for classical solution concepts!
Each CGSP with preferences corresponds to a normal form game.

\[ \begin{array}{c|cccc} \eta_1 \setminus \eta_2 & S_{hh} & S_{ht} & S_{th} & S_{tt} \\ \hline S_{hh} & 1, 1 & 0, 0 & 0, 1 & 0, 1 \\ S_{ht} & 0, 0 & 0, 1 & 0, 1 & 0, 1 \\ S_{th} & 0, 1 & 0, 1 & 1, 1 & 0, 0 \\ S_{tt} & 0, 1 & 0, 1 & 0, 0 & 0, 1 \end{array} \]
Characterizing Solution Concepts

\[ BR^{\eta}_{\bar{a}}(\sigma) \equiv (\text{set-pl } \sigma[\text{Agt}\backslash\{a\}]) \text{Pl} (\langle a \rangle \eta_a \rightarrow (\text{set-pl } \sigma)\langle \emptyset \rangle \eta_a) \]

\[ NE^{\eta}(\sigma) \equiv \bigwedge_{a \in \text{Agt}} BR^{\eta}_{\bar{a}}(\sigma) \]

\[ SPN^{\eta}(\sigma) \equiv \langle \emptyset \rangle \square NE^{\eta}(\sigma) \]

\[ PO^{\eta}(\sigma) \equiv \forall \sigma' \text{ Pl} \left( \bigwedge_{a \in \text{Agt}} ((\text{set-pl } \sigma')\langle \emptyset \rangle \eta_a \rightarrow (\text{set-pl } \sigma)\langle \emptyset \rangle \eta_a) \lor \right. \]

\[ \left. \bigvee_{a \in \text{Agt}} ((\text{set-pl } \sigma)\langle \emptyset \rangle \eta_a \land \neg (\text{set-pl } \sigma')\langle \emptyset \rangle \eta_a) \right). \]
Characterizing Solution Concepts (2)

Theorem

All these characterizations correspond to their game-theoretical counterparts.

Example

All plays:

\[ M, q_0 \models \neg \langle a_2 \rangle \bigcirc \text{money}_2 \]

Both agents play a Nash equilibrium strategy:

\[ M, q_0 \models (\text{set-pl } \sigma.\text{NE}^\eta(\sigma))\text{Pl} \langle a_2 \rangle \bigcirc \text{money}_2 \]
8 Strategic Logics

8.4 Defining Equilibria in ATLP

**ATLP with ATLI based plausibility specifications**

**Remark**

We can also define **quantitative** temporalized versions: \( BR_T^a, NE_T, SPN_T \), where \( T = \Box, \square, \Diamond \) and states are labeled with propositions which represent payoffs.

We then have the following theorem:

**Theorem 8.26**

Let \( \Gamma \) be an extensive game (with a finite set of proposition), and \( \mathcal{M}(\Gamma) \) a CGS corresponding to \( \Gamma \). Then

\[
\mathcal{M}(\Gamma), \emptyset \models NE_{\Diamond}(\sigma) \ (\text{resp. } SPN_{\Diamond}(\sigma)) \iff \sigma \text{ is NE (resp. a subgame-perfect NE) in } \Gamma.
\]
The Full Language: $L_{ATLP}$

Plausibility terms:

$$\sigma. \forall \sigma_1 \exists \sigma_2 \ldots \forall \sigma_n \varphi$$

where

$$\varphi \in L_{ATLP}^{base}$$

What about nesting (set-pl $\cdot$) operators?

$$\text{(set-pl \ldots (set-pl \ldots (set-pl \ldots) \ldots) \ldots)}$$

We get a hierarchy of logics:

$$L_{ATLP}^k : k \text{ nestings}$$

$$L_{ATLP} := \lim_{k \to \infty} L_{ATLP}^k$$
8.5 References

