Game Theory

Prof. Dr. Jürgen Dix

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.
Check also StudIP, which contains updated slides and exercise sheets.

Lecture: Prof. Dix, N. Fiekas
Exam: oral examination

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).

About this lecture (2)

- 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)

Bordini, R. and Dix, J. (2013).
Chapter 13: Programming Multi Agent Systems.
In G. Weiß (Editor), *MultiAgent Systems*, pages 587–639.
MIT Press.

Bulling, N., Jamroga, W. and Dix, J. (2008).
Reasoning about Temporal Properties of Rational Play,
Annals of Mathematics and Artificial Intelligence, **53(1–4)***51–114.

Bulling, N., Dix, J., and Jamroga, W. (2010).
Model checking logics of strategic ability: Complexity.
In Dastani, M., Hindriks, K. V., and Meyer, J.-J. C., editors, *Specification and Verification of Multi-Agent Systems*, pages 125–158.
Springer.

Dix, J. and Fisher, M. (2013).
Chapter 14: Verifying Multi Agent Systems.
In G. Weiß (Editor), *MultiAgent Systems*, pages 641–693.
MIT Press.

Dix, J. Jamroga, W. and Samet, D. (2011).
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)

- Fudenberg, D. and J. Tirole (1991).
  Game Theory.
  MIT Press.
- Osborne, M.J., Rubinstein, Ariel, M. (1994).
  A Course in Game Theory.
  MIT Press.
- Shoham, Y. and Leyton-Brown, K. (2009).
  Multiagent Systems - Algorithmic, Game-Theoretic, and Logical Foundations.
  Cambridge University Press.

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

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 simultanously,
- 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 a the outcome, assuming that the given **protocol** is followed and the agents follow the given 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

Find a protocol such that if each agent choses its best local strategy, the **outcome** is a global optimum.

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

there is no solution x' with:

(1) $\exists$ agent $\mathbf{ag}$: $\mathbf{ut}_{\mathbf{ag}}(x') > \mathbf{ut}_{\mathbf{ag}}(x)$
(2) $\forall$ agents $\mathbf{ag}'$: $\mathbf{ut}_{\mathbf{ag}'}(x') \geq \mathbf{ut}_{\mathbf{ag}'}(x)$.

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 $s^*_{\mathbf{A}} = \langle s^*_1, s^*_2, \ldots, s^*_{|\mathbf{A}|}\rangle$ is called a Nash-equilibrium, iff $\forall \mathbf{i} : s^*_{\mathbf{i}}$ is the best strategy for agent $\mathbf{i}$ if all the others choose

$$\langle s^*_1, s^*_2, \ldots, s^*_{\mathbf{i}-1}, s^*_{\mathbf{i}+1}, \ldots, s^*_{|\mathbf{A}|}\rangle.$$

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.

		Prisoner 2	
		cooperate	defect
Prisoner 1	cooperate	(3,3)	(0,5)
	defect	(5,0)	(1,1)

- **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: $c \gneqq a \gneqq d \gneqq b$

		Prisoner 2	
		cooperate	defect
Prisoner 1	cooperate	(a,a)	(c,b)
	defect	(b,c)	(d,d)

Example 1.2 (Trivial mixed-motive, Type 0)

		Player 2	
		C	D
Player 1	C	(4, 4)	(2, 3)
	D	(3, 2)	(1, 1)

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**.

		Japanese	
		Sail North	Sail South
Allies	Search North	2 days	2 days
	Search South	1 day	3 days

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 $\langle \mathbf{A}, \mathrm{Act}, O, \varrho, \boldsymbol{\mu} \rangle$, where

- $\mathbf{A} = \{\mathbf{1}, \dots, \mathbf{i}, \dots, \mathbf{n}\}$ is a finite set of players or agents.
- $\mathrm{Act} = A_{\mathbf{1}} \times \cdots \times A_{\mathbf{i}} \times \cdots \times A_{\mathbf{n}}$ where $A_{\mathbf{i}}$ is the finite set of actions available to player $\mathbf{i}$. $\vec{a} \in \mathrm{Act}$ is called action profile. Elements of A_i are called pure strategies.
- O is the set of outcomes.
- $\varrho : \mathrm{Act} \longrightarrow O$ assigns each action profile an outcome.
- $\boldsymbol{\mu} = \langle \boldsymbol{\mu}_{\mathbf{1}}, \dots, \boldsymbol{\mu}_{\mathbf{i}}, \dots, \boldsymbol{\mu}_{\mathbf{n}} \rangle$ where $\boldsymbol{\mu}_{\mathbf{i}} : O \longrightarrow \mathbb{R}$ is a real-valued utility (payoff) function for player $\mathbf{i}$.

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:

		Player 2	
		a_2^1	a_2^2
Player 1	a_1^1	$(\boldsymbol{\mu}_1(a_1^1, a_2^1), \boldsymbol{\mu}_2(a_1^1, a_2^1))$	$(\boldsymbol{\mu}_1(a_1^1, a_2^1), \boldsymbol{\mu}_2(a_1^1, a_2^2))$
	a_1^2	$(\boldsymbol{\mu}_1(a_1^2, a_2^2), \boldsymbol{\mu}_2(a_1^2, a_2^2))$	$(\boldsymbol{\mu}_1(a_1^2, a_2^1), \boldsymbol{\mu}_2(a_1^2, a_2^2))$

We often forget about ϱ (thus we are making no distinction between **actions and outcomes**). Thus we simply write $\mu_1(a_1^1, a_2^1)$ instead of the more precise $\mu_1(\varrho(\langle a_1^1, a_2^1\rangle))$. 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 $\mathtt{i}, \mathtt{j}$ the following holds: $\mu_{\mathtt{i}}(\vec{a}) = \mu_{\mathtt{j}}(\vec{a})$.

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) $\mu_2(a_1^2, a_2^2)$ in each entry:

		Player 2	
		C	D
Player 1	C	4	2
	D	3	1

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

Pure vs. mixed strategies

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 $\vec{a} = \langle a_1, \dots, a_n \rangle$.

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.

		Min		
		P	S	R
Max	P	0	-1	1
	S	1	0	-1
	R	-1	1	0

What about pure vs mixed strategies?

Definition 1.9 (Mixed Strategy for NF Games)

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

Note: Some books use

- S_i, with elements s_i, to denote the set of **pure** strategies,
- Σ_i with elements σ 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 $\mu^{expected}$.

Definition 1.10 (Expected Utility for player i)

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

$$\mu^{expected}(s_1, \ldots, s_n) = \sum_{\vec{a} \in \mathrm{Act}} \mu_{\mathbf{i}}(\varrho(\vec{a})) \prod_{j=1}^{n} s_j(a_j).$$

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**.

		Bomber Crew	
		Look Up	Look Down
Fighter Pilots	Hun-in-the-Sun	**0.95**	**1**
	Ezak-Imak	**1**	**0**

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).

		Wife	
		Theatre	Restaurant
Husband	Theatre	(4,3)	(2,2)
	Restaurant	(1,1)	(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.

		Driver 2	
		C	D
Driver 1	C	(2,2)	(3,4)
	D	(4,3)	(1,1)

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.

		Player 2	
		Head	Tails
Player 1	Head	(1, -1)	(-1, 1)
	Tails	(-1, 1)	(1, -1)

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:

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

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

The minmax strategy for player i is

$$\arg\min_{s_i} \max_{s_{-i}} \mu_{-i}^{expected}(s_i, s_{-i})$$

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

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:

		Player B		
		B-I	B-II	B-III
Player A	A-I	0	$\frac{5}{6}$	$\frac{1}{2}$
	A-II	1	$\frac{1}{2}$	$\frac{3}{4}$

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

$$\text{arg max}_{s_i} \min_{s_{-i}} \boldsymbol{\mu}_i^{expected}(s_1, s_2)$$

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 $\boldsymbol{\mu}_1^{expected}(s_1, s_2)$

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

thus

$$\boldsymbol{\mu}_1^{expected}(s_1, s_2) = y((-\frac{4}{3}x) + \frac{1}{2}) \;+\; \frac{1}{3}x \;+\; \frac{1}{2}$$

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 μ).

What is the optimal strategy (maximising the expected payoff) for an agent in an n-agent setting?

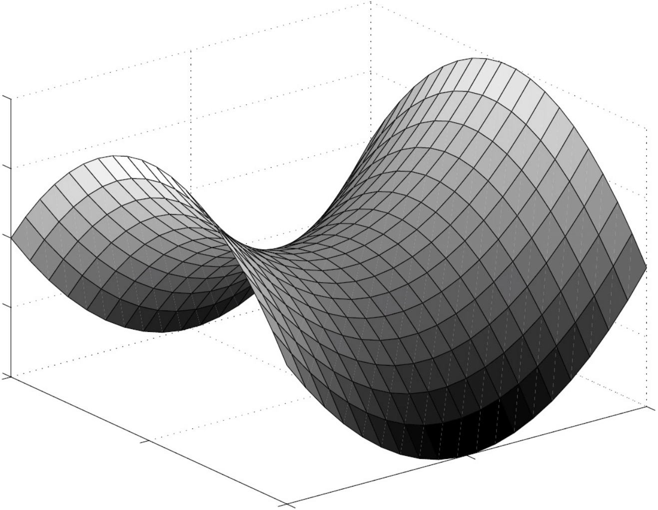


Figure 1: A saddle.

Definition 1.18 (Notation s_{-i}, S_{-i})

Note that from now on, for $\vec{s} = \langle s_1, s_2, \ldots, s_n \rangle$ we use the notation $s_{-i}^{\rightarrow}$ 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 $S_{-i} = \{s_{-i}^{\rightarrow} \mid \vec{s} \in S\}$.

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

$$\langle s_1, s_2, \ldots, \boldsymbol{s_i}, \ldots, s_n \rangle \;=\; \langle \boldsymbol{s_i}, s_{-i}^{\rightarrow} \rangle.$$

In the last vector, although the $\boldsymbol{s_i}$ 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

$$s_{\vec{-i}} = \langle s_1, s_2, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n \rangle,$$

a best response of player i to $s_{\vec{-i}}$ is any mixed strategy $s_i^* \in S_i$ such that

$$\mu_i(s_i^*, s_{\vec{-i}}) \geqq \mu_i(s_i, s_{\vec{-i}})$$

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 $.5$ and *scissors* with probability $.5$.
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.

		USSR	
		Withdrawal	Maintain
U. S.	Blockade	(3, 3) **Compromise**	(2, 4) **USSR victory**
	Air strike	(4, 2) **U.S. victory**	(1, 1) **Nuclear War**

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 for player 1, s_2 for player 2).*
All Nash equilibria have the same payoff*: the value of the game, that player 1 gets.*

Proof

We use Kakutani's theorem: Let X be a nonempty subset of n-dimensional Euclidean space, and $f : X \longrightarrow 2^X$. 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: $f(\langle s_1, \ldots, s_n \rangle)$ 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 Fixed Point Theorem)

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

Proof

Reduction to the C^1-differentiable case: Let $r : \mathbb{R}^n \to \mathbb{R}$ 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 $\phi : \mathbb{R}^n \to \mathbb{R}$ given by:

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

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

Proof (cont.)

Let $f : B^n \to B^n$ be continuous. Let $F : \mathbb{R}^n \to B^n$ be the extension of f to $\mathbb{R}^n$, for which we have

$$F(x) = f(\frac{x}{||x||}) \text{ on } \mathbb{R}^n \setminus B^n.$$

For $k \in \mathbf{N}, k \neq 0$, put $f_k(x) := \int_{\mathbb{R}^n} k^n \phi(ky)F(x-y)dy$. Show that the restriction of f_k to B^n maps B^n into B^n. 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 $f : B^n \to B^n$ without fixed points, then there will also exist a continuously differential mapping without fixed points. It follows, that it suffices to prove the Brouwer Fixed Point Theorem only for continuously differentiable mappings.

Proof (cont.)

Proof for C^1-differentiable mappings:

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

$$\begin{array}{rll} 0 & \neq \int_{B^n} dx_1 \wedge dx_2 \wedge \dots \wedge dx_n & = \int_{\partial B^n} x_1 \wedge dx_2 \wedge \dots \wedge dx_n \\ & = \int_{\partial B^n} g_1 \wedge dg_2 \wedge \dots \wedge dg_n & = \int_{B^n} dg_1 \wedge dg_2 \wedge \dots \wedge dg_n \\ & = \int_{B^n} 0 = 0 & \end{array}$$

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:
$\boldsymbol{\mu}_1(A) = \boldsymbol{\mu}_2(B) = \boldsymbol{\mu}_3(C) = 2$, $\boldsymbol{\mu}_1(B) = \boldsymbol{\mu}_2(C) = \boldsymbol{\mu}_3(A) = 1$ and $\boldsymbol{\mu}_1(C) = \boldsymbol{\mu}_2(A) = \boldsymbol{\mu}_3(B) = 0$. What are the Nash equilibria and what are their outcomes?

Example 1.27 (Unique Equilibrium)

The following game has exactly one Nash equilibrium.

	L	C	R
U	$\langle 1,-2\rangle$	$\langle -2,1\rangle$	$\langle 0,0\rangle$
M	$\langle -2,1\rangle$	$\langle 1,-2\rangle$	$\langle 0,0\rangle$
D	$\langle 0,0\rangle$	$\langle 0,0\rangle$	$\langle 1,1\rangle$

Minmax value and feasible payoffs

Definition 1.28 (Minmax value and feasible payoffs)

In a n-person normal form game $\langle \mathbf{A}, \mathrm{Act}, O, \varrho, \boldsymbol{\mu} \rangle$ we define the minmax value of player $\mathtt{i}$ as follows

$$\underline{v}_{\mathtt{i}} = \min_{s_{-\mathtt{i}}} \max_{s_{\mathtt{i}}} \boldsymbol{\mu}_{\mathtt{i}}^{expected}(s_{\mathtt{i}}, \vec{s_{-\mathtt{i}}},).$$

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

$$r_{\mathtt{i}} = \sum_{\vec{a} \in \mathrm{Act}} \alpha_{\vec{a}} \boldsymbol{\mu}_{\mathtt{i}}(\vec{a}) \text{ where } \sum_{\vec{a} \in \mathrm{Act}} \alpha_{\vec{a}} = 1.$$

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.

Definition 1.29 (Dominating Strategy: Weakly, Strictly)

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 $\vec{a_{-i}} = \langle a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n \rangle$, we have

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

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 $\gneqq$ and the inequality is strict for **at least one** of the other $\vec{a_{-i}}$.

Definition 1.30 (Reduced Sets A_i^∞, S_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 ($A_i^0 := A_i$, $S_i^0 := S_i$)

$$A_i^n := \{a_i \in A_i^{n-1} \mid \text{there is no } s_i' \in S_i^{n-1} \text{ s.t. } \mu_i(s_i', \vec{a}_{-i}) \gneqq \mu_i(a_i, \vec{a}_{-i}) \text{ for all } \vec{a}_{-i} \in A_{-i}^{n-1}\}$$

$$S_i^n := \{s' \in S_i \mid s'(a_i) \gneqq 0 \text{ only if } a_i \in A_i^n\}$$

Finally, $A_i^\infty := \bigcap_{n=0}^{\infty} A_i^n$ and S_i^∞ is the set of all **mixed strategies** s_i, such that there is no s_i' with $\mu_i(s_i', a_{-i}) \gneqq \mu_i(s_i, \vec{a}_{-i})$ for all $\vec{a}_{-i} \in A_{-i}^\infty$.

Some Comments to A_i^∞ and S_i^∞

- 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^∞ are those that we have to keep.
- Note that S_i^∞ is defined wrt. A_i^∞, not wrt. S_i^n.
- S_i^∞ is the set of mixed strategies that are **not strictly dominated** by pure action profiles from A_i^∞.
- Note that S_i^∞ can be strictly smaller than the set of all mixed strategies over A_i^∞.

Theorem 1.31 (Solvable by Iterated Strict Dominance)

If for a finite normal form game, the sets A_i^∞ 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.

	L	C	R
U	$\langle 3, 2\rangle$	$\langle 2, 1\rangle$	$\langle 3, 1\rangle$
M	$\langle 1, 1\rangle$	$\langle 1, 1\rangle$	$\langle 2, 2\rangle$
D	$\langle 0, 1\rangle$	$\langle 4, 2\rangle$	$\langle 0, 1\rangle$

1. Eliminate row **M**.
2. Eliminate column **R**.

This leads to

	L	**C**
U	$\langle 3, 2 \rangle$	$\langle 2, 1 \rangle$
D	$\langle 0, 1 \rangle$	$\langle 4, 2 \rangle$

Elimination of Weakly Dominated actions

We consider the normal form game

	L	**R**
T	$\langle 1, 1\rangle$	$\langle 0, 0\rangle$
M	$\langle 1, 1\rangle$	$\langle 2, 1\rangle$
B	$\langle 0, 0\rangle$	$\langle 2, 1\rangle$

1. If we first eliminate **T**, and then **L** we get the outcome $\langle 2, 1\rangle$.
2. If we first eliminate **B**, and then **R** we get the outcome $\langle 1, 1\rangle$.

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 $\Gamma = \langle \mathbf{A}, \mathrm{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle$ where

- $\mathbf{A}$ is a set of n players, 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**,
- $\alpha : H \to 2^{\mathrm{Act}}$ 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,
- $\sigma : H \times \mathrm{Act} \to H \cup Z$ assigns to each (node,action) a successor node ($h_1 \neq h_2$ 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.

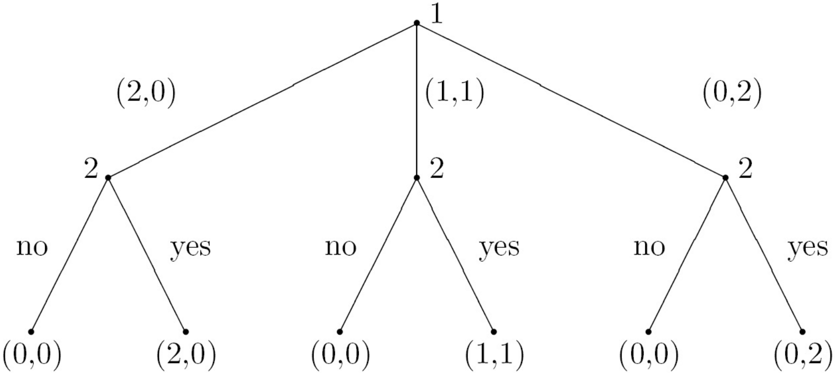


Figure 2: The Sharing game.

Strategies in extensive form games

Definition 1.35 (Strategies in Extensive Form Games)

Let $\Gamma = \langle \mathbf{A}, \mathrm{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle$ be a finite perfect information game in extensive form.

A strategy for player $\mathbf{i}$ in Γ is any function that assigns a legal move to each history owned by $\mathbf{i}$.

The pure strategies of player $\mathbf{i}$ are the elements of $\Pi_{h \in H, \rho(h) = \mathbf{i}} \alpha(h)$. These are also functions: whenever player $\mathbf{i}$ 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 $\mathbf{i}$'s choices at the respective moves.

In the sharing game, a pure strategy for player 2 is $\langle no, yes, no \rangle$. A (better) one is $\langle no, yes, yes \rangle$.

Why don't we introduce mixed strategies?

Best response, Nash Equlibrium

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?

$\langle 1, \langle \mathsf{y}, \mathsf{y}, \mathsf{y} \rangle\rangle$, $\langle 1, \langle \mathsf{n}, \mathsf{n}, \mathsf{n} \rangle\rangle$, $\langle 1, \langle \mathsf{n}, \mathsf{n}, \mathsf{y} \rangle\rangle$, $\langle 1, \langle \mathsf{y}, \mathsf{n}, \mathsf{n} \rangle\rangle$, $\langle 1, \langle \mathsf{y}, \mathsf{n}, \mathsf{y} \rangle\rangle$, $\langle 1, \langle \mathsf{y}, \mathsf{y}, \mathsf{n} \rangle\rangle$ are NE's, $\langle 1, \langle \mathsf{n}, \mathsf{y}, \mathsf{y} \rangle\rangle$ is not. Also $\langle 2, \langle \mathsf{n}, \mathsf{y}, \mathsf{n} \rangle\rangle$, $\langle 2, \langle \mathsf{n}, \mathsf{y}, \mathsf{y} \rangle\rangle$ and $\langle 3, \langle \mathsf{n}, \mathsf{n}, \mathsf{y} \rangle\rangle$ are NE's.

We claim that only $\langle 1, \langle \mathsf{y}, \mathsf{y}, \mathsf{y} \rangle\rangle$ and $\langle 2, \langle \mathsf{n}, \mathsf{y}, \mathsf{y} \rangle\rangle$ make sense.

Transforming extensive form games into normal form

Lemma 1.36 (Extensive form $\hookrightarrow$ Normal form)

Each game Γ in perfect information extensive form *can be transformed to a game $\mathrm{NF}(\Gamma)$ 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 $\mathrm{NF}(\Gamma)$ by **enumerating** all strategy profiles and filling the payoff matrix with the resulting payoffs. □

Sharing Game in normal form

1\ 2	nnn	nny	nyn	nyy	ynn	yny	yyn	yyy
$(2,0)$	**(0, 0)**	**(0, 0)**	(0, 0)	(0, 0)	**(2, 0)**	**(2, 0)**	**(2, 0)**	**(2, 0)**
$(1,1)$	(0, 0)	(0, 0)	**(1, 1)**	**(1, 1)**	(0, 0)	(0, 0)	(1, 1)	(1, 1)
$(0,2)$	(0, 0)	**(0, 2)**	(0, 0)	(0, 2)	(0, 0)	(0, 2)	(0, 0)	(0, 2)

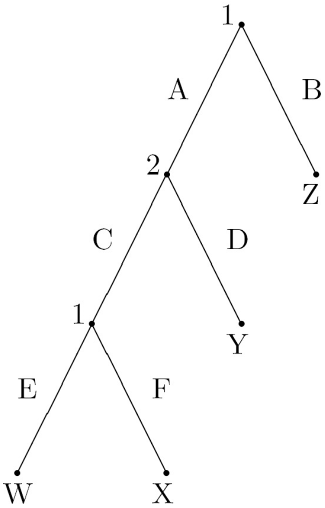


Figure 3: A generic game.

Example 1.37 (Generic Game in normal form)

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

	C	D
AE	W	Y
AF	X	Y
BE	Z	Z
BF	Z	Z

Note that $\langle B, E\rangle, \langle B, F\rangle$ 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 NE.*
The theorem can be strengthened (Kuhn's theorem): For each perfect information game in extensive form there exists *a pure strategy* **subgame perfect** *NE.*

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.**

Example 1.39 (Unintended Nash equilibria)

Consider the following game in extensive form.

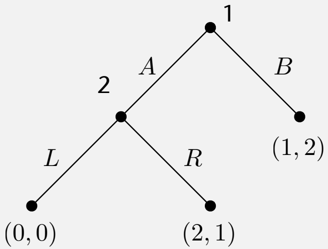


Figure 4: Unintended Equilibrium.

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 NE:

Definition 1.40 (Subgame Perfect NE (SPE))

Let Γ be a perfect information game in extensive form.

Subgame: A subgame of G rooted at node h is the restriction of Γ to the descendants of h.

SPE: The subgame perfect Nash equilibria (SPE) of a perfect information game Γ in extensive form are those Nash equilibria of Γ, that are also Nash equilibria for all subgames Γ' of Γ.

- 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 (Example 1.37):

	C	D
AE	$\langle 2, 0 \rangle$	$\langle 1, 1 \rangle$
AF	$\langle 0, 2 \rangle$	$\langle 1, 1 \rangle$
BE	$\langle 3, 3 \rangle$	$\langle 3, 3 \rangle$
BF	$\langle 3, 3 \rangle$	$\langle 3, 3 \rangle$

Theorem 1.41 (Existence of SPE (Kuhn))

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

The proof is by induction on the length of histories. The SPE is therefore defined constructively.

Proof.

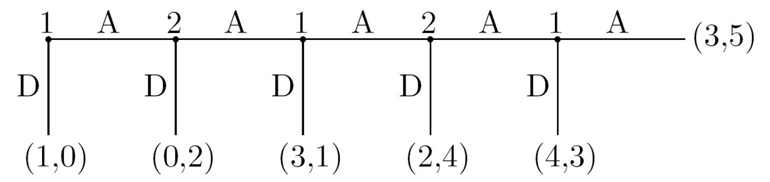
By backward induction we construct a subgame perfect NE:

- 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 $s_{\mathbf{i}}(h')$ 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 NE 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 NE.

□

Example 1.42 (Centipede Game)

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



Centipede Revisited

- 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 actions might 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 A, we define a partition I_i of agent $i \in A$ 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.

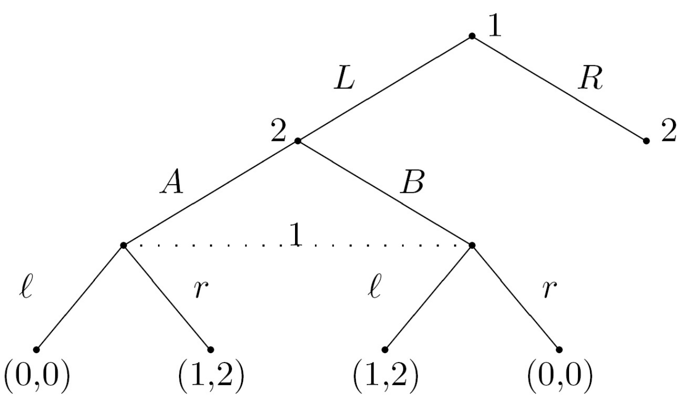
A partition I_i is a set of subsets $W_{i1}, \ldots W_{is}$ of W such that: (1) $\bigcup_i W_{ij} = W$ and (2) $W_{ij} \cap W_{ij'} = \emptyset$ for $j \neq j'$.

Definition 1.44 (Extensive Games, Imperfect Inf.)

A **finite imperfect information game in extensive form** is a tuple $G = \langle \mathbf{A}, \mathrm{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n, I_1, \ldots, I_n \rangle$ where

- $\langle \mathbf{A}, \mathrm{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \ldots, \mu_n \rangle$ is a perfect information game in the sense of Definition 1.33 on Slide 75,
- $I_{\mathbf{i}}$ are partitions on $\{h \in H : \rho(h) = \mathbf{i}\}$ such that $h, h' \in I_{\mathbf{i},\mathbf{j}}$ implies $\alpha(h) = \alpha(h')$.

Example 1.45



(Utility of the rightmost leaf node is $(2, 1)$.)

- 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 $a_{\mathbf{j}} \in \alpha(I_{\mathbf{ij}})$ where $I_{\mathbf{i}1}, \ldots, I_{\mathbf{i}k}$ are the k equivalence classes for agent **i**. Note that this vector is just 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?

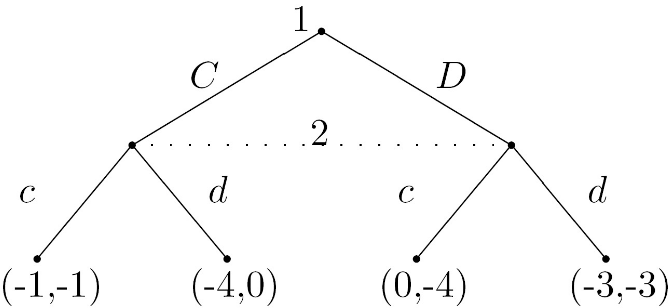


Figure 5: Prisoner's dilemma in extensive form.

There is a pure strategy Nash equilibrium.
But we could have chosen to switch player 1 with player 2.

NF game $\hookrightarrow$ 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.

- Let Γ be an imperfect information game in extensive form.
- 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 Γ.

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 $.7$, B with $.3$ and E with probability $.4$ and F with $.6$. 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 $.6$ and $\langle B, F\rangle$ with probability $.4$. 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 $G = \langle \mathbf{A}, \mathrm{Act}, H, Z, \alpha, \rho, \sigma, \mu_1, \dots, \mu_n, I_1, \dots, I_n \rangle$ be an imperfect information game in extensive form.

Mixed: A mixed strategy of player $\mathbf{i}$ is one single probability distribution over $\mathbf{i}$'s pure strategies.

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

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.

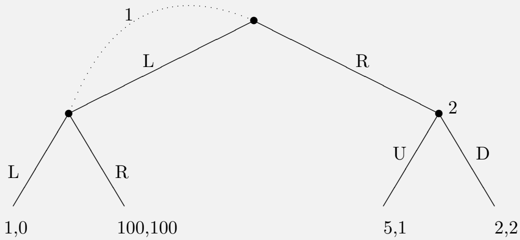
Are behavioral strategies more general?

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 $\langle \frac{1}{2}\mathbf{LL}, \frac{1}{2}\mathbf{RR}\rangle$?
- 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: $(\frac{98}{198}L, \frac{100}{198}R)$.

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)

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

$$\mu_{\mathtt{i}}(\langle s_{\mathtt{i}}^*, s_{-\mathtt{i}}^* \rangle \geq \mu_{\mathtt{i}}(\langle s_{\mathtt{i}}, s_{-\mathtt{i}}^* \rangle$$

for all mixed strategies $s_{\mathtt{i}}$ of player $\mathtt{i}$.

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 $\mathtt{i}$:

$$\mu_{\mathtt{i}}(\langle s_{\mathtt{i}}^*, s_{-\mathtt{i}}^* \rangle \geq \mu_{\mathtt{i}}(\langle s_{\mathtt{i}}, s_{-\mathtt{i}}^* \rangle$$

for all behavioral strategies $s_{\mathtt{i}}$ of player $\mathtt{i}$.

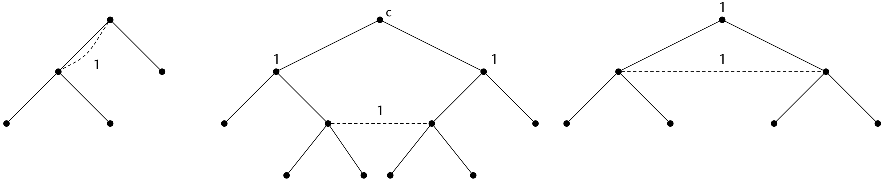
Definition 1.50 (Perfect Recall)

Let Γ be an imperfect information game in extensive form. We say that player $\mathbf{i}$ has **perfect recall** in Γ, if the following holds. If h, h' are two nodes in the same $I_{\mathbf{i}j}$ (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

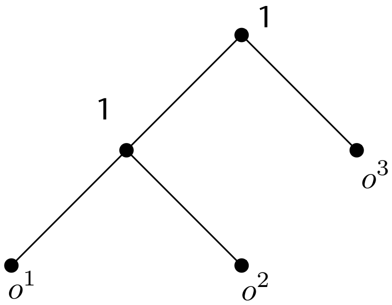
1. $n = m$,
2. for all $0 \leq j \leq n$: h_j and h'_j are in the same partition class,
3. for all $0 \leq j \leq n$: if $\alpha(h_j) = \mathbf{i}$ then $a_j = a'_j$.

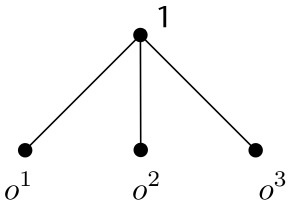
Γ is a **game of perfect recall**, if all players have perfect recall. Otherwise it is called of **imperfect recall**.

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 Γ 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.*

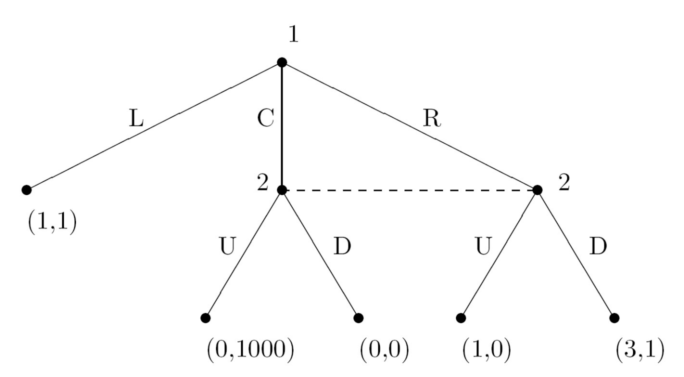
Corollary 1.52

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 90 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)



- Nash equilibria: (L, U) and (R, D). Can we see them as **subgame perfect**?
- In one subtree, 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 subtrees might be too strong.

Ways out

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** μ: functions which assign to each information set I_{ij} a probability measure over nodes in I_{ij}. $\mu(I_{ij})(h)$ is the probability that player **i** assigns to history h, provided that I_{ij} is being reached.

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

Given a behavioral strategy profile β which is **completely mixed** (i.e. assigns non-zero probability to all actions), a belief system μ can be uniquely assigned to β.

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 Γ 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 β^n is completely mixed (where μ^n is uniquely determined by β^n),
2. for any I_i of agent i and any alternative strategy β_i' of agent i: $\mu_i(\beta^* \mid h, \mu(h)) \ \geq \ \mu_i((\beta', \beta_{-i}) \mid h, \mu(h))$.

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 $\mathbf{p} = [p_1, p_2, \ldots, p_n] \in \mathbb{R}^n$: p_i is the price of the good i.

Consumers: Consumer $\mathbf{i}$ has $\mu_{\mathbf{i}}(\mathbf{x})$ encoding its preferences over consumption bundles $\mathbf{x_i} = [x_{i1}, ..., x_{in}]^t$, where $x_{ig} \in \mathbb{R}^+$ is consumer $\mathbf{i}$'s allocation of good g. Each consumer also has an initial endowment $\mathbf{e_i} = [e_{i1}, ..., e_{in}]^t \in \mathbb{R}$.

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

Profit of producer $\mathbf{j}$: $\mathbf{p} \times \mathbf{y_j}$, where $\mathbf{y_j} \in \boldsymbol{Y_j}$.

Profits: The profits are divided among the consumers (given predetermined proportions Δ_{ij}): Δ_{ij} is the fraction of producer j that consumer i owns (stocks). Profits are divided according to Δ_{ij}.

Definition 1.56 (General Equilibrium)

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

I. The markets are in equilibrium:

$$\sum_i \mathbf{x}_i^* = \sum_i \mathbf{e}_i + \sum_j \mathbf{y}_j^*$$

II. Producer $\mathbf{j}$ maximises profit wrt. the market

$$\mathbf{y}_i^* = \arg\max_{\{\mathbf{y}_j \in \mathbf{Y}_j\}} \mathbf{p}^* \times \mathbf{y}_j$$

III. Consumer i maximises preferences according to the prices

$$x_i^* = \arg\max_{\{x_i \in \mathbb{R}^n_+ \mid \boxed{\mathbf{cond}_i}\}} \mu_i(x_i)$$

where $\mathbf{cond}_i$ 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 μ_i be continuous, strictly convex and strongly monotone. Assume further that at least one bundle $\mathbf{x_i}$ 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 μ_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 $\mathbf{p}$ that are not proportional to an equilibrium vector $\mathbf{p}^*$, the following holds:

$$\sum_{i}(\mathbf{x}_i(\mathbf{p}) - \mathbf{e}_i) - \sum_{j}\mathbf{y}_j(\mathbf{p}) \gneq 0$$

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 $\mathbf{p}$ to consumers and producers.
    Receive a production plan $\mathbf{y_j}$ from each producer $\mathbf{j}$.
    Broadcast the plans $\mathbf{y_j}$ to consumers.
    Receive a consumption plan $\mathbf{x_i}$ from each consumer $\mathbf{i}$.
    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 $| \sum_i(x_{ig} - e_{ig}) - \sum_j y_{jg} | \lesssim \epsilon$ for all $1 \leq g \leq n$
Inform consumers and producers that an equilibrium has been reached.

Price tâtonnement: Consumer i

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.

Price tâtonnement: Producer $\mathbf{j}$

repeat
Receive $\mathbf{p}$ from the adjustor.
Receive a production plan for all $\mathbf{j}$ from the adjustor.
Announce to the adjustor a production plan $\mathbf{y_j} \in Y_\mathbf{j}$ that maximizes $\mathbf{p} \times \mathbf{y_j}$
until Informed by adjustor that equilibrium has been reached.
Exchange and produce.

1.5 References

Fudenberg, D. and J. Tirole (1991).
Game Theory.
MIT Press.

Nash, J. (1950).
Equilibrium points in n-person games.
Proceedings of the National Academy of Sciences of the United States of America 36, 48–49.

Osborne, M. and A. Rubinstein (1994).
A Course in Game Theory.
MIT Press.

2. Repeated Games

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 $\mu_i : Z \to \mathbb{R}$. If the agents do not agree on a result r a fixed fallback r_{fallback} is taken.

Example 2.1 (Sharing 1 Pound)

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

Agent 1 offers ρ $(0 < \rho < 1)$. 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**.

Repeated Bargaining

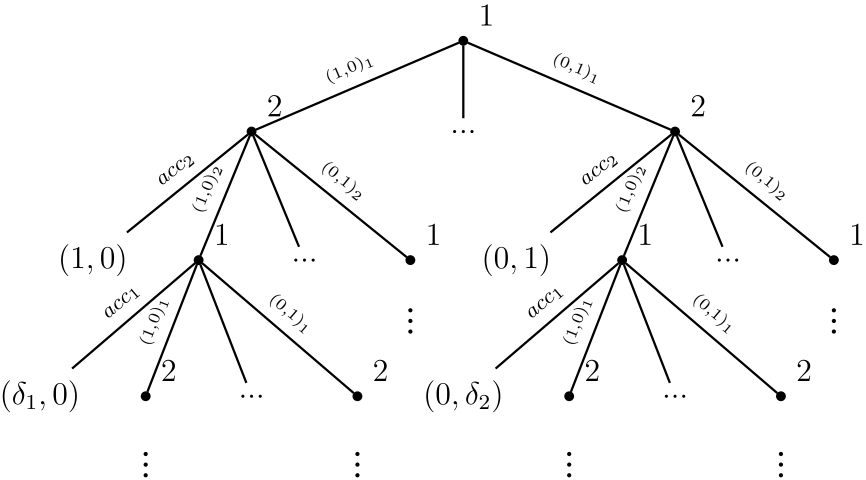


Figure 6: The bargaining game.

Lemma 2.2 (Trivial Bargaining)

Each strategy profile

$$s^x : \begin{cases} 1 \text{ offers } \langle x, 1-x\rangle, \text{ agrees to } \langle y, 1-y\rangle \text{ for } y \geq x \\ 2 \text{ offers } \langle x, 1-x\rangle, \text{ agrees to } \langle y, 1-y\rangle \text{ iff } 1-y \geq 1-x \end{cases}$$

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 ϵ. 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 *infinite* games ($\leadsto$ Subsection 2.3).
2. Keep finite games, but:
   1. *Add a discountfactor* δ: in round n, only the δ^{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 $\langle \mathbf{A}, \mathrm{Act}, O, \varrho, \boldsymbol{\mu} \rangle$, 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 $\underline{v_i}$: $\underline{v_i} = \min_{s_{-i}} \max_{s_i} \boldsymbol{\mu}_i^{expected}(s_{-i}, s_i)$ 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**.

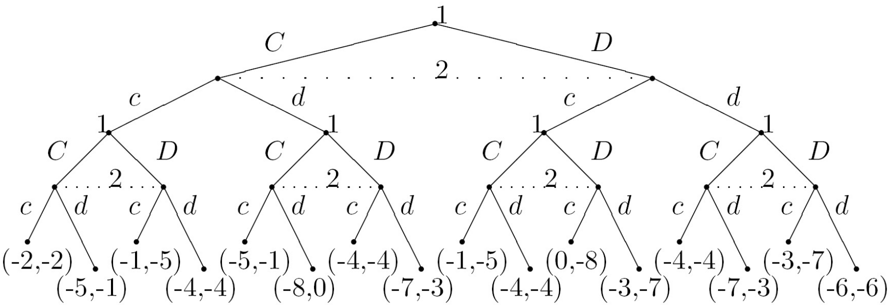


Figure 7: Iterated Prisoners Dilemma.

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.
- Robert Axelrod: **The Evolution of Cooperation**, New York: Basic Books, 1984.

Backward Induction (2)

- **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 $\mu^* = \langle \mu_1(r^*), \mu_2(r^*) \rangle$.

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 Z is made smaller but r^* still remains, then r^* remains the solution.

Pareto: The players can not get a higher utility than $\mu^* = \langle \mu_1(r^*), \mu_2(r^*) \rangle$.

Theorem 2.6 (Unique solution)

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

$$r^* = \mathit{arg\ max}_r \{(\mu_1(r) - \mu_1(r_{\mathit{fallback}})) \times (\mu_2(r) - \mu_2(r_{\mathit{fallback}}))\}.$$

Strategic bargaining for finite games

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

Round	1's share	2's share	Total value	Offerer
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$n-3$	0.819	0.181	0.9^{n-4}	2
$n-2$	0.91	0.09	0.9^{n-3}	1
$n-1$	0.9	0.1	0.9^{n-2}	2
n	1	0	0.9^{n-1}	1

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 1, c_2 from 2). Therefore the game is **finite**.

Theorem 2.7

(1) $\mathbf{c}_1 = \mathbf{c}_2$: *Any split is a SPE.*

(2) $\mathbf{c}_1 < \mathbf{c}_2$: *Only one SPE:* **Agent 1 gets all.**

(3) $\mathbf{c}_1 > \mathbf{c}_2$: *Only one SPE:* **Agent 1 gets c_2, agent 2 gets $1 - c_2$.**

Proof.

(1) is obvious.

(2) Assume agent 2 offered π 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).

$\square$

An Infinite Extensive Game

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* δ_1 (for player 1) and δ_2 (for player 2). So, after t rounds, the goods are worth only $\langle \delta_1^t, \delta_2^t \rangle$. Subsequently, 1 (if t is even) or 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 1 takes $x\delta_1^t$, and 2 gets $(1-x)\delta_2^t$; 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 $r : \mathbb{R}_0^+ \to \mathbb{R}_0^+$.
- We assume from the rounding function r that there is $\epsilon \gneq 0$ 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 **1** takes $r(x\delta_1^t)$, and **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 90.

Bargaining Revisited

What about SPE's in Example 2.8?

Because of the finite precision, there is a minimal round T with $r(\delta_{\mathbf{i}}^{T+1}) = 0$ for $\mathbf{i} = \mathbf{1}$ or $\mathbf{i} = \mathbf{2}$. For simplicity, assume that $\mathbf{i} = \mathbf{2}$ and agent $\mathbf{1}$ 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 ϵ in the rounding function satisfies $\epsilon <| \delta_1 - \delta_2 |$.

Theorem 2.10 (Benoit and Krishna, 1987)

Suppose that for each player $\mathtt{i}$ there is a static equilibrium in which $\mathtt{i}$ has a payoff that is strictly greater than its minmax value $\min_{s_{-\mathtt{i}}} \max_{s_\mathtt{i}} \boldsymbol{\mu}_\mathtt{i}^{expected}(s_{-\mathtt{i}}, s_\mathtt{i})$.
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 ∞?*
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 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 deviates, then the others punish her by minmaxing her for the rest of the game.
- In the second phase, the **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 ϵ range).
- Let $\alpha^*(\mathbf{i})$ 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 $\alpha^*(\mathbf{1})$ is played, then player **2**'s $\alpha^*(\mathbf{2})$ 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 **cooperating**, i.e. accepting a non-optimal payoff, namely the minmax value $\underline{v_i}$, in the first phase?
- If a player deviates, she is getting **in that round** a better payoff, but will be punished by all other players: they simply *minmax* her in all remaining rounds (and cooperate among them).
- So we choose R large enough, so that accepting just $\underline{v_i}$ in some rounds plus the terminal R-cycle gives a better payoff than *deviating and being minmaxed* for the rest of the game.

□

Proof (cont.)

- We are now designing the first phase. Given $\epsilon \gneq 0$, we choose T large enough, so that there exist sets of pure actions $\{a_1(\mathbf{i}), \ldots, a_{j_\mathbf{i}}(\mathbf{i})\}$ 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 ϵ-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.

$\square$

Proof (cont.)

- Deviating in this first phase does not pay off (minmaxing the deviating player). So these strategies are a NE as long as $T \gneq R \times n$.
- The average payoffs are within 2ϵ-range of v for $T \gneq R \times n \times \frac{\max_{\vec{a}} \boldsymbol{\mu}_{\mathbf{i}}(\vec{a}) - \underline{v}_{\mathbf{i}}}{\epsilon}$.

$\square$

2.3 Infinite Horizon Games

Strategic bargaining for infinite games

We introduce δ_1 factor for agent 1, δ_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.

Round	1's share	2's share	offerer
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$t-2$	$1-\delta_2(1-\delta_1\bar{\pi}_1)$		1
$t-1$		$1-\delta_1\bar{\pi}_1$	2
t	$\bar{\pi}_1$		1
$\vdots$	$\vdots$	$\vdots$	$\vdots$

Let $\overline{\pi_1}$ 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 $1-\delta_2(1-\delta_1\overline{\pi_1})$. Setting both equal gives us the result. □

- 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 $\langle \mathbf{A}, Act, O, \varrho, \boldsymbol{\mu} \rangle$.

Definition 2.12 (Average and discounted reward)

Let $r_{\mathbf{i}}^{(1)}, r_{\mathbf{i}}^{(2)}, \ldots$ denote an infinite sequence of payoffs for player $\mathbf{i}$ and let δ be a discount factor $0 \leq \delta \lneq 1$.
We define the *average reward* of player $\mathbf{i}$ as

$$\limsup_{k \to \infty} \frac{1}{k} \sum_{j=1}^{k} r_{\mathbf{i}}^{(j)}.$$

We define the *future discounted reward* of player $\mathbf{i}$ as

$$\sum_{j=1}^{\infty} \delta^j r_{\mathbf{i}}^{(j)}.$$

Model of infinitely repeated games (2)

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

- In each period t certain actions $a_1^t, \ldots, a_{l_t}^t$ are played: denoted by $\mathbf{a^t}$.
- $h^t = (\mathbf{a^0}, \ldots, \mathbf{a^{t-1}})$ is the sequence of all realized actions until t.
- $\mathbf{H^t}$ is the space of all possible period-t histories.
- **Pure strategy of $\mathfrak{i}$ in the infinite game:** $\mathbf{a_i}$ is a sequence of mappings $a_{\mathfrak{i}}^t$ (for each period t) from $\mathbf{H^t}$ into $A_{\mathfrak{i}}$. In each period t a pure strategy $a_{\mathfrak{i}}$ is chosen.
- **Mixed (behavioral) strategy in the infinite game:** $\mathbf{s_i}$ is a sequence of mappings $s_{\mathfrak{i}}^t$ (for each period t) from $\mathbf{H^t}$ into mixed actions $S_{\mathfrak{i}}$. In each period t a mixed strategy $s_{\mathfrak{i}}$ 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 $\vec{\mathbf{a}^t}$ and $\vec{\mathbf{s}^t}$ 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 $\mathbf{H}^{\mathbf{t}}$.
- 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 $\mathbf{i}$'s intention is to maximize the value

$$\mu_{\mathbf{i}}^{\text{repeated}} = E_{\vec{\mathbf{s}}}\,(1-\delta)\sum_{t=0}^{\infty}\delta^t \mu_{\mathbf{i}}(\vec{\mathbf{s}}^t(h^t))$$

- Why the factor $(1-\delta)$?
- Because $\sum_{i=0}^{\infty}\delta^i = \frac{1}{1-\delta} \rightsquigarrow$ 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 α^* be a NE of the stage game (above we called this *static equilibrium*).
- Then the strategies **"each player i plays $\alpha_{\mathbf{i}}^*$ 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**?
  $\rightsquigarrow$ 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 δ).

Feasible payoffs are convex outcomes of the stage game.

- **Question:** Suppose we have a **better** payoff profile $\langle r_1, \ldots r_{\mathbf{i}}, \ldots, r_n \rangle$ (i.e. all r_i are strictly greater than their minmax values) for a stage game $\langle \boldsymbol{A}, \mathrm{Act}, O, \varrho, \boldsymbol{\mu} \rangle$.
- 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 $\delta' \lneq 1$ 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 characeterizing 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 $a_{\mathbf{i}}$ 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 $m_j^{\mathbf{i}}$ 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 $\max_a \mu_i(\vec{a})$.
- 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

$$(1-\delta^t)v_i + \delta^t(1-\delta)\max_{\vec{a}} \mu_i(\vec{a}) + \delta^{t+1}\underline{v_i}$$

- which is less than v_i for $\delta \gneq \underline{\delta_i}$,
- where $\underline{\delta_i}$ is defined by $(1-\underline{\delta_i})\max_a \mu_i(\vec{a}) + \underline{\delta_i v_i} = v_i$. ($\underline{\delta_i}$ is smaller than 1 because $v_i \gneq \underline{v_i}$.

$\square$

- 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 $\rightsquigarrow$ 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 $v_{\mathbf{i}} \gneq \underline{v_{\mathbf{i}}}$ for all agents* **i***, there exists a $\delta' \lneq 1$ 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

3 Coalitional Games
- Coalition Formation in CFG's
- General Contract Nets
- Classes of Games
- The Core and its refinements
- Payoff Division: Shapley value and Banzhaf Index

Outline

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;
- the task allocation problem and present different types of contracts between agents (no IR-contract leads to the global optimum, even if all types are allowed), 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.

Coalition Structure

Definition 3.2 (Coalition Structure CS)

A coalition structure $\mathbf{CS}$ over the set $\mathbf{A}$ is any partition $\{C^1, \ldots, C^k\}$ of $\mathbf{A}$, i.e. $\bigcup_{j=1}^{k} C^j = \mathbf{A}$ and $C^i \cap C^j = \emptyset$ for $i \neq j$.
We denote by $\mathcal{CS}_M$ the set of all coalition structures $\mathbf{CS}$ over the set $M \subseteq \mathbf{A}$.
Finally, we define the social welfare of a coalition structure $\mathbf{CS}$ by

$$\boldsymbol{v}(\mathbf{CS}) := \sum_{C \in \mathbf{CS}} \boldsymbol{v}(C).$$

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 $\mathcal{A}$ by finding a coalition structure

$$\mathbf{CS}^* = \text{arg max}_{\mathbf{CS} \in \mathcal{CS}_{\mathbf{A}}} \boldsymbol{v}(\mathbf{CS}),$$

where

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

How many coalition structures are there?

A lot: $\Omega(|A|^{\frac{|A|}{2}})$. Enumerating is feasible for $|A| < 15$.

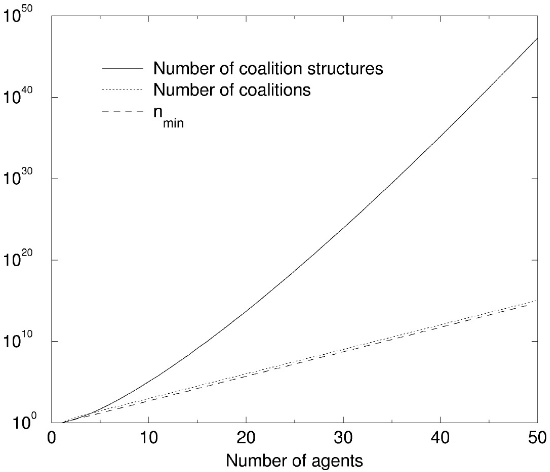


Figure 8: Number of Coalition (Structures).

Approximation of $v(\mathrm{CS})$.

How can we approximate $v(\mathrm{CS})$?

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

$$\mathrm{CS}^*_{\mathrm{N}} = \arg\max_{\mathrm{CS} \in \mathrm{N}} v(\mathrm{CS}).$$

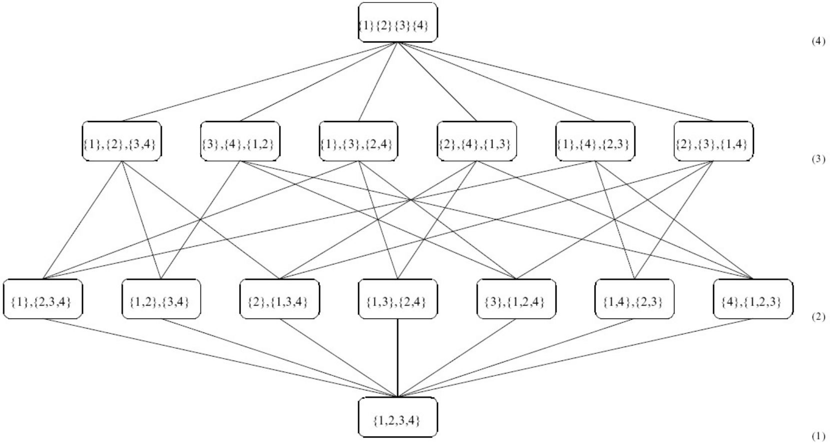


Figure 9: Coalition Structure Graph $\mathcal{CS}_A$.

We want our approximation as good as possible.

We want to find a small k and a small N such that

$$\frac{\boldsymbol{v}(\mathbf{CS}^*)}{\boldsymbol{v}(\mathbf{CS}^*_{\mathrm{N}})} \leq k.$$

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 ($k = |\mathbf{A}|$) 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* $\mathbf{CS}$*-graph. Using this search, the bound $k = |\mathbb{A}|$ can be taken. This bound is tight and the number of nodes searched is $2^{|\mathbb{A}|-1}$.*

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

Proof.

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

$$\boldsymbol{v}(\mathbf{CS}^*) \leq |\mathbf{A}| \max_S \boldsymbol{v}(S) \leq |\mathbf{A}| \max_{\mathbf{CS} \in \mathbf{N}} \boldsymbol{v}(\mathbf{CS}) = |\mathbf{A}| \boldsymbol{v}(\mathbf{CS}^*_{\mathbf{N}})$$

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

Number of **coalition structures** at the second lowest level: $\frac{1}{2}(2^{\mathbf{A}} - 2) = 2^{\mathbf{A}-1} - 1$.

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

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

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

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:*

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

where $h =_{def} \lfloor \frac{|\mathbf{A}|-l}{2} \rfloor + 2$.

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

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 $|A|$;
3. values were superadditive;
4. values were subadditive.

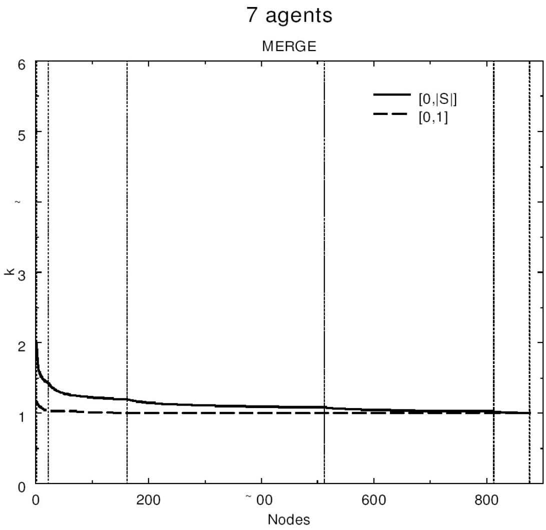


Figure 10: MERGE.

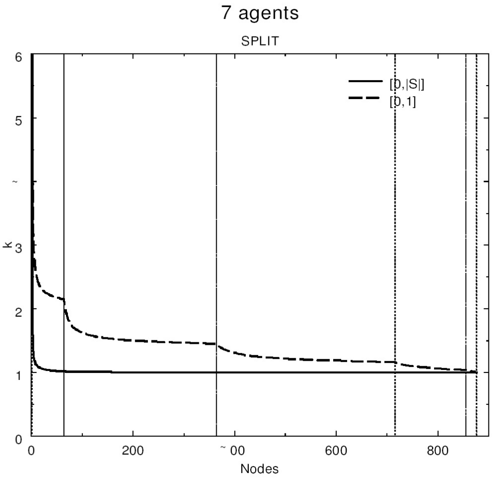


Figure 11: SPLIT.

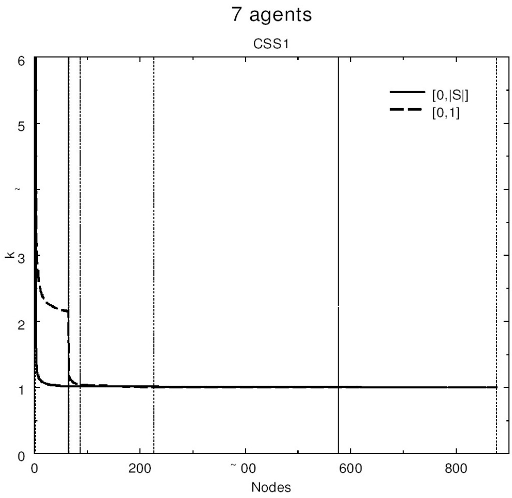


Figure 12: CS-Search-1.

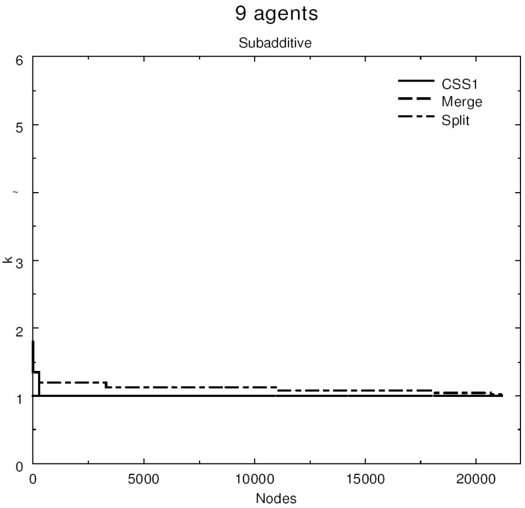


Figure 13: Subadditive Values.

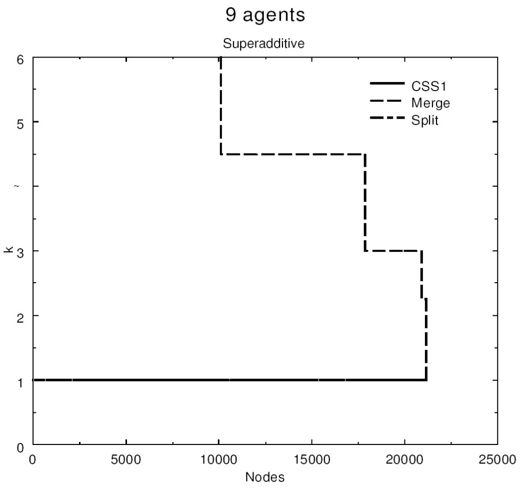


Figure 14: Superadditive Values.

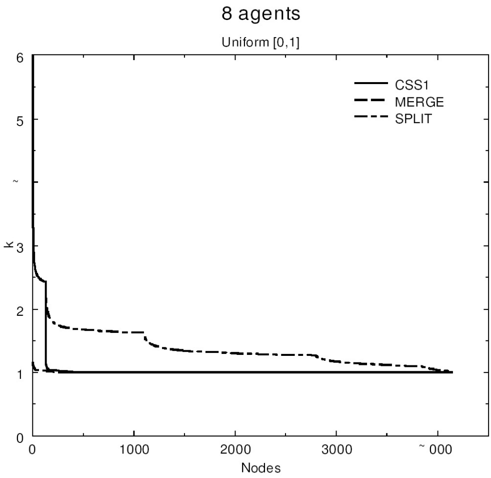


Figure 15: Coalition values chosen uniformly from $[0, 1]$.

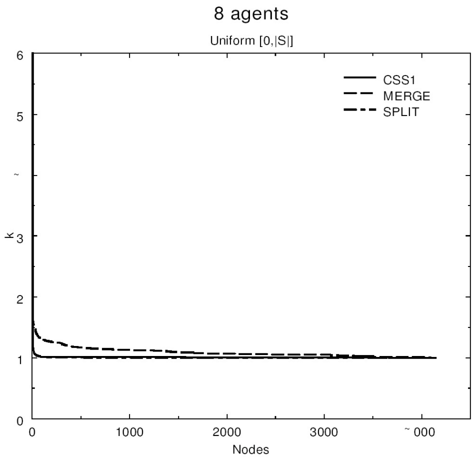


Figure 16: Coalition values chosen uniformly from $[0, |S|]$.

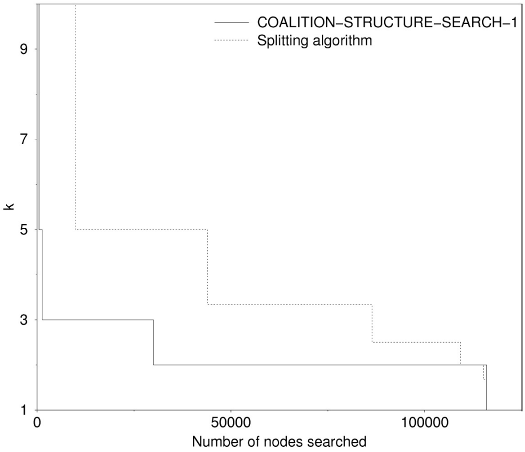


Figure 17: Comparing CS-Search-1 with SPLIT.

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 $\boldsymbol{v}(\mathrm{CS})$ but also $\boldsymbol{v}(S)$ in the searched CS'.

3.2 General Contract Nets

How to distribute tasks?

- Global Market Mechanisms. Implementations use a **single centralised mediator**.
- *Announce, bid, award* -cycle. **Distributed Negotiation**.

We need the following:

1. **Define a task allocation problem in precise terms.**
2. **Define a formal model for making bidding and awarding decisions.**

Definition 3.6 (Task-Allocation Problem)

A task allocation problem is given by

1. a set of tasks T,
2. a set of agents A,
3. a cost function $\mathbf{cost}_i : 2^T \longrightarrow \mathbb{R} \cup \{\infty\}$ (stating the costs that agent i incurs by handling some tasks), and
4. the initial allocation of tasks
$$\langle T_1^{init}, \dots, T_{|A|}^{init} \rangle,$$
where $T = \bigcup_{i \in A} T_i^{init}$, $T_i^{init} \cap T_j^{init} = \emptyset$ for $i \neq j$.

Definition 3.7 (Accepting Contracts, Allocating Tasks)

A contractee q accepts a contract if it gets paid more than the marginal cost of handling the tasks of the contract

$$
\begin{aligned}
MC^{add}(T^{contract}|T_{\mathsf{q}}) \quad =_{def} \quad & \mathbf{cost}_{\mathsf{q}}(T^{contract} \cup T_{\mathsf{q}}) \\
& -\mathbf{cost}_{\mathsf{q}}(T_{\mathsf{q}}).
\end{aligned}
$$

A contractor r is willing to allocate the tasks $T^{contract}$ from its current task set T_{r} to a contractee, if it has to pay less than it saves by handling them itself:

$$
\begin{aligned}
MC^{remove}(T^{contract}|T_{\mathsf{r}}) \quad =_{def} \quad & \mathbf{cost}_{\mathsf{r}}(T_{\mathsf{r}}) \\
& -\mathbf{cost}_{\mathsf{r}}(T_{\mathsf{r}} - T^{contract}).
\end{aligned}
$$

Definition 3.8 (The Protocol)

Agents suggest contracts to others and make their decisions according to the above MC^{add} and MC^{remove} sets.
Agents can be both contractors and contractees. Tasks can be recontracted.

- The protocol is domain independent.
- Can only improve at each step: Hill-climbing in the space of all task allocations. Maximum is social welfare: $-\sum_{i \in A} \mathbf{cost}_i(T_i)$.
- **Anytime algorithm!**

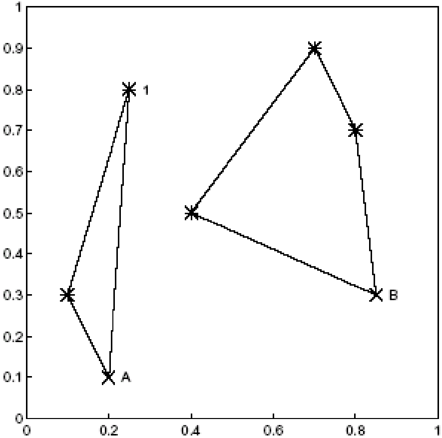


Figure 18: TSP as Task Allocation Problem.

Definition 3.9 (O-, C-, S-, M- Contracts)

A contract is called of type

O (Original): only one task is moved: $\langle T_{i,j}, \rho_{i,j} \rangle$, $|T_{i,j}| = 1$.

C (Cluster): a set of tasks is moved: $\langle T_{i,j}, \rho_{i,j} \rangle$, $|T_{i,j}| \gneqq 1$.

S (Swap): if a pair of agents swaps a pair of tasks: $\langle T_{i,j}, T_{j,i}, \rho_{i,j}, \rho_{j,i} \rangle$, $|T_{i,j}| = |T_{j,i}| = 1$.

M (Multi): if more than two agents are involved in an atomic exchange of tasks: $\langle \mathbf{T}, \rho \rangle$, both are $|A| \times |A|$ matrices. At least 3 elements are non-empty, $|T_{i,j}| \leq 1$.

Lemma 3.10 (O-Path reaches Global Optimum)

A path of O-contracts always exists from any task allocation to the optimal one. The length of the shortest such path is at most $|T|$.

Does that solve our problem? (Lookahead)

Definition 3.11 (Task Allocation Graph)

The *task allocation graph* has as vertices all possible task allocations (i.e. $|A|^{|T|}$) and directed edges from one vertex to another if there is a possible contract leading from one to the other.

For O-contracts, searching the graph using breadth-first search takes how much time?

Lemma 3.12 (Allocation graph is sparse)

We assume that there are at least 2 agents and 2 tasks. We consider the task allocation graph for O contracts*. Then the fraction*

$$\frac{\textit{number of edges}}{\textit{number of edges in the fully connected graph}}$$

converges to 0 both for $|T| \rightarrow \infty$ as well as $|\mathbf{A}| \rightarrow \infty$

Lemma 3.13 (No IR-Path to Global Optimum)

There are instances where no path of IR O contracts exists from the initial allocation to the optimal one. The length of the shortest IR path (if it exists) may be greater than $|T|$.
But the shortest IR path is never greater than $|A|^{|T|} - (|A| - 1)|T|$.

Problem: local maxima.

A contract may be individually rational but the task allocation is not globally optimal.

Lemma 3.14 (No Path)

There are instances where no path of C-contracts *(IR or not) exists from the initial allocation to the optimal one.*

k-optimal

A task allocation is called k-optimal if no beneficial C-contract with clusters of k tasks can be made between any two agents.

Let $m \lneqq n$. Does

- m optimality imply n optimality;
- n optimality imply m optimality?

Lemma 3.15 (No Path)

There are instances where no path of S-contracts *(IR or not) exists from the initial allocation to the optimal one.*

Lemma 3.16 (No Path)

There are instances where no path of M-contracts *(IR or not) exists from the initial allocation to the optimal one.*

Lemma 3.17 (Reachable allocations for S contracts)

We assume that there are at least 2 agents and 2 tasks. We consider the task allocation graph for S contracts*. Given any vertex v the fraction*

$$\frac{\textit{number of vertices reachable from } v}{\textit{number of all vertices}}$$

converges to 0 both for $|T| \to \infty$ as well as for $|\mathbf{A}| \to \infty$.

Proof.

S contracts preserve the number of tasks of each agent. So any vertex has certain allocations $t_1, \ldots, t_{|\mathbf{A}|}$ for the agents. How many allocations determined by this sequence are there? There are exactly $|T|$**!** many. This is to be divided by $|A|^{|T|}$. $\square$

Theorem 3.18 (All Types necessary)

For each of the 4 types there exist task allocations where no IR contract with the remaining 3 types is possible, *but an IR contract with the fourth type is.*

Proof.

Consider O contracts (as fourth type). One task and 2 agents: $T_1 = \{t_1\}$, $T_2 = \emptyset$.
$c_1(\emptyset) = 0, c_1(\{t_1\}) = 2, c_2(\emptyset) = 0, c_2(\{t_1\}) = 1$. The O contract of moving t_1 would decrease global cost by 1. No C-, S-, or M-contract is possible.

To show the same for C or S contracts, two agents and two tasks suffice.

For M-contracts 3 agents and 3 tasks are needed. □

Theorem 3.19 (O-, C-, S-, M- $\not\Rightarrow$ Global Optima)

There are instances of the task allocation problem where no IR sequence from the initial task allocation to the optimal one exists *using O-, C-, S-, and M- contracts.*

Proof.

Construct cost functions such that the deal *where agent 1 gives one task to agent 2 and agent 2 gives 2 tasks to agent 1* is the only one increasing welfare. This deal is not possible with O-, C-, S-, or M-contracts. $\square$

Corollary 3.20 (O-, C-, S-, M- $\not\Rightarrow$ Global Optima)

There are instances of the task allocation problem where no IR sequence from the initial task allocation to the optimal one exists using any pair or triple of O-, C-, S-, or M- contracts.

Definition 3.21 (OCSM Nets)

A *OCSM-contract* is a pair $\langle \boldsymbol{T}, \boldsymbol{\rho} \rangle$ of $|\mathbf{A}| \times |\mathbf{A}|$ matrices. An element $T_{\mathbf{i},\mathbf{j}}$ stands for the set of tasks that agent $\mathbf{i}$ gives to agent $\mathbf{j}$. $\rho_{\mathbf{i},\mathbf{j}}$ is the amount that $\mathbf{i}$ pays to $\mathbf{j}$.

- **How many OCSM contracts are there?**
- **How much space is needed to represent one?**

Theorem 3.22 (OCSM-Nets Suffice)

Let $|\mathbf{A}|$ and $|T|$ be finite. If a protocol allows OCSM-contracts, any hill-climbing algorithm finds the globally optimal task allocation in a finite number of steps without backtracking.

Proof.

An OCSM contract can move from any task allocation to any other (in one step). So moving to the optimum is IR. Any hill-climbing algorithm strictly improves welfare. As there are only finitely many allocations, the theorem follows. □

Theorem 3.23 (OCSM-Nets are Necessary)

If a protocol does not allow a certain OCSM contract, then there are instances of the task allocation problem where no IR-sequence exists from the initial allocation to the optimal one.

Proof.

If one OCSM contract is not allowed, then the task allocation graph contains two vertices without an edge. We let the initial and the optimal allocation be these two vertices. We construct it in such a way, that all adjacent vertices to the vertex with the initial allocation have lower social welfare. So there is no way out of the initial allocation. □

[?] consider the multiagent version of the TSP problem and apply different sorts of contracts to it.

Several salesmen visit several cities on the unit square. Each city must be visited by exactly one salesman. They all have to return home and want to minimise their travel costs.

salesman = agent, task = city

- Experiments with up to 8 agents and 8 tasks. Initial allocation randomly chosen.
- Ratio bound (welfare of obtained local optimum divided by global optimum) and mean ratio bound (over 1000 TSP instances) were computed (all for fixed number of agents and tasks). Global optimum was computed using IDA*.
- A protocol consisting of 5 intervals is considered. In each interval, a particular contract type was considered (and all possible contracts with that type): 1024 different sequences.
- In each interval a particular order for the agents and the tasks is used. First all contracts involving agent 1. Then all involving agent 2 etc. Thus it makes sense to have subsequent intervals with the same contract type.

Order No.	Sequence	Social Welfare
1	OCOCO	1.03113
2	OOCCO	1.03268
3	OCCOC	1.03276
4	OOCOC	1.03279
5	OCOOC	1.03413
6	SOCOC	1.03488
7	SOCCO	1.03536
8	COCOC	1.03755
9	OCOCC	1.03857
10	MCOCO	1.03945
11	OCCCO	1.03954
12	MOCCO	1.03988
13	MOCOC	1.04001
14	MCCOC	1.04304
15	COCCO	1.04407

Table 1: The best Contract sequences.

Order No.	Sequence	Social Welfare
1	OCOCO	1.03113
2	OOCCO	1.03268
3	OCCOC	1.03276
4	OOCOC	1.03279
375	*C-local*	*1.13557*
565	*O-local*	*1.2025*
579	OOOOO	1.21298
696	CCCCC	1.23515
1021	CSSSS	1.61181
1022	CMMMM	1.65965
1023	MMMMM	1.76634
1024	SSSSS	1.89321

Table 2: Best, average and worst Contracts.

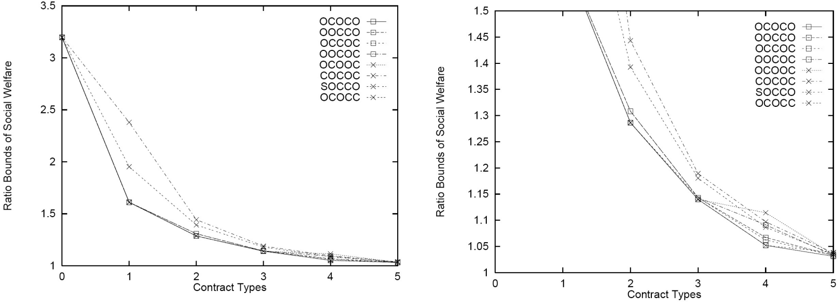
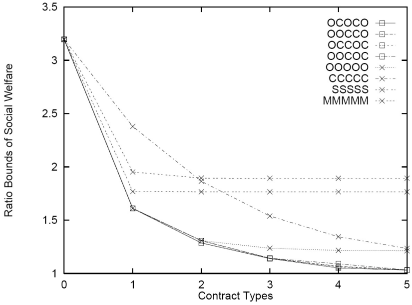


Figure 19: Ratio bounds (second graph is zoomed in).



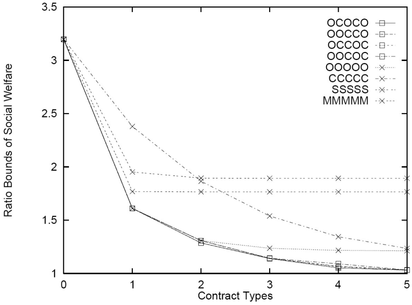
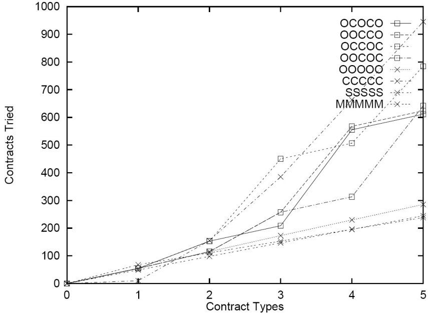
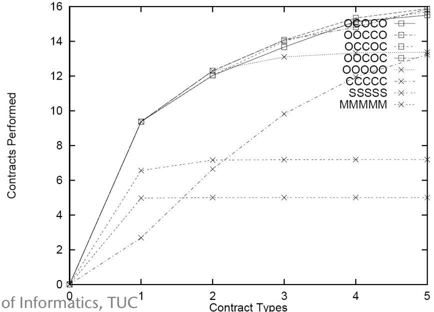


Figure 20: Ratio bounds for the 4 best and single type sequences.


Contracts Tried
Contract Types
OCOCO
OOCCO
OCCOC
OOCOC
OOOOO
CCCCC
SSSSS
MMMMM


Contracts Performed
Contract Types
OCOCO
OOCCO
OCCOC
OOCOC
OOOOO
CCCCC
SSSSS
MMMMM

1. Best protocol is 3.1% off the optimum.
2. 12 best protocols are between 3% and 4% off the optimum.
3. Single type contracts do not behave well.
4. More contracts are tried and performed in the second interval (compared with the first).
5. The more mixture between contract types, the higher the social welfare.

3.3 Classes of Games

Idea: Consider a protocol (to build coalitions) as a game and consider Nash-equilibrium.

Problem: Nash-Eq is too weak!

Definition 3.24 (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 3.25 (Monotone Games)

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

$$\boldsymbol{v}(C) \leq \boldsymbol{v}(D),$$

for every pair of coalitions $C, D \subseteq \boldsymbol{A}$ 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 3.26 (Superadditive Games)

A CFG $\langle \mathbf{A}, \boldsymbol{v} \rangle$ is called superadditive, if

$$\boldsymbol{v}(S \cup T) \geq \boldsymbol{v}(S) + \boldsymbol{v}(T),$$

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

Lemma 3.27

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 3.28 (Subadditive Games)

A CFG $\langle \mathbf{A}, \boldsymbol{v} \rangle$ is called subadditive, if

$$\boldsymbol{v}(S \cup T) \lneqq \boldsymbol{v}(S) + \boldsymbol{v}(T),$$

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

Coalition formation for subadditive games is trivial.

Superadditive Cover

Definition 3.29 (Superadditive Cover)

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

$$\boldsymbol{v}^*(C) := max_{\mathbf{CS} \in \mathcal{CS}_C} \boldsymbol{v}(\mathbf{CS})$$

This game is called the superadditive cover of G.

Convex Games

Definition 3.30 (Convex Game)

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

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

- **Convex games are superadditive.**
- **Superadditive games are monotone.**
- The other directions do not hold.

Example 3.31 (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 3.32 (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 α ($0 \leq \alpha \leq 1$). The third player gets nothing.

Example 3.33 (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 ($\gneq 50\%$).
Version 2: Any majority over 80% wins.

How do the $v(S)$ look like in the last three examples?

Definition 3.34 (Payoff Vector)

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

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

Note that the last condition is only supposed to hold for all **coalitions in the given coalition structure**.

If $\sum_{i \in C} x_i \gneq v(C)$ would mean that a coalition gets more than its value, that is not possible.

3.4 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 CS forms.

Definition 3.35 (Core of a CFG)

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

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

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.

If $\sum_{\mathbf{i} \in S} x_{\mathbf{i}} \lneq \boldsymbol{v}(S)$, then these agents would form a coalition and get a higher payoff than in CS.

When the grand coalition forms, we can simplify the last definition.

Definition 3.36 (Core of Superadditive Games)

The core of a supperadditive CFG is the set of all payoff vectors $\langle x_1, \ldots, x_n \rangle$ such that the following holds:

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

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

What about the core in the above examples?

Lemma 3.37

If $\langle \mathbf{CS}, \langle x_1, \ldots, x_n \rangle \rangle$ *is in the core of a CFG* $\langle \mathbf{A}, \boldsymbol{v} \rangle$*, then* $\boldsymbol{v}(\mathbf{CS}) \geq \boldsymbol{v}(\mathbf{CS}')$ *for all coalition structures* $\mathbf{CS}' \in \mathcal{CS}_{\mathbf{A}}$.

Proof.

We can write $\boldsymbol{v}(\mathbf{CS}) = \sum_{\mathbf{i} \in \mathbf{A}} x_{\mathbf{i}} = \sum_{C' \in \mathbf{CS}'} x(C')$ and $\boldsymbol{v}(\mathbf{CS}') = \sum_{C' \in \mathbf{CS}'} \boldsymbol{v}(C')$.
Because of the definition of the core, $x(C') \geq \boldsymbol{v}(C')$ for all C' and therefore $\boldsymbol{v}(\mathbf{CS}) \geq \boldsymbol{v}(C')$. □

Theorem 3.38 (Core and superadditive cover)

Let a CFG $G = \langle \mathbf{A}, \boldsymbol{v} \rangle$ 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 $\rightsquigarrow$ **exercise**

Theorem 3.39 (Convex games and their cores)

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

Proof.

Let π be a permutation of $\mathbf{A}$ and let $S_\pi(i)$ be the set of all **predecessors** of i wrt. π.
We claim that for $x_i := \boldsymbol{v}(S_\pi(i) \cup \{i\}) - \boldsymbol{v}(S_\pi(i))$, 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 ($\rightsquigarrow$ **exercise**). □

(Proof of Theorem 3.39, cont.)

Assume there is a coalition $C = \{i_1, \ldots, i_s\}$ such that $\boldsymbol{v}(C) \not\geq x(C)$. Wlog we assume $\pi(i_1) \leq \ldots \leq \pi(i_s)$. Obviously

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

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:

$$\boldsymbol{v}(T_j \cup \{i_j\}) - \boldsymbol{v}(T_j) \leq \boldsymbol{v}(S_j \cup \{i_j\}) - \boldsymbol{v}(S_j) = x_{i_j}$$

Adding these pairs up, we get $\boldsymbol{v}(C) \leq x(C)$, which is a contradiction.

3.5 Payoff Division: Shapley value and Banzhaf Index

We now assume w.l.o.g. that the grand coalition forms.
The payoff division should be fair between the agents, otherwise they would leave the coalition.

Definition 3.40 (Dummies, Interchangeable)

Agent $\mathbf{i}$ is called a dummy, if for all coalitions S with $\mathbf{i} \notin S$:

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

Agents $\mathbf{i}$ and $\mathbf{j}$ are called interchangeable, if for all coalitions S with $\mathbf{i} \in S$ and $\mathbf{j} \notin S$:

$$\boldsymbol{v}(S \setminus \{\mathbf{i}\} \cup \{\mathbf{j}\}) = \boldsymbol{v}(S)$$

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 3.41 (Marginal Contribution in two games)

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

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

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 3.42 (Shapley-Value: 1st Characterisation)

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

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

Theorem 3.43 (Shapley-Value: 2nd Characterisation)

For a CFG $G = \langle \mathbf{A}, \boldsymbol{v} \rangle$ there is only one payoff division satisfying the efficiency, symmetry and marginality. *It is the* Shapley value.

The **expected gain** can be computed by taking a random joining order and computing the Shapley value.

- $(|\mathbf{A}| - S)!$ is the number of all possible joining orders of the agents (to form a coalition).
- There are $|S|!$ ways for S to be built before $\mathtt{i}$'s joining. There are $(|\mathbf{A}| - |S| - 1)!$ ways for the remaining agents to form S (after $\mathtt{i}$).
- $(\boldsymbol{v}(S \cup \{\mathtt{i}\}) - \boldsymbol{v}(S))$ is $\mathtt{i}$'s marginal contribution when added to set S.
- The Shapley value sums up the marginal contributions of agent $\mathtt{i}$ *averaged over all joining orders*.

We have shown in Theorem 3.39 that convex games have non-empty cores. In fact, we can show a stronger statement.

Theorem 3.44 (Shapley and Core for convex games)

In each convex game, at least the Shapley value is contained in the core.

Definition 3.45 (Banzhaf Index)

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

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

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

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

Outline (1)

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.

Candidates and Voters

From now on we assume there is a fixed set of alternatives (candidates, outcomes) O, in addition to the set of agents $\mathcal{A}$, 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).

Often e.g. in the french elections, runoff systems are considered: two candidates are singled out in the first round, one of which is then selcted in the second round.

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**.

Condorcet helps

	1	2	3
1	A	B	C
2	B	C	A
3	C	B	A

Figure 22: A Tie, but Condorcet helps.

Condorcet does not help

	1	2	3
1	A	B	C
2	B	C	A
3	C	A	B

Figure 23: Strict Condorcet rules out all candidates.

Comparing A and B: majority for A.

Comparing A and C: majority for C.

Comparing B and C: majority for B.

Desired Preference ordering: $A > B > C > A$

Smith Set

Another interesting set is the following

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.

Approval Voting

Definition 4.4 (Winner in Approval Voting)

A **winner in approval voting** is any candidate that received **at least as many votes as any other candidate**.

Can we model the situation in Figure 23 in approval voting?

Lemma 4.5 (Approval Voting and Condorcet)

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

Variants of Borda

Definition 4.6 (Borda Protocol: Standard and Nauru)

In *standard* Borda, each voter gives its best candidate $|O|$ points, the second best gets $|O| - 1$ points, etc.

In **Nauru** Borda, any first preference as assigned **1** point, any second just $1/2$, any third just $1/3$ and so on.

After all votes have been cast, they are *summed up, across all voters*. Winners are those with the *highest count*.

Originally, Jean-Charles de Borda wanted his system to determine a single winner. He also assumed all voters are **honest**.

Truncated Ballots

What, if voters do not want to **express full preference on all** candidates: truncated ballots.

Nauru: All candidates must be ranked.

Kiribati: Rank only a subset (but this subset completely), and all others get 0 points.

Modified BC: Points given depend on the **number of candidates ranked** (in each individual ballot).

Problem with Kiribati: Tactical voting. Bullet votes are more effective than fully ranked ballots.

Ties between candidates

Later we introduce weak orders: although they are **total**, they allow for ties (candidates among which the voter is indifferent).

Winner turns loser and loser turns winner.

Agent	Preferences
1	$A \succ B \succ C \succ D$
2	$B \succ C \succ D \succ A$
3	$C \succ D \succ A \succ B$
4	$A \succ B \succ C \succ D$
5	$B \succ C \succ D \succ A$
6	$C \succ D \succ A \succ B$
7	$A \succ B \succ C \succ D$
Borda count	C wins: 20, B: 19, A: 18, D loses: 13
Borda count without D	A wins: 15, B: 14, C loses: 13

Figure 24: Winner turns loser and vice versa.

Absolute Majority, but not elected

Example 4.7 (Nauru/Standard Borda, Plurality Voting)

	1	2	3	4
51 voters	A	C	B	D
5 voters	C	B	D	A
23 voters	B	C	D	A
21 voters	D	C	B	A

Who wins and who should win???

1. Plurality voting:
2. Standard Borda:
3. Nauru Borda:

Borda and Tactical Voting

Example 4.8 (Compromising and Burying)

We assume four cities M, N, K, and C want to become the capital of the state. M has most voters but is far away from the others. N, K, and C are close to the centre of the state.

	1	2	3	4
42 % (M)	M	N	C	K
26 % (N)	N	C	K	M
17 % (K)	K	C	N	M
15 % (C)	C	K	N	M

Borda would elect N. **How could voters of K change their poll to ensure C is chosen and not N?**

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?

35% of agents have preferences $C \succ D \succ B \succ A$
33% of agents have preferences $A \succ C \succ D \succ B$
32% of agents have preferences $B \succ A \succ C \succ D$

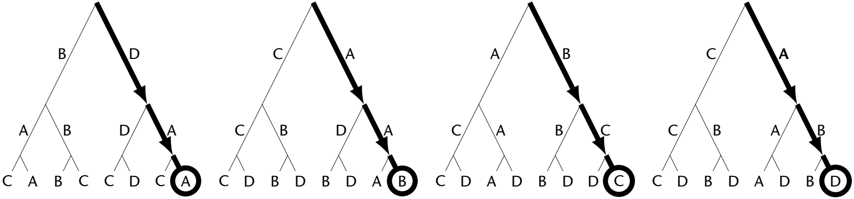


Figure 25: Four different orderings and four alternatives.

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”, than “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 expresss the ballot of an agent as a dag, a directed acyclic graph.

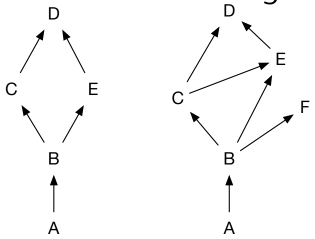


Figure 26: Two Examples of dags.

In all voting systems considered in Subsection 4.1, the voters' ballot can be modelled with dags.

Dags are just strict partial orders: irreflexive and transitive binary relations. Most ballots are even based on **linear orders** (no indifference between any pair of candidates).

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)

$\rightsquigarrow$ 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 $\prec_i$

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

$$\prec_i \subseteq O \times O.$$

- Which properties should we assume from such a binary relation?

Some Terminology on Posets

Here are some important properties of binary relations $\prec$:

Reflexivity: For all x: $x \prec x$.
Transitivity: For all x, y, z: if $x \prec y$, $y \prec z$ then $x \prec z$.
Antisymmetry: For all x: $x \prec y$ and $y \prec x$ implies $x = y$.
Totality: For all x, y: either $x = y$, or $x \prec y$ or $y \prec x$.
Irreflexivity: For all x: not $x \prec x$.
Asymmetry: If $x \prec y$ then not $y \prec x$.

A (non-strict) partial order, denoted by $\preceq$ or $\preccurlyeq$ or $\leq$ or $\leqslant$ or $\leqq$, is any relation satisfying reflexivity, transitivity and antisymmetry.

A strict partial order, denoted by $\precneqq$ or $\lneq$ or $\lneqq$ or $\lvertneqq$, is any relation satisfying irreflexivity, and transitivity (and therefore also asymmetry).

Often, one simply writes $\prec$ or $<$ to denote a strict order.

Ballots of the voters (4)

What are the **right** properties for $\prec_i$?

- To express *dags*: irreflexive, transitive (both impy asymmetry) (*strict partial orders* $\lneqq$).

  We could also use non-strict partial orders $\leqq$, which are in one-to-one correspondence via:
  - $\leqq$ is the *reflexive closure* of $\lneqq$,
  - $\lneqq$ is the *irreflexive kernel* of $\leqq$.

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 298 elements C and E are tied). **Voters have more freedom when they are allowed not to order candidates.**

Ballots of the voters (5)

- **Second try:** We could also consider total (or linear) orders: they are transitive and **strictly total** (for all $a \neq b$ 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 4.9 (Weak Order, $L(O)$)

Any binary relation $\precsim$ satisfying transitivity and totality (for all a, b: $a \precsim b$ or $b \precsim a$) 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).

While weak orders rank all pairs of candidates, they allow ties: it is perfectly possible that there are pairs $a \neq b$ with $a \precsim b$ and $b \precsim a$.

Thus a and b are indifferent: the weak order treats them as equivalent. This can not happen with linear partial orders, because they are antisymmetric.

Using partial orders, such ties have to be modelled as incomparable. However, note the subtle differences between the two concepts.

Ballots of the voters = Weak Orders (cont.)

Any weak order induces a strict weak order $\prec_i$:

- $a \prec_i b$ iff $a \precsim_i b$ and not $b \precsim_i a$: "**i** strictly prefers b over a".

$\prec_i$ is irreflexive and transitive, but not **total** anymore.

So we allow elements to be "equivalent" (or **indifferent**):

- $a \sim_i b$: $a \precsim_i b$ and $b \precsim_i a$. "**i** is indifferent between a and b".

However, **not any strict 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.

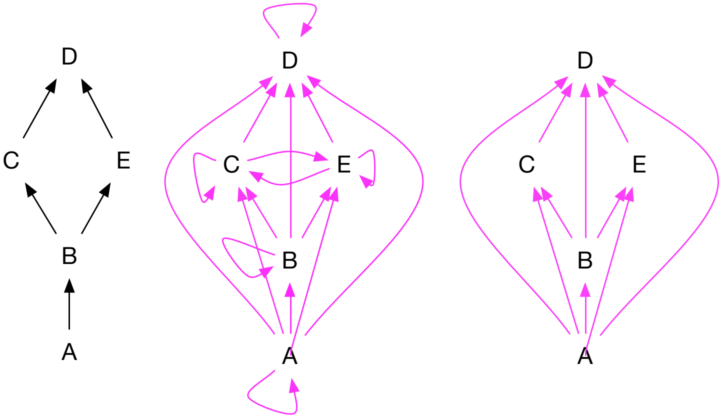


Figure 27: Equivalent modelings as dag (left), as weak order (middle), and as strict weak order.

Definition 4.10 (Preference Profile)

A preference profile for agents $1, \ldots, |A|$ is a tuple

$$\langle \precsim_1, \ldots, \precsim_{|A|} \rangle \in L^{|A|} \quad (:= \prod_{i=1}^{|A|} L)$$

We often write $\vec{\precsim}$ for $\langle \precsim_1, \ldots, \precsim_{|A|} \rangle$ if the set of agents is clear from context.

So the weak orders $\precsim_i$ 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

$$U \subseteq \prod_{\mathbf{i}=1}^{|\mathbf{A}|} L(V).$$

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 three different election systems. The characterization of the voters preferences is different in all of them:

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 $\precsim$.

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

$$\mathbf{C}^* : V \times U \to O; (v, \vec{\prec}) \mapsto o$$

as an election. The outcome is one single winner.

Social Choice Correspondence (SCC): We define any

$$\mathbf{W}^* : V \times U \to 2^O; (v, \vec{\prec}) \mapsto v'$$

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

Social Welfare Function (SWF): This approach views a

$$\mathbf{f}^* : U \to L; \vec{\precsim} \mapsto \precsim^*$$

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

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 $\precsim$?

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

For a strict total order $\prec$, $top(\prec)$ and $bot(\prec)$ are singletons.

Dictators (2)

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

Let now $U \subseteq \prod_{\mathbf{i}=1}^{|\mathbf{A}|} L(V)$ be given.

Social Choice Function: A dictator is an agent $\mathbf{i}$ if

$$\text{for all } \vec{\preceq}: \quad top(\prec_{\mathbf{i}}|_v) = \{\mathbf{C^*}(v, \vec{\preceq})\}$$

Social Choice Correspondence: A dictator is an agent $\mathbf{i}$ if

$$\text{for all } \vec{\preceq}: \quad top(\prec_{\mathbf{i}}|_v) \subseteq \mathbf{W^*}(v, \vec{\preceq})$$

Social Welfare Function: A dictator is an agent $\mathbf{i}$ if

$$\text{for all } \vec{\precsim}: \quad \text{for all } o, o' \in O,\ o \prec_{\mathbf{i}} o' \Rightarrow o \prec^* o'$$

where $\prec^*$ is the **strict** version of the weak order $\mathbf{f^*}(\langle \precsim_1, \ldots, \precsim_{|\mathbf{A}|}\rangle)$

4.3 Social Choice Functions

This section: ballots are strict total orders $\prec$.

Definition 4.11 (Social choice function (SCF))

A *social choice function* is any function

$$\mathbf{C}^* : V \times U \to O; (v, \vec{\prec}) \mapsto o$$

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 4.12 (Unanimity, Monotonicity)

A SCF $\mathbf{C}^*$ satisfies

surjectivity: for any $v \in V$ and outcome $o \in v$ there is a profile $\vec{\preceq}$ such that $\mathbf{C}^*(v, \vec{\preceq}) = o$.

unanimity: for any profile $\vec{\preceq}$, $v \in V$ and outcome $o \in v$:
If for all $1 \leq \mathbf{i} \leq |\mathbf{A}| : \; o = top(\prec_{\mathbf{i}}|_v)$ then $\mathbf{C}^*(v, \vec{\preceq}) = o$,

Unanimity implies surjectivity.

strong monotonicity: for any $\vec{\preceq}$, $v \in V$ and $o = \mathbf{C}^*(v, \vec{\preceq}) \in v$:
if $\vec{\preceq}'$ is a different profile such that
for all $o' \neq o$ and $\mathbf{i}$: "$o' \prec_{\mathbf{i}} o$ implies $o' \prec'_{\mathbf{i}} o$",
then $\mathbf{C}^*(v, \vec{\preceq}') = o$.

Strong monotonicity means: **any additional support for a winning alternative, should only benefit that alternative.**

Example 4.13 (Plurality with runoff)

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

6 voters support $B \prec C \prec A$
5 voters support $C \prec A \prec B$
6 voters support $A \prec B \prec C$

Under plurality with runoff, A and C make it to the second round, where A 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 A.**

But now, A and B make it to the second round, where B beats A with 9 to 8.

Theorem 4.14 (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 331.

In an exercise, you have to show that surjectivity together with strong monotonicity implies unanimity.

Theorem 4.15 (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_{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. **Note: this is among all $\vec{\precsim}$.**

Definition 4.16 (Decisive sets $G \subset \mathbf{A}$)

A set $G \subset \mathbf{A}$ is called decisive for the pair $(p, o) \in O^2$, if for all $\vec{\precsim}$: "for all $\mathbf{i} \in G$: $p \prec_{\mathbf{i}} o$" implies $\mathbf{C}^*(O, \vec{\precsim}) \neq p$.

Definition 4.17 (Independence)

A SCF $\mathbf{C}^*$ satisfies the Independence property, if the following holds: If $\mathbf{C}^*(O, \vec{\precsim}) = o$, then $\mathbf{C}^*(O, \vec{\precsim}') \neq p$ for all $p \neq o$ and all $\vec{\precsim}'$ with "for all $\mathbf{i}$: $p \prec_{\mathbf{i}} o$ iff $p \prec'_{\mathbf{i}} o$".

Contraction-Lemma

Two observations are important:

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

Contraction-Lemma

Let $G \subseteq$ **A** with $|\, G \,| \geq 2$ be **decisive on all pairs** $(p, o) \in O^2$ and $G = G_1 \dot{\cup} G_2$ (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).

Theorem of Muller-Satterthwaite (1977) (1)

Proof of Theorem 4.15.

- **Claim**: $G \subset \mathbf{A}$ is decisive on any pair $(p', o') \in O^2$, if the following holds: $G = \{\mathbf{i} \in \mathbf{A} : p \prec_{\mathbf{i}} o\}$ s.t. their ranking ensures $\mathbf{C}^*(O, \vec{\precsim}) \neq p$. $\rightsquigarrow$ whiteboard.
- We then show that strong monotonicity implies independence. Hint: Given a profile $\vec{\precsim}$ consider the following profile $\vec{\precsim}''$ and apply strong monotonicity (twice): "$o \prec''_{\mathbf{i}} o'$" iff: $o \prec_{\mathbf{i}} o'$ and o, o' are in each $\prec''_{\mathbf{i}}$ 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).

$\square$

Theorem of Muller-Satterthwaite (1977) (2)

Proof (cont.)

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

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

Because G **is decisive on all pairs**, q can not be the winner. So either o or p is the winner. $\square$

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 (using our **Claim** on Slide 320).

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.

$\square$

Manipulation

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

Example 4.18 (Plurality Voting)

49 %:	Nader	$\prec$	Gore	$\prec$	Bush
20 %:	Bush	$\prec$	Nader	$\prec$	Gore
20 %:	Nader	$\prec$	Bush	$\prec$	Gore
11 %:	Bush	$\prec$	Gore	$\prec$	Nader

If the last group of voters change their preferences by putting their favorite at the end, then they achieve a better result overall! (Compare with Example 4.8 on Slide 291.)

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 4.19 (Strategy-Proofness)

A social choice function $\mathbf{C}^*$ is called *strategy-proof*, if there is no agent i and profile $\langle \vec{\precsim}_{-i}, \prec_i' \rangle$ such that

$$\mathbf{C}^*(O, \vec{\precsim}) \prec_i \mathbf{C}^*(O, \langle \vec{\precsim}_{-i}, \prec_i' \rangle)$$

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 4.20

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 $\vec{\preceq}$, $\vec{\preceq}'$ s.t.

- $\mathbf{C}^*(O, \vec{\preceq}) = o$, and $\mathbf{C}^*(O, \vec{\preceq}') = o'$, and
- for all $\mathbf{i} \in \mathbf{A}$, for all $p \in O \setminus \{o\}$: $p \prec_{\mathbf{i}} o$ implies $p \prec'_{\mathbf{i}} o$.

We are now modifying $\vec{\preceq}$ as follows. For $\mathbf{i} = \mathbf{1}, \mathbf{2}, \ldots, \mathbf{n}$ we replace successively $\prec_{\mathbf{i}}$ by $\prec'_{\mathbf{i}}$, until the winner of the new action profile under $\mathbf{C}^*$ is no more o but somebody else (which must happen, because at the end it is o'). Let $\mathbf{j}$ be this agent.
Because we can adapt our assumption to this new situation, we assume wlog that $\vec{\preceq}$, $\vec{\preceq}'$ differ only at the entry $\mathbf{j}$. So wlog we can use the notation o' as in our assumption. $\square$

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 4.21 (Gibbard-Satterthwaite (1973/1975))

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

Using Lemma 4.20 this is just a corollary to Theorem 4.15.

Note that we could replace surjectivity by the stronger unanimity.

4.4 Social Choice Correspondences

This section: ballots are strict total orders $\prec$. We also assume $V = O$ in order to simplify notation.

Definition 4.22 (Social choice correspondence)

A social choice correspondence is any function

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

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 4.15 are rare. The most important one is due to **Duggan and Schwartz**: See Theorem 4.28 on Slide 337 .

Definition 4.23 (Positive responsiveness)

A SCC $\mathbf{W}^*$ satisfies

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

$$o' \prec_{\mathbf{i}} o \text{ implies } o' \prec'_{\mathbf{i}} o, \quad \text{and} \quad o' \prec_{\mathbf{i}} o'' \text{ iff } o' \prec'_{\mathbf{i}} o''.$$

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. This is very intuitive in case there are only two candidates!

Anonymity and neutrality

Definition 4.24 (Anonymity and Neutrality)

$\mathbf{W}^*$ satisfies *anonymity* if for all permutations π on $\{1, \ldots, |\mathbf{A}|\}$:

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

$\mathbf{W}^*$ satisfies *neutrality* if for all permutations π on O

$$\pi(\mathbf{W}^*(\vec{\prec})) = \mathbf{W}^*(\pi(\vec{\prec})),$$

where $\pi(\vec{\prec})$ is defined componentwise and $\pi(\prec_i)$ is defined in the obvious way: $o\,\pi(\prec_i)\,o'$ iff: $\pi(o) \prec_i \pi(o')$.

Which properties satisfies the SCC that always declares

- all candidates as winners;
- the **two** top choices of plurality voting as winners?

Only two candidates: $|O| = 2$

Theorem 4.25 (May (1952))

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

Assume there are only 2 candidates and 2 voters, and we are considering a social choice function $\mathbf{C}^*$ (not a correspondence). $\rightsquigarrow$ 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

"$\mathbf{W}^*(\vec{\prec}') = \{o\}$" *by* $o \in \mathbf{W}^*(\vec{\prec}')$

- Does Theorem 4.25 hold for weak monotonicity instead of positive responsiveness?

Optimistic and pessimistic voters

Strategy-proof in Definitions 4.27 and 4.47 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).

$\rightsquigarrow$ We need to rank such sets.

Optimistic and pessimistic voters (2)

Definition 4.26 (Optimist and pessimists)

An agent i is an optimist, if she ranks X higher than Y, whenever $top(\prec_i|_Y) \prec_i top(\prec_i|_X)$.

An agent i is a pessimist, if she ranks X higher than Y, whenever $bot(\prec_i|_Y) \prec_i bot(\prec_i|_X)$.

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 weak orders all maximal elements (and all minimal elements) are indifferent to each other, so both definitions are well-defined (see Slide 304).

Strategy-Proofness for correspondences

Definition 4.27 (Strategy-Proofness)

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

$$\mathbf{W}^*(\langle \vec{\precsim}_{-i}, \prec_i \rangle) \prec_i \mathbf{W}^*(\langle \vec{\precsim}_{-i}, \prec'_i \rangle)$$

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 $\langle \prec_1, \ldots, \prec_{i-1}, \prec_i, \ldots, \prec_A \rangle$.

Note that we consider the relation $\mathbf{W}^*(\langle \vec{\precsim}_{-i}, \prec_i \rangle) \prec_i \mathbf{W}^*(\langle \vec{\precsim}_{-i}, \prec'_i \rangle)$ only for optimistic or pessimistic voters i (see Definition 4.26).

Theorem 4.28 (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 $\vec{\precsim}$ s.t. $\mathbf{W}^*(\vec{\precsim}) = \{o\}$) 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 4.29 (Down-monotonicity)

Suppose we have a profile $\vec{\precsim}$ where $\mathbf{W}^*(\vec{\precsim})$ is a singleton and the following holds: If we modify $\vec{\precsim}$ by letting one agent move a losing alternative down one spot (obtaining profile $\vec{\precsim}'$), then $\mathbf{W}^*(\vec{\precsim}) = \mathbf{W}^*(\vec{\precsim}')$.

Then we call $\mathbf{W}^*$ down-monotone for singleton winners.

Lemma 4.30

A SCC $\mathbf{W}^*$ satisfying strategy-proofness for both optimistic and pessimistic voters also satisfies down-monotonicity for singleton winners.

Definition 4.31 (Dictating Sets $\mathbf{G}$, $p\mathbf{G}o$)

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

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

Lemma 4.32

Suppose $\mathbf{G} \subseteq \mathbf{A}$, $p, o \in O$, and $p\,\mathbf{G}^{\mathbf{W}^*} o$. Let $o \neq o' \neq p$ and $\mathbf{G} = \mathbf{G}_1 \dot{\cup} \mathbf{G}_2$.

Then we have $o'\,\mathbf{G}_1^{\mathbf{W}^*} o$ or $p\,\mathbf{G}_2^{\mathbf{W}^*} o'$.

Lemma 4.33

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

Lemma 4.34

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

- *Let* $\mathbf{G}$ *be a dictating set. If it is the disjoint union of sets* $\mathbf{G}_1$ *and* $\mathbf{G}_2$*, then one of these sets is dictating too.*
- *There is an agent whose maximal element is the unique winner, whenever* $\mathbf{W}^*(\vec{\preceq})$ *is a singleton.*

4.5 Social Welfare Functions

In this section agents ballots are weak orders $\precsim$. We also use $o \prec_i o'$ to denote "$o \precsim o'$ and not $o' \precsim o$": o' is strictly greater than o.

Definition 4.35 (Social welfare function)

A social welfare function is any function

$$\mathbf{f^*} : U \rightarrow L; (\precsim_1, \ldots, \precsim_{|\mathbf{A}|}) \mapsto \precsim^*$$

For each $V \subseteq 2^O \setminus \{\emptyset\}$ the function $\mathbf{f^*}$ w.r.t. U induces a choice correspondence $\mathbf{C}_{\langle \precsim_1, \ldots, \precsim_{|\mathbf{A}|} \rangle}$ as follows:

$$\mathbf{C}_{\langle \precsim_1, \ldots, \precsim_{|\mathbf{A}|} \rangle} =_{def} \begin{cases} V & \longrightarrow & V \\ v & \mapsto & top(\precsim^*|_v) \end{cases}$$

Each tuple $\vec{\precsim}$ determines the election for all possible $v \in V$.

What are desirable properties for f^*?

- **Weak Pareto Efficiency:**
  for all $o, o' \in O$: $(\forall \mathbf{i} \in \mathbf{A} : o \prec_{\mathbf{i}} o')$ implies $o \prec^* o'$.
- **Independence of Irrelevant Alternatives (IIA):**
  for all $o, o' \in O$:
  - $(\forall \mathbf{i} \in \mathbf{A} : \ o \prec_{\mathbf{i}} o' \ \text{ iff } \ o \prec'_{\mathbf{i}} o') \ \Rightarrow \ (o \prec^* o' \ \text{ iff } \ o \prec'^* o')$,
  - $(\forall \mathbf{i} \in \mathbf{A} : \ o \precsim_{\mathbf{i}} o' \ \text{ iff } \ o \precsim'_{\mathbf{i}} o') \ \Rightarrow \ (o \precsim^* o' \ \text{ iff } \ o \precsim'^* o'))$
  - $(\forall \mathbf{i} \in \mathbf{A} : \ o \sim_{\mathbf{i}} o' \ \text{ iff } \ o \sim'_{\mathbf{i}} o') \ \Rightarrow \ (o \sim^* o' \ \text{ iff } \ o \sim'^* o'))$

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

$$(\forall \mathbf{i} \in \mathbf{A} : \ \precsim_{\mathbf{i}}|_v = \precsim'_{\mathbf{i}}|_v)$$
$$\Rightarrow$$
$$\forall o, o' \in v, \ \forall v' \in V \text{ s.t. } v \subseteq v' : \ (o \prec^*|_{v'} o' \ \text{ iff } \ o \prec'^*|_{v'} o')$$

- Shouldn't we also require in the pareto efficiency condition that

$$(\forall i \in A : o \precsim_i o') \text{ implies } o \precsim^* o' ?$$

The answer is *no*, the stated condition, weak pareto efficiency, is perfectly sufficient for proving Arrows theorem. The stronger condition, called pareto efficiency, is not needed. Note that for strict linear orders, there is no such distinction.

Example 4.36 (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."

In this example, an irrelevant alternative (the **Rosé**) **does matter**.

Plurality Vote

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

We consider 7 voters ($\mathbf{A} = \{\mathbf{1}, \mathbf{2}, \ldots, \mathbf{7}\}$) 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).

	$\prec_1$ $(\prec_1)$		$\prec_2$ $(\prec_2)$		$\prec_3$ $(\prec_3)$		$\prec_4$ $(\prec_4)$		$\prec_5$ $(\prec_5)$		$\prec_6$ $(\prec_6)$		$\prec_7$ $(\prec_7)$	
A	1	(2)	1	(2)	1	(1)	1	(1)	2	(2)	2	(2)	2	(2)
B	2	(3)	2	(3)	2	(2)	2	(2)	1	(1)	1	(1)	1	(1)
C	3	(4)	3	(4)	3	(3)	3	(3)	3	(3)	3	(3)	3	(3)
D	4	(1)	4	(1)	4	(4)	4	(4)	4	(4)	4	(4)	4	(4)

Let $\prec^*$ be the solution generated by $\vec{\prec}$ and $\prec^*$ the solution generated by the $\vec{\prec}$. Then we have for $\mathbf{i} = 1, \ldots, 7$: $B \prec_{\mathbf{i}} A$ iff $B \prec_{\mathbf{i}} A$, 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 $\prec_{\mathbf{i}}|_{\{A,B,C\}} = \prec_{\mathbf{i}}|_{\{A,B,C\}}$.
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)**.

Theorem 4.37 (Arrow (1951))

We assume that there are at least two voters and three candidates ($|O| \geq 3$). If the SWF f^* *satisfies* Weak Pareto Efficiency *and* IIA, *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 4.38 (Strict Neutrality)

We assume Pareto Efficiency and IIA and consider two pairs of alternatives a, b and α, β. 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 α, β 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 4.39 (Extremal Lemma)

Let the social welfare function f^* *satisfy* Pareto Efficiency *and* IIA. *Let* $o \in O$ *and suppose each voter* i *puts* o *either on the very top (unique top element wrt.* $\precsim_i$*) or to the very bottom (unique bottom element wrt.* $\precsim_i$*).*
Then o *is* either a unique bottom or a unique top element *of* $\precsim^*$.

Proof (of the lemma).

We assume wlog that (a, b) is distinct from (α, β) and that $b \precsim^* a$ (we have to show the preference is strict).

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

- If $a \neq \alpha$, we change $\precsim_i$ by moving α just strictly above a.
- If $b \neq \beta$, we change $\precsim_i$ by moving β just strictly below b.

This can be done by maintaining the old preferences between α and β (as preferences between a and b are strict).

By pareto efficiency, we have $a \prec'^{*} \alpha$ (for $\alpha \neq a$) and $\beta \prec'^{*} b$ (for $\beta \neq b$).
By IIA, we have $b \precsim'^{*} a$. Using transitivity, we get $\beta \prec'^{*} \alpha$.
By IIA again, we also get $\beta \prec^{*} \alpha$ (because $\alpha \prec'_{i} \beta$ iff $\alpha \prec_{i} \beta$).

We now reverse the roles of (a, b) with (α, β) and apply IIA again to get $b \prec^{*} a$. $\square$

The proof of Arrows theorem is by considering two alternatives a, b and the profile where $a \prec_{\mathbf{i}} b$ for all agents $\mathbf{i}$. 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 $\langle \prec_{\mathbf{1}}^{l}, \ldots, \prec_{|\mathbf{A}|}^{l} \rangle$ (from $l = 0, \ldots, |\mathbf{A}|$, starting with the one described above ($l = 0$)), where in step l, we let all agents numbered $\leq l$ change their profile by **moving a strictly above b** (leaving all other rankings untouched). In fact, only one agent, number $\mathbf{l}$ changes its ranking in step l.

We consider the rankings $\prec^{l*}$ obtained from $\vec{\prec}^{l}$.

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 $\langle \prec_1^l, \ldots, \prec_{|A|}^l \rangle$, **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 α, β and assume wlog $\beta \prec_d^d \alpha$ (otherwise the following argument works as well for $\alpha \prec_d^d \beta$).

$\mathbf{d}$ 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 $1 \leq \mathbf{i} \lneqq d$: we put c on top of each $\prec_i^d$.
- for d: we put c inbetween α and β.
- for $d \lneqq \mathbf{i} \leq |A|$: 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 $\langle \prec''_1, \ldots, \prec''_{|A|} \rangle$ obtained as follows from $\langle \prec^d_1, \ldots, \prec^d_{|A|} \rangle$:

- for $1 \leq i \lneqq d$: we put c to the bottom of each $\prec^d_i$.
- for d: we put c in between α and β.
- for $d \lneqq i \leq |A|$: we put c to the top of each $\prec^d_i$.

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

By transitivity: $\beta \prec^{d^*} \alpha$. $\square$

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 4.40 (Incomplete order)

An incomplete order (IO) $\precsim$ 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!).

- $a \prec_i b$ iff $a \precsim_i b$ and not $b \precsim_i a$ (i strictly prefers a over b)
- $a \sim_i b$ iff $a \precsim_i b$ and $b \precsim_i a$ (i is indifferent between a and b)
- $a \bowtie_i b$ iff not $a \precsim_i b$ and not $b \precsim_i a$ (i considers a and b incomparable)

Which axiomatic property distinguishes $\sim$ from $\bowtie$?

Definition 4.41 (Strict IO)

A strict incomplete order (SIO) is a IO $\precsim$ such that for all distinct alternatives o and o', we have that not $o \sim o'$.

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

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

Definition 4.42 (Restricted IO)

A IO $\precsim$ is called restricted, denoted by rIO, if all top elements or all bottom elements are indifferent. Formally,

- for all $o, o' \in top(\precsim)$, $o \sim o'$, or
- or all $o, o' \in bot(\precsim)$, $o \sim o'$.

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:

$$\mathbf{C}^* : SIO(O) \to O; \vec{\precsim} \mapsto o$$

Social Choice Correspondence:

$$\mathbf{W}^* : SIO(O) \to 2^O; \vec{\precsim} \mapsto v'$$

Social Welfare Function over IO:

$$\mathbf{f}^* : IO(O) \to IO(O); (\precsim_1, \ldots, \precsim_{|\mathbf{A}|}) \mapsto \precsim^*$$

Social Welfare Functions on IO

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

IIA: if for any two profiles $\precsim, \precsim' \in IO(O)$ and all $a, b \in O$, if $\precsim_i$ and $\precsim'_i$ define the same ordering on a and b for each $i \in A$, then $\precsim^*$ and $(\precsim')^*$ define the same ordering on a and b.

Strong dictator: Agent i is a strong dictator, if
$f^*(\precsim) = \precsim_i$, for every IO $\precsim$.

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

Weak dictator: An agent i is a weak dictator, if
$a \prec_i b$ implies $a \prec^* b$ or $a \bowtie^* b$, for every IO $\precsim$.

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 4.43

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 4.44 (Arrow's theorem for IO)

Suppose there are at least two agents and three outcomes. If the SWF $\mathbf{f}^*: IO(O) \rightarrow rIO(O)$ *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 $\{a\} = \mathbf{W}^*(\prec)$.

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

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

- if $a \in \mathbf{W}^*(\prec)$ and for every $b \in O$ and $i \in A$, $(b \prec_i a$ or $b \bowtie_i a)$ implies $(b \prec'_i a$ or $b \bowtie'_i a)$, then $a \in \mathbf{W}^*(\prec')$; and
- if $A = \mathbf{W}^*(\prec)$ and for every $a \in A$, and any other $b \in O$, and $i \in A$, $(b \prec_i a$ implies $b \prec'_i a)$ and $(b \bowtie_i a$ implies $(b \prec'_i a$ or $b \bowtie'_i a))$, then $A = \mathbf{W}^*(\prec')$.

Also for social choice correspondences we distinguish between three types of dictators.

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

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

Weak dictator: An agent i is a weak dictator over SIO, if
$$\mathbf{W}^*(\prec) \cap top(\prec_i) \neq \emptyset \text{ for all profiles } \prec.$$

Again, in the case of (standard) dictators, we have the following positive result:

Proposition 4.45

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 4.46 (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 4.47 (Strategy Proofness)

A SCC $\mathbf{W}^*$ over SIOs is **strategy proof** (over SIO's), if for all agents $\mathbf{i}$ and all $\prec, \prec' \in SIO(O)$ with $\prec_{-\mathbf{i}} = \prec'_{-\mathbf{i}}$ and $\mathbf{W}^*(\prec) \neq \mathbf{W}^*(\prec')$ we have that:

1. for all $a \in \mathbf{W}^*(\prec) \setminus \mathbf{W}^*(\prec')$ and for all $b \in \mathbf{W}^*(\prec) \cap \mathbf{W}^*(\prec')$ we have $b \prec_{\mathbf{i}} a$ or $a \bowtie_{\mathbf{i}} b$,
2. for all $b \in \mathbf{W}^*(\prec') \setminus \mathbf{W}^*(\prec)$ we have that
   1. for all $a \in \mathbf{W}^*(\prec)$, $b \prec_{\mathbf{i}} a$ or $a \bowtie_{\mathbf{i}} b$, and
   2. there is an $a \in \mathbf{W}^*(\prec)$ such that $b \prec_{\mathbf{i}} a$.

Lemma 4.48

If a SCC over SIO is **strategy-proof** *and* **onto** *then it is* unanimous *and* monotonic.

Theorem 4.49 (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 4.48, a SCC which is strategy-proof and onto is also unanimous and monotonic. Thus, by Theorem 4.46 there is a weak dictator. □

Note, over strict total orders, the Theorem is equivalent to Theorem 4.21.

Remarks

- 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 4.50 (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 4.51 (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:

risk-neutral risk-averse risk-seeking	Jap.	= = =	Engl.	= = =	2nd	= < >	1st	= = =	Dutch

Why are second price auctions not so popular among humans?

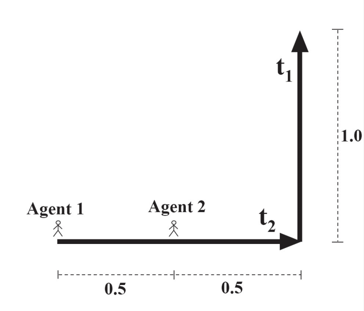
1. **Lying auctioneer.**
2. **When the results are published, subcontractors know the true valuations and what the winner saved. So they might want to share the profit.**

Inefficient Allocation

Auctioning heterogenous, **interdependent** items.

Example 4.52 (Task Allocation)

Two delivery tasks t_1, t_2. Two agents.



Inefficient Allocation (cont.)

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 **cost**$(\{t_1, t_2\})$ − **cost**$(\{t_1\}) = 2.5 - 1.5 = 1)$ she will get it only with a probability of 0.5.

What about full lookahead?

$\rightsquigarrow$ **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 4.53 (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?

$\rightsquigarrow$ **blackboard.**

Answer: Bidder 1 should invest **cost** if and only if

$$v_2 \geq (2\mathbf{cost})^{\frac{1}{2}}.$$

4.8 References

- Geanakoplos, J. (1996, April).
  Three brief proofs of arrow's impossibility theorem.
  Cowles Foundation Discussion Papers 1123R3, Cowles Foundation, Yale University.
  available at http://ideas.repec.org/p/cwl/cwldpp/1123r3.html.

- Sandholm, T. W. (1997).
  Chapter 5: Distributed Rational Decision Making.
  In G. Weiss (Ed.), *Multiagent Systems*. Cambridge, MA: MIT Press.

5. Incomplete Information Games

5 Incomplete Information Games
- Examples and Motivation
- Bayesian Games
- Bayesian Nash equilibrium

Outline

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.

	Buy	**Don't**	**Buy**	**Don't**
Build	$\langle 0, -1\rangle$	$\langle 2, 0\rangle$	$\langle 3, -1\rangle$	$\langle 5, 0\rangle$
Don't	$\langle 2, 1\rangle$	$\langle 3, 0\rangle$	$\langle 2, 1\rangle$	$\langle 3, 0\rangle$
	1's costs are high		1's costs are low	

Agent 1 knows the costs, agent 2 does not.

Agent 1 has a clear dominant strategy: Don't, if costs are high, **Build**, if costs are low.

Example 5.2

Same example as before, but some payoffs are different.

	Buy	**Don't**	**Buy**	**Don't**
Build	$\langle 0, -1\rangle$	$\langle 2, 0\rangle$	$\langle 1.5, -1\rangle$	$\langle 3.5, 0\rangle$
Don't	$\langle 2, 1\rangle$	$\langle 3, 0\rangle$	$\langle 2, 1\rangle$	$\langle 3, 0\rangle$
	1's costs are high		1's costs are low	

Now, **agent 1 has a clear dominant strategy only if costs are high: Don't**. If costs are low, there is no dominant strategy anymore.

Analysis of the game (1)

Player 1: Let x be the probability of player 1 for building (she builds only when her costs are 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 (2)

We compute the expected utilities for both players: how does player 1 reason, how does player 2?

- For player 1: **Build** is strictly better when $y \lneqq \frac{1}{2}$, **Don't** is strictly better when $y \lneqq \frac{1}{2}$. For $y = .5$ the expected utilities are identical. **Player 1 knows the costs, but does not disclose them.**
- For player 2: if costs are high, expected utility is just y. If costs are low, it is $y(1-2x)$. So player 2 is maximizing the value of $yp + y(1-2x)(1-p) = y(1+2x(p-1))$. This value is 0 for $x = \frac{1}{2(1-p)}$, it is $\lneqq 0$ for $x \lneqq \frac{1}{2(1-p)}$ and it is $\gneqq 0$ for $x \gneqq \frac{1}{2(1-p)}$. **Player 2 does not know the costs, therefore probability p comes in.**

Analysis of the game (3)

- $\langle 0, 1\rangle$ is an equilibrium, independently of p.
- $\langle 1, 0\rangle$ is an equilibrium iff $p \leqq 0.5$.
- $\langle \frac{1}{2(1-p)}, \frac{1}{2}\rangle$ is a (mixed) equilibrium iff $p \leqq 0.5$.

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.

Common-prior assumption

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:

	L	R
U	$(1,2)$	$(3,4)$
D	$(5,6)$	$(7,8)$

or

	L	R	S
U	$(1,2)$	$(3,4)$	$(9,10)$
D	$(5,6)$	$(7,8)$	$(11,12)$

- 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:

	L	R	S
U	$(1,2)$	$(3,4)$	$(9,-100)$
D	$(5,6)$	$(7,8)$	$(11,-100)$

Therefore: wlog we assume the same number of agents and the same strategy space for all agents.

Bayesian Game: Informal

A Bayesian game for n agents consists of (1) a set $\mathbf{G}$ 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 $\mathbf{G}$. 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 $G_{i1}, \ldots, G_{is}$ of $\mathbf{G}$ such that: (1) $\bigcup_j G_{ij} = W$ and (2) $G_{ij} \cap G_{ij'} = \emptyset$ for $j \neq j'$.

$I_{\mathbf{i}}$ 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 G_{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 $\langle \mathbf{A}, \mathbf{G}, P, I \rangle$, where

- $\mathbf{A} = \{\mathbf{1}, \ldots \mathbf{i}, \ldots, \mathbf{n}\}$ is a finite set of players.
- $\mathbf{G}$ is a set of n-person normal form games, each of which has the same action profiles $\mathrm{Act} = A_1 \times \ldots \times A_i \times \ldots \times A_n$ where $A_{\mathbf{i}}$ is the set of actions available to player $\mathbf{i}$.
- P is a probability distribution over the set $\mathbf{G}$ of all games (the set of all distributions is denoted by $\Pi(\mathbf{G})$).
- $I = \{I_1, \ldots, I_n\}$ is a set of partitions of $\mathbf{G}$.

	$I_{2,1}$		$I_{2,2}$	
$I_{1,1}$	A $\begin{array}{\|c\|c\|}\hline 1 & 0 \\ \hline 0 & 3 \\ \hline\end{array}$ $p = 0.2$	B $\begin{array}{\|c\|c\|}\hline 2 & 1 \\ \hline 1 & 4 \\ \hline\end{array}$ $p = 0.1$	C $\begin{array}{\|c\|c\|}\hline 1 & 3 \\ \hline 2 & 1 \\ \hline\end{array}$ $p = 0.1$	D $\begin{array}{\|c\|c\|}\hline 1 & 0 \\ \hline 4 & 0 \\ \hline\end{array}$ $p = 0$
$I_{1,2}$	E $\begin{array}{\|c\|c\|}\hline 3 & 2 \\ \hline 0 & 3 \\ \hline\end{array}$ $p = 0.1$	F $\begin{array}{\|c\|c\|}\hline 2 & 4 \\ \hline 1 & 1 \\ \hline\end{array}$ $p = 0.1$	G $\begin{array}{\|c\|c\|}\hline 0 & 1 \\ \hline 3 & 1 \\ \hline\end{array}$ $p = 0.25$	H $\begin{array}{\|c\|c\|}\hline 2 & 0 \\ \hline 3 & 4 \\ \hline\end{array}$ $p = 0.15$

Figure 28: A Bayesian game consisting of zero sum games

5.3 Bayesian Nash equilibrium

Strategies for Bayesian Games

What is a strategy for agent $\mathbf{i}$ 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 $\mathbf{G}$:

$$s_{\mathbf{i}} : I_{\mathbf{i}} \to \Pi(A_{\mathbf{i}}).$$

Definition 5.4 (Bayesian-Nash Equilibrium)

Given a Bayesian game $\langle \mathbf{A}, \mathbf{G}, P, I \rangle$ a strategy profile $s^* = \langle s_1^*, s_2^*, \ldots, s_n^* \rangle$ is a **Bayesian-Nash equilibrium** if for each agent $\mathbf{i}$ the following holds

$$s_{\mathbf{i}}^* \in \mathrm{argmax}_{s_{\mathbf{i}} \in S_{\mathbf{i}}} \sum_{r \in I_{\mathbf{i}}} \sum_{g \in \mathbf{G}} \boldsymbol{\mu}_{g,\mathbf{i}}^{expected}(s_{-\mathbf{i}}^*, s_{\mathbf{i}}) P(g|r)$$

$\boldsymbol{\mu}_{g,\mathbf{i}}^{expected}$ is the expected utility function for agent $\mathbf{i}$ in game g. $P(g|r)$ is the probability for game g under the assumption that the game is in r (**conditional probability**).

Example 5.5 (Bayesian game made of zero-sum games)

	$I_{2,1}$		$I_{2,2}$	
$I_{1,1}$	A $\begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & 3 \\ \hline\end{array}$ $p = 0.2$	B $\begin{array}{|c|c|}\hline 2 & 1 \\ \hline 1 & 4 \\ \hline\end{array}$ $p = 0.1$	C $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 1 \\ \hline\end{array}$ $p = 0.1$	D $\begin{array}{|c|c|}\hline 1 & 0 \\ \hline 4 & 0 \\ \hline\end{array}$ $p = 0$
$I_{1,2}$	E $\begin{array}{|c|c|}\hline 3 & 2 \\ \hline 0 & 3 \\ \hline\end{array}$ $p = 0.1$	F $\begin{array}{|c|c|}\hline 2 & 4 \\ \hline 1 & 1 \\ \hline\end{array}$ $p = 0.1$	G $\begin{array}{|c|c|}\hline 0 & 1 \\ \hline 3 & 1 \\ \hline\end{array}$ $p = 0.25$	H $\begin{array}{|c|c|}\hline 2 & 0 \\ \hline 3 & 4 \\ \hline\end{array}$ $p = 0.15$

What are the equilibria in the game depicted in Example 5.5?

- The players have 4 different strategies: two partition classes and again 2 for the 2×2 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 4×4 normal form game?

	LL	LR	RL	RR
UU	1.3	1.45	1.1	1.25
UD	1.8	1.65	1.8	1.65
DU	1.1	0.7	2.0	1.95
DD	1.5	1.15	2.8	2.35

Figure 29: Associated normal form game

Exercise: Show how to obtain these values.

Reduced game: The unique NE in Figure 29 is $\langle UD, LR \rangle$.

Bayesian game: Therefore the Bayesian-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 30.

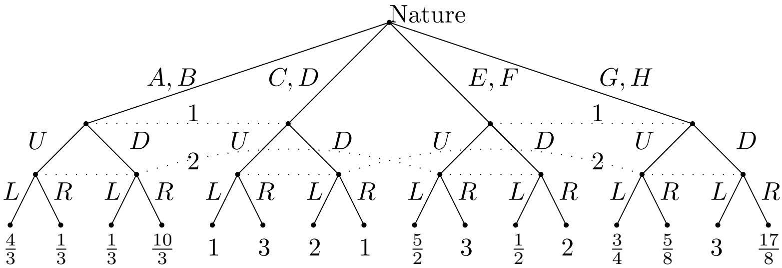


Figure 30: Bayesian Game in Extensive Form with Nature-Agent

Example 5.6 (Another Bayesian Game)

$I_{2,1}$ $I_{2,2}$

$I_{1,1}$

MP

2, 0	0, 2
0, 2	2, 0

$p = 0.3$

PD

2, 2	0, 3
3, 0	1, 1

$p = 0.1$

$I_{1,2}$

Coord

2, 2	0, 0
0, 0	1, 1

$p = 0.2$

BoS

2, 1	0, 0
0, 0	1, 2

$p = 0.4$

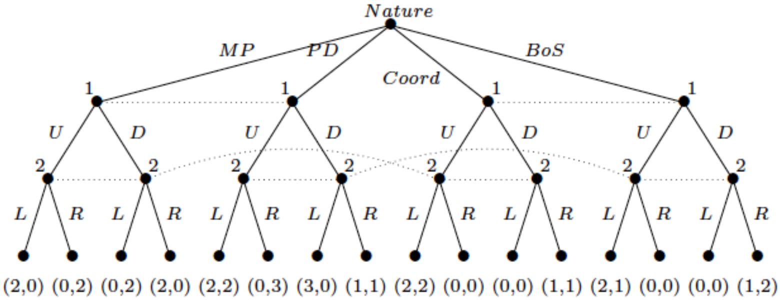


Figure 31: Bayesian Game of Example 5.6 with Nature-Agent

How does the induced NF game look like?
$\rightsquigarrow$ **blackboard.**

Bayesian Game using Types

Definition 5.7 (Bayesian Game using types and utilities)

A Bayesian game is a tuple $\langle A, Act, \Theta, \mathbf{P}, \mu \rangle$, where

- $A = \{1, \ldots, i, \ldots, n\}$ is a finite set of players or agents;
- $Act = A_1 \times \cdots \times A_i \times \cdots \times A_n$ where A_i is the finite set of actions available to player i. $\vec{a} \in Act$ is called action profile. Elements of A_i are called pure strategies;
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$, Θ_i is the type space of agent i;
- $\mathbf{P} : \Theta \mapsto [0, 1]$ is a common prior over types, and
- $\mu = \langle \mu_1, \ldots, \mu_i, \ldots, \mu_n \rangle$ where $\mu_i : Act \times \Theta \longrightarrow \mathbb{R}$ is a real-valued utility (payoff) function for player i.

Comparison of Definitions 5.3 and 5.7

- Definition 5.7 is closer in spirit to the original formulation of a NF game (Definition 1.4 on Slide 23).
- The utilities are explicitly mentioned (and not hidden as in Definition 5.3, namely in the set of games $\mathbb{G}$).
- The information sets are now encoded in the types.
- The type space encodes the information of the agents that is not common knowledge.
- But there is the common-prior on the type space that is common knowledge (as it was for the information sets).
- The set of outcomes is not needed as we define the utility directly on the action profile and the types.

How to formulate Example 5.6 in this framework?
$\rightsquigarrow$ **blackboard.**

Strategies and Equilibria

Pure strategy: $\alpha_{\mathbf{i}} : \Theta_{\mathbf{i}} \mapsto Act_{\mathbf{i}}$. This is similar to imperfect-information extensive form games (mapping from information sets to actions).

Mixed strategy: $s_i \in S_i$, probability distribution over pure strategies. Notation: $s_j(a_j|\theta_j)$ is **the probability**, that agent **j** plays action a_j when using strategy s_j, given that agent **j**'s type is indeed θ_j.

Expected Utility

For Bayesian games, the expected utility can be defined in various ways and leads to (1) **ex post expected utility** (where all agent's types are known to each other), (2) **ex interim expected utility** (where an agent knows only her own types), (3) ex ante expected utility (where an agent does not know anybodys type).

We use the last utility and, based on it, we define the best response set and then the Bayes-Nash equilibrium.

Ex ante expected utility

Definition 5.8 (Ex ante expected utility)

Given a Bayesian game $\langle \mathbf{A}, \mathrm{Act}, \Theta, \mathbf{P}, \boldsymbol{\mu} \rangle$, where the strategies of agents are given by mixed-strategy profiles s, agent $\mathbf{i}$'s expected *ex ante utility* is defined by

$$\boldsymbol{\mu}_{\mathbf{i}}^{\text{ex ante}}(s) = \sum_{\theta \in \Theta} \mathbf{P}(\theta) \sum_{\vec{a} \in \mathrm{Act}} \Big(\prod_{j \in \mathbf{A}} s_j(a_j | \theta_j) \Big) \boldsymbol{\mu}_{\mathbf{i}}(\vec{a}, \theta)$$

Bayes-Nash Equilibrium

Definition 5.9 (Best Response Set)

In a Bayesian game, the set $\mathbf{BR}_i(s_{-i})$ of best responses of agent i to the mixed-strategy profile s_{-i} is the set

$$\mathbf{BR}_i(s_{-i}) = \text{argmax}_{s'_i \in S_i} \mu_i^{\text{ex ante}}(s'_i, s_{-i})$$

Definition 5.10 (Bayesian Nash Equilibrium)

In a Bayesian game, a Bayesian Nash equilibrium is a mixed strategy profile s satisfying for all $i \in A$:

$$s_i \in \mathbf{BR}_i(s_{-i})$$

5.4 References

Fudenberg, D. and J. Tirole (1991).
Game Theory.
MIT Press.

Nash, J. (1950).
Equilibrium points in n-person games.
Proceedings of the National Academy of Sciences of the United States of America 36, 48–49.

Osborne, M. and A. Rubinstein (1994).
A Course in Game Theory.
MIT Press.

6. Mechanism Design

6 Mechanism Design
- Examples and Motivation
- MD for Voting procedures
- Quasilinear preferences
- The Vickrey-Clarke-Groves Mechanism

Outline

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 pure 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 Bayesian 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 tactically? I.e. they know the voting design and do not vote truthfully, but such that their preferred choice is elected after all?
- Is it possible to come up with an implementation of a social choice function (or correspondence) that can not be manipulated?

Example 6.1 (Babysitting, Shoham)

You are babysitting 4 kids (Will, Liam, Vic and Ray). They can chose among (**a**: going to the video **a**rcade, **b**: playing **b**aseball, **c**: going for a leisurely **c**ar ride). The kids give their true preferences as follows:

	Will	**Liam**	**Vic**	**Ray**
1	b	b	a	c
2	a	a	c	a
3	c	c	b	b

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.

Attention: Other kids may have other preferences, so choosing “**a**” is not a dominant strategy for Ray.

How can we avoid such tactical behaviour?

Definition 6.2 (Mechanism)

A **mechanism** wrt. a set of agents A and a set of outcomes O is a pair $\langle Act, M \rangle$ where

1. $Act = A_1 \times \ldots \times A_{|A|}$, where A_i is the set of actions available to agent i.
2. $M : Act \to \Pi(O)$, 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 $\mathrm{M}(\langle a_1, \ldots, a_{|\mathbb{A}|}\rangle)(o) \in \{0, 1\}$ for all action profiles and outcomes o, we can view M as a function ϱ as in Definition 1.4: $\mathrm{M}(\langle a_1, \ldots, a_{|\mathbb{A}|}\rangle)(o) = 1$ becomes $\mathrm{M}(\langle a_1, \ldots, a_{|\mathbb{A}|}\rangle) = o$. Then

Mechanisms and games

If μ is a utility profile and $\langle Act, \mathrm{M}\rangle$ is a mechanism wrt. $\mathbb{A}$ and O, then $\langle \mathbb{A}, Act, O, \mathrm{M}, \mu\rangle$ is a $|\mathbb{A}|$-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.

Utilities and preferences

- Suppose we are given a utility function $\mu_i : O \mapsto \mathbb{R}$.
- Then we can define a weak preference by $o_1 \preceq o_2$ iff $\mu_i(o_1) \leq \mu_i(o_2)$.
- In the following we often use the notation $C^*(\mu)$ for a social choice function C^*, when we mean the induced preference order.

Definition 6.3 (Mechanism/Implementation)

Let $\mathbf{A}$ and O be fixed and a social choice function $\mathbf{C}^*$ be given.
We say that a mechanism $\langle Act, M\rangle$ **implements** the function $\mathbf{C}^*$ **in dominant strategies**, if for all utility profiles $\boldsymbol{\mu}$ the game $\langle \mathbf{A}, Act, O, M, \boldsymbol{\mu}\rangle$ has an equilibrium in dominant strategies and for any such equilibrium $\langle s_1^*, s_2^*, \ldots, s_n^*\rangle$ we have:

$$M(\langle s_1^*, s_2^*, \ldots, s_n^*\rangle) = \mathbf{C}^*(\boldsymbol{\mu})$$

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?

This definition builds on the first definition of a Bayesian game (Definition 5.3 on Slide 403).

Definition 6.4 (Bayes-Nash Implementation)

Let $\mathbf{A}$ and O be fixed. Let U be the set of all utility profiles $\boldsymbol{\mu}$ over O. Let G be the set of games $\langle \mathbf{A}, \mathrm{Act}, O, \mathrm{M}, \boldsymbol{\mu} \rangle$ for $\boldsymbol{\mu} \in U$. Let $\mathbf{P}$ 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 $\mathbf{C}^*$ be given.

We say that a mechanism $\langle \mathrm{Act}, \mathrm{M} \rangle$ **implements** the function **$\mathbf{C}^*$ in Bayes-Nash equilibria wrt. P and I**, if there exists a Bayes-Nash equilibrium of the Bayesian game $\langle \mathbf{A}, \mathbf{G}, \mathbf{P}, I \rangle$, such that for each game $g \in \mathbf{G}$ and each action profile $\langle a_1, \ldots, a_n \rangle \in \mathrm{Act}$ that can arise in g, it holds that $M(\langle a_1, \ldots, a_n \rangle) = \mathbf{C}^*(\boldsymbol{\mu})$.

For a Bayesian game defined using types, a slightly modified notation is used, but it is nevertheless equivalent: **everything can be defined using all three definitions of Bayesian games**.

We define a *Bayesian game setting* as a tuple $\langle \mathbf{A}, O, \Theta, \mathrm{P}, \boldsymbol{\mu} \rangle$: i.e. like a Bayesian game in the sense of Definition 5.7 on Slide 416 but without the set of actions and instead a set of outcomes O. This is because the actions are used in the mechanism: each action profile is mapped to a probability distribution on the outcomes.

Definition 6.5 (Mechanism for Bayesian game setting)

A mechanism for a Bayesian game setting $\langle \mathbb{A}, O, \Theta, \mathbf{P}, \mu \rangle$ is a pair $\langle \mathrm{Act}, \mathrm{M} \rangle$ where

1. $\mathrm{Act} = A_1 \times \ldots \times A_{|\mathbb{A}|}$, where A_i is the set of actions available to agent i.
2. $\mathrm{M} : \mathrm{Act} \to \Pi(O)$, where $\Pi(O)$ is the set of all probability distributions over the set of outcomes.

A mechanism is deterministic, if for every action profile $\vec{a}$ there is a $o \in O$ with $\mathrm{M}(\vec{a})(o) = 1$.

Definitions for Baysian games with types

Given a Bayesian game setting $\langle \mathbf{A}, O, \Theta, \mathbf{P}, \boldsymbol{\mu} \rangle$ and a social choice function $\mathbf{C}^*$ over $\mathbf{A}, O$, we say that a mechanism $\langle \mathrm{Act}, \mathrm{M} \rangle$

implements $\mathbf{C}^*$ in dominant strategies, if for any profile $\vec{\boldsymbol{\mu}}$ the game has an equilibrium in dominant strategies and in any such equilibrium $\vec{a^*}$ we have: $\mathrm{M}(\vec{a^*}) = \mathbf{C}^*(\vec{\boldsymbol{\mu}})$.

implements $\mathbf{C}^*$ in Bayes-Nash equilibrium, if there exists Bayesian-Nash equilibrium of the game (of incomplete information) $\langle \mathbf{A}, \mathrm{Act}, \Theta, \mathbf{P}, \boldsymbol{\mu} \rangle$ such that for each $\theta \in \Theta$ and each profile $\vec{a} \in \mathrm{Act}$ that can arise given type θ in this equilibrium: $\mathrm{M}(\vec{a}) = \mathbf{C}^*(\vec{\boldsymbol{\mu}}(\cdot, \theta))$.

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 6.6 (Direct Mechanism)

A direct mechanism wrt. a set of agents $\mathbb{A}$ and a set of outcomes O is a mechanism $\langle Act, M\rangle$ 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 6.7 (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. □

So which social choice functions **can be implemented in dominant strategies**?

Remember Theorem 4.21. Strategy-proofness is exactly **dominant-strategy truthful** (and we can restrict to truthful mechanisms because of the revelation principle), therefore we have to relax some conditions in the theorem.

6.3 Quasilinear preferences

Ways out of Gibbard-Satterthwaite

We relax the conditions in three ways

- agents can no more express **any** preferences,
- we do not assume **ontoness** any more, and
- we look at implementation in Bayesian-Nash equilibrium.

In fact, these relaxations enable us to deal with the class of **Groves-mechanisms**, in particular with the Vickrey-Clarke-Groves (VCG) mechanism.

Quasilinearity

Remember Slide 434, where utilities induce preferences.

Definition 6.8 (Quasilinear Utility Functions)

Agents have *quasilinear* preferences (or utility functions), when the following holds:

Outcome: there is a finite set X such that $O = X \times \mathbb{R}^n$,

Utility: $\mu_i(\langle x, \vec{p} \rangle, \theta) = u_i(x, \theta) - f_i(p_i)$, where u_i is any function from $X \mapsto \mathbb{R}$ and f_i is any strictly monotonically increasing function from $\mathbb{R} \mapsto \mathbb{R}$.

X can be seen as nonmonetary outcomes (allocation of an object, selection of a candidate), $\vec{p}$ represents the payments of each agent i to the mechanism (e.g.auctioneer).

Risk Attitudes

What are restrictions we should impose on the functions f_i?

- They could be linear, which corresponds to risk-neutral.
- They could be sublinear, which corresponds to risk-averse.
- They could be superlinear, which corresponds to risk-seeking.

Direct Quasilinear Mechanism

Definition 6.9 (Quasilinear mechanism)

A **mechanism** in a quasilinear setting for a Bayesian game setting $\langle \mathbf{A}, X \times \mathbb{R}^n, \Theta, \mathbf{P}, \boldsymbol{\mu} \rangle$ is a triple $\langle Act, \chi, \wp \rangle$ where

1. $Act = A_1 \times \ldots \times A_{|\mathbf{A}|}$, where $A_{\mathbf{i}}$ is the set of actions available to agent $\mathbf{i}$.
2. $\chi : Act \mapsto \Pi(X)$, where $\Pi(X)$ is the set of all probability distributions over choices,
3. $\wp : Act \mapsto \mathbb{R}^n$ is a mapping to payments for the agents.

A **direct** quasilinear mechanism is a pair $\langle \chi, \wp \rangle$, where for each $\mathbf{i} : A_i = \Theta_{\mathbf{i}}$, i.e. each agent reveals his own type.

χ is also called **choice rule** and $\wp$ is called **payment rule**.

Valuations

We assume that for all agents **i**, for all outcomes $o \in O$ and for all pairs of joint types $\theta, \theta' \in \Theta$ for which $\theta_i = \theta'_i$, it holds that $\mu_i(o, \theta) = \mu_i(o, \theta')$. This condition is called conditional utility independance. It ensures that the utilities of agents only depend on their own types.

For $x \in X$ the value $v_i(x) := \mu_i(x, \theta_i)$ is called **i's valuation of** x. It is the maximal amount agent **i** is willing to pay to make sure the mechanism designer implements the choice x. By $\hat{v}_i$ we denote agent **i**'s declaration of its valuation to a direct mechanism.

v and $\hat{v}$ denote the profiles for all agents. $\hat{v}_{-i}$ is the vector of all agents other than **i**.

Truthfulness and Efficiency

Definition 6.10 (Truthfulness)

A quasilinear mechanism ist truthful, if it is direct and for all agents i and all v_i: the equilibrium strategy of agent i is the strategy $\hat{v}_i = v_i$.

Definition 6.11 (Efficiency)

A quasilinear mechanism is efficient, if in equilibrium a choice x is selected such that for all v and all x':

$$\sum_i v_i(x) \geq \sum_i v_i(x').$$

A choice is selected by the mechanism, that maximizes the sum of the utilities of all agents.

6.4 The Vickrey-Clarke-Groves Mechanism

Definition 6.12 (Groves Mechanisms)

A *Groves mechanism* is a direct quasilinear mechanism $\langle \chi, \wp \rangle$, for which

$$\chi(\hat{v}) = \text{argmax}_x \sum_{\mathbf{i}} \hat{v}_{\mathbf{i}}(x)$$

$$\wp_{\mathbf{i}}(\hat{v}) = h_{\mathbf{i}}(\hat{v}_{-\mathbf{i}}) - \sum_{\mathbf{j} \neq \mathbf{i}} \hat{v}_{\mathbf{j}}(\chi(\hat{v}))$$

Groves mechanisms are dominant-strategy truthful implementations of a social welfare maximizing social choice function.

VCG Mechanism

In Groves mechanisms, the functions h_i are arbitrary. However, particular choices lead to particular special mechansims with interesting properties.

Definition 6.13 (Clarke Tax)

The payment functions are set to $h_i(\hat{v}_{-i}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i}))$. Note this does not depend on the agent's own declaration $\hat{v}_i$

With this setting, the resulting particular Groves mechanism is called Vickrey-Clarke-Groves mechanism.

Vickrey-Clarke-Groves

- The Clarke tax charges agent i the sum of all other agent's utilities **as if i had not participated**.
- The second sum pays each agent the sum of every other agents utility for the mechanisms choice.

Transportation network with selfish agents: $\rightsquigarrow$ **blackboard**

7. From Classical to Temporal Logics

7 From Classical to Temporal Logics
- Sentential Logic (**SL**)
- First-Order Logic (**FOL**)
- How to deal with time?
- 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 **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 **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 $\mathcal{L}_{\mathsf{SL}}$, Lang. $\mathcal{L} \subseteq \mathcal{L}_{\mathsf{SL}}$)

The language $\mathcal{L}_{\mathsf{SL}}$ of propositional (or sentential) logic consists of

- $p, q, r, x_1, x_2, \ldots x_n, \ldots$: a countable set $\mathcal{AT}$ of **SL**-constants,
- $\neg$, $\vee$: the sentential connective ($\neg$ is unary, $\vee$ is binary),
- $($, $)$: the parentheses to help readability.

In most cases we consider only a finite set of **SL**-constants. They define a language $\mathcal{L} \subseteq \mathcal{L}_{\mathsf{SL}}$. The set of $\mathcal{L}$-formulae $Fml_{\mathcal{L}}$ is defined inductively.

Macros

$$\begin{aligned} \top &:= \mathsf{p} \vee \neg \mathsf{p} \\ \bot &:= \neg \top \end{aligned}$$

$$\begin{aligned} \varphi \wedge \psi &:= \neg(\neg\varphi \vee \neg\psi) \\ \varphi \rightarrow \psi &:= \neg\varphi \vee \psi \\ \varphi \leftrightarrow \psi &:= (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi) \end{aligned}$$

Definition 7.2 (Semantics, Valuation, Model)

A valuation v for a language $\mathcal{L} \subseteq \mathcal{L}_{\mathsf{SL}}$ is a mapping from the set of **SL**-constants defined by $\mathcal{L}$ into the set $\{\text{true}, \text{false}\}$. Each valuation v can be uniquely extended to a function $\bar{v} : Fml_{\mathcal{L}} \to \{\text{true}, \text{false}\}$ so that:

- $\bar{v}(\neg p) = \begin{cases} \text{true}, & \text{if } \bar{v}(p) = \text{false}, \\ \text{false}, & \text{if } \bar{v}(p) = \text{true}. \end{cases}$
- $\bar{v}(\varphi \vee \gamma) = \begin{cases} \text{true}, & \text{if } \bar{v}(\varphi) = \mathbf{t} \text{ or } \bar{v}(\gamma) = \text{true}, \\ \text{false}, & \text{else} \end{cases}$

Definition (continued)

Thus each valuation v uniquely defines a $\bar{v}$. We call $\bar{v}$ $\mathcal{L}$-structure.
A structure determines for each formula if it is true or false.
If a formula ϕ is true in structure $\bar{v}$ we also say $\mathcal{A}_v$ is a model of ϕ. From now on we will speak of models, structures and valuations synonymously.

Semantics

The process of mapping a set of $\mathcal{L}$-formulae into $\{\text{true}, \text{false}\}$ is called semantics.

Definition 7.3 (Model, Theory, Tautology (Valid))

1. A formula $\varphi \in Fml_{\mathcal{L}}$ **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** φ.
2. A **theory** is a set of formulae: $T \subseteq Fml_{\mathcal{L}}$. v satisfies T if $\bar{v}(\varphi) = \text{true}$ for all $\varphi \in T$. We write $v \models T$.
3. A $\mathcal{L}$-formula φ is called **$\mathcal{L}$-tautology** (or simply called **valid**) if **for all possible valuations v in $\mathcal{L}$ $v \models \varphi$ holds**.

From now on we suppress the language $\mathcal{L}$ when obvious from context.

Truth Tables

Truth tables are a conceptually simple way of working with PL (invented by Wittgenstein in 1918).

p	q	$\neg p$	$p \vee q$	$p \wedge q$	$p \rightarrow q$	$p \leftrightarrow q$
t	**t**	**f**	**t**	**t**	**t**	**t**
f	**t**	**t**	**t**	**f**	**t**	**f**
t	**f**	**f**	**t**	**f**	**f**	**f**
f	**f**	**t**	**f**	**f**	**t**	**t**

Fundamental Semantical Concepts

- If it is possible to find some valuation v that makes φ true, then we say φ is satisfiable.
- If $v \models \varphi$ for all valuations v then we say that φ is valid and write $\models \varphi$. φ is also called tautology.
- A theory is a set of formulae: $\Phi \subseteq \mathcal{L}_{PL}$.
- A theory Φ is called consistent if there is a valuation v with $v \models \Phi$.
- A theory Φ is called complete if for each formula φ in the language, $\varphi \in \Phi$ or $\neg\varphi \in \Phi$.

Two simple examples

Consider the two formulae p ∧ ¬b and a ∨ ¬a.

- Are they satisfiable or valid?
- Are they both consistent? What if we add b?

Consequences

Given a theory Φ we are interested in the following question: **Which facts can be derived from Φ?** We can distinguish two approaches:

1. semantical consequences, and
2. syntactical inference.

- Let Φ be a theory and φ be a formula. We say that φ is a semantical consequence of Φ if for all valuations v:

$$v \models \Phi \text{ implies } v \models \varphi.$$

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 $\mathcal{V}ar$. Typically, we denote variables by $x_0, x_1, \ldots$.

Example 7.5

$$\varphi := \exists x_0 \forall x_1 (P_0^2(f_0^1(x_0), x_1) \wedge P_2^1(f_1^0))$$

Functions

Definition 7.6 (Function symbols)

Let $k \in \mathbb{N}_0$. The set of k-ary function symbols is denoted by $\mathcal{F}unc^k$. Elements of $\mathcal{F}unc^k$ 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

$$\mathcal{F}unc := \bigcup_k \mathcal{F}unc^k$$

A 0-ary function symbol is called constant.

Predicates

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 $P_1^k, P_2^k \ldots$. 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 $\mathcal{L}_{FOL}$ 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 $\mathcal{F}unc$ and $\mathcal{V}ar$ is inductively defined as follows:

1. Each variable from $\mathcal{V}ar$ is a term.
2. If $t_1, \dots t_k$ are terms then $f^k(t_1, \dots, t_k)$ is a term as well, where f^k is an k-ary function symbol from $\mathcal{F}unc^k$.

Definition 7.9 (Language)

The first-order language with equality $\mathcal{L}_{FOL}(\mathcal{V}ar, \mathcal{F}unc, \mathcal{P}red)$ is defined by the following grammar:

$$\varphi ::= P^k(t_1, \dots, t_k) \mid \neg\varphi \mid \varphi \lor \varphi \mid \exists x(\varphi) \mid t \doteq r$$

where $P^k \in \mathcal{P}red^k$ is a k-ary predicate symbol and $t_1, \dots, t_k$ and t, r are terms over $\mathcal{V}ar$ and $\mathcal{F}unc$.

Definition 7.10 (Macros)

We define the following syntactic constructs as macros ($P \in \mathcal{P}red^0$):

$$
\begin{array}{rcl}
\bot & := & P \wedge \neg P \\
\top & := & \neg \bot \\
\varphi \wedge \psi & := & \neg(\neg\varphi \vee \neg\psi) \\
\varphi \rightarrow \psi & := & \neg\varphi \vee \psi \\
\varphi \leftrightarrow \psi & := & (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi) \\
\forall x(\varphi) & := & \neg\exists x(\neg\varphi)
\end{array}
$$

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 $u, v, w, \dots$
- Function symbols are also denoted by $f, g, h, \dots$
- Constants are also denoted by $a, b, c, \dots, c_0, c_1, \dots$
- Predicate symbols are also denoted by $P, Q, R, \dots$
- 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**.

Semantics

Definition 7.11 (Model, Structure)

A model or structure for **FOL** over $\mathcal{V}ar$, $\mathcal{F}unc$ and $\mathcal{P}red$ is given by $\mathfrak{M} = (U, I)$ 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 $x \in \mathcal{V}ar$ an element $I(x) \in U$.

We write:

1. $\mathfrak{M}(P^k)$ for $I(P^k)$,
2. $\mathfrak{M}(f^k)$ for $I(f^k)$, and
3. $\mathfrak{M}(x)$ 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

$\varphi := Q(x) \vee \forall z(P(x, g(z))) \vee \exists x(\forall y(P(f(x), y) \wedge Q(a)))$

- $U = \mathbb{R}$
- $I(a) : \{\emptyset\} \to \mathbb{R}$, $\emptyset \mapsto \pi$ constant functions,
- $I(f) : I(f) = \sin : \mathbb{R} \to \mathbb{R}$ and $I(g) = \cos : \mathbb{R} \to \mathbb{R}$,
- $I(P) = \{(r, s) \in \mathbb{R}^2 : r \leq s\}$ 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 $\mathfrak{M} = (U, I)$ be a model. We define inductively the **value of t wrt $\mathfrak{M}$**, written as $\mathfrak{M}(t)$, as follows:

$\mathfrak{M}(x) := I(x)$ for a variable $t = x$,

$\mathfrak{M}(t) := I(f^k)(\mathfrak{M}(t_1), \ldots, \mathfrak{M}(t_k))$ if $t = f^k(t_1, \ldots, t_k)$.

Definition 7.14 (Semantics)

Let $\mathfrak{M} = (U, I)$ be a model and $\varphi \in \mathcal{L}_{FOL}$. φ is said to be **true in $\mathfrak{M}$**, written as $\mathfrak{M} \models \varphi$, if the following holds:

$\mathfrak{M} \models P^k(t_1, \ldots t_k)$ iff $(\mathfrak{M}(t_1), \ldots, \mathfrak{M}(t_k)) \in \mathfrak{M}(P^k)$

$\mathfrak{M} \models \neg\varphi$ iff not $\mathfrak{M} \models \varphi$

$\mathfrak{M} \models \varphi \vee \psi$ iff $\mathfrak{M} \models \varphi$ or $\mathfrak{M} \models \psi$

$\mathfrak{M} \models \exists x(\varphi)$ iff $\mathfrak{M}^{[x/a]} \models \varphi$ for some $a \in U$ where $\mathfrak{M}^{[x/a]}$ denotes the model equal to $\mathfrak{M}$ but $\mathfrak{M}^{[x/a]}(x) = a$.

$\mathfrak{M} \models t \doteq r$ iff $\mathfrak{M}(t) = \mathfrak{M}(r)$

Given a set $\Sigma \subseteq \mathcal{L}_{FOL}$ we write $\mathfrak{M} \models \Sigma$ iff $\mathfrak{M} \models \varphi$ for all $\varphi \in \Sigma$.

Example: FO($\leq$)

Monadic first-order logic of order, denoted by **FO($\leq$)**, is first-order logic with the only binary symbol $\leq$ (except equality, which is also allowed) and, additionally, any number of **unary predicates**. The theory assumes that $\leq$ 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($\leq$)?

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 How to deal with time?

LT properties of transition systems

Transition systems are abstractions from systems in the real world. We would like to

- reason about particular computations of a *TS*;
- reason about all possible executions in a *TS*,
- formulate interesting properties such a *TS* should satisfy.

Main notion: a path in a *TS*. It is a sequence of states starting in an initial state. It corresponds to an execution of the *TS*.

Definition 7.15 (Transition system)

A **transition system** *TS* is a tuple $\langle \mathrm{S}, \mathrm{Act}, \longrightarrow, I, \mathcal{P}rop, \pi \rangle$ where

- S is a set of **states**, denoted by $s_1, s_2, \ldots$,
- Act is a set of **actions**, denoted by $\alpha, \beta, \gamma, \ldots$,
- $\longrightarrow \subseteq \mathrm{S} \times \mathrm{Act} \times \mathrm{S}$ is a **transition relation**,
- $I \subseteq \mathrm{S}$ is the set of **initial states**,
- $\mathcal{P}rop$ is a set of **atomic propositions**, and
- $\pi : \mathrm{S} \to 2^{\mathcal{P}rop}$ is a **labelling (or valuation)**.

A *TS* is **finite** *if, by definition,* S, Act and $\mathcal{P}rop$ are all finite.

This is in accordance with $\mathcal{L}_{SL}$. States are what we called **models**. A labelling corresponds to **a set of valuations**. The new feature is that we can **move between states by means of actions**: instead of $\langle s, \alpha, s' \rangle$ we write $s \xrightarrow{\alpha} s'$.

Transitions systems without terminal nodes

From now on, we assume wlog that all transition systems do not have any terminal nodes. In case a *TS* has terminal nodes, we could simply add for each such node a new action and a new state (with an arrow pointing to itself) and extend the *TS* appropriately.

Definition 7.16 (Path λ (of a *TS*))

A path $\lambda : \mathbb{N}_0 \rightarrow \mathrm{S}$ in a *TS* is a sequence of states starting with an initial state such that this sequence corresponds to a **run of the *TS***.

Paths versus traces

Often, we are not so much interested in the states as such, only **what is true** in them, i.e. which propositions from $\mathcal{P}rop$ are true: $\pi(s_i)$.

Definition 7.17 (Trace of λ of a *TS*)

The **trace** of a path $\lambda : \mathbb{N}_0 \to \mathrm{S}$ in a *TS* with $\mathcal{P}rop$ is the following infinite sequence

$$\pi(\lambda(0))\pi(\lambda(1))\pi(\lambda(2))\ldots\pi(\lambda(i))\ldots$$

($\pi(\lambda(i))$ is the set of all propositions that are true in state $\lambda(i)$).
We call this also an **ω-word** over $2^{\mathcal{P}rop}$.

The set of all traces of a *TS* is the set of the traces of all paths from *TS*: it is denoted by *Traces*(*TS*).

Runs or paths?

We want to define what it means that
two transition systems behave the same.

1. We take the set of **runs** of a *TS* as the defining behaviour.
2. Often the actions do not play any role, only the states do. Then we could take the set of **paths** of a *TS* as the defining behaviour.

Both possibilities rely on the **internal** behaviour, that we might not be able to determine.

Runs or paths? (cont.)

Often one cannot distinguish between certain states, we only know what is true in them (and that **depends on the language** $\mathcal{P}rop$).

- Therefore we choose from now on the observable behaviour to describe a *TS*: the set of traces as the defining behaviour of a TS.

Die Grenzen meiner Sprache bedeuten die Grenzen meiner Welt.

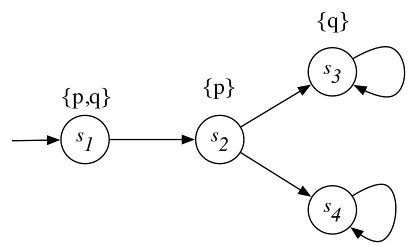
L. Wittgenstein, TLP, Satz 5.6

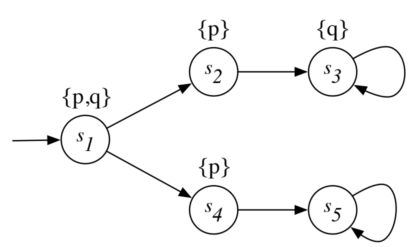
Definition 7.18 (Equivalence of transition systems)

Let TS_1 and TS_2 be two transition systems over $\mathcal{P}rop$ and let $A \subseteq \mathcal{P}rop$. We say that

1. TS_1 and TS_2 are A-equivalent, *if, by definition,* $\mathrm{Traces}(TS_1)|_A = \mathrm{Traces}(TS_2)|_A$,
2. TS_1 is a correct implementation of TS_2, *if, by definition,* $\mathrm{Traces}(TS_1) \subseteq \mathrm{Traces}(TS_2)$.

Example 7.19 (Two transition systems TS_1, TS_2)





What are the paths, traces of them?

1. $s_1 s_2 s_3^\omega$, $s_1 s_2 s_4^\omega$. $\{p,q\}\{p\}\{q\}^\omega$, $\{p,q\}\{p\}\{\}^\omega$
2. $s_1 s_2 s_3^\omega$, $s_1 s_4 s_5^\omega$. $\{p,q\}\{p\}\{q\}^\omega$, $\{p,q\}\{p\}\{\}^\omega$.
3. Both transition systems have the same set of traces. **Is there any difference between them?**

Properties of transition systems

- "Whenever p holds, a state with q is reachable."
- Obviously this is true in TS_1 but not in TS_2.
- Later: this cannot be expressed in LTL.
- Any property that is solely based on the set of traces of a TS can not distinguish between TS_1 and TS_2.
- But still many useful properties can be defined: linear time (LT) properties.
- The former two transition systems **cannot be distinguished** by any LT-property.

Properties of transition systems

Definition 7.20 (LT properties)

Given a set $\mathcal{P}rop$, a **LT property** over $\mathcal{P}rop$ is any subset of $(2^{\mathcal{P}rop})^\omega$.
A transition system *TS* **satisfies** such a property, if all its traces are contained in it.

- A LT property is a, possibly infinite, set of infinite words.

Attention

$\{p,q\}\{p\}\{q\}^\omega$ really means the infinite word $\{p,q\}\{p\}\{q\}\ldots\{q\}\ldots$, not $\{p\}\{p\}\{q\}\ldots\{q\}\ldots$, or $\{q\}\{p\}\{q\}\ldots\{q\}\ldots$ as in regular expressions.

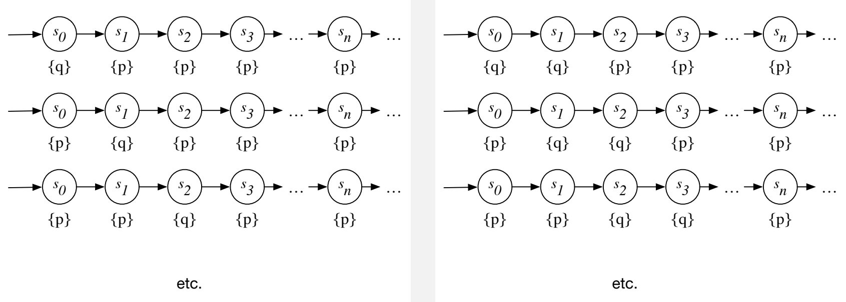
Properties of transition systems

We also use $\{\{p,q\},\{p\}\}\{p\}\{q\}^\omega$ to denote the set of words that either start with $\{p,q\}$ or with $\{p\}$.

Example 7.21

1. When p then q two steps farther ahead.
2. It is never the case that p and q. This is important for mutual exclusion algorithms: one wants to ensure that never two processes are at the same time in their critical section.
3. When p then eventually q. When a message has been sent, it will eventually be received.

Example 7.22 (Sets of paths)



The first infinite set of paths M_1 can be denoted more succinctly by $\{p\}^*\{q\}\{p\}^\omega$ and the second, M_2, by $\{p\}^*\{q\}\{q\}\{p\}^\omega$ (we are using the Kleene * as in regular expressions). These expressions are called ω **regular expressions**.

Which properties does M_1 satisfy?

- At some time q is true and then always p holds.
- q is true exactly once.
- p is always true with one single exception, when q is true.
- q **is true exactly once (and in that state** p **is not true), and before and after that, only** p **holds true.** This will be expressed as a LTL formula shortly.

Can M_1 be represented as a (finite) transition system at all?

Theorem 7.23 (Traces and LT-properties)

Let TS_1 and TS_2 be two transition systems over $\mathcal{P}rop$. Then the following are equivalent:

- $Traces(TS_1) \subseteq Traces(TS_2)$
- *for all LT properties M: if TS_2 satisfies M, so does TS_1.*

Corollary 7.24

TS_1 and TS_2 satisfy the same LT properties if and only if $Traces(TS_1) = Traces(TS_2)$.

More interesting properties

For the formal specification and verification of concurrent and distributed systems, the following **useful concepts can be formally, and concisely, specified** as LT properties (and later also using temporal logics):

- safety properties,
- liveness properties,
- fairness properties.

Safety Properties

Many safety properties are quite simple, they are just conditions on the states and called invariants. They have the form ($A_i \subset 2^{\mathcal{P}rop}$)

$$M = \{A_0 A_1 \dots A_i \dots : \text{ for all } i\text{: } A_i \models \varphi\}$$

for a **propositional formula** φ.

Examples are

- **mutual exclusion properties**: $\neg \text{crit}_1 \lor \neg \text{crit}_2$,
- **deadlock freedom**: $\neg \text{wait}_0 \lor \dots \lor \neg \text{wait}_5$. Deadlock freedom does not imply a fair distribution, i.e. $\neg \text{wait}_0$ can always hold.

Others require **conditions on finite fragments**, for example a traffic light with three phases requiring an **orange phase immediately before a red phase**. This is not an invariant.

Safety Properties (cont.)

Definition 7.25 (Safety property)

A LT property M_{safe} is called a **safety property**, if for all words $\lambda \in (2^{\mathcal{P}rop})^\omega \setminus M_{\text{safe}}$ there exists a finite prefix (a **bad prefix**) $\hat{\lambda}$ of λ such that

$$M_{\text{safe}} \cap \{\lambda' : \lambda' \in (2^{\mathcal{P}rop})^\omega, \hat{\lambda} \text{ is a finite prefix of } \lambda'\} =$$

So it is a **condition on a finite initial fragment**: no extended word resulting from such a bad prefix is allowed.

Liveness Properties

Definition 7.26 (Liveness property)

A LT property M is a liveness property, if the set of finite prefixes of the elements of M is identical to $(2^{\mathcal{P}rop})^*$. I.e. each finite prefix can be extended to an infinite word that satisfies the property.

- Each process will **eventually** enter its critical section.
- Each process will enter its critical section **infinitely** often.
- Each waiting process will **eventually** enter its critical section.

Liveness Properties (cont.)

Starvation freedom in the dining philosophers is a typical example: each philosopher is getting her sticks *infinitely often*.

Starvation freedom: For all timepoints i, if there is a waiting process at time i, then the process gets into its critical section *eventually*.

Safety versus Liveness

- Are safety properties also liveness properties? Vice versa?
- There is only one property that is both: $(2^{\mathcal{P}rop})^{\omega}$, i.e. the trivial property that contains all paths.

Theorem 7.27 (LT properties as intersections)

Each LT-property can be represented as the intersection of a safety with a liveness property.

But there are LT properties that are neither safe nor live.

Fairness Properties

Definition 7.28 (Fairness property)

A LT property is a *fairness property*, if one of the following applies:

Each process gets its turn infinitely often

provided that

unconditional: (no restrictions)

strong: it is enabled infinitely often,

weak: it is continuously enabled from a certain time on.

LT properties used in practice

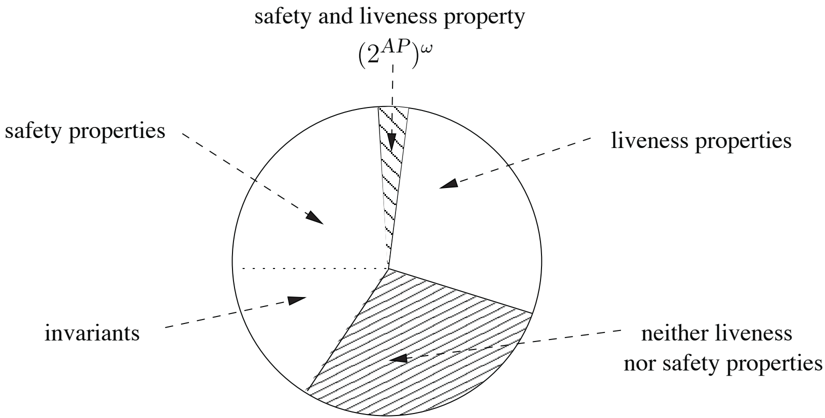


Figure 32: Overview LT-properties

7.4 Linear Time Logic (**LTL**)

Typical temporal operators

$\mathbf{X}\varphi$	φ is true in the neXt moment in time
$\mathbf{G}\varphi$	φ is true Globally: in all future moments
$\mathbf{F}\varphi$	φ is true Finally: eventually (in the future)
$\varphi\,\mathcal{U}\,\psi$	φ is true Until at least the moment when ψ becomes true (and this eventually happens)

$$\mathbf{G}((\neg \mathsf{passport} \vee \neg \mathsf{ticket}) \rightarrow \mathbf{X}\neg \mathsf{board\_flight})$$

$$\mathsf{send}(\mathsf{msg}, \mathsf{rcvr}) \rightarrow \mathbf{F}\mathsf{receive}(\mathsf{msg}, \mathsf{rcvr})$$

Safety: **Something bad will not happen, something good will always hold.**

$\mathbf{G}\neg\mathsf{bankrupt}$,

$\mathbf{G}\mathsf{fuelOK}$,

Usually: $\mathbf{G}\neg \ldots$.

Liveness: **Something good will happen.**

$\mathbf{F}\mathsf{rich}$,

$\mathsf{power\_on} \rightarrow \mathbf{F}\mathsf{online}$,

Usually: $\mathbf{F} \ldots$.

Fairness: Combinations of safety and liveness:

$\mathbf{FG}\neg$dead or

$\mathbf{G}(\text{request\_taxi} \rightarrow \mathbf{F}\text{arrive\_taxi})$.

Strong fairness: "If something is requested then it will be allocated":

$$\mathbf{G}(\text{attempt} \rightarrow \mathbf{F}\text{success}),$$

$$\mathbf{GF}\text{attempt} \rightarrow \mathbf{GF}\text{success}.$$

Scheduling processes, responding to messages, no process is blocked forever, etc.

Definition 7.29 (Language $\mathcal{L}_{LTL}$ [Pnueli, 1977])

The language $\mathcal{L}_{\mathsf{LTL}}(\mathcal{P}rop)$ is given by all formulae generated by the following grammar, where $\mathsf{p} \in \mathcal{P}rop$ is a proposition:

$$\varphi ::= \mathsf{p} \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \,\mathcal{U}\, \varphi \mid \mathbf{X}\varphi.$$

The additional operators

- **F** (eventually in the future) and
- **G** (always from now on)

can be defined as macros :

$$\mathsf{F}\varphi \equiv (\neg\Box)\,\mathcal{U}\,\varphi \qquad \text{and} \qquad \mathsf{G}\varphi \equiv \neg\mathsf{F}\neg\varphi$$

The standard Boolean connectives $\top, \wedge, \rightarrow$, and $\leftrightarrow$ are defined in their usual way as **macros**.

Models of $\mathcal{L}_{LTL}$

The semantics is given over paths, which are **infinite sequences of states** from S, and a standard labelling function $\pi : \mathrm{S} \to \mathcal{P}(\mathcal{P}rop)$ that determines which propositions are true at which states.

Definition 7.30 (Path $\lambda = q_0 q_1 q_2 q_3 \ldots$)

- A path λ over a set of states S is an infinite sequence of states. We can view it as a mapping $\mathbb{N}_0 \to \mathrm{S}$. The set of all sequences is denoted by S^ω.
- $\lambda[i]$ denotes the ith position on path λ (starting from $i = 0$) and
- $\lambda[i, \infty]$ denotes the subpath of λ starting from i ($\lambda[i, \infty] = \lambda[i]\lambda[i+1]\ldots$).

$$\lambda = q_0 q_1 q_2 q_3 \ldots \in \mathrm{S}^\omega$$

Definition 7.31 (Semantics of $\mathcal{L}_{LTL}$)

Let λ be a path and π be a labelling function over S. The semantics of **LTL**, $\models^{\mathsf{LTL}}$, is defined as follows:

- $\lambda, \pi \models^{\mathsf{LTL}} \mathsf{p}$ *if, by definition,* $\mathsf{p} \in \pi(\lambda[0])$ and $\mathsf{p} \in \mathcal{P}rop$;
- $\lambda, \pi \models^{\mathsf{LTL}} \neg\varphi$ *if, by definition,* **not** $\lambda, \pi \models^{\mathsf{LTL}} \varphi$ (we write $\lambda, \pi \not\models^{\mathsf{LTL}} \varphi$);
- $\lambda, \pi \models^{\mathsf{LTL}} \varphi \vee \psi$ *if, by definition,* $\lambda, \pi \models^{\mathsf{LTL}} \varphi$ **or** $\lambda, \pi \models^{\mathsf{LTL}} \psi$;
- $\lambda, \pi \models^{\mathsf{LTL}} \mathbf{X}\varphi$ *if, by definition,* $\lambda[1, \infty], \pi \models^{\mathsf{LTL}} \varphi$; and
- $\lambda, \pi \models^{\mathsf{LTL}} \varphi \mathcal{U} \psi$ *if, by definition,* there is an $i \in \mathbb{N}_0$ such that $\lambda[i, \infty], \pi \models \psi$ and $\lambda[j, \infty], \pi \models^{\mathsf{LTL}} \varphi$ for all $0 \leq j < i$.

Other temporal operators

$\lambda, \pi \models \mathbf{F}\varphi$ *if, by definition,* $\lambda[i,\infty], \pi \models \varphi$ for some $i \in \mathbb{N}_0$;
$\lambda, \pi \models \mathbf{G}\varphi$ *if, by definition,* $\lambda[i,\infty], \pi \models \varphi$ for all $i \in \mathbb{N}_0$;

Exercise

Prove that the semantics does indeed match the definitions;

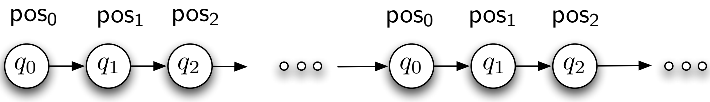
- $\mathbf{F}\varphi$ is equivalent to $(\neg\Box)\,\mathcal{U}\,\varphi$, and
- $\mathbf{G}\varphi$ is equivalent to $\neg\mathbf{F}\neg\varphi$.

Validity, satisfiability

satisfiable: a LTL formula is satisfiable, *if, by definition,* there is a model for it,

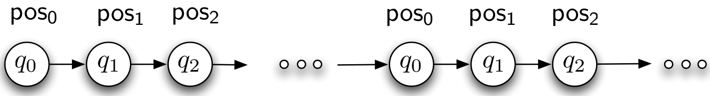
valid: a LTL formula is valid, *if, by definition,* it is true in all models,

contradictory: a LTL formula is contradictory, *if, by definition,* there is no model for it.



$$\lambda, \pi \models \mathbf{F}\mathsf{pos}_1$$

$$\lambda' = \lambda[1,\infty], \pi \models \mathsf{pos}_1$$
$$\mathsf{pos}_1 \in \pi(\lambda'[0])$$



$\lambda, \pi \models \mathbf{GF}\mathsf{pos}_1$ *if and only if*

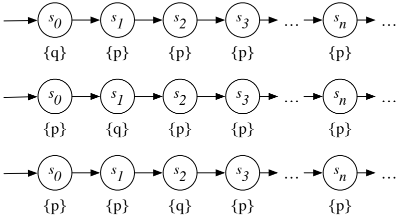
$$\lambda[0,\infty], \pi \models \mathbf{F}\mathsf{pos}_1 \text{ and}$$
$$\lambda[1,\infty], \pi \models \mathbf{F}\mathsf{pos}_1 \text{ and}$$
$$\lambda[2,\infty], \pi \models \mathbf{F}\mathsf{pos}_1 \text{ and}$$
$$\ldots$$

Paths and infinite sets of paths

- Paths are infinite entities, and so are infinite sets of them.
- They are both theoretical constructs.
- In order to work with them we need a finite representation:
- namely transition systems (also called pointed Kripke structures).

A set of paths (1)

We reconsider the paths $\lambda_0, \lambda_1, \ldots, \lambda_i, \ldots$ from Example 7.22 on Slide 495:



- $q \wedge \neg p \wedge \mathbf{XG}(p \wedge \neg q)$
- $p \wedge \mathbf{X}q \wedge \mathbf{XXG}p$
- $p \wedge \mathbf{X}p \wedge \mathbf{XX}q$

etc.

Can we distinguish between them (using LTL)?

Indistinguishable paths

Observation

While any two paths can be distinguished by appropriate LTL formulae, these formulae get more and **more complicated**: operators need to be nested.

- The first two paths can be distinguished by just propositional logic, no LTL connectives are needed.
- But the second and third cannot: we need **X**.
- For the third and fourth we need a nesting **XX**.

By induction over the structure of φ

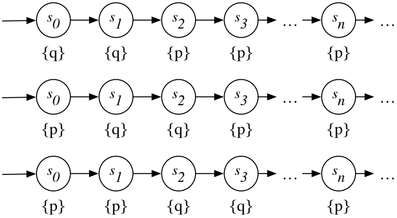
Given any LTL formula φ, there is a $i_0 \in \mathbb{N}$ such that for all $i, j \geq i_0$: $\lambda_i \models \varphi$ if and only if $\lambda_j \models \varphi$.

A set of paths (2)

- Can we find a LTL formula, or a set of LTL formulae, that **characterize exactly the whole set of paths**?
- So what holds true in *all of these paths*?
- $p\mathcal{U}q$. But this is also true in other paths not listed above.
- $(p \wedge \neg q)\mathcal{U}(q \wedge \neg p)$. Again, this is also true in other paths.
- $(p \wedge \neg q)\mathcal{U}(q \wedge \neg p \wedge \mathbf{XG}(p \wedge \neg q))$. *That is it.* This describes *exactly* the set of paths above.

Another set of paths (1)

We reconsider the second set of paths from Example 7.22 on Slide 495:



etc.

- $q \wedge \mathbf{X}q \wedge \mathbf{XXG}(p \wedge \neg q)$
- $p \wedge \mathbf{X}q \wedge \mathbf{XX}q \wedge \mathbf{XXXG}p$
- etc.

What holds true in exactly these paths?

$$(p \wedge \neg q)\,\mathcal{U}\,(q \wedge \neg p \wedge \mathbf{X}(q \wedge \neg p) \wedge \mathbf{XXG}(p \wedge \neg q))$$

LTL formulae and transition systems: $TS \models \phi$

- Up to now we have defined LTL formulae **only for paths**.
- For a transition system *TS*, we say that an LTL formula is true in a state s, if it is true in all runs resulting from that state.
- A LTL formula is true in the whole transition system, if it is true in all runs resulting from the initial states.

LTL formulae and transition systems: *TS* $\models \varphi$

Definition 7.32 (*TS* and LTL formulae: *TS* $\models \phi$)

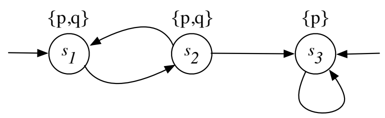
Let a *TS* and a LTL formula φ be given.

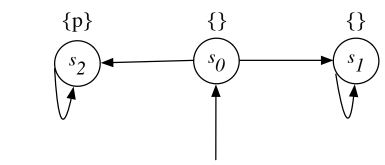
- **A** LTL **formula** φ **is true in a state** s **of *TS***, *if, by definition*, it is true in **all runs resulting from** s.

- A LTL formula φ is true in TS, *if, by definition,* it is true in all runs resulting from all initial states.

LTL formulae and transition systems

Example 7.33 (LTL formulae)





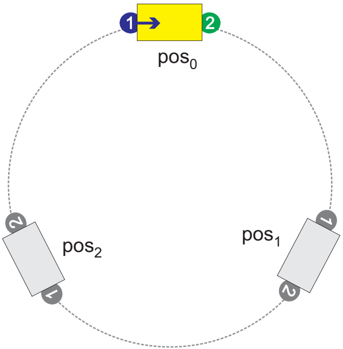
Which formulae hold true?

TS_1: $TS_1 \models \mathbf{G}p$, $TS_1 \not\models \mathbf{X}(p \wedge q)$,
$TS_1 \not\models \mathbf{G}(q \rightarrow \mathbf{G}(p \wedge q))$,
but $TS_1 \models \mathbf{G}(q \rightarrow (p \wedge q)))$

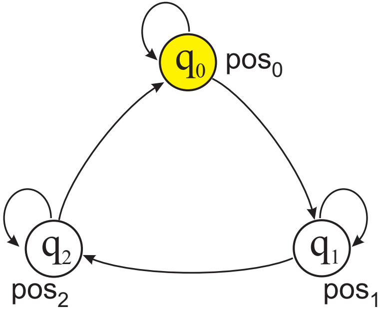
TS_2: $TS_2 \not\models \mathbf{F}p$ and $TS_2 \not\models \neg\mathbf{F}p$

From system to behavioral structure

System

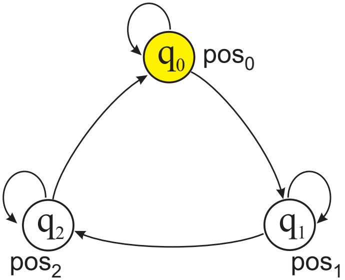


Computational str.

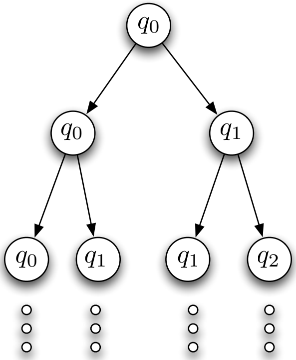


From computational to behavioral structure

Computational str.



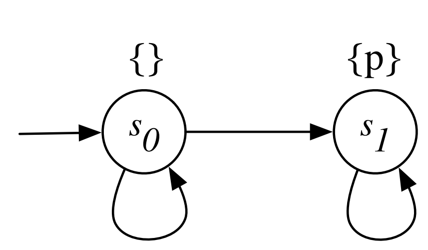
Behavioral str.



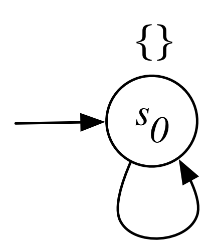
The behavioral structure is usually infinite! Here, it is an infinite tree. We say it is the q_0-unfolding of the model.

Example 7.34 (LTL indistinguishable transition systems)

TS_1:



TS_2:



Both systems can be distinguished by the property **"a state where p holds can be reached"**.

- Each trace of TS_2 is also one of TS_1: **each LTL formula true in TS_1 is also true in TS_2.**
- So the above property cannot be expressed in LTL.

LT-Properties and their expressibility

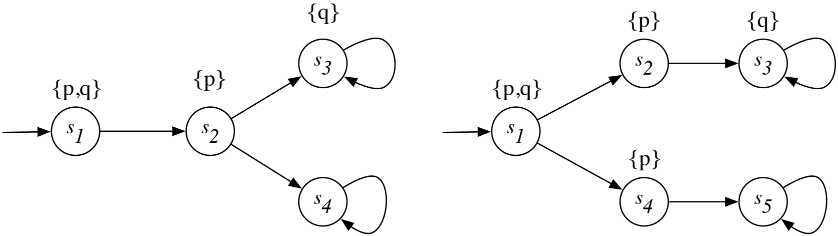
- The example on the preceding slide shows that a certain property is not a LT property.
- **This does not mean that the two transition systems cannot be distinguished.**
- In fact the formula $\mathbf{X}\neg\mathrm{p}$ is true in TS_2 but not in TS_1.
- The traces of the two systems are also different, so both define different LT properties.

Are there TS_1 and TS_2 that **define different LT-properties but cannot be distinguished by any set of $\mathcal{L}_{LTL}$ formulae**?

Yes. $\mathcal{L}_{LTL}$ formulae express exactly "$*$-free ω-regular properties", a strict subset of "ω-regular properties", which is itself a strict subset of LT-properties.

LT-Properties and their expressibility (cont.)

We consider again Example 7.19 with the two transition systems TS_1 and TS_2.



1. The property **whenever** p **a state with** q **can be reached** distinguishes them.
2. Can this property be expressed in LTL?
3. No, because both have the same set of traces.

Some Exercises

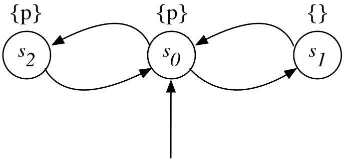
Example 7.35 (Formalizing properties in LTL)

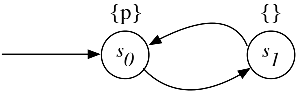
Formalise the following properties as **LTL** formulae over $\mathcal{P}rop = \{\mathrm{p}\}$

1. p should never occur.
2. p should occur exactly once.
3. At least once p is directly be followed by $\neg$p.
4. p is true at exactly all even states.

Some Exercises (cont.)

Compare the following two transition systems:





Example 7.36 (Evenness)

Formalise the following as a **LTL** formula: p is true at **all even** states (the odd states do not matter).

Does $\mathrm{p} \wedge \mathbf{G}(\mathrm{p} \rightarrow \mathbf{XX}\mathrm{p})$ work?

Satisfiability of LTL formulae

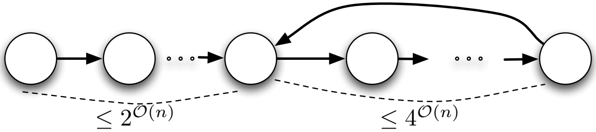
A formula is satisfiable, if there is a model (i.e. path) where it holds true. Can we **restrict the structure** of such paths? I.e. can we restrict to simple paths, for example paths that are periodic?

- 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 long the period is** and **within which initial segment** of the path it starts, depending on the length of the formula φ.

Satisfiability of LTL formulae (cont.)

Theorem 7.37 (Periodic model theorem [Sistla and Clarke, 1985])

*A formula $\varphi \in$ **LTL** is satisfiable if and only if there is a path λ which is ultimately periodic, and the period starts within $2^{1+|\varphi|}$ steps and has a length which is $\leq 4^{1+|\varphi|}$.*



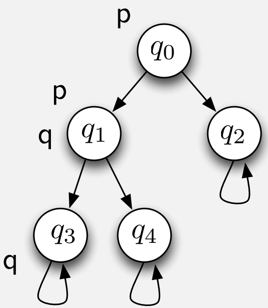
7.5 Branching Time Logic (**CTL**)

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). ($\leadsto$ Kripke models).
- Paths: courses of action, computations. ($\leadsto$ **LTL**)

- Path quantifiers: A (for all paths), E (there is a path);
- Temporal operators: **X** (nexttime), **F** (finally), **G** (globally) and $\mathcal{U}$ (until);
- **CTL**: each temporal operator must be immediately preceded by exactly one path quantifier;
- **CTL***: no syntactic restrictions.

Example 7.38 (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 ¬q holds in the next step. **And this holds along all paths (there are three infinite paths).**

Definition 7.39 ($\mathcal{L}_{CTL^*}$ [Emerson and Halpern, 1986])

The language $\mathcal{L}_{\mathsf{CTL}^*}(\mathcal{P}rop)$ is given by all formulae generated by the following grammar:

$$\varphi ::= \mathsf{p} \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{E}\gamma$$

where

$$\gamma ::= \varphi \mid \neg\gamma \mid \gamma \vee \gamma \mid \gamma \,\mathcal{U}\, \gamma \mid \mathbf{X}\gamma$$

and $\mathsf{p} \in \mathcal{P}rop$. Formulae φ (resp. γ) are called state (resp. path) formulae.

We use the same abbreviations as for $\mathcal{L}_{LTL}$:

$\lambda, \pi \models \mathbf{F}\varphi$ iff $\lambda[i, \infty], \pi \models \varphi$ for some $i \in \mathbb{N}_0$;
$\lambda, \pi \models \mathbf{G}\varphi$ iff $\lambda[i, \infty], \pi \models \varphi$ for all $i \in \mathbb{N}_0$;

- The $\mathcal{L}_{CTL^*}$-formula $\mathsf{E}\mathbf{F}\varphi$, for instance, ensures that there is at least one path on which φ holds at some (future) time moment.
- The formula $\mathsf{A}\mathbf{FG}\varphi$ states that φ 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 7.40 ($\mathcal{L}_{CTL}$ [Clarke and Emerson, 1981])

The language $\mathcal{L}_{\mathsf{CTL}}(\mathcal{P}rop)$ is given by all formulae generated by the following grammar, where $\mathsf{p} \in \mathcal{P}rop$ is a proposition:

$$\varphi ::= \mathsf{p} \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{E}(\varphi \boldsymbol{\mathcal{U}}\, \varphi) \mid \mathsf{E}\mathbf{X}\varphi \mid \mathsf{E}\mathbf{G}\varphi.$$

We introduce the following macros:

- $\mathbf{F}\varphi \equiv \top \boldsymbol{\mathcal{U}}\, \varphi$,
- $\mathsf{A}\mathbf{X}\varphi \equiv \neg\mathsf{E}\mathbf{X}\neg\varphi$,
- $\mathsf{A}\mathbf{G}\varphi \equiv \neg\mathsf{E}\mathbf{F}\neg\varphi$, and
- $\mathsf{A}\varphi \boldsymbol{\mathcal{U}}\, \psi \equiv \ldots$ Exercise!

For example, A**G**E**X**p is a $\mathcal{L}_{CTL}$-formula whereas A**GF**p is not.

Example 7.41 (CTL* or CTL?)

Are the following **CTL*** or **CTL** formulae? What do they express?

1. E**F**A**X**shutdown
2. E**FX**shutdown
3. A**GF**rain
4. A**G**A**F**rain (Is it different from (3)?)
5. E**FG**broken
6. A**G**(p → (E**X**q ∧ E**X**¬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 S (each is a propositional model), a Kripke model $\mathfrak{M}$ is simply a tuple $(\mathrm{S}, \mathcal{R})$ where $\mathcal{R} \subseteq \mathrm{S} \times \mathrm{S}$ 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 $q\mathcal{R}q_1$, . . . , $q_i\mathcal{R}q_{i+1}, \ldots$

Definition 7.42 (Semantics $\models^{\mathsf{CTL}^*}$)

Let $\mathfrak{M}$ be a Kripke model, $q \in S$ and $\lambda \in \Lambda$. The semantics of $\mathcal{L}_{\mathsf{CTL}^*}$- and $\mathcal{L}_{\mathsf{CTL}}$-formulae is given by the satisfaction relation $\models^{\mathsf{CTL}^*}$ for state formulae by

- $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \mathsf{p}$ iff $\lambda[0] \in \pi(\mathsf{p})$ and $\mathsf{p} \in \mathcal{P}rop$;
- $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \neg\varphi$ iff $\mathfrak{M}, q \not\models^{\mathsf{CTL}^*} \varphi$;
- $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \varphi \vee \psi$ iff $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \varphi$ or $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \psi$;
- $\mathfrak{M}, q \models^{\mathsf{CTL}^*} \mathsf{E}\varphi$ iff there is a path $\lambda \in \Lambda(q)$ such that $\mathfrak{M}, \lambda \models^{\mathsf{CTL}^*} \varphi$;

and for path formulae by:

- $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \varphi$ iff $\mathfrak{M}, \lambda[0] \models^{\mathbf{CTL}^*} \varphi$;
- $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \neg\gamma$ iff $\mathfrak{M}, \lambda \not\models^{\mathbf{CTL}^*} \gamma$;
- $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \gamma \vee \delta$ iff $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \gamma$ or $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \delta$;
- $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \mathbf{X}\gamma$ iff $\mathfrak{M}, \lambda[1, \infty] \models^{\mathbf{CTL}^*} \gamma$; and
- $\mathfrak{M}, \lambda \models^{\mathbf{CTL}^*} \gamma \,\mathcal{U}\, \delta$ iff there is an $i \in \mathbb{N}_0$ such that $\mathfrak{M}, \lambda[i, \infty] \models^{\mathbf{CTL}^*} \delta$ and $\mathfrak{M}, \lambda[j, \infty] \models^{\mathbf{CTL}^*} \gamma$ for all $0 \leq j < i$.

Is this complicated semantics over paths really necessary for CTL?

State-based semantics for CTL

- $\mathfrak{M}, q \models^{\textsf{CTL}} p$ iff $q \in \pi(p)$;
- $\mathfrak{M}, q \models^{\textsf{CTL}} \neg\varphi$ iff $\mathfrak{M}, q \not\models^{\textsf{CTL}} \varphi$;
- $\mathfrak{M}, q \models^{\textsf{CTL}} \varphi \vee \psi$ iff $\mathfrak{M}, q \models^{\textsf{CTL}} \varphi$ or $\mathfrak{M}, q \models^{\textsf{CTL}} \psi$;
- $\mathfrak{M}, q \models^{\textsf{CTL}} \mathsf{E}\mathbf{X}\varphi$ iff there is a path $\lambda \in \Lambda(q)$ such that $\mathfrak{M}, \lambda[1] \models^{\textsf{CTL}} \varphi$;
- $\mathfrak{M}, q \models^{\textsf{CTL}} \mathsf{E}\mathbf{G}\varphi$ iff there is a path $\lambda \in \Lambda(q)$ such that $\mathfrak{M}, \lambda[i] \models^{\textsf{CTL}} \varphi$ for every $i \geq 0$;
- $\mathfrak{M}, q \models^{\textsf{CTL}} \mathsf{E}\varphi\,\boldsymbol{\mathcal{U}}\,\psi$ iff there is a path $\lambda \in \Lambda(q)$ such that $\mathfrak{M}, \lambda[i] \models^{\textsf{CTL}} \psi$ for some $i \geq 0$, and $\mathfrak{M}, \lambda[j] \models^{\textsf{CTL}} \varphi$ 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 φ, we construct φ' by just removing all path operators and putting A in front of it. Then

φ is equivalent to a **LTL** formula
iff
φ and φ' are equivalent under the semantics of **CTL***.

Theorem of Clarke und Draghiescu

Theorem 7.43 (Clarke und Draghiescu)

*Given a **CTL*** formula φ, we construct $\hat{\varphi}$ by simply* removing all path operators*. Then the following are equivalent:*

- *φ is equivalent to a **LTL** formula,*
- *φ and $\mathsf{A}\hat{\varphi}$ are equivalent in **CTL***.*

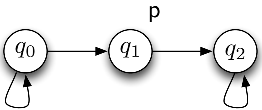
Application of Clarke and Draghiescu

We consider the **LTL** formula **GF**p. Viewed as a **CTL*** formula it becomes A**GF**p. But this is equivalent (in **CTL***) to A**G**A**F**p, a CTL formula.

Now we consider the **CTL** formula E**G**E**F**p. It is not equivalent to any LTL formula. This is because

E**G**E**F**p and A**GF**p

are not equivalent in **CTL***: a counterexample is



The first formula holds, the second does not.

LTL as subset of CTL* (2)

- How do LTL and CTL compare?
- The **CTL** formula $A\mathbf{G}(p \rightarrow (E\mathbf{X}q \wedge E\mathbf{X}\neg q))$ describes Kripke structures of the form in Example 7.38. No LTL formula can describe this class of Kripke structures.
- The **LTL** formula $A\mathbf{F}(p \wedge \mathbf{X}p)$ **can not be expressed** by a **CTL** formula. Check why neither $A\mathbf{F}(p \wedge A\mathbf{X}p)$ nor $A\mathbf{F}(p \wedge E\mathbf{X}p)$ are equivalent. Similarly, the **LTL** formula $A\mathbf{FG}p$ **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 7.44 (Robots and Carriage)

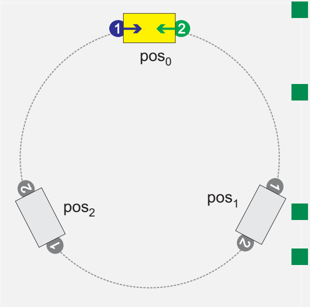
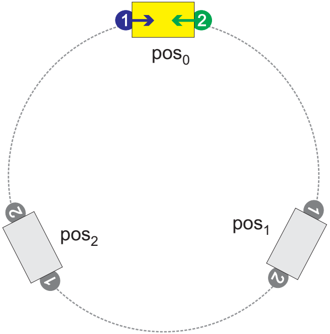


Figure 33: Two robots and a carriage.

- 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 pos_0, pos_1, pos_2, respectively, to allow for referring to the current position of the carriage in the object language.



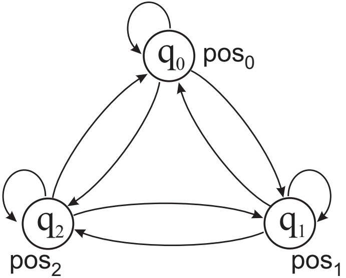
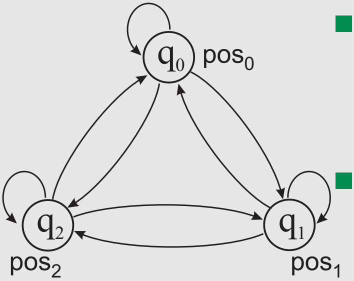


Figure 34: Two robots and a carriage: A schematic view (left) and a transition system $\mathfrak{M}_0$ that models the scenario (right).



- $\mathfrak{M}_0, q_0 \models^{CTL}$ **E****F**pos$_1$: In state q_0, there is a path such that the carriage will reach position 1 sometime in the future.
- The same is not true for *all* paths, so we also have: $\mathfrak{M}_0, q_0 \not\models^{CTL}$ **A****F**pos$_1$.

It becomes more interesting if abilities of agents are considered $\rightsquigarrow$ **ATL**.

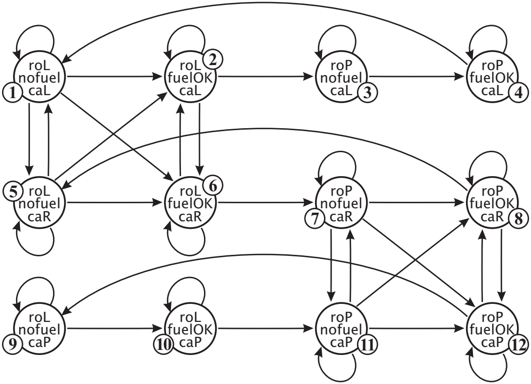
Example: Rocket and Cargo

Example 7.45 (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, and nofuel holds after each flight.

Example: Rocket and Cargo (cont.)

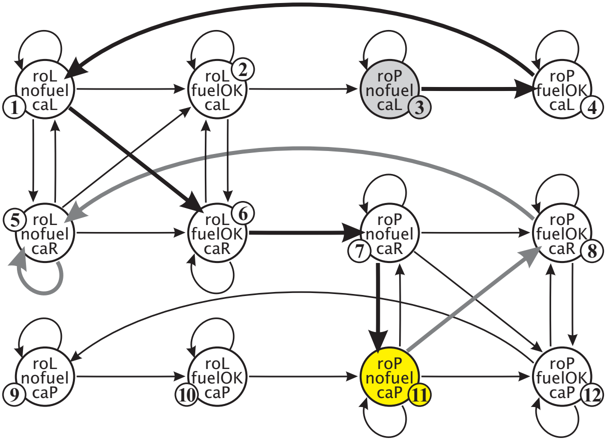


$\mathsf{roL} \to \mathbf{EF}\mathsf{roP}$

$\mathbf{AG}(\mathsf{roL} \vee \mathsf{roP})$

$\mathsf{roL} \to \mathbf{AX}(\mathsf{roP} \to \mathsf{nofuel})$

Example: Rocket and Cargo (cont.)



E**F**caP

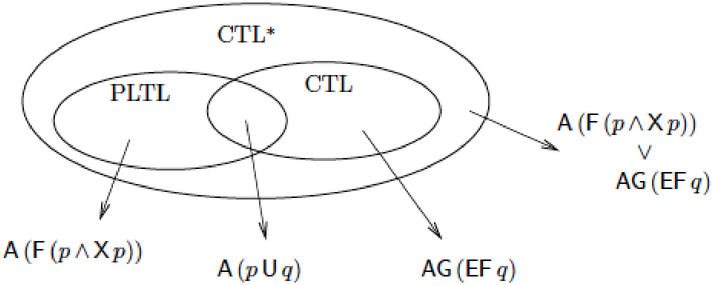
- In our logics, we assumed a serial accessibility relation: no deadlocks are possible.
- One can also allow states with no outgoing transitions. In that case, in the semantical definition of E on Slide 543 one has to replace “there is a path” by “there is an infinite path or one which can not be extended”.
- Similar modifications are needed in the definition of **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?

$\mathsf{A}\mathbf{G}\mathbf{X}\top \quad (\rightsquigarrow \textbf{CTL}^*)$ $\qquad$ $\mathsf{A}\mathbf{G}\mathsf{E}\mathbf{X}\top \quad (\rightsquigarrow \textbf{CTL})$

A Venn diagram showing typical formulae in the respective areas.



7.6 References

Clarke, E. and Emerson, E. (1981).
Design and synthesis of synchronization skeletons using branching time temporal logic.
In *Proceedings of Logics of Programs Workshop*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71.

Dix, J. and Fisher, M. (2013).
Chapter 14: Specification and Verification of Multi-agent Systems.
In G. Weiss (Ed.), Multiagent Systems, MIT Press.

E. A. Emerson (1990).
Temporal and modal logic.
Handbook of Theoretical Computer Science, volume B, 995–1072. Elsevier.

Emerson, E. and Halpern, J. (1986).
"Sometimes" and "Not Never" Revisited: On Branching versus Linear Time Temporal Logic.
Journal of the ACM, **33(1)**:151–178.

Maidl, M. (2000).
The Common Fragment of CTL and LTL.
41st Annual Symposium on Foundations of Computer Science, FOCS 2000, pages 643–652.
IEEE Computer Society

Pnueli, A. (1977).
The Temporal Logic of Programs.
In *Proceedings of FOCS*, pages 46–57.

Sistla, A. P. and Clarke, E. M. (1985).
The complexity of propositional linear temporal logics.
J. ACM, **32(3)**:733–749.

8. Strategic Logics

8 Strategic Logics
- Alternating-Time Temporal Logic (ATL)
- Perfect vs. imperfect recall
- Imperfect Information
- Perfect Recall Revisited
- Perfect Recall Revisited
- Defining Equilibria in ATLP

Outline

- 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*

$\langle\langle A\rangle\rangle\varphi$: **coalition A has a collective strategy to** *bring about* φ

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 $\mathcal{L}_{ATL^*}$ [Alur et al., 1997])

The **language** $\mathcal{L}_{ATL^*}$ is given by all formulae generated by the following grammar:

$$\varphi ::= \mathsf{p} \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\langle A \rangle\rangle\gamma \quad \text{where}$$
$$\gamma ::= \varphi \mid \neg\gamma \mid \gamma \vee \gamma \mid \gamma \,\mathcal{U}\, \gamma \mid \bigcirc\gamma,$$

$A \subseteq \mathbb{A}\mathrm{gt}$, and $\mathsf{p} \in \mathcal{P}rop$. Formulae φ (resp. γ) are called **state** (resp. **path**) formulae.

Notation: $\bigcirc$ instead of **X**

Note that we are now using the symbol "$\bigcirc$" instead of "**X**" as it is more custom when dealing with **ATL**.

The language $\mathcal{L}_{ATL}$ **restricts** $\mathcal{L}_{ATL^*}$ in the same way as $\mathcal{L}_{CTL}$ restricts $\mathcal{L}_{CTL^*}$:

Each temporal operator must be directly preceded by a cooperation modality.

Definition 8.2 (Language $\mathcal{L}_{ATL}$ [Alur et al., 1997])

The *language* $\mathcal{L}_{ATL}$ is given by all formulae generated by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\langle A \rangle\rangle \bigcirc \varphi \mid \langle\langle A \rangle\rangle \Box \varphi \mid \langle\langle A \rangle\rangle \varphi \,\mathcal{U}\, \varphi$$

where $A \subseteq \mathbb{A}\mathrm{gt}$ and p $\in \mathcal{P}rop$.

Notation: □ instead of **G**

Note that we are using now the symbol "□" instead of "**G**" as it is more custom when dealing with **ATL**.

The language $\mathcal{L}_{ATL^+}$ **restricts** $\mathcal{L}_{ATL^*}$ but extends $\mathcal{L}_{ATL}$. It allows for Boolean combinations of path formulae.

Definition 8.3 (Language $\mathcal{L}_{ATL^+}$)

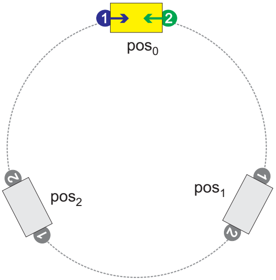
The *language* $\mathcal{L}_{ATL^+}$ is given by all formulae generated by the following grammar:

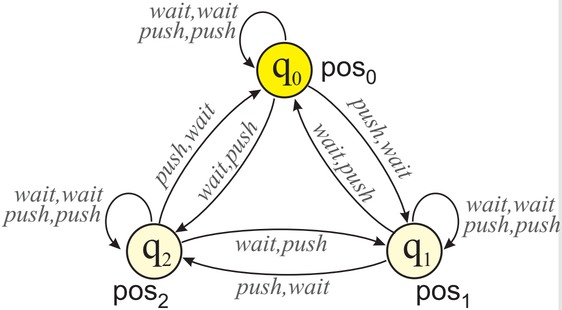
$$\varphi ::= \mathsf{p} \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\!\langle A \rangle\!\rangle\gamma, \qquad \gamma ::= \neg\gamma \mid \gamma \vee \gamma \mid \bigcirc\varphi \mid \varphi\,\mathcal{U}\,\varphi.$$

where $A \subseteq \mathbb{A}\mathrm{gt}$ and $\mathsf{p} \in \mathcal{P}rop$.

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 $\mathfrak{M} = \langle \mathbb{A}\mathrm{gt}, \mathrm{S}, \pi, Act, d, o \rangle$, where:

- $\mathbb{A}\mathrm{gt}$: a finite set of all **agents**;
- S: a set of **states**;
- $\pi : \mathrm{S} \to \mathcal{P}(\mathcal{P}rop)$: a **valuation** of propositions;
- Act: a finite set of (atomic) **actions**;
- $d : \mathbb{A}\mathrm{gt} \times \mathrm{S} \to \mathcal{P}(Act)$ 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 ($\rightsquigarrow$ **memoryless strategies**)

  $$s_a : \mathrm{S} \to \mathit{Act}.$$

- on the **whole history** of events that have happened ($\rightsquigarrow$ **perfect recall strategies**)

  $$s_a : \mathrm{S}^+ \to \mathit{Act}.$$

We also distinguish between agents with

- perfect information (all states are distinguishable).
- imperfect information (some state 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

$$s_a : \mathrm{S}^+ \to Act \textbf{ such that } s_a(q_0 q_1 \dots q_n) \in d_a(q_n).$$

The set of such strategies is denoted by Σ_a^{IR}.

- A perfect information memoryless strategy for agent a (***Ir*-strategy** for short) is given by a function

$$s_a : \mathrm{S} \to Act \textbf{ where } s_a(q) \in d_a(q).$$

The set of such strategies is denoted by Σ_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 $A = \{a_1, \ldots, a_r\} \subseteq \mathbb{A}\mathrm{gt}$ 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.
- Σ_A denotes the set of all collective strategies of A.
- $\Sigma = \Sigma_{\mathbb{A}\mathrm{gt}}$

Outcome of a strategy

$out(q, s_A)$= set of **all paths** that **may occur** when agents A **execute** s_A **from** state q **onward**.

Definition 8.6 (Outcome)

$\lambda = q_0 q_1 \ldots \in \mathrm{S} \in out(q, s_A) \subseteq \mathrm{S}^\omega$ iff

1. $q_0 = q$
2. for each $i = 1, \ldots$ there is a tuple $(\alpha_1^{i-1}, \ldots, \alpha_k^{i-1}) \in Act^k$ such that
   - $\alpha_a^{i-1} \in d_a(q_{i-1})$ for each $a \in \mathbb{A}\mathrm{gt}$,
   - $\alpha_a^{i-1} = s_A|_a(q_0 q_1 \ldots q_{i-1})$ for each $a \in A$, and
   - $o(q_{i-1}, \alpha_1^{i-1}, \ldots, \alpha_k^{i-1}) = q_i$.

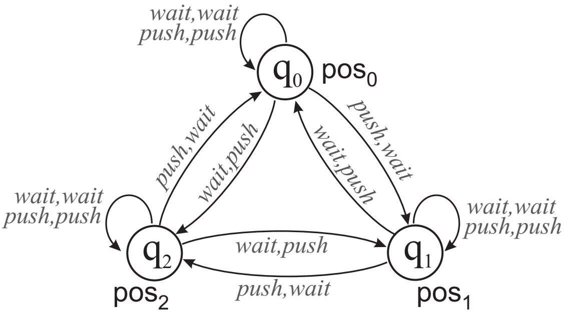
For an Ir-strategy replace “$s_A|_a(q_0 q_1 \ldots q_{i-1})$” by “$s_A|_a(q_{i-1})$”.

Definition 8.7 (Perfect information semantics)

$\mathfrak{M}, q \models_{\mathsf{Ix}} p$	iff p is in $\pi(q)$;
$\mathfrak{M}, q \models_{\mathsf{Ix}} \varphi \lor \psi$	iff $\mathfrak{M}, q \models_{\mathsf{Ix}} \varphi$ or $\mathfrak{M}, q \models_{\mathsf{Ix}} \psi$;
$\mathfrak{M}, q \models_{\mathsf{Ix}} \langle\langle A \rangle\rangle \Phi$	iff **there is a collective Ix-strategy** s_A such that, **for each path** $\lambda \in out(q, s_A)$, we have $\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \Phi$.
$\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \bigcirc\varphi$	iff $\mathfrak{M}, \lambda[1, \infty] \models_{\mathsf{Ix}} \varphi$;
$\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \Diamond\varphi$	iff $\mathfrak{M}, \lambda[i, \infty] \models_{\mathsf{Ix}} \varphi$ for some $i \geq 0$;
$\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \Box\varphi$	iff $\mathfrak{M}, \lambda[i, \infty] \models_{\mathsf{Ix}} \varphi$ for all $i \geq 0$;
$\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \varphi \mathcal{U} \psi$	iff $\mathfrak{M}, \lambda[i, \infty] \models_{\mathsf{Ix}} \psi$ for some $i \geq 0$, and $\mathfrak{M}, \lambda[j, \infty] \models_{\mathsf{Ix}} \varphi$ forall $0 \leq j \leq i$.

Note that temporal formulae and Boolean connectives are handled as before.

Example: Robots and Carriage



$$\mathsf{pos}_0 \rightarrow \langle\langle 1 \rangle\rangle \Box \neg \mathsf{pos}_1$$

Definition 8.8 ($\mathbf{ATL}_{Ix}$, $\mathbf{ATL}^{+}_{Ix}$, $\mathbf{ATL}^{*}_{Ix}$, ATL, $\mathbf{ATL}^{*}$)

We define **$\mathbf{ATL}_{Ix}$**, **$\mathbf{ATL}^{+}_{Ix}$**, and **$\mathbf{ATL}^{*}_{Ix}$** as the logics $(\mathcal{L}_{\mathsf{ATL}}, \models_{Ix})$, $(\mathcal{L}_{\mathsf{ATL}^+}, \models_{Ix})$ and $(\mathcal{L}_{\mathsf{ATL}^*}, \models_{Ix})$ where $x \in \{r, R\}$, respectively. Moreover, we use **ATL** (resp. **$\mathbf{ATL}^*$**) as an abbreviation for **$\mathbf{ATL}_{IR}$** (resp. **$\mathbf{ATL}^*_{IR}$**).

As usual, a *logic* is given by the *set of all valid formulae*.

8.2 Perfect vs. imperfect recall

Theorem 8.9

For $\mathcal{L}_{ATL}$, the perfect recall semantics is equivalent to the memoryless semantics under perfect information, i.e., $\mathfrak{M}, q \models_{IR} \varphi$ *iff* $\mathfrak{M}, q \models_{Ir} \varphi$. *That is*

$$ATL_{Ir} = ATL_{IR}.$$

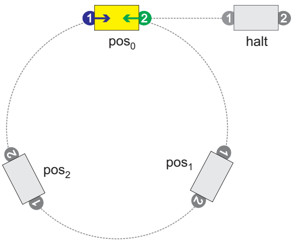
Proof idea.

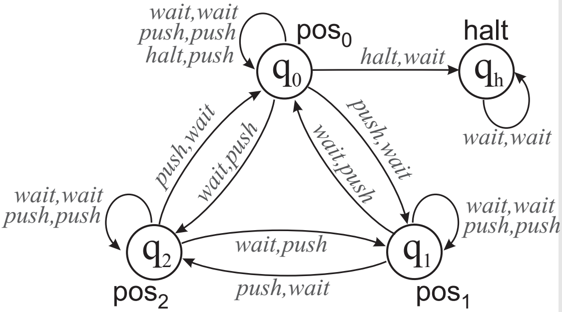
The first "non-looping part" of each path has to satisfy a formula.
$\rightsquigarrow$ Exercise $\square$

The property has been first observed in [Schobbens, 2004] but it follows from [Alur et al. 2002] in a straightforward way.

Both semantics are different for $\mathcal{L}_{ATL^*}$.

Example: Robots and Carriage (2)





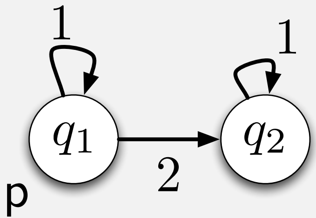
What about $\langle\langle 1,2 \rangle\rangle(\Diamond \mathsf{pos}_1 \wedge \Diamond \mathsf{halt})$?

$\mathfrak{M}, q_0 \models_{IR} \langle\langle 1,2 \rangle\rangle(\Diamond \mathsf{pos}_1 \wedge \Diamond \mathsf{halt})$

$\mathfrak{M}, q_0 \not\models_{Ir} \langle\langle 1,2 \rangle\rangle(\Diamond \mathsf{pos}_1 \wedge \Diamond \mathsf{halt})$

Example 8.10 ($\mathrm{ATL}^*_{IR} \neq \mathrm{ATL}^*_{Ir}$)

There is one agent, **a**, with two actions available in q_1 and just one in q_2.



$$\varphi = \langle\langle a \rangle\rangle (\bigcirc p \wedge \bigcirc \bigcirc \neg p)$$

IR-Tree Unfolding

- 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 $\mathbf{ATL}^*_{Ir} \subseteq \mathbf{ATL}^*_{IR}$; i.e., more properties of games are valid if perfect recall strategies are considered.
- For this purpose, we show that every *IR*-satisfiable formula is also *Ir*-satisfiable.
- Then, the claim follows: Suppose $\varphi \in \mathbf{ATL}^*_{Ir}$ and $\varphi \notin \mathbf{ATL}^*_{IR}$. By the latter, $\neg\varphi$ is *IR*-satisfiable hence also *Ir*-satisfiable. Contradiction!

How can we show that IR-satisfiability implies Ir-satisfiability?

- Suppose $(\mathfrak{M}, q)$ *IR*-satisfies φ. Then, we show that there is a pointed model $(\mathfrak{M}', q)$ which satisfies the same formulae and in which memoryless and perfect-recall strategies coincide.
- Which properties must $\mathfrak{M}'$ have so that both kind of strategies have the same expressive power?

Definition 8.11 (Tree-like CGS)

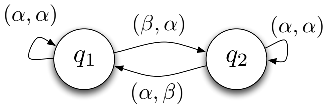
Let $\mathfrak{M}$ be a CGS. $\mathfrak{M}$ is called **tree-like** iff, by definition, there is a state q_0 (the *root*) such that for every q there is a unique history leading from q_0 to q.

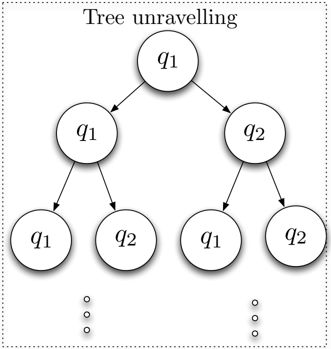
Proposition 8.12 (Recall invariance for tree-like CGS)

For every tree-like CGS $\mathfrak{M}$, state q in $\mathfrak{M}$, and ***ATL******-formula φ, we have:* $\mathfrak{M}, q \models_{Ir} \varphi$ **iff** $\mathfrak{M}, q \models_{IR} \varphi$.

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 $\mathfrak{M} = (\mathbb{A}\mathrm{gt}, \mathrm{S}, \Pi, \pi, Act, d, o)$ be a CGS and q be a state in it. The (perfect information) tree unfolding of the pointed model $(\mathfrak{M}, q)$ denoted $T(\mathfrak{M}, q)$ is defined as $(\mathbb{A}\mathrm{gt}, \mathrm{S}', \mathcal{P}rop, \pi', Act, d', o')$ where

- $\mathrm{S}' := \Lambda^{fin}_{\mathfrak{M}}(q)$,
- $d'(a, h) := d(a, last(h))$,
- $o'(h, \vec{\alpha}) := h \circ o(last(h), \vec{\alpha})$, and
- $\pi'(h) := \pi(last(h))$.

The node q in the unfolding is called *root* of $T(\mathfrak{M}, q)$.

Theorem 8.14

For every CGS $\mathfrak{M}$, state q in $\mathfrak{M}$, and ***ATL******-formula φ we have:*

$$\mathfrak{M}, q \models_{IR} \varphi \text{ iff } T(\mathfrak{M}, q), q \models_{IR} \varphi \text{ iff } T(\mathfrak{M}, q), q \models_{Ir} \varphi.$$

We now compare perfect vs. imperfect memory.

Proposition 8.15

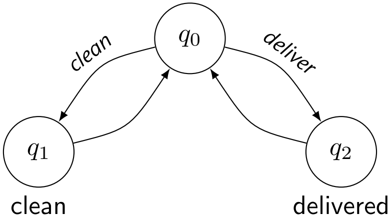
$\mathrm{ATL}^*_{\mathrm{Ir}} \subsetneq \mathrm{ATL}^*_{\mathrm{IR}}$ (In fact: $\mathbf{ATL}^+_{\mathrm{Ir}} \subsetneq \mathbf{ATL}^+_{\mathrm{IR}}$)

Membership: If $\models_{Ir} \varphi$ then $Treemodels \models_{Ir} \varphi$ then $Treemodels \models_{IR} \varphi$ then $\models_{IR} \varphi$

Strict inclusion:

$$\mathfrak{M}, q_0 \not\models_{Ir} \langle\langle a\rangle\rangle(\Diamond \mathsf{p}_1 \wedge \Diamond \mathsf{p}_2) \leftrightarrow \langle\langle a\rangle\rangle\Diamond((\mathsf{p}_1 \wedge \langle\langle a\rangle\rangle\Diamond \mathsf{p}_2) \vee (\mathsf{p}_2 \wedge \langle\langle a\rangle\rangle\Diamond \mathsf{p}_1)).$$

$\mathsf{p}_1 = \text{clean}$
$\mathsf{p}_2 = \text{delivered}$



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 $\sim_a \subseteq \mathrm{S} \times \mathrm{S}$, one per agent. The relations are assumed to be **equivalence relations**.
- We interpret $\langle\!\langle A \rangle\!\rangle$ epistemically ($\rightsquigarrow \models_{iR}$ and $\models_{ir}$)

Definition 8.16 (CEGS)

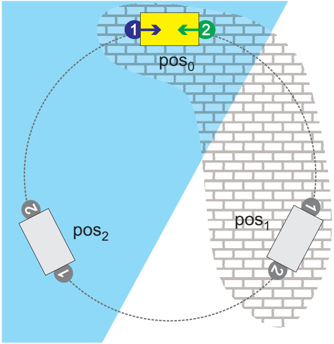
A concurrent epistemic game structure (CEGS) is a tuple

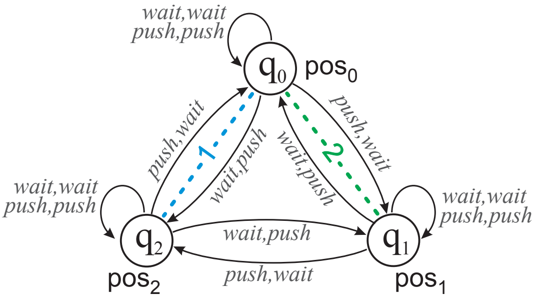
$$\mathfrak{M} = (\mathbb{A}\mathrm{gt}, \mathrm{S}, \Pi, \pi, Act, d, o, \{\sim_a \mid a \in \mathbb{A}\mathrm{gt}\})$$

with

- $(\mathbb{A}\mathrm{gt}, \mathrm{S}, \Pi, \pi, Act, d, o)$ a CGS and
- $\sim_a \subseteq \mathrm{S} \times \mathrm{S}$ equivalence relations (**indistinguishability relations**).

Example: Robots and Carriage





What about $\langle\!\langle \mathbb{A}\mathrm{gt} \rangle\!\rangle \bigcirc \mathsf{pos}_1$ in q_0?

$\mathfrak{M}, q_0 \models_{\mathit{Ir}} \langle\!\langle \mathbb{A}\mathrm{gt} \rangle\!\rangle \bigcirc \mathsf{pos}_1$

$\mathfrak{M}, q_0 \not\models_{\mathit{ir}} \langle\!\langle \mathbb{A}\mathrm{gt} \rangle\!\rangle \bigcirc \mathsf{pos}_1$

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!

$\langle\langle A \rangle\rangle \Phi$ = A can **bring about** Φ

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 $\lambda \approx_a \lambda'$ iff $\lambda[i] \sim_a \lambda'[i]$ 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

$\mathfrak{M}, q \models_{\mathsf{Ix}} \langle\langle A \rangle\rangle\varphi$ iff **there is a collective Ix-strategy** s_A such that, for each path $\lambda \in out(q, s_A)$, we have $\mathfrak{M}, \lambda \models_{\mathsf{Ix}} \varphi$.

is replaced by ($x \in \{r, R\}$ and $\sim_A := \cup_{a \in A} \sim_a$.)

$\mathfrak{M}, q \models_{\mathsf{ix}} \langle\langle A \rangle\rangle\varphi$ iff **there is a uniform ix-strategy** s_A such that, for each path $\lambda \in \bigcup_{q' : q \sim_A q'} out(q', s_A)$, we have $\mathfrak{M}, \lambda \models_{\mathsf{ix}} \varphi$

Remark 8.19

This definition models that "everybody in A knows that φ".

The fixed-point characterization does not hold anymore!

Theorem 8.20

The following formulae are not valid for ***ATL***$_{ir}$*:*

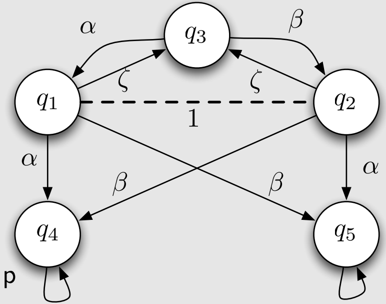
- $\langle\langle A\rangle\rangle\Box\varphi \quad\leftrightarrow\quad \varphi\wedge\langle\langle A\rangle\rangle\bigcirc\langle\langle A\rangle\rangle\Box\varphi$
- $\langle\langle A\rangle\rangle\varphi_1\,\mathcal{U}\,\varphi_2 \quad\leftrightarrow\quad \varphi_2\vee(\varphi_1\wedge\langle\langle A\rangle\rangle\bigcirc\langle\langle A\rangle\rangle\varphi_1\,\mathcal{U}\,\varphi_2)$.

Proof: $\leadsto$ Exercise.

Proof idea

We construct a counterexample for

$\langle\langle 1\rangle\rangle \Diamond \mathsf{p} \leftrightarrow \mathsf{p} \vee \langle\langle 1\rangle\rangle \bigcirc \langle\langle 1\rangle\rangle \Diamond \mathsf{p}$



$\mathfrak{M}, q_1 \not\models_{ir} \langle\langle 1\rangle\rangle \Diamond \mathsf{p}$ iff
not ($\exists s \in \Sigma_u^{ir}\, \forall \lambda \in \bigcup_{q\in\{q_1,q_2\}} out(q,s) \exists i \in \mathbb{N}_0 : \mathfrak{M}, \lambda[i] \models_{ir} \mathsf{p}$)

$\mathfrak{M}, q_1 \models_{ir} \mathsf{p} \vee \langle\langle 1\rangle\rangle \bigcirc \langle\langle 1\rangle\rangle \Diamond \mathsf{p}$

Comparing ATL$_{ir}$ vs. ATL$_{Ir}$

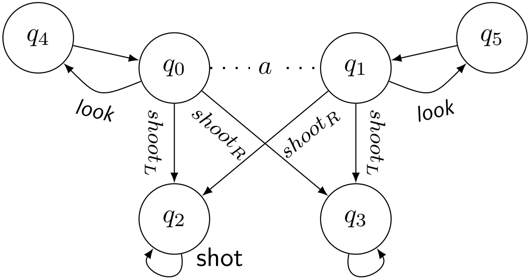
Incomplete vs. perfect information.

Proposition 8.21

$ATL_{ir} \subsetneq ATL_{Ir}$

Inclusion: Every CGS can be seen as a special CEGS

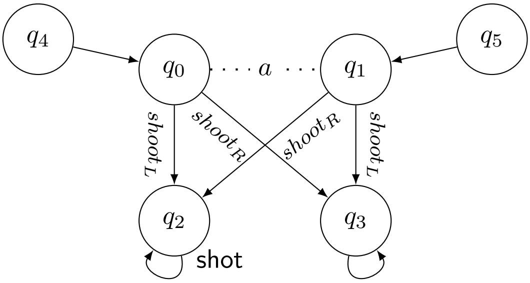
$$\mathfrak{M}, q_0 \not\models_{ir} (\mathsf{shot} \vee \langle\langle a\rangle\rangle \bigcirc \langle\langle a\rangle\rangle \Diamond \mathsf{shot}) \rightarrow \langle\langle a\rangle\rangle \Diamond \mathsf{shot}$$



Proposition 8.22

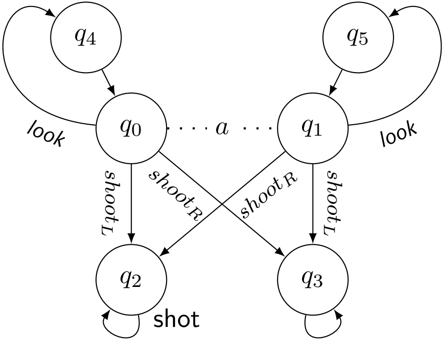
$ATL_{iR} \subsetneq ATL_{IR}$

$$\mathfrak{M}, q_4 \not\models_{iR} \langle\langle a \rangle\rangle \Diamond \mathsf{shot} \rightarrow (\mathsf{shot} \vee \langle\langle a \rangle\rangle \bigcirc \langle\langle a \rangle\rangle \Diamond \mathsf{shot})$$



iR-Tree Unfolding

The tree unfolding for the i-semantics is more sophisticated. Consider the following model and the formula $\langle\langle a\rangle\rangle \bigcirc \langle\langle a\rangle\rangle \bigcirc \langle\langle a\rangle\rangle \bigcirc \mathsf{shot}$. **How can an iR-tree unfolding look like?**



A first approach is to connect **separate unfoldings** of the indistinguishable states **by epistemic links**.

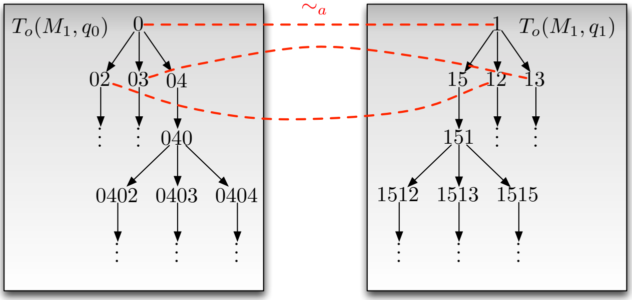
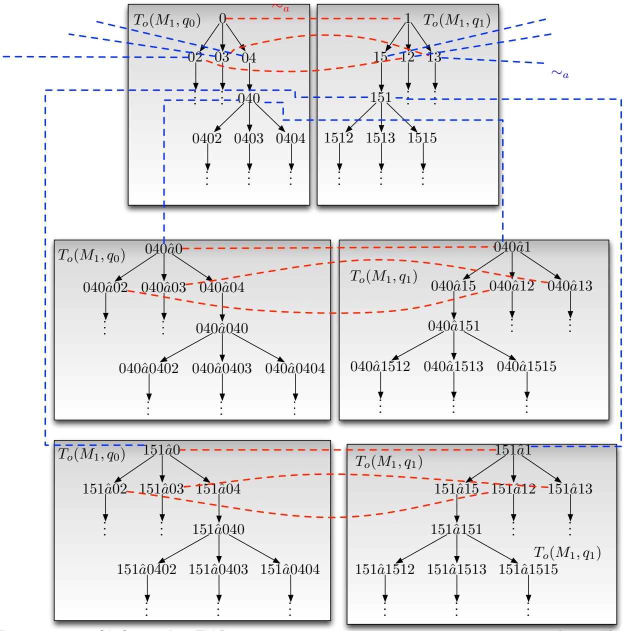


Figure 35: Two separate unfoldings connected by an epistemic link. We use number $i_1 i_2 \ldots$ to refer to the history $q_{i_1} q_{i_2} \ldots$.

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.


$\sim_a$
$T_o(M_1, q_0)$
0
02 03 04
040
0402 0403 0404
$T_o(M_1, q_1)$
1
15 12 13
151
1512 1513 1515
$\sim_a$
$T_o(M_1, q_0)$
040â0
040â02 040â03 040â04
040â040
040â0402 040â0403 040â0404
$T_o(M_1, q_1)$
040â1
040â15 040â12 040â13
040â151
040â1512 040â1513 040â1515
$T_o(M_1, q_0)$
151â0
151â02 151â03 151â04
151â040
151â0402 151â0403 151â0404
$T_o(M_1, q_1)$
151â1
151â15 151â12 151â13
151â151
151â1512 151â1513 151â1515
$T_o(M_1, q_1)$

Now we can state our main result for iR-tree unfoldings.

Theorem 8.23

For every CEGS $\mathfrak{M}$, state q in $\mathfrak{M}$, and ***ATL*****-formula φ, it holds that*

$$\mathfrak{M}, q \models_{iR} \varphi \texttt{ iff } T_s(\mathfrak{M}, q), q \models_{iR} \varphi \texttt{ iff } T_s(\mathfrak{M}, q), q \models_{ir} \varphi.$$

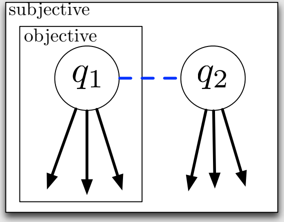
Summary

If a formula is *IR*-, or iR-satisfiable then it also is *Ir*-, or ir-satisfiable, respectively.

Objective vs. subjective ability

There is another notion of ability for imperfect information, we denote it by $\mathrm{i_o}$. What we did up to now is denoted by $\mathrm{i_s}$

- **Subjective ability ($\mathrm{i_s}$):** All paths from all indistinguishable states are taken into account.
- **Objective ability ($\mathrm{i_o}$):** Only paths from the (actual) current state are considered.



Definition 8.24 (Subjective epistemic outcome, xy-outcome)

(a) The (subjective) epistemic outcome $out^s(q, s_A)$ is defined as

$$out^s(q, s_A) = \bigcup_{q \sim_A q'} out(q', s_A).$$

(b) Let $x \in \{\mathrm{i_s}, \mathrm{i_o}, I\}$ and $y \in \{r, R\}$ The xy-outcome $out^{xy}(q, s_A)$ is defined as follows:

$$out^{xy}(q, s_A) = \begin{cases} out^s(q, s_A) & \text{if } x = \mathrm{i_s}; \\ out(q, s_A) & \text{else.} \end{cases}$$

Remark 8.25 (Strategies and semantics)

*In order to ensure a uniform notation, we introduce xy-**strategies** for $x \in \{\mathrm{i_s}, \mathrm{i_o}, I\}$ and $y \in \{r, R\}$ as follows:*

- *IR:* $s_a : \mathrm{S}^+ \to Act$ *such that* $s_a(q_0 \ldots q_n) \in d(a, q_n)$*;*
- *Ir:* *as IR with the additional constraint* $s(hq) = s(h'q)$ *for all histories* h *(or, alternatively,* $s_a : \mathrm{S} \to Act$ *such that* $s_a(q) \in d(a, q)$ *for all* q*);*
- $\mathrm{i_o}$*r*, $\mathrm{i_s}$*r*: *like Ir, with the additional constraint that* $q \sim_a q'$ **implies** $s_a(hq) = s_a(hq')$ *for all histories* h*;*
- $\mathrm{i_o}$*R*, $\mathrm{i_s}$*R*: *like IR, with the additional constraint that* $h \approx_a h'$ **implies** $s_a(h) = s_a(h')$.

Definition 8.26 (Imperfect information semantics)

$\mathfrak{M}, q \models_{xy} \langle\langle A \rangle\rangle \varphi$ *iff*

- *there is a collective* **xy-strategy** s_A
- *such that, for each path* $\lambda \in out^{xy}(q', s_A)$,
- *we have* $\mathfrak{M}, \lambda \models_{xy} \varphi$

where $x \in \{\mathrm{i_o}, \mathrm{i_s}\}$, $y \in \{r, R\}$ and $\sim_A := \cup_{a \in A} \sim_a$.

Analogously to Definition 8.27, we define the following sets:

Definition 8.27 ($\text{ATL}_{i_s x}$, $\text{ATL}^*_{i_s x}$, $\text{ATL}_{i_o x}$, $\text{ATL}^*_{i_o x}$)

We define the following **logics**:

- **ATL_{yx}** is the set of valid sentences over $(\mathcal{L}_{\text{ATL}}, \models_{yx})$
- **ATL^*_{yx}** is the set of valid sentences over $(\mathcal{L}_{\text{ATL}^*}, \models_{yx})$

where $y \in \{i_s, i_o\}$ and $x \in \{r, R\}$, respectively.

How does the picture look?

objective

subjective

perfect recall

$\mathbf{ATL}^*_{i_o\mathbf{R}}$

$\mathbf{ATL}^*_{i_s\mathbf{R}}$

language

$\mathbf{ATL}_{i_o\mathbf{R}}$

$\mathbf{ATL}^*_{\mathbf{Ir}} \neq \mathbf{ATL}^*_{\mathbf{IR}}$

$\mathbf{ATL}_{\mathbf{IR}} = \mathbf{ATL}_{\mathbf{Ir}}$

$\mathbf{ATL}_{i_s\mathbf{R}}$

$\mathbf{ATL}^*_{i_o\mathbf{r}}$

$\mathbf{ATL}^*_{i_s\mathbf{r}}$

memoryless

$\mathbf{ATL}_{i_or}$

$\mathbf{ATL}_{i_s\mathbf{r}}$

«««< .mine ======= »»»> .r185

«««< .mine

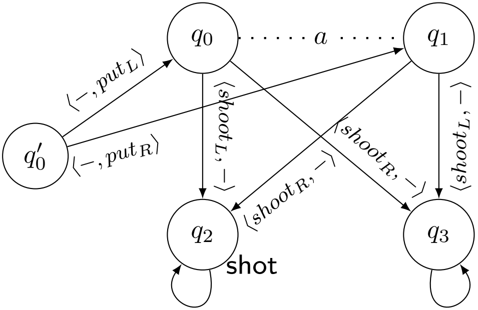
8.4 Perfect Recall Revisited

Comparing ATL$_{\mathrm{i_o r}}$ vs. ATL$_{\mathrm{Ir}}$

Proposition 8.28

$\mathrm{ATL}_{\mathrm{i_o r}} \subsetneq \mathrm{ATL}_{\mathrm{Ir}}$

$$\mathfrak{M}, q_0' \not\models_{\mathrm{i_o} r} (\mathsf{shot} \vee \langle\langle a\rangle\rangle \bigcirc \langle\langle a\rangle\rangle \Diamond \mathsf{shot}) \rightarrow \langle\langle a\rangle\rangle \Diamond \mathsf{shot}$$



======= »»»> .r185

Comparing ATL_{i_oR} vs. ATL_{IR}

Objective incomplete information vs. perfect information under perfect recall.

By the same reasoning as above:

Corollary 8.29

$ATL_{i_oR} \subsetneq ATL_{IR}$

«««< .mine =======

8.5 Perfect Recall Revisited

»»»> .r185

$\mathrm{i_o}$R-Tree Unfolding

We have already considered the tree unfolding for the $\mathrm{i_s}$-semantics above (where we did not distinguish between subjective and objective).
In the case of incomplete information for objective semantics, we only have to take into account epistemic relations in the tree:

$$h \sim_a^{T_{\mathrm{i_oR}}(\mathfrak{M},q)} h' \quad \textbf{iff} \quad h \approx_a^{\mathfrak{M}} h'$$

Theorem 8.30

For every CEGS $\mathfrak{M}$, state q in $\mathfrak{M}$, and **ATL*****-formula φ we have:*

$$\mathfrak{M}, q \models_{\mathrm{i_o}\boldsymbol{R}} \varphi \textbf{ iff } T_o(\mathfrak{M}, q), q \models_{\mathrm{i_o}\boldsymbol{R}} \varphi \textbf{ iff } T_o(\mathfrak{M}, q), q \models_{\mathrm{i_o}\boldsymbol{r}} \varphi.$$

Summary

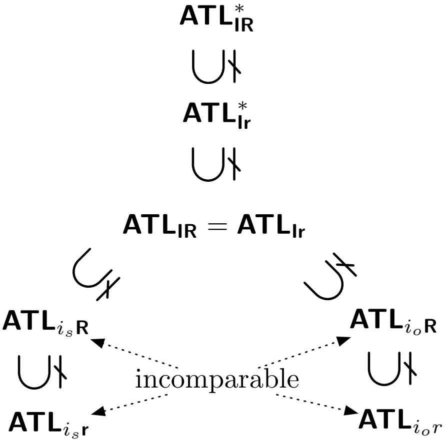
If a formula is *IR*-, $\mathrm{i_oR}$- or $\mathrm{i_sR}$-satisfiable then it also is *Ir*-, $\mathrm{i_or}$- or $\mathrm{i_sr}$-satisfiable, respectively.

Remark 8.31 (Important Validities and Invalidities)

- $\langle\langle a\rangle\rangle\Diamond \mathsf{p} \leftrightarrow \mathsf{p} \vee \langle\langle a\rangle\rangle \bigcirc \langle\langle a\rangle\rangle\Diamond \mathsf{p}$
  - Invalid *in all variants with* imperfect information*.*
  - Valid *for* perfect information*.*
- $\langle\langle a\rangle\rangle(\Diamond \mathsf{p}_1 \wedge \Diamond \mathsf{p}_2) \leftrightarrow \langle\langle a\rangle\rangle\Diamond((\mathsf{p}_1 \wedge \langle\langle a\rangle\rangle\Diamond \mathsf{p}_2) \vee (\mathsf{p}_2 \wedge \langle\langle a\rangle\rangle\Diamond \mathsf{p}_1))$
  - Invalid *for* imperfect information
  - Valid *for* perfect information *and* perfect recall
- $\neg\langle\langle\emptyset\rangle\rangle\Diamond\neg \mathsf{p} \leftrightarrow \langle\langle \mathbb{A}\mathrm{gt}\rangle\rangle\Box \mathsf{p}$
  - 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.



8.6 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.

$\langle\langle A \rangle\rangle\varphi$: Agents A have **a** collective strategy to enforce φ against **any** response of their opponents.

ATLP: Reasoning about plausible behaviors.

$\mathbf{Pl}\,\langle\langle A \rangle\rangle\varphi$: Agents A have a plausible collective strategy to enforce φ against any plausible response of their opponents.

Playing **undominated** strategies is plausible,...

The Base Logic: $\mathcal{L}_{ATLP}^{base}$

Definition 8.32 ($\mathcal{L}_{ATLP}^{base}$)

The language $\mathcal{L}_{ATLP}^{base}$ is defined over nonempty sets:

- $\mathcal{P}rop$ of **propositions**, $p \in \mathcal{P}rop$,
- $\mathbb{A}\mathrm{gt}$ of **agents**, $a \in \mathbb{A}\mathrm{gt}$, $A \subseteq \mathbb{A}\mathrm{gt}$, and
- Ω of basic plausibility terms, $\omega \in \Omega$.

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle A\rangle\rangle \bigcirc \varphi \mid \langle\langle A\rangle\rangle \Box \varphi \mid \langle\langle A\rangle\rangle \varphi \,\mathcal{U}\, \varphi \mid \mathbf{Pl}_{A}\varphi \mid (\textbf{set-pl}\ \omega)\varphi$$

The Logic $\mathcal{L}_{ATLP}^{base}$

Semantics

$\mathbf{Pl}_B$: Assuming plausible play of B

$$\mathfrak{M}, q \models \mathbf{Pl}_B \langle\langle A \rangle\rangle \gamma$$

is true iff

1. A can **enforce** γ **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:

ω_{NE}: Nash equilibria
ω_{PO} : Pareto optimal strategies

How to activate them?

(**set-pl** ω) : Sets plausible strategies to $[\![\omega]\!] \subseteq \Sigma$

And where do the terms come from?

Concurrent game structures with plausibility

$\mathfrak{M} = (\mathbb{A}\mathrm{gt}, \mathrm{S}, \mathcal{P}rop, \pi, \mathcal{A}ct, d, \delta, \boldsymbol{\Upsilon}, \boldsymbol{\Omega}, [\![\cdot]\!])$

- $\Upsilon \subseteq \Sigma$: set of (plausible) strategy profiles

  Example: $\Upsilon = \{(head, head)\}$

- $\Omega = \{\omega_1, \omega_2, \dots\}$: set of plausibility terms

  Example: ω_{NE} stands for all Nash equilibria

- $[\![\cdot]\!] : \mathrm{S} \to (\Omega \to \mathcal{P}(\Sigma))$: plausibility mapping, it assigns a set of strategy profiles to each state and plausibility term

  Example: $[\![\omega_{NE}]\!]_q = \{(head, head), (tail, tail)\}$

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

$$\Sigma_A(P) := \{s_A \in \Sigma_A \mid \exists t \in P \quad (t[A] = s_A)\}$$

$P(s_A)$: plausible strategy profiles of $\mathbb{A}\mathrm{gt}$ 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.

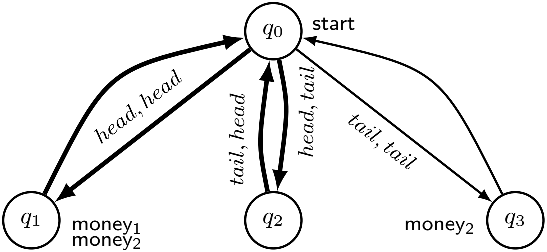
Outcome of a strategy

Outcome = **Paths** that may occur when agents A perform s_A

$$out(q, s_A, P) =$$

$$\{q_0 q_1 \ldots \in \mathrm{S}^+ \mid q = q_0 \wedge \exists \mathbf{t} \in \mathbf{P}(\mathbf{s_A})\ \forall i \in \mathbb{N}\ \big(q_{i+1} = \delta(\mathbf{q_i}, \mathbf{t}(\mathbf{q_i}))\big)\}$$

The outcome is given wrt to a set of (plausible) strategy profiles P, **restricting the opponents choices**!



$$out(q_0, head_1, \Sigma) = \{q_0q_2q_0q_2\ldots, q_0q_1q_0q_1\ldots\}$$

Semantics of $\mathcal{L}_{ATLP}$

We use a satisfaction relation $\models_P$ annotated with a set of strategy profiles.

P: strategies currently considered available

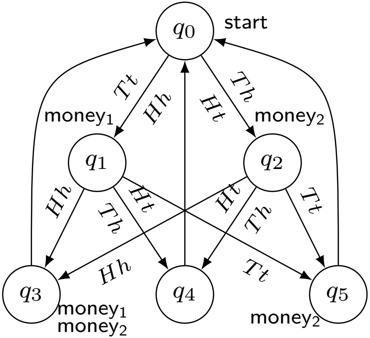
$\mathfrak{M}, q \models_P \langle\langle A \rangle\rangle \bigcirc \varphi$ iff $\exists \mathbf{s_A} \in \mathbf{\Sigma_A}(\mathbf{P}) \quad \forall \lambda \in out(q, s_A, \mathbf{P})$
$\mathfrak{M}, \lambda[1] \models_P \varphi$

$\mathfrak{M}, q \models_P \mathbf{Pl}\, \varphi$ iff $\mathfrak{M}, q \models_{\mathbf{\Upsilon}} \varphi$

$\mathfrak{M}, q \models_P \mathbf{Ph}\, \varphi$ iff $\mathfrak{M}, q \models_{\mathbf{\Sigma}} \varphi$

$\mathfrak{M}, q \models_P$ (**set-pl** $\omega)\varphi$ iff $\mathfrak{M}^\omega, q \models_P \varphi$ where the new model $\mathfrak{M}^\omega$ is equal to $\mathfrak{M}$ but the “new” set Υ^ω **of plausible strategy profiles is set to** $[\![\omega]\!]_q$.

Example: A Penny 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 ω:

$$\sigma.\forall\sigma_1\exists\sigma_2\ldots\forall\sigma_n \qquad \underbrace{\varphi(\sigma,\sigma_1,\ldots\sigma_n)}_{\text{Property that } \sigma \text{ should fulfill} \in \mathcal{L}_{ATLP}^{base}(\Omega\cup\{\sigma,\sigma_1,\ldots,\sigma_n\})}$$

Example: $\omega_{\mathsf{DOM}} = \sigma.\forall\sigma' \quad (\sigma$ better than $\sigma')$

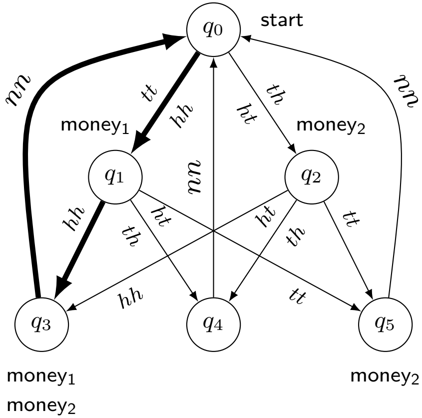
How to determine whether a strategy is good?

General Solution Concepts

Agents have preferences: $\vec{\eta} = \langle \eta_1, \dots, \eta_k \rangle$

η_i: **ATL** path formulas

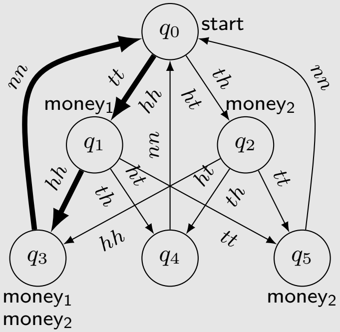
Example: $\eta_2 = \Diamond \mathsf{money}_2$



No payoffs needed as for classical solution concepts!

CGSP +preferences $\rightsquigarrow$ Normal form game

Each CGSP with preferences corresponds to a normal form game.



$\rightsquigarrow$

$\eta_1 \backslash \eta_2$	s_{hh}	s_{ht}	s_{th}	s_{tt}
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

Characterizing Solution Concepts

$$
\begin{aligned}
BR_a^{\vec{\eta}}(\sigma) &\equiv (\textbf{set-pl}\ \sigma[\mathbb{A}\mathrm{gt}\setminus\{\mathbf{a}\}])\mathbf{Pl}\left(\langle\langle a\rangle\rangle\eta_a \rightarrow (\textbf{set-pl}\ \sigma)\langle\langle\emptyset\rangle\rangle\eta_a\right)\\
NE^{\vec{\eta}}(\sigma) &\equiv \bigwedge_{a\in\mathbb{A}\mathrm{gt}} BR_a^{\vec{\eta}}(\sigma)\\
SPN^{\vec{\eta}}(\sigma) &\equiv \langle\langle\emptyset\rangle\rangle\Box NE^{\vec{\eta}}(\sigma)
\end{aligned}
$$

$$
\begin{aligned}
PO^{\vec{\eta}}(\sigma) \equiv\ & \forall\sigma'\ \mathbf{Pl}\Big(\bigwedge_{a\in\mathbb{A}\mathrm{gt}} ((\textbf{set-pl}\ \sigma')\langle\langle\emptyset\rangle\rangle\eta_a \rightarrow (\textbf{set-pl}\ \sigma)\langle\langle\emptyset\rangle\rangle\eta_a) \vee\\
& \bigvee_{a\in\mathbb{A}\mathrm{gt}} ((\textbf{set-pl}\ \sigma)\langle\langle\emptyset\rangle\rangle\eta_a \wedge \neg(\textbf{set-pl}\ \sigma')\langle\langle\emptyset\rangle\rangle\eta_a\Big).
\end{aligned}
$$

Characterizing Solution Concepts (2)

Theorem

All these characterizations correspond to their game-theoretical counterparts.

Example

All plays:

$$\mathfrak{M}, q_0 \models \neg\langle\langle a_2\rangle\rangle \bigcirc \text{money}_2$$

Both agents play a **Nash equilibrium** strategy:

$$M, q_0 \models (\textbf{set-pl}\ \sigma.\mathrm{NE}^{\eta}(\sigma))\mathbf{Pl}\,\langle\langle a_2\rangle\rangle \bigcirc \text{money}_2$$

ATLP with ATLI based plausibility specifications

Remark

We can also define quantitative temporalized versions: BR_a^T, NE^T, SPN^T, where $T = \bigcirc, \Box, \Diamond$ and states are labeled with propositions which represent payoffs.

We then have the following theorem:

Theorem 8.33

Let Γ be an extensive game (with a finite set of proposition), and $\mathfrak{M}(\Gamma)$ a CGS corresponding to Γ. Then $\mathfrak{M}(\Gamma), \varnothing \models NE^{\Diamond}(\sigma)$ (resp. $SPN^{\Diamond}(\sigma)$) iff σ is NE (resp. a subgame-perfect NE) in Γ.

The Full Language: $\mathcal{L}_{ATLP}$

Plausibility terms:

$$\sigma.\forall\sigma_1\exists\sigma_2\dots\forall\sigma_n \quad \varphi$$

where

$$\varphi \in \mathcal{L}_{ATLP}^{base}$$

➡ What about nesting (**set-pl** $\cdot$) operators?

(**set-pl** ...(**set-pl** ...(**set-pl** ...)...)...)

➡ We get a hierarchy of logics:

$\mathcal{L}_{ATLP}^{k}$: k nestings

$\mathcal{L}_{ATLP} := \lim_{k\to\infty} \mathcal{L}_{ATLP}^{k}$

8.7 References

Alur, R., Henzinger, T. A., and Kupferman, O. (1997).
Alternating-time Temporal Logic.
In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 100–109. IEEE Computer Society Press.

R. Alur, T. A. Henzinger, and O. Kupferman (2002).
Alternating-time Temporal Logic.
Journal of the ACM, 49:672–713.

Bulling, N., Dix, J., and Jamroga, W. (2010).
Model checking logics of strategic ability: Complexity.
In Dastani, M., Hindriks, K. V., and Meyer, J.-J. C., editors, *Specification and Verification of Multi-Agent Systems*. Springer.

Bulling, N., Dix, J., and Jamroga, W. (2008).
Reasoning about temporal properties of rational play.
Annals of mathematics and Artificial Intelligence 53(1–4)*51-114, 2008.

Wojtek Jamroga and Jürgen Dix. (2008)
Model Checking Abilities of Agents: A closer look.
Journal of Computing Systems, **42(3)**, 366–410.

Schobbens, P. Y. (2004).
Alternating-time logic with imperfect recall.
Electronic Notes in Theoretical Computer Science, 85(2):82–93.