Algorithmic Game Theory

Lecture Notes

based on a lecture by Prof. F. Brandt

Marvin Jahn
mail@marvin-jahn.de

July 26, 2021

Contents

These are (unofficial) lecture notes for the lecture *Algorithmic Game Theory* held by Prof. F. Brandt at the Technical University Munich in the summer semester 2021.

Some proofs, many examples and some complexity-theoretic results (hardness results) are omitted.

1 Preference Relations and Lotteries

Lec 1
2021-04-13

We start with the following informal definition.

Definition 1.1. An **agent** A is an entity in game theory able to make decisions. The set of possible decisions the agent can make is denoted by $\mathcal{A}(A)$.

The preferences of an agent between the different alternatives is modeled by *preference relations*.

Definition 1.2. The **(binary) preference relation** on the set of alternatives $\mathcal{A}$ of some agent is a binary relation, where $x \geq y$ is interpreted to mean that the agent preferes x over y.

Note that any binary relation can be decomposed into a **asymmetric relation** and a **symmetric relation**:

$$B = \{(x,y) \in B : (y,x) \notin B\} \sqcup \{(x,y) \in B : (y,x) \in B\}.$$

In the case of preference relations, the asymmetric part corresponds to *strict preference* and the symmetric part corresponds to *indifference*.
In the following, we write $\leq$ for the preference relation, $<$ for its asymmetric part and $\sim$ for its symmetric part.

Definition 1.3. A preference relation is called **rational** if it is a complete preorder (i.e. transitive and complete).

Lemma 1.4. A transitive relation on a nonempty finite set S admits a maximal element.

Proof. This is equivalent to the fact that there is no infinite path in a finite directed graph.
Alternatively: Take an arbitrary $x_0 \in S$ and form a chain $x_0 \leq x_1 \leq \cdots \leq x_n$, such that there is no $x_{n+1} \in S \setminus \{x_0, \dots, x_n\}$ with $x_n \leq x_{n+1}$. Then x_n is a maximal element. □

Another way to represent the preferences of an agent are *utility functions*.

Definition 1.5. A **utility function** is a function $u : \mathcal{A} \to \mathbb{R}$, where $u(x) \geq u(y)$ is interpreted to mean that the agent prefers x over y.

Given a utility function $u : \mathcal{A} \to \mathbb{R}$, the corresponding preference relation is given by

$$x \leq y \ :\Longleftrightarrow\ u(x) \leq u(y).$$

On the other hand, a given preference relation may correspond to no or many utility functions, since given a corresponding utility function u, we can derive many others by composition $f \circ u$ with some strictly increasing function $f : \mathbb{R} \to \mathbb{R}$.

Proposition 1.6. If the set $\mathcal{A}$ of alternatives is countable, then a preference relation $\leq$ permits a corresponding utility function if and only if $\leq$ is rational.

Proof. Clearly $\leq$ must be rational, because $(\mathbb{R}, \leq)$ is a totally ordered set.
On the other hand, suppose that $\leq$ is rational. The equivalence relation

$$x \sim y \ :\Longleftrightarrow\ x \leq y \land y \leq x$$

on $\mathcal{A}$ is compatible with $\leq$, so $\leq$ descends to $\mathcal{A}/\sim$. By construction, $\leq$ becomes a total order on $\mathcal{A}/\sim$. Thus there is a countable chain containing all elements of $\mathcal{A}/\sim$, which can be identified in an order-preserving way with a subset of $\mathbb{Z}$; i.e. there exists an order-preserving map $\phi\colon \mathcal{A}/\sim \to \mathbb{Z}$. Writing $\pi\colon \mathcal{A} \twoheadrightarrow \mathcal{A}/\sim$ for the projection and $i\colon \mathbb{Z} \hookrightarrow \mathbb{R}$ for the inclusion, a utility function of $\leq$ is given by the composition $\psi : i \circ \phi \circ \pi\colon \mathcal{A} \to \mathbb{R}$. □

Example 1.7. The lexicographical ordering on $[0, \infty) \times [0, \infty)$ cannot be represented by a utility function.

Lec 2
2021-04-20

In the real world, the consequences of actions are often best described using probabilities. Instead of knowing that a certain alternative $a \in \mathcal{A}$ will happen if we make some decision, we usually only know that it increases the probability that a occurs. Thus we need to generalize our notion of alternatives.

Definition 1.8. A **(simple) lottery** is a probability distribution over the set of alternatives $\mathcal{A}$. If $\mathcal{A} = \{a_1, \ldots, a_n\}$ is finite, it can be represented as a probability vector $p \in \mathbb{R}^n$. A lottery is often also written as $[p_1 : a_1, \ldots, p_n : a_n]$.
The set of all lotteries over a finite set of alternatives $\mathcal{A} = \{a_1, \ldots, a_n\}$ is thus the set of all probability distributions over $\mathcal{A}$, denoted by

$$\mathcal{L}(a_1, \ldots, a_n) = \{p \in \mathbb{R}^n \text{ probability vector}\}.$$

For convenience, lotteries may instead be a probability distribution on the set of all simple lotteries $\mathcal{L}$. Such a lottery is called **compound lottery**. By multiplying probabilities, a compound lottery can be translated into a simple one.
On the other hand, every alternative $a \in \mathcal{A}$ gives rise to a canonical lottery $[a : 1]$ (also usually denoted by a), called **degenerate lottery**.
Therefore, the notions of simple lottery and compound lottery are equivalent.

In other words, any lottery makes $\mathcal{A}$ into a probability space and $u : \mathcal{A} \to \mathbb{R}$ is a random variable.
Whenever randomness is involved, instead of having a preference on the alternatives $\mathcal{A}$, an agent should have a preference on the set of lotteries $\mathcal{L}$. Thus we extend Definition 1.2 as follows.

Definition 1.9. A **(binary) preference relation** on the set of lotteries $\mathcal{L}$ of some agent is a binary relation. It is called **rational** if it constitutes a complete preorder.

Given the preference relation of some agent on $\mathcal{A}$, it is not clear how to construct a preference relation on $\mathcal{L}$ from that. For example, the agent might prefer the lottery for which the most likely outcome is preferred to the others.

To judge which preference relations are "good", we need to introduce additional terminology.

Definition 1.10. A preference relation $\geq$ on lotteries is called **continuous**, if for all $L_1, L_2, L_3 \in \mathcal{L}$, $L_1 > L_2 > L_3$, there is some $\epsilon \in (0, 1)$, such that

$$[(1-\epsilon) : L_1, \epsilon : L_3] > L_2 > [(1-\epsilon) : L_3, \epsilon : L_1].$$

Definition 1.11. A preference relation $\geq$ on lotteries is called **independent**, if for all $L_1, L_2, L_3 \in \mathcal{L}$ and $p \in (0,1)$, we have

$$L_1 \geq L_2 \iff [p : L_1, (1-p) : L_3] \geq [p : L_2, (1-p) : L_3].$$

Von Neumann and Morgenstern showed the following advanced theorem in 1947.

Theorem 1.12. Let $\mathcal{A}$ be a set of alternatives and $\mathcal{L}$ the set of lotteries. A preference relation $\geq$ on $\mathcal{L}$ is rational, continuous and independent if and only if there exists a utility function $u : \mathcal{A} \to \mathbb{R}$, such that for any two lotteries $L_1 = [p_1 : x_1, \dots, p_n : x_n]$ and $L_2 = [q_1 : x_1, \dots, q_n : x_n]$, we have

$$L_1 \geq L_2 \iff \sum_{i=1}^{n} p_i u(x_i) \geq \sum_{i=1}^{n} q_i u(x_i).$$

Such a utility function u is called **vNM**.

Note that the composition $f \circ u$ of any vNM utility function u with any positive affine transformation $f(x) = ax + b$, $a > 0$ is another vNM utility function.

2 Solution Concepts of Normal-Form Games

2.1 Normal-Form Games

Lec 3
2021-04-27

We want to formally describe a large class of games.

Definition 2.1. A **normal-form game** is a finite set of *players* $\{1, \dots, n\}$ together with a finite set of *actions* A_i for each player and a *utility* (or *payoff*) function $u \colon A \to \mathbb{R}^n$, where $A \coloneqq \prod_{i=1}^{n} A_i$ is the set of *action profiles*. The *utility function* of a player i is simply $u_i \coloneqq \pi_i \circ u \colon A \to \mathbb{R}$. For an action $a \in A$, the vector $u_i(a)$ is called the *outcome* for that action.
We write $A_{-i} \coloneqq A_1 \times \dots \times A_{i-1} \times A_{i+1} \times \dots \times A_n$ for the set of all actions from all players but player i and similarly A_{-C} for sets of players C.

In order to characterize which outcomes are optimal, an obvious strategy would be to define a partial order on the set of all outcomes and consider its maximal elements. One example for this is *Pareto-dominance*, which compares outcomes by their "social" impact.

Definition 2.2. As a subset of $\mathbb{R}^n$ the set of outcomes inherits a partial order (i.e. component-wise comparison). An outcome is called **(strongly) Pareto-optimal** if it is a maximal element with respect to this order and **(weakly) Pareto-dominated** otherwise.
In other words, it is **Pareto-dominated**, if there exists another outcome in which all players obtain at least as much utility and one player increases their utility. It is called **strongly Pareto-dominated**, if there exists another outcome in which all players increase their utility; otherwise it is called **weakly Pareto-optimal**.

Clearly, a strongly Pareto-dominated outcome is also weakly Pareto-dominated and any strongly Pareto-optimal outcome is also weakly Pareto-optimal.
If an outcome is Pareto-optimal, it is impossible to increase the utility of a player without

reducing the utility of another. Of course, these notions do not change when using an equivalent utility function. Any finite game must contain a Pareto-optimal outcome. Pareto properties only deal with the outcomes, not actions. In particular, they say nothing about strategies, since they completely ignore the actions of the players.

Instead of looking at the "global" welfare, we now want to focus on a single player i and determine which of their actions are better than others.

Definition 2.3. For a player i, we have the function $A_i \to (A_{-i} \to \mathbb{R})$, assigning to each possible action the function returning the resulting utility of i. Therefore, in order to specify a relation (in particular, a partial order) on A_i, it is enough to define one on the set $\{A_{-i} \to \mathbb{R}\}$. We can view $\{A_{-i} \to \mathbb{R}\}$ as a product and thus get a product order $\leq$. An action $a_i \in A_i$ **very weakly dominates** $b_i \in A_i$, if $a_i \geq b_i$ with respect to the induced partial order and it **weakly dominates** $b_i \in A_i$, if $a_i > b_i$ with respect to the asymmetric part of the induced partial order. We may also use the asymmetric part of the product order to induce an asymmetric order on A_i. We say that $a_i \in A_i$ **(strictly) dominates** $b_i \in A_i$, if $a_i > b_i$ with respect to the induced asymmetric order.
It is called **(...) dominant**, if it is a greatest element and it is called **(...) dominated**, if it is no maximal element with respect to one of the previously defined relations.
The outcome that arises when all players pick a dominant action (assuming such actions exist) is called **dominant strategy outcome**.

Definition 2.4. A **mixed strategy** $s_i \in S_i := \mathcal{L}(A_i)$ is a lottery (probability distribution) over actions. The action a_i is played with probability $s_i(a_i)$.
A degenerate lottery is just an action and is called **pure strategy**.
For a given strategy profile $s \in S = S_1 \times \cdots \times S_n$, the set of action profiles A becomes a probability space and $u_i : A \to \mathbb{R}$ becomes a random variable. The **expected utility** of player i in a given strategy profile $s \in S = S_1 \times \cdots \times S_n$ is the expected value of u_i; that is,

$$u_i(s) := \mathbb{E}[u_i] = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j).$$

We will always assume that our utility functions are vNM, so that we may assume that the agents want to maximize their expected payoff.

For any player i, the set of their strategies S_i is by definition the convex hull of A_i. In particular, we can form convex combinations of the strategies. For strategies $s_1, \ldots, s_l \in S_i$ and a probability vector $x \in \mathbb{R}^l$, we write $\sum_{i=1}^{l} x_i s_i$ for their convex combination $\{x_1 : s_1, \ldots, x_l : s_l\}$.

Lemma 2.5. In any normal-form game with n players and for any player i, the function

$$u\colon S_i \times S_{-i} \to \mathbb{R}^n$$

is "linear" in S_i; that is, for any $s_i, t_i \in S_i$ we have

$$p \cdot u(s_i, -) + (1-p) \cdot u(t_i, -) = u(ps_i + (1-p)t_i, -).$$

Equivalently, we have

$$p \cdot u_j(s_i, -) + (1-p) \cdot u_j(t_i, -) = u_j(ps_i + (1-p)t_i, -)$$

for all players i, j and any $s_i, t_i \in S_i$.

Proof. Since $u_i = \pi_i \circ u$, the claimed equivalence is clear. Because for $s_{-i} \in S_{-i}$, we have

$$\begin{aligned} u_j(s_i, s_{-i}) &= \sum_{a \in A} u_j(a) s_i(a_i) \prod_{k \in \{1,\dots,n\} \setminus \{i\}} s_k(a_k) \\ &= \sum_{a_{-i} \in A_{-i}} \left(\prod_{k \in \{1,\dots,n\} \setminus \{i\}} s_k(a_k) \right) \sum_{a_i \in A_i} u_j(a_i, a_{-i}) s_i(a_i), \end{aligned}$$

and conclude

$$\begin{aligned} & p \cdot u_j(s_i, s_{-i}) + (1-p) \cdot u_j(t_i, s_{-i}) \\ &= \sum_{a_{-i} \in A_{-i}} \left(\prod_{k \in \{1,\dots,n\} \setminus \{i\}} s_k(a_k) \right) \sum_{a_i \in A_i} u_j(a_i, a_{-i}) (p \cdot s_i(a_i) + (1-p) \cdot t_i(a_i)) \\ &= \sum_{a_{-i} \in A_{-i}} \left(\prod_{k \in \{1,\dots,n\} \setminus \{i\}} s_k(a_k) \right) \sum_{a_i \in A_i} u_j(a_i, a_{-i}) (p s_i + (1-p) t_i)(a_i) \\ &= u_j(p s_i + (1-p) t_i, s_{-i}). \end{aligned}$$

□

Of course, the lemma implies the statement for any (finite) probability vector x instead of just for $x = (p, 1-p) \in \mathbb{R}^2$.

Our notions of dominance from Definition 2.3 directly extend from actions to mixed strategies. It is important to realize that when checking our various notions of dominance for strategies, it is enough to consider pure strategies of the opponents, as the following lemma shows.

Lemma 2.6. *Let s_i and t_i be two strategies of player i. Then s_i dominates t_i if and only if $u_i(s_i, a_{-i}) > u_i(t_i, a_{-i})$ for all $a_{-i} \in A_{-i}$.*
The analogous statement is true for weak and very weak dominance.

Proof. We only have to show that if $u_i(s_i, a_{-i}) > u_i(t_i, a_{-i})$ for all $a_{-i} \in A_{-i}$, then $u_i(s_i, s_{-i}) > u_i(t_i, s_{-i})$ for all $s_{-i} \in S_{-i}$. Because

$$u_i(s_i, a_{-i}) = \sum_{a_i \in A_i} u_i(a_i, a_{-i}) s_i(a_i),$$

it follows

$$\begin{aligned} u_i(s_i, s_{-i}) &= \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j) \\ &= \sum_{a_{-i} \in A_{-i}} \left(\prod_{j \in \{1,\dots,n\} \setminus \{i\}} s_j(a_j) \right) \sum_{a_i \in A_i} u_i(a_i, a_{-i}) s_i(a_i) \\ &> \sum_{a_{-i} \in A_{-i}} \left(\prod_{j \in \{1,\dots,n\} \setminus \{i\}} s_j(a_j) \right) \sum_{a_i \in A_i} u_i(a_i, a_{-i}) t_i(a_i) \\ &= \sum_{a \in A} u_i(a) s_i(a_i) \prod_{j \in \{1,\dots,n\} \setminus \{i\}} s_j(a_j) = u_i(t_i, s_{-i}). \end{aligned}$$

The same argument works for weak and very weak dominance. □

Definition 2.7. A strategy s_i is called **best response** to the strategy profile s_{-i}, if $u_i(s_i, s_{-i}) \geq u(t_i, s_{-i})$ for all $t_i \in S_i$. The set of all best responses is denoted by $B_i(s_{-i})$.

Notice that any convex combination of best responses is another best response and that there always exists a best response that is a pure strategy.

Theorem 2.8. If a strategy s_i is dominated, then it is never a best response. Furthermore, the other direction is true for two-player game.

We also make the following (inprecise) definition.

Definition 2.9. An action is called **rationalizable** if a rational player could justifiably play it against rational opponents when everyone knows that everyone plays rationally.

Intuitively, a rational player should never play dominated actions. This intuition agrees with the previous definition, as Pearce and Bernheim separately showed in 1984.

Theorem 2.10. In two-player games, rationalizable actions are precisely those actions that survive the iterated elimination of dominated actions.

Of course, every game has one or more rationalizable action.

We note an alternative characterization of dominated and weakly dominated actions.

Theorem 2.11. (a) An action is dominated if and only if it is never a best response.

(b) An action is weakly dominated if and only if it is never a best response to a full-support strategy.

2.2 Iterated Dominance

Lec 4 2021-05-04

Definition 2.12. A game can be **solved via iterated strict dominance** (**ISD**), if only a single action profile survives the iterated elimination of dominated actions.
It can be **solved via iterated weak dominance** (**IWD**), if only a single action profile survives the iterated elimination of weakly dominated actions.

A two-player game can be solved via iterated strict dominance if and only if both players only have a single rationalizable action.
Note that ISD is order independent and its solution (if existent) is unique, whereas IWD is order dependent and may have multiple solutions.

Theorem 2.13. Deciding whether a game can be solved via ISD is computable in polynomial time (using a linear program), whereas deciding the same for IWD is NP-complete.

We make the following informal definition.

Definition 2.14. A **solution concept** is a function mapping a game to a (possibly empty) set of strategy profiles. These strategy profiles are the ones recommended by the solution concept.

It is reasonable to assume that a solution concept should be invariant under positive affine transformations.

Example 2.15. ISD and IWD are solution concepts.

2.3 Maximin Strategies

Definition 2.16. The set of **maximin stategies** of player i is given by

$$\text{argmax}_{s_i \in S_i} \min_{s_{-i} \in S_{-i}} u_i(s_i, s_{-i}).$$

The **security level** of player i is the corresponding maximum

$$\max_{s_i \in S_i} \min_{s_{-i} \in S_{-i}} u_i(s_i, s_{-i}).$$

It is the highest utility player i can achieve, no matter what the other players do.

This strategy tries to maximize the worst-case outcome of the game. It again suffices to consider pure strategies of the opponents, as the following lemma shows.

Lemma 2.17. The set of maximin strategies of player i is

$$\text{argmax}_{s_i \in S_i} \min_{a_{-i} \in A_{-i}} u_i(s_i, a_{-i}).$$

Proof. We have to show that $\min_{s_{-i} \in S_{-i}} u_i(s_i, s_{-i}) = \min_{a_{-i} \in A_{-i}} u_i(s_i, a_{-i})$ for all $s_i \in S_i$. The inequality "$\geq$" holds by definition. For the other inequality, let $t_{-i} \in S_{-i}$ be a strategy profile attaining the minimum. For any player $j \neq i$, we may consider a game with modified payoffs. Namely, j's payoff for the action profile $a \in A$ is defined to be $-u_i(a)$. Because

$$t_j \in \text{argmin}_{s_j \in S_j} u_i(s_i, s_j, t_{-\{i,j\}}) = \text{argmax}_{s_j \in S_j} -u_i(s_i, s_j, t_{-\{i,j\}}),$$

it follows that t_j is a best response to $s_i \cup t_{-\{i,j\}}$ in the modified game and thus t_j can be chosen to be pure. Hence, t_j can be replaced by a pure strategy a_j that satisfies $u_i(s_i, a_j, t_{-\{i,j\}}) = u_i(s_i, s_j, t_{-\{i,j\}})$, so $a_j \cup t_{-\{i,j\}} \in S_{-i}$ still attains the minimum $\min_{s_{-i} \in S_{-i}} u_i(s_i, s_{-i})$. By iterating this argument, we can construct a collection of pure strategies $a_{-i} \in A_{-i}$ that attains the minimum and thus the claim follows. □

Lemma 2.18. Convex combinations of maximin strategies are again maximin strategies.

Proof. This is a direct consequence of Lemma 2.5, since for $s_i = \sum_{a_i \in A_i} s_i(a_i) a_i$ and $t_i = \sum_{a_i \in A_i} t_i(a_i) a_i$ two maximin strategies with security level c, we have

$$\min_{s_{-i} \in S_{-i}} u_i(ps_i + (1-p)t_i, s_{-i}) = p \cdot \min_{s_{-i} \in S_{-i}} u_i(s_i, s_{-i}) + (1-p) \cdot \min_{s_{-i} \in S_{-i}} u_i(t_i, s_{-i}) = c.$$

□

Theorem 2.19. Using a linear program, maximin strategies and the corresponding security levels can be computed in polynomial time.

2.4 Nash Equilibria

Definition 2.20. A strategy profile $s = (s_1, \ldots, s_n)$ is a **Nash equilibrium** if for all players i and all strategies $t_i \in S_i$, we have

$$u_i(s_i, s_{-i}) \geq u_i(t_i, s_{-i}).$$

A **pure Nash equilibrium** is a Nash equilibrium consisting only of pure strategies.

Intuitively, a Nash equilibrium can be thought of as a steady state of mutual best responses. Equivalently, every strategy s_i is a best response to the strategies of all others. Unlike for maximin strategies, the set of Nash equilibria is generally not convex. Lec 5 2021-05-11

Lemma 2.21 (Indifference principle). A strategy profile s is a Nash equilibrium if and only if for every player i the following holds true, assuming that the other players play s_{-i}:

(a) All actions in the support of s_i yield the same expected payoff.

(b) No action outside the support of s_i yields more expected payoff.

In particular, any randomization of player i among actions in the support of s_i yield the same expected payoff. Informally, one could describe this as "a player randomizes for the other players"; if they change their randomization, then the best response of other players might change.
The indifference principle is convenient for determining Nash equilibria.

Proposition 2.22. The following two statements hold for any normal-form game.

(a) Only rationalizable actions can be in the support of an equilibrium.

(b) The payoff in any equilibrium is always at least as large as the player's security level.

Proof. (a) An action that is not rationalizable is never a best response.

(b) If there was a player for which the payoff in an equilibrium was strictly less than the player's security level, then that player could just play their maximin strategy in order to achieve a higher payoff.

□

Nash equilibria always exist, which was shown by Nash in 1950.

Theorem 2.23 (Existence of Nash equilibria). Every normal-form game contains a Nash equilibrium.

We define three axioms that uniquely characterize Nash equilibria. Lec 6 2021-05-18

Definition 2.24. A solution concept may satisfy some of the following axioms:

(a) **Utility maximization**: In a one player game, only expected utility-maximizing strategies are chosen.

(b) **Consistency**: For a n-player game let s be the solution of the solution concept. For any $k \in \{1, \dots, n\}$, fixing the strategy of k of the players to the ones recommended by the solution concept gives rise to a $(n-k)$-player game. Then the remaining strategies from s should be a solution of this $(n-k)$-player game with respect to the solution concept.

(c) **Existence**: Every game has at least one solution.

Norde et al. showed in 1996:

Theorem 2.25. The Nash equilibrium is the only solution concept that (simultaneously) satisfies utility maximization, consistency and existence.

While the proof of that theorem is rather involved, the following weaker statement is surprisingly simple to prove.

Theorem 2.26. A solution concept that satisfies utility maximization and consistency maps a game to a subset of the set of Nash equilibria of that game. In other words, any solution returned by such a solution concept is necessarily a Nash equilibrium.

We now study the question if and how one can efficiently calculate Nash equilibria. By simply checking each outcome, one can compute any pure Nash equilibrium in polynomial time (in the size of the game), provided one exists.
However, the size of a normal-form game is exponential in the number of players, so while the algorithm needs polynomial time, this is in terms of the size of the game, which is exponential. Thus in that case more efficient representations of normal-form games are required to handle large games efficiently.

Fictitious play is an algorithm used to find Nash equilibria. It was proposed by Brown in 1951.

Definition 2.27. Given a game G, **fictitious play** is a "simulation" of the game in the following way:

1. In the first round, every player arbitrarily chooses an action.

2. In subsequent rounds, each player simultaneously plays a pure best response to the strategy profile given by the empirical distribution of their opponent in the previous rounds.

Fictitious play can be understood as a learning procedure of two players. It is also remarkable that in order to apply this method, each player only has to know their own payoffs and not the payoffs of their opponents.

We summarize some statements of the convergence of fictitious play.

Theorem 2.28. If fictitious play converges, it converges to a Nash equilibrium.
In zero-sum games, $2 \times k$ games and games solvable by ISD, fictitious play converges.

Definition 2.29. A two-player game is called **degenerate**, if there exists a strategy $s_{-i} \in S_{-i}$ and a best response $s_i \in S_i$ to it, such that the best response has strictly larger support than s; i.e. $|\text{supp}(s_i)| > |\text{supp}(s_{-i})|$.

Wilson showed the following theorem in 1971.

Theorem 2.30. In a non-degenerate two-player game, the number of Nash equilibria is finite and odd.

In particular, in a non-degenerate game two-player game, all Nash equilibria must have same sized support for both players.
But just because all Nash equilibria have the same size, we do not necessarily have a non-degenerate game, as the following example shows.

Example 2.31. The game

$(1,1)$	$(1,2)$
$(0,0)$	$(0,0)$

has a unique Nash equilibrium with outcome $(1,2)$, but any convex combination of the rows is a best response to the second row.

Finding all Nash equlibria of a normal form game requires exponential time in the worst case, because the number of Nash equilibria can be exponential in the number of total actions.

A useful algorithm to determine a Nash equilibrium of a two-player game is the **support enumeration algorithm**.

Theorem 2.32. The problem of finding a Nash equilibrium is PPAD-complete.

2.5 Correlated Equilibria

We mention a generalization of Nash equilibria.

Definition 2.33. A **correlated strategy profile** in a normal-form game is a probability distribution $p : \mathcal{A} \to [0,1]$ over the action profiles. p is called a **correlated equilibrium**, if for each player i and all actions $a_i, b_i \in A_i$, we have

$$\sum_{a_{-i} \in A_{-i}} p(a_{-i}, a_i) u_i(a_{-i}, a_i) \geq \sum_{a_{-i} \in A_{-i}} p(a_{-i}, a_i) u_i(a_{-i}, b_i).$$

In a correlated strategy profile, the probability distribution over the actions is not necessarily given by a product distribution. It therefore assumes that the players communicate. It can be thought of as a situation where the players (e.g. leaders of different nations) come together to decide on a strategy and then randomize together (e.g. one lucky wheel that decides the action for all players) and then each player plays their assigned action.

If we think of a situation where the other players cannot check whether a player really has played their assigned action or not, then the concept of correlated equilibrium becomes quite intuitive. It simply states that no player has an incentive to play a different action than the one assigned to them.

3 Cooperative Game Theory

Lec 7 2021-06-01

So far, we have considered non-cooperative games, for which no binding agreements are possible. Now we will consider cooperative game theory, in which binding agreements can be made.

Definition 3.1. A **cooperative game (in characteristic function form)** is a finite set of players N together with a **characterstic function** $v : \mathcal{P}(N) \to \mathbb{R}$ with $v(\emptyset) = 0$.

The characterstic function represents the costs or payoffs that occur if a certain coalition is formed.

Definition 3.2. A cooperative game (N, v) is called **additive**, if for any two disjoint subsets $S, T \subset N$, we have

$$v(S \cup T) = v(S) + v(T).$$

Additive games are precisely those games where the total costs cannot be decreased or increased by form a coalition, so no synergy effects are possible. Furthermore, the characterstic function v in additive games is uniquely determined by its values on the singletons. Thus, additive games can be represented as a vector in $\mathbb{R}^n$ with i-th entry equal to $v(i)$.

Definition 3.3. A cooperative game (N, v) is called **convex**, if for all sets of players $S \subset N$, $T \subset N$, we have

$$v(S \cup T) \geq v(S) + v(T) - v(S \cap T).$$

Intuitively, convex games are those where synergy effects by cooperation of the players is always non-negative.

Lemma 3.4. Every additive cooperative game is convex.

Proof. If (N, v) is an additive cooperative game and $S, T \subset N$, then

$$\begin{aligned} v(S \cup T) + v(S \cap T) &= \sum_{i \in S \cup T} v(i) + \sum_{i \in S \cap T} v(i) \\ &= \sum_{i \in S \setminus T} v(i) + \sum_{i \in T \setminus S} v(i) + 2 \cdot \sum_{i \in S \cap T} v(i) \\ &= \sum_{i \in S} v(i) + \sum_{i \in T} v(i) = v(S) + v(T). \end{aligned}$$

□

Definition 3.5. A cooperative game (N, v) is called **superadditive**, if for any two disjoint subsets $S, T \subset N$, we have

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

Intuitively, superadditive games are games for which the slogan

"The whole is more than the sum of its parts."

applies. Most cooperative games occuring in reality are superadditive.

Lemma 3.6. Ever convex game is superadditive.

Proof. This follows since $v(\emptyset) = 0$. □

We conclude that for cooperative games, the following implications hold:

$$\text{additive} \implies \text{convex} \implies \text{superadditive}.$$

Note that even for additive games, a subset of a larger set can be assigned more payoff than the larger set; e.g. $N = \{1, 2\}$, $v(\{1\}) = -1$, $v(\{2\}) = 2$, $v(\{1, 2\}) = 1$.

Definition 3.7. A cooperative game (N, v) is called **simple** (or **voting games**), if $v : N \to \mathbb{R}$ only obtains the values 0 or 1. Furthermore, whenver $v(S) = 1$ for a subset $S \subset N$, then $v(T) = 1$ for all supersets $T \supset S$.
A player $i \in N$ with $v(S) = 0$ for all $S \subset N \setminus \{i\}$ is called **vetoer**. Equivalently, a player $i \in N$ is a vetoer if and only if $v(N \setminus \{s\}) = 0$.

After coalitions have been formed, it still needs to be decided how to distribute the costs or payoffs on the members of the coalition. This is what solution concepts model. After a coalition has been formed, we may restrict the game to the set of players forming the coalition and thus we may assume that the coalition consists of all players.

Definition 3.8. A **solution concept** of cooperative games is a function ϕ mapping a cooperative game (N, v) to a **cost/payoff vector** $\phi(N, v) \in \mathbb{R}^{|N|}$, whose i-th component represents the costs/payoff of the i-th player.
Writing $n := |N|$, a payoff vector $x \in \mathbb{R}^n$ is called **feasible**, if $\sum_{i=1}^n x_i \leq v(N)$. This means that it does not distribute more payoff than is available.
A payoff vector $x \in \mathbb{R}^n$ is called **efficient**, if $\sum_{i=1}^n x_i = v(N)$.
It is called **individually rational**, if $x_i \geq v(\{i\})$ for all $i \in N$; i.e. no player obtains more payoff by working alone instead of joining the coalition.

The *core* constitutes a "weaker form" of a solution concept. It roughly measures the stability of a coalition; i.e. how likely the coalition is to break.

Definition 3.9. For a cooperative game (N, v), a payoff vector $x \in \mathbb{R}^n$ is in the **core**, if it is efficient and $\sum_{i \in S} x_i \geq v(S)$ for all $S \subset N$.

Note that the core as a solution of a set of linear inequalities is convex and it may also be empty. If we ignore the games for which the core is empty and choose for each other game a single vector in the core, the resulting "weaker form" of a solution concept is individually rational.

Theorem 3.10. Every convex game has nonempty core.

Theorem 3.11. A simple game has nonempty core if and only if it has a vetoer.

Definition 3.12. In a cooperative game (N, v), the **marginal contribution** of player $i \in N$ to $S \subset N$ is $v(S \cup \{i\}) - v(S)$.

We now introduce terminology to describe "good" solution concepts of cooperative games.

Definition 3.13. For each set of players $N \subset \mathbb{N}$, the set of all characterstic functions forms an abelian group.
A solution concept ϕ of cooperative games is called **additive**, if its components $\phi(N, -)$ are a group homomorphism for each N; i.e. if $\phi(N, v + v') = \phi(N, v) + \phi(N, v')$ for all subsets $N \subset \mathbb{N}$ and characterstic functions $v, v' : \mathcal{P}(N) \to \mathbb{R}$.
A solution concept ϕ of cooperative games is called **symmetric**, if whenever $v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \subset N \setminus \{i, j\}$, then $\phi_i(N, v) = \phi_j(N, v)$. Intuitively, this means that if two players are completely interchangeable, then a symmetric payoff function must assign them the same payoff.
A solution concept ϕ of cooperative games satisfies **nullity**, if any player $i \in N$ satisfying $v(S \cup \{i\}) = v(S)$ for all $S \subset N \setminus \{i\}$ receives zero payoff $\phi_i(N, v) = 0$. In other words, a player that does not increase the payoff of any coalition should get a payoff of 0.

Clearly, a solution concept satisfying all three of the defined axioms exists, since an example is the zero function $\phi = 0$. However, if one furthermore wants the solution concept to be efficient, then there is only one such concept, as Shapley proved in 1953.

Theorem 3.14. There is a unique solution concept of cooperative games that is efficient, additive, symmetric and satisfies nullity. It is called **Shapley value**.
Given a cooperative game (N, v), the payoff given to player i is

$$\begin{aligned}\mathrm{Sh}_i(N, v) &:= \sum_{S \subset N, i \notin S} \frac{|S|!(n - |S| - 1)!}{n!}(v(S \cup \{i\}) - v(S)) \\ &= \frac{1}{n} \sum_{S \subset N, i \notin S} \frac{1}{\binom{n-1}{|S|}}(v(S \cup \{i\}) - v(S)).\end{aligned}$$

In words, we consider all nonempty coalitions that i is not a part of and average the marginal contribution of i to the coalition by the number of coalitions excluding i that have this size. We then average over the number of players.

In Theorem 3.10 we saw that every convex game has a nonempty core. The following theorem strengthens that result.

Theorem 3.15. In every convex game, the Shapley value lies within the core.

Theorem 3.16. In superadditive games, the Shapley value satisfies individual rationality.

4 Stable Matchings

Definition 4.1. A **matching** of two sets A, B is a subset of the cartesian product $A \times B$. Lec 8 2021-06-08

We are only interested in the case that A denotes the set of agents. B may also be the set of agents or a set of items. However, we don't want "random" matchings, but matchings that are compatible with the preferences of the agents.
A matching of agents with a set of "items" is called **resource allocation**, whereas a matching between agents is called **one-sided matching** or **two-sided matching**, depending on whether there are two different "types" of agents or not.

We first focus on two-sided matchings where one agent is assigned to a single other (**marriage setting**).
Suppose we want to match a finite set $W = \{w_1, \ldots, w_n\}$ (*women*) with the finite set $M = \{m_1, \ldots, m_n\}$ (*men*), where each agent $w \in W$ has preferences regarding the agents in M and vice versa, which is given by a total order. We restrict ourselves to the case that one agent is assigned to a single other (*marriage*), so a matching is just a bijection $\mu : W \to M$.

Definition 4.2. A **blocking pair** for a matching $\mu : W \to M$ in the marriage setting is a tuple $(w_i, m_j) \in W \times M$, such that $m_j >_{w_i} \mu(w_i)$ and $w_i >_{m_j} \mu(m_j)$. Intuitively, this means that both candidates prefer each other over their current partner.
A matching is called **stable**, if it does not have a blocking pair.

Whether there exists a blocking pair in a matching with n men and women can be checked in polynomial time $O(n^2)$.
The following simple algorithm allows us to compute a stable matching.

Algorithm 4.3 (**Gale-Shapley Algorithm** (or **Deferred Acceptance Algorithm**)).
Start with an "empty matching" and iterate until there are no unengaged men:

(a) Each man who does not have a temporary partner yet proposes to the woman he prefers the most among the women who did not reject him yet.

(b) Each woman temporarily accepts the proposition of the man she prefers and rejects all other propositions.

The usefuless of the algorithm lies in the following theorem, which was proven by Gale and Shapley in 1962.

Theorem 4.4. Any marriage setting with n men and women permits a stable matching, which can be constructed in polynomial time $O(n^2)$ using the *Gale-Shapley Algorithm*.

By interchanging the role of men and women in the algorithm, one can oftentimes produce a different matching than the original one. In particular, stable matchings are not unique. Therefore, it is useful to introduce some terminology in order to classify different solutions to the *stable marriage* problem.

Definition 4.5. Consider the set S of stable matchings in a marriage setting. We may define a partial order $\geq_M$ on S, where $\mu \geq_M \mu'$ for stable matchings $\mu, \mu' \in S$, if $\mu(m) \geq_m \mu'(m)$ for all $m \in M$. The analogous definition for W gives us another partial order $\geq_W$.
A stable matching in the marriage setting is **M-optimal** (**W-optimal**), if it is a greatest element with respect to $\geq_M$ ($\geq_W$); i.e. if every man (woman) weakly prefers this matching to any other stable matching.

In particular, a M-optimal (W-optimal) matching is unique if it exists. In fact, it always exists, as the following theorem shows.

Theorem 4.6. The matching resulting from the *Gale Shapley Algorithm* is the unique M-optimal matching. In particular, by switching the role of M and W in the algorithm, we obtain the unique W-optimal matching.

Proof. Let μ denote the matching obtained from the *Gale Shapley Algorithm*. Aiming for contradiction, suppose that there is a stable matching μ^* and $m \in M$, such that $\mu^*(m) >_m \mu(m)$. By definition of the *Gale Shapley Algorithm*, a man proposes to his preferred women first, so

$$\mu^*(m) >_m \mu(m) \iff \mu^*(m) \text{ has rejected } m \iff \exists\ m' \in M : m' >_{\mu^*(m)} m.$$

Furthermore, since the algorithm consists of an iteration of propositions of men, there must have been a first man $m^* \in M$, who was rejected by $\mu^*(m^*)$; i.e. all other men $x \in M$ that proposed before him to $\mu^*(x)$ were accepted. Replacing m^* by m, we may assume that m has this property.
At the time in the algorithm when m is rejected, there must have been another man $m' \in M$ that $\mu^*(m)$ preferred to m; i.e. $m' >_{\mu^*(m)} m$. But this implies that m' did not propose to $\mu^*(m')$ up to that point, because otherwise he would have been accepted. Therefore $\mu^*(m) >_{m'} \mu^*(m')$ and $(m', \mu^*(m)) \in M \times W$ is a blocking pair, yielding a contradiction. □

Roth showed in 1982:

Theorem 4.7. The unique M-optimal matching is weakly Pareto-optimal for men; i.e. there is no matching that every man strictly prefers to it.

However, the M-optimal matching is "bad" for women:

Theorem 4.8. In the unique M-optimal matching, every woman is matched with the worst partner she can have in any stable matching.

The result of the *Gale Shapley Algorithm* is not *strategy-proof*. In fact, there is no mechanism that is both stable and strategy-proof.

We now focus on one sided matchings, where one agent is assigned to a single other (**roommate setting**). Denote the set of agents by $R := \{r_1, \dots, r_{2n}\}$, where R is supposed to be even and each agent $r \in R$ has a total order over $R \setminus \{r\}$.
In this case, a matching is an involution $\mu : R \to R$ (i.e. $r \circ r = \mathrm{id}_R$) with no fixed points ($\mu(r) \neq r$ for all $r \in R$).

Definition 4.9. A **blocking pair** for a matching $\mu : R \to R$ in the roommate setting is a pair $(r, r') \in R^2$, such that $r' >_r \mu(r)$ and $r >_{r'} \mu(r')$.
A matching is called **stable**, if it does not have a blocking pair.

In contrast to the situation in the marriage setting, stable matchings in the roommate setting do not always exist.

Theorem 4.10. In a roommate setting, we can use *Irving's Algorithm* to determine if a stable matching exists and if so, compute one, both in polynomial time.

The first step of the algorithm relies on the following lemma. A proof of the lemma can be found in Irving's paper *An efficient algorithm for the "stable roommates" problem* from 1985 (Lemma 1 and Corollary 1.1).

Lemma 4.11. Consider a roommate setting and two different candidates A and B. If B is A's favorite candidate of the ones that proposed to A in the first part of the algorithm, then any stable matching must match A with a candidate that A likes at least as much as B.

Stated differently, the lemma says that no partner in a stable matching can be worse than the best proposal received.

5 Refinements of Nash Equilibria

In this section we highlight some refinements of Nash equilibria. Lec 9 2021-06-15

5.1 Trembling-hand perfect Equilibria

The idea is to only consider Nash equilibria that are not affected if a player slightly alters their strategy ("trembling hand").
In a game with N players and $n_i \in \mathbb{N}$ actions for the player i, a strategy profile is a tuple of probability vectors in $\prod_{i=1}^{|N|} \mathbb{R}^{n_i}$, which can be identified with a closed subset of $\mathbb{R}^m$, where $m := \sum_{i=1}^{n} n_i$. We thus get a notion of convergence of strategy profiles.

Definition 5.1. A strategy profile s is called a **trembling-hand perfect equilibrium**, if there exists a sequence $(s^{(n)})_{n\in\mathbb{N}}$ of full-support strategy profiles $s^{(n)}$, such that $\lim_{n\to\infty} s^{(n)} = s$ and s_i is a best response to $s_{-i}^{(n)}$ for all $n \in \mathbb{N}$, $i \in N$.

Lemma 5.2. Any trembling-hand perfect equilibrium is a Nash equilibrium.

Proof. Let s be a trembling-hand perfect equilibrium with sequence $(s^{(n)})_{n\in\mathbb{N}}$. We have to show that for any player i, s_i is a best response to s_{-i}. Because s_i is a best response to $s_{-i}^{(n)}$, it follows $u_i(s_{-i}^{(m)}, s_i) \geq u_i(s_{-i}^{(m)}, s'_i)$ for any other strategy profile s'_i of i. Together with the continuity of sums and products, we see that for any player i, s_i is a best response to s_{-i}:

$$\begin{aligned}
u_i(s) &= \sum_{a\in A} u_i(a) \prod_{j=1}^{n} s_j(a_j) \\
&= \lim_{m\to\infty} \sum_{a\in A} u_i(a) \cdot s_i(a_i) \prod_{j\in\{1,\dots,n\}\setminus\{i\}} s_j^{(m)}(a_j) \\
&= \lim_{m\to\infty} u_i(s_{-i}^{(m)}, s_i) \\
&\geq \lim_{m\to\infty} u_i(s_{-i}^{(m)}, s'_i) \\
&= \sum_{a\in A} u_i(a) \cdot s'_i(a_i) \prod_{j\in\{1,\dots,n\}\setminus\{i\}} s_j(a_j) = u_i(s_{-i}, s').
\end{aligned}$$

□

Clearly, every full-support Nash equilibrium is trembling-hand perfect.

Theorem 5.3. In any game, trembling-hand perfect equilibria exist. The problem of finding such equilibria is PPAD-complete.

5.2 Strong Equilibria

While ordinary Nash equilibria by definition ensure that a single individual has no motivation to deviate from the proposed strategy, a *strong equilibrium* ensures that the same holds true for any coalition of players.

Definition 5.4. A **strong equilibrium** of a game is a strategy profile s, such that for any coalition of players $C \subset N$, there does not exist a strategy profile t_C of the coalition, such that $u_i(t_C, s_{-C}) > u_i(s)$ for all $i \in C$.

Lemma 5.5. Any strong equilibrium yields a weakly Pareto optimal outcome.

Proof. Let s and t be two strategy profile, such that the outcome of t is strictly better for a nonempty set of players $C \subset N$ and equal for the rest. Then $u_i(t_C, s_{-C}) > u_i(s)$ for all $i \in C$, so s is not a strong equilibrium. □

Every Nash equilibrium in a two-player zero-sum game is strong.
The main problem with strong equilbria is that there are many games for which they do not exist.

5.3 Coalition-proof Equilibria

Coalition-proof equilibria generalize strong equilibria by only demanding that there are no stable coalitions that have an incentive to deviate from the proposed strategy.

Definition 5.6. A **coalition-proof equilibrium** of a game is a strategy profile s, such that for any stable coalition of players $C \subset N$, there does not exist a strategy profile t_C of the coalition, such that $u_i(t_C, s_{-C}) > u_i(s)$ for all $i \in C$.

By definition, every strong equilibrium is coalition-proof. Just like strong equilbria, coalition-proof equilibria do not exist in many games.

5.4 Quasi-strict Equilibria

The idea of the *quasi-strict equilibria* is to ensure that every player will be motivated to play their assigned strategy by getting strictly less payoff whenever they deviate.

Definition 5.7. A Nash equilibrium s is called **quasi-strict equilibrium**, if for any player i and any $a_i \in \operatorname{supp}(s_i)$, $b_i \notin \operatorname{supp}(s_i)$, we have $u_i(a_i, s_{-i}) > u_i(b_i, s_{-i})$.

In particular, any full-support Nash equilibrium is quasi-strict.

Theorem 5.8. Quasi-strict equilibria exist for all two-player games.

6 Zero-Sum Games

Definition 6.1. A two-player game is called a **zero-sum game**, if for every action profile $a \in A$, we have $u_1(a) + u_2(a) = 0$.

Intuitively, zero-sum games are two-player games for which one player's win is the other ones loss and vice versa. Any outcome of a zero-sum game is necessarily weakly Pareto optimal. They can be represented with a single matrix (by convention representing u_1).

Note that two player games for which the sum of the payoffs for both players are constant (*constant-sum games*) are strategically equivalent to zero-sum games, as there exists a positive affine transformation, which transforms one into the other. Most of our solution concepts (e.g. Nash equilibria) are invariant under affine transformations.

A key observation for zero-sum games is that the payoff of a player should lie between their security level and the negative of their opponent's security level. That these two bounds actually agree is known as the *Minimax theorem*, one of the most important results in game theory. It was proven by von Neumann in 1928.

Theorem 6.2 (Minimax Theorem). In a zero-sum game, the security level of player 1 is the negative of the security level of player 2:

$$\max_{s_1} \min_{s_2} u_1(s_1, s_2) = -\max_{s_2} \min_{s_1} u_2(s_1, s_2) = \min_{s_2} \max_{s_1} u_1(s_1, s_2).$$

Proof. Since the security level of player 2 is necessarily an upper bound for the security level of player 1, we have

$$v_1 \coloneqq \max_{s_1} \min_{s_2} u_1(s_1, s_2) \leq \min_{s_2} \max_{s_1} u_1(s_1, s_2) \eqqcolon v_2.$$

Using the guaranteed existence of a Nash equilibrium s^*, we set $v_1^* := u_1(s^*)$ and show $v_1^* \le v_1$ by the calculation

$$\begin{aligned} v_1 &= \max_{s_1} \min_{s_2} u_1(s_1, s_2) \\ &\ge \min_{s_2} u_1(s_1^*, s_2) \\ &= -\max_{s_2} -u_1(s_1^*, s_2) \\ &= -\max_{s_2} u_2(s_1^*, s_2) \\ &= -u_2(s^*) = u_1(s^*) = v_1^*. \end{aligned}$$

Analogously, we see $v_1^* \ge v_2$, so $v_1 = v_1^* = v_2$. □

In particular, every zero-sum game is determined, in the sense that there is a unique "individually rational" outcome. This of course does not mean that the strategy to get to that outcome is unique, for example in the zero game (all payoffs are zero).

Definition 6.3. The **value** of a zero-sum game is the security level of player 1.

The minimax theorem also implies that Nash equilibria and maximin strategies agree for zero-sum games. In particular, all combinations of Nash equilibria in zero-sum games are again Nash equilibria, which yield the same payoff.

Lec 10
2021-06-22

Definition 6.4. A **generalized saddle point** is a set consisting of nonempty sets of actions $\emptyset \ne B_i \subset A_i$ for each player $i \in N$, such that for all players $i \in N$ and $a_i \notin B_i$, there exists $b_i \in B_i$ that strictly dominates a_i in the subgame $G|_{B_1 \times \cdots \times A_i \times \cdots \times B_n}$.

This can be thought of as a set-valued variation of quasi-strict Nash equilibria. Intuitively, a player will be dissuaded from playing an action that is not in their assigned set of actions, assuming that all other players stick to their set of assigned actions.

Of course, the set of all actions for each player constitutes a generalized saddle point. It is thus natural to ask for generalized saddle points that are as small as possible.

Definition 6.5. The set of generalized saddle points is partially ordered by inclusion. A **saddle** is a minimal element of this partially ordered set.

Since this is a partial order on a finite set, any game has a saddle. The following theorem shows that they are unique in zero-sum games.

Theorem 6.6. Every zero-sum game contains a unique saddle.

The theorem follows from the following two lemmata.

Lemma 6.7. In a zero-sum game, any two generalized saddle points (B_1, B_2), (C_1, C_2) must satisfy $B_1 \cap C_1 \ne \emptyset$ and $B_2 \cap C_2 \ne \emptyset$.

Lemma 6.8. In a zero-sum game, if (B_1, B_2), (C_1, C_2) are two generalized saddle points, then their intersection $(B_1 \cap C_1, B_2 \cap C_2)$ is also a generalized saddle point.

There is a straightforward (greedy) algorithm to determine the smallest generalized saddle point that contains a given set of actions for each player.

Algorithm 6.9. Let G be a zero-sum game and $B_1 \subset A_1$, $B_2 \subset A_2$ subsets of the action profiles of the players. The smallest generalized saddle point that contains (B_1, B_2) can be obtained as follows:

(a) Find all rows X_1 that are not already in B_1 and are purely undominated in the subgame $G|_{A_1 \times B_2}$.

(b) Find all columnns to X_2 that are not already in B_2 and are purely undominated in the subgame $G|_{B_1 \times A_2}$.

(c) Set $B_1 := B_1 \cup X_1$, $B_2 := B_2 \cup X_2$ and iterate until both X_1 and X_2 are empty.

(d) The resulting (B_1, B_2) is the desired set.

By iterating this algorithm for each cell of the game and then returning the smallest one found, we get a polynomial-time algorithm that determines the unique saddle of a zero-sum game.
To make the algorithm more efficient, we want to determine a point that is always contained in the saddle.
Our first simple observation towards this goal is the following.

Lemma 6.10. In a zero-sum game G, if $a_1 \in A_1$ is contained in the saddle, then any action $a_2 \in A_2$ that minimizes $u_1(a_1, a_2)$ is also contained in the saddle.

Definition 6.11. A **maximin point** of a zero-sum game G is an action profile $(a_1, a_2) \in A_1 \times A_2$, such that

$$(a_1, a_2) \in \arg_{(a_1,a_2)\in A_1 \times A_2} \left(\max_{a_1} \min_{a_2} u_1(a_1, a_2) \right).$$

In words, a maximin point is just a maximum among all row minima.

Even though the definition of a maximin point is similar to that of maximin strategy, it does not necessarily constitute a maximin strategy for either of the players. The reason is that it does not consider mixed responses of the opponent.
For example, consider the zero-sum game

0	2
3	1

It has a unique maximin point, but the only maximin strategy (Nash equilibrium) is randomizing uniformly for both players.

Our interest in maximin points is justified by the following theorem.

Theorem 6.12. Every maximin point is contained in the saddle.

Thus we get an efficient algorithm that computes the saddle of a zero-sum game in linear time.

Algorithm 6.13. The saddle point of a zero-sum game can be determined by invoking Algorithm 6.9 with (B_1, B_2) set to a maximin point of the game.

In fact, all saddles of any normal-form game can be found in polynomial time.

7 Succinct Games and Commitments

So far we have only looked at games in normal-form games. One inconvenient feature of normal-form games is that their required space usage is exponential in the number of players. In contrast, *succinct games* are precisely those games which can be represented in such a way that the required space is polynomial in the number of players.

Definition 7.1. A class of games is called **succinct**, if its utility functions can be represented in such a way that their space is polynomial in the number of players.

Succinct games usually have some form of "symmetry" on the set of players, which is exploited in order to achieve the reduced space usage.

7.1 Anonymous Games

One example for such a class of games are *anonymous* games, which are those games in which all players have the same set of actions and the payoff of a player only depends on their own action played and the number of times the actions were played by the other players. It does not depend on who of the other players played which action.

Definition 7.2. A game with action sets $A_1, \dots, A_n$ is called **anonymous**, if $A_1 = \cdots = A_n$ and for any $i \in N$ and permutation $\pi : N \to N$ with $\pi(i) = i$, we have

$$u_i(a) = u_i(a_{\pi(1)}, a_{\pi(2)}, \dots, a_{\pi(n)}).$$

In other words, since the group of permutations fixing a player $i \in N$ acts on $\mathcal{A}$ in the obvious way and acts trivially on $\mathbb{R}$, the above amounts to demanding that for all $i \in N$, the payoff function $u_i : A \to \mathbb{R}$ is an equivariant map.

By definition, every two-player game with the same set of actions for both players is anonymous.
An anonymous game with n players and k actions for each player can be specified as follows: For each player, a partial function

$$\tilde{u}_i : \{1, \dots, k\} \times \prod_{i=1}^{k} \{1, \dots, n-1\} \to \mathbb{R},$$

where $\tilde{u}_i(a, v)$ describes the payoff of player i if they play a and the action j is played v_j times by the other players. Therefore, the space required is bounded from above by $k \cdot (n-1)^k$, which is a polynomial in n and thus anonymous games are succinct whenever k is constant (i.e. independent of n).

If additionally the payoff functions for all players are the same, then the game is called *symmetric*.

7.2 Symmetric Games

Definition 7.3. A game with action sets $A_1, \dots, A_n$ is called **symmetric**, if $A_1 = \cdots = A_n$ and for any $i \in N$ and permutation $\sigma : N \to N$, we have

$$u_i(a) = u_{\pi(i)}(a_{\pi(1)}, a_{\pi(2)}, \dots, a_{\pi(n)}).$$

In particular, every symmetric game is anonymous. Additionally, a symmetric game must necessarily assign the same utility to all players that play the same action. A two-player game is symmetric if and only if the sets of actions of the two players agree and $u_1(a_1, a_2) = u_2(a_2, a_1)$ for all $a_1, a_2 \in A_1 = A_2$; i.e. if the payoff matrix of the second player is the transpose of that of the first player.
Therefore, a zero-sum game is symmetric if and only if both players have the same actions and the payoff matrix is antisymmetric. In particular, the diagonal must consist of only zeros.

We already discussed that finding a Nash equilibrium in normal-form games is PPAD-complete (even for two players). The same holds for anonymous games by the following argument: Any two-player normal-form game can be made into an anonymous two-player game by renaming and adding actions. The added actions should give worst possible utility to both players, so any Nash equilibrium must correspond to actions from the original game. Therefore, if we could efficiently find a Nash equilibrium for anonymous games, then the same would be true for two-player normal-form games. Lec 11 2021-06-29
Using *Gale's symmetrization procedure*, it is not hard to see that finding a Nash equilibrium in symmetric games is PPAD-complete.

Nash showed in 1951:

Theorem 7.4. Every symmetric game contains a symmetric Nash equilibrium; i.e. an equilibrium in which all players play the same strategy.

Theorem 7.5. In a symmetric zero-sum game, the security level of both players is 0.

7.3 Graphical Games

Another class of games are graphical games, which are those games for which the payoff of any player i is only affected by the actions of a set of other players $\Gamma(i)$.

Definition 7.6. A **graphical game** is a game for which there exists a function $\Gamma : N \to \mathcal{P}(N)$, such that for all action profiles $a, b \in A$ with $a_i = b_i$ and $a_j = b_j$ for all $j \in \Gamma(i)$, we have $u_i(a) = u_i(b)$. The set $\Gamma(i)$ is called *local neighborhood* of i.

Clearly, ever normal-form game can be viewed as a graphical game with maximal neighborhoods and graphical games are succinct if the size of the neighborhoods is bounded from above by a constant.

Theorem 7.7. Finding a Nash equilibrium in a graphical game with degree bounded by 3 is PPAD-complete.

7.4 Stackelberg Games

In some settings, players can commit to their performance of an action (e.g. via a binding contract).

Definition 7.8. A two-player game is called **Stackelberg game**, if one player (called *leader*) commits to a strategy before the other player (called *follower*) picks their action.

There exist games for which it is optimal to commit to dominated actions, assuming the opponent wants to maximize their payoff. Under the same assumption, one can construct games for which it is beneficial to be the leader and games for which it is better to be the follower.

Of course, committing to a mixed strategy instead of a pure one can increase the payoff. If we only allow committing to pure strategies, then we can find an optimal pure strategy to commit to by iterating through each of our actions, computing a best response for our opponent and then choosing the action that maximizes our payoff.

The following theorem shows that even if we are allowed to commit to mixed strategies, we can still efficiently determine an optimal commitment.

Theorem 7.9. An optimal mixed strategy to commit to can be computed in polynomial time.

Proof (sketch). For every pure strategy s_2 of the follower, compute a mixed strategy s_1 for the leader that maximizes $u_1(s_1, s_2)$ and such that s_2 is a best response to s_1. This can be done efficiently using linear programming. □

8 Extensive-Form Games

Lec 12
2021-07-06

Extensive-form games are games with sequential moves. They are usually modeled by graphs. We will only consider extensive-form games in which all moves are observable by all players; that is, every player has *perfect information* about the actions taken by the other players.

Definition 8.1. An *extensive-form game (with perfect information)* consists of the following data:

(a) A finite set of *players* N.

(b) A finite set of *actions* A for all players (the union of the actions of each player).

(c) A finite set of *nonterminal nodes* H and a finite set of *terminal nodes* Z.

(d) An *action function* $\chi : H \to \mathcal{P}(A) \setminus \{\emptyset\}$, specifying which actions are available at a particular point of the game.

(e) A *player function* $\rho : H \to N$, specifying whose turn it is at a particular point of the game.

(f) A *successor function* $\sigma : H \times A \to H \cup Z$, which represents the structure of a directed rooted tree. In particular, every node has at most one ingoing edge.

(g) A utility function $u : Z \to \mathbb{R}^{|N|}$, assigning each player their utility.

The *strategy set* of player i is $S_i := \prod_{h \in H, \rho(h)=i} \chi(h)$.

In this data, the assumptions on σ ensure that no cycles can occur and thus any game will always terminate. It also allows us to identify every node with its history (the nodes coming before it in the tree).

Any extensive-form game can be converted into a normal-form game by considering the strategy sets of each player as their set of actions. The outcome corresponding to a choice of an action for each player is just the outcome of the original extensive-form game if every player plays according to their chosen strategy. This even works for extensive-form games without perfect information.

The concept of Nash equilibria directly transfers from normal-form to extensive-form games. We now define a refinement of Nash equilibria, that makes use of the sequential nature of extensive-form games.

Definition 8.2. A Nash equilibrium s of an extensive-form game G is called **subgame-perfect**, if s is a Nash equilibrium in every subtree of G.

Selten showed the following result in 1965.

Theorem 8.3. Every extensive-form game contains a pure subgame-perfect equilibrium. It is unique if there all outcomes are different and using *backward induction* it can be computed in polynomial time.

Proof. Using *backward induction*, we construct strategies $s_i \in S_i$ for each player $i \in N$, by starting at the leaves and iteratively going up the tree, choosing (one of) the best option for the player whose turn it is.
Claim: This yields a subgame-perfect equilibrium.
We show this by induction on the number of nodes in the tree. The base case for the singleton trees consisting of the terminal nodes is clear. Now consider a subtree T of height n from the bottom of our original tree. By the inductive hypothesis, the s_i constitute a Nash equilibrium in all proper subtrees of T. It is left to prove that the s_i also represent a Nash equilibrium for T. By definition, the chosen action of the player whose turn it is at the root of T is a best response, so no improvement can be made by changing the action at the root of T. Therefore, if any player j could improve their payoff by deviating from s_j, then this would also apply to a strict subtree of T, which cannot be the case.
Since this constitutes a necessary condition for subgame-perfect equilibria, it is unique whenever the best responses at each iteration of the algorithm are unique. This is in particular the case if all outcomes are different.
It is clear that this algorithm can be computed in polynomial time. □

For the special case of extensive-form zero-sum games, backward induction is called *minimax algorithm*. In that case, the game is described by the payoff of the first player, who tries to maximize that value, whereas the second player tries to minimize it. The runtime of the minimax algorithm can be improved using *alpha–beta pruning*.

We conclude with the following remarkable statement.

Theorem 8.4 (Zermelo's Theorem). Every extensive-form zero-sum game is *strongly determined*; i.e. there is a unique value (corresponding to rational play) obtainable by pure strategies.

Proof. By Theorem 8.3, there exists a pure Nash equilibrium and since it is a zero-sum game, it corresponds to a maximin strategy of both players. □

In particular, since the value of such games is obtainable by pure strategies, there are only finitely many values that the value of the game may take. For example, for the game of chess, the value of the game thus must be a win for white, a tie or a win for black.