UNIVERSITY OF PASSAU

Faculty of Computer Science and Mathematics

Lecture

Computational Game Theory

read in SS 2023

Prof. Dr. Tobias Harks

Universität Passau
Lehrstuhl für Mathematische Optimierung
Fakultät für Informatik und Mathematik
Dr.-Hans-Kapfinger-Str. 30
94032 Passau

Email: tobias.harks@uni-passau.de

July 18, 2023

Preface

This script has been developed during the course of several lectures held at TU Berlin (2007-2011), Maastricht University (2011-2015) and Augsburg University (2015-2021).

For the many hints and improvements I would like to thank Lukas Graf, Florian Göttl and Julian Schwarz.

Passau, March 2023
Prof. Dr. Tobias Harks

Contents

Chapter 1

Strategic Games

In this chapter, we introduce the central game-theoretic concepts used in this textbook. For a comprehensive treatment, see also the textbooks by Fudenberg and Tirole [3] and Osborne and Rubinstein [9].

Definition 1.1. A strategic game (or game in normal form) is a tuple $G := (N, S, u)$ such that

- $N = \{1, \ldots, n\}$ is a finite set of players.
- A strategy space $S = \times_{i \in N} S_i$, where for every $i \in N$, a set $S_i \neq \emptyset$ of feasible (pure) strategies is given. An element $s = (s_1, \ldots, s_n) \in S$ is called strategy profile or just profile.
- A combined utility function $u = (u_i)_{i \in N}$, where for every $i \in N$, there is a utility function $u_i : S \to \mathbb{R}$.

Is S_i finite for all $i \in N$, we speak of a finite strategic game.

For $\{1, \ldots, n\}$ we sometimes write $[n]$. For $G := (N, S, u)$, the following point of view applies:

Complete information: Every $i \in N$ knows $S_j, j \neq i$ and n $u_j, j \neq i$ of the other players $j \in N \setminus \{i\}$; Based on this information, every player $i \in N$ chooses a strategy $s_i \in S_i$.

Simultaneous decision: The selected strategies are chosen simultaneously; no player i knows ahead the strategies $s_j \in S_j$ chosen by $j \in N \setminus \{i\}$.

Rationality: Every $i \in N$ wants to maximize $u_i(s)$. If u_i represents costs, then players want to minimize costs.

Representation:

Finite two-player games (N, S, u) with $N = \{1, 2\}$ can be compactly represented by a bi-matrix, where the row labels k correspond to strategies of player 1 (the row-player) and the column labels ℓ to strategies of player 2 (the column-player):

1.1 Nash-Equilibrium

A strategy profile from which no player has an incentive to unilaterally deviate is called Nash-Equilibrium. This solution concept dates back to Cournot (18. Century) but is

$$
\begin{array}{c|ccccc|}
 & \dots & l & \dots & l' & \dots \\
\hline
\vdots & & \vdots & & \vdots & \\
k & \dots & (u_1(k,l), u_2(k,l)) & \dots & (u_1(k,l'), u_2(k,l')) & \dots \\
\vdots & & \vdots & & \vdots & \\
k' & \dots & (u_1(k',l), u_2(k,l)) & \dots & (u_1(k',l'), u_2(k',l')) & \dots \\
\vdots & & \vdots & & \vdots & \\
\hline
\end{array}
$$

Figure 1.1: Bi-matrix representation of a finite two-player game in strategic form.

named after John F. Nash (1951), because he first derived a rigorous existence proof of so-called mixed strategy Nash equilibria for finite strategic games, see [7].
Before we can present the formal definition of Nash equilibria, we need the following notation: Let $s = (s_i)_{i \in N} \in S$. We definine s_{-i} as the vector

$$s_{-i} := (s_j)_{j \in N \setminus \{i\}} = (s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n).$$

For some $k \in S_i$, we set $(k, s_{-i}) := (s_1, \dots, s_{i-1}, k, s_{i+1}, \dots, s_n)$.

Definition 1.2. A pure Nash equilibrium (PNE) of a strategic game $G = (N, S, u)$ is a profile $s^* \in S$, such that for every $i \in N$ we have:

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*) \quad \forall s_i \in S_i.$$

For an interpretation, let $i \in N$ and fix the strategies chosen by the rival players to some $a_j^*, j \in N \setminus \{i\}$. Then, the strategy $s_i^* \in S_i$, maximizes the utility of player i holding the other strategies of players $j \in N \setminus \{i\}$ fixed. The defintion of PNE raises the following

questions:

- Does a PNE always exsist?
- Is it unique?
- Is it efficiently computable?

Examples

Example 1.3 (Bach or Stravinsky (BoS)). Two friends want to go to a concert. Friend 1 likes Bach more, Friend 2 likes Stravinsky more. Both prefer to go together to the concert instead of alone. The utility function of the strategic game is defined as

	Bach	Stravinsky
Bach	$\boxed{(2,1)}$	$(0,0)$
Stravinsky	$(0,0)$	$\boxed{(1,2)}$

The two PNE are illustrated via boxes.

Example 1.4 (Coordination Game). Setting as in BoS, but this time both like Bach more.

	Bach	Stravinsky
Bach	$\boxed{(2,2)}$	$(0,0)$
Stravinsky	$(0,0)$	$\boxed{(1,1)}$

As in BoS there are still two PNE, despite the fact that both like Bach more.

Example 1.5 (Prisoners-Dilemma). Two members of a criminal organization are arrested and imprisoned. Each prisoner is in solitary confinement with no means of communicating with the other. The prosecutors lack sufficient evidence to convict the pair on the principal charge, but they have enough to convict both on a lesser charge. Simultaneously, the prosecutors offer each prisoner a bargain. Each prisoner is given the opportunity either to betray the other by confessing that the other committed the crime, or to cooperate with the other by not confessing and remaining silent. The possible outcomes are:

- If A and B each betray the other, each of them serves three years in prison
- If A betrays B but B remains silent, A will be set free and B will serve four years in prison
- If A remains silent but B betrays A, A will serve four years in prison and B will be set free
- If A and B both remain silent, both of them will serve only one year in prison (on the lesser charge).

Every player wants to minimize the time (maximize negative time) spent in prison.

	confess	not confess
confess	$\boxed{(-3,-3)}$	$(0,-4)$
not confess	$(-4,0)$	$(-1,-1)$

Even though both would be better off if both do not confess, there is a unique pure Nash equilibrium in which both confess and go for 3 years in prison.

Example 1.6 (Matching Pennies). Every player selects either head or tail. If both players choose differently, player 1 pays 1 Euro to player 2; If both players choose equally, player 2 pays 1 Euro to player 1.

	Head	Tail
Head	$(1,-1)$	$(-1,1)$
Tail	$(-1,1)$	$(1,-1)$

This game is a strict competitive two-player game with $u_1(a) = -u_2(a)$ for all $a \in A$. There is no (pure) Nash equilibrium.

Best-Response-Correspondence

Define

$$S_{-i} := \times_{j \in N \setminus \{i\}} S_j.$$

Definition 1.7 (Best-Response-Correspondence). For all $s_{-i} \in S_{-i}$, define the best-response $B_i : S_{-i} \rightrightarrows 2^{S_i}$ of player $i \in N$ as

$$B_i(s_{-i}) := \{s_i \in S_i \ : \ u_i(s_i', s_{-i}) \leq u_i(s_i, s_{-i}) \ \forall s_i' \in S_i\}.$$

With this definition we get an alternative characterization of Nash equilibria.

Lemma 1.8. $s^* \in S$ is a Nash equilibrium for the strategic game $G = (N, S, u)$ iff $s_i^* \in B_i(s_{-i}^*)$ for all $i \in N$.

This leads to an algorithm computing a NE:

1. Compute $B_i(\cdot)$ for all $i \in N$.
2. Compute s^* with $s_i^* \in B_i(s_{-i}^*)$ for all $i \in N$.

If the best-response-correspondences are singletons, the above algorithms reduces to solving a system of n equations in n variables.

Example 1.9 (Nash equilibrium via best-response-correspondence). Mark the best-response strategies of every player with a *. A joint profile is a NE iff it is market with two stars.

	L	C	R
T	$(1, 2^*)$	$(2^*, 1)$	$(1^*, 0)$
M	$\boxed{(2^*, 1^*)}$	$(0, 1^*)$	$(0, 0)$
B	$(0, 1)$	$(0, 0)$	$\boxed{(1^*, 2^*)}$

Definition 1.10 (Dominance). In $G = (N, S, u)$ a strategy $s_i \in S_i$ of player $i \in N$ is strictly dominated by $s_i' \in S_i$, if

$$u_i(s_i, s_{-i}) < u_i(s_i', s_{-i}) \quad \forall s_{-i} \in S_{-i}.$$

$s_i \in S_i$ is weakly dominated by $s_i' \in S_i$, if

$$u_i(s_i, s_{-i}) \leq u_i(s_i', s_{-i}) \quad \forall s_{-i} \in S_{-i}$$

and there is $s_{-i} \in S_{-i}$, such that

$$u_i(s_i, s_{-i}) < u_i(s_i', s_{-i}).$$

Remark 1.11. A strictly dominated strategy $s_i \in S_i$ is never a best-response to any s_{-i} and will therefore never occur in a NE.

Thus, we can w.l.o.g. remove all strictly dominated strategies for any $i \in N$ from the set S_i without loosing NE.

Example 1.12 (Prisoners-Dilemma). For both players: the strategy "not confess" is strictly dominated by "confess".

	confess	not confess
confess	(-3,-3)	$(0, -4)$
not confess	$(-4, 0)$	$(-1, -1)$

1.2 Zero-Sum Games and Existence of Pure Nash Equilibria

We consider strict competitive two-player games.

Definition 1.13 (Zero-Sum Game). A strategic two-player game $G = ([2], S, u)$ is called zero-sum game, if it is strictly competitive, that is, $u_1 = -u_2$.

The utility functions are complementary to each other:

$$u_1(s) \leq u_1(s') \Leftrightarrow u_2(s) \geq u_2(s') \quad \forall s, s' \in S \text{ und } u_1(s) + u_2(s) = 0 \quad \forall s \in S.$$

For zero-sum games, we will represent u_2 solely via u_1: Player 1 wants to maximize u_1 while player 2 wants to minimize u_1. Associate with every $k \in S_1$ the minimum guaranteed utility of player 1 regardless of what player 2 plays. A maxmin strategy k^* of player 1 maximizes this minimum guaranteed utility among all strategies:

$$k^* \in \arg\max_{k \in S_1} \min_{\ell \in S_2} u_1(k, \ell).$$

Definition 1.14. Let $G = ([2], S, u)$ be a finite zero-sum game. $x^* \in S_1$ is maxmin strategy of player 1, if:

$$k^* \in \arg\max_{k \in S_1} \min_{\ell \in S_2} u_1(k, \ell).$$

$\ell^* \in S_2$ is a minmax strategy of player 2, if:

$$\ell^* \in \arg\min_{\ell \in S_2} \max_{k \in S_1} u_1(k, \ell).$$

Observe that the following inequalities hold for k^*:

$$\min_{\ell \in S_2} u_1(k^*, \ell) \geq \min_{\ell \in S_2} u_1(k, \ell) \quad \forall k \in S_1.$$

ℓ^* satisfies the following inequalities:

$$\max_{k \in S_1} u_1(k, \ell^*) \leq \max_{k \in S_1} u_1(k, \ell) \quad \forall \ell \in S_2.$$

Example 1.15. Consider the following example with entries corresponding to the u_1 values:

	$\boxed{L}$	C	$\boxed{R}$
T	2	0	1
$\boxed{M}$	$\boxed{2}$	3	$\boxed{2}$
B	1	2	0

The maxmin strategy of player 1 is M. The minmax strategies of player 2 are L und R. The NE are (M, L) and (M, R).

The following theorem shows, that any NE must consist of maxmin and minmax strategies, respectively

Theorem 1.16. Let $G = ([2], S, u)$ be a finite zero-sum game. (k^*, ℓ^*) is a NE for G with value $v^* := u_1(k^*, \ell^*)$ iff k^* is a maxmin strategy of player 1 and ℓ^* is a minmax strategy of player 2, and

$$v^* = \max_{k \in S_1} \min_{\ell \in S_2} u_1(k, \ell) = \min_{\ell \in S_2} \max_{k \in S_1} u_1(k, \ell).$$

Exercise 1.17. Prove Theorem 1.16.

Implications of Theorem 1.16 for two-player zero-sum games:

1. Every NE has the same value (assuming there exists one).
2. Nash strategies are exchangable: If (k, ℓ) and (k', ℓ') are NE of G, then also (k, ℓ') and (k', ℓ).

1.3 Mixed Nash Equilibria

Let $G = (N, (S_i), (u_i))$ be a strategic game with $|N| = n$. We introduce the following notation:

- The elements of S_i are the pure strategies of player $i \in N$.
- Let $\Delta(S_i) := \{x_i \in \mathbb{R}_+^{m_i} | \sum_{j \in S_i} x_{ij} = 1\}$ be the set of all probability distributions over S_i and let $m_i := |S_i| \in \mathbb{N}$ denote the number of such pure strategies for player $i \in N$. A vector $x_i \in \Delta(S_i)$ is called mixed strategyof player i and the entry x_{ij} is the probability that i chooses the pure strategy $j \in S_i$.
- $x := (x_1, \dots, x_n) \in \Delta := \times_i \Delta(S_i)$, $n = |N|$, is called mixed strategy profile.
- Every $x = (x_1, \dots, x_n) \in \Delta$ defines a joint probability distribution on $S := \times_i S_i$: The probability that a particular pure profile $s = (s_1, \dots, s_n) \in S$ realizes is given as:

$$x(s) := \prod_{i=1}^{n} x_{is_i}.$$

Using this notation, we can now define the mixed extension of a strategic game G.

Definition 1.18. The mixed extension of a strategic game $G = (N, S, u)$ is given by $(N, \Delta(S), U)$, where $\Delta(S_i)$ is defined as above and $U_i : \Delta \to \mathbb{R}$ is the expected utility function of player i that assigns to every mixed strategy profile $x \in \Delta$ the expected utility $U_i(x)$ of player i. For finite games we have for a mixed strategy profile x:

$$U_i(x) := \sum_{s \in S} u_i(a) \cdot x(a).$$

Remark 1.19. The function $x_i \mapsto U_i(x_i, x_{-i})$ is linear on $\Delta(S_i)$.

Von Neumann und Morgenstern initiated in 1944 the first systematic research of modeling strategic behavior by randomization. If the preferences of players are modeled by expectation values we speak of von Neumann/Morgenstern preferences (short: vNM preferences). A mixed Nash equilibrium can now be defined as follows:

Definition 1.20. A mixed Nash equilibrium of a strategic game $G = (N, S, u)$ is a Nash equilibrium of the mixed extension of G. That is, a mixed strategy profile $x^* \in \Delta$ is a mixed Nash equilibrium of G iff for every $i \in N$ we have:

$$U_i(x^*_{-i}, x^*_i) \geq U_i(x^*_{-i}, x_i) \quad \forall x_i \in \Delta(S_i).$$

Analogously to the case of pure strategies, we can define the notion of maxmin and minmax strategies.

Corollary 1.21. Let $G = ([2], S, u)$ be a finite zero-sum game and $([2], \Delta, U)$ be the mixed extension. The mixed strategy $(x^*, y^*) \in \Delta$ is a mixed NE with value $v^* := U_1(x^*, y^*)$, iff x^* is a minmax strategy of player 1 and y^* is a maxmin strategy of player 2 with value

$$v^* = \max_{x \in \Delta(S_1)} \min_{y \in \Delta(S_2)} U_1(x, y) = \min_{y \in \Delta(S_2)} \max_{x \in \Delta(S_1)} U_1(x, y).$$

Exercise 1.22. Prove Corollary 1.21.

1.4 Zero-Sum Games and Mixed NE

We consider a zero-sum game $G = (\{1, 2\}, S, u)$. In the example (Matching Pennies), we have seen that pure NE may not exist. Let the strategy sets of the players 1 and 2 be given as:

$$S_1 := (s_{11}, s_{12}, \ldots, s_{1m}) \quad \text{and} \quad S_2 := (s_{21}, s_{22}, \ldots, s_{2n}).$$

(We use in this section the number n for the number of strategies of player 2 and not, as before, for the number of players in N.) We define a utility matrix $U := (u_{ij})_{i \in [m], j \in [n]} \in \mathbb{R}^{m \times n}$, such that

$$u_{ij} = u_1(s_{1i}, s_{2j}) = -u_2(s_{1i}, s_{2j}).$$

Let $x = (x_1, \dots, x_m)$ be a mixed strategy of player 1 and $y = (x_1, \dots, x_n)$ be a mixed strategy of player 2. The expected utility of player 1, given the mixed strategies x and y, is given as:

$$U_1(x, y) := \sum_{i=1}^{m} \sum_{j=1}^{n} u_{ij} x_i y_j = x^T U y.$$

(we use vectors as column vectors if not stated otherwise.) If player 1 uses x, then the guaranteed expected utility for player 2 is

$$\min_{\text{stoch. } y} x^T U y.$$

With Corollary 1.21 we get that in any mixed NE, player 1 plays a mixed strategy x maximizing the guaranteed expected utility.

The following lemma shows that among all mixed strategies that are best-responses, there is always a pure best-response.

Lemma 1.23. Let x be a mixed strategy of player 1 and y a mixed strategy of player 2. The following holds true.

$$\min_{y \text{ stoch.}} x^T U y = \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} x_i \text{ and } \max_{x \text{ stoch.}} x^T U y = \max_{i \in [m]} \sum_{j=1}^{n} u_{ij} y_j.$$

Proof. Define $t(x) := \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} x_i$. Since y is a stochastic vector, we get for all such y:

$$x^T U y = \sum_{i=1}^{m} \sum_{j=1}^{n} u_{ij} x_i y_j = \sum_{j} \left(y_j \cdot \left(\sum_{i} u_{ij} x_i \right) \right) \geq \sum_{j} y_j \cdot t(x) = t(x) = \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} x_i.$$

Moreover:

$$\min_{y} x^T U y \leq \sum_{i} u_{ij} x_i \quad \forall j \in [n],$$

as every unit vector $e_j = (0, \dots, 0, 1, 0, \dots, 0)$, $j \in [n]$, is stochastic. Both inequalities show the first claim. The second follows analogously. □

With the above lemma we get: in a mixed NE, player 1 wants to choose $x \in \mathbb{R}^m$ so that it solves:

$$\begin{aligned} \max \quad & \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} x_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} x_i = 1 \\ & x_i \geq 0 \quad \forall i \in [m]. \end{aligned}$$

Rformulating, we get (LP1):

$$\begin{array}{lrcl} \max & z & & \\ \text{s.t.} & z - \sum_{i=1}^{m} u_{ij} x_i & \leq & 0 \quad \forall j \in [n] \\ & \sum_{i=1}^{m} x_i & = & 1 \\ & x_i & \geq & 0 \quad \forall i \in [m]. \end{array}$$

As we maximize z, for an optimal solution $(\hat{x}_1, \ldots, \hat{x}_m, \hat{z})$ of (LP1), we get:

$$\hat{z} = \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} \hat{x}_i.$$

Analogously, player 2 chooses a strategy solving:

$$\begin{array}{lrcl} \min & \max_{i \in [m]} \sum_{j=1}^{n} u_{ij} y_j & & \\ \text{s.t.} & \sum_{j=1}^{n} y_j & = & 1 \\ & y_j & \geq & 0 \quad \forall j \in [n] \end{array}$$

which is equivalent to (LP2):

$$\begin{array}{lrcl} \min & w & & \\ \text{s.t.} & w - \sum_{j=1}^{n} u_{ij} y_j & \geq & 0 \quad \forall i \in [m] \\ & \sum_{j=1}^{n} y_j & = & 1 \\ & y_j & \geq & 0 \quad \forall j \in [n]. \end{array}$$

We get to the following central result of this section.

Theorem 1.24 (Minimax-Theorem, of Neumann (1928)). For every matrix $U \in \mathbb{R}^{m \times n}$ there are stochastic vectors $\hat{x} \in \mathbb{R}^m$ and $\hat{y} \in \mathbb{R}^n$ with:

$$\hat{x}^T U \hat{y} = \min_{y \text{ stoch.}} \hat{x}^T U y = \max_{x \text{ stoch.}} x^T U \hat{y}.$$

Moreover, every two player zero-sum game $G = (\{1, 2\}, S, u)$ admits a mixed NE.

Proof. Consider (LP1) and its dual (LP2). Both LPs are feasible, hence we can use the strong duality theorem: There are optimal solutions $(\hat{x}_1, \ldots, \hat{x}_m, \hat{z})$ for (LP1) and $(\hat{y}_1, \ldots, \hat{y}_m, \hat{w})$ for (LP2), such that: $\hat{z} = \hat{w}$. The first statement follows from

$$\hat{z} = \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} \hat{x}_i \quad \text{and} \quad \hat{w} = \max_{i \in [m]} \sum_{j=1}^{n} u_{ij} \hat{y}_j$$

and using Lemma 1.23.
For the existence of a mixed NG, we choose a minimax solution $\hat{x}, \hat{y}$ w.r.t. U. We obtain the following series of inequalities for $y \in \Delta(S_2)$:

$$U_1(\hat{x}, \hat{y}) = \hat{x}^T U \hat{y} = \hat{z} = \min_{j \in [n]} \sum_{i=1}^{m} u_{ij} \hat{x}_i \leq U_1(\hat{x}, y).$$

With $U_2(\hat{x}, \hat{y}) = -U_1(\hat{x}, \hat{y})$this is equivalent to:

$$U_2(\hat{x}, \hat{y}) \geq U_2(\hat{x}, y),$$

and therefore $\hat{y}$ is a best-response of player 2. On the other hand, for arbitrary $x \in \Delta(S_1)$, we have:

$$U_1(\hat{x}, \hat{y}) = \hat{x}^T U \hat{y} = \hat{w} = \max_{i \in [m]} \sum_{j=1}^{n} u_{ij} \hat{y}_j \geq x^T U \hat{y} = U_1(x, \hat{y}).$$

Hence $\hat{x}$ is a best response of player 1. □

Remark 1.25. The proof shows (cf. Theorem 1.16):

$$\max_x \min_y \; x^T U y = \min_y \max_x \; x^T U y.$$

1.5 Existence of Mixed NE for n Players

We consider now strategic games with an arbitrary number of players $n \in \mathbb{N}$. Nash's famous theorem [7] for the existence of mixed equilibria in finite games relies on a fixed point theorem due to Kakutani. The main idea is to consider for each player i the best-reply correspondence that maps each strategy profile x to the set of strategies that maximize player i's payoff when her opponents play x_{-i}. Kakutani's fixed point theorem [4] then implies the existence of a fixed point, which is a mixed Nash equilibrium.[1] We define a best-response-correspondence as we have seen before.

Definition 1.26. Let $G = (N, S, u)$ be a finite strategic game and for all $i \in N$ let $\Delta_{-i} := \times_{j \in N \setminus \{i\}} \Delta(S_j)$. For $x_{-i} \in \Delta_{-i}$ define the best-response-correspondence $B_i : \Delta_{-i} \rightrightarrows \Delta(S_i)$ as

$$B_i(x_{-i}) := \{x_i \in \Delta(S_i) \;:\; U_i(x_{-i}, x_i) \geq U_i(x_{-i}, x_i') \; \forall x_i' \in \Delta(S_i)\}.$$

A mixed profile $x^* \in \Delta$ is a mixed NE for G, if $x_i^* \in B_i(x_{-i}^*) \; \forall i \in N$. With the definition of

$$B : \Delta \rightrightarrows \Delta, x \mapsto (B_i(x_{-i}))_{i \in N},$$

we are asking for a fixpoint of $B(x)$ over Δ, i.e.:

[1] In his book Nash [8] uses an alternative proof relying on Brouwer's fixed point theorem which circumvents dealing with set-valued functions.

Search for $x \in \Delta$ with $x \in B(x)$.

Definition 1.27. A set S in $\mathbb{R}^n$ is compact, if it is bounded and closed. S is convex, if for all $x, y \in S$ and all $\lambda \in [0,1]$ we have

$$\lambda x + (1-\lambda)y \in S.$$

Theorem 1.28 (Kakutani '41)**.** Let $S \subseteq \mathbb{R}^k$ be non-empty. Let $f : S \rightrightarrows S$ be a correspondence from S to S, $x \in S \mapsto f(x) \subseteq S$, such that the following conditions hold:

1. S is compact and convex.
2. f is non-empty for all $x \in S$ and a convex-valued correspondence: for all $x \in S$ the set $f(x) \subseteq S$ is convex.
3. f has a closed graph: $(x_k, y_k) \rightarrow (x, y)$ with $y_k \in f(x_k)$ for all $k \in \mathbb{N}$ implies $y \in f(x)$.

Then, f has a fixpoint, i.e., there is $x \in S$ with $x \in f(x)$.

We use Kakutani's Fixpunkt in order to show the existence of mixed Nash equilibria.

Theorem 1.29. Let $G = (N, S, u)$ be an n-player finite strategic game. Then, the mixed extension (N, Δ, U) of G admits a mixed NE.

Proof. We show that the best-response correspondence $B(x)$ satifies the conditions of Theorem 1.28.
For 1:
Note that $\Delta(S_i)$ is a simplex of dimesnion $|S_i| - 1$ and hence convex and compact and so is the Cartesian product $\times_{i \in N} \Delta(S_i)$.
For 2:
Per definition

$$B_i(x_{-i}) = \arg \max_{x_i \in \Delta(S_i)} U_i(x).$$

As U_i is linear in x_i, U_i is continuous and with the Theorem of Weierstrass (see Optimization I+II), we get that for all x_{-i} the set $B_i(x_{-i})$ is non-empty and thus $B(x)$ is non-empty for all $x \in \Delta$. Furthermore, $B(x)$ – as finite Cartesan product of the sets $B_i(x_{-i}), i \in N$ – is convex, if for all $i \in N$ the set $B_i(x_{-i})$ is convex. To show the latter, let $x, y \in B_i(x_{-i})$, this is equivalent with

$$U_i(x, x_{-i}) \geq U_i(z, x_{-i}) \text{ for all } z \in \Delta(S_i)$$
$$U_i(y, x_{-i}) \geq U_i(z, x_{-i}) \text{ für alle } z \in \Delta(S_i).$$

Let $\lambda \in [0,1]$ be arbitrary. We obtain

$$\lambda U_i(x, x_{-i}) + (1-\lambda) U_i(y, x_{-i}) \geq U_i(z, x_{-i}) \text{ für alle } z \in \Delta(S_i).$$

With the linearity of U_i we get

$$U_i(\lambda x + (1-\lambda)y, x_{-i}) \geq U_i(z, x_{-i}) \text{ for all } z \in \Delta(S_i),$$

which implies $\lambda x + (1-\lambda)y \in B_i(x_{-i})$.
For 3:
It remains to show that $B(x)$ has a closed graph. Suppose the graph of $B(x)$ is not closed. Hence there exists a sequence $(x_k, y_k) \to (x, y)$ with $y_k \in B(x_k)$ but $y \notin B(x)$, i.e., there exists $i \in N$ with $y_i \notin B_i(x_{-i})$. Hence, there is $y_i' \in \Delta(S_i)$ and $\epsilon > 0$ with

$$U_i(y_i', x_{-i}) > U_i(y_i, x_{-i}) + 3\epsilon. \tag{1.1}$$

Because $x_{-i}^k \to x_{-i}$ and with continuity of U_i we get for k large enough:

$$U_i(y_i', x_{-i}^k) \geq U_i(y_i', x_{-i}) - \epsilon. \tag{1.2}$$

With $(x_k, y_k) \to (x, y)$ and with continuity of U_i we get for k large enough:

$$U_i(y_i, x_{-i}) \geq U_i(y_i^k, x_{-i}^k) - \epsilon. \tag{1.3}$$

All three inequalities (1.1),(1.2),(1.3) yield:

$$U_i(y_i', x_{-i}^k) \underset{(1.2)}{\geq} U_i(y_i', x_{-i}) - \epsilon \underset{(1.1)}{>} U_i(y_i, x_{-i}) + 2\epsilon \underset{(1.3)}{\geq} U_i(y_i^k, x_{-i}^k) + \epsilon.$$

This contradicts $y_i^k \in B_i(x_{-i}^k)$. □

The proof allows for the following generalization towards Euclidian games.

> **Theorem 1.30** (Euclidian games). Let $G = (N, S, u)$ be a strategic game with $S_i \subseteq \mathbb{R}^{m_i}, m_i \in \mathbb{N}$ non-empty, convex and compact for all $i \in N$. The functions $u_i : S \to \mathbb{R}$ are continuous and quasi-concave in x_i for all $i \in N$. Then, G admits pure NE.

Chapter 2

Computation of Mixed NE

We describe two algorithms to compute a mixed Nash equilibrium in a finite, two-player strategic game. The first algorithm is based on the complete enumeration of support sets of a mixed NE. The second more efficient algorithm is due to Lemke and Howson and we will use LH for short to refer to it. We show that LH finds a Nash equilibrium after a finite number of steps and thus also prove the existence of mixed Nash equilibria in two-player games.

2.1 Support Enumeration

Let $G = ([2], (S_i), (u_i))$ be a finite two-player strategic game. It will be convenient to assume that G is given as follows:

$$S_1 = M := \{1, \dots, m\} \quad \text{and} \quad S_2 = N := \{m+1, \dots, m+n\},$$

and u_1 is given by a utility matrix $A \in \mathbb{R}^{m \times n}$ and u_2 is given by a utility matrix $B \in \mathbb{R}^{m \times n}$. We assume that player 1 is a row player and player 2 is a column player. The mixed strategies of player 1 (resp. player 2) are denoted by $x \in \mathbb{R}^m$ (resp. $y \in \mathbb{R}^n$).
We next define the support of a stochastic vector.

Definition 2.1. Let $x \in \mathbb{R}^d$ be a stochastic vector. The support of x is defined as

$$\text{supp}(x) := \{i \in [d] \ : \ x_i > 0\}.$$

Recall:

- A best response of player 1 to a mixed strategy y of player 2 is a mixed strategy x that maximizes the expected utility $x^T Ay$.
- Similarly, a best response of player 2 to a mixed strategy x of player 1 is a mixed strategy y that maximizes $x^T By$.
- A mixed Nash equilibrium is a mixed stratgy profile (x, y) where x and y are best responses to each other.

The following theorem, which we call the best-response-condition, will be very useful subsequently.

Theorem 2.2 (Best-Response Condition (BRC)). Let x and y be mixed strategies of player 1 and player 2, resp. Then x is a best response of player 1 to y iff for all $i \in M$ we have

$$x_i > 0 \qquad \Longrightarrow \qquad (Ay)_i = u = \max\{(Ay)_k \;:\; k \in M\}$$

Analogously, y is a best response of player 2 to x iff for all $j \in N$ we have

$$y_j > 0 \qquad \Longrightarrow \qquad (x^T B)_j = v = \max\{(x^T B)_k \;:\; k \in N\}$$

Said differently, the mixed strategy x of player 1 is a best response to y iff all pure strategies in the support are pure best responses to y (note that $(Ay)_i = e_i^T Ay$). Similarly, the mixed strategy y of player 2 is a best response to x iff all pure strategies in the support are pure best responses to x.

Proof of Theorem 2.2. The expected utility of player 1 is

$$x^T Ay = \sum_{i \in M} x_i (Ay)_i = \sum_{i \in M} x_i (u - (u - (Ay)_i)) = u - \sum_{i \in M} x_i (u - (Ay)_i)$$

where the last equality follows since x is stochastic. Note that by the definition of u and since $x_i \geq 0$, we have $x^T Ay \leq u$. Moreover, equality holds only iff $x_i > 0$ implies that $u - (Ay)_i = 0$. An analogous argument proves the second part of the claim. $\square$

	4	5
1	$(3,3)$	$(3,2)$
2	$(2,2)$	$(5,6)$
3	$(0,3)$	$(6,1)$

(a) Representation as bimatrix game

$$A = \begin{pmatrix} 3 & 3 \\ 2 & 5 \\ 0 & 6 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 3 & 2 \\ 2 & 6 \\ 3 & 1 \end{pmatrix}.$$

(b) Representation with matrices A, B

Figure 2.1: Bimatrix game for support enumeration.

Example 2.3. Consider the following game in Fig. 2.1. Suppose $I = \{1,2\} \subseteq M$ is the support of x and $J = \{1,2\} \subseteq N$ is the support of y. Then player 2 must be indifferent between its two columns 1 and 2, i.e.,

$$3x_1 + 2x_2 = 2x_1 + 6x_2 \quad \Leftrightarrow \quad x_1 = 4x_2,$$

which together with $x_1 + x_2 = 1$ yields: $x_1 = 4/5$ and $x_2 = 1/5$. Similarly, player 1 must be indifferent between its two rows 1 and 2. That is

$$3y_4 + 3y_5 = 2y_4 + 5y_2 \quad \Leftrightarrow \quad y_4 = 2y_5,$$

which together with $y_4 + y_5 = 1$ yields $y_4 = 2/3$ and $y_5 = 1/3$. Thus, (x,y) with $x = (4/5, 1/5, 0)$ and $y = (2/3, 1/3)$ is a Nash equilibrium.

1: Guess the support (I, J) with $I \subseteq M$ and $J \subseteq M$.
2: Solve the equations

$$\sum_{i\in I} x_i b_{ij} = \nu \quad \forall j \in J \qquad \text{and} \qquad \sum_{i\in I} x_i = 1$$
$$\sum_{j\in J} a_{ij} y_j = u \quad \forall i \in I \qquad \text{and} \qquad \sum_{j\in J} y_j = 1$$

3: Verify whether $x \geq 0$, $y \geq 0$ and BRC is satisfied for x and y.

Algorithm 2.1.1: Computing Nash equilibria by complete enumeration

The BRC actually provides a first way to compute mixed Nash equilibria for finite games: Simply try all possible supports, solve a system of equations and verify whether BRC is satisfied (see Algorithm 2.1.1).

2.2 The Lemke-Howson Algorithm

We now capture the best responses of each player by means of a best repsonse polyhedron. Define

$$\bar{P}_1 := \{(x, \nu) \in \mathbb{R}^m \times \mathbb{R} \;:\; (x^T B)_j \leq \nu \;\; \forall j \in N,\; \mathbb{1}^T x = 1,\; x \geq 0\},$$
$$\bar{P}_2 := \{(y, u) \in \mathbb{R}^n \times \mathbb{R} \;:\; (Ay)_i \leq u \;\; \forall i \in M,\; \mathbb{1}^T y = 1,\; y \geq 0\}.$$

Consider $\bar{P}_2$. We give some intuition: The constraints $(Ay)_i \leq u \;\; \forall i \in M$ require that u is at least as large as the expected utility for each pure strategy of player 1. The two remaining constraints, $\mathbb{1}^T y = 1$ and $y \geq 0$, ensure that y is stochastic.

Definition 2.4. A point (y, u) of $\bar{P}_2$ has label $k \in M \cup N$, if the corresponding defining inequality of $\bar{P}_2$ is binding, i.e., if either $k = i \in M$ and $(Ay)_i = u$ (meaning i is a best reponse to y) or $k = j \in N$ and $y_j = 0$ (meaning j is not in the support). Analogously, a point (x, ν) of $\bar{P}_1$ has label $k \in M \cup N$ if either $k = j \in N$ and $(x^T B)_j = \nu$ (meaning j is a best reponse to x) or $k = i \in M$ and $x_i = 0$ (meaning i is not in the support).

Theorem 2.5. The mixed strategy profile (x, y) is a Nash equilibrium with expected utilities ν and u for player 1 and player 2, respectively, iff the points $(x, \nu) \in \bar{P}_1$ and $(y, u) \in \bar{P}_2$ together have all labels in $M \cup N$.

Proof. Suppose there is a missing label, say k, for $(x, \nu) \in \bar{P}_1$ and $(y, u) \in \bar{P}_2$. Suppose $k = i \in M$ (the case $k = j \in N$ is proved analogously). We then have $(Ay)_i < u$ since i is not a label of (y, u); also, $x_i > 0$ since i is not a label of (x, ν). But this violates the BRC and thus (x, y) is not a Nash equilibrium.
Conversely, if every label in $M \cup N$ appears for $(x, \nu) \in \bar{P}_1$ and $(y, u) \in \bar{P}_2$, then the BRC is satisfied and thus (x, y) is a Nash equilibrium with expected utilities ν and u for player 1 and player 2, respectively. □

We now normalize the polyhedra $\bar{P}_1$ and $\bar{P}_2$ by eliminating the utility values u and ν. The described transformation works if we make the following assumption:

Assumption 2.6. A and B^T are non-negative and have no all-zero column.

This assumption is without loss of generality, since adding a large positive constant to every entry of A (or B) does not change the structure of the corresponding Nash equilibria. Assumption 2.6 implies in particular that u and v are positive. By dividing all constraints by u and v, respectively, and dropping the stochastic constraints, we obtain the following two normalized polyhedra:

$$P_1 := \{x \in \mathbb{R}^m \;:\; (x^T B)_j \le 1 \;\; \forall j \in N, \;\; x \ge 0\},$$
$$P_2 := \{y \in \mathbb{R}^n \;:\; (Ay)_i \le 1 \;\; \forall i \in M, \;\; y \ge 0\}.$$

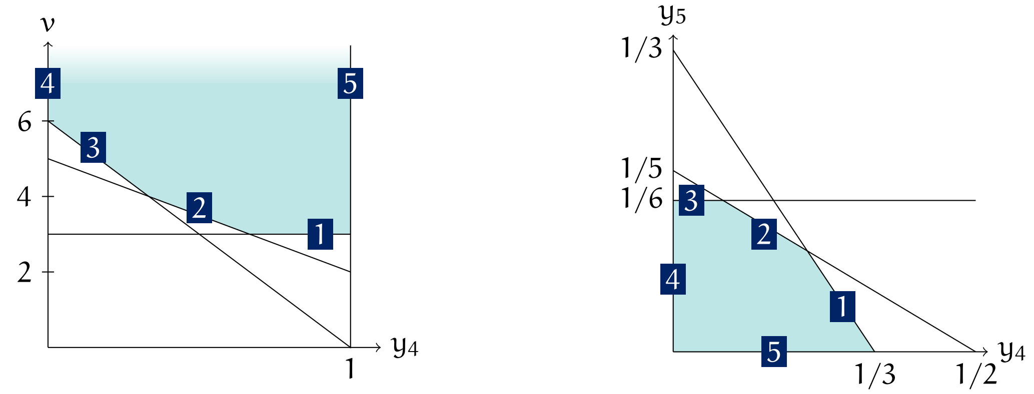
The vectors in the normalized best response polyhedra are no longer stochastic, but there is a direct correspondence between the extreme points of $\bar{P}_i$ and P_i, except for the zero vector. For each extreme point (x, v) of $\bar{P}_1$, x/v is an extreme point of P_1 (analogously for P_2). For each extreme point x of P_1, apart from the zero vector, $(\frac{x}{x^T \mathbb{1}}, \frac{1}{x^T \mathbb{1}})$ is an extreme point of $\bar{P}_1$. To see this statement, define a bijection between $\bar{P}_1$ and $P_1 \setminus \{0\}$ by $(x, v) \mapsto x \cdot (1/v)$. Similarly, $(y, u) \mapsto y \cdot (1/u)$ is a bijection that maps $\bar{P}_2 \to P_2 \setminus \{0\}$. These transformations have the property that a binding inequality in $\bar{P}_1$ (resp. $\bar{P}_2$) is a binding inequality in P_1 (resp. P_2) and vice versa. As a consequence, points have the same labels, which are defined by the binding inequalities. In the following, for a non-zero vector x, we write $\bar{x}$ as a shorthand for $\frac{x}{x^T \mathbb{1}}$.
We proceed to give a characterization of those extreme points of P_1 and P_2 that together constitute an equilibrium of the underlying game. For given extreme points x and y, let $L(x)$ and $L(y)$ denote the label sets of x and y, respectively, that is,

$$L(x) = \{i \in M : x_i = 0\} \cup \{j \in N : (x^T B)_j = 1\},$$
$$L(y) = \{j \in N : y_j = 0\} \cup \{i \in M : (Ay)_i = 1\}.$$

We call a pair $(x, y) \in P_1 \times P_2$ fully labeled if $L(x) \cup L(y) = M \cup N$.
Theorem 2.5 thus reduces to: (x, y) is a Nash equilibrium iff the pair $(x, y) \in P_1 \times P_2 \setminus \{(0, 0)\}$ has all labels, i.e., for every label $i \in M$ the respective binding inequality is in $x \ge 0$ or $Ay \le 1$, and for every label $j \in N$ the respective binding inequality is in $y \ge 0$ or $x^T B \le 1$. Computing a Nash equilibrium thus reduces to finding a pair of points $(x, y) \in P_1 \times P_2 \setminus \{(0, 0)\}$ that has all labels.
From Assumption 2.6, it follows that P_1 and P_2 are bounded m-dimensional and n-dimensional polyhedra, respectively. A bounded polyhedron is called a polytope. A d-dimensional polytope is simple if every vertex (i.e., extreme point) meets exactly d of the defining inequalities with equality.

Assumption 2.7. The polytopes P_1 and P_2 are simple.

A game that does not satisfy this assumption is called degenerate. Non-degeneracy is of course not always satisfied but there is a standard way, called perturbations, to make degenerate games non-degenerate. However, we do not address this issue here, but instead assume that Assumption 2.7 holds.
For a point x of a polytope P, let $L(x)$ be the set of all defining inequalities of P that are binding for x - this set will be the label set of x. The following are basic facts from polyhedral theory.



(a) Best-Response Polyhedra (b) Normalized best-response polytope

Figure 2.2: Best-Response polyhedron $\bar{P}_2$ and normalized Best-Response polyhedron P_2 for player 2 for the gam of Example 2.3. **(a)** The bounding halfspaces are marked with their corresponding label. The halfspaces with label 1 to 3 enforce that labels 1 to 3 of player 2 give no larger utility than u; The halfspaces with labels 4 and 5 correspond to the constraints that y_4 and y_5 must be nonnegative. In Fig. **(b)** there is the normalized polytope whith $u = 1$.

Fact 2.8. In a simple d-dimensional polytope:

1. Every vertex v has $|L(v)| = d$, i.e., is incident to exactly d faces.
2. For two distinct vertices v, v' we have $L(v) \neq L(v')$.
3. Every vertex is incident to exactly d edges; in particular, for each $\kappa \in L(v)$, there is a unique neighbor v' of v with $L(v') \cap L(v) = L(v) \setminus \{\kappa\}$.

We state an immediate consequence of Assumption 2.7.

Lemma 2.9. In a non-degenerate game, for a pair of vertices (x, y), we have $|L(x)| = m$ and $|L(y)| = n$.

Proof. P_1 and P_2 are simple m- and n-dimensional polytopes, thus, Fact 2.8 implies the lemma. □

The application of Fact 2.8(c) to polytope P_1 (or P_2) and vertex v will be called removing label k from v and obtaining new vertex v', where k is the label corresponding to inequality κ. Similarly, since vertex v' has exactly one new label k' that v did not have, namely the one corresponding to the unique inequality in $L(v') \setminus L(v)$, we say that label k' was added. That is, if we move from v along an edge of the polytope to v', we remove label k and add a new label k'.

We are now ready to formulate the Lemke-Howson Algorithm LH. In the description given below, x is a vertex of P_1 and y is vertex of P_2. The algorithm starts from an artificial equilibrium $(x, y) = (0, 0) \in P_1 \times P_2$, which we consider a completely labeled pair. The algorithm now chooses an arbitrary label k_0 of x, called the missing label and alternately

$x \leftarrow 0$ and $y \leftarrow 0$
Let k_0 be any label of x
$k \leftarrow k_0$
loop
  In P_1, remove label k from x; let x' be the new vertex and k' the label added
  $x \leftarrow x'$
  If $k' = k_0$, stop looping
  In P_2, remove label k' from y; let y' be the new vertex and k'' the label added
  $y \leftarrow y'$
  If $k'' = k_0$, stop looping
  $k \leftarrow k''$
end loop
return $(\bar{x}, \bar{y})$

Algorithm 2.2.1: The Lemke-Howson Algorithm LH

follows edges of P_1 and P_2, keeping the vertex in the other polytope fixed, until a pair of vertices (x, y) is found that has all labels. By removing label $k = k_0$, we move along an edge of P_1 to a new vertex x'. The newly added label now occurs twice at (x', y) (since every label was present at (x, y)). This new label is removed in the other polytope P_2 which yields a new vertex y' and newly added label k''. As we prove below, by repeating this procedure, we eventually must collect the missing label k_0 and thus have identified a Nash equilibrium.

Theorem 2.10. LH outputs a Nash equilibrium.

Proof. We call each time we apply Fact 2.8(c) and update the point x or y a pivot.

Definition 2.11. Define a configuration to be a pair (x, y) such that x is a vertex of P_1, y is a vertex of P_2, and (x, y) has every label in $(M \cup N) \setminus \{k_0\}$.

Remark 2.12. Using induction, it is easy to prove that at every point during the execution of the algorithm, (x, y) forms a configuration.

We call two configurations (x, y) and (x', y') adjacent if either

1. $x = x'$ and an edge of P_2 connects y to y', or
2. $y = y'$ and an edge of P_1 connects x to x'.

Thus, every pivot moves from one configuration to an adjacent configuration.
We distinguish the following two configurations:

1. (x, y) has all labels: This configuration is adjacent to exactly one other configuration, since either x or y has k_0 (but not both) and we thus must remove it from whichever vertex has it.
2. (x, y) shares a duplicate label: This configuration is adjacent to exactly two other configurations, since we can remove the duplicate label from x and pivot on P_1 or from y and pivot on P_2.

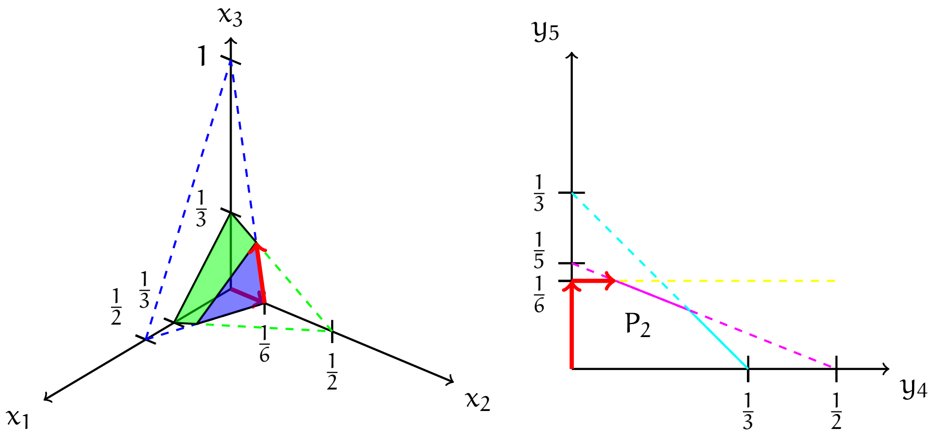


Figure 2.3: Run of the Lemke-Howson-Algorithm for the game in example 2.3. The green face of P_1 (left) is defined via the fourth inequality, the blue via the fifth. For P_2 the light blue edge corresponds to inequality 1, the magenta colored edge to inequality 2 and the yellow one to inequality 3. The missing label is 2. The pair of inal extreme points (x, y) is $x = (0, 1/8, 1/4)^T$, $y = (1/12, 1/6)^T$.

The above observations lead to the following viewpoint. Consider the graph whose nodes are all configurations and whose edges are all pairs of adjacent configurations. The above insights show that every node has degree 1 or 2. Therefore, every connected component is a path or a cycle. Viewed in this graph, LH starts from the configuration $(0, 0)$, which is an endpoint of a path since it has all labels. The algorithm then moves along the path until it finds the other endpoint of the path, which then again must have all labels. The proof now follows from Theorem 2.5. □

Remark 2.13. Observe that an endpoint cannot be one of the configurations $(0, 0)$, $(x, 0)$ or $(0, y)$.

A direct consequence of the above proof is that the configuration graph has an even number of degree-1 nodes and every such node corresponds to a Nash equilibrium, except $(0, 0)$. Thus, the proof of the following corollary is straight-forward:

Corollary 2.14. A non-degenerate, finite two-player strategic game has a finite, odd number of mixed Nash equilibria. In particular, there is at least one mixed Nash equilibrium.

The algorithm of Lemke and Howson is guaranteed to find a Nash equilibrium after a finite number of steps. However, there are examples that show that the number of needed pivots can be exponential.

2.3 The Tableau Method

To apply the tableau method to find Nash equilibria using the Lemke-Howson algorithm, we use the following four steps.

1. Initialization of tableau.

2. Repeated pivoting.

3. Recover Nash equilibrium from final tableau.

In the tableau method, we introduce slack variables, and use the terminology basic and non-basic variables. For our purposes the basic variables and set of labels have opposite meanings since labels imply a tight inequality and basic variables are not tight. Hence, "enters the basis" means the same as "label is removed" and "leaves the basis" means that "label is added".

Step 1. Initialization.

For the purposes of solving the game we need two tableaus, one for each player. Let r_i be the slack in the constraint $A_i y \leq 1$ and let s_j be the slack in the constraint $x^T B_j \leq 1$. We then obtain the system

$$Ay + r = \mathbb{1},\ x^T B + s = \mathbb{1},\ \text{where } x, y, r, s \text{ are non negative.}$$

In the initial tableau, the basis is $\{r_i | i \in M\} \cup \{s_j | j \in N\}$ since we start with $(0, 0)$ and hence all inequalities are not tight leading to $r_i > 0, s_j > 0$ for all $i \in M, j \in N$. As an example we will use the following game.

	4	5	6
1	$(1,2)$	$(3,1)$	$(0,0)$
2	$(0,1)$	$(0,3)$	$(2,1)$
3	$(2,0)$	$(1,0)$	$(1,3)$

The initial tableau are $r = \mathbb{1} - Ay$;

$$r_1 = 1 \ -y_4 - 3y_5 \tag{A1}$$
$$r_2 = 1 \qquad\qquad -2y_6 \tag{A2}$$
$$r_3 = 1 - 2y_4 - y_5 \ \ -y_6 \tag{A3}$$

and $s = \mathbb{1} - x^T B$;

$$s_4 = 1 - 2x_1 - x_2 \tag{B1}$$
$$s_5 = 1 \ -x_1 - 3x_2 \tag{B2}$$
$$s_6 = 1 \qquad - x_2 \ -3x_3 \tag{B3}$$

Step 2: Pivoting.

The main feature of the Lemke-Howson algorithm, as we discussed in the previous section, is that the variable which just left the basis determines the variable to enter the basis next. There are $m + n$ complementary pairs of variables: $\{r_i, x_i\}$ for $i \in M$ and $\{s_j, y_j\}$ for $i \in N$ of the form

$$r_i x_i = 0, i \in M \qquad s_j y_j = 0, j \in N.$$

The condition $r_i x_i = 0$ requires that $x_i > 0$ can only occur in a NE if the strategy $i \in M$ is a best response. This is only true if $r_i = 0$. Otherwise, if $r_i > 0$, the corresponding strategy $i \in M$ is not a best response, hence for a NE, we need to have $x_i = 0$.
Initially, for $(0, 0)$ all complementarity conditions are satisfied. We keep performing pivots until the complementarity conditions are again satisfied, or equivalently, we search for a pair (x, y) that has all labels in $M \cup N$. We start in the artificial point $(0, 0)$ that has all labels. This point is obviously no NE and we choose an arbitrary label $k_0 \in N \cup M$ and bring x_{k_0} or y_{k_0} (depending on whether $k_0 \in M$ or $k_0 \in N$) into the basis. Therefore, we loose k_0. In the example, let us choose $k_0 = 1$, hence, x_1 is now a basis variable. Intuitively we increase x_1 from 0 on until an inequality $0 \leq \mathbb{1} - x^T B$ becomes tight. The index of such an inequality is contained in $\arg\min_{j \in N}\{\frac{1}{b_{1,j}} | b_{1,j} > 0\}$ (min-ratio rule) and determines the variable leaving the basis. In the example, s_4 leaves the current basis. Keep in mind that the nominator in the min-ratio rule corresponds to the value of the constant of the right hand side of the respective inequality and this value may change during the course of the algorithm. In general, the pivot row is determined via the minimum ratio of the constant on the right hand side of the tableau and the coefficients (with positive sign) on the right hand side. Therefore we solve (B1) for x_1, obtaining a new equation (B'1), and we substitute the new equation into (B2) and (B3) obtaining

$$
\begin{aligned}
x_1 &= 1/2 - 1/2 s_4 - 1/2 x_2 && \text{(B'1)}\\
s_2 &= 1/2 + 1/2 s_4 - 5/2 x_2 && \text{(B'2)}\\
s_3 &= 1 \qquad\qquad - x_2 \quad - 3 x_3 && \text{(B'3)}
\end{aligned}
$$

In this case, since s_4 just left the basis, y_4 must be brought in. Examining the (A) tableau we see that r_3 is the winner of the min-ratio rule, and is therefore to leave the basis. We obtain the following.

$$
\begin{aligned}
r_1 &= 1/2 + 1/2 r_3 - 5/2 y_5 + 1/2 y_6\\
r_2 &= 1 \qquad\qquad\qquad\qquad\quad - 2 y_6\\
y_4 &= 1/2 - 1/2 r_3 - 1/2 y_5 - 1/2 y_6
\end{aligned}
$$

Since r_3 left, now x_3 enters the other tableau, and by the min-ratio rule s_6 leaves.

$$
\begin{aligned}
x_1 &= 1/2 - 1/2 s_4 - 1/2 x_2\\
s_5 &= 1/2 + 1/2 s_4 - 5/2 x_2\\
x_3 &= 1/3 \qquad\qquad - 1/3 x_2 - 1/3 s_6
\end{aligned}
$$

Since s_6 left, now y_6 enters, and by the min-ratio rule r_2 leaves.

$$
\begin{aligned}
r_1 &= 3/4 + 1/2 r_3 - 5/2 y_5 - 1/4 r_2\\
y_6 &= 1/2 \qquad\qquad\qquad\qquad\quad - 1/2 r_2\\
y_4 &= 1/4 - 1/2 r_3 - 1/2 y_5 + 1/4 r_2
\end{aligned}
$$

Since r_2 left, now x_2 enters the other tableau, and by the min-ratio rule s_5 leaves.

$$
x_1 = 2/5 \quad - 3/5 s_4 + 1/5 s_5
$$

$$x_2 = 1/5 \quad +1/5s_4 - 2/5s_5$$
$$x_3 = 4/14 - 1/15s_4 + 2/15s_5 - 1/3s_6$$

Since s_5 left, now y_5 enters the other tableau, and by the min-ratio rule r_1 leaves.

$$y_5 = 3/10 + 1/5r_3 - 2/5r_1 - 1/10r_2$$
$$y_6 = 1/2 \qquad\qquad\qquad -1/2r_2$$
$$y_4 = 1/10 - 3/5r_3 + 1/5r_1 + 3/10r_2$$

Step 3: Output.

Since x_1 was the initial variable to enter the basis, and r_1 just left, the complementarity conditions are now satisfied. (More generally, if x_i was the initial variable to enter, we stop when x_i or its complement leaves.) In a tableau, we obtain values for the basic variables by setting the non-basic variables to zero. Hence the values of the variables are

$$r = (0, 0, 0), s = (0, 0, 0), x = (2/5, 1/5, 4/15), y = (1/10, 3/10, 1/2).$$

Therefore, the Nash equilibrium we just found is

$$(\bar{x}, \bar{y}) = ((6/13, 3/13, 4/13), (1/9, 3/9, 5/9)).$$

Degeneracy

The effect of non-degeneracy on the tableau method is the following: When running the Lemke-Howson algorithm in tableau form on a nondegenerate game, in each iteration there is a unique variable that wins the min-ratio test. However, degenerate games occur frequently in practice. In a general game, we can still use the tableau method, but we will be faced with the problem of breaking ties in some manner. Furthermore, just as in the simplex algorithm, if we have a "bad" tie-breaking rule, then our program can enter a loop and run forever. There are, however, other tie-breaking rules similar to the lexicographic simplex variant that are guaranteed to terminate.

2.4 Some Complexity Theory

After the non-constructive proof of the existence of mixed Nash equilibria for general n-player games in chapter 1 (Theorem 1.29) we have now seen two algorithms fro actually computing (mixed) Nash equilibria in 2 player games: Complete support enumeration (Algorithm 2.1.1) and the Lemke-Howson Algorithm (Algorithm 2.2.1). Both these algorithms, however, exhibit an exponential worst-case runtime. In this section we want to justify this runtime by showing that calculating Nash equilibria is a hard problem.
We start by recalling the definitions of the complexity classes P, NP and NPC (cf. Optimierung III):

Definition 2.15. Let Π, Π' be two decision problems.

1. Problem Π is a member of the class P (deterministic polynomial time) if there exists an polynomial time algorithm for Π. That is an algorithm which for every

instane $I \in \Pi$ decides whether this instance is a Yes- or a No-instance *and* which has a runtime bounded by some polynomial in the encoding length of the instance I.

2. Problem Π is a member of the class NP (non-deterministic polynomial time) if every Yes-instance I has a polynomial size certificate proving that I is a Yes-instance *and* it is possible to verify the correctness of such a certificate in polynomial time.

3. We say that problem Π can be reduced to problem Π' if there exists a transformation $f : \Pi \to \Pi'$ such that for every instance $I \in \Pi$ we have

   - $f(I)$ can be constructed in polynomial time and
   - $f(I)$ is a Yes-instance of Π' if and only if I is a Yes-instance of Π.

   We then write $\Pi \leq \Pi'$ and also say that "Π is easier/not harder than Π'".

4. Problem Π is an element of the class NPC (NP-complete) if it is a member of NP and every problem in NP can be reduced to Π, i.e.

$$\Pi \in \text{NPC} :\iff \Pi \in \text{NP and for all } \Pi' \in \text{NP} : \Pi' \leq \Pi.$$

Apparently, we have $P \subseteq NP$. However, it is still an open problem whether this is a proper inclusion (i.e. whether $P = NP$ or $P \neq NP$ is true). Most mathematicians assume that $P \neq NP$, i.e. not every problem in NP can be solved in polynomial time. Since NPC contains the hardest problems from NP this would, in particular, imply that there can be no polynomial time algorithm for an NP-hard problem. One such NP-hard problem is SAT:

SAT
Input: A set of m clauses $C_1, \dots, C_m$ in n variables $x_1, \dots x_n$. Hereby, clauses are expressions of the form $C_i = y_{i_1} \vee y_{i_2} \vee \dots \vee y_{i_{k_i}}$, where $y_{i_j} \in \{x_{i_j}, \bar{x}_{i_j}\}$ are so called literals.
Question: Is there an interpretation of the variables such that all clauses are satisfied simultaneously. In other words: Can we set each variable to either TRUE or FALSE in such a way that every clause contains at least one literal which then evaluates to TRUE.

As our goal is to show that computing (mixed) Nash equilibria is hard, we first have to formulate an appropriate decision problem. A straightforward way to do that would be the following:

NASH
Input: A finite 2-player game G
Questions: Does G have a (mixed) Nash equilibrium?

However, this problem is actually a very simple decision problem: According to Theorem 1.29 *every* instance of this problem is a Yes-instance (since every finite n-player game has a Nash equilibrium). Thus, NASH is a member of P (and, therefore, not of NPC unless P = NP). At the same time, this doe not tell us anything about whether *computing* Nash equilibria is hard or not since Theorem 1.29 allows us to solve the decision problem NASH without actually computing any Nash equilibria.

This motivates the following variations of NASH, in which we don't just ask for the existence of *any* Nash equilibrium but for the existence of a Nash equilibrium with some additional property (which does not always exist). Intuitively, by doing that we want to enforce a closer link between answering the question of the given decision problem and actually computing an equilibrium.

NASH+$_1$
Input: A finite 2-player game G and some number $k \in \mathbb{Q}$
Questions: Does G have a Nash equilibrium (x^*, y^*) with $U_1(x^*, y^*) \geq k$?

NASH+$_2$
Input: A finite 2-player game G and some number $k \in \mathbb{Q}$
Questions: Does G have a Nash equilibrium (x^*, y^*) with $U_1(x^*, y^*) + U_2(x^*, y^*) \geq k$?

NASH+$_3$
Input: A finite 2-player game G and some number $k \in \mathbb{N}$
Questions: Does G have a Nash equilibrium (x^*, y^*) in which player 1 uses at least k different (pure) strategies?

NASH+$_4$
Input: A finite 2-player game G und some strategy $s \in S_1$
Questions: Does G have a Nash equilibrium (x^*, y^*) with $x^*_s > 0$?

NASH+$_5$
Input: A finite 2-player game G und some strategy $s \in S_1$
Questions: Does G have a Nash equilibrium (x^*, y^*) with $x^*_s = 0$?

NASH+$_6$
Input: A finite 2-player game G
Questions: Does G have at least two different Nash equilibria?

We will now show that each of these six decision problems is NP-complete.

Theorem 2.16. The decision problems NASH+$_1$, ..., NASH+$_6$ are NP-complete.

Proof. We will start with NASH+$_6$.
NASH+$_6$$\in$ NP is obvious (a certificate just consists of two different Nash equilibria – checking whether they are in fact Nash equilibria (and different from each other) can certainly be done in polynomial time).
So it remains to show that every problem in NP can be reduced to NASH+$_6$. To do that we will show that SAT $\leq$ NASH+$_6$. Since SAT is NP-complete itself (i.e. every problem in NP can be reduced to SAT) this directly implies that every problem in NP can be reduced to NASH+$_6$.

Step 1: We start by describing the transformation $f : \text{SAT} \to \text{NASH+}_6$.

Let I be an instance of SAT consisting of a set of variables V with cardinality $n := |V|$, a corresponding set of literals $L := \{+v, -v \mid v \in V\}$ and a set of clauses $C \subseteq \mathcal{P}(L)$. We now define a symmetric 2-player game $G = ([2], S, u)$. The strategy sets of both players are $S_1 := S_2 := V \dot\cup L \dot\cup C \dot\cup \{\perp\}$ and the utility functions are defined as follows

$$u_1(v, \ell) := u_2(\ell, v) := \begin{cases} 0, & \ell \in \{+v, -v\} \\ n, & \text{else} \end{cases} \qquad \text{f.a. } v \in V, \ell \in L$$

$$u_1(\ell, \ell') := u_2(\ell', \ell) := \begin{cases} n-4, & \ell = -\ell' \\ n-1, & \text{else} \end{cases} \qquad \text{f.a. } \ell, \ell' \in L$$

$$u_1(c, \ell) := u_2(\ell, c) := \begin{cases} 0, & \ell \in c \\ n, & \text{else} \end{cases} \qquad \text{f.a. } c \in C, \ell \in L$$

$$u_1(s, s') := u_1(s', s) := n-4 \qquad \text{f.a. } s \in V \cup L \cup C, s' \in V \cup C$$

$$u_1(s, \perp) := u_2(\perp, s) := 0 \qquad \text{f.a. } s \in V \cup L \cup C$$

$$u_1(\perp, s) := u_2(s, \perp) := n-1 \qquad \text{f.a. } s \in V \cup L \cup C$$

$$u_1(\perp, \perp) := u_2(\perp, \perp) := \varepsilon,$$

where ε is some positive number < 1. An example for the resulting utility matrix of player 1 is shown in figure 2.4.

Step 2: Next, we determine all Nash equilibria in the game $f(I)$ constructed in step 1. Specifically, we will show:

a) $(\perp, \perp)$ is a (pure) Nash equilibrium with $u_1(\perp, \perp) = \varepsilon$.

b) For any satisfying interpretation $\{\ell_1, \dots, \ell_n\} \subseteq L$ of I the strategy profile $(\bar{x}, \bar{y})$ with $\bar{x}_{\ell_i} = \bar{y}_{\ell_i} = \frac{1}{n}$ for all $i = 1, \dots, n$ is a mixed Nash equilibrium with utility $U_1(\bar{x}, \bar{y}) = n - 1$.

c) There are no other (pure or mixed) Nash equilibria.

Regarding a): We have $u_1(\perp, \perp) = \varepsilon > 0 = u_1(s, \perp)$ for every alternative strategy $s \in V \cup L \cup C$ of player 1. Thus, player 1 has no incentive to deviate. The same is true for player 2. Therefore, $(\perp, \perp)$ is a Nash equilibrium.

Regarding b): Let $\{\ell_1, \dots, \ell_n\} \subseteq L$ be a satisfying interpretation of I. In particular, we then have $\ell_i \neq -\ell_j$ for all $i, j \in [n]$ (otherwise this would be an inconsistent

	x_1	x_2	$+x_1$	$-x_1$	$+x_2$	$-x_2$	$x_1 \vee -x_2$	$-x_1 \vee x_2$	f
x_1	$n-4$	$n-4$	0	0	n	n	$n-4$	$n-4$	0
x_2	$n-4$	$n-4$	n	n	0	0	$n-4$	$n-4$	0
$+x_1$	$n-4$	$n-4$	$n-1$	$n-4$	$n-1$	$n-1$	$n-4$	$n-4$	0
$-x_1$	$n-4$	$n-4$	$n-4$	$n-1$	$n-1$	$n-1$	$n-4$	$n-4$	0
$+x_2$	$n-4$	$n-4$	$n-1$	$n-1$	$n-1$	$n-4$	$n-4$	$n-4$	0
$-x_2$	$n-4$	$n-4$	$n-1$	$n-1$	$n-4$	$n-1$	$n-4$	$n-4$	0
$x_1 \vee -x_2$	$n-4$	$n-4$	0	n	n	0	$n-4$	$n-4$	0
$-x_1 \vee x_2$	$n-4$	$n-4$	n	0	0	n	$n-4$	$n-4$	0
f	$n-1$	$n-1$	$n-1$	$n-1$	$n-1$	$n-1$	$n-1$	$n-1$	ε

Figure 2.4: An example for a utility matrix of player 1 in the 2-player game constructed in the proof of Theorem 2.16. The underlying SAT-formula is $\phi = (x_1 \vee -x_2) \wedge (-x_1 \vee x_2)$, i.e. our set of variables is $V = \{x_1, x_2\}$, the set of literals is $L = \{+x_1, -x_1, +x_2, -x_2\}$ and the set of clauses is $C = \{\{+x_1, -x_2\}, \{-x_1, +x_2\}\}$.

interpretation of the literals). This implies $u_1(\ell_i, \ell_j) = n - 1$ for all i, j. Thus, we have $U_1(\bar{x}, \bar{y}) = n - 1$.

To prove that $(\bar{x}, \bar{y})$ is a Nash equilibrium we show that it satisfies the BRC. In order to do that we calculate $U_1(s, \bar{y})$ for all $s \in V \cup L \cup C \cup \{\perp\}$.

Case 1: For $s = v \in V$ there exists a unique $i \in [n]$ with $\ell_i \in \{+v, -v\}$ (as all ℓ_j together form an interpretation of I). This gives us

$$U_1(s, \bar{y}) = \frac{1}{n} u_1(v, \ell_i) + \sum_{j \neq i} \frac{1}{n} u_1(v, \ell_j) = 0 + \sum_{j \neq i} \frac{1}{n} \cdot n = n - 1 = U_1(\bar{x}, \bar{y}).$$

Case 2: For $s = \ell \in L$ we get

$$U_1(s, \bar{y}) = \sum_{j=1}^{n} \frac{1}{n} u_1(\ell, \ell_j) \leq \sum_{j=1}^{n} \frac{1}{n} \cdot (n - 1) = n - 1 = U_1(\bar{x}, \bar{y}).$$

Case 3: For $s = c \in C$ there exists at least one $i \in [n]$ with $\ell_i \in c$ (as a satisfying interpretation of I in particular satisfies clause c). This gives us

$$U_1(s, \bar{y}) = \frac{1}{n} u_1(c, \ell_i) + \sum_{j \neq i} \frac{1}{n} u_1(c, \ell_j) \leq 0 + \sum_{j \neq i} \frac{1}{n} n = n - 1 = U_1(\bar{x}, \bar{y}).$$

4. Fall: For $s = \perp$ we finally get

$$U_1(s, \bar{y}) = \sum_{j=1}^{n} \frac{1}{n} u_1(\perp, \ell_j) = \sum_{j=1}^{n} \frac{1}{n} \cdot (n - 1) = n - 1 = U_1(\bar{x}, \bar{y}).$$

This shows that $(\bar{x}, \bar{y})$ satisfies the BRC for player 1. Because of the symmetry of the game as well as the strategy profile the same also holds for player 2. Thus, $(\bar{x}, \bar{y})$ is indeed a Nash equilibrium.

Regarding c): Let (x, y) be an arbitrary mixed Nash equilibrium. We will show that it then has to be either of the form described in a) or the form described in b).

Case 1: One of the players (wlog. player 2) only plays strategy $\perp$. In that case the unique best response of player 1 is also strategy $\perp$. Thus, we must have $(x, y) = (\perp, \perp)$, i.e. the strategy profile is of the form described in a).

$\implies$ From now on we can assume that no player plays $\perp$ with propability 1.

Case 2: One of the players (wlog player 2) plays one of the strategies $s \in V \cup C$ with positive propability $x_s > 0$. We then have

$$u_1(s, s_2) + u_2(s, s_2) \leq 2n - 4 \qquad \text{f.a. } s_2 \neq \perp$$

as well as

$$u_1(s_1, s_2) + u_2(s_1, s_2) \leq 2n - 2 \qquad \text{f.a. } s_1, s_2 \neq \perp.$$

Together this shows that under the assumption that no player plays $\perp$ the expected social utility is

$$U_1(x, y|\neg\perp) + U_2(x, y|\neg\perp) \\ := \frac{1}{\sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2}} \sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2} \Big(u_1(s_1, s_2) + u_2(s_1, s_2) \Big) < 2n - 2.$$

Thus, under the assumption that no player plays $\perp$ at least one of the players has an expected utility of strictly less than $\frac{1}{2}(2n-2) = n - 1$, i.e.

$$U_i(x, y|\neg\perp) := \frac{1}{\sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2}} \sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2} u_1(s_1, s_2) < n - 1.$$

However, this player can now improve their expected utility by only playing $\perp$: We will show this only for player 1 (i.e. the case $i = 1$). The case of player 2 can be proven in exactly the same way.

$$\begin{aligned} U_1(x, y) &= \sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2} u_1(s_1, s_2) + \sum_{s_1 \neq \perp} x_{s_1} y_{\perp} u_1(s_1, \perp) \\ &\quad + \sum_{s_2 \neq \perp} x_{\perp} y_{s_2} u_1(\perp, s_2) + x_{\perp} y_{\perp} u_1(\perp, \perp) \\ &< \sum_{s_1, s_2 \neq \perp} x_{s_1} y_{s_2} (n-1) + \sum_{s_1 \neq \perp} 0 + \sum_{s_2 \neq \perp} x_{\perp} y_{s_2} (n-1) + x_{\perp} y_{\perp} u_1(\perp, \perp) \\ &= \sum_{s_2 \neq \perp} \Big(\sum_{s_1} x_{s_1} \Big) y_{s_2} (n-1) + x_{\perp} y_{\perp} u_1(\perp, \perp) \\ &\leq \sum_{s_2 \neq \perp} y_{s_2} (n-1) + y_{\perp} u_1(\perp, \perp) = \sum_{s_2 \neq \perp} y_{s_2} u_1(\perp, s_2) + y_{\perp} u_1(\perp, \perp) \\ &= U_1(\perp, y) \end{aligned}$$

Thus, (x, y) cannot be a Nash equilibrium (in contradiction to our initial assumption).

$\implies$ From now on we can assume that no player plays any strategy from $V \cup C$ with positive propability. This also implies that the utility of each player can be at most $n-1$.

Case 3: One of the players (wlog. player 2) plays strategy $\perp$ with propability $y_\perp > 0$ (but less than 1 because of case 1). Then player 1 can strictly improve their propability by switching to only playing strategy $\perp$:

$$\begin{aligned} U_1(\perp, y) &= y_\perp u_1(\perp,\perp) + \sum_{\ell \in L} y_\ell u_1(\perp, \ell) = y_\perp \varepsilon + \sum_{\ell \in L} y_\ell (n-1) \\ &> \sum_{s_1 \in L \cup \{\perp\}} x_{s_1} y_\perp u_1(s_1, \perp) + \sum_{s_1 \in L \cup \{\perp\}, \ell \in L} x_{s_1} y_\ell u_1(s_1, \ell) = U_1(x, y) \end{aligned}$$

Hence, (x, y) again was not a Nash equilibrium.

$\implies$ From now on we can assume that both players on play strategies from L.

Case 4: One of the players (wlog. player 2) plays one of the literal pairs $+v, -v$ with a combined propability of less than $\frac{1}{n}$, i.e. $y_{+v} + y_{-v} < \frac{1}{n}$. Then player 1 can strictly improve their utility by switching to only playing strategy $v \in V$ (which they do not play at all in their current strategy x because of case 3):

$$\begin{aligned} U_1(v, y) &= y_{+v} u_1(v, +v) + y_{-v} u_1(v, -v) + \sum_{\ell \in L \setminus \{\pm v\}} y_\ell u_1(v, \ell) \\ &= 0 + 0 + \sum_{\ell \in L \setminus \{\pm v\}} y_\ell n = (1 - y_{+v} - y_{-v}) \cdot n > \frac{n-1}{n} \cdot n = n - 1 \geq U_1(x, y) \end{aligned}$$

Therefore, in this case (x, y) cannot be a Nash equilibrium as well.

$\implies$ From now on we can assume that for all variables $v \in V$ we have $x_{+v} + x_{-v} = \frac{1}{n} = y_{+v} + y_{-v}$.

Case 5: There exists some pair of literals $+v, -v$ such that $x_{+v} > 0$ and $y_{-v} > 0$ or such that $x_{-v} > 0$ and $y_{+v} > 0$ hold. In both cases player 1 can strictly improve by switching to only playing strategy $\perp$. We only show this for the first case as the other one is completely analogous:

$$\begin{aligned} U_1(x, y) &= x_{+v} y_{-v} u_1(+v, -v) + \sum_{(\ell, \ell') \neq (+v, -v)} x_\ell y_{\ell'} u_1(\ell, \ell') \\ &\leq x_{+v} y_{-v} (n-4) + \sum_{(\ell, \ell') \neq (+v, -v)} x_\ell y_{\ell'} (n-1) < n - 1 = U_1(\perp, y). \end{aligned}$$

Thus, (x, y) cannot be a Nash equilibrium.

$\implies$ From now on we can assume that for any variable $v \in V$ we either have $x_{+v} = y_{+v} = \frac{1}{n}$ and $x_{-v} = y_{-v} = 0$ or $x_{+v} = y_{+v} = 0$ and $x_{-v} = y_{-v} = \frac{1}{n}$. In other words: The strategy profile (x, y) corresponds to a feasible (but not necessarily satisfying) interpretation of I (in the sense described in b)).

Case 6: The interpretation of I corresponding to the strategy profile (x, y) leaves at least one clause $c \in C$ unsatisfied, i.e. we have $y_\ell = 0$ for all $\ell \in c$. Then player 1 can strictly improve by switching to only playing strategy c (which they currently do not play by case 3):

$$U_1(c, y) = \sum_{\ell \in L, y_\ell > 0} y_\ell u_1(c, \ell) = \sum_{\ell \in L \setminus c} y_\ell n = n > n - 1 \geq U_1(x, y).$$

Again, (x, y) cannot be a Nash equilibrium.

$\implies$ From now on we can assume that (x, y) corresponse to a satisfying interpretation of I leaving us with only one possible case to consider.

Case 7: The strategy profile (x, y) now has to correspond to a satisfying interpretation of I. Thus, (x, y) is of the form described in b)

This concludes the proof of c) showing that in a) and b) we already exhaustively described all Nash equilibria of $f(I)$.

Step 3: It remains to show that the transformation $f : \text{SAT} \to \text{NASH}+_6$ is in fact a polynomial reduction. It should be obvious that the construction described in step 1 can be done in polynomial time.

For the correctness of the transformation we first assume that I is a Yes-instanceof SAT. Then I has a satisfying interpretation which, by b) of step 2, corresponse to a Nash equilibrium in $f(I)$ which is different to the Nash equilibrium described in a). Thus, $f(I)$ does in fact have at least two different Nash equilibria, i.e. $f(I)$ is a Yes-instance of $\text{NASH}+_6$.

If, on the other hand, $f(I)$ is a Yes-instance of $\text{NASH}+_6$ then it has at least two different Nash equilibria. Since at most one of them can be of the form described in a) at least one of them has to be of the form described in b) (as there are no other types of Nash equilibria by c)). This Nash equilibrium then corresponds to a satisfying interpretation of I proving that I is a Yes-instance of SAT.

Thus, we have finally proven $\text{SAT} \leq \text{NASH}+_6$. Together with the fact that $\text{NASH}+_6$ a member of the class NP, this shows that $\text{NASH}+_6$ is NP-complete.
To prove that $\text{NASH}+_1$ to $\text{NASH}+_5$ are NP-complete as well we can essentially use the same transformation as the one described in step 1. Filling in the remaining details for those proofs is left as an exercise. □

We have now seen that several decision problems closely related to the computation of Nash equilibria are hard problems (in the sense of beeing NP-complete). This already suggests that the computation of Nash equilibria itself should be a hard problem itself. In order to also show this more directly we first have to introduce a different class of problems to decision problems: The so called search problems:

Definition 2.17. We call a problem Π a search problem if it has the following form:
Input: An instance I of Π
Output: A solution x of instance I, provided such a solution exists, or the answer "no", otherwise.

Clearly, the computation of Nash equilibria is such a search problem:

FNASH
Input: A finite 2-player game G
Output: A Nash equilibrium (x, y) of G

We can get many more examples of search problems by reformulating decision problems – e.g.:

FSAT
Input: A boolean formula
Output: A satisfying interpretation (if such an interpretation exists)

Similarly to the classes P, NP and NPC for decision problems we can also define different complexity classes for search problems:

Definition 2.18. Let Π, Π' be two search problems.

1. Problem Π is a member of class FNP if it has the following two properties:
   - Whenever an instance I of Π has a solution, in has a polynomial size solution (i.e. a solution with size bounded by a polynomial in the encoding length of the instance).
   - There exists a polynomial time algorithm that for every instance I and supposed solution x can decide whether x is a solution to I.
2. Problem Π is a member of class FP if it is a member of FNP and additionally there exists an algorithm which for every instance I of Π in polynomial time computes a solution x to I (or answers "no" if no such solution exists for I).
3. Problem Π is a member of class TFNP if it is a member of FNP and every instance of Π has a solution.
4. We say that problem Π can be reduced to problem Π' (and then write $\Pi \leq \Pi'$ for short) if there exist transformations f and g which can be computed in polynomial time and such that for every instance $I \in Pi$ and every supposed solution x of an instance of Π' the following holds: x is a solution to $f(I) \in \Pi'$ if and only if $g(x)$ is a solution to $I \in \Pi$. In other words: We can use f to transform an instance of Π into one of Π' and then use g to transform any solution to the transformed instance back into one of the original instance.
5. Problem Π is a member of class FNPC (FNP-complete) if it is a member of FNP and every problem in FNP can be reduced to Π, i.e.

$$\Pi \in \text{FNPC} :\iff \Pi \in \text{FNP and for all } \Pi' \in \text{FNP} : \Pi' \leq \Pi$$

One can show that FSAT is a member of FNPC (basically with the same proof as the one showing that SAT $\in$ NPC).

Fact 2.19. FSAT is FNP-complete.

Using this fact one can easily show that FNP is probably a proper superset of FP:

Exercise 2.20. We have: $\mathrm{FP} = \mathrm{FNP} \iff \mathrm{P} = \mathrm{NP}$.

This implies that for FNP-complete search problems we should not expect to be able to solve them in polynomial time (unless we believe that $\mathrm{P} = \mathrm{NP}$). In particular: If we were able to show that FNASH is FNP-complete we would have succeeded in showing that we cannot expect to find a polynomial time algorithm for computing Nash equilibria. However, it turns out that it is rather unlikely that FNASH $\in$ FNPC is true due to the following fact:

Fact 2.21. If FNASH is FNP-complete, we have $\mathrm{NP} = \mathrm{coNP}$.

Most mathematicians, however, assume that $\mathrm{NP} \neq \mathrm{coNP}$ holds[1].
In order to still be able to say something about the hardness of FNASH we first observe that FNASH is in fact a member of the subclass TFNP of FNP (since Theorem 1.29 guarantees the existence of a Nash equilibrium for any finite game). Next we introduce a further subclass of TFNP containing all the total search problem for which the existence of a solution for all instances is proven in a certain way similar to the correctness proof of the algorithm of Lemke-Howson:

Definition 2.22. Let $\Pi \in \mathrm{FNP}$. Then Π is a member of class PPAD (polynomial parity argument, directed case) if for every instance I there exists a (possibly huge) set S_I of polynomial size supposed solution to I containing, in particular, all polynomial size actual solutions to I. Furthermore, S_I contains a special element $=$ and is equipped with two polynomial algorithms $\mathrm{prev}_I, \mathrm{next}_I : S_I \to S_I \cup \{\emptyset\}$ such that the following three properties holds:

- $x = \mathrm{prev}_I(y) \iff \mathrm{next}_I(x) = y$ for all $x, y \in S_I$
- $\mathrm{prev}_I(0) = \emptyset$
- For every $x \in S_I \setminus \{0\}$: x is a solution to $I \iff |\{\mathrm{prev}_I(x), \mathrm{next}_I(x)\}| = 1$.

Problem Π is PPAD-complete if it is a member of PPAD and for every problem $\Pi' \in$ PPAD we have: $\Pi' \leq \Pi$.

Remark 2.23. PPAD is a subset of TFNP.

Looking at the proof for the algorithm of Lemke-Howson we can directly see that FNASH $\in$ PPAD.[2] With quite a lot additional work one can even show that FNASH is PPAD-complete.

---

[1] P=NP would imply NP=coNP. NP=coNP would not imply P=NP. Thus, NP≠coNP is a stronger assumption compared to P≠NP

[2] There is, in fact, one non-obvious aspect here: Namely, the definition of the algorithms/maps prev_I and next_I. While it is easy to compute both neighbouring configurations for any given configuration, it is not clear how to consistently determine which of those should be the previous one and which the next one. During the algorithm of Lemke-Howson this is not a problem. Since we always know from

Fact 2.24. FNash is PPAD-complete.

Thus, FNash is a hardest problem in the class PPAD. In order to deduce from this fact that FNash is a hard problem, it remains to show that the class PPAD also contains other problems which are considered to be hard to solve. We will present two such examples here. The first one is the problem of finding (approximate) fixed points of continuous functions:

> εBrouwer
> Input: A continuous map $f : \Delta \to \Delta$ (where $\Delta := \{x \in \mathbb{R}^2_{\geq 0} \,|\, x_1 + x_2 \leq 1\}$ denotes the unit triangle), a Lipschitz constant $L \in \mathbb{R}_{\geq 0}$ and some approximation bound $\varepsilon > 0$.
> Output: A pair of points $x, y \in \Delta$ for which the Lipschitz constant L does not hold or an approximate fix point $x \in \Delta$, i.e. some $x \in \Delta$ such that $\|f(x) - x\|_\infty \leq \varepsilon$.
> Here $\|y\|_\infty := \max\{|y_1|, |y_2|\}$ denotes the uniform norm.

In the second problem we consider a triangle together with a triangulation of said triangle (i.e. the big triangle is covered by a grid of small triangles). Our goal is to color the corners of smaller triangles with three colors (red, blue and yellow) in such a way that the following properties hold:

- No corner on the right side of the large triangle is colored blue,
- vo corner on the bottom side of the large triangle is colored red
- and no corner on the third side of the large triangle is colored yellow.

We call such a coloring a feasible coloring. An example for such a coloring can be seen in figure 2.5.

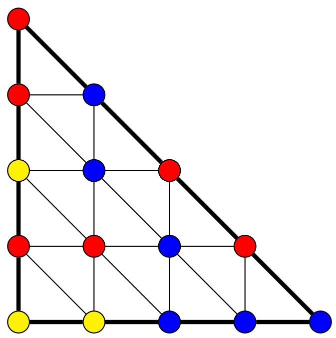


Figure 2.5: An example for a feasible coloring

which configuration we came, we also immediately know which configuration is the next one (the other neighbouring one) . However, the definition of the class PPAD requires us to be able to tell for any given configuration what it's next and previous configurations are without considering the whole set S_I (as this would not be possible within polynomial time). It turns out that there is a way to achieve this by calculating appropriate determinants (cf. "A Note on the Lemke-Howson Algorithm" by Lloyd S. Shapley (1974)) and, thus, conclude the proof for FNash ∈ PPAD – we will, however, not explore this in more details.

SPERNER
Input: A triangulated unit triangle together with a feasible coloring.
Output: A small triangle of the triangulation where all three corners have different colors.

Theorem 2.25. We have SPERNER $\in$ PPAD.

Theorem 2.26. We have εBROUWER $\leq$ SPERNER.

Both theorems will be proven on the exercise sheets. To be able to formally prove those we have to think carefully about how exactly instances of the two problems will be encoded. In particular for SPERNER it is important to encode an instance in such a way that the encoding size does not depend linearly on the number of small triangles but only logarithmically. Additionally, we must be able to efficiently obtain the three corners' colors as well as the three neighbouring triangles for any given small triangle. We will see an example for such an efficient encoding in the proof of Theorem 2.26.
Taking both these theorems as given, we now know that computinng Nash equilibria (i.e. the problem FNASH) is at least as hard as computing approximate fixed points (i.e. the problem εBROUWER). On the other hand in the proof of Theorem 1.29 we used a fixed point theorem[3] to show the existence of Nash equilibria. So, intuitively, both problems should be equally hard – which, as it turns out, is in fact true:

Fact 2.27. εBROUWER and SPERNER are PPAD-complete.

[3] We used the Kakutani fixed-point theorem here – however there also exist proofs using the Brouwer fixed-point theorem.

Chapter 3

Potential Games

The notion of potential functions is one of the most powerful tools in order to derive sufficient conditions for the existence of equilibria and for the convergence of improvement dynamics. Potential functions were introduced in the game theory literature by Rosenthal (1973) and further studied and generalized by Monderer and Shapley (1996).

3.1 Improving Moves

Let $G = (N, S, u)$ be a strategic game. If $x \in S$ is not a NE, then there is $y_i \in S_i$ for some $i \in N$ with

$$u_i(y_i, x_{-i}) > u_i(x).$$

We call the change from x_i to y_i of player i an improving move. This leads to the following algorithm 3.1.1.

Input: arbitrary $x \in S$
Output: NE $x^* \in S$
1: $x^0 := x$
2: $k := 0$
3: **while** x^k is not a NE **do**
4: compute $i \in N$ and $y_i \in S_i$ with $u_i(y_i, x^k_{-i}) > u_i(x^k)$
5: $x^{k+1} := (y_i, x^k_{-i})$
6: $k := k + 1$
7: **end while**
8: **Return** x^k

Algorithm 3.1.1: IMPROVING MOVES

Definition 3.1. We define an improvement graph $\mathcal{G}(G) = (S, E)$, where S is the node set and $e = (x, y) \in E \subseteq S \times S$ iff there is $i \in N$ with $y = (y_i, x_{-i})$. A path in $\mathcal{G}(G)$ is a sequence $\gamma = (x^0, x^1, \dots)$, such that for all $k \geq 1$ there is a unique $i \in N$ with $x^k = (y, x^{k-1}_{-i})$ for $y \in S_i$ with $y \neq x^{k-1}_i$. The profile x^0 is the start vertex of γ, and if γ is finite, the last vertex is the end vertex. γ is called improvement path of $\mathcal{G}(G)$ if for all $k \geq 1$, we have $u_i(x^k) > u_i(x^{k-1})$, where i is the unique deviating player of x^k w.r.t. x^{k-1}.

Remark 3.2. IMPROVING MOVES creates an improvement path in $\mathcal{G}(G)$.

Definition 3.3. A strategic game G has the "finite improvement property (FIP)", if every improvement path is finite.

3.2 Potential Functions and Characterization of the FIP

Definition 3.4 (Potential game). A strategic game $G = (N, S, u)$ is an exact potential game, if there is a function

$$P : S \to \mathbb{R}$$

such that for all $i \in N, x_{-i} \in S_{-i}$ and all $x_i, y_i \in S_i$:

$$u_i(y_i, x_{-i}) - u_i(x_i, x_{-i}) = P(y_i, x_{-i}) - P(x_i, x_{-i}). \tag{3.1}$$

G is a b-potential game if there is $b = (b_i)_{i \in N} \in \mathbb{R}^n_{>0}$ of positive weights and a function $P : S \to \mathbb{R}$ such that for all $i \in N, x_{-i} \in S_{-i}$ and all $x_i, y_i \in S_i$:

$$u_i(y_i, x_{-i}) - u_i(x_i, x_{-i}) = b_i \left(P(y_i, x_{-i}) - P(x_i, x_{-i}) \right). \tag{3.2}$$

G is an ordinal potential game, if for all $i \in N, x_{-i} \in S_{-i}$ and all $x_i, y_i \in S_i$:

$$\begin{aligned} & u_i(y_i, x_{-i}) - u_i(x_i, x_{-i}) > 0 \\ \Leftrightarrow\ & P(y_i, x_{-i}) - P(x_i, x_{-i}) > 0. \end{aligned} \tag{3.3}$$

G is a generalized ordinal potential game, if for all $i \in N, x_{-i} \in S_{-i}$ and all $x_i, y_i \in S_i$:

$$\begin{aligned} & u_i(y_i, x_{-i}) - u_i(x_i, x_{-i}) > 0 \\ \Rightarrow\ & P(y_i, x_{-i}) - P(x_i, x_{-i}) > 0. \end{aligned} \tag{3.4}$$

We define the neighbourhood of x via the set of tuples $\{(y_i, x_{-i}) | y_i \in S_i, i \in N\}$, where only one player (with index i) deviates. x is a local maximum of P, if

$$P(x) \geq P(y_i, x_{-i}), \text{ for all } y_i \in S_i, i \in N.$$

Theorem 3.5. Let P be an ordinal potential function for G. Then the set of NE coincides with the set of local maxima of P. If P attains its maximum on S, then G admits a NE.

Theorem 3.6. Every finite ordinal potential game G has the FIP and thus admits a NE.

Proof. For every improvement path $\gamma = (y_0, y_1, \dots)$, we get

$$P(y_0) < P(y_1) < \dots$$

As S is finite, γ is finite. □

Example 3.7. For a strategic game G, having the FIP is not necessarily equivalent to being an ordinal potential game. Consider the following example.

	c	d
a	(1, 0)	(2, 0)
b	(2, 0)	(0, 1)

The game has the FIP, but does not allow for an ordinal potential function as otherwise the following inequalities would need to hold:

$$P(a, c) < P(b, c) < P(b, d) < P(a, d) = P(a, c).$$

Theorem 3.8. Let G be a finite strategic game. Then, G has the FIP iff G is a generalized ordinal potential game.

Proof. The direction $\Leftarrow$ is trivial.
$\Rightarrow$: We consider a (directed) subgraph of $\mathcal{G}(G)$, the so-called improvement graph $\mathcal{G}^{imp}(G) = (S, E)$. The vertices coincide with S and there is a directed edge $e = (x, y) \in E \subseteq S \times S$ iff there is $i \in N$ with $y = (y_i, x_{-i}), y_i \in S_i$ and $u_i(y_i, x_{-i}) > u_i(x)$. Under FIP the Digraph is acyclic, hence we can use a topological sorting as the wanted generalized potential.

> Recall (Optimization II): A topological sorting of $G = (V, E)$ is a renumbering of the nodes in $V(G) = \{v_1, v_2, ..., v_n\}$ such that $e = (v_i, v_j) \Rightarrow i < j$ for all $e \in E(G)$.

□

3.3 Characterization of Exact Potential Games

Definition 3.9. A strategic game $G = (N, S, (u_i)_{i \in N})$ is a

- coordination game if there is a function $\bar{u} : S \to \mathbb{R}$ such that $u_i = \bar{u}$ for all $i \in N$.
- dummy game, if for all $i \in N$ and all $x_{-i} \in S_{-i}$ there is $k \in \mathbb{R}$ with $u_i(x_i, x_{-i}) = k$ for all $x_i \in S_i$, or equivalent, if for all $i \in N$, $x_{-i} \in S_{-i}$ and $x_i, y_i \in S_i$:

$$u_i(x_i, x_{-i}) = u_i(y_i, x_{-i}).$$

Theorem 3.10. Let $G = (N, S, (u_i)_{i \in N})$ be a strategecic game. $G = (N, S, (u_i)_{i \in N})$ is an exact potential game iff there are functions $(c_i)_{i \in N}$ and $(d_i)_{i \in N}$ with:

- $u_i = c_i + d_i$ for all $i \in N$,
- $(N, S, (c_i)_{i \in N})$ is a coordination game,
- $(N, S, (d_i)_{i \in N})$ is a dummy game.

Proof. $\Leftarrow$:
c of the coordination game is an exact potential function: $u_i(x) - u_i(y_i, x_{-i}) = c_i(x) - c_i(y_i, x_{-i}) + d_i(x) - d_i(y_i, x_{-i}) = c(x) - c(y_i, x_{-i})$.
$\Rightarrow$: Let Pbe an exact potential for G. For all $i \in N$, we subdivide u_i as: $u_i = P + (u_i - P)$. We get that $(N, S, (P)_{i \in N})$ is a coordination game. Let $i \in N$ and $x_{-i} \in S_{-i}$. Then,

$$u_i(x_i, x_{-i}) - u_i(y_i, x_{-i}) = P(x_i, x_{-i}) - P(y_i, x_{-i})$$
$$\Rightarrow u_i(x_i, x_{-i}) - P(x_i, x_{-i}) = u_i(y_i, x_{-i}) - P(y_i, x_{-i}).$$

Hence, $(N, S, (u_i - P)_{i \in N})$ is a dummy game. □

Theorem 3.11. Let $G = (N, S, (u_i)_{i \in N})$ be a strategic game with exact potentials P and Q. Then, $P - Q$ is constant.

Proof. Let $i \in N$. With the previous theorem $u_i - Q$ and $u_i - P$ do not depend on the strategy chosen by i (dummy game). Hence, $(P - Q) = (u_i - Q) - (u_i - P)$ does also not depend on the strategy chosen by i. As $i \in N$ was arbitrary, we get that $(P - Q)$ is constant. □

Chapter 4

Congestion Games

In this chapter, we consider a general class of strategic games related to the strategic use of scarce resources. Congestion games have become a standard game-theoretic model describing the allocation of exhaustible resources by selfish players. In the basic model of Rosenthal [10], there is a finite and non-empty set of players and resources and each player is associated with a set of allowable subsets of resources. A pure strategy of a player consists of choosing one subset from the allowable sets. Once all players have chosen a subset, there are congestion effects on the resources due to their usage by possibly several players. This congestion effect is modeled by a load-dependent cost function which solely depends on the number of players using the resource. Let us just mention one obvious application of this model in the context of traffic networks. Suppose the resources correspond to edges of a graph, the allowable subsets correspond to the simple paths connecting a source and a sink and commuters (players) choose minimum cost paths, where the cost of an edge depends on its congestion (see Fig. 4.1 for an illustration).

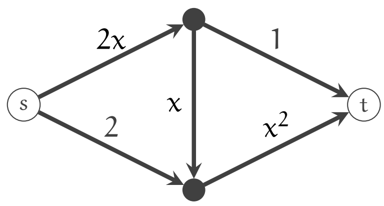


Figure 4.1: A network with edges having load-dependent congestion functions. Resources correspond to the edges of the graph and the allowable subsets of resource correspond to s-t path.

4.1 Generic Congestion Games

We introduce first the most general congestion model from which several well-known congestion models will be derived as special cases.

For an integer $k \in \mathbb{Z}_{\geq 0}$, let $[k] := \{1, \dots, k\}$. For a non-empty set $R = [m], m \in \mathbb{Z}_{>0}$ we denote by 2^R the set-system of all subsets of R including the empty set. Thus, the number of elements of 2^R is given by 2^m. For two sets R, T with $R = [m]$, it is convenient to write the elements of T^R as $|R|$-dimensional vectors of the form $x = (x_r)_{r \in R}$. For $S \subset R = [m]$, the incidence vector $\mathbf{1}^S \in \mathbb{R}^m$ is defined as

$$\mathbf{1}^S_r = \begin{cases} 1, & \text{if } r \in S \\ 0, & \text{if } r \in R \setminus S. \end{cases}$$

Let $N = [n]$ be a finite set of players and $R = [m]$ be a finite set of m resources. For each player i, the set of strategies available to player i is an arbitrary set $X_i \subseteq \mathbb{R}^m$. We call $x = (x_1, \ldots, x_n)$ with $x_i \in X_i$ for all $i \in N$ a strategy profile and $X = X_1 \times \cdots \times X_n$ the strategy space. We use standard game theory notation; for a strategy profile $x \in X$, we write $x = (x_i, x_{-i})$ meaning that x_i is the strategy that player i plays in x and x_{-i} is the partial strategy profile of all players except i. Every strategy profile $x = (x_1, \ldots, x_n) \in X$ induces a load or congestion vector

$$\ell(x) := \sum_{i \in [n]} x_i \in \mathbb{R}^m.$$

It is worth noting that there may be different $x, y \in X, x \neq y$ with $\ell(y) = \ell(x)$. We further denote by

$$\ell(X) := \{\ell(z) \in \mathbb{R}^m | \, \exists \, x \in X \text{ with } \ell(x) = \ell(z)\}$$

the set of such feasible load vectors. We are further given a possibly player-specific cost function $c_i : \mathbb{R}^m \to \mathbb{R}^m, i \in N$ that maps a load vector $\ell(x) \in \mathbb{R}^m$ to a cost vector $c_i(\ell(x)) \in \mathbb{R}^m$, i.e., $c_i(\ell(x)) = (c_{i1}(\ell(x)), \ldots, c_{im}(\ell(x)))^\top$ and for a resource $r \in R$, the cost experienced by player i using r when the load vector is x is $c_{ir}(\ell(x))$.

Definition 4.1. A generic congestion game is the strategic game $G = (N, X, (\pi_i)_{i \in N})$, where the private cost of player i in strategy profile $x \in X$ is defined as

$$\pi_i(x) = x_i^\top c_i(\ell(x)) = \sum_{r \in R} x_{ir} c_{ir}(\ell(x)).$$

Let us emphasize that generic congestion games exhibit a special structure compared to general games in strategic form in the sense that the cost functions $c_i, i \in N$ depend on the (aggregated) load vector $\ell(x)$ only instead of the entire strategy profile x. This property leads to a complexity reduction in representing the player's cost functions as well as to further interesting structural aspects.

Given fixed strategies x_{-i} of the opponent players, every player aims at choosing a strategy $x_i \in X_i$ minimizing her own private cost, that is, ideally every player wants to solve the following optimization problem:

$$\min_{y_i \in X_i} \pi_i(y_i, x_{-i}). \tag{4.1}$$

Recall that an optimal solution $x_i^* \in X_i$ is called best-response to the combined and fixed strategies x_{-i} of the opponent players. A combined strategy profile $x^* \in X$ is called a pure Nash equilibrium of the game, if under x^*, every player $i \in N$ plays a best-response to x^*_{-i}. This reasoning is formalized in the following definition.

Definition 4.2. Let $G = (N, X, (\pi_i)_{i \in N})$ be a generic congestion game. A strategy profile $x \in X$ is a pure Nash equilibrium, if for all $i \in N$ the following inequality holds true:

$$\pi_i(x) \leq \pi_i(y_i, x_{-i}) \text{ for all } y_i \in X_i. \tag{4.2}$$

Using the definition of generic congestion games, we can now further classify subclasses of congestion games according to the specific structure of the strategy spaces and/or cost functions. In particular, we classify in the following different congestion games according to the following criteria:

- If $X_i \subseteq \{0, 1\}^m$ for all $i \in N$, we speak of an unweighted congestion game. If $X_i \subseteq \{0, d_i\}^m, d_i \in \mathbb{R}_{>0}$ for all $i \in N$, we speak of a weighted congestion game. Further variants such as $X_i \subseteq \times_{j=1}^m \{0, d_{ij}\}, d_{ij} \in \mathbb{R}_{>0}$ are of course also possible.
- If $c_i \equiv c_j$ for all $i, j \in N$, we speak of a congestion game with player-independent or homogeneous costs. Otherwise, we speak of a congestion game with player-specific costs.
- If $c_{ir}(\ell(x)) = c_{ir}(\ell(y))$ for all $\ell(x), \ell(y) \in \ell(X)$ with $\ell_r(x) = \ell_r(y)$, we speak of a congestion game with separable costs. Otherwise, we say that the game has non-separable costs.

4.2 Unweighted Congestion Games

We start in this section with the original unweighted congestion model of Rosenthal [10]. Each player is associated with a strategy space $X_i \subseteq \{0, 1\}^m$ containing the set of pure strategies. By interpreting a binary vector as an indicator function, we can easily interpret a pure strategy of a player as a set of resources and the strategy space then corresponds to a given set of allowable subsets.

Definition 4.3 (Unweighted congestion games). A game $G = (N, X, (\pi_i)_{i \in N})$ is an unweighted congestion game with separable and homogeneous cost functions, if it satisfies the following two properties:

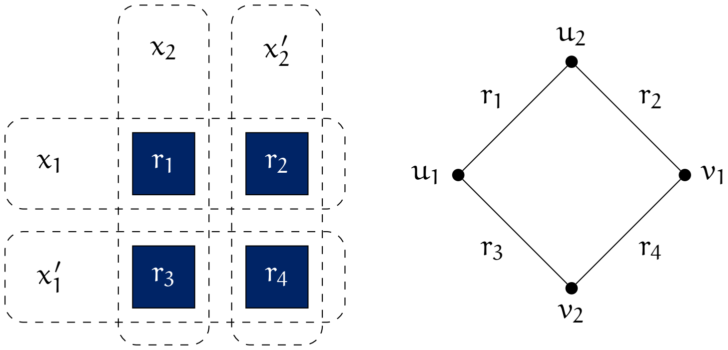
1. The strategy space of every player satisfies $X_i \subseteq \{0, 1\}^m$ for all $i \in N$.
2. The cost maps $c_i, i \in N$ are homogeneous and separable, that is, we only have single cost map denoted by $c(\ell(x)) = (c_1(\ell_r(x)), \dots, c_m(\ell_r(x)))^\top$, where $c_r : \mathbb{N} \to \mathbb{R}$ just maps the load $\ell_r(x)$ of resource r under x to the reals.

For this class of congestion games, the private cost function has the form:

$$\pi_i(x) = x_i^\top c(\ell(x)) = \sum_{r \in R} x_{ir} c_r(\ell_r(x)).$$

As already mentioned above, a feasible strategy can be interpreted as an incidence vector of a subset of resources:

$$R(x_i) := \{r \in R | x_{ir} = 1\} \text{ for all } x_i \in X_i, i \in N.$$



(a) Resource representation (b) Graph representation

	x_2	x_2'
x_1	$(24, 24)$	$(29, 29)$
x_1'	$(29, 29)$	$(24, 24)$

(c) Strategic representation

Figure 4.2: Unweighted congestion game with separable cost functions. The game has two pure Nash equilibria, i.e., (x_1, x_2) and (x_1', x_2').

This naturally leads to a one-to-one correspondence between strategy spaces X_i and player-specific families of subsets of resources

$$R(X_i) := \{R(x_i) | x_i \in X_i\} \subseteq 2^R \text{ for all } i \in N.$$

Let us give an example for an unweighted congestion game.

Example 4.4. Consider the game in Figure 4.2. There are four resources $R := \{r_1, r_2, r_3, r_4\}$ with cost functions defined as $c_{r_1}(z) = c_{r_4}(z) = 2z^3$ and $c_{r_2}(z) = c_{r_3}(z) = (z+1)^3$ for all $z \geq 0$. The feasible strategies of the players are given as $X_1 := \{x_1, x_1'\}$ and $X_2 := \{x_2, x_2'\}$ with $R(x_1) = \{r_1, r_2\}$ and $R(x_1') = \{r_3, r_4\}$; for player 2 the feasible subsets are $R(x_2) = \{r_1, r_3\}$ and $R(x_2') = \{r_2, r_4\}$. Inspecting the strategic representation of the game in Figure 4.2c, we observe that the game has two pure Nash equilibria, i.e., (x_1, x_2) and (x_1', x_2').

Congestion games in which the feasible allocations of each player i can be represented as the set of paths between two designated vertices u_i and v_i of a given graph $D = (V, E)$ are of particular interest.

Definition 4.5 (Network congestion game). An unweighted congestion game $G = (N, X, (\pi_i)_{i \in N})$ is called a network congestion game, if there is a graph $D = (V, E)$ and, for each player $i \in N$, two designated vertices $s_i, t_i \in V$ such that

$$R(X_i) = \{P \subseteq E : P \text{ is a } s_i, t_i\text{-path in } D.\}$$

for all players $i \in N$.

The game from Example 4.4 obviously is a network congestion game as the graph representation in Figure 4.2 shows.

4.2.1 Rosenthal's Potential

In this section, we show that every congestion games has a pure Nash equilibrium.

Theorem 4.6 (Rosenthal [10]). Every congestion game is an exact potential game.

Proof. Rosenthals' potential function $P : X \to \mathbb{R}$ is defined as

$$P(x) := \sum_{r \in R} \sum_{k=1}^{\ell_r(x)} c_r(k). \tag{4.3}$$

Let $x \in X$ be an arbitrary strategy profile and let $y_i \in X_i, y_i \neq x_i$ be an arbitrary deviation of player i. We can easily observe that for resources $r \in R(y_i) \setminus R(x_i)$, the load under (y_i, x_{-i}) increases to $\ell_r(x) + 1$. Hence, the difference between player i's private cost is given by

$$\pi_i(y_i, x_{-i}) - \pi_i(x) = \sum_{r \in R(y_i) \setminus R(x_i)} c_r(\ell_r(x) + 1) - \sum_{r \in R(x_i) \setminus R(y_i)} c_r(\ell_r(x)),$$

where we use that all terms $r \in R(x_i) \cap R(y_i)$ cancel out due to the separability of the cost function c.
The potential function is also separable over $r \in R$ and thus the potential under (y_i, x_{-i}) can be written as the potential under x plus a correction term caused by the deviation of player i:

$$\begin{aligned} P(y_i, x_{-i}) &= \sum_{r \in R} \sum_{k=1}^{\ell_r(x)} c_r(k) + \underbrace{\sum_{r \in R(y_i) \setminus R(x_i)} c_r(\ell_r(x) + 1) - \sum_{r \in R(x_i) \setminus R(y_i)} c_r(\ell_r(x))}_{= \pi_i(y) - \pi_i(x)} \\ &= P(x) + \pi_i(y_i, x_{-i}) - \pi_i(x). \end{aligned}$$

□

It is quite remarkable that for the above result, no assumptions on the cost functions $c_r, r \in R$ were needed, that is, these functions may be non-monotonic, negative and/or positive.

4.2.2 Congestion Games versus Potential Games

We will now show an even stronger connection of potential games with congestion games. In particular, we show that potential games are isomorphic to congestion games. This result was shown by Monderer and Shapley [6] but we present here a proof by Voorneveld et al. [11].

Definition 4.7 (Isomorphic Games). Let $G = (N, X, (u_i)_{i \in N})$ and $H = (N, Y, (v_i)_{i \in N})$ be two strategic games with identical player set N. G and H are said to be isomorphic, if for all $i \in N$, there is a bijective mapping $\phi_i : X_i \to Y_i$ with

$$u_i(x_1, \dots, x_n) = v_i(\phi_1(x_1), \dots, \phi_n(x_n)) \text{ for all } x \in X.$$

We obtain now that every coordination game is isomorphic to a congestion game. Recall the definition of a coordination game in Def. 3.9.

Theorem 4.8. Every coordination game is isomorphic to a congestion game.

Proof. Let $G = (N, X, (u_i)_{i \in N})$ be a coordination game with n players. For all $x \in X$ we create a resource $r(x)$. We define a congestion game $H = (N, (Y_i)_{i \in N}, (\pi_i)_{i \in N})$ with $R = \cup_{x \in X}\{r(x)\}$ as follows:

- For all $i \in N : R(Y_i) := \{g_i(x_i) | x_i \in X_i\}$, where $g_i(x_i) := \cup_{x_{-i} \in X_{-i}}\{r(x_i, x_{-i})\}$.
- For all $r(x) \in R$:

$$c_{r(x)}(\ell) = \begin{cases} u(x), & \text{if } \ell = n \\ 0, & \text{otherwise.} \end{cases}$$

For every $x \in X$ we have $\cap_{i \in N} g_i(x_i) = \{r(x)\}$, hence the congestion game is isomorphic to G (the isomorphism g_i maps x_i to $g_i(x_i)$). □

Example 4.9 (Coordination Game with Isomorphic Congestion Game.).

	(0,0)	(1,1)
	(2,2)	(3,3)

Figure 4.3: Coordination Game

	A	B
	C	D

Figure 4.4: Resources.

	{A, C}	{B, D}
{A, B}	(0,0)	(1,1)
{C, D}	(2,2)	(3,3)

Figure 4.5: Congestion Game.

Figure 4.6: Illustration of Coordination Games with Isomorphic Congestion Games

Consider in Fig. 4.6 a coordination game (left). Every strategy profile will be associated with a resource (middle). The rows of the row player will be associated with the resources of the corresponding row. Similarly we proceed with the column player.

Theorem 4.10. Every dummy game is isomorphic to a congestion game.

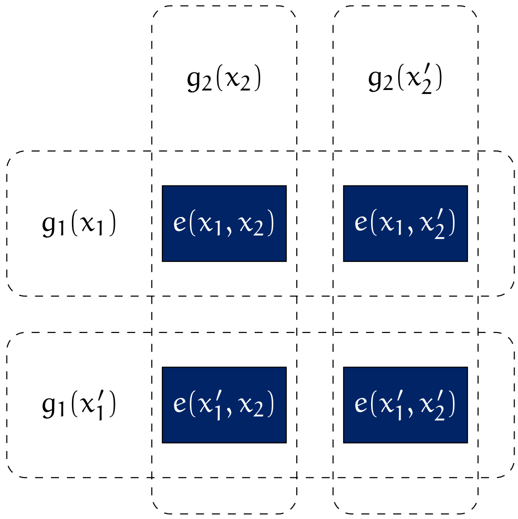
Proof. Let $G = (N, X, (u_i)_{i \in N})$ be a dummy game. For all $i \in N$ and all $x_{-i} \in X_{-i}$ we create a resource $r(x_{-i})$. We define a congestion game $H = (N, (Y_i)_{i \in N}, (\pi_i)_{i \in N})$ with $R = \cup_{i \in N} \cup_{x_{-i} \in X_{-i}} \{r(x_{-i})\}$ as follows:

- For every $i \in N : Y_i := \{h_i(x_i) | x_i \in X_i\}$, where $h_i(x_i) := \{r(x_{-i}) : x_{-i} \in X_{-i}\} \cup \{r(y_{-j}) : j \in N \backslash \{i\}$ und $y_{-j} \in Y_{-j}$ with $y_i \neq x_i\}$
- For all $r(x_{-i}) \in R$:

$$c_{r(x_{-i})}(\ell) = \begin{cases} u_i(x_i, x_{-i}), & \text{if } \ell = 1 \ (x_i \text{ arbitrary}) \\ 0, & \text{else.} \end{cases}$$

	x_2	x_2'
x_1	$(0,0)$	$(1,1)$
x_1'	$(2,2)$	$(3,3)$

(a) Coordination game.



(b) Isomorphic congestion game.

Lemma 4.11. For all $i \in N, \bar{x}_{-i} \in X_{-i}, \bar{x}_i \in X_i$ the player i is the unique user of resource $r(\bar{x}_{-i})$ in $(h_j(\bar{x}_j))_{j\in N}$ and all other resources in $h_i(\bar{x}_i)$ have more than one user.

Proof. Let $i \in N, \bar{x}_{-i} \in X_{-i}, \bar{x}_i \in X_i$. We show first that under the strategy $(h_j(\bar{x}_j))_{j\in N}$ resource $r(\bar{x}_{-i})$ is only used by player i, that is, $r(\bar{x}_{-i}) \in h_i(\bar{x}_i)$ and for every other player $j \in N\backslash\{i\}$ we have $r(\bar{x}_{-i}) \notin h_j(\bar{x}_j)$. Assume $r(\bar{x}_{-i}) \in h_j(\bar{x}_j)$ for some $j \neq i$. As $\bar{x}_{-i} \neq y_{-j}$ for all $y_{-j} \in Y_{-j}$ it suffices to consider the case

$$r(\bar{x}_{-i}) \in \{r(y_{-k}) : k \in N/\{j\} \text{ and } y_{-k} \in X_{-k}\}.$$

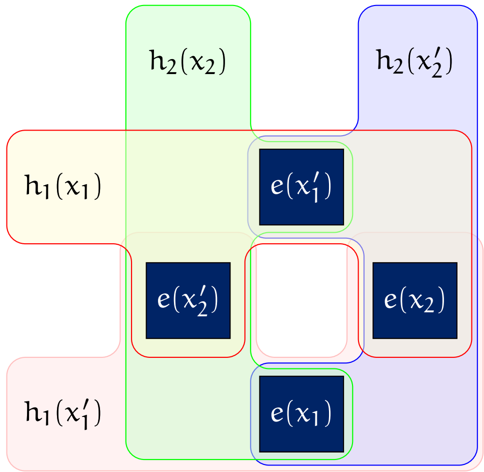
In this case, we have $\bar{x}_{-i} = y_{-k}$ in contradiction to $y_j \neq \bar{x}_j$ (see the definition of $h_j(\bar{x}_j)$). In the following we show that for the strategy $(h_j(\bar{x}_j))_{j\in N}$ every other resource in $h_i(\bar{x}_i)$ is used by at least two players. Let $r \in h_i(\bar{x}_i), r \neq r(\bar{x}_{-i})$. We differentiate between two cases:
(i) If $r = r(y_{-i})$ for some $y_{-i} \in X_{-i}$, then $y_{-i} \neq \bar{x}_{-i}$ implies that $y_j \neq \bar{x}_j$ for some $j \in N\backslash\{i\}$ with $r = r(y_{-i}) \in h_j(\bar{x}_j)$.
(ii) If $r = r(y_{-j})$ for some $j \in N\backslash\{i\}$ with $y_{-j} \in Y_{-j}$ and $y_i \neq \bar{x}_i$, then we have $r = r(y_{-j}) \in h_j(\bar{x}_j)$.
In both cases, r has more than one user. □

Altogether, the constructed game is isomorphic to G, where the isomorphism maps x_i to $h_i(x_i)$. □

Example 4.12 (Dummy Game with Isomorphic Congestion Game.). We create for every opponents strategy vector x_{-i} a resource $r(x_{-i})$. In the middle Fig. 4.10, the value α represents the strategy "left column" of player 2 (the only opponent of player 1) and γ represents the "first row" of player 1 (opponent of player 2). The strategy $h_1(x_1)$ corresponds to the resources $\{\alpha, \beta\} \cup \{\delta\}$, where one should note that γ is not contained in the second set as for the case $j = 2$ it must hold that $y_{-2} \in X_{-2}$ and $y_1 \neq x_1$.

	x_2	x_2'
x_1	$(0,2)$	$(1,2)$
x_1'	$(0,3)$	$(1,3)$

(a) Dummy game.



(b) Isomorphic congestion game.

$(0,2)$	$(1,2)$
$(0,3)$	$(1,3)$

Figure 4.7: Dummy Game

α,γ	β,γ
α,δ	β,δ

Figure 4.8: Introduced Resources.

	$\{\beta,\gamma,\delta\}$	$\{\alpha,\gamma,\delta\}$
$\{\alpha,\beta,\delta\}$	$(0,2)$	$(1,2)$
$\{\alpha,\beta,\gamma\}$	$(0,3)$	$(1,3)$

Figure 4.9: Congestion Game.

Figure 4.10: Illustration of a dummy game with isomorphic congestion game.

Theorem 4.13. Every potential game is isomorphic to a congestion game.

Proof. We subdivide the potential game in a coordination game and dummy game. Then we take the union of the isomorphic congestion games with disjunct resource sets:

$$Y_i = \{g_i(x_i) \cup h_i(x_i) | x_i \in X_i\}.$$

The resources have as cost function the sum of the respective costs of the isomorphic congestion games. □

Example 4.14 (Potential Game with Isomorphic Congestion Game.).

Bibliographic Notes

Unweighted congestion games were first introduced by Rosenthal [10]. His proof of existence of pure Nash equilibria is based on providing a exact potential function can be seen as the discrete analog of the Beckmann potential for non-atomic congestion games given in Beckmann et al. [1]. Monderer and Shapley [6] formalized Rosenthal's approach of using potential functions to show existence of equilibria. They also derive the connection of potential functions with respect the convergence of better response dynamics, i.e., sequences

	$(0,2)$	$(2,3)$
	$(2,5)$	$(4,6)$

Figure 4.11: Potential Game.

	$\{A,C\} \cup \{\beta,\gamma,\delta\}$	$\{B,D\} \cup \{\alpha,\gamma,\delta\}$
$\{A,B\} \cup \{\alpha,\beta,\delta\}$	$(0+0,0+2)$	$(1+1,1+2)$
$\{C,D\} \cup \{\alpha,\beta,\gamma\}$	$(2+0,2+3)$	$(3+1,3+3)$

Figure 4.12: Congestion Game.

Figure 4.13: Illustration of a Potential Game with Isomorphic Congestion Game.

of unilateral deviations strictly reducing the deviating player's private costs always converge to a pure Nash equilibrium provided the game is finite and admits a potential. In addition, they proved that w-potential games have other desirable properties, e.g., that the Fictitious Play Process introduced by Brown [2] converges to a pure Nash equilibrium, see Monderer and Shapley [5].

The characterization of exact potential games as congestion games was first proved by Monderer and Shapley [6]. The proof given in this monograph is based on that given by Voorneveld et al. [11].

4.2.3 Complexity of Computing Nash Equilibria

We will relate the complexity of finding Nash equilibria for congestion games to the complexity of computing local optima for other local search problems. In particular, we consider problems from the class PLS (Polynomially Local Search).

We define a local search problem:

Definition 4.15. A local search problem Π is given by

- a set of instances $\mathcal{I}$;
- for an instance $I \in \mathcal{I}$, let $F(I)$ denote the set of feasible solutions;
- the objective $c : F(I) \to \mathbb{Z}$ maps every feasible solution $S \in F(I)$ to the value $c(S)$;
- for a solution $S \in F(I)$, we denote by $N(S, I) \subseteq F(I)$ the neighborhood of S.

We associate a transition graph with an instance I of a local search problem Π: Every solution $S \in F(I)$ corresponds to a node $v(S)$. There is a directed arc from $v(S_1)$ to $v(S_2)$ if and only if $S_2 \in N(S_1, I)$ and $c(S_2) < c(S_1)$. The sinks of this graph are the local optima.

In case of a congestion game, the set of feasible solutions corresponds to the set of states, the objective function is defined by Rosenthal's potential function and the neighborhood of a state S consists of those states that deviate from S only in one player's strategy. Sequences of improvement steps correspond to paths in the transition graph and the sinks of this graph are the Nash equilibria of the game. Given a congestion game, how difficult is it to compute a Nash equilibrium? One way to find a Nash equilibrium is to use a heuristic following the local search paradigm, i.e., start at any state and perform improvement steps until a Nash equilibrium is reached. Observe that the number of states is exponential in the number of players, even if there are only two alternative strategies for each player. Thus, there might be instances for which a heuristic following this paradigm takes an exponential number of steps. Notice that, even if there are states for which all improvement paths have exponential length, there might be other, indirect algorithmic approaches for computing Nash equilibria efficiently.

Definition 4.16. A local search problem Π belongs to PLS, if the following can be computed in polynomial time:

1. an initial feasible solution $S \in F(I)$ for every instance $I \in \mathcal{I}$;
2. the objective value $c(S)$ for every solution $S \in F(I)$ and every instance I;
3. for every instance $I \in \mathcal{I}$ and every $S \in F(I)$, the decision whether S is locally optimal or not and if not, to find a better solution in the neighbourhood of S, i.e., to find $S' \in N(S, I)$ with $c(S') < c(S)$.

We illustrate the class PLS with the following example.

Example 4.17. The problem Max Cut with the Flip-Neighborhood is an example of a problem in PLS. An instance I of Max Cut consists of an undirected graph $G = (V, E)$ and arc weights $w : E \to \mathbb{R}^+$. A feasible solution $S := (A, B) \in F(I)$ is a cut, i.e., a partition of $A \cup B = V$ of the node set V such that $A, B \neq \emptyset$. For a node set $U \subseteq V$ we define $\delta(U) := \{\{u, v\} \in E \ : \ u \in U,\ v \in V \setminus U\}$ and $w(\delta(U)) := \sum_{e \in \delta(U)} w(e)$. The goal is to find a cut (A, B) such that the weight of all arc traversing the cut is maximal; that is, the objective of a cut $S = (A, B)$ is given by $c(S) := w(\delta(A))$. The Flip-Neighborhood $N(S, I)$ for a cut $S = (A, B)$ corresponds to all cuts (A', B') that are constructed from (A, B) by shifting a single node from A to B (or vice versa). Formally, $A' = A \cup \{r\}$ for some $x \in B$ or $B' = B \cup \{y\}$ for some $y \in A$. It can be seen that all conditions of Definition 4.16 are met.

Next, we define a PLS-Reduction:

Definition 4.18. Let $\Pi_1 = (\mathcal{I}_1, F_1, c_1, N_1)$ and $\Pi_2 = (\mathcal{I}_2, F_2, c_2, N_2)$ be two problems in PLS. Π_1 is PLS-reducible to Π_2, if there are functions f and g that can be determined in polynomial time and

1. f maps every instance $I \in \mathcal{I}_1$ of Π_1 to an instance $f(I) \in \mathcal{I}_2$ of Π_2;
2. g maps every tuple (S_2, I) with $S_2 \in F_2(f(I))$ to a solution $S_1 \in F_1(I)$;
3. for all $I \in \mathcal{I}_1$: if S_2 is a local optimum of $f(I)$, then $g(S_2, I)$ is a local optimum of I.

A problem Π in PLS is PLS-complete, if every problem of PLS can be reduced to Π. Max Cut is for instance PLS-complete.
The above definitions show the following: if we have an efficient algorithm (polynomial time) that determines a local optimum for a PLS-complete problem Π, then we also have

an efficient algorithm for finding local optima for problems in PLS. This holds since every problem in PLS can be reduced in polynomial time to Π and local optima of Π correspond to local optima of the original problem.

We will show that calculating a Nash equilibrium is PLS-complete. The statement also holds for symmetric congestion games, however, we will only show the general case.

Theorem 4.19. Calculating Nash equilibria for congestion games is PLS-complete.

Proof. We have argued before that the calculation of a Nash equilibrium for a congestion game can be interpreted as a local search problem. In particular, one can easily check that there exist polynomial time algorithms satisfying conditions 1–3 of Definition 4.16. Hence, this problem clearly belongs to the class PLS.

We want to show that the problem of calculating a Nash equilibrium in Congestion Games is PLS-complete. To show this, it is sufficient to find a PLS-reduction of an PLS-complete problem to our problem. We will use the PLS-complete problemMAX CUT.

Let $I := (G, w)$ be an instance of MAX CUT. Let W be the weight of all arcs in E: $W := \sum_{e \in E} w(e)$. Observe, that for every cut (A, B) we have: $w(\delta(A)) = W - w(E(A)) - w(E(B))$, where $E(U) \subseteq E$ is the set of all arcs having both endpoints in $U \subseteq V$. Instead of maximizing the weight of arcs that traverse the cut, we can also minimize the weight of the arcs the remain inside the partition.

The reduction is now as follows: For every arc $e \in E$ we add two resources $r_e(A)$ and $r_e(B)$. Hence, the complete set of resources is $R := \cup_{e \in E}\{r_e(A), r_e(B)\}$. We define for $U \in \{A, B\}$ the following homogeneous and separable cost functions

$$c_{r_e(U)}(\ell) := \begin{cases} 0, & \text{if } \ell \leq 1, \\ w(e)/2, & \text{else.} \end{cases}$$

Furthermore, for every node $v \in V$ we have a player $v \in N$. The strategy space for player v has two actions: $X_v := \{x_v(A), x_v(B)\}$, where $x_v(U) := \{r_e(U) \,:\, e \in \delta(v)\}$ for $U \in \{A, B\}$. Consider now the constructed unweighted congestion game $G = (N, X, \pi)$. The interpretation is the following: If player v chooses action $x_v(A)$, then the node $v \in V$ belongs to the set A of the partition $A \cup B = V$; if the player chooses $x_v(B)$, then the node v belongs to the set B. An arc $e = \{u, v\}$ has cost $w(e)$ if and only if player u and v both belong to the same part of the partition. This transformation can be done in polynomial time (corresponds to the function f in Definition 4.18). We obtain

$$P(x) = \sum_{r \in R} \sum_{i=1}^{\ell_r(x)} c_r(i) = w(E(A)) + w(E(B)),$$

and a unilateral deviation corresponds to switching the node to the other partition of the cut. We obtain for a NE $x \in X$ and a node $v \in N$ with $x_v = x_v(A)$:

$$\pi_v(x) \leq \pi_v(x_v(B), x_{-v}) \Leftrightarrow \sum_{e=(vw) \in E:\, w \in A} w(e) \leq \sum_{e=(vw) \in E:\, w \in B} w(e).$$

Any NE thus corresponds to a local optimum of P and thus to a local optimum of the MAX CUT instance with respect to the Flip neighborhood. □

Theorem 4.19 shows that an efficient algorithm for calculating Nash equilibria in symmetric congestion games would imply the existence of efficient algorithms for many local search problems. This is very unlikely but not impossible. However, it is known that efficient algorithms cannot use the local search paradigm: There exist instances of Max Cut, such that every path from a node to a sink in the transition graph has exponential length.

4.2.4 Symmetric Network Congestion Games

Theorem 4.20. Let $G = (N, X, \pi)$ be a symmetric network congestion game on the digraph $D = (V, E)$, i.e., there is a source $s \in V$ a sink $t \in V$ and $X_i = X_j, i, j \in N$, where every X_i is given by the set of incidence vectors of simple s-t paths in G. The cost functions $c_e : \mathbb{N}_+ \to \mathbb{N}$ are nondecreasing. Then, PNE can be computed in polynomial time.

Proof. The congestion game is reduced to a min-cost b-flow problem as follows. Each edge e is replaced by n parallel edges $e_1, \ldots, e_n$ between the same nodes. Edge e_i is assigned cost $c_e(i)$, for $1 \leq i \leq n$. All edges have capacity 1 and balances are defined as $b(s) = n, b(t) = -n, b(v) = 0, v \in V \setminus \{s, t\}$. Observe, if a min-cost b-flow uses some of the edges $e_1, \ldots, e_n$, then we can assume w.l.o.g. that it sends an integral amount of flow along these edges. If it sends k units of flow along these edges, then it uses the k cheapest edges. W.l.o.g., these are the edges $e_1, \ldots, e_k$ as the cost functions are nondecreasing. Thus, the cost for sending the flow along these edges is $c_e(1) + \cdots + c_e(k)$, which corresponds to the potential that Rosenthal's potential function assigns to edge e if k players use this edge. Consequently, we can translate the optimal solution of the min-cost b-flow problem into a state of the congestion game whose potential corresponds to the cost of the flow. Hence, the min-cost flow solution corresponds to a Nash equilibrium that globally minimizes Rosenthal's potential function. □

Let us remark that the result above does not imply that one can reach a Nash equilibrium in symmetric network congestion games with best responses efficiently. In fact, there exist instances of symmetric network congestion games with non-decreasing delay functions that have states with an exponential distance to any Nash equilibrium in the transition graph.

4.2.5 Matroid Congestion Games

An important role in the theory of congestion games plays the combinatorial structure of the set system $\mathcal{S}_i \subseteq 2^E, i \in N$. One structure that is of particular interest are so-called matroid set systems which we will first describe.
Let $R = \{r_1, \ldots, \rho_M\}$ be a set of elements and $\mathcal{I} \subseteq 2^R$ be a set system over R. For two subsets I and $\{r\}$ of R we write $I + r$ or $I - r$ for $I \cup \{r\}$ or $I \setminus \{r\}$, respectively.

Definition 4.21 (Matroid). The pair $M = (R, \mathcal{I})$ is called matroid, if the following conditions are satisfied:

1. $\emptyset \in \mathcal{I}$.
2. $\forall I \in \mathcal{I} \, \forall J \in 2^R \colon J \subseteq I \Rightarrow J \in \mathcal{I}$.

3. $\forall I, J \in \mathcal{I}$: $|J| < |I| \Rightarrow$ there is $r \in I \setminus J$ with $J + r \in \mathcal{I}$.

Here, $|I|$ defines the cardinality of the set I. The tuple $(R, \mathcal{I})$ is called independence system, if only (1) und (2) are satisfied. $\mathcal{I}$ is called a system of independent sets and $2^R \setminus \mathcal{I}$ is called a system of dependent sets.
Inclusion-wise minimal dependent sets are called circuits, inclusions-wise maximal independent sets are called basis. The set of bases is denoted by $\mathcal{B}$ and the set of circuits by $\mathcal{C}$.

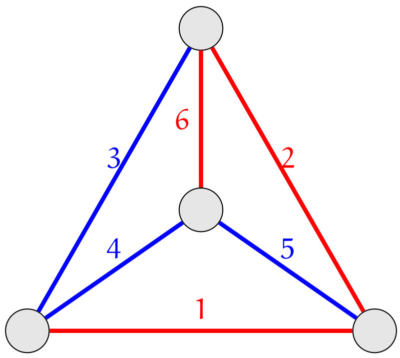


Figure 4.14: Graphical matroid: K_4 with two bases B_1 (red), B_2 (blue). The cycle $\{1, 4, 5\}$ is a circuit.

Example 4.22. The most prominent example of a matroid is the so-called graphical matroid. Let $D = (V, E)$ be a connected undirected graph. The graphical matroid $M(G) = (E, \mathcal{I})$ consists of the edge set as the element set and the independent sets are the cycle-free subsets of E. A maximal cycle-free set of a connected graph is a spanning tree. Therefore, the spanning trees of G correspond to the set of bases of $M(G)$. For an example consider Fig. 4.14.

One can show that for matroid, every bases has the same cardinality.

Theorem 4.23. Let $(R, \mathcal{I})$ be an independence system. The following statements are equivalent:

(M1) $M = (R, \mathcal{I})$ is a matroid.

(M2) $I, J \in \mathcal{I}$ with $|I| = |J| + 1 \Rightarrow$ there is $r \in I \setminus J$ with $J + r \in \mathcal{I}$.

(M3) For every $I \subseteq R$,every basis of I has the same cardinality. Note that a basis of $I \subseteq R$ is defined as a maximal independent subset of I.

Proof. Statement (M1) $\Leftrightarrow$ (M2) follows directly. Also (M1) $\Rightarrow$ (M3) is trivial. It remains to show (M3) $\Rightarrow$ (M1). Let $I, J \in \mathcal{I}$ and $|I| > |J|$. With (M3) we have that J is no basis of $I \cup J$. Hence, there is $r \in (I \cup J) \setminus J = I \setminus J \in \mathcal{I}$ with $J + r \in \mathcal{I}$. □

We define a rank function ρ_M of a matroid $M = (R, \mathcal{I})$.

Definition 4.24 (rank function). The rank function of a matroid $M = (R, \mathcal{I})$ is defined as

$$\rho_M : 2^R \to \mathbb{N},\ U \mapsto \max\{|Z| : Z \in \mathcal{I}, Z \subseteq U\}.$$

We simply write $\rho = \rho_M$ if the matroid in question is clear from the context. The value $\rho(E)$ is called the rank of M.

Matroids und Bases

A key result in matroid theory is the basis exchange property.

Theorem 4.25 (Weak bases exchange property). Let $(R, \mathcal{I})$ be a matroid with basis system $\mathcal{B}$. Then,

1. $\mathcal{B} \neq \emptyset$
2. For every two sets $B_1, B_2 \in \mathcal{B}$ and element $r \in B_1 \setminus B_2$ there is $s \in B_2 \setminus B_1$ such that $B_1 - r + s \in \mathcal{B}$.

Proof. The set of bass of a matroid $(R, \mathcal{I})$ satisfy (1), because $\emptyset \in \mathcal{I}$. In order to show (2), let $B_1, B_2 \in \mathcal{B}$ and $r \in B_1 \setminus B_2$. As $B_1 - r \in \mathcal{I}$ we can apply (M2), that is, $|B_1 - r| + 1 = |B_2|$. Hence, there is $s \in B_2 \setminus (B_1 - r)$ such that $B_1 - r + s \in \mathcal{I}$. As all bases have the same cardinality (see (M3)) we also get $B_1 - r + s \in \mathcal{B}$. □

In the following we want to derive an even stronger exchange property.

Theorem 4.26 (Strong bases exchange property). Let $(R, \mathcal{I})$ be a matroid with basis system $\mathcal{B}$. Let $B_1, B_2 \in \mathcal{B}$ and let $r \in B_1 \setminus B_2$. Then, there is $s \in B_2 \setminus B_1$ such that

$$B_1 - r + s \in \mathcal{B} \text{ and } B_2 - s + r \in \mathcal{B}.$$

The proof can be found in the appendix.
We now derive an important exchange property for matroid bases. For two bases $B_1, B_2 \in \mathcal{B}$ we define a bipartite directed graph $G(B_1 \triangle B_2) = (V, E)$, with $V = B_1 \triangle B_2$ and Partitions $B_1 \setminus B_2$ and $B_2 \setminus B_1$. There is an edge $e = (r, s) \in E$ if and only if $B_1 - r + s \in \mathcal{B}$.

Lemma 4.27. Let $(R, \mathcal{I})$ be a matroid with bases system $\mathcal{B}$ and $B_1, B_2 \in \mathcal{B}$. Then, $G(B_1 \triangle B_2)$ admits a perfect matching.

Proof. We use induction on $|B_1 \setminus B_2|$. Assume $|B_1 \setminus B_2| \geq 1$. Choose $r \in B_1 \setminus B_2$. With Theorem 4.26 there is $s \in B_2 \setminus B_1$ with

$$B_1 - r + s \in \mathcal{B} \text{ and } B_2 - s + r \in \mathcal{B}.$$

Applying the induction hypothesis on B_1 and $B_2' = B_2 - s + r$ the graph $G(B_1 \triangle B_2')$ admits a perfect matching say $N \subseteq E$. Extend N by edge (r, s) and we obtain a perfect matching $N \cup (r, s)$ for $G(B_1 \triangle B_2)$. □

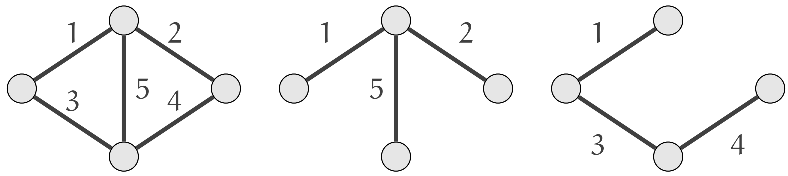


Figure 4.15: Two bases B (middle) and B' (right) of a graphic matroid

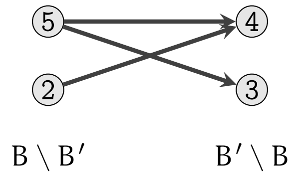


Figure 4.16: Bipartite graph $G(B \triangle B') = (B \triangle B', E)$ with $(r, s) \in E \Leftrightarrow B - r + s \in \mathcal{B}$.

Matroid Congestion Games

A congestion game is a matroid-game, if for every $i \in N$ there is a matroid $M_i = (R, \mathcal{I}_i)$ such that $R(X_i) \subseteq 2^R$ corresponds to the base set of M_i. For a matroid game $G = (N, X, \pi)$ we denote by r_i the rank of M_i and by $\text{rk}(G) := \sum_{i \in N} r_i$ the rank of the game G.
We obtain the following result on the convergence of best-response dynamics.

Theorem 4.28. Let G be a matroid game. Every best-response sequence converges after at most $n \cdot m \cdot \text{rk}(G) \leq n^2 m^2$ to a PNE.

Proof. Let L be a sorted list of all cost values that can occur on every resource. We assume that L lists the values $c_r(i), r \in R = \{r_1, \ldots, r_m\}, i \in N = \{1, \ldots, n\}$ in nondecreasing order. For every resource we define an alternative cost function

$$c'_r : \mathbb{N} \to \{1, \ldots, n \cdot m\},$$

that maps $i \in [0, n]$ to the list position of $c_r(i)$ in L. Equal cost values are mapped to the same list position.
We need the following lemma.

Lemma 4.29. Let $x \in X$ and let x_i^* be a best-response of player i with $x_i^* \neq x_i$, which strictly decreases the cost $\pi_i(x)$. Then x_i^* also strictly decreases $\pi'_i(x)$ w.r.t. the alternative cost function c'.

Proof. Consider the bipartite graph $G(x_i^* \Delta x_i)$ which by Lemma 4.27 contains a perfect matching P. Let $(r, s) \in P$, i.e.,

$$r \in R(x_i^*) \setminus R(x_i), s \in R(x_i) \setminus R(x_i^*) \text{ and } R(x_i^*) - r + s \in \mathcal{B}_i.$$

For $x^* = (x_i^*, x_{-i})$ we have

$$c_r(\ell_r(x^*)) \leq c_s(\ell_s(x^*) + 1),$$

as otherwise $x'_i := x_i^* - \mathbf{1}_r^R + \mathbf{1}_s^R \in X_i$ would lead to smaller costs than under x_i^* in contradiction to the fact that x_i^* is a best-response to x_{-i}. Moreover, there must exist an edge $(r, s) \in P$ with strictly smaller costs

$$c_r(\ell_r(x^*)) < c_s(\ell_s(x^*) + 1),$$

as under strategy x_i^* the cost of i is strictly decreased. The same inequalities must hold for c' as they correspond to the list ranks of the corresponding cost values. We get

$$\pi_i'(x_i^*, x_{-i}) < \pi_i'(x).$$

□

Now we consider Rosenthals' potential function P' w.r.t. c'. Lemma 4.29 shows that every best-response strictly decreases the potential. Because there are only $n \cdot m$ different cost values, we obtain

$$c_r'(i) \leq n \cdot m \text{ for all } r \in R \text{ and all } 1 \leq i \leq n.$$

It follows that:

$$P'(x) = \sum_{r \in R} \sum_{i=1}^{\ell_r(x)} c_r'(i) \leq \sum_{r \in R} \sum_{i=1}^{\ell_r(x)} n \cdot m \leq n \cdot m \cdot \mathrm{rk}(G),$$

where the last inequality follows since the total load is equal $\mathrm{rk}(G)$. The theorem follows since every best-response decreases the potential by at least 1 and all potential values are nonnegative. □

4.3 Weighted Congestion Games

Weighted congestion games are a straightforward generalization of unweighted congestion in which the player contribute differently to the congestion on the resources.

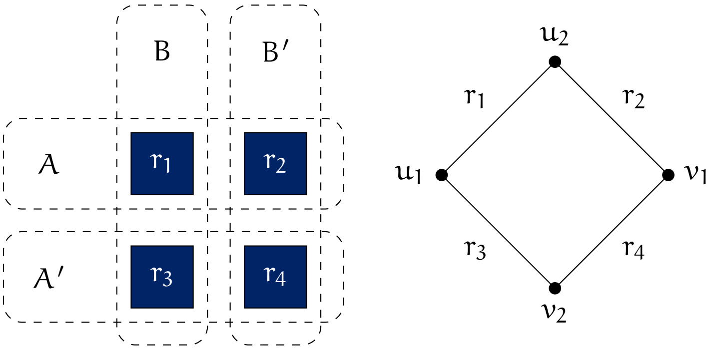
Definition 4.30 (Weighted Congestion Game). Let $R = \{1, \ldots, m\}$ be a finite set of resources and and cost functions $c_r : \mathbb{R}_{\geq 0} \to \mathbb{R}, r \in R$. Let $X = (X_1, \ldots, X_n)$ be the combined strategy space, where $X_i \subseteq \{0, d_i\}^m$ for all $i \in N$. The vector $d = (d_1, \ldots, d_n)$ is a demand vector with $d_i > 0$ for all $i \in N$. A weighted congestion game is the strategic game $G = (N, X, \pi)$ with $\pi_i(x) = c^\top(\ell(x))x_i$.

Note that by simple calculations, we obtain the following equivalent representations of $\pi_i(x)$:

$$\pi_i(x) = c^\top(\ell(x))x_i = \sum_{r \in R} c_r\big(\ell_r(x)\big)x_{ir} = \sum_{r \in R(x_i)} c_r\big(\ell_r(x)\big)d_i \text{ for all } i \in N.$$

Example 4.31. Consider a weighted variant of the congestion game from Example 4.4 the game in Figure 4.17. Cost functions are defined as $c_{r_1}(x) = c_{r_4}(x) = 2x^3$ and $c_{r_2}(x) = c_{r_3}(x) = (x+1)^3$ for all $x \geq 0$. The players' demands are $d_1 = 1$ and $d_2 = 2$. This game does not have a pure Nash equilibrium.

Theorem 4.32. Weighted congestion games with affine cost functions have a pure Nash equilibrium.

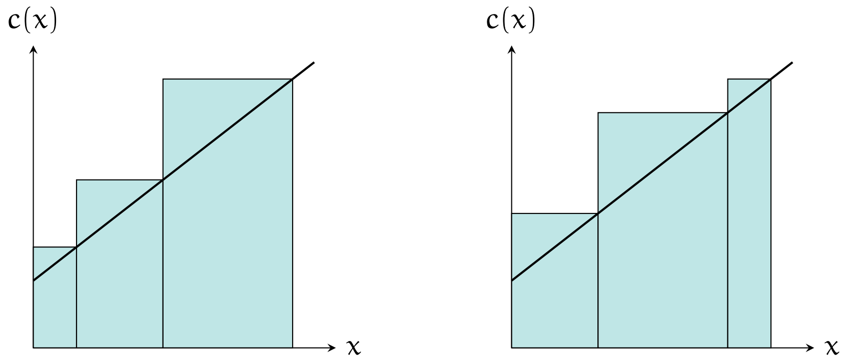


(a) Resource representation (b) Graph representation

	B	B′
A	(62, 162)	(66, 160)
A′	(66, 160)	(62, 162)

(c) Strategic representation

Figure 4.17: Weighted congestion Game



(a) $d_1 = 1,\ d_2 = 2,\ d_3 = 3$ (b) $d_1 = 2,\ d_2 = 3,\ d_3 = 1$

Figure 4.18: For an affine function the potential function is independent of the ordering of the players.

Proof. Consider the function $P : X \longrightarrow \mathbb{R}$ defined as

$$P(x) = \sum_{r \in R} \sum_{i \in N : r \in R(x_i)} d_i c_r \Big(\sum\nolimits_{j \in \{1,\dots,i\} : r \in R(x_j)} d_j \Big).$$

For players of unit weight this function is equal to the potential function given for unweighted congestion games. Again, the contribution of a resource to the potential function corresponds to a discrete approximation of the integral. However, this approximation is not uniform and, thus, depends on the indices of the players (respectively, their demands), in general. The crucial observation is that for affine function, the potential is independent of the ordering of the players, see Figure 4.18.

In fact, we obtain for cost functions of the form $c_r(x) = a_r x + b_r$ with $a_r, b_r \in \mathbb{R}$ that

$$\begin{aligned} P(x) &= \sum_{r \in R} \sum_{i \in N : r \in R(x_i)} d_i c_r \Big(\sum\nolimits_{j \in \{1,\dots,i\} : r \in R(x_j)} d_j \Big) \\ &= \sum_{r \in R} \left(b_r \ell_r(x) + \sum\nolimits_{i \le j \in N : r \in R(x_i) \cap R(x_j)} a_r d_i d_j \right) \end{aligned}$$

$$= \sum_{r\in R}\left(b_r\ell_r(x) + \frac{1}{2}\sum\nolimits_{i,j\in N: r\in R(x_i)\cap R(x_j)} a_r d_i d_j + \frac{1}{2}\sum_{i\in N: r\in R(x_i)} a_r d_i^2\right)$$
$$= \frac{1}{2}\sum_{r\in R}\Big(c_r\big(\ell_r(x)\big) + c_r(0)\Big)\ell_r(x) + \frac{1}{2}\sum_{i\in N}\sum_{r\in R(x_i)}\Big(c_r(d_i) - c_r(0)\Big)d_i.$$

We claim that P is an exact potential, i.e., $P(x_i, x_{-i}) - P(x) = \pi_i(x_i, x_{-i}) - \pi_i(x)$ for all $i \in N$, $x \in X$ and $x_i \in X_i$. Since the potential function is independent of the ordering of the players, it is without loss of generality to assume that $i = n$. We then calculate for $y := (y_n, x_{-n})$

$$P(y_n, x_{-n}) = P(x) + d_n \sum_{r\in R(y_n)\setminus R(x_n)} c_r\big(\ell_r(y)\big)\big) - d_n \sum_{r\in R(x_n)\setminus R(y_n)} c_r\big(\ell_r(y)\big)$$
$$= P(x) + \pi_n(y_n, x_{-n}) - \pi_n(x).$$

We conclude that P is a potential function and, thus, the game has a pure Nash equilibrium. □

For $\phi \in R$, let

$$\mathcal{C}_{\exp}(\phi) = \{f : \mathbb{R}_{\geq 0} \to \mathbb{R} \mid f(x) = ae^{\phi x} + b \text{ with } a, b \in \mathbb{R}\}.$$

> **Theorem 4.33.** For every $\phi \in \mathbb{R}$, every weighted congestion game with cost functions in $\mathcal{C}_{\exp}(\phi)$ has a pure Nash equilibrium.

Proof. For $\phi = 0$, the theorem follows from Theorem 4.32 so we may assume that $\phi \neq 0$. Consider the function $P : X \to \mathbb{R}$ defined as

$$P(x) = \sum_{r\in R}\sum_{i\in N: r\in R(x_i)} \operatorname{sgn}(\phi)\big(1 - e^{-\phi d_i}\big)c_r\Big(\sum\nolimits_{j\in\{1,\dots,i\}: r\in R(x_j)} d_j\Big).$$

For cost functions of the form $c_r(x) = a_r e^{\phi x} + b_r$, this give

$$P(x) = \sum_{r\in R}\sum_{i\in N: r\in R(x_i)}\Big(\operatorname{sgn}(\phi)\big(1 - e^{-\phi d_i}\big)a_r \exp\Big(\phi\sum\nolimits_{j\in\{1,\dots,i\}: r\in R(x_j)} d_j\Big) + b_r\Big)$$
$$= \sum_{r\in R}\sum_{i\in N: r\in R(x_i)}\Big(\operatorname{sgn}(\phi)a_r e^{\phi\sum_{j\in\{1,\dots,i\}: r\in R(x_j)} d_j} - \operatorname{sgn}(\phi)a_r e^{\phi\sum_{j\in\{1,\dots,i-1\}: r\in R(x_j)} d_j}$$
$$+ b_r\operatorname{sgn}(\phi)\big(1 - e^{\phi d_i}\big)\Big).$$

Collapsing the telescoping sum, we obtain

$$P(x) = \operatorname{sgn}(\phi)\sum_{r\in R}\Big(a_r\big(e^{\phi\ell_r(x)} - 1\big) + b_r\sum\nolimits_{i\in N: r\in R(x_i)}\big(1 - e^{-\phi d_i}\big)\Big),$$

which is independent of the ordering of the players. For $\lambda_i = \operatorname{sgn}(\phi)\big(1 - e^{-\phi d_i}\big)$ we then obtain

$$P(y_n, x_{-n}) = P(x) + \lambda_n \sum_{r\in R(y_n)\setminus R(x_n)} c_r\big(\ell_r(y)\big) - \lambda_n \sum_{r\in R(x_n)\setminus R(y_n)} c_r\big(\ell_r(x)\big)$$
$$= P(x) + \frac{d_n}{\lambda_n}\big(\pi_n(y_n, x_{-n}) - \pi_i(x)\big).$$

Using that $d_n/\lambda_n > 0$, we conclude that every sequence of profitable unilateral improvements is finite and, thus, the game has a pure Nash equilibrium. □

In the following, we show that the set of affine cost functions and the set of exponential cost functions are the only sets of cost functions for which general existence results as in Theorem 4.32 and Theorem 4.33 are possible. In order to make the statement precise, we introduce the notion of consistency of a set of cost functions.

Definition 4.34 (Consistent Set of Cost Functions). A set $\mathcal{C}$ of cost functions $c : \mathbb{R}_{\geq 0} \to \mathbb{R}$ is called consistent (for weighted congestion games) if every weighted congestion game with the property that $c_r \in \mathcal{C}$ for all $r \in R$ has a pure Nash equilibrium.

The following lemma states that when characterizing consistent sets of cost functions, it is without loss of generality to assume that the set is closed under positive integer scalar multiplication.

Lemma 4.35. A set $\mathcal{C}$ of cost functions is consistent if and only if the set

$$\bar{\mathcal{C}} := \{g : \mathbb{R}_{\geq 0} \to \mathbb{R} \mid f(x) = \lambda f(x) \text{ for some } \lambda \in \mathbb{N}, f \in \mathcal{C}\}$$

is consistent.

Sketch. Every resource e with cost function λf where $\lambda \in \mathbb{N}$ and $f \in \mathcal{C}$ can be replaced by λ resources with cost function f. □

In order to characterize the sets of consistent cost functions, the following lemma will be useful.

Lemma 4.36. Let $\mathcal{C}$ be a consistent set of strictly increasing functions. Then,

$$\frac{f(x+y) - f(x)}{f(x+y) - f(y)} = \frac{g(x+y+t) - g(x+t)}{g(x+y+t) - g(y+t)} \tag{4.4}$$

for all $f, g \in \mathcal{C}$ and $x, y, t \in \mathbb{R}_{\geq 0}$.

Proof. By Lemma 4.35, it is without loss of generality to assume that $\mathcal{C}$ is closed under scalar multiplication. Let $\kappa, \lambda \in \mathbb{N}$ be arbitrary and consider a weighted congestion game with four resources $E = \{r_1, r_2, r_3, r_4\}$ with cost functions $c_{r_1} = c_{r_4} = \kappa g$, $c_{r_2} = c_{r_3} = \lambda f$ and four players i, j, k, k' with demands $d_i = x$, $d_j = y$, $d_k = d_{k'} = t$ and strategy sets $S_i = \{\{r_1, r_2\}, \{r_3, r_4\}\}$, $S_j = \{\{r_1, r_3\}, \{r_2, r_4\}\}$, $S_k = \{\{r_1\}\}$, $S_{k'} = \{\{r_4\}\}$.
Players i and j have two strategies each and players k and k' have one strategy each resulting in four distinct strategy profiles. There are only two types of strategy profiles: in the first type players i and j are together on a resource with cost function κg, but are not together on a resource with cost function λf. For the second type, it is the other way around. In a strategy profile s^1 of the first type we have

$$\pi_i(s^1) = \kappa g(x+y+t) + \lambda f(x) \qquad \pi_j(s^1) = \kappa g(x+y+t) + \lambda f(y) \tag{4.5a}$$

while in a strategy profile s^2 of the second type we have

$$\pi_i(s^2) = \kappa g(x+t) + \lambda f(x+y) \qquad \pi_j(s^2) = \kappa g(y+t) + \lambda f(x+y). \tag{4.5b}$$

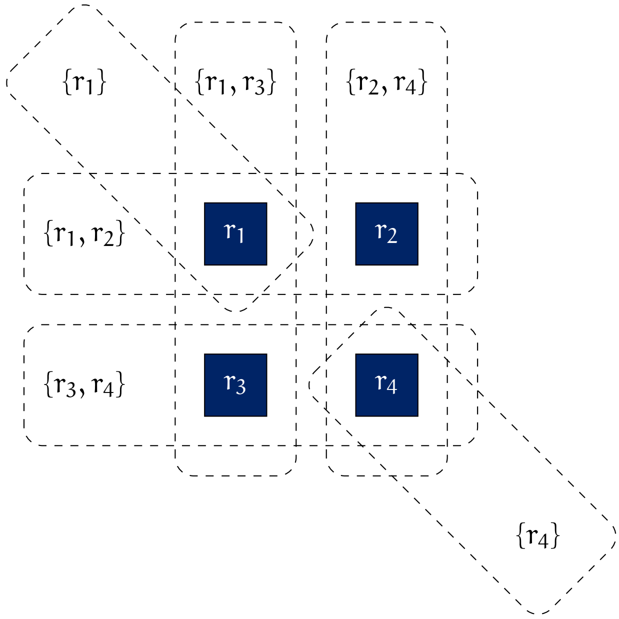


Figure 4.19: Congestion game constructed in the proof of Lemma 4.36.

In order to have a pure Nash equilibrium, one of these two types must be beneficial to both players, i.e., at least one of the following two cases holds:

$$\pi_\ell(s^1) \leq \pi_\ell(s^2) \text{ for } \ell \in \{i, j\} \qquad \text{or} \qquad \pi_\ell(s^1) \geq \pi_\ell(s^2) \text{ for } \ell \in \{i, j\}. \tag{4.6}$$

Combining (4.5) and (4.6), we obtain that at least one of the following cases holds:

$$\frac{\kappa}{\lambda} \leq \min\left\{\frac{f(x+y)-f(x)}{g(x+y+t)-g(x+t)}, \frac{f(x+y)-f(y)}{g(x+y+t)-g(y+t)}\right\} := \alpha_1 \qquad \text{or}$$

$$\frac{\kappa}{\lambda} \geq \max\left\{\frac{f(x+y)-f(x)}{g(x+y+t)-g(x+t)}, \frac{f(x+y)-f(y)}{g(x+y+t)-g(y+t)}\right\} := \alpha_2.$$

Because this must hold for all $\frac{\kappa}{\lambda} \in \mathbb{Q}_+$, it follows that any $\alpha \in (\alpha_1, \alpha_2)$ violates both constraints implying $\alpha_1 = \alpha_2$. Thus, we obtain

$$\frac{f(x+y)-f(x)}{g(x+y+t)-g(x+t)} = \frac{f(x+y)-f(y)}{g(x+y+t)-g(y+t)}.$$

Multiplying this equation with $\frac{g(x+y+t)-g(x+t)}{f(x+y)-f(y)}$ gives the claimed result. □

Lemma 4.37. Let f be a continuous and strictly increasing function such that

$$\frac{f(x+y)-f(x)}{f(x+y)-f(y)} = \frac{f(x+y+t)-f(x+t)}{f(x+y+t)-f(y+t)} \tag{4.7}$$

for all $x, y, t \in \mathbb{R}_{\geq 0}$. Then, f is one of the two functional forms:

$$f(x) = ax + b, \qquad \text{or} \qquad f(x) = ae^{\phi x} + b.$$

Proof. Let $\epsilon > 0$ be arbitrary and let $x = \epsilon$, $y = 2\epsilon$, and $t = m\epsilon$ for some $m \in \mathbb{N}$. Using (4.7), we obtain

$$\gamma := \frac{f(3\epsilon) - f(1\epsilon)}{f(3\epsilon) - f(2\epsilon)} = \frac{f((m+3)\epsilon) - f((m+1)\epsilon)}{f((m+3)\epsilon) - f((m+2)\epsilon)}$$

for all $m \in \mathbb{N}$. We note that $\gamma \neq 1$ since $f(1\epsilon) \neq f(2\epsilon)$. Setting $a_m := f(m\epsilon)$ for all $m \in \mathbb{N}$ and rearranging terms, we derive that the sequence $(a_m)_{m\in\mathbb{N}}$ obeys the recursive formula

$$0 = a_{m+3} - \frac{\gamma}{\gamma - 1} a_{m+2} + \frac{1}{\gamma - 1} a_{m+1}$$

for all $m \in \mathbb{N}$. The characteristic equation of this recurrence relation equals

$$x^2 - \frac{\gamma}{\gamma - 1} x + \frac{1}{\gamma - 1} = (x - 1)\Big(x - \frac{1}{\gamma - 1}\Big).$$

If $\gamma \neq 2$, the characteristic equation has two distinct roots and a_m can be calculated explicitly and uniquely as

$$a_m = a\Big(\frac{1}{\gamma - 1}\Big)^m + b \qquad (4.8)$$

for some constants $a, b \in \mathbb{R}$. If, on the other hand, $\gamma = 2$, we can calculate a_m as

$$a_m = am + b \qquad (4.9)$$

for some constants $a, b \in \mathbb{R}$. Thus, f is affine or exponential for every integer multiple of ϵ. As ϵ was arbitrary we may conclude that f is affine or exponential on a dense subset of $\mathbb{R}_{\geq 0}$. Using that f is continuous, the claimed result follows. □

Theorem 4.38. A set $\mathcal{C}$ of strictly increasing and continuous functions is consistent for weighted congestion games if and only if one of the following two cases holds:

1. $\mathcal{C}$ contains only affine functions of the form $c(x) = ax + b$.
2. $\mathcal{C}$ contains only exponential functions of the form $c(x) = ae^{\phi x} + b$, where ϕ is equal for all functions in $\mathcal{C}$.

Proof. The "if"-part of the statement follows from Theorem 4.32 and Theorem 4.33, so it suffices to show the "only if"-part.
By Lemma 4.37, all functions in $\mathcal{C}$ are either linear or exponential. It is left to show that $\mathcal{C}$ neither contains both affine and exponential functions nor $\mathcal{C}$ contains exponential functions with different ϕ in the exponent. To this end, recall that, by Lemma 4.36 it is necessary that

$$\frac{f(x+y) - f(x)}{f(x+y) - f(y)} = \frac{g(x+y+t) - g(x+t)}{g(x+y+t) - g(y+t)}$$

for all $f, g \in \mathcal{C}$ and $x, y, t \in \mathbb{R}_{\geq 0}$. For all $x, y, t \in \mathbb{R}_{\geq 0}$, the ratio $\frac{f(x+y+t)-f(x)}{f(x+y+t)-f(y+t)}$ is equal to x/y if f is linear, and is equal to $e^{\phi(x-y)}$ if f is of the form $ae^{\phi x} + b$. Thus, (4.7) is satisfied for all x, y only if $\mathcal{C}$ is as claimed. □

4.4 Nonatomic Congestion Games

A nonatomic congestion game is given by a non-empty and finite set of populations $N = \{1, \dots, n\}$, each consisting of a continuum of infinitesimally small agents represented by an interval $[0, d_i]$ for some $d_i > 0$. There is a set of resources $R = \{r_1, \dots, r_m\}$ and for each population $i \in N$, there is a non-empty set $\mathcal{S}_i \subseteq 2^R$ of allowable subsets. A strategy for an agent of population i is to choose exactly one of the sets $S \in \mathcal{S}_i$. This leads to a strategy distribution of population i represented by a vector $x_i = (x_{iS})_{S \in \mathcal{S}_i}$. A feasible distribution fulfills the following properties:

$$x_{iS} \geq 0, S \in \mathcal{S}_i \text{ and } \sum_{S \in \mathcal{S}_i} x_{iS} = d_i.$$

The space of feasible distributions and the Cartesian product of the feasible distributions is given by

$$X_i := \left\{ x_i \in \mathbb{R}_+^{|\mathcal{S}_i|} \middle| \sum_{S \in \mathcal{S}_i} x_{iS} = d_i \right\} \text{ and } X := \times_{i \in N} X_i.$$

Alternativly, we can define the set of feasible resource usage vector for every population $i \in N$:

$$\ell_i(X) := \left\{ \ell_i(x) \in \mathbb{R}_+^m \middle| \ell_i(x) = \sum_{S \in \mathcal{S}_i} x_{iS} \mathbf{1}^S,\ x_i \in X_i \right\} \text{ and } \ell(X) := \times_{i \in N} \ell_i(X).$$

The set $\ell_i(X)$ is an affine transformation (projection) of a polytop and hence a polytope itself. In particular, $\ell_i(X)$is a convex and closed subset of $\mathbb{R}_+^m$. For every $x \in X$ there is a unique $\ell(x) \in \ell(X)$ via the identity $\ell(x) = \left(\sum_{S \in \mathcal{S}_i} x_{iS} \mathbf{1}^S\right)_{i \in N}$. The following exercise demonstrate an important difference between X and $\ell(X)$.

Definition 4.39. Let $\ell(x) \in \ell(X)$. The set of $y \in X$ with $\ell(y) = \ell(x)$ are called decompositions of $\ell(x)$. Note that decompositions need not be unique.

Exercise 4.40. Find a nonatomic congestion game for which the following is true: there is $\ell(x) \in \ell(X)$ and $z \neq y \in X$ with

$$\ell(z) = \ell(x) = \ell(y).$$

For $x \in X$ let

$$\ell_r(x) := \sum_{i \in N} \ell_{i,r}(x)$$

denote the load of resource $r \in R$.

For every resource $r \in R$, there is a nonnegative, nondecreasing and continuous cost function $c_r : \mathbb{R}_+ \to \mathbb{R}_+$. If for $x \in X$, a player of population i selects the subset $S \in \mathcal{S}_i$, she obtains the resulting private cost of:

$$\pi_{iS}(x) = \sum_{r \in S} c_r(\ell_r(x)).$$

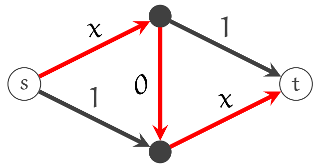


Figure 4.20: In a WE all players follow the red path.

The total cost of a strategy distribution $x \in X$ sums up the costs of all populations and is given as:

$$C(x) = \sum_{i \in N} \sum_{S \in \mathcal{S}_i} \pi_{iS}(x) x_{iS}. \tag{4.10}$$

Exercise 4.41. Show:

$$C(x) = \sum_{r \in R} c_r(\ell_r(x))\ell_r(x) \text{ for all } x \in X.$$

Wardrop-Equilibrium

A similar concept as NE was introduced by Wardrop (1952) in his first principle: A strategy distribution is a Wardrop-Equilibrium (WE), if every used set of resources has minimum cost.

Definition 4.42 (Wardrop-Equilibrium (WE)). A strategy distribution $x \in X$ is a Wardrop-Equilibrium, if

$$\pi_{iS}(x) \leq \pi_{iT}(x), \text{ for all } i \in N \text{ and } S, T \in \mathcal{S}_i \text{ with } x_{iS} > 0. \tag{4.11}$$

Exercise 4.43. Let $x \in X$ be a WE. Show: for all $i \in N$ there is $k_i \in \mathbb{R}_+$ with

$$\pi_{iS}(x) = k_i \text{ for all } S \in \mathcal{S}_i \text{ with } x_{iS} > 0$$

and

$$\pi_{iT}(x) \geq k_i \text{ for all } t \in S_i.$$

We give an example.

Example 4.44 (Routing-Games). Let $G = (V, E)$ be a directed graph with source sink pair (s, t) as in Fig.. 4.20. There is one population with $d_1 = 1$. The set of reosurces corresponds to E and the subsets of resources $\mathcal{S}_1$ correspond to the set of simple s-t paths in G. In the unique WE, all players travel along the zig-zag-path.

Existence and Uniqueness

We give a characterization of WE via solutions of an associated convex optimization problem.

Theorem 4.45. Let $\ell(x) \in \ell(X)$ with arbitrary decomposition $x \in X$. Then, the following statements are equivalent:

1. $x \in X$ is a WE.
2. $\ell(x) \in \ell(X)$ satisfies the following variational inequality
$$\sum_{r \in R} c_r(\ell_r(x))\,(\ell_r(x) - \ell_r(y)) \le 0 \text{ for all } \ell(y) \in \ell(X). \tag{4.12}$$
3. $\ell(x) \in \ell(X)$ is an optimal solution of
$$\min_{\ell(z) \in \ell(X)} \left\{ \Phi(\ell(z)) := \sum_{r \in R} \int_0^{\ell_r(z)} c_r(t)\, dt \right\}. \tag{4.13}$$

Proof. (1) ⇒ (2). Let $x \in X$ be a WE and $\ell(y) \in \ell(X)$ arbitrary. Since x is a WE, for all $i \in N$, there is $k_i \in \mathbb{R}_+$ with $k_i = \sum_{r \in S} c_r(\ell_r(x))$ for all $S \in \mathcal{S}_i$ with $x_{iS} > 0$ and $k_i \le \sum_{r \in T} c_r(\ell_r(x))$ for all $T \in \mathcal{S}_i$. We get

$$\begin{aligned}
\sum_{r \in R} c_r(\ell_r(x))\,\ell_r(x) &= \sum_{i \in N} \sum_{S \in \mathcal{S}_i} k_i x_{iS} \\
&= \sum_{i \in N} k_i \left(\sum_{S \in \mathcal{S}_i} x_{iS} \right) \\
&= \sum_{i \in N} k_i \left(\sum_{S \in \mathcal{S}_i} y_{iS} \right) \\
&= \sum_{i \in N} \sum_{S \in \mathcal{S}_i} k_i y_{iS} \\
&\le \sum_{i \in N} \sum_{S \in \mathcal{S}_i} \sum_{r \in S} c_r(\ell_r(x)) y_{iS} \\
&= \sum_{r \in R} c_r(\ell_r(x))\,\ell_r(y).
\end{aligned}$$

(2) ⇒ (3). Let $\ell(x) \in \ell(X)$ such that (4.12) is valid. Consider a linear approximation of Φ in $\ell(x)$ defined as

$$T(\ell(z); \ell(x)) := \Phi(\ell(x)) + \sum_{r \in R} c_r(\ell_r(x))\,(\ell_r(z) - \ell_r(x)).$$

With the convexity of Φ on $\ell(X)$ we have for all $\ell(z) \in \ell(X)$:

$$T(\ell(z); \ell(x)) = \Phi(\ell(x)) + \sum_{r \in R} c_r(\ell_r(x))\,(\ell_r(z) - \ell_r(x)) \le \Phi(\ell(z))$$

$$\Leftrightarrow \Phi(\ell(x)) \leq \Phi(\ell(z)) + \underbrace{\sum_{r\in R} c_r(\ell_r(x))\,(\ell_r(x)-\ell_r(z))}_{\leq 0 \text{ with } (4.12)} \leq \Phi(\ell(z)).$$

(3) $\Rightarrow$ (1). Let $\ell(x) \in \ell(X)$ be an optimal solution to (4.13). As $\ell(X)$ is convex, we get that for all $\ell(x), \ell(y) \in \ell(X), \ell(x) \neq \ell(y)$ the vector $\ell(y) - \ell(y)$ is a feasible direction at $\ell(x)$: there is $\epsilon > 0$ such that $\ell(x) + \epsilon(\ell(y) - \ell(x)) \in \ell(X)$. With differentiability of Φ and the optimality of $\ell(x)$ w.r.t. (4.13) we obtain

$$\nabla\Phi(\ell(x))(\ell(y)-\ell(x)) = \sum_{r\in R} c_r\big(\ell_r(x)\big)\,(\ell_r(y)-\ell_r(x)) \geq 0. \tag{4.14}$$

Let $x_i \in X_i$ for some $i \in N$. For $S, T \in \mathcal{S}_i$ with $x_{iS} > 0$, choose $\epsilon \in (0, x_{iS}]$ arbitrary and define $\ell(y) = \ell(x) - \epsilon\mathbf{1}^S + \epsilon\mathbf{1}^T$. Per construction we have $\ell(y) \in \ell(X)$, hence with (4.14) we get:

$$\begin{aligned} 0 &\geq \sum_{r\in R} c_r\big(\ell_r(x)\big)\,(\ell_r(x)-\ell_r(y)) \\ &= \sum_{r\in R} c_r(\ell_r(x))\,(\epsilon\mathbf{1}_r^S - \epsilon\mathbf{1}_r^T) \\ &= \epsilon\Big(\sum_{r\in S} c_r(\ell_r(x)) - \sum_{r\in T} c_r(\ell_r(x))\Big). \end{aligned}$$

Division of the inequality with $\epsilon > 0$ yields (4.11). □

We obtain the following properties of WE.

Theorem 4.46. 1. WE $x \in X$ do exist.

2. For strictly nondecreasing $(c_r)_{r\in R}$ the load vector $(\ell_r(x))_{r\in R}$ is unique – but there may be different decompositions, cf. Exercise 4.40.

3. For nondecreasing $(c_r)_{r\in R}$, the cost vector $(c_r(\ell_r(x)))_{r\in R}$ is unique.

Proof. For 1.:
With Theorem 4.45 we get that every optimal solution to (4.13) is a WE. As Φ is continuous and $\ell(X)$ compact, with the theorem of Weierstrass the existence of an optimal solution is guaranteed.
For 2. and 3.:
Let $x, y \in X$ be two WE. With (4.12) we get for both WE the following variational inequalities:

$$\sum_{r\in R} c_r\big(\ell_r(x)\big)\,(\ell_r(y)-\ell_r(x)) \geq 0 \tag{4.15}$$

$$\sum_{r\in R} c_r\big(\ell_r(y)\big)\,(\ell_r(x)-\ell_r(y)) \geq 0 \tag{4.16}$$

Addition of (4.15) and (4.16) yields

$$\sum_{r\in R} \big(c_r\big(\ell_r(y)\big) - c_r\big(\ell_r(x)\big)\big)\,(\ell_r(x)-\ell_r(y)) \geq 0 \tag{4.17}$$

As every summand is nonpositiv (use that costs are nonnegative and nondecreasing), we get

$$\big(c_r\big(\ell_r(y)\big) - c_r\big(\ell_r(x)\big)\big)\,(\ell_r(x) - \ell_r(y)) = 0 \text{ for all } r \in R.$$

This condition implies that either $\ell_r(x) = \ell_r(y)$ or $c_r\big(\ell_r(y)\big) = c_r\big(\ell_r(x)\big)$ must hold. If costs are strictly nondecreasing, the optimal solution is unique. For nondecreasing costs we only get uniqueness of cost values $c_r\big(\ell_r(y)\big) = c_r\big(\ell_r(x)\big)$ for all $r \in R$ and x, y WE.

□

Chapter 5

Pricing in Resource Allocation Games

5.1 Model

A resource allocation model is compactly described by a tuple $I = (N, R, X, \pi)$, where $N = \{1, \ldots, n\}$ describes a nonempty finite set of players and $R = \{1, \ldots, m\}$ denotes a nonempty finite set of resources. The set $X := \times_{i \in N} X_i$ describes the combined strategy space of the players where $X_i \subseteq \mathbb{R}^m$ is the nonempty strategy space of player $i \in N$. For $x_i = (x_{ij})_{j \in R} \in X_i$ the entry $x_{ij} \in \mathbb{R}$ can be interpreted as the level of resource usage of player i for resource j.
We call the vector of resource usage $x = (x_{ij}) \in \mathbb{R}^{n \cdot m}$ a strategy profile. Given $x \in X$, we can define the load on resource $j \in R$ as $\ell_j(x) := \sum_{i \in N} x_{ij}$, where x_{ij} is the j-th component of x_i. We denote the set of feasible loads, or the load space, by the Minkowski sum $\ell(X) := \sum_{i \in N} X_i$. We assume that cost functions are parameterized by an exogenously given vector $u \in \mathbb{R}^m$ and depend on the own strategy vector only. In this regard, for $u \in \mathbb{R}^m$, the costs of a player $i \in N$ at the strategy profile $x_i \in X_i$ are given by $\pi_i(u, x_i)$ for some function $\pi_i : \mathbb{R}^m \times X_i \to \mathbb{R}$.
For a model I, a vector $u \in \mathbb{R}^m$ leads to a strategic game $G(u) = (N, X, (\pi_i(u, x_i))_{i \in N})$ and the vector u can be interpreted as the induced load of an equilibrium.
We are concerned with the problem of defining prices $\lambda_j \geq 0, j \in R$ on the resources in order to incentivize an efficient usage of the resources. If player i uses resource j at consumption level x_{ij}, she needs to pay $\lambda_j x_{ij}$. For any vector $u \in \mathbb{R}^m$, the quantities $\pi_i(u, x_i)$ and $\lambda^\intercal x_i$ are assumed to be normalized to represent the same unit (say money in Euro) and we assume that the private cost functions are quasi-linear: $\pi_i(u, x_i) + \lambda^\intercal x_i$.
We write $G(u, \lambda) = (N, X, (\pi_i(u, x_i) + \lambda^\intercal x_i)_{i \in N})$ as the resulting strategic game augmented with prices λ. If the parameter $u = (u_j)_{j \in R} \in \mathbb{R}^m$ represents a targeted load vector, then, the task is to find prices $\lambda \in \mathbb{R}^m_{\geq 0}$ so that an equilibrium of the game with prices $G(u, \lambda)$ realizes this load.

Definition 5.1 (Enforceability). Let I be a resource allocation model.

1. A vector $u \in \mathbb{R}^m$ is enforceable for I, if there is a tuple $(x^*, \lambda^*) \in X \times \mathbb{R}^m_{\geq 0}$ such that the following two conditions are satisfied:
   (a) $\ell_j(x^*) = u_j$ for all $j \in R$.
   (b) $x_i^* \in \arg\min_{x_i \in X_i} \{\pi_i(u, x_i) + (\lambda^*)^\intercal x_i\}$ for all $i \in N$.
2. A vector $u \in \mathbb{R}^m$ is weakly enforceable for I, if there is a tuple (x^*, λ^*) that satisfies (1b) but (1a) is replaced with $\ell(x^*) \leq u$ and λ^* satisfies $\ell_j(x^*) < u_j \Rightarrow$

$\lambda_j^* = 0$ for all $j \in R$.

If the model I is clear from the context, we say that $u \in \mathbb{R}^m$ is (weakly) enforceable.

Condition (1a) requires that x^* realizes the targeted load $\ell(x^*) = u$ while Condition (1b) implements x^* as a pure Nash equilibrium of the game augmented with prices $G(u, \lambda^*)$. Let us consider in the following two examples:

Example 5.2 (Toll Pricing in Networks). Let $G = (V, E)$ be a digraph and N a set of players (populations) with $(s_i, t_i, d_i), i \in N$, where s_i is the source, t_i the sink and $d_i > 0$ the demand from s_i to t_i. The set of edges E represents the set of resources and X_i is given by a s_i-t_i flow polytope. We may consider both, the nonatomic (Wardrop equilibrium) and the atomic (Nash equilibrium) setting. For a load $u \in \mathbb{R}^m$, the goal is to determine prices $\lambda_j^* \geq 0, j \in R$ such that a Wardrop equilibrium/Nash equilibrium x^* with $\ell(x^*) = u$ in the price-augmented nonatomic/atomic congestion game exists.

Example 5.3 (Market Equilibria). Suppose there are items $R = \{1, \dots, m\}$ for sale and there is a set $N = \{1, \dots, n\}$ of potential buyers. For every subset $S \subseteq R$ of items, player i experiences value $w_i(S) \in \mathbb{R}$ giving rise to a valuation function $w_i : 2^m \to \mathbb{R}, i \in N$. The market manager wants to determine a price vector $\lambda^* \in \mathbb{R}^m_{\geq 0}$ for selling the items such that every player receives a subset $S_i^* \subseteq R$ maximizing her quasi-linear utility $S_i^* \in \arg\max_{S_i \subseteq R}\{w_i(S_i) - \sum_{j \in S_i} \lambda_j^*\}$ and unsold items have prices equal to zero. The tuple $((S_i^*)_{i \in N}, \lambda^*)$ is known as a Walrasian competitive equilibrium.
This model fits into the framework by defining $I = (N, R, X, \pi)$ with $X_i = \{0, 1\}^m, i \in N$ representing the set of incidence vectors of R. The cost function $\pi_i := -\nu_i$ is given by the negative valuation function $\nu_i : \mathbb{R}^m \times X_i \to \mathbb{R}, (u, x_i) \mapsto w_i(R(x_i))$ for any $u \in \mathbb{R}^m$. One can easily verify that any strategy profile which weakly enforces the load $\mathbf{1} \in \mathbb{R}^m$ corresponds to a Walrasian competitive equilibrium and vice versa.

5.2 Connection to Lagrangean Duality in Optimization

Let us fix a model I. For $u \in \mathbb{R}^m$, we define the following minimization problem that we call master problem:

$$\inf_x \pi(u, x) \qquad (P(u))$$
$$\text{s.t.: } \ell(x) \leq u, \qquad (5.1)$$
$$x \in X,$$

where the objective function is defined as $\pi(u, x) := \sum_{i \in N} \pi_i(u, x_i)$.

The Lagrangian function for problem $P(u)$ becomes $L(x, \lambda) := \pi(u, x) + \lambda^\intercal(\ell(x) - u)$, $\lambda \in \mathbb{R}^m_{\geq 0}$, and we can define the Lagrangian-dual as: $\mu : \mathbb{R}^m_{\geq 0} \to \mathbb{R}$, $\mu(\lambda) = \inf_{x \in X} L(x, \lambda) = \inf_{x \in X}\{\pi(u, x) + \lambda^\intercal(\ell(x) - u)\}$. We assume that $\mu(\lambda) = -\infty$, if $L(x, \lambda)$ is not bounded from below on X. The dual problem is defined as:

$$\sup_{\lambda \geq 0} \mu(\lambda) \qquad (5.2)$$

Definition 5.4. Problem $P(u)$ has zero-duality gap, if there is $\lambda^* \in \mathbb{R}^m_{\geq 0}$ and $x^* \in X$ with $\pi(u, x^*) = \mu(\lambda^*)$. In this case, we say that the pair (x^*, λ^*) is primal-dual optimal.

If problem $P(u)$ has zero-duality gap, the two solutions $\lambda^* \in \mathbb{R}^m_{\geq 0}$ and $x^* \in X$ are optimal for their respective problems (5.2) and $P(u)$ and infima/suprema in the definition of μ become a minimum/maximum.
In the following, we prove a key structure, namely that $\min_{x \in X} L(x, \lambda)$ decomposes into independent problems, one for each player.

Lemma 5.5. Let $\lambda \in \mathbb{R}^m_{\geq 0}$. For a problem of type $P(u)$, the following holds true:

$$x^* \in \arg\min_{x \in X} L(x, \lambda) \Leftrightarrow x_i^* \in \arg\min_{x_i \in X_i} \{\pi_i(u, x_i) + \lambda^\intercal x_i\} \text{ for all } i \in N. \tag{5.3}$$

Proof. We calculate:

$$\begin{aligned}\min_{x \in X} L(x, \lambda) &= \min_{x_i \in X_i, i \in N} \left\{ \left[\sum_{i \in N} (\pi_i(u, x_i) + \lambda^\intercal x_i \right] - \lambda^\intercal u \right\} \\ &= \sum_{i \in N} \min_{x_i \in X_i} \{\pi_i(u, x_i) + \lambda^\intercal x_i\} - \lambda^\intercal u,\end{aligned}$$

where the first equality follows by the linearity of $\ell(x)$ w.r.t. $x_i, i \in N$ and the last equality by the assumption that $\pi_i(u, x_i)$ only depends on $x_i \in X_i$ and the fact that $\lambda^\intercal u$ does not depend at all on x. □

We obtain the following main result concerning the enforceability of a load u.

Theorem 5.6. The following equivalences hold for I.

1. A vector $u \in \mathbb{R}^m$ is enforceable via (x^*, λ^*) if and only if (x^*, λ^*) has zero duality gap for $P(u)$ and x^* satisfies (5.1) with equality.

2. A vector $u \in \mathbb{R}^m$ is weakly enforceable via (x^*, λ^*) if and only if (x^*, λ^*) has zero duality gap for $P(u)$.

Proof. For the proof we only show 2., since 1. follows from 2. as the additional condition $\ell(x^*) = u$ holds true for both statements of 1.
For 2.: $\Leftarrow$: Assume there are $\lambda^* \in \mathbb{R}^m_{\geq 0}, x^* \in X$ with $\ell(x^*) \leq u$ so that $\mu(\lambda^*) = \pi(u, x^*)$. We obtain

$$\begin{aligned}\mu(\lambda^*) &= \min_{x \in X} \{\pi(u, x) + (\lambda^*)^\intercal(\ell(x) - u)\} \leq \pi(u, x^*) + (\lambda^*)^\intercal(\ell(x^*) - u) \\ &\leq \pi(u, x^*) = \mu(\lambda^*).\end{aligned}$$

Hence, all inequalities must be tight leading to $(\lambda^*)^\intercal(\ell(x^*) - u) = 0$ as claimed. It remains to prove Condition (1b). With $x^* \in \arg\min_{x \in X} L(x, \lambda^*)$ we get

$$x^* \in \arg\min_{x \in X} L(x, \lambda^*) \underset{\text{Lem. 5.5}}{\Leftrightarrow} x_i^* \in \arg\min_{x_i \in X_i} \{\pi_i(u, x_i) + \sum_{j \in R} \lambda_j^* x_{ij}\} \text{ for all } i \in N.$$

$\Rightarrow$: Let $u \in \mathbb{R}^m$ be weakly enforceable via $(x^*, \lambda^*) \in X \times \mathbb{R}^m_{\geq 0}$, that is, $\ell(x^*) \leq u$, $(\lambda^*)^\intercal(\ell(x^*) - u) = 0$ and $x_i^* \in \arg\min_{x_i \in X_i}\{\pi_i(u, x_i) + (\lambda^*)^\intercal x_i\}$ for all $i \in N$. We calculate

$$\begin{aligned}\mu(\lambda^*) &= \inf_{x \in X}\{\pi(u, x) + (\lambda^*)^\intercal(\ell(x) - u)\} \\ &= \pi(u, x^*) + (\lambda^*)^\intercal(\ell(x^*) - u) \qquad (5.4) \\ &= \pi(u, x^*), \qquad (5.5)\end{aligned}$$

where (5.4) follows from Lemma 5.5 and (5.5) uses the condition $(\lambda^*)^\intercal(\ell(x^*) - u) = 0$. Hence, strong duality holds for the pair (x^*, λ^*). □

The theorem does not rely on any further assumption on the functions $\pi_i(u, x_i), i \in N$ nor on the feasible sets X_i but only on the duality gap of $P(u)$. This is in particular interesting as several classes of optimization problems are known to have zero duality gap even without convexity of feasible sets and objective functions.

In cost minimization games, the feasible sets X_i usually contain some sort of covering conditions on the resource consumption. For example in network routing, one needs to send some prescribed amount of flow.

In this regard, we introduce a natural candidate set of vectors u for which we know that any feasible solution satisfying (5.1) does so with equality.

Definition 5.7. A vector $u \in \mathbb{R}^m$ is called minimal w.r.t. $\ell(X)$, if

$$\{\tilde{u} \in \mathbb{R}^m : \tilde{u} \leq u\} \cap \ell(X) = \{u\}$$

Exercise 5.8. A vector $u \in \mathbb{R}^m$ is minimal w.r.t. $\ell(X)$, if and only if there are strictly increasing functions $h_j : \mathbb{R} \to \mathbb{R}, j \in R$ such that

$$u \in \arg\min_{\tilde{u} \in \ell(X)} \sum_{j \in R} h_j(\tilde{u}_j).$$

Corollary 5.9. Let $u \in \mathbb{R}^m$ be minimal w.r.t. $\ell(X)$. Then, the following two statements are equivalent:

1. u is enforceable via price vector $\lambda^* \in \mathbb{R}^m_{\geq 0}$ and $x^* \in X$.
2. (x^*, λ^*) satisfies $\pi(u, x^*) = \mu(\lambda^*)$.

5.3 Convexified Games

So far, the strategy spaces $X_i, i \in N$ and the cost functions $\pi_i(u, \cdot), i \in N$ of a model I were not restricted and are allowed to be non-convex. For instance integrality restrictions in $X_i \subseteq \mathbb{Z}^m, i \in N$ are allowed. In what follows, we connect I with a related convexified resource allocation model I^{conv}.

For $Z \subseteq \mathbb{R}^m$ denote $\text{conv}(Z) := \cap\{K \subseteq \mathbb{R}^m | Z \subseteq K, K \text{ convex}\}$ the convex hull of Z.

Definition 5.10. For a resource allocation model $I = (N, R, X, \pi)$ and load u, the associated convexified model I^{conv} is defined as

$$I^{conv} = (N, R, X^{conv}, (\phi_i)_{i \in N}),$$

where $X^{conv} := \times_{i \in N} \operatorname{conv}(X_i)$ and for all $\tilde{u} \in \mathbb{R}^m$:

$$\begin{aligned} \phi_i(\tilde{u}, \cdot) &: \operatorname{conv}(X_i) \to \mathbb{R} \cup \{-\infty\} \\ \bar{x}_i &\mapsto \inf_{\alpha_{ik}, x_i^k} \left\{ \sum_{k=1}^{m+1} \alpha_{ik} \pi_i(u, x_i^k) \;\middle|\; \begin{array}{l} \sum_{k=1}^{m+1} \alpha_{ik} x_i^k = \bar{x}_i, \alpha_i \in \Lambda, \\ x_i^k \in X_i, 1 \leq k \leq m+1 \end{array} \right\} \end{aligned} \tag{5.6}$$

in which $\Lambda := \{\alpha \in \mathbb{R}_{\geq 0}^{m+1} | \mathbb{1}^\top \alpha = 1\}$. Note that ϕ_i is constant in $\tilde{u} \in \mathbb{R}^m$.

Theorem 5.11. Let $x^* \in X$, $\lambda^* \in \mathbb{R}_{\geq 0}^m$. The following statements are equivalent.

1. u is enforceable for I via (x^*, λ^*).
2. $\phi_i, i \in N$ are real-valued functions, u is enforceable for I^{conv} via (x^*, λ^*) and $\phi(u, x^*) = \pi(u, x^*)$ holds.

The equivalence remains true by replacing the term "enforceable" with "weakly enforceable".

Proof. We first derive another description of the dual for the master problem $P(u)$ of I. We get for all $\lambda \in \mathbb{R}_{\geq 0}^m$:

$$\begin{aligned} \mu(\lambda) &= \inf_{x_i \in X_i, i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^\top (\ell(x) - u) \\ &= \sum_{i \in N} \inf_{\alpha_{ik}, x_i^k} \left\{ \sum_{k=1}^{m+1} \alpha_{ik} \left[\pi_i(u, x_i^k) + \lambda^\top x_i^k \right] \;\middle|\; \begin{array}{l} \alpha_i \in \Lambda, x_i^k \in X_i, \\ 1 \leq k \leq m+1 \end{array} \right\} - \lambda^\top u \quad (5.7) \\ &= \inf_{\bar{x} \in X^{conv}} \sum_{i \in N} \phi_i(u, \bar{x}_i) + \lambda^\top (\ell(\bar{x}) - u) \quad (5.8) \end{aligned}$$

where (5.7) follows by the linearity of $\alpha_i \mapsto \sum_{k=1}^{m+1} \alpha_{ik} \left[\pi_i(u, x_i^k) + \lambda^\top x_i^k \right]$. Equation (5.8) follows as $\sum_{k=1}^{m+1} \alpha_{ik} \lambda^\top x_i^k = \lambda^\top \sum_{k=1}^{m+1} \alpha_{ik} x_i^k$ holds.

Now we are ready to prove 1.⇔2..

1.⇒2.: By Theorem 5.6, we have $\mu(\lambda^*) = \pi(u, x^*) > -\infty$ and thus, $\phi_i, i \in N$ need to be real-valued by the above description of μ. Subsequently I^{conv} belongs to the resource allocation model defined in Section 5.1. The dual $\mu^{conv}(\lambda)$ of the master problem $P(u)$ of I^{conv} is then given by the expression in (5.8). We get $\pi(u, x^*) = \mu(\lambda^*) = \mu^{conv}(\lambda^*) \leq \phi(u, x^*)$, where the last inequality follows from weak duality. Since $\phi(u, x) \leq \pi(u, x)$ for all $x \in X$, we get $\mu^{conv}(\lambda^*) = \phi(u, x^*)$ and $\phi(u, x^*) = \pi(u, x^*)$. Thus the result follows by Theorem 5.6 and the fact that the load of x^* in I equals the load of x^* in I^{conv}.

2.⇒1.: As $\phi_i, i \in N$ are real-valued, I^{conv} belongs to the resource allocation model defined in Section 5.1. Thus, the result follows by Theorem 5.6 together with $x^* \in X$, $\pi(u, x^*) = \phi(u, x^*)$, $\mu^{conv}(\lambda^*) = \mu(\lambda^*)$ and using that the respective induced loads of x^* in I and x^* in I^{conv} coincide. □

5.4 LP-based characterizations of enforceability

We now discuss a special class of models I, which allow for an LP-based characterization of enforceability. The main property needed is a special structure of the Lagrangian-dual function of the master problem $P(u)$.

Assumption 5.12. For every $i \in N$, there exist $\{x_i^1, \dots, x_i^{k_i}\} \subseteq X_i$ for some $k_i \in \mathbb{N}$ such that the dual of $P(u)$ may be represented as follows:

$$\mu(\lambda) = \min_{x_i \in \{x_i^1, \dots, x_i^{k_i}\}, i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^\intercal(\ell(x) - u)$$

Clearly, this assumption is fulfilled in the important case of finite models where the strategy sets are finite point sets (see Fig. 5.1 left). We will show in Corollary 5.15 that this assumption also holds for concave models, that are, models where the convex hull of each $X_i, i \in N$ is finitely generated and the functions $\pi_i(u, x_i), i \in N$ are concave on $\text{conv}(X_i)$ (see Fig. 5.1 right).

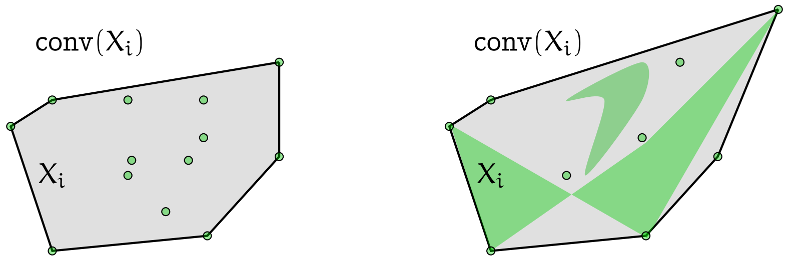


Figure 5.1: Left is the scenario of X_i consisting of a finite point set. Right, X_i may consist of connected components (in green) and isolated points but the convex hull is assumed to be finitely generated and additionally $\pi_i(u, x_i)$ are assumed to be concave on $\text{conv}(X_i)$.

For a model I fulfilling Assumption 5.12, we define the following LP in the variable $\alpha = (\alpha_i)_{i \in N}$.

$$\min_{\alpha} \sum_{i \in N} \pi_i^\intercal \alpha_i \tag{LP(u)}$$

$$\text{s.t.: } \ell(\alpha) \leq u, \tag{5.9}$$

$$\alpha_i \in \Lambda_i, i \in N,$$

where $\pi_i := (\pi_i(u, x_i^k))_{k \in \{1,\dots,k_i\}}$, $\ell(\alpha) := \sum_{i \in N} \sum_{k \in \{1,\dots,k_i\}} \alpha_{ik} x_i^k$ and $\Lambda_i := \{\alpha_i \in \mathbb{R}^{k_i}_{\geq 0} | \mathbf{1}^\intercal \alpha_i = 1\}, i \in N$.

Theorem 5.13. Let I be a model for which Assumptions 5.12 holds. Then, the following statements are equivalent.

1. The vector $u \in \mathbb{R}^m$ is enforceable for I.
2. There exists $x^* \in X$ with $\ell(x^*) = u$ and an optimal solution α^* of LP(u) such that $\pi(u, x^*) = \sum_{i \in N} \pi_i^\intercal \alpha_i^*$.

The equivalence remains true by replacing the term “enforceable” with “weakly enforceable” and replacing the condition $\ell(x^*) = u$ in Statement 2. by $\ell(x^*) \leq u$.

Proof. We first show that the duals of $LP(u)$ and $P(u)$ coincide. We get

$$\begin{aligned}
\mu^{LP}(\lambda) &:= \min_{\alpha_i \in \Lambda_i, i \in N} \sum_{i \in N} \pi_i^T \alpha_i + \lambda^T(\ell(\alpha) - u) \\
&= \min_{x_i \in \{x_i^1, \ldots, x_i^{k_i}\}, i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^T(\ell(x) - u) && (5.10) \\
&= \mu(\lambda), && (5.11)
\end{aligned}$$

where (5.10) follows by the linearity of the objective function w.r.t. α_i, $i \in N$ and equation (5.11) by Assumption 5.12.

Let us denote by $\pi^*, (\pi^{LP})^*, \mu^*, (\mu^{LP})^*$ the respective optimal values of $P(u)$, $LP(u)$ and their duals. By weak duality of $P(u)$ and strong duality of $LP(u)$, we get

$$\pi^* \geq \mu^* = (\mu^{LP})^* = (\pi^{LP})^*.$$

Now for 1.$\Rightarrow$ 2., we have $\pi^* = \pi(u, x^*) = \mu(\lambda^*) = \mu^*$ for one $(x^*, \lambda^*) \in X \times \mathbb{R}^m_{\geq 0}$ with $\ell(x^*) = u$ by Theorem 5.6. Thus, $\pi^* = (\pi^{LP})^*$ and there is an optimal solution α^* of $LP(u)$ such that $\pi(u, x^*) = \sum_{i \in N} \pi_i^T \alpha_i^*$. For the converse 2.$\Rightarrow$ 1., we observe that by $\pi(u, x^*) = \sum_{i \in N} \pi_i^T \alpha_i^* = (\mu^{LP})^* = \mu^*$, there exists one $\lambda^* \in \mathbb{R}^m_{\geq 0}$ with $\pi(u, x^*) = \mu^* = \mu(\lambda^*)$. Therefore, the statement follows by Theorem 5.6 and the assumption that $\ell(x^*) = u$ holds. $\square$

In the following, we describe the consequences of Theorem 5.13 for the important cases of finite and concave models:

Definition 5.14. We call a model I

1. finite, if $X_i = \{x_i^1, \ldots, x_i^{k_i}\}$ for some $k_i \in \mathbb{N}$ and all $i \in N$.
2. concave, if for all $i \in N$, there exist $\{x_i^1, \ldots, x_i^{k_i}\} \subseteq X_i$ for some $k_i \in \mathbb{N}$ such that $\mathrm{conv}(X_i) = \mathrm{conv}\left(\{x_i^1, \ldots, x_i^{k_i}\}\right)$. Furthermore, $\pi_i(u, \cdot), i \in N$ can be extended to the domain $\mathrm{conv}(X_i)$ so that they are concave on $\mathrm{conv}(X_i)$.

Corollary 5.15. For a finite model I, the following statements are equivalent.

1. The vector $u \in \mathbb{R}^m$ is enforceable for I.
2. $LP(u)$ admits an integral optimal solution $\tilde{\alpha}$ for which (5.9) is tight.

For a concave model I, the following statements are equivalent.

3. The vector $u \in \mathbb{R}^m$ is enforceable for I.
4. $LP(u)$ admits an optimal solution $\tilde{\alpha}$ with $x_i^{\tilde{\alpha}} := \sum_{j=1}^{k_i} \tilde{\alpha}_{ij} x_i^j \in X_i, i \in N$ so that $\ell(x^{\tilde{\alpha}}) = u$ and $\pi(u, x^{\tilde{\alpha}}) = \sum_{i \in N} \pi_i^T \tilde{\alpha}_i$.

The equivalences remain true by replacing the term “enforceable” with “weakly enforceable” and removing the condition that (5.9) needs to be tight in Statement 2. and replacing $\ell(x^{\tilde{\alpha}}) = u$ in Statement 4. by $\ell(x^{\tilde{\alpha}}) \leq u$.

The proof and further complexity theoretic consequences can be found in the Appendix.

5.5 Aggregated Formulations

In this section we will consider so-called aggregated models:

Definition 5.16. A model I is called aggregated, if the cost functions are quasi-separable over the resources, that is,

$$\pi_i(u, x_i) = \sum_{j \in R} \pi_j(u) \cdot x_{ij},$$

where $\pi_j : \mathbb{R}^m \to \mathbb{R}_{\geq 0}$ denotes the per-unit cost on resource j mapping a vector u to the reals.

We will assume that the per-unit cost $\pi_j(u) > 0$ of a resource j is strictly positive. All results in this section can be extended to the case where the per-unit cost of resources may be zero (Exercise 5.23). For an aggregated model, we can assume w.l.o.g. that $x_i = x_i, i \in N$ (exercise).
It follows that the objective of the master problem $P(u)$ of an aggregated model I does not depend on the specific decomposition of a strategy distribution $x \in X$ but only on its induced load $\ell(x)$. The same holds true for the corresponding convexified model I^{conv} as the latter also belongs to the class of aggregated formulations, because $\pi_i(u, x_i) = \phi_i(u, x_i)$, $i \in N$ holds, since $\pi_i(u, x_i)$ is linear in x_i. Note that $X^{\text{conv}} = \times_{i \in N} \operatorname{conv}(X_i)$. This insight leads to the following optimization problems and subsequent lemma.

$$\begin{aligned} \min_{\tilde{u}} \ & \pi(u)^\intercal \tilde{u} \qquad (\tilde{P}(u)) \\ \text{s.t.:} \ & \tilde{u} \leq u, \\ & \tilde{u} \in \ell(X), \end{aligned}$$

$$\begin{aligned} \min_{\tilde{u}} \ & \pi(u)^\intercal \tilde{u} \qquad (\tilde{P}^{\text{conv}}(u)) \\ \text{s.t.:} \ & \tilde{u} \leq u, \\ & \tilde{u} \in \ell(X^{\text{conv}}), \end{aligned}$$

where $\pi(u) := (\pi_j(u))_{j \in R}$.

Lemma 5.17. For an aggregated model I, the following assertions are equivalent:

1. (x^*, λ^*) is primal-dual optimal for $P(u)$.
2. $(\ell(x^*), \lambda^*)$ is primal-dual optimal for $\tilde{P}(u)$.

The analogues statements also hold for I^{conv} and $\tilde{P}^{\text{conv}}(u)$.

Proof. As I^{conv} belongs to the class of aggregated models, it is sufficient to show the statements for the original model I. To verify the stated equivalence, we first observe that the following description of $\pi(u, x)$ holds for any $x \in X$:

$$\pi(u, x) = \sum_{i \in N} \pi_i(u, x_i) = \sum_{i \in N} \sum_{j \in R} \pi_j(u) x_{ij} = \sum_{j \in R} \pi_j(u) \sum_{i \in N} x_{ij} = \pi(u)^\intercal \ell(x).$$

Thus, it suffices to show that the duals coincide. We denote by $\mu(\lambda)$ the dual of $P(u)$, and by $\tilde{\mu}(\lambda)$ the dual of $\tilde{P}(u)$, respectively, and observe:

$$\mu(\lambda) = \inf_{x \in X} \pi(u, x) + \lambda^\intercal(\ell(x) - u) = \inf_{x \in X} \pi(u)^\intercal \ell(x) + \lambda^\intercal(\ell(x) - u)$$

$$= \inf_{\tilde{u} \in \ell(X)} \pi(u)^\intercal \tilde{u} + \lambda^\intercal (\tilde{u} - u) = \tilde{\mu}(\lambda).$$

□

In the following, we want to better understand which vectors u are enforceable. In this regard, we define the following refinements of minimality of a load u.

Definition 5.18 (Concave and Linear Minimality).

1. $u \in \ell(X)$ is called concave-minimal w.r.t. $\ell(X)$, if there exists a strictly increasing, differentiable and concave function $h : \mathbb{R}^m \to \mathbb{R}$ such that $u \in \arg\min_{\tilde{u} \in \ell(X)} h(\tilde{u})$.
2. $u \in \ell(X)$ is called linearly-minimal w.r.t. $\ell(X)$, if u is concave-minimal for a linear function $h(\tilde{u}) = a^\intercal \tilde{u}$ with $a \in \mathbb{R}^m_{>0}$.

Remark 5.19. Obviously we have the following relationship among the different notions of minimality of u w.r.t. $\ell(X)$:

$$\text{linearly-minimal} \subseteq \text{concave-minimal} \subseteq \text{minimal}.$$

Now we give a complete characterization of enforceability of u for I.

Theorem 5.20. Let $u \in \ell(X)$ be a feasible load vector for the aggregated model I. Then the following statements are equivalent.

1. u is enforceable for I.
2. u is enforceable for I^{conv}.
3. $\tilde{P}(u)$ has zero duality gap and u is minimal w.r.t. $\ell(X)$.
4. $\tilde{P}^{conv}(u)$ has zero duality gap and u is minimal w.r.t. $\text{conv}(\ell(X))$.
5. u is linearly-minimal w.r.t. $\ell(X)$.
6. u is concave-minimal w.r.t. $\ell(X)$.

Proof. The equivalence 1.⇔2. follows immediately by Theorem 5.11, the fact that $\phi(u, x) = \pi(u, x) = \pi(u)^\intercal \ell(x)$ holds for any $x \in X$ and $u \in \ell(X)$ is feasible for I.
Next we show 1.⇔3. which also shows 2.⇔4. since I^{conv} belongs to the class of aggregated models and $\ell(X^{conv}) = \text{conv}(\ell(X))$ holds.
1.⇔3.: By Theorem 5.6, a vector u is enforceable if and only if there exists $(x^*, \lambda^*) \in X \times \mathbb{R}_{\geq 0}$ primal-dual optimal for $P(u)$ of I with $\ell(x^*) = u$. By Lemma 5.17, this is again equivalent to the existence of $\lambda^* \in \mathbb{R}^m_{\geq 0}$ such that (u, λ^*) is primal-dual optimal for $\tilde{P}(u)$. Since $\pi(u) \in \mathbb{R}^m_{>0}$, the optimality of u for $\tilde{P}(u)$ is equivalent to u being minimal w.r.t. $\ell(X)$. Thus the existence of a primal-dual optimal $(u, \lambda^*) \in \ell(X) \times \mathbb{R}_{\geq 0}$ for $\tilde{P}(u)$ is equivalent to $\tilde{P}(u)$ having zero duality gap and u being minimal w.r.t. $\ell(X)$.
It remains to show 1.⇔5. and 6.⇔5. For 1.⇔5. we start with the following observation:

By 1.⇔3., u is enforceable for I if and only if there exists $\lambda^* \in \mathbb{R}^m_{\geq 0}$ such that (u, λ^*) is primal-dual optimal for $\tilde{P}(u)$, i.e.:

$$\min_{\tilde{u} \in \ell(X)} \pi(u)^\intercal \tilde{u} + (\lambda^*)^\intercal (\tilde{u} - u) = \pi(u)^\intercal u.$$

By adding to both sides the term $(\lambda^*)^\intercal u$, this is again equivalent to

$$\min_{\tilde{u} \in \ell(X)} (\pi(u) + \lambda^*)^\intercal \tilde{u} = (\pi(u) + \lambda^*)^\intercal u. \tag{5.12}$$

1.⇒5.: With the above argumentation, if u is enforceable for I, (5.12) holds for a $\lambda^* \in \mathbb{R}^m_{\geq 0}$ and we can define $a := (\pi(u) + \lambda^*) \in R^m_{>0}$, since $\pi(u) \in \mathbb{R}^m_{>0}$. Thus the so-defined vector a meets the requirement of Definition 5.18.
5.⇒1.: Let u be linearly-minimal w.r.t. $\ell(X)$. Then, there exists a vector $a \in \mathbb{R}^m_{>0}$ which meets the requirements of Definition 5.18. We can assume w.l.o.g. that $\pi(u) \leq a$ since we can just scale a sufficiently otherwise. By setting $\lambda^* := a - \pi(u) \in \mathbb{R}^m_{\geq 0}$ we obtain a non-negative vector for which (5.12) holds.
6.⇒5.: Let $u \in \arg\min_{\tilde{u} \in \ell(X)} h(\tilde{u})$ be given. With the concavity of h, we get: $u \in \arg\min_{\tilde{u} \in \mathrm{conv}(\ell(X))} h(\tilde{u})$. As $\mathrm{conv}(\ell(X))$ is obviously convex, for any $\tilde{u} \in \mathrm{conv}(\ell(X))$, the direction vector $(u - \tilde{u})$ belongs to the tangent cone of $\mathrm{conv}(\ell(X))$ at u. With the differentiability of h, we get as a first order necessary optimality condition:

$$\nabla h(u)(u - \tilde{u}) \leq 0 \text{ for all } \tilde{u} \in \mathrm{conv}(\ell(X)).$$

Thus, as h is strictly increasing and concave in $\tilde{u}$, we have $a := \nabla h(u) \in \mathbb{R}^m_{>0}$ and the requirements of Definition 5.18. are fulfilled.
5.⇒6.: By Remark 5.19. □

From the latter proof follows directly another insight:

Corollary 5.21. If λ^* enforces u for I, then every $\tilde{\lambda}$ on the half-ray

$$\{\tilde{\lambda} \in \mathbb{R}^m_{\geq 0} \mid \tilde{\lambda} = d \cdot (\pi(u) + \lambda^*) - \pi(u),\ d \in \mathbb{R}_{\geq 0}\}$$

also enforces u for I. In particular, if u is enforceable, then there exists a price-vector $\tilde{\lambda}$ which enforces u and $\tilde{\lambda}_j = 0$ holds for at least one $j \in R$.

We conclude the section with an example showing that not every minimal u is also linearly-minimal/enforceable even for a two player model with $X_i, i = 1, 2$ consisting of strategy sets with three vectors each.

Exercise 5.22. Show that there exist aggregated models where not every minimal load u w.r.t. $\ell(X)$ is also linearly-/concave-minimal w.r.t. $\ell(X)$. Subsequently (?), not every minimal load u w.r.t. $\ell(X)$ needs to be enforceable.
Hint: Figure 5.2

Exercise 5.23. Show that Theorem 5.20 still holds in the general case of arbitrary per unit-costs $\pi(u) \in \mathbb{R}^m_{\geq 0}$ when we use the the following relaxed versions of minimality instead of the original ones.

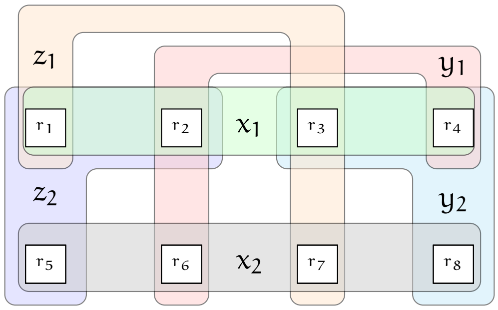


Figure 5.2: Representation of the strategy space of the game constructed in the proof of Exercise 5.22.

1. A vector $u \in \ell(X)$ is called minimal with respect to $\ell(X)$ and $\pi(u)$, if for all $\tilde{u} \in \{\bar{u} \in \mathbb{R}^m : \bar{u} \leq u\} \cap \ell(X)$, the equalities $\tilde{u}_j = u_j$ for all $j \in R(\pi(u)) := \{r \in R \mid \pi_r(u) > 0\}$ hold.

2. $u \in \ell(X)$ is called concave-minimal w.r.t. $\ell(X)$ and $\pi(u)$, if there exists a non-decreasing, differentiable and concave function $h : \mathbb{R}^m \to \mathbb{R}$ such that $u \in \arg\min_{\tilde{u} \in \ell(X)} h(\tilde{u})$ and $\frac{\partial}{\partial u_j} h(u) = 0$ implies $\pi_j(u) = 0$ for all $j \in R$.

3. $u \in \ell(X)$ is called linearly-minimal w.r.t. $\ell(X)$ and $\pi(u)$, if u is concave-minimal for a linear function $h(\tilde{u}) = a^T \tilde{u}$ with $a \in \mathbb{R}^m_{\geq 0}$.

5.6 Applications in Atomic Congestion Games

In this section we discuss the applicability of our framework for resource-weighted atomic congestion games where we even allow for player-specific cost functions on the resources, that is, the costs of player $i \in N$ are defined by $C_i(x) := \pi_i(\ell(x), x_i)$ for a function $\pi_i : \mathbb{R}^m \times X_i \to \mathbb{R}$ defined as

$$\pi_i(\ell(x), x_i) := \sum_{j \in R} c_{ij}(\ell_j(x)) x_{ij}, \tag{5.13}$$

where $c_{ij} : \mathbb{R}_{\geq 0} \to \mathbb{R}, i \in N, j \in R$ and the strategy space is given as $X_i \subset \times_{j \in R} \{0, d_{ij}\}, i \in N$. The definition in (5.13) shows that the private cost $C_i(x)$ depends not only on the own strategy x_i but also on the aggregated load vector $\ell(x)$ which in turn depends on the strategies of all players. Thus, the class of atomic congestion games does not fit into the model considered so far.
However, also in atomic congestion games, the question of whether or not the players can be incentivized by prices in order to realize a targeted load vector u is of particular interest and leads to the following definition of enforceability for atomic congestion games:

Definition 5.24 (Enforceability for Atomic Congestion Games). Consider an atomic congestion game $G^{cg} = (N, X, (C_i(x))_{i \in N})$. The vector $u \in \mathbb{R}^m$ is enforceable for G^{cg}, if there is a tuple $(x^*, \lambda^*) \in X \times \mathbb{R}^m_{\geq 0}$ such that the following two conditions are satisfied:

1. $\ell_j(x^*) = u_j$ for all $j \in R$.

2. $x_i^* \in \arg\min_{x_i \in X_i} \left\{\pi_i(\ell(x_i, x_{-i}^*), x_i) + (\lambda^*)^\intercal x_i\right\}$ for all $i \in N$.

As previously, Condition 1. requires that x^* realizes the targeted load $\ell(x^*) = u$ while Condition 2. implements x^* as a pure Nash equilibrium of the atomic congestion game augmented with prices $G^{cg}(\lambda^*) := (N, X, (C_i(x) + (\lambda^*)^\intercal x_i))$.
In order to apply the framework to the current setting, we define $I := (N, R, X, \pi)$ with N, X, R, π as described above and $x_i = x_i$. In the case of nondecreasing cost functions $c_{ij}, j \in R, i \in N$, the following lemma implicates that whenever u is enforceable for I via (x^*, λ^*), then u is also enforceable for G^{cg} via (x^*, λ^*).

Lemma 5.25. Assume that cost functions $c_{ij}, i \in N, j \in R$ are nondecreasing. Let $x^* \in X$ and fix $u := \ell(x^*)$. Then x^* is a pure Nash equilibrium of the atomic congestion game augmented with prices $G^{cg}(\lambda^*)$ for $\lambda^* \in \mathbb{R}^m_{\geq 0}$, if

$$x_i^* \in \arg\min_{x_i \in X_i} \left\{ \sum_{j \in R} c_{ij}(u_j) x_{ij} + (\lambda^*)^\intercal x_i \right\} \text{ for all } i \in N. \tag{5.14}$$

Proof. Let $i \in N$ and $x_i \in X_i$. For x_i denote $R(x_i) := \{j \in R | x_{ij} = d_{ij}\}$ the support of x_i. We calculate

$$\begin{aligned}
\pi_i(\ell(x^*), x_i^*) + (\lambda^*)^\intercal x_i^* &= \sum_{j \in R(x_i^*)} (c_{ij}(u_j) + \lambda_j^*) x_{ij}^* \\
&\leq \sum_{j \in R(x_i)} (c_{ij}(u_j) + \lambda_j^*) x_{ij} && (5.15)\\
&\leq \sum_{j \in R(x_i) \setminus R(x_i^*)} (c_{ij}(u_j + d_{ij}) + \lambda_j^*) x_{ij} + \sum_{j \in R(x_i) \cap R(x_i^*)} (c_{ij}(u_j) + \lambda_j^*) x_{ij} && (5.16)\\
&= \pi_i(\ell(x_i, x_{-i}^*), x_i) + (\lambda^*)^\intercal x_i,
\end{aligned}$$

where (5.15) follows from the optimality of x_i^* for problem (5.14) and (5.16) follows from the monotonicity of the cost functions $c_{ij}, i \in N, j \in R$. □

Clearly the condition (5.14) is only sufficient for enforcing u and not necessary. Prices λ^* that induce equilibria x^* with the property stated in (5.14) are also called cost-balancing.

Exercise 5.26. We consider an (atomic) symmetric network congestion game G^{cg} with strictly increasing, nonnegative edge costs $c_e : \mathbb{R}_+ \to \mathbb{R}_+$. The game is therefore described by a graph $G = (V, E)$ with a common source $s \in V$ and a common sink $t \in V$. All players have the same stragy space, namely the set of all s-t paths.
Show that for any social optimum x^{soc} for G^{cg}, there exist prices λ^* such that x^{soc} is a Nash equilibrium of the price-augmented game $G^{cg}(\lambda^*)$.

Exercise 5.27. We consider a nonatomic network congestion game: Let $G = (V, E)$ be a directed graph and $N := \{1, \dots, n\}$ a set of populations, where each population $i \in N$ has a demand $d_i > 0$ that has to be routed from a source $s_i \in V$ to a destination $t_i \in V$. Thus, the strategy space of a population $i \in N$ is the s_i-t_i flow polytope with flow value equal to d_i. There are continuous cost functions $c_{ie} : \mathbb{R}^m_{\geq 0} \to \mathbb{R}_{\geq 0}, i \in N, e \in E$ which may depend on the population identity and also on the aggregate load vector, that is, the costs of a single edge may depend not only on the load on the respective edge, but on the load on all edges.

Let u be minimal w.r.t. $\ell(X)$. Show that there exist prices (tolls) for the edges such that a Wardrop equilibrium (w.r.t. the sum of edge costs and edge prices) exists which realizes the load u.

Chapter 6

Combinatorial Auctions

In this section, we present a few examples from the area of mechanism design. The fundamental questions that one attempts to address in mechanism design is the following: Assuming that players act strategically, how should we design the rules of the game such that the players' strategic behavior leads to a certain desirable outcome of the game? As a motivating example, we first consider one of the simplest auctions, known as Vickrey Auction. We then turn to more general combinatorial auctions.

6.1 Vickrey Auction

Suppose there is an auctioneer who wishes to auction off a single item. We are given:

- A set of players N.
- For every player $i \in N$ a private value $v_i \geq 0$ for the good, which specifies how much the item is worth to player i; one may think of it as the monetary value that the item has for player i. It is important to note that this valuation is private, i.e., it is only known to player i himself and nobody else.
- Every player i submits a sealed bid $b_i \geq 0$, to the auctioneer, which can be thought of as the maximum amount player i declares to be willing to pay for the item. To whom should the auctioneer give the item and at what price?
- A mechanism can be thought of as a protocol that the auctioneer runs in order to make this decision. That is, based on the submitted bids $(b_i)_{i \in N}$, the mechanism determines:
  - <u>An allocation</u>:
  $$x_i := \begin{cases} 1 & \text{if } i \text{ wins,} \\ 0 & \text{else.} \end{cases}$$
  The unique player i in N with $x_i = 1$ is called the winner.
  - <u>A payment</u>: the winning player has to pay for the item a price $p_i \geq 0$ with $p_i \leq b_i$; all other players pay nothing, i.e. $p_j = 0$ for all $j \in N \setminus \{i\}$.

  We denote by $M(b) = (x, p)$ the outcome of the mechanism M for the profile b.
- Define the net utility of $i \in N$ w.r.t. the outcome (x, p) as $u_i(x, p) := x_i(v_i - p_i)$ (quasi-linear utility).

Input: Collect the bids $(b_i)_{i \in N}$ of all players.
Output: Outcome (x, p)
1: Let $i \in N$ with $b_i = \max_{j \in N} b_j$
(breaking ties by choosing i with smallest index)
2: Set $x_i := 1$ and $x_j := 0$ for all $j \neq i$.
3: Define $p_i := \max_{j \neq i} b_j$ and $p_j := 0$ for all $j \neq i$.
4: **return** (x, p)

Algorithm 6.1.1: SECOND-PRICE-AUCTION

- Every player $i \in N$ wants to maximize her utility u_i.

Note that the outcome (x, p)only depends on the bid vector $(b_i)_{i \in N}$ whereas the utility of i depends on v_i and (x, p). A player can use her bid b_i strategically in order to influence the outcome. The goal is develop a mechanism $M(b)$ that, for given bids $b := (b_i)_{i \in N}$, determines an outcome (x, p) satisfying the following properties:

(1) Strategyproofness (sp): Every player maximizes his utility by bidding truthfully, i.e., $b_i = v_i$.

(2) Efficiency: Assuming that every player bids truthfully, the mechanism computes an outcome that maximizes the social welfare, i.e., among all possible outcomes it chooses one that maximizes the total valuation $\sum_{i \in N} x_i v_i$.

(3) Polynomial-time computability: The outcome should be computable in polynomial time.

Definition 6.1 (Strategyproofness). A mechanism M is sp, if for all $i \in N$, the strategy $b_i = v_i$ is a dominant strategy , i.e.,

$$u_i(M(v_i, b_{-i})) \geq u_i(M(b_i, b_{-i})) \quad \forall b_{-i}, \ \forall b_i.$$

As it turns out, there is a remarkable mechanism due to Vickrey that satisfies all these properties; this mechanism is also known as Vickrey auction or second-price auction (see Algorithm 6.1.1).

Theorem 6.2. Vickrey's second price auction is sp, efficient and polynomial. This statement even holds if the other players know the bids of the opponent players.

Proof. Consider player i and fix a bidding profile b_{-i} of the other players. Let $B = \max_{j \neq i} b_j$ be the highest bid if player i does not participate in the game.
Case 1: $v_i > B$.

$$\begin{aligned} b_i > v_i: &\quad u_i = v_i - B \\ b_i = v_i: &\quad u_i = v_i - B \\ b_i < v_i: &\quad u_i = \begin{cases} v_i - B & \text{if } b_i > B, \\ 0 & \text{else} \end{cases} \end{aligned}$$

$b_i = v_i$ is thus a dominant strategy for i.

Case 2: $v_i \leq B$.

$$b_i > v_i: \quad u_i = \begin{cases} v_i - B & \text{if } b_i > B, \\ 0 & \text{else} \end{cases}$$

$$b_i = v_i: \qquad u_i = 0$$

$$b_i < v_i: \qquad u_i = 0$$

In the second case, the utility is even below 0; $b_i = v_i$ is thus also a dominant strategy for i.

It is easy to see that the Vickrey Auction satisfies (P2) and (P3) as well. More specifically, it satisfies (P2) since it selects the winner i^* to be a player whose valuation is maximum, assuming that every bidder bids truthfully. Moreover, its computation time is linear in the number of players n. □

Moreover, we can prove that if a player does not bid truthfully, he may actually run the risk to be strictly worse off.

Lemma 6.3. For every bid $b_i \neq v_i$ of player i there is a bidding profile b_{-i} of the other players such that $u_i(b_{-i}, b_i) < u_i(b_{-i}, v_i)$.

Exercise 6.4. Prove the Lemma.

6.2 Transport Coordination

We want to send a good from a source s to a sink t in a directed graph $G = (V, E)$ with edge costs $c : E \rightarrow \mathbb{R}^+$. We assume that every edge $e \in E$ corresponds to a player $e \in N$. If player $e \in N$ transports the good along her edge, it costs c_e and only e has this information (private). The bid b_e of player e corresponds to the declared cost.

The coordinator chooses an s,t-path P in G w.r.t. $(b_e)_{e \in E}$ and selects payments $p_e \geq 0$ for all $e \in P$. We have: $x_e = 1$ if $e \in P$ and $x_e = 0$, else. The payoff u_e of e is given by $u_e = (p_e - c_e)\, x_e$.

We want to design a mechanism M that computes (x, p) with

1. $b_e = c_e$ is a dominant strategy for all $e \in E$.
2. Total transport costs are minimized; i.e., if everybody declares truthfully, we want to compute a shortest path.
3. M needs to be polynomially computable.

If we consider Fig. 6.1, we see that the problem reduces to a second price auction. We obtain an sp mechanis by chhosing an edge with minimal declared cost and let the payment be the second smallest cost. Hence we choose the edge with cost 1 and pay 4.

This intuition can be generalized: Consider Fig. 6.2 and suppose that e bids truthfully $b_e = c_e$. All other $e' \neq e$ bid $b_{e'}$.The shortest s,t-path P has cost 3 and e is an edge on this path. By how much could e increase her declared cost without decreasing the payoff? The payoff of e is 0, if $e \notin P$; e stays on the shortest path as long $b_e \leq 5$. The coordinator must ensure that the surplus for e with $b_e = 5$ is compensated by the payment. A payment of 5 would be enough for e.

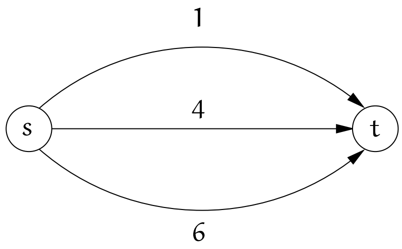


Figure 6.1: Transport-Coordination with parallel edges corresponds to a Second-Price-Auction.

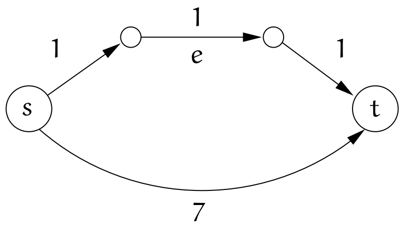


Figure 6.2: Transport-Coordination with declared edge costs (b_{-e}, c_e).

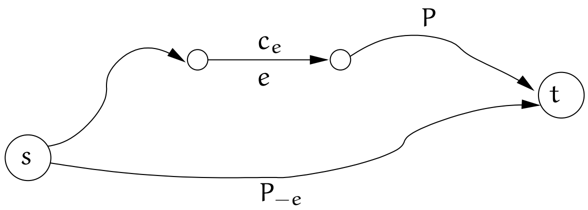


Figure 6.3: Payment $p_e := b_e + d_{G_{-e}}(P_{-e}) - d_G(P)$

Notation.

LetP be a shortest s,t-path in G w.r.t. $(b_e)_{e \in E}$. Let P_{-e} be a shortest s,t-path in $G_{-e} = (V, E_{-e})$ (w.r.t. b_{-e}) with $E_{-e} := E \setminus \{e\}$. For $e \in P$, definie

$$p_e := b_e + d_{G_{-e}}(P_{-e}) - d_G(P),$$

where $d_H(P)$ is the cost of P in H(see Fig. 6.3).

Input: Graph $G = (V, E)$ and declared costs $(b_e)_{e \in E}$
Output: (P, p)
1: Compute shortest s,t-path P in (G, b).
2: For every $e \in E$, let

$$p_e := \begin{cases} b_e + d_{G_{-e}}(P_{-e}) - d_G(P) & \text{if } e \in P \\ 0 & \text{else} \end{cases}$$

3: **return** (P, p)

Algorithm 6.2.1: Mechanism KW

We define the mechanis KW as in Algorithm 6.2.1.

Input: $(N, (b_i)_{i \in N})$
Output: $x^* \in X$, $(p_i)_{i \in N}$

1: Compute $x^* \in X$ with
$$x^* = \arg\max_{x \in X} \sum_{i \in N} b_i(x).$$
2: Compute payment for $i \in N$ via:
$$p_i := b_i(x^*) - \underbrace{\left(\max_{x \in X} \sum_{j \in N} b_j(x) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x) \right)}_{i\text{'s share of the overall utility}}.$$
3: **return** (x^*, p)

Algorithm 6.3.1: VCG-Mechanism

Theorem 6.5. KW satisfies (1)-(3).

Exercise 6.6. Prove Thm. 6.5.

6.3 vcg Mechanism

We now turn to a more general model of auctions.

- Set of players $N = \{1, \ldots, n\}$.
- Set $E = \{1, \ldots, m\}$ of objects.
- An allocation x is a function $x : E \to N \cup \{\perp\}$ (we map an object to $\perp$, if no player receives it).
- With $x^{-1}(i)$ we denote the objects that $i \in N$ received.
- X is a set of allocations of objects to players.
- Let $v_i : X \to \mathbb{R}^+$ denote the private utility function for i; for every $x \in X$, the value $v_i(x)$ is the utility of i for receiving the objects $x^{-1}(i)$.
- Let $b_i : X \to \mathbb{R}^+$ denote the declared utility function of player $i \in N$.

A mechanism M computes an allocation $x^* \in X$ and payments $(p_i)_{i \in N}$ given $(b_i)_{i \in N}$. Player i pays $p_i \in \mathbb{R}_+$ for $x^{-1}(i)$ and wants to maximize $u_i(x, p) = v_i(x) - p_i$. M is efficient, if the computed allocation x^* maximizes the overall utility, under truthful $b_i = v_i$ for all $i \in N$:
$$x^* = \arg\max_{x \in X} \sum_{i \in N} v_i(x).$$
Vickrey, Clarke and Groves devised the following VCG-Mechanism (see Algorithm 6.3.1).

Theorem 6.7. VCG is sp and efficient.

Proof. Obviously, VCG is efficient (assuming $b_i = v_i$ for all i).
VCG is sp: Let $i \in N$ and $b := (b_{-i}, b_i)$ with $\bar{b} := (b_{-i}, v_i)$. Let (x^*, p) and $(\bar{x}^*, \bar{p})$ be the output of VCG w.r.t. b and $\bar{b}$, respectively.
Observe:

$$\bar{b}_j(x) = b_j(x) \quad \forall x \in X \,\forall j \neq i \quad \text{and} \quad \bar{b}_i(x) = v_i(x) \quad \forall x \in X. \tag{6.1}$$

Moreover, $\bar{x}^*$ is chosen such that:

$$\sum_{j \in N} \bar{b}_j(\bar{x}^*) \geq \sum_{j \in N} \bar{b}_j(x) \quad \forall x \in X. \tag{6.2}$$

This implies:

$$\begin{aligned}
u_i(\bar{x}^*, \bar{p}) = v_i(\bar{x}^*) - \bar{p}_i = v_i(\bar{x}^*) - \Big[\bar{b}_i(\bar{x}^*) - \Big(\max_{x \in X} \sum_{j \in N} \bar{b}_j(x) - \max_{x \in X} \sum_{j \in N, j \neq i} \bar{b}_j(x)\Big)\Big] \\
&\overset{(6.1)}{=} \max_{x \in X} \sum_{j \in N} \bar{b}_j(x) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x) \\
&= \sum_{j \in N} \bar{b}_j(\bar{x}^*) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x) \\
&\overset{(6.2)}{\geq} \sum_{j \in N} \bar{b}_j(x^*) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x) \\
&\overset{(6.1)}{=} \sum_{j \in N, j \neq i} b_j(x^*) + v_i(x^*) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x) \\
&= v_i(x^*) - \Big[b_i(x^*) - \Big(\max_{x \in X} \sum_{j \in N} b_j(x) - \max_{x \in X} \sum_{j \in N, j \neq i} b_j(x)\Big)\Big] \\
&= v_i(x^*) - p_i = u_i(x^*, p).
\end{aligned}$$

For i the strategy $(b_i = v_i)$ is thus dominant. □

Although the VCG mechanism satisfies strategyproofness and efficiency, it is highly computationally intractable. In particular, there are two sources of inefficiency:

1. Collecting the bids of a single player already takes exponential time (in the number m of objects).
2. Computing the optimal allocation $x^* \in X$ may be a computationally hard problem. This problem is typically also called the allocation problem.
3. Payments are not bounded.

6.4 Single-Minded Bidders

In this section, we consider the special case of a combinatorial auction, where all bidders are said to be single-minded. More precisely, we say that player i is single-minded if there is some (private) set $\Sigma_i \subseteq E$ and a (private) value $\theta_i \geq 0$ such that for every $T \subseteq M$,

$$v_i(T) = \begin{cases} \theta_i & \text{if } T \supseteq \Sigma_i \\ 0 & \text{otherwise.} \end{cases}$$

Intuitively, player i is only interested in getting the items in Σ_i (or some more) and its valuation for these items is θ_i.

Note that in the single-minded case, the first source of inefficiency mentioned above vanishes since now every player simply reports a pair (S_i, b_i) (not necessarily equal to (Σ_i, θ_i)) to the auctioneer. Nevertheless, the second source of inefficiency remains, as will be proven below.

The allocation problem for the single-minded case is as follows:

> Given the bids $\{(S_i, b_i)_{i \in N}\}$, determine a subset $W \subseteq N$ of winners such that $S_i \cap S_j = \emptyset$ for every $i, j \in W$, $i \neq j$, with maximum social welfare $\sum_{i \in W} b_i$.

Theorem 6.8. The allocation problem for single-minded bidders is NP-hard.

Proof. We give a polynomial-time reduction from the NP-complete problem independent set. The independent set problem is as follows: Given an undirected graph $G = (V, E)$ and a non-negative integer k, determine whether there exists an independent set of size k.[1]

Given an instance (G, k) of the independent set problem, we can construct a single-minded combinatorial auction as follows: The set of items E corresponds to the edge set E of G. We associate a player $i \in N$ with every vertex $u_i \in V$ of G. The bundle that player i desires corresponds to the set of all adjacent edges, i.e., $S_i := \{e = \{u_i, u_j\} \in E\}$, and the value that i assigns to its bundle S_i is $b_i = 1$.

Now observe that a set $W \subseteq N$ of winners satisfies $S_i \cap S_j = \emptyset$ for every $i \neq j \in W$ iff the set of vertices corresponding to W constitute an independent set in G. Moreover, the social welfare obtained for W is exactly the size of the independent set. □

Given the above hardness result and insisting on polynomial-time computability, we are thus forced to consider approximation algorithms. The idea is to relax the efficiency condition and to ask for an outcome that is (only) approximately efficient. We call a mechanism α-efficient for some $\alpha \geq 1$ if it computes an allocation $x \in X$ (assuming truthful bidding $(S_i, b_i) = (\Sigma_i, \theta_i)$ for all $i \in N$) such that

$$\sum_{i \in N} v_i(x) \geq \frac{1}{\alpha} \max_{x \in X} \sum_{i \in N} v_i(x).$$

The proof of Theorem 6.8 even shows that the reduction is approximation preserving. That is, it specifies a bijection that preserves the objective function values of the corresponding solutions (of the allocation problem and the independent set problem). It is known that the independent set problem is hard even from an approximation point of view:

Fact 6.9. For every fixed $\epsilon > 0$, there is no $O(n^{1-\epsilon})$-approximation algorithm for the independent set problem, where n denotes the number of vertices in the graph (unless $NP \subseteq ZPP$).

[1] Recall that an independent set $I \subseteq V$ of G is a subset of the vertices such that no two vertices in I are connected by an edge.

Collect the bids $\{(S_i, b_i)_{i \in N}\}$ of all players.
Reindex the bids such that

$$\frac{b_1}{\sqrt{|S_1|}} \geq \frac{b_2}{\sqrt{|S_2|}} \geq \cdots \geq \frac{b_n}{\sqrt{|S_n|}}.$$

$W \leftarrow \emptyset$
for $i = 1, \ldots, n$ **do**
  if no items in S_i have been assigned to players in W, i.e., $S_i \cap \left(\bigcup_{j \in W} S_j\right) = \emptyset$ **then**
    add i to W: $W \leftarrow W \cup \{i\}$.
  end if
end for
for each $i \in W$ **do**
  define i's payment as:

$$p_i := \frac{b_j}{\sqrt{|S_j|}} \cdot \sqrt{|S_i|},$$

  where $j > i$ is the smallest index such that $S_i \cap S_j \neq \emptyset$ and for all $k < j$, $k \neq i$, $S_k \cap S_j = \emptyset$; if no such j exists, set $p_i := 0$.
end for
return (W, p)

Algorithm 6.4.1: Greedy mechanism for single-minded bidders.

Since the number of edges in a (simple) directed graph is at most $O(n^2)$, we obtain the following corollary:

Corollary 6.10. For every fixed $\epsilon > 0$, there is no $O(m^{1/2-\epsilon})$-efficient mechanism for single-minded bidders, where m denotes the number of items (unless NP $\subseteq$ ZPP).

We next devise a mechanism that is strategyproof and $\sqrt{m}$-approximate efficient. Thus, from a computational point of view, this is the best we can hope for. The mechanism is the greedy algorithm described in Algorithm 6.4.1.

Theorem 6.11. The greedy mechanism is strategyproof, $\sqrt{m}$-efficient and runs in polynomial-time.

The following lemma characterizes properties of strategyproof mechanisms for single-minded bidders.

Lemma 6.12. A mechanism for single-minded bidders is strategyproof if it satisfies the following two conditions:

1. Monotonicity: A bidder who wins with bid (S_i, b_i) keeps winning for any $b_i' > b_i$ and for any $S_i' \subset S_i$ (for any fixed bids of the other players).
2. Critical value: A bidder who wins pays the minimum value, also called the critical value, needed for winning, i.e., the payment is the minimum over all values b_i' such that (S_i, b_i') still wins.

Before proving the lemma, lets verify that the greedy mechanism satisfies these two properties and is thus strategyproof. It is easy to see that the greedy mechanism satisfies monotonicity: Suppose (S_i, b_i) is a winning bid. If player i increases his bid or submits a subset $S_i' \subset S_i$, he can only move further to the front of the greedy ordering. Since S_i is disjoint from all sets S_j of previously picked players $j \in W$, i remains a winner. Next consider the critical value property. The critical value of a winning bid (S_i, b_i) corresponds to the bid b_i' for which (S_i, b_i') still wins. Consider Step 10 of Algorithm 6.4.1. (S_i, b_i') remains a winning bid as long as $b_i' \geq p_i$, since if $b_i' < p_i$ player j precedes i in the greedy order and thus j wins and prevents i to enter the winning set W. Thus the payment p_i corresponds to the critical value.

Proof. We first observe that a truthful bidder will never receive a negative utility: His utility is zero when he loses. In order to win, his valuation must be at least the critical value, which is exactly his payment.

We next show that a bidder i can never improve his utility by reporting some bid $(S_i, b_i) \neq (\Sigma_i, \theta_i)$. If (S_i, b_i) is a losing bid (zero utility), or if S_i does not contain Σ_i (non-positive utility), then clearly reporting (Σ_i, θ_i) can only help.

Assume that (S_i, b_i) is a winning bid and $S_i \supseteq \Sigma_i$. We first show that i is never worse off by reporting (Σ_i, b_i) instead of (S_i, b_i). Let p_i be the payment for (S_i, b_i) and let p_i' be the payment for (Σ_i, b_i). Note that (Σ_i, b_i) is a winning bid by monotonicity. For every $x < p_i'$, bidding (Σ_i, x) will lose since p_i' is a critical value. By monotonicity, (S_i, x) will also be a losing bid for every $x < p_i'$ and therefore the critical value p_i is at least p_i'. It follows that by bidding (Σ_i, b_i) instead of (S_i, b_i), player i still wins and his payment does not increase.

Next, we show that player i is not worse off by bidding (Σ_i, θ_i) instead of bidding the winning bid (Σ_i, b_i). First suppose that (Σ_i, θ_i) is a winning bid with payment (critical value) $\bar{p}_i$. As long as $b_i \geq \bar{p}_i$, player i still wins by bidding (Σ_i, b_i) (by monotonicity) and receives the same payment (by critical value). If $b_i < \bar{p}_i$, player i loses and receives zero utility. In both cases, misreporting does not increase the utility of player i. Finally, suppose that player i loses by bidding (Σ_i, θ_i). Then θ_i must be smaller than the critical value and thus the payment for the winning bid (Σ_i, b_i) will be greater than θ_i. Therefore, the utility that player i receives by bidding (Σ_i, b_i) is negative. □

We next show that the greedy algorithm computes an outcome that is $\sqrt{m}$-efficient.

> **Lemma 6.13.** Assume that $(S_i, b_i) = (\Sigma_i, \theta_i)$ for every player $i \in N$. Let W^* be the winner set of an optimal allocation OPT with maximum social welfare value opt $:= \sum_{i \in W^*} \theta_i$. Let W be the winner set output by the greedy mechanism. Then $\sum_{i \in W} \theta_i \geq \frac{1}{\sqrt{m}}$opt, where m refers to the number of items in M.

Proof. Let W^* be the winner set of OPT. For each $i \in W$, let $W_i^* := \{j \in W^*, j \geq i : S_i \cap S_j \neq \emptyset\}$ be the set of winners in OPT that did not enter W because of i. (Observe that $i \in W \cap W^*$ implies that $i \in W_i^*$.) Note that every $j \in W^*$ must occur in some set W_i^* for some $i \in W$: either $j \in W$ and thus $j \in W_j^*$ or $j \notin W$ and thus there must exist some $i < j$ with $i \in W$ and $S_i \cap S_j \neq \emptyset$ which implies that $j \in W_i^*$. Thus, $W^* \subseteq \cup_{i \in W} W_i^*$.

Therefore, if we can show that for every $i \in W$

$$\sum_{j \in W_i^*} \theta_j \leq \sqrt{m}\theta_i, \tag{6.3}$$

the claim follows since then

$$\text{opt} = \sum_{i \in W^*} \theta_i \leq \sum_{i \in W} \sum_{j \in W_i^*} \theta_j \leq \sum_{i \in W} \sqrt{m}\theta_i = \sqrt{m} \sum_{i \in W} \theta_i.$$

Note that every $j \in W_i^*$ appears after i (or is i itself) in the greedy order and thus we must have $\theta_j \leq \frac{\theta_i}{\sqrt{|S_i|}} \sqrt{|S_j|}$. Summing over all $j \in W_i^*$, we obtain

$$\sum_{j \in W_i^*} \theta_j \leq \frac{\theta_i}{|\sqrt{S_i}|} \sum_{j \in W_i^*} \sqrt{|S_j|}. \tag{6.4}$$

Using the Cauchy-Schwarz inequality (see below), we can bound

$$\sum_{j \in W_i^*} \sqrt{|S_j|} \leq \sqrt{|W_i^*|} \sqrt{\sum_{j \in W_i^*} |S_j|}. \tag{6.5}$$

By definition, every S_j for $j \in W_i^*$ intersects S_i. Since OPT is an allocation, these intersections must all be disjoint and thus $|W_i^*| \leq |S_i|$. Moreover, since OPT is an allocation, $\sum_{j \in W_i^*} |S_j| \leq m$. We thus get

$$\sqrt{|W_i^*|} \sqrt{\sum_{j \in W_i^*} |S_j|} \leq \sqrt{|W_i^*|}\sqrt{m} \leq \sqrt{|S_i|}\sqrt{m}.$$

Plugging this together with (6.5) into (6.4) proves (6.3). □

Lemma 6.14. Let $x, y \in \mathbb{R}^n$. Then

$$\left(\sum_{i \in [n]} x_i \cdot y_i \right)^2 \leq \left(\sum_{i \in [n]} x_i^2 \right) \cdot \left(\sum_{i \in [n]} y_i^2 \right).$$

6.5 Sponsored Search Auction

A web search results in a list of results using a search algorithm and a list of sponsored links that are paid by ad customers.

I f one googles the term “Damenschuhe”, then on top there is an ordered list of companies selling “Damenschuhe”.

Whenever a search is executed, an online auction decides on the order of ads. Some surveys suggest that the bulk of Googles earnings result from these ads.

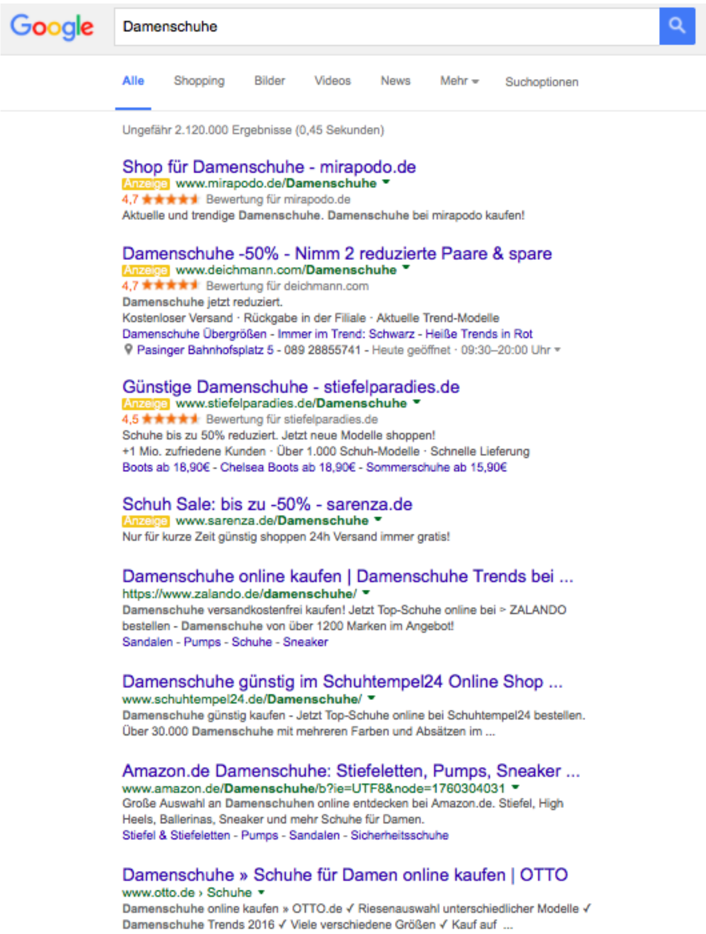


Figure 6.4: Google Suche

Model.

We discuss a model for sponsored search auctions.
There are k slots for the positions of ad slots. The bidders bid for a certain search word like "Damenschuhe". For example, "Mirapodo" and "Deichmann" are the customers bidding for the term "Damenschuhe".

- there are several goods ($k > 1$) and
- goods are not identical: higher listed slots are more valuable.

We quantify the value of different slots by the click-through-rate (CTR): CTR $\alpha_j \in [0, 1]$ denotes the probability that a user j clicks for slot $j = 1, \dots, k$. We order slots in decreasing order

$$\alpha_1 \geq \alpha_2 \geq \dots \alpha_k.$$

For ease of presentation we assume that the α values are independent of the actual content of the link. This can be avoided by imposing quality-scores β_i and use instead $\alpha_i \beta_i$. In the following we stick to the simple α model. Every customer i has a private value v_i for every click on a link. Hence, the utility for i for slot j is $v_i \alpha_j$. We want to achieve the following goals:

1. Strategyproofness.
2. Efficiency: Allocation should maximize $\sum_{i=1}^{n} v_i x_i$, where x_i is the CTR of the slot of bidder i bezeichnet.
3. Polynomial implementation of the auction.

Design in two steps.

Step 1: Suppose bidders bid truthfully. How can we guarantee (2) and (3)?
Step1: For$j = 1, 2, \dots, k$, assign the j-th highest bid to the j-th highest slot. Allocation is obviously efficient and can be implemented in polynomial time. Step 2 follows by using Myersons Lemma to compute payments so that (1) can be guaranteed.
Step 2:
We need the following definition.

> **Definition 6.15** (One-Parameter Environment). In a One-Parameter Environment there are n bidders and every bidder i has a private value $v_i \in \mathbb{R}$ per unit $x_i \geq 0$ assigned. There is a feasible set of assignments $X \subseteq \mathbb{R}_+^n$, where for $x = (x_1, \dots, x_n) \in X$ the entry x_i represents the amount assigned to i.

Consider the following examples.

Example 6.16.

- In a single item auction, X is the set of $\{0, 1\}$-vectors where at most one entry is equal 1, i.e., $X = \{x \in \{0, 1\}^n | \sum_{i=1}^{n} x_i \leq 1\}$.
- In an auction with k identical items so that every bidder gets at most one item, the set is $X = \{x \in \{0, 1\}^n | \sum_{i=1}^{n} x_i \leq k\}$.
- In sponsored search, the set X models the assignment of bidders to slots, where every slot is assigned to at most one bidder and vice versa.We set

$$x_i = \begin{cases} \alpha_j, & \text{if slot } j \text{ is assigned to bidder } i, \\ 0, & \text{else.} \end{cases}$$

Allocation and Payments.

We have the following three steps for a mechanism:

1. Collect bids $b = (b_1, \dots, b_n)$.
2. Compute allocation $x(b) \in X$ (Allocation).
3. Compute payments $p(b) \in \mathbb{R}^n$ (Payment).

Bidders have quasi-linear utilities:

$$u_i(b) = v_i x_i(b) - p_i(b).$$

We consider payments that satisfy:

$$p_i(b) \in [0, b_i x_i(b)] \forall i \in N, \forall b \in \mathbb{R}^n.$$

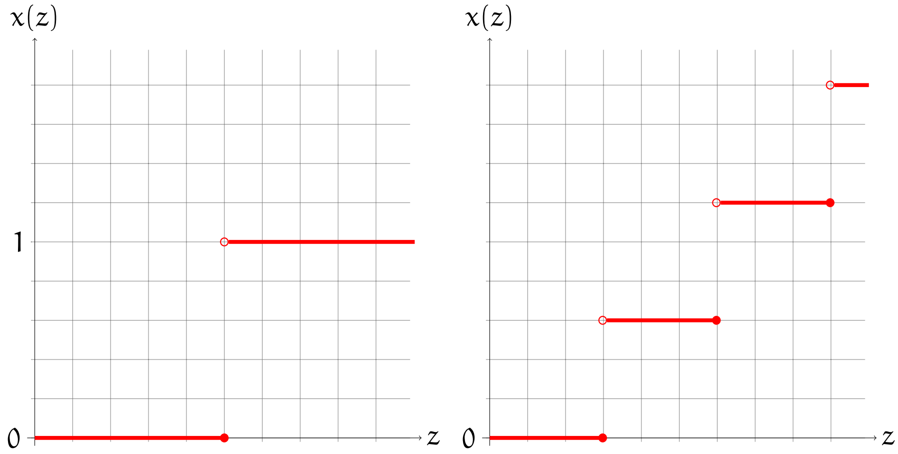


Figure 6.5: Piece-wise monotone allocation function.

$p_i(b) \geq 0$ avoids that the bidder looses money. $p_i(b) \leq b_i x_i(b)$ enforces truthful bidding and ensures that every bidder receives nonnegative utility.

Definition 6.17. An allocation rule x (see (2)) is called implementable if there is a payment rule (see (3)) such that the mechanism defined via (1)-(3) is sp .

Definition 6.18. An allocation rule x for a One-Parameter Environment is monoton, if for every bidder i and bids b_{-i} the following inequality holds:

$$x_i(z, b{-}i) \leq x_i(z', b_{-i}) \forall z, z' \in \mathbb{R} \text{ with } z \leq z'.$$

In other words: a higher bid results in a higher allocation quantity.

Theorem 6.19 (Myersons Lemma '81). For a One-Parameter Environment, the following properties hold true:

(a) A (differentiable) allocation rule x is implementable iff the rule is monotone.

(b) If x is implementable, then there is a unique payment rule so existiert sp that the mechanism defined via (1)-(3) is sp (under the condition that $b_i = 0$ implies $p_i(b) = 0$).

(c) The payment rule (b) is defined through (6.10).

Proof. **Part 1:** implementable $\Rightarrow$ monotone. Consider an arbitrary (possibly non-monotone) allocation rule. Suppose that the mechanism defined through (x, p) is implementable. We will derive conditions on p and x . sp implies that for every i every possible private value b_i and every b_{-i}, the bid $b_i = v_i$ maximizes the utility for i. We choose arbitrary b_{-i}. In order to simplify the presentation we write for $x_i(z, b_{-i})$ shorthand $x(z)$ and $p(z)$ for $p_i(z, b_{-i})$ of bidder i, if i bids the value z.

Assume (x, p) is implementable. Recall:

$$v_i x(v_i, b_{-i}) - p_i(v_i, b_{-i}) \geq v_i x(b_i, b_{-i}) - p_i(b_i, b_{-i}) \quad \forall b_{-i},\ \forall b_i. \tag{6.6}$$

Consider two arbitrary bids y and z with $0 \leq z \leq y$. Bidder i can have a private value of z and lie with the bid y. Condition (6.6) yields

$$z \cdot x(z) - p(z) \geq z \cdot x(y) - p(y). \tag{6.7}$$

On the other hand, i can have private value y and bids with the wrong bid z:

$$y \cdot x(y) - p(y) \geq y \cdot x(z) - p(z). \tag{6.8}$$

Combining (6.7) and (6.8) yields an upper bound on $p(y) - p(z)$:

$$z \cdot [x(y) - x(z)] \leq p(y) - p(z) \leq y \cdot [x(y) - x(z)]. \tag{6.9}$$

Inequality (6.9) gives that $x(y) - x(z)$ is nonnegative for $z \leq y$. Hence, x is monotone.
Part 2: Uniqueness of payments. Suppose x is sp . To show statements (b) and (c) consider (6.9). Fix z and let $y \rightarrow_{y \geq z} z$.
Suppose x is differentiable. Divide the difference inequality by $y - z$ and take the limit $y \rightarrow z$:

$$p'(z) = z \cdot x'(z),$$

and with $p(0) = 0$, we get the payment rule

$$p_i(b_i, b_{-i}) = \int_0^{b_i} z \cdot \frac{dx_i(z, b_{-i})}{dz} dz \tag{6.10}$$

for every i and all b_i, b_{-i}. It follows that (6.10) is the unique payment rule that gives an sp mechanism
Part 3: x monotone $\Rightarrow$ mechanism (x, p) is sp. Suppose that x is piece-wise constant and that the payment rule is given by (6.10). We show with Fig.6.6 graphically, that truthfull bidding is a dominant strategy.
The red curve is the allocation function and the blue area is the value times quantity (top), payment (middle) and utility (bottom), where the utility is given as $v \cdot x(b) - p(b)$. □

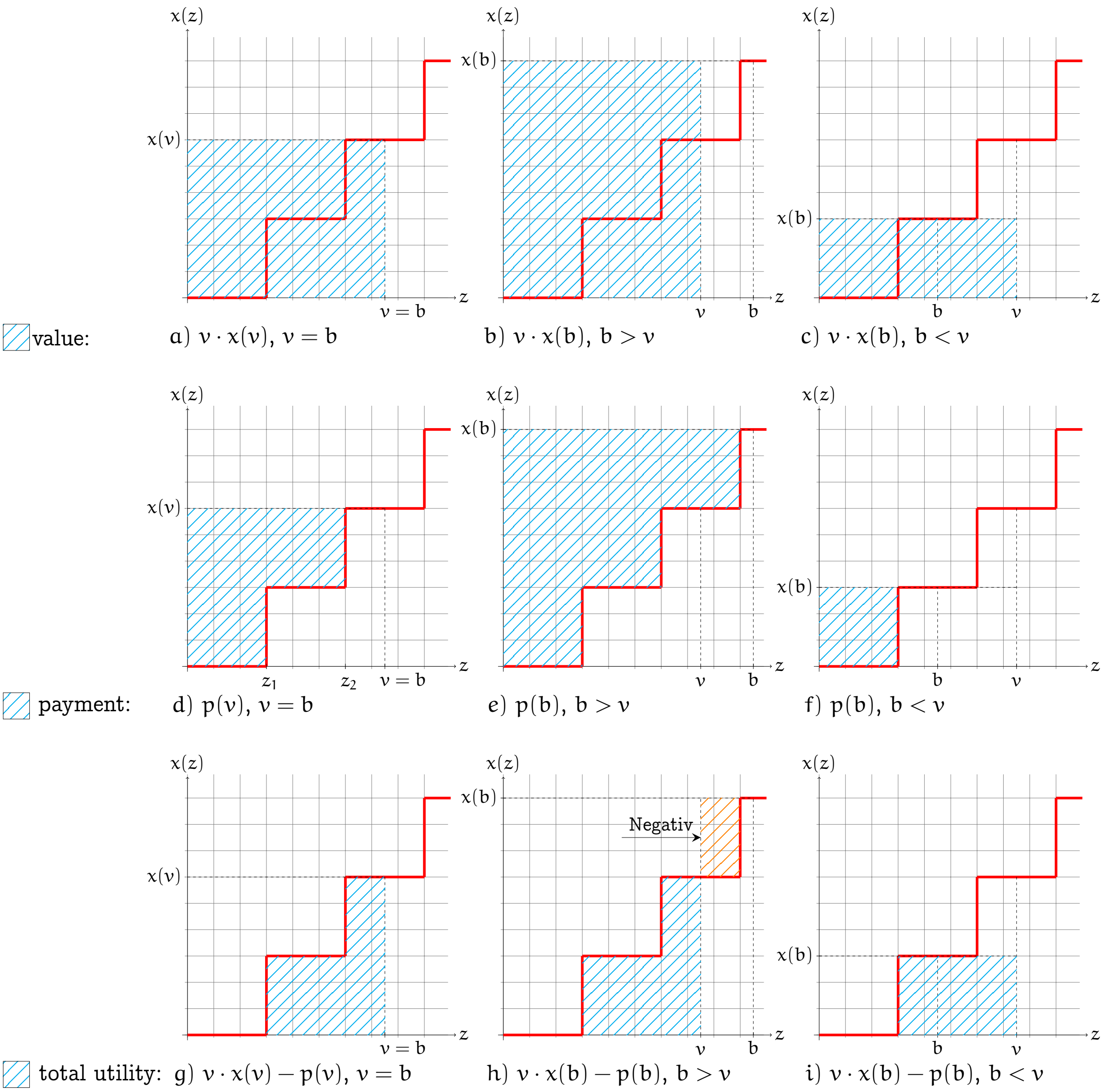


Figure 6.6: Proof via illustration.

Chapter 7

Cooperative Game Theory

Games where players are allowed to cooperate are also called cooperative games, as opposed to non-cooperative games which we have considered so far.

Definition 7.1. A cooperative game with transferable utilities consists of:

- a finite set N of players;
- $v : 2^N \to \mathbb{R}^+$, devising for every $S \subseteq N$ a total utility $v(S)$.

Definition 7.2. Let (N, v) be a cooperative game. An imputation $(x_i)_{i \in S}$ for $S \subseteq N$, defines for every $i \in S$ a nonnegative payoff $x_i \geq 0$ such that

$$\sum_{i \in S} x_i = v(S).$$

We assume that v staisfies the following property: For every partition $\{P_1, \ldots, P_k\}$ of N :

$$v(N) \geq \sum_{j=1}^{k} v(P_j).$$

How can we divide the value $v(N)$ fairly among N.

7.1 Core

An imputation $(x_i)_{i \in N}$ belongs to the core of (N, v), if no coalition has an incentive to deviate from the grand coalition N.

Definition 7.3. The core of (N, v) is the set of all imputations $(x_i)_{i \in N}$, such that

$$\sum_{i \in S} x_i \geq v(S) \quad \forall S \subseteq N.$$

(Equivalently: There is no coalition $S \subseteq N$ and imputation $(y_i)_{i \in S}$, such that $y_i(S) > x_i(S)$ for all $i \in S$.)

Example 7.4 (Treasure Hunt). An expedition of n people discover a treasure on a mountain. A piece of the treasure can only be carried down by two people. This gives rise to the following game: (N, v) with $N := [n]$ and

$$v(S) := \left\lfloor \frac{|S|}{2} \right\rfloor \quad \forall S \subseteq N.$$

Suppose $|N| \geq 2$ is even: The imputation $(x_i)_{i \in N} = (1/2, \dots, 1/2)$ belongs to the core of (N, v), because for all $S \subseteq N$:

$$\sum_{i \in S} x_i = \frac{|S|}{2} \geq \left\lfloor \frac{|S|}{2} \right\rfloor.$$

Assume$|N| \geq 3$ odd: The core is empty! Assume, $(x_i)_{i \in N}$ is in the core. We have:

$$\sum_{i \in N} x_i = v(N) = \left\lfloor \frac{|N|}{2} \right\rfloor = \frac{|N| - 1}{2}.$$

Hence, there is a player $j \in N$ with $x_j \geq (|N| - 1)/(2|N|)$. We get:

$$\frac{|N| - 1}{2} = v(N \setminus \{j\}) \leq \sum_{i \in N \setminus \{j\}} x_i = v(N) - x_j \leq \frac{|N| - 1}{2} - \frac{|N| - 1}{2|N|}$$

where the first inequality is implied by the definition of the core. This is a contradiction for $|N| > 1$.

When is the core empty/nonempty?
Let $\mathcal{C} := 2^N$ be the set of all subsets of N.

Definition 7.5. A vector $(\delta_S)_{S \in \mathcal{C}}$ with $\delta_S \in [0, 1]$ is called balanced if for all $i \in N$:

$$\sum_{S \in \mathcal{C} : i \in S} \delta_S = 1.$$

A game (N, v) is balanced, if for every balanced $(\delta_S)_{S \in \mathcal{C}}$:

$$\sum_{S \in \mathcal{C}} \delta_S v(S) \leq v(N).$$

Intuitively, we can interpret a balanced vector as follows: Every player $i \in N$ has a budget of 1 that he can distribute among all subsets $S \in \mathcal{C}$ with $i \in S$. The "average worth" of a coalition is then represented by $\delta_S v(S)$.

Theorem 7.6 (Bondareva–Shapley). A game (N, v) (with transferable utilities) has a nonempty core iff (N, v) is balanced.

Proof. Consider the following linear program:

$$\min \sum_{i \in N} x_i \quad \text{s.t.} \quad \sum_{i \in S} x_i \geq v(S) \quad \forall S \subseteq N \tag{LP}$$

and its dual:

$$\max \sum_{S \in \mathcal{C}} \delta_S v(S) \quad \text{s.t.} \quad \sum_{S \in \mathcal{C}: i \in S} \delta_S = 1 \;\; \forall i \in N, \; \delta_S \geq 0 \;\; \forall S \in \mathcal{C}. \tag{D}$$

The LP satisfies all conditions of the core except $\sum_{i \in N} x_i \leq v(N)$; Note that also $x_i \geq 0$ is satisfied as $v(\{i\}) \geq 0$ for all $i \in N$. The LP is therefore bounded. For (D), every feasible solution corresponds to a balanced vector δ satisfying $\delta_S \in [0, 1]$ for all $S \in \mathcal{C}$.
Suppose the core is nonempty. Then, there is an imputation $(x_i)_{i \in N}$ which is a feasible solution for (LP). Using weak duality, we obtain $\sum_{i \in N} x_i \geq \max \sum_{S \in \mathcal{C}} \delta_S v(S)$ for every feasible solution $(\delta_S)_{S \in \mathcal{C}}$ of (D). Hence, (N, v) is balanced.
Suppose (N, v) is balanced. Let $(\delta_S^*)_{S \in \mathcal{C}}$ be a balanced vector solving (D). Let $(x_i^*)_{i \in N}$ be an optimal solution to (LP). (Note that both solutions exist!). Using strong duality:

$$\sum_{i \in N} x_i^* = \sum_{S \in \mathcal{C}} \delta_S^* v(S) \leq v(N),$$

where the last inequality follows as (N, v) is balanced. Since $(x_i^*)_{i \in N}$ is a feasible solution to (LP), we get that $(x_i^*)_{i \in N}$ belongs to the core of (N, v). □

7.2 Cost Sharing with the Shapley-Value

As the core for several cooperative games may be empty, the search for alternative cost sharing methods is relevant. We introduce now the Shapley-Value and it will turn out that these cost-shares exist and can be axiomatically characterized. In this section, we consider for a coalition $S \subseteq N$ the term costs instead of payoffs.
We are given:

- Set of players $N = [n]$;
- Cost function $c : 2^N \to \mathbb{R}^+$, that gives for every coalition $S \subseteq N$ its cost $c(S)$; we assume that c is monoton, i.e., $c(S) \leq c(S')$ for all $S \subseteq S'$, and $c(\emptyset) = 0$.

We want to determine a cost allocation vector $(x_i)_{i \in N}$ with

$$\sum_{i \in N} x_i = c(N).$$

We define the marginal costs of a player: Intuitively, the marginal costs measure the additional costs when adding a player to a coalition S.

Definition 7.7. The marginal cost $\Delta_i(S)$ of $i \in N$ w.r.t. $S \subset N$ with $i \notin S$ is defined as

$$\Delta_i(S) := c(S \cup \{i\}) - c(S).$$

- We call $i \in N$ Dummy, if her marginal cost is independent of S: $\Delta_i(S) = c(\{i\})$ for all $S \subset N$, $i \notin S$.
- Two players $i, j \in N$ are exchangeable, if their marginal cost w.r.t. $S \subset N$ are equal: $\Delta_i(S) = \Delta_j(S)$ for all $S \subset N$, $i, j \notin S$.

We require that cost sharing methods satisfy the following axioms:

Axiom 7.8. If $i \in N$ is a dummy-player, then $x_i = c(\{i\})$.

Axiom 7.9. If i and j are exchangeable, then $x_i = x_j$.

Axiom 7.10. Let $c(S) := c_1(S)+c_2(S)$ for all $S \subseteq N$, where c_1 and c_2 are monotone cost functions. If $(x_i^1)_{i\in N}$ and $(x_i^2)_{i\in N}$ are cost share vectors for c_1 or c_2, so are also $(x_i)_{i\in N}$ with

$$x_i := x_i^1 + x_i^2 \quad \forall i \in N$$

a cost share vector for c.

Assume we add the players according to given order one after the other to S. The marginal cost of player i when added according to the given order are called ordered marginal costs:

Definition 7.11. Let σ be a permutation of N. $(x^{\sigma}_{\sigma_i})_{i\in N}$ are called ordered marginal costs, if

$$\begin{aligned}
x^{\sigma}_{\sigma_1} &= c(\{\sigma_1\}) \\
x^{\sigma}_{\sigma_2} &= c(\{\sigma_1, \sigma_2\}) - c(\{\sigma_1\}) \\
&\vdots \\
x^{\sigma}_{\sigma_i} &= c(\{\sigma_1, \dots, \sigma_i\}) - c(\{\sigma_1, \dots, \sigma_{i-1}\}) = c(\{\sigma_1, \dots, \sigma_i\}) - \sum_{\ell=1}^{i-1} x^{\sigma}_{\sigma_\ell}
\end{aligned}$$

Let us derive some properties of these costs:

1. We have $x^{\sigma}_{\sigma_i} \geq 0$ for all $i \in N$, as c is monotone.
2. With the above definition, we get $\sum_{i\in N} x^{\sigma}_{\sigma_i} = c(N)$.
3. Consider a dummy-player σ_i. We have: $c(S \cup \{\sigma_i\}) - c(S) = c(\{\sigma_i\})$ for all $S \subset N$, $\sigma_i \notin S$. Hence: $x^{\sigma}_{\sigma_i} = c(\{\sigma_i\})$. Axiom 7.8 is thus satisfied.
4. Let c_1 and c_2 be cost functions with $c_1(S) + c_2(S) = c(S)$ for all $S \subseteq N$. Assume $(x^1_{\sigma_i})_{i\in N}$ and $(x^2_{\sigma_i})_{i\in N}$ are the ordered marginal cost vectors w.r.t. a permutation σ. Then, we have:

$$\begin{aligned}
x^{\sigma}_{\sigma_i} =& c(\{\sigma_1, \dots, \sigma_i\}) - c(\{\sigma_1, \dots, \sigma_{i-1}\}) \\
=& c_1(\{\sigma_1, \dots, \sigma_i\}) - c_1(\{\sigma_1, \dots, \sigma_{i-1}\}) + c_2(\{\sigma_1, \dots, \sigma_i\}) - c_2(\{\sigma_1, \dots, \sigma_{i-1}\}) \\
=& x^1_{\sigma_i} + x^2_{\sigma_i}.
\end{aligned}$$

   Axiom 7.10 is thus also satisfied.
5. The ordered marginal costs do not satisfy axiom 7.9; as the following example shows (see Fig.7.1). Suppose we have n players, that all want to send one unit of flow to a common root vertex r. The installation of capacity n on this edge costs n; no matter how many players are actually using the edge. The cost function has the following form: $c(S) = n$ for all $\emptyset \neq S \subseteq N$ and $c(\emptyset) = 0$. The ordered marginal cost of the first player is $\sigma_1$and hence n, whereas all other players pay 0. Note that $\Delta_i(S) = \Delta_j(S)$ for all $i, j \in N$ and $S \subset N$, $i, j \notin S$; i.e., all players are exchangeable.

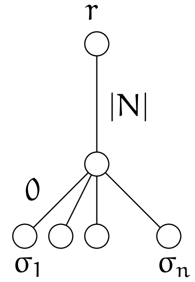


Figure 7.1: Example of a network creation game.

In order to avoid the problem just mentioned, we will use an averaging argument. Let Π be the set of the $n!$ permutations of the set N. For a permutation $\sigma \in \Pi$, let S_i^σ be the set of player that appear before i in σ:

$$S_i^\sigma := \{\sigma_j \ : \ j < k \ \text{ where } \sigma_k = i\}.$$

Definition 7.12. Define the Shapley-value of $i \in N$ as

$$\varphi_i := \frac{1}{n!} \sum_{\sigma \in \Pi} \Delta_i(S_i^\sigma).$$

The intuition is as follows: We assume that all permutations in Π are equally likely. The Shapley-value i corresponds to the expected marginal costs over all permutations. Formally:

$$\varphi_i = \mathbf{E}[x^\sigma_{\sigma_k = i}],$$

where the expectation is taken over all permutations in Π.
We can alternatively express the Shapley-value for i as a sum over all $S \subseteq N$, $i \notin S$: The number of permutations for which S_i^σ equals a fixed set S is $|S|! \cdot (n - |S| - 1)!$. Hence, we get:

$$\varphi_i := \sum_{S \subseteq N, i \notin S} \frac{|S|! \cdot (n - |S| - 1)!}{n!} \Delta_i(S).$$

In the above example, every $i \in N$ would get a cost share of $x_i = (n-1)!/n! \cdot n = 1$.

Theorem 7.13. The Shapley-value $(\varphi_i)_{i \in N}$ satisfies axiomns 7.8–7.10. Moreover, it is the unique cost allocation vector, that satifies axioms 7.8–7.10.

Proof. It follows easily that axiom 7.8 and axiom 7.10 are satisfied (see Properties 1–4 of the ordered marginal costs).
Axiom 7.9. Consider axiom 7.9. We have: For any two exchangeable players $i, j \in N$ it holds $x_i = x_j$; formally:

$$\Delta_i(S) = \Delta_j(S) \quad \forall S \subset N, \ i, j \notin S \quad \Rightarrow \quad x_i = x_j. \tag{7.1}$$

We denote for every permutation $\sigma \in \Pi$ the permutation that switches positions of i and j in σ as $\bar{\sigma} \in \Pi$.

Case 1: i appears before j in σ. We get: $S_i^\sigma = S_j^{\bar\sigma}$ and thus: $\Delta_i(S_i^\sigma) = \Delta_j(S_j^{\bar\sigma})$.
Case 2: j appears before i in σ. Define $S := S_i^\sigma \setminus \{j\}$. With this definition we get:

$$S_i^\sigma = S \cup \{j\} \quad \text{and} \quad S_j^{\bar\sigma} = S \cup \{i\}.$$

We get:

$$\begin{aligned}
\Delta_i(S_i^\sigma) - \Delta_j(S_j^{\bar\sigma}) &= c(S_i^\sigma \cup \{i\}) - c(S_i^\sigma) - c(S_j^{\bar\sigma} \cup \{j\}) + c(S_j^{\bar\sigma}) \\
&= c(S \cup \{i,j\}) - c(S \cup \{j\}) - c(S \cup \{i,j\}) + c(S \cup \{i\}) \\
&= c(S \cup \{i\}) - c(S \cup \{j\}) \\
&= c(S \cup \{i\}) - c(S) - (c(S \cup \{j\}) - c(S)) \\
&= \Delta_i(S) - \Delta_j(S) = 0,
\end{aligned}$$

where the last equality follows from (7.1).
In both cases, we get: $\Delta_i(S_i^\sigma) = \Delta_j(S_j^{\bar\sigma})$. Hence:

$$x_i = \frac{1}{n!}\sum_{\sigma\in\Pi} \Delta_i(S_i^\sigma) = \frac{1}{n!}\sum_{\sigma\in\Pi} \Delta_j(S_j^{\bar\sigma}) = \frac{1}{n!}\sum_{\sigma\in\Pi} \Delta_j(S_j^\sigma) = x_j.$$

Uniqueness.
We show that the Shapley-value is the unique cost allocation vector that satisfies axioms 7.8–7.10.
Define cost function c_T with $T \subseteq N$ as follows:

$$c_T(S) := \begin{cases} 1 & \text{if } T \subseteq S, \\ 0 & \text{else.} \end{cases} \tag{7.2}$$

We can represent c for (N, c) as a $(2^n - 1)$-dimensional vector $(c(S))_{S\subseteq N}$, where $n = |N|$.

Proposition 7.14. For every such c, there are unique scalars $(\alpha_T)_{T\subseteq N} \in \mathbb{R}^{2^n-1}$, such that

$$c(S) = \sum_{T\subseteq N} \alpha_T c_T(S) \quad \forall S \subseteq N.$$

Proof. We show that $(c_T)_{T\subseteq N}$ constitutes a basis of the $(2^n - 1)$-dimensional space. We need to show that the vectors $(c_T)_{T\subseteq N}$ are linearly independent: Let

$$\sum_{T\subseteq N} \beta_T c_T = 0.$$

We need to show: $\beta_T = 0$ for all $T \subseteq N$.
Suppose there is $Q \subseteq N$ with $\beta_Q \neq 0$. We choose Q such that for all $Q' \subset Q$ we have: $\beta_{Q'} = 0$. It follows that:

$$\sum_{T\subseteq N} \beta_T c_T(Q) = \beta_Q \neq 0,$$

where we used $\beta_{Q'} = 0$ for all $Q' \subset Q$ and the fact that $c_T(Q) = 0$ for all $T \not\subseteq Q$. This is a contradiction. □

Proposition 7.15. Let $T \subseteq N$. The vector $(x_i)_{i \in T}$ w.r.t. αc_T is uniquely determined via axioms 7.8–7.9.

Proof. Consider the marginal costs of player $i \in N$ w.r.t. αc_T.
Case 1: $i \notin T$. We get: $\Delta_i(S) = 0$ for all $S \subseteq N$, $i \notin S$. Player i is hence a dummy-player and with axiom 7.8 we get: $x_i = 0$.
Case 2: $i \in T$. It follows easily that for all $S \subseteq N$, $i \notin S$: $\Delta_i(S) = 1$, if $S \cap T = T \setminus \{i\}$ and $\Delta_i(S) = 0$, else. Consider two players $i, j \in T$ and a set $S \subseteq N$, $i, j \notin S$. We have: $\Delta_i(S) = 0 = \Delta_j(S)$. This means that every i and j in T are exchangeable and with axiom 7.9 we get: $x_i = x_j$.
As $(x_i)_{i \in N}$ is a cost allocation vector for $(N, \alpha c_T)$we get:

$$\sum_{i \in N} x_i = \alpha c_T(N) = \alpha.$$

This shows together with the above considerations that $(x_i)_{i \in N}$ is uniquely determined as:

$$x_i := \begin{cases} 0, & \text{if } i \notin T, \\ \frac{\alpha}{|T|} & \text{else.} \end{cases}$$

We note that $(x_i)_{i \in N}$ is equal to the Shapley-value for αc_T. □

Let $(y_i)_{i \in N}$ be a cost allocation vector for c, that satisfies axioms 7.8–7.10. With Proposition 7.14, c has a unique representation as

$$c(S) = \sum_{T \subseteq N} \alpha_T c_T(S)$$

Proposition 7.15 shows, that the cost allocation vector for every cost function $\alpha_T c_T$ is uniquely determined via the Shapley-value. As $(y_i)_{i \in N}$ satisfies axiom 7.10 we get that $(y_i)_{i \in N}$ is uniquely determined. □

7.3 Group-Strategyproofness in Cooperative Cost Sharing Games

We consider the problem of designing truthful mechanisms for cooperative cost sharing games arising from combinatorial optimization problems.

- player set $N = \{1, \ldots, n\}$ want to use a common service
- service cost $C : 2^N \to \mathbb{R}_+$
- C is given implicitly via solution of an optimization problem $\mathcal{P}$
- $v_i \geq 0$ private value of i for serice
- $b_i \geq 0$ bid of i.
- input of $M(b)$: $b := (b_i)_{i \in N}$
- output of $M(b)$: allocation $x := (x_i)_{i \in N}$ and payment $p := (p_i)_{i \in N}$.
- Q winning set x, i.e., $i \in Q \Leftrightarrow x_i = 1$, i.e., Q receives service.

- $M(b)$ is budget balanced, i.e.,

$$C(Q) = \sum_{i \in Q} p_i.$$

- the goal of player i is to maximize his own (quasi-linear) utility function $u_i(x, p) := v_i x_i - p_i$.

Our mechanisms should adhere to the following economic constraints:

1. No positive transfer: no player is payed for receiving service, i.e., $p_i \geq 0$.
2. Individual rationality: a player is charged only if he receives service and the charged price does not exceed his bid.
3. Consumer sovereignty: a player is guaranteed to receive service if only his bid is high enough.

We assume that players act strategically and . Since the outcome (x, p) computed by the mechanism M solely depends on the bids b of the players, a player may have an incentive to declare a bid b_i that differs from his valuation v_i. We say that M is strategyproof if bidding truthfully, i.e., $b_i = v_i$, is a dominant strategy for every player i.
Here we are interested in what happens if players may form coalitions. A coalition of players simply corresponds to a subset $S \subseteq N$ of the players. We assume that players may form arbitrary coalitions among them (i.e., every subset $S \subseteq N$ corresponds to a possible coalition). Players that form a coalition S may coordinate their bids. That is, they may agree on a bidding profile $(b_i)_{i \in S}$ and thus collectively attempt to manipulate the mechanism reporting these bids.
We next define a very strong truthfulness notion for coalitional games. In words, a mechanism is called group-strategyproof if no coordinated bidding of a coalition $S \subseteq N$ can ever strictly increase the utility of some player in S without strictly decreasing the utility of another player in S. We define this formally:

Definition 7.16. Let S be an arbitrary coalition of players and b, b' two bid vectors satisfying $b_i = b'_i$ for all $i \notin S$ and $b_i = v_i$ for every $i \in S$ (that is, players in S bid truthfully in b and arbitrarily in b'). Let (x, p) and (x', p') be the respective outcomes computed by the mechanism M given bid vectors b and b' as input. M is group-strategyproof if for every coalition $S \subseteq N$

$$u_i(x', p') \geq u_i(x, p) \quad \forall i \in S \qquad \Longrightarrow \qquad u_i(x', p') = u_i(x, p) \quad \forall i \in S.$$

7.4 Moulin Mechanisms

The Moulin mechanism is based on a cross-monotonic cost sharing method ξ and is described in Algorithm 7.4.1.

Definition 7.17. A cost-sharing method ξ wrt. $C : 2^N \to \mathbb{R}_+$ is a mapping $\xi : (N, 2^N) \to \mathbb{R}_+$ such that $i \notin S \Rightarrow \xi_i(S) = 0$. ξ is budget balanced for $S \subseteq N$, if

$$C(S) = \sum_{i \in S} \xi_i(S).$$

Input: Collect the bids $(b_i)_{i \in N}$ of all players.
Output: (Q, p)

$Q \leftarrow N$
while there is a player $i \in Q$ with $b_i < \xi_i(Q)$ **do**
  $Q \leftarrow Q \setminus \{i \in Q \ : \ b_i < \xi_i(Q)\}$.
end while
for every $i \in Q$ **do**
  define $p_i := \xi_i(Q)$
end for
return (Q, p)

Algorithm 7.4.1: Moulin mechanism $M(\xi)$.

Definition 7.18. A cost-sharing method ξ is cross-monotone if

$$\forall S \subseteq T \subseteq N: \quad \xi_i(T) \leq \xi_i(S) \quad \forall i \in S.$$

Theorem 7.19. Given a cross-monotonic and budget balanced cost sharing method ξ, the Moulin mechanism $M(\xi)$ is group-strategyproof and budget balance.

The proof of this theorem relies on the following lemma.

Lemma 7.20. Let ξ be a cross-monotonic cost sharing method. Then there is a unique inclusion-wise maximal set $S^* \subseteq N$ satisfying the property that for every $i \in S^*$, $b_i \geq \xi_i(S^*)$. The mechanism $M(\xi)$ returns this set.

Proof. Assume there are two different maximal sets S and T satisfying that for every $i \in S$, $b_i \geq \xi_i(S)$ and for every $i \in T$, $b_i \geq \xi_i(T)$. Then for every $i \in S$, $b_i \geq \xi_i(S) \geq \xi_i(S \cup T)$ by cross-monotonicity. Similarly, for every $i \in T$, $b_i \geq \xi_i(T) \geq \xi_i(S \cup T)$. Thus, the set $S \cup T$ also satisfies the condition of the lemma an thus S and T are not maximal.
Let S^* be the unique maximal set satisfying that for every $i \in S^*$, $b_i \geq \xi_i(S^*)$. We prove that $M(\xi)$ never removes a player in S^* from the serviced set Q. For a contradiction, consider the first iteration where $M(\xi)$ removes a player $i \in S^*$ from the current set Q. Thus, we must have $b_i < \xi_i(Q)$. However, since $S^* \subseteq Q$ we have $\xi_i(Q) \leq \xi_i(S^*)$ by cross-monotonicity and therefore $b_i < \xi_i(S^*)$. This contradicts the definition of S^*. Therefore, the set Q returned by the mechanism $M(\xi)$ contains S^*. By maximality of S^*, it cannot contain any other agent and thus $Q = S^*$. □

Suppose S^* is the unique maximal set satisfying the property of Lemma 7.20 with respect to a given bidding profile $b = (b_i)_{i \in N}$. We call the players in S^* the winners. Note that ξ does not dependent on the bidding profile b. Thus, if i is a winner with respect to b, then i remains a winner for every bidding profile (b_{-i}, b'_i) with $b'_i \geq b_i$.

Proof of Theorem 7.19. Clearly, $M(\xi)$ is budget balanced because ξ is budget balanced. It remains to show that $M(\xi)$ is group-strategyproof.
Let $S \subseteq N$ be a coalition of players and b, b' be two bidding profiles with $b'_i = b_i$ for all $i \notin S$ and $b_i = v_i$ for all $i \in S$. Moreover, let (x, p) and (x', p') be the respective

outcomes computed by the Moulin mechanism $M(\xi)$ given b and b'. Assume for the sake of a contradiction that the players in S profit by bidding b' instead of their true valuations b, i.e.,

$$u_i(x', p') \geq u_i(x, p) \quad \forall i \in S \qquad \text{and} \qquad \exists i \in S: \ u_i(x', p') > u_i(x, p). \tag{7.3}$$

We first argue that we can assume without loss of generality that no player in S bids (strictly) more than his true valuation. Let $S^+ \subseteq S$ be the set of all players $i \in S$ with $b'_i > b_i = v_i$. Assume the players in S^+ would lower their bid, one after the other, from their respective bid in b' to the true one in b. If at any step, the outcome of the mechanism changes, say because player $i \in S^+$ changes his bid from b'_i to b_i then by Lemma 7.20 (and, in particular, by the observation above) i must be a winner with respect to b' and a loser with respect to b. Thus, $b'_i \geq \xi_i(Q') > b_i = v_i$, where Q' denotes the unique winner set output by $M(\xi)$ for bidding profile b'. However, this implies that player i's utility $u_i(x', p')$ is negative, which is a contradiction to (7.3). Thus, repeating this argument, every player i in S^+ can lower his bid to $b_i = v_i$ without changing the outcome computed by the mechanism.

We can now assume that $b'_i \leq b_i$ for every $i \in N$. Let Q' and Q be the winner set with respect to the bidding profile b' and b, respectively. By Lemma 7.20, we thus have $Q' \subseteq Q$. By the cross-monotonicity of ξ, $\xi_i(Q') \geq \xi_i(Q)$ for every player $i \in Q'$. Thus, the payment of every player in Q' with respect to b' is at least as much as for the bid b. We conclude that no player in S can be strictly better off by bidding b' instead of b, which is a contradiction to (7.3). □

7.5 Submodular Cost Functions

Suppose the cost function C is submodular, i.e.,

$$\forall S, T \subseteq N : \quad C(S) + C(T) \geq C(S \cup T) + C(S \cap T).$$

Note that the above definition is equivalent to the following: For every $i \in N$ and every set $S \subseteq T \subseteq N \setminus \{i\}$

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

Example 7.21 (Multicast Problem). In the multicast problem we are given an undirected graph $G = (V, E)$ with non-negative edge costs $c_e \geq 0$ and a root node $r \in V$. The player set N corresponds to a subset of the nodes. Every player in N wishes to get connected to the root r. For a given subset $S \subseteq N$, the players in S are connected to r via a pre-specified routing tree $T(S) \subseteq E$. We assume that the routing trees satisfy that if $S_1 \subseteq S_2$ then $T(S_1) \subseteq T(S_2)$. The cost of the routing tree $T(S)$ is equal to the total cost of the edges in the tree, i.e., $C(S) = \sum_{e \in T(S)} c_e$. It can easily be verified that C is submodular.

Consider the following linear programm, which we denote by (LP):

$$\begin{aligned} \max \quad & \sum_{i \in N} \alpha_i \\ \text{s.t.} \quad & \sum_{i \in S} \alpha_i \leq C(S) \quad \forall S \subseteq N \\ & \alpha_i \geq 0 \quad \forall i \in N. \end{aligned}$$

Input: player set $S \subseteq N$
Output: cost shares $(\xi_i(S))_{i \in S}$
1: $\alpha_i \leftarrow 0$ for all $i \in S$
2: $F \leftarrow \emptyset$
3: **while** $S \setminus F \neq \emptyset$ **do**
4: increase all α_i's with $i \in S \setminus F$ uniformly until a new set gets tight
5: let F be the maximal tight set
6: **end while**
7: **return** $(\xi_i(S))_{i \in S} := (\alpha_i)_{i \in S}$

Algorithm 7.5.1: Cost sharing method for submodular cost functions.

Given a feasible solution $\alpha = (\alpha_i)_{i \in N}$, we say that a set $S \subseteq N$ is tight iff $\sum_{i \in S} \alpha_i = C(S)$.

Lemma 7.22. Let α be a feasible solution to (LP). If two sets $S_1, S_2 \subseteq N$ are tight, then so is $S_1 \cup S_2$.

Proof. Since α is feasible, we have $\sum_{i \in S_1 \cap S_2} \alpha_i \leq C(S_1 \cap S_2)$. Moreover, by submodularity of C, we have

$$C(S_1 \cup S_2) \leq C(S_1) + C(S_2) - C(S_1 \cap S_2) \leq \sum_{i \in S_1} \alpha_i + \sum_{i \in S_2} \alpha_i - \sum_{i \in S_1 \cap S_2} \alpha_i = \sum_{i \in S_1 \cup S_2} \alpha_i.$$

Thus, $S_1 \cup S_2$ is tight. □

Corollary 7.23. Let α be a feasible solution to (LP). Then there is a unique maximal tight set, which is simply the union of all tight sets.

A cross-monotonic cost sharing method ξ is described in Algorithm 7.5.1. Call the cost share of a player frozen if the player is contained in a tight set and unfrozen otherwise. The algorithm simply raises all unfrozen cost shares at the same rate until a new set gets tight. The cost shares of all players of the new unique maximal tight set become frozen and the algorithm continues. The algorithm terminates if the cost shares of all players are frozen.

Theorem 7.24. The cost sharing method ξ defined in Algorithm 7.5.1 is cross-monotonic and budget-balanced.

Proof. First observe that the algorithm computes a feasible solution to (LP). (Note that once a set T becomes tight the cost shares of all players contained in T become frozen and remain frozen throughout the execution of the algorithm. Therefore, all constraints of the (LP) are satisfied.) Thus, F is always well defined in Step 5. Moreover, the algorithm terminates if $C(S) = \sum_{i \in S} \alpha_i$ and thus the cost shares $(\xi_i(S))$ satisfy budget balance.
It remains to be shown that ξ is cross-monotonic. Let $S_1 \subset S_2$ and consider the two runs of the algorithm on S_1 and S_2, respectively. We associate a notion of time with the algorithm: At time t all unfrozen cost shares have the same cost share equal to t. It suffices to prove that at any time t the set F_1 of the frozen cost shares in the S_1-run is a

subset of the set of frozen cost shares F_2 in the S_2-run. Let α^1 and α^2 be the cost shares at this time of the S_1-run and S_2-run, respectively. We then have

$$\begin{aligned} C(F_1 \cup F_2) &\leq C(F_1) + C(F_2) - C(F_1 \cap F_2) \\ &\leq \sum_{i \in F_1} \alpha_i^1 + \sum_{i \in F_2} \alpha_i^2 - \sum_{i \in F_1 \cap F_2} \alpha_i^1 \\ &= \sum_{i \in F_1 \setminus F_2} \alpha_i^1 + \sum_{i \in F_2} \alpha_i^2 \\ &\leq \sum_{i \in F_1 \cup F_2} \alpha_i^2, \end{aligned}$$

where the first inequality follows from the submodularity of C, the second follows from the tightness of F_i $(i = 1, 2)$ and the feasiblity of α^1, and the last follows from the fact for every $i \in F_1 \setminus F_2$, since $S_1 \subset S_2$ and i is not frozen at time t in the S_2-run, we have $\alpha_i^2 = t \geq \alpha_i^1$. The above inequality implies that $F_1 \cup F_2$ is tight with respect to α^2. Since F_2 is the maximal tight set we must have $F_1 \cup F_2 \subseteq F_2$ and thus $F_1 \subseteq F_2$ as desired. □

Appendix A

Foundations of Matroids

A.1 Matroids and rank functions

We derive the following properties of matroid rank functions.

Theorem A.1. The rank function of a matroid $M = (R, \mathcal{I})$ is normalized, monotone, subcardinal, and submodular, i.e.,

1. $\rho(\emptyset) = 0$
2. $\rho(T) \leq \rho(U)$ for all $T \subseteq U$, $T, U \subseteq R$.
3. $\rho(T) \leq |T|$ for all $T \subseteq E$.
4. $\rho(T \cap U) + \rho(T \cup U) \leq \rho(T) + \rho(U)$ for all $T, U \subseteq R$.

Proof. (1-3) follows directly from the definition of r. We only show (4).
Let $T, U \subseteq R$. Moreover, let I be an inclusion-wise maximal set in $\mathcal{I}$ with $I \subseteq T \cap U$ and J inclusion-wise maximal in $\mathcal{I}$ with $I \subseteq J \subseteq T \cup U$. As r is a matroid rank function, we get $\rho(T \cap U) = |I|$ and $\rho(T \cup U) = |J|$. We obtain

$$\begin{aligned}\rho(T) + \rho(U) &\geq |J \cap T| + |J \cap U| = |J \cap (T \cap U)| + |J \cap (T \cup U)| \\ &\geq |I| + |J| = \rho(T \cap U) + \rho(T \cup U).\end{aligned}$$

□

We introduce another useful function.

Definition A.2. Let $M = (R, \mathcal{I})$ be a matroid and let $U \subseteq R$. The span of U is defined as:

$$\mathrm{span}(U) := \{r \in R | \rho(U + r) = \rho(U)\}.$$

Proposition A.3.

1. $T \subseteq U \Rightarrow \mathrm{span}(T) \subseteq \mathrm{span}(U)$.
2. $r \in \mathrm{span}(T) \Rightarrow \mathrm{span}(T + r) = \mathrm{span}(T)$

The proof is left as an exercise.

A.2 Circuits

Recall: maximal independent subsets are called bases and minimal dependent ones are called circuits. The set of circuits are denoted by $\mathcal{C}$.

Theorem A.4. Let $M = (R, \mathcal{I})$ be a matroid, $J \in \mathcal{I}$ and $e \in R \setminus J$. Then, either $J + r \in \mathcal{I}$ or $J + r$ contains a uniquely determined circuit C.

Proof. Let $J + r$ be dependent. Define

$$C := \{c \in R : (J + r) - c \in \mathcal{I}\}.$$

Observe that $C \neq \emptyset$ because $e \in C$.

Remark A.5. C is a circuit.

Proof is an exercise. We now show that $C \in \mathcal{C}$ is unique. Let $D \subseteq J + r$ be a circuit. Assume there is $c \in C \setminus D$. Then D is a subset of $J + r - c \in \mathcal{I}$, and therefore $D \in \mathcal{I}$, contradiction. Hence, we get $C \subseteq D$ and since D inclusion-wise minimal, we get $C = D$. □

A.3 Strong Basis Exchange

Now we are ready to prove Theorem 4.26.

Proof of Theorem 4.26. Let $r \in B_1 \setminus B_2$. As $B_2 \in \mathcal{B}$ the set $B_2 + r$ contains a unique circuit C (Thm. A.4) and we have $r \in C$ since every subset of B_2 is independent. We obtain $r \in \mathit{span}(C - r)$, implying $r \in \mathit{span}((B_1 \cup C) - r)$. With Proposition A.3 we get $\mathit{span}((B_1 \cup C) - r) = \mathit{span}(B_1 \cup C) = R$, where the last equality follows, because $B_1 \in \mathcal{B}$. Thus, the set $B_1 \cup C - r$ contains a basis $B_3 \in \mathcal{B}$. We get that $B_1 - r$ and B_3 are independent and $|B_3| > |B_1 - r|$. Thus, there exists $s \in B_3 \setminus (B_1 - r)$, such that $(B_1 - r) + s \in \mathcal{B}$. As $B_3 \setminus (B_1 - r) \subseteq ((B_1 \cup C) - r) \setminus (B_1 - r) \subseteq C - r$ it follows that $s \in C - r$ with $B_1 - r + s \in \mathcal{B}$. Since C is a circuit and $r, s \in C$ we get $B_2 - s + r \in \mathcal{B}$. □

Appendix B

Missing Proofs from Chapter 6.

Proof of Corollary 5.15. Finite models clearly fulfill Assumption 5.12. Thus, the statement for finite models follows by the equivalence of the assertions Theorem 5.13 (2.) $\Leftrightarrow$ Corollary 5.15 (2.). To observe that $\Rightarrow$ holds, note that $x_i^* = x_i^j$ for some $j \in \{1, \ldots, k_i\}$. By setting $\tilde{\alpha}_{ij} = 1$ and $\tilde{\alpha}_{il} = 0, l \neq j$, we may observe that $\pi(u, x^*) = \sum_{i \in N} \pi_i^T \tilde{\alpha}_i$ and $\ell(\tilde{\alpha}) = \ell(x^*)$ hold. Thus, Corollary 5.15 (2.) follows as $\pi(u, x^*) = \sum_{i \in N} \pi_i^T \alpha_i^*$ is assumed to be the optimal value of $LP(u)$. The converse follows immediately.

For concave models, we verify first that Assumption 5.12 is satisfied. We calculate:

$$\mu(\lambda) = \inf_{x_i \in X_i, i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^T(\ell(x) - u)$$

$$= \inf_{x_i \in \text{conv}(X_i), i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^T(\ell(x) - u) \tag{B.1}$$

$$= \inf_{x_i \in \text{conv}(\{x_i^1, \ldots, x_i^{k_i}\}), i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^T(\ell(x) - u) \tag{B.2}$$

$$= \inf_{x_i \in \{x_i^1, \ldots, x_i^{k_i}\}, i \in N} \sum_{i \in N} \pi_i(u, x_i) + \lambda^T(\ell(x) - u) \tag{B.3}$$

The equations (B.1) and (B.3) follow as the function $x_i \mapsto \pi_i(u, x_i) + \lambda^T x_i$ is concave over $\text{conv}(X_i) = \text{conv}(\{x_i^1, \ldots, x_i^{k_i}\})$. The equation (B.2) follows as $\text{conv}(X_i) = \text{conv}\left(\{x_i^1, \ldots, x_i^{k_i}\}\right)$. Furthermore the argumentation shows that all infima are in fact minima as the last expression in (B.3) clearly attains the infimum.

It therefore suffices to show that the assertions Theorem 5.13 (2.) $\Leftrightarrow$ Corollary 5.15 (4.) are equivalent for concave models. Clearly, $\Leftarrow$ holds. For the other direction, we argue as follows: By $\text{conv}(X_i) = \text{conv}\left(\{x_i^1, \ldots, x_i^{k_i}\}\right)$, there exists an $\tilde{\alpha}_i \in \Lambda_i$ with $x_i^* = \sum_{j=1}^{k_i} \tilde{\alpha}_{ij} x_i^j$ for all $i \in N$. As $\pi_i(u, \cdot), i \in N$ are concave, we get $\pi(u, x^*) \geq \sum_{i \in N} \pi_i^T \tilde{\alpha}_i$ and $u = \ell(x^*) \geq \ell(\tilde{\alpha})$ which implies that $\tilde{\alpha}$ is optimal for $LP(u)$ since $\pi(u, x^*) = \sum_{i \in N} \pi_i^T \alpha_i^*$ is assumed to be the optimal value of $LP(u)$. Thus, Corollary 5.15 (4.) follows. □

The LP Complexity

$LP(u)$ may in general involve (exponentially) many variables $\alpha_i, i \in N$ depending on the number $k := \sum_{i \in N} k_i$. A common approach is to dualize $LP(u)$ to yield an LP with less variables at the cost of obtaining (exponentially) many constraints. In the following we dualize the primal problem in the form $-\max\{-\sum_{i \in N} \pi_i^T \alpha_i | \ell(\alpha) \leq u,\ \alpha_i \in \Lambda_i,\ i \in N\}$.

$$\min_{\mu,\lambda} \sum_{i \in N} \mu_i + \sum_{j \in R} \lambda_j u_j \tag{DP(u)}$$
$$\text{s.t.: } \sum_{j \in R} x_{ij}^k \lambda_j + \mu_i \geq -\pi_{ik} \text{ for all } i \in N, k = 1, \dots, k_i,$$
$$\mu_i \in \mathbb{R}, i \in N, \ \lambda_j \geq 0, j \in R.$$

Note that $\mu_i, i \in N$ is not sign-constrained as it is the dual variable to $\sum_k \alpha_{ik} = 1, i \in N$. Moreover, recall that $x_{ij}^k \in \mathbb{R}$ are just parameters in $\mathrm{DP}(u)$. The dual has $n + m$ many variables but exponentially many constraints, but, if we have a polynomial time separation oracle, we can use the ellipsoid method to obtain a polynomial time algorithm. A standard way to obtain such an oracle is to assume an efficient demand oracle.

Definition B.1. For a model I fulfilling Assumption 5.12, a demand oracle for player $i \in N$ gets as input prices $\lambda \in \mathbb{R}^m_{\geq 0}$ and outputs

$$x_i(\lambda) \in \arg \min_{x_i \in \{x_i^1, \dots, x_i^{k_i}\}} \{\pi_i(u, x_i) + \lambda^\intercal x_i\}.$$

We obtain the following result for polynomial time computable demand oracles. Let us remark here that we assume that there is a succinct representation of the model I and hence of $\mathrm{LP}(u)$.

Theorem B.2. Let I be a model which fulfills Assumption 5.12. If for all $\lambda \in \mathbb{R}^m_{\geq 0}$ and $i \in N$, the demand oracle $x_i(\lambda)$ can be computed in polynomial time, then, the optimal value of $\mathrm{LP}(u)$ can be computed in polynomial time.

Proof. In order to use the ellipsoid method, we need to check whether we get a polynomial time separation oracle for the constraints:

$$\sum_{j \in R} x_{ij}^k \lambda_j + \mu_i \geq -\pi_{ik}, i \in N, k = 1, \dots, k_i.$$

With the demand oracle we can compute $\pi_i^*(\lambda) := \pi_i(u, x_i(\lambda)) + \lambda^\intercal x_i(\lambda)$. Now, if $\pi_i^*(\lambda) \geq -\mu_i$ for all $i \in N$, the current point (μ, λ) is feasible. Otherwise, suppose $\pi_i^*(\lambda) < -\mu_i$. As $x_i(\lambda) = x_i^k$ holds for some $k \in \{1, \dots, k_i\}$, we get

$$\pi_i^*(\lambda) = \pi_i(u, x_i^k) + \sum_{j \in R} x_{ij}^k \lambda_j = \pi_{ik} + \sum_{j \in R} x_{ij}^k \lambda_j < -\mu_i,$$

which represents a violated inequality. □

Consequences and Impossibility Results

The characterization result in Theorem 5.13 together with the assumption of a polynomial time demand oracle can be used to establish non-existence results based on complexity-theoretic assumptions like $P \neq NP$. If the optimal value of the master problem $P(u)$ (which is also called the welfare maximization problem in some applications) is NP-hard to compute but there is a polynomial demand oracle, then, assuming $P \neq NP$, the guaranteed (weak) enforceability of u is ruled out since otherwise, we can just compute the optimal solution value of $LP(u)$ in polynomial time (by solving the dual $DP(u)$) which corresponds to the optimal solution value of the master problem.

Appendix C

Grundlagen zu Stetigen Monotonen Funktionen

Lemma C.1. Sei $c : \mathbb{R}_{\geq 0} \to \mathbb{R}$ eine stetige Funktion. Dann sind die folgenden Aussagen äquivalent:

1. c ist monoton auf $\mathbb{R}_{\geq 0}$.
2. Die folgenden Bedingungen gelten:
   (a) Für alle $x > 0$ mit $c(x) > c(0)$ existiert $\epsilon > 0$, so dass $c(y) \geq c(x)$ für alle $y \in (x, x + \epsilon)$.
   (b) Für alle $x > 0$ mit $c(x) < c(0)$ existiert $\epsilon > 0$, so dass$c(y) \leq c(x)$ für alle $y \in (x, x + \epsilon)$.

Proof. $1 \Rightarrow 2$: Trivial.
Für $2 \Rightarrow 1$, leiten wir erst eine Eigenschaft von Funktionen die 2 erfüllen her. Sei $c : \mathbb{R}_{\geq 0} \to \mathbb{R}$ stetig und erfülle 2. Nehme es gebe ein offenes interval (α, ω) mit $c(x) \neq c(0)$ für alle $x \in (\alpha, \omega)$. Wir behaupten, dass c nichtfallend auf (α, ω) ist, falls $c(x) > c(0)$ für alle $x \in (\alpha, \omega)$ und dass c nichtwachsend auf (α, ω) ist, falls $c(x) < c(0)$ für alle $x \in (\alpha, \omega)$. Wir zeigen nur den ersten Fall, da der zweite analog behandelt werden kann. Sei $c(x) > c(0)$ für alle $x \in (\alpha, \omega)$. Per Widerspruch nehmen wir an, dass es $p_1, p_2 \in (\alpha, \omega)$ gibt mit $p_1 < p_2$ und $c(p_1) > c(p_2)$. Wir definieren $p_1' = \max\{x \in [p_1, p_2] : c(x) \geq c(p_1)\}$. Beachte, dass die Menge $\{x \in [p_1, p_2] : c(x) \geq c(p_1)\}$ nichtleer ist da sie p_1 enthält und weiterhin abgeschlossen ist, da c stetig ist. Mit 2 gibt es $\epsilon = \epsilon(p_1') > 0$, so dass $c(y) \geq c(p_1') \geq c(p_1)$ für alle $y \in (p_1', p_1' + \epsilon)$, im Widerspruch zur Maximalität von p_1'.
Nun zeigen wir $2 \Rightarrow 1$. Sei $\alpha = \inf\{x > 0 : c(x) \neq c(0)\}$. Falls $\alpha = \infty$, sind wir fertig, da c konstant ist. Anderenfalls, gilt $c(x) \neq c(0)$ für alle $x > \alpha$. Angenommen diese Behauptung gilt nicht, d.h., es gibt $\omega = \min\{x > \alpha : c(x) = c(0)\}$ und sei $\delta = c(\frac{\omega+\alpha}{2})$ (das Minimum ω wird angenommen, da c stetig ist). Per Konstruktion gilt $c(x) \neq c(0)$ für alle $x \in (\alpha, \omega)$. Falls $c(x) > c(0)$ für alle $x \in (\alpha, \omega)$, gilt $c(x) \geq \delta > c(0)$ für alle $x \in (\frac{\omega+\alpha}{2}, \omega)$ und somit $c(0) = c(\omega) = \lim_{x \nearrow \omega} c(x) \geq \delta > c(0)$, Widerspruch. Anderenfalls gilt $c(x) < c(0)$ für alle $x \in (\alpha, \omega)$ und wir bekommen $c(0) = c(\omega) = \lim_{x \nearrow \omega} c(x) \leq \delta < c(0)$, wiederum ein Widerspruch. Wir folgern $c(x) \neq c(0)$ für alle $x > \alpha$. Somit gilt für jedes $\omega > \alpha$, dass die Funkton c monoton auf dem offenen interval (α, ω) ist und somit ist c monoton auf $\mathbb{R}_{\geq 0}$. □

Wir erhalten eine wichtige Folgerung.

Lemma C.2. Sei $c : \mathbb{R}_{\geq 0} \to \mathbb{R}$ eine stetige, nicht-monotone Funktion. Dann gibt es $x, y \in \mathbb{R}_{>0}$ mit $y > x$, sodass entweder $c(y-x) < c(y) < c(x)$ oder $c(y-x) > c(y) > c(x)$ gilt.

Proof. Mittels Lemma C.1, gilt für jede stetige, nicht-monotone Funktion c, dass es $x > 0$ gibt, so dass einer der folgenden Fälle gilt: $c(x) > c(0)$ und für jedes $\epsilon > 0$ gibt es $y = y(\epsilon) \in (x, x+\epsilon)$, so dass $c(y) < c(x)$; oder $c(x) < c(0)$ und für jedes $\epsilon > 0$ gibt es $y = y(\epsilon) \in (x, x+\epsilon)$, so dass$c(y) > c(x)$. Fixiere ein solches x. Mit der Stetigkeit von c folgt $c(y(\epsilon) - x) \to c(0)$ und $c(y(\epsilon)) \to c(x)$ für $\epsilon \to 0$. Für hinreichend kleines ϵ, haben x und $y(\epsilon)$ die gewünschte Eigenschaft. $\square$

Bibliography

[1] M. Beckmann, C. McGuire, and C. Winsten. *Studies in the Economics and Transportation*. Yale University Press, New Haven, CT, USA, 1956.

[2] G. Brown. *Iterative Solutions of Games by Fictitious Play*, chapter 24. Wiley, Hoboken, NJ, USA, 1951.

[3] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA, USA, 1991.

[4] S. Kakutani. A generalization of Brouwer's fixed point theorem. *Duke Mathematics Journal*, 8(3):457–458, 1941.

[5] D. Monderer and L. Shapley. Fictitious play property for games with identical interests. *J. Econom. Theory*, 68(1):258–265, 1996.

[6] D. Monderer and L. Shapley. Potential games. *Games Econom. Behav.*, 14(1):124–143, 1996.

[7] J. Nash. Equilibrium points in n-person games. *Proc. Natl. Acad. Sci. USA*, 36:48–49, 1950.

[8] J. Nash. *Non-cooperative games*. PhD thesis, Princeton, 1950.

[9] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, MA, USA, 1994.

[10] R. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *Internat. J. Game Theory*, 2(1):65–67, 1973.

[11] M. Voorneveld, P. Borm, F. van Megen, S. Tijs, and G. Facchini. Congestion games and potentials reconsidered. *Int. Game Theory Rev.*, 1(3-4):283–299, 1999.