

NOTIZEN ZUR VORLESUNG

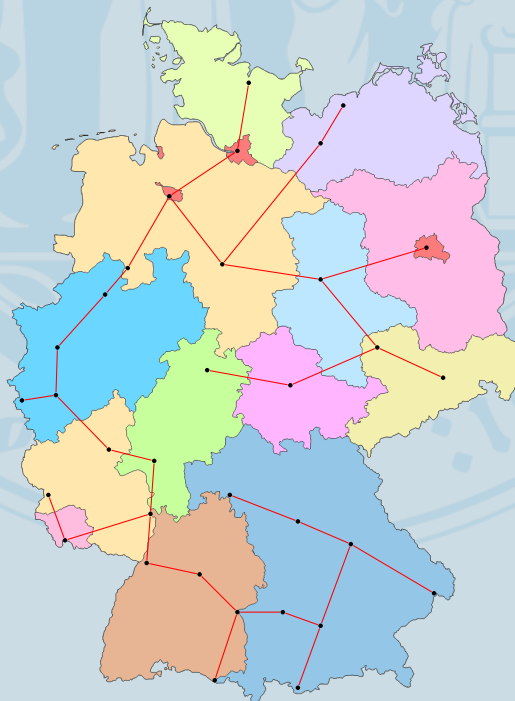
Operations Research

-Lineare & diskrete Optimierung-

im Sommersemester 2014
– Prof. Dr. Frank Vallentin –

Stand:
Samstag, 26. Juli 2014

Version 1.5



Mathematisches Institut
Universität zu Köln

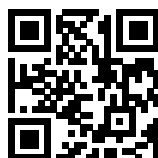
© ⓘ ⓘ ⓘ ⓘ 2018 – Tim Steinbach

Dieses Buch  ist, wie auch weitere Skripte & Zusammenfassungen zu den Themen

- ☒ Analysis
- ☒ Lineare Algebra
- ☒ Numerische Mathematik

online als PDF  verfügbar.

Download  unter  <https://goo.gl/5mbCQc>.



Anmerkungen, Kommentare, Fehlerhinweise oder Verbesserungsvorschläge können gerne dort online abgegeben werden.

Alternativ natürlich auch per E-Mail an:

Tim_Steinbach@T-Online.de

Inhaltsverzeichnis

	Seite
I. Operations Research	
Einführung in die Mathematik des Operations Research	1
i. Kombinatorische Graphen-Algorithmen	
1. Einführung: Stabile Matchings	3
1.1. Grundbegriffe	3
1.2. Der Gale-Shapley-Algorithmus	4
1.3. Geometrische Modellierung	7
2. Kürzeste Wege	9
2.1. Nichtnegative Kantenlängen	10
2.2. Beliebige Kantenlängen	12
2.3. Geometrische Modellierung: Potentiale	15
2.4. Das Rucksackproblem	17
3. Spannbäume	18
3.1. Minimal aufspannende Bäume	19
4. Matchings in bipartiten Graphen	23
4.1. Matchings mit maximaler Kardinalität	23
4.2. Matchings mit maximalem Gewicht	25
4.3. Das Matchingpolytop	27
5. Netzwerkflüsse	32
5.1. Das Max-Flow-Min-Cut-Theorem	32
5.2. Zirkulationen	37
5.3. Flüsse mit minimalen Kosten	39
ii. Theorie der linearen Optimierung	
6. Polyedertheorie	43
6.1. Motivation	43
6.2. Konvexe Mengen	44
6.3. Trenn- und Stütz-Hyperebenen	49
6.4. Extrempunkte & Ecken	54
6.5. Theorem von Minkowski-Weyl	56
6.6. Lemma von Farkas	61
6.7. Lineare Programmierung	63
7. Ganzzahlige Optimierung und vollständig-unimodulare Matrizen	67
7.1. Ganzzahlige lineare Programme	67
7.2. Vollständig-unimodulare Matrizen	69
7.3. Vollständig-unimodulare Matrizen und bipartite Graphen	71
7.4. Vollständig-unimodulare Matrizen und gerichtete Graphen	74

8. Ganzzahlige Optimierung, Schnittebenen und vollständig duale ganzzahlige Systeme	77
8.1. LP und ILP	77
8.2. Chvátal-Gomory-Abschluss	79
8.3. Total duale ganzzahlige Systeme und Hilbertbasen	81
9. Spieltheorie	85
9.1. Bimatrixspiele	85
9.2. Matrixspiele	89
 iii. Algorithmen der linearen Optimierung	
10. Das Eliminationsverfahren von Fourier und Motzkin	92
11. Das Simplexverfahren	94
11.1. Der Algorithmus	95
12. Die Ellipsoidmethode	97
12.1. Grundlegendes zu Ellipsoiden	97
12.2. Separieren und optimieren	99
13. Die Innere-Punkte-Methode	103
13.1. Der zentrale Pfad	103
13.2. Der Algorithmus und seine Analyse	108

II. Verzeichnisse

A. Bibliografie	I
A.a. Literaturverzeichnis	I
B. Index	II
C. Algorithmen	V
D. Definitionen	VI
E. Theoreme	VIII

Einführung in die Mathematik des Operations Research

Operations Research

- Entwicklung und Einsatz **quantitativer** Modelle & Methoden zur Entscheidungsunterstützung
- Geprägt durch die Zusammenarbeit von
 - Mathematik
 - Wirtschaftswissenschaften
 - Informatik

In dieser Vorlesung

- Lineare Modelle und Methoden
- nur schnelle/effiziente Algorithmen

Kombinatorische Graphen-Algorithmen

1. Einführung: Stabile Matchings	3
1.1. Grundbegriffe	3
1.2. Der Gale-Shapley-Algorithmus	4
1.3. Geometrische Modellierung	7
2. Kürzeste Wege	9
2.1. Nichtnegative Kantenlängen	10
2.2. Beliebige Kantenlängen	12
2.3. Geometrische Modellierung: Potentiale	15
2.4. Das Rucksackproblem	17
3. Spannbäume	18
3.1. Minimal aufspannende Bäume	19
4. Matchings in bipartiten Graphen	23
4.1. Matchings mit maximaler Kardinalität	23
4.2. Matchings mit maximalem Gewicht	25
4.3. Das Matchingpolytop	27
5. Netzwerkflüsse	32
5.1. Das Max-Flow-Min-Cut-Theorem	32
5.2. Zirkulationen	37
5.3. Flüsse mit minimalen Kosten	39

1. Einführung: Stabile Matchings

09.04.2014
1. Vorlesung

1.1. Grundbegriffe	3
1.2. Der Gale-Shapley-Algorithmus	4
1.3. Geometrische Modellierung	7

Motivation

Nobelpreis Ökonomie 2012 für ROTH & SHAPLEY
„Theory of stable allocations and the practice of market design“

1.1. Grundbegriffe

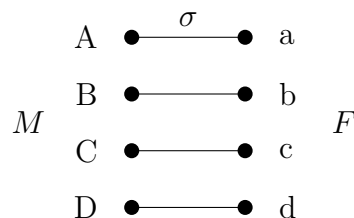
Definition 1.1.1 (Matching):

Seien $M = \{A_1, \dots, A_n\}$, $F = \{a_1, \dots, a_n\}$ Mengen mit n Elementen. Ein **Matching** ist eine Bijektion

$$\sigma: M \rightarrow F$$

Interpretation:

M = Menge der Männer
 F = Menge der Frauen



Ein Matching $\sigma: M \rightarrow F$ entspricht einer Massenhochzeit ohne Homoehe, ohne Bigamie und ohne Singles, d.h. niemand bleibt alleine. Jeder Mann hat eine *Präferenzliste* für Frauen und umgekehrt. Dargestellt wird diese durch zwei Matrizen mit je n^2 Einträgen.

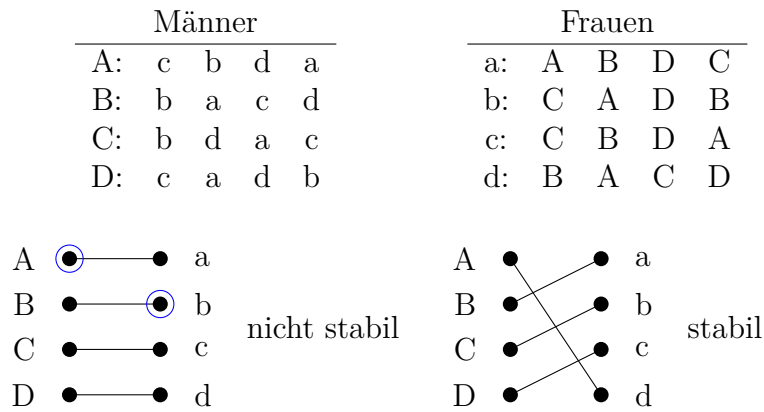
Definition 1.1.2 (stabiles und nicht stabiles Matching):

Ein Matching heißt **nicht stabil**, wenn $\exists A \in M, a \in F$

1. $\sigma(A) \neq a$
2. $a >_A \sigma(A)$
3. $A >_a \sigma^{-1}(a)$

(Dann sind A und a einem Seitensprung nicht abgeneigt.)

Ein **stabiles Matching** ist ein nicht nicht stabiles Matching.

Beispiel 1.1.1:

Notation: $c >_A b$, d. h. Mann A bevorzugt Frau c über Frau b .

Anwendungen: Zum Beispiel die Verteilung von Studenten auf Hochschulen, etc.

Fragen:

1. Gibt es immer ein stabiles Matching?
2. Wie findet man ein stabiles Matching?

Naive Antwort auf Frage 2.: Teste alle $n!$ Matchings. Der Nachteil ist die Laufzeit.

Tabelle 1.1.: Laufzeit

n	Rechenzeit (Eine Bijektion $\hat{=}$ 1 ns)
10	0.003 Sekunden
15	21 Minuten
20	77 Jahre
25	$4.9 \cdot 10^8$ Jahre ($\hat{=}$ $\frac{1}{10}$ Alter der Erde)

1.2. Der Gale-Shapley-Algorithmus (1962)

Viel bessere Antwort auf die beiden Fragen 1. und 2. oben.

Algorithmus 1.2.1 Gale-Shapley-Algorithmus

```

1: Verlobe alle Frauen  $a_1, \dots, a_n$  mit dem virtuellen Mann  $A_0$ 
2: for  $k \leftarrow 1$  to  $n$  do
3:    $X \leftarrow A_k$ 
4:   while  $X \neq A_0$  do
5:      $x \leftarrow$  beste Frau auf der Liste von  $X$ 
6:      $X$  macht  $x$  Heiratsantrag
7:     if  $X >_x$  aktueller Verlobter von  $x$  then
8:       Verlobe  $X$  mit  $x$ 
9:        $X =$  bisheriger Verlobter von  $x$ 
10:    end if

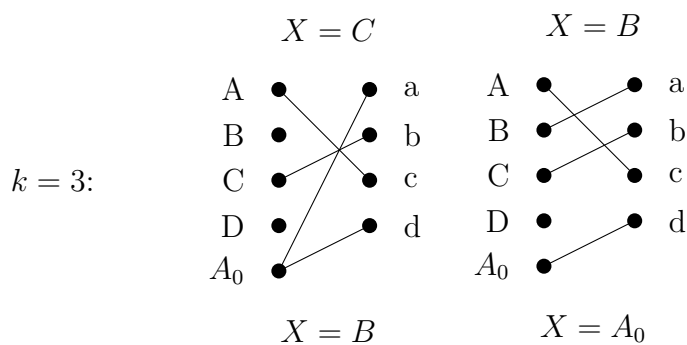
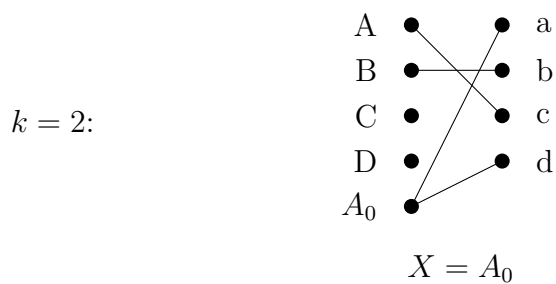
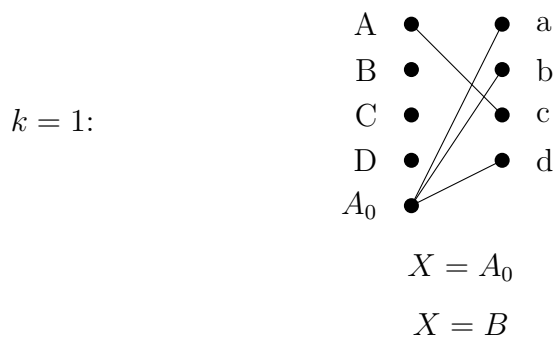
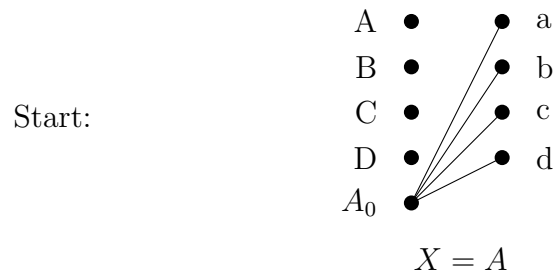
```

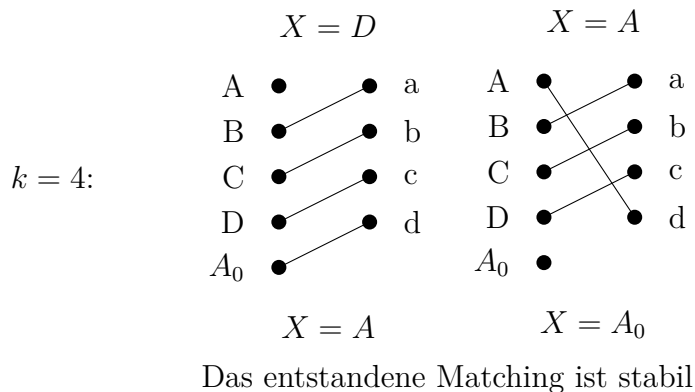
Algorithmus Gale-Shapley-Algorithmus (Fortsetzung)

```

11:   if  $X \neq A_0$  then
12:       Streiche  $x$  von der Liste von  $X$ 
13:   end if
14: end while
15: end for
16: Massenhochzeit der Verlobten

```

Beispiel 1.2.1:

**Satz 1.2.1:**

Der Gale-Shapley-Algorithmus berechnet ein stabiles Matching. Insbesondere existiert immer ein stabiles Matching.

Beweis:

Der Algorithmus ist *wohldefiniert*, d. h. die Listen der Männer sind niemals leer: Angenommen die Liste von A wäre leer. Dann ist A von allen Frauen abgewiesen worden. Aber jede Frau ist mit höchstens einem Mann verlobt. Die Situation der Frau verbessert sich stets und A ist besser als A_0 . Also gibt es n andere, echte Männer, neben A . ⚡

Der Algorithmus 1.2.1 liefert ein Matching: trivial.

Das resultierende Matching ist stabil:

Seien $A \in M$ und $a \in F$ mit $\sigma(A) \neq a$. Annahme: $a >_A \sigma(A)$. Dann ist A von a im Laufe des Algorithmus abgewiesen worden und $\sigma^{-1}(a) >_a A$, d. h. das Matching ist stabil. $\square$

Bemerkung 1.2.1:

Im Gale-Shapley-Algorithmus werden maximal $n + (n - 1)^2$ Heiratsanträge gemacht. Was viel besser ist als $n!$.

11.04.2014
2. Vorlesung

Lemma 1.2.1:

Falls „streiche x von X 's Liste“ aufgerufen wird, gibt es einen Mann $\tilde{X}$, so dass $\tilde{X} >_x X$ und $x >_{\tilde{X}}$ alle Frauen auf $\tilde{X}$'s verbleibender Liste.

Beweis:

Falls „streiche x von X 's Liste“ ausgeführt wird, gilt

1. X möchte x heiraten, aber $\tilde{X} >_x X$, wobei $\tilde{X}$ der aktuelle Verlobte von x ist.

oder

2. X und x waren verlobt, aber x bekam ein Angebot von dem attraktiveren Mann $\tilde{X}$ mit $\tilde{X} >_x X$.

Es gibt also in beiden Situationen einen Mann $\tilde{X}$ mit $\tilde{X} >_x X$ und $x >_{\tilde{X}}$ für alle Frauen auf $\tilde{X}$'s verbleibender Liste. $\square$

Satz 1.2.2:

Der Gale-Shapley-Algorithmus berechnet das beste stabile Matching für die Männer, d. h. wenn x von der Liste von X entfernt wird, dann existiert kein stabiles Matching σ mit $\sigma(X) = x$.

Beweis:

Zu zeigen: Für alle stabilen Matchings σ mit $X \in M$ und $x \in F$ gilt

$$\sigma(X) = x \Rightarrow x \text{ wird nicht von } X' \text{'s Liste gestrichen}$$

Angenommen das stimmt nicht, dann existiert ein stabiles Matching σ mit mindestens einem Paar (X, x) , so dass $\sigma(X) = x$ und x wird von X 's Liste gestrichen. Wähle unter diesen Paaren (X, x) dasjenige aus, bei dem die „Streichung“ im Laufe des Algorithmus am frühesten passiert.

Nach Lemma 1.2.1 gibt es $\tilde{X}$ mit $\tilde{X} >_x X$ und $x >_{\tilde{X}}$ für alle Frauen auf $\tilde{X}$'s verbleibender Liste, zum Zeitpunkt der Streichung.

Beachte $\sigma(\tilde{X})$: Da σ stabil, gilt $\sigma(\tilde{X}) >_{\tilde{X}} x = \sigma(X)$. Das heißt $\sigma(\tilde{X})$ wurde zu einem früheren Zeitpunkt von $\tilde{X}$'s Liste gestrichen. $\nLeftarrow$ zur Wahl von (X, x) .

□

Korollar 1.2.2.a:

Das stabile Matching, welches der Gale-Shapley-Algorithmus berechnet, ist unabhängig von der Reihenfolge der Männer und Frauen.

1.3. Geometrische Modellierung

Der Gale-Shapley-Algorithmus ist ein kombinatorischer Algorithmus, verwendet also keine Zahlen.

Vorteile:

+ sehr schnell bzw. effizient

Nachteile:

– relativ kompliziert

– sehr problemspezifisch

Jetzt: Geometrische Modellierung mit linearen Ungleichungen

Vorteile:

+ einfache Modellierung

+ leicht veränderbar

Nachteile:

– im Allgemeinen nicht die schnellste Lösungsmethode

Man kann ein stabiles Matching $\sigma : M \rightarrow F$, mit $M = \{A_1, \dots, A_n\}$, $F = \{a_1, \dots, a_n\}$, mit einer Permutationsmatrix $X \in \{0, 1\}^{n \times n}$ codieren, wobei

$$X_{i,j} = \begin{cases} 1, & \text{falls } \sigma(A_i) = a_j \\ 0, & \text{sonst} \end{cases}$$

Satz 1.3.1:

$X \in \{0, 1\}^{n \times n}$ codiert ein stabiles Matching

$\Leftrightarrow X$ erfüllt die folgenden linearen Ungleichungen

1. Zeilensummen:

$$\sum_{j=1}^n X_{i,j} \leq 1 \quad \forall i = 1, \dots, n$$

2. Spaltensummen:

$$\sum_{i=1}^n X_{i,j} \leq 1 \quad \forall j = 1, \dots, n$$

3. $X_{i,j} \geq 0 \quad \forall i, j = 1, \dots, n$

4. Stabilitätsbedingung:

$$X_{i,j} + \sum_{\substack{k \\ a_k >_{A_i} a_j}} X_{i,k} + \sum_{\substack{k \\ A_k >_{a_j} A_i}} X_{k,j} \geq 1 \quad \forall i, j = 1, \dots, n$$

Beweis:

trivial ✓

□

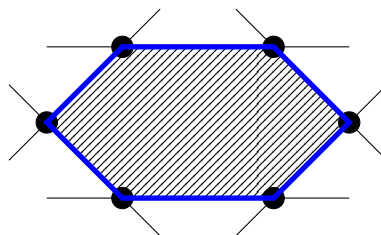
Später in der Vorlesung (Kapitel 6: Polyedertheorie)

Die Menge

$$SMP := \{X \in \mathbb{R}^{n \times n} : X \text{ erfüllt die Bedingungen aus Satz 1.3.1}\}$$

ist ein **Polytop** (d. h. eine beschränkte Menge, gegeben durch lineare Ungleichungen).

Beispiel 1.3.1:



Ein 2-dimensionales Polytop mit 6 Ecken

Später: Die Ecken von SMP entsprechen den stabilen Matchings (Vande Vate, 1989).

Sei $C \in \mathbb{R}^{n \times n}$ die Heiratszielmatrix mit $C_{i,j}$ = Wert der Heirat von A_i mit a_j .

Das optimale stabile Matching bezüglich C ist gegeben durch ein *lineares Optimierungsproblem*

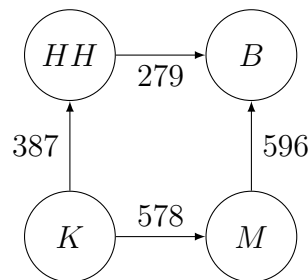
$$\max \sum_{i=1}^n \sum_{j=1}^n C_{i,j} X_{i,j}$$

wobei $X \in \mathbb{R}^{n \times n}$ die Bedingungen von Satz 1.3.1 erfüllt.

2. Kürzeste Wege

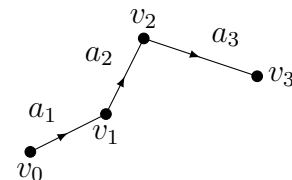
2.1. Nichtnegative Kantenlängen	10
2.2. Beliebige Kantenlängen	12
2.3. Geometrische Modellierung: Potentiale	15
2.4. Das Rucksackproblem	17

Ziel: Finde den kürzesten Weg in einem Netzwerk.



Notation:

- $D = (V, A)$ **gerichteter Graph**
- V = endliche Menge von **Knoten**
- $A \subseteq \{(v, w) \in V \times V : v \neq w\}$ Menge der **gerichteten Kanten**
- $l: A \rightarrow \mathbb{Z}_{\geq 0} = \{0, 1, 2, \dots\}$ **Längenfunktion**
- $P = (v_0, a_1, v_1, a_2, \dots, a_m, v_m)$ heißt **Kantenfolge**,
wenn $v_0, \dots, v_m \in V$, $a_1, \dots, a_m \in A$, $a_i = (v_{i-1}, v_i)$
- $l(P) = \sum_{i=1}^m l(a_i)$ **Länge** von P
- v_0 = **Startknoten**
- v_m = **Endknoten**
- Wenn $|\{v_0, \dots, v_m\}| = m + 1$, wobei $v_i \neq v_j$ für $i \neq j$, dann heißt P **v_0 - v_m -Weg**
oder kurz **Weg**
- Wenn $U \subseteq V$ sei
 - $\delta^+(U) = \{(v, w) \in A : v \in U, w \notin U\}$
 - $\delta^-(U) = \{(v, w) \in A : v \notin U, w \in U\}$



2.1. Nichtnegative Kantenlängen

Definition 2.1.1 (Abstand):

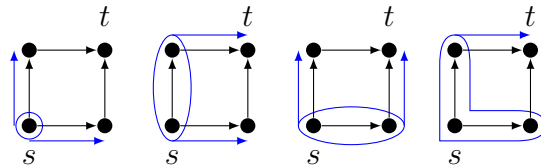
Seien $s, t \in V$.

$$\text{dist}(s, t) = \textbf{Abstand von } s \text{ zu } t = \min l(P) \text{ mit } P \text{ } s\text{-}t\text{-Weg}$$

Definition 2.1.2 (s - t -Schnitt):

Seien $s, t \in V$. $A' \subseteq A$ heißt **s - t -Schnitt**, falls $A' = \delta^+(U)$ für ein $U \subseteq V$ und $s \in U, t \notin U$.

Beispiel 2.1.1 (s - t -Schnitte):



Satz 2.1.1 (min-max-Charakterisierung kürzester Wege, Robacker, 1956):

Sei $D = (V, A)$ ein gerichteter Graph, $l: A \rightarrow \mathbb{Z}_+$, $s, t \in V$. Dann ist

$$\begin{aligned} \text{dist}(s, t) &= \min_{P \text{ } s\text{-}t\text{-Weg}} l(P) \\ &= \max k \end{aligned}$$

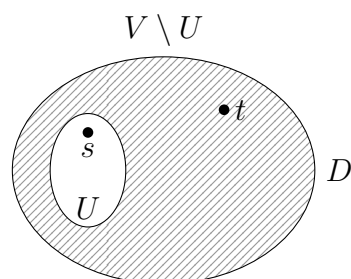
wobei k die Anzahl von s - t -Schnitten $C_1, \dots, C_k$ ist, so dass gilt

$$\forall a \in A: \left| \{j \in \{1, \dots, k\} : a \in C_j\} \right| \leq l(a)$$

Beweis:

Falls es keinen s - t -Weg gibt ist die Aussage trivial, beide Seiten sind $+\infty$. Im folgenden existiere ein s - t -Weg.

„min $\geq$ max“: Sei $P = (s = v_0, a_1, v_1, \dots, a_m, v_m = t)$ ein s - t -Weg und seien $C_1, \dots, C_k$ s - t -Schnitte mit den oben genannten Eigenschaften. Dann



$$\begin{aligned} l(P) &= \sum_{i=1}^m l(a_i) \\ &\geq \sum_{i=1}^m \left| \{j = 1, \dots, k : a_i \in C_j\} \right| \\ &= \sum_{j=1}^k \left| C_j \cap \{a_1, \dots, a_m\} \right| \\ &\geq \sum_{j=1}^k 1 = k \end{aligned}$$

„ $\min \leq \max$ “: Sei $U_i = \{v \in V : \text{dist}(s, v) < i\}$ für $i = 1, \dots, \text{dist}(s, t) = \min$.

Definiere $C_i = \delta^+(U_i) = \{(v, w) \in A : v \in U_i, w \notin U_i\} \forall i$. Diese C_i sind $\text{dist}(s, t)$ -viele s - t -Schnitte. Sei $a = (u, v) \in A$ eine Kante. Es gilt $l(a) \geq \text{dist}(s, v) - \text{dist}(s, u)$. Andererseits enthalten nur die s - t -Schnitte $C_{\text{dist}(s, u)+1}, \dots, C_{\text{dist}(s, v)}$ diese Kante a , was $\text{dist}(s, v) - \text{dist}(s, u)$ viele sind.

□

Beispiel 2.1.2:

$\text{dist}(K, B) = 387 + 279$, weil es $387 + 279$ K - B -Schnitte gibt, nämlich



Verwendung als Zertifikat

16.04.2014
3. Vorlesung

Sei P ein s - t -Weg. Dann gilt $l(P) \geq \text{dist}(s, t)$. Seien $C_1, \dots, C_{k'}$ s - t -Schnitte mit den oben genannten Eigenschaften. Dann ist $k^* \geq k'$, wobei k^* das Maximum ist. Angenommen $k' = l(P)$. Dann gilt $l(P) = \text{dist}(s, t)$, weil

$$l(P) \geq \text{dist}(s, t) = k^* \geq k'$$

$$\Rightarrow l(P) = \text{dist}(s, t).$$

Bemerkung 2.1.1:

Das heißt die s - t -Schnitte $C_1, \dots, C_{k'}$ geben ein Zertifikat dafür, dass P ein kürzester s - t -Weg ist.

Später in der Vorlesung:

Dies ist ein Beispiel für Dualität in linearer Optimierung (Abschnitt 6.7).

Algorithmus 2.1.1 Dijkstra-Algorithmus

Eingabe: $D = (V, A)$ gerichteter Graph, $s \in V$ Startknoten, $l: A \rightarrow \mathbb{Z}_+$

- 1: Initialisiere: $f(s) = 0, f(v) = +\infty \forall v \in V \setminus \{s\}$
 - 2: $U = V$
 - 3: **while** $U \neq \emptyset$ **do**
 - 4: **find** $u \in U$ **so that** $f(u) = \min_{u' \in U} f(u')$
 - 5: **for** $a \leftarrow (u, v) \in A$ **do**
 - 6: **if** $f(v) > f(u) + l(a)$ **then**
 - 7: $f(v) = f(u) + l(a)$
 - 8: **end if**
 - 9: $U = U \setminus \{u\}$
 - 10: **end for**
 - 11: **end while**
-

Satz 2.1.2:

Nach Ablauf des Dijkstra-Algorithmus gilt

$$f(v) = \text{dist}(s, v) \quad \forall v \in V$$

Beweis:

Klar: $f(v) \geq \text{dist}(s, v)$.

Es bleibt zu zeigen, dass $f(v) \leq \text{dist}(s, v) \forall v \in V$. Dies zeigen wir per Induktion über die **while**-Schleife.

Induktionsanfang: trivial, da $u = v$

Sei $u \in U$ das in der **while**-Schleife betrachtete u . Angenommen es gelte $f(u) > \text{dist}(s, u)$.

Sei $s = (v_0, a_1, v_1, \dots, a_m, v_m) = u$ ein kürzester s - u -Weg. Sei i minimal mit $v_i \in U$.

1. Fall: $i = 0$

Dann $f(v_i) = f(s) = 0 = \text{dist}(s, v_i)$

2. Fall: $i > 0$

Dann

$$\begin{aligned} f(v_i) &\leq f(v_{i-1}) + l(v_{i-1}, v_i) \\ &\leq \text{dist}(s, v_{i-1}) + l(v_{i-1}, v_i) \\ &= \text{dist}(s, v_i) \end{aligned}$$

Also:

$$\begin{aligned} f(v_i) &\leq \text{dist}(s, v_i) \\ &\leq \text{dist}(s, u) \\ &< f(u) \quad \nexists \text{ Wahl von } u \end{aligned}$$

$$\Rightarrow f(u) \leq \text{dist}(s, u)$$

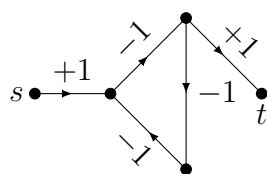
□

Rechenzeit des Dijkstra-Algorithmus

Anzahl der Iterationen:	$ V $
in jeder Iteration:	Berechnung des Minimums $ V $
Betrachte Kanten:	Maximal $ V $
$\Rightarrow$ Rechenzeit ist proportional zu $ V ^2$	

2.2. Beliebige Kantenlängen

Manchmal sind auch negative Kantenlängen nützlich, z. B. für das Rucksackproblem (siehe Abschnitt 2.4), bei dem man den längsten Weg zwischen s und t finden möchte (d. h. finde den kürzesten Weg mit negativen Kantenlängen).



Der kürzeste s - t -Weg hat Länge 1

Die kürzeste s - t -Kantenfolge ist nicht definiert

Problem: Kreise negativer Länge.

Definition 2.2.1 (gerichteter Kreis):

Eine Kantenfolge $P = (v_0, a_1, v_1, \dots, a_m, v_m)$ heißt **(gerichteter) Kreis**, falls

$$v_0 = v_m \text{ und } \left| \{v_i : i = \{0, \dots, m\}\} \right| = m$$

Wir gehen in diesem Abschnitt von der Annahme aus, dass $D = (V, A)$, $l: A \rightarrow \mathbb{Z}$ keine Kreise negativer Länge enthält.

Satz 2.2.1:

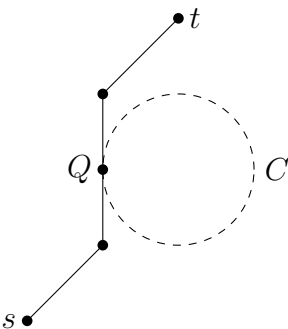
$D = (V, A)$, $l: A \rightarrow \mathbb{Z}$ enthält keine Kreise negativer Länge, $s, t \in V$ sind durch eine Kantenfolge verbunden. Dann ist unter der Menge kürzester s - t -Kantenfolgen ein kürzester s - t -Weg.

Beweis:

Klar: $\exists$ kürzester s - t -Weg P .

Angenommen es existiert eine kürzere s - t -Kantenfolge Q mit $l(Q) < l(P)$.

Wähle ein solches Q mit minimaler Kantenanzahl. Da Q kein Weg ist enthält Q einen Kreis C . Nach Voraussetzung ist $l(C) \geq 0$. Sei Q' die s - t -Kantenfolge, die aus Q durch das Löschen von C entsteht. Dann ist Q' ebenfalls eine s - t -Kantenfolge mit



$$\begin{aligned} l(Q') &= l(Q) - \underbrace{l(C)}_{\geq 0} \\ &\leq l(Q) \\ &< l(P) \end{aligned}$$

und Q' hat weniger Kanten als Q . Widerspruch zur Wahl von Q . □

Algorithmus 2.2.1 Bellman-Ford-Algorithmus

Eingabe: $D = (V, A)$ gerichteter Graph, $|V| = n$, $s \in V$ Startknoten, $l: A \rightarrow \mathbb{Z}$

- 1: Initialisiere: $f_0(s) = 0$, $f_0(v) = \infty \quad \forall v \in V \setminus \{s\}$
- 2: **for** $k \leftarrow 0$ **to** $n - 1$ **do**
- 3: $f_{k+1}(v) = f_k(v) \quad \forall v \in V$
- 4: **for** $a \leftarrow (u, v) \in A$ **do**
- 5: **if** $f_{k+1}(v) > f_k(u) + l(a)$ **then**
- 6: $f_{k+1}(v) = f_k(u) + l(a)$
- 7: $g(v) = u$
- 8: **end if**
- 9: **end for**
- 10: **end for**
- 11: **if** $f_n(v) \neq f_{n-1}(v)$ **then**
- 12: D enthält einen, von s aus erreichbaren, negativen gerichteten Kreis
- 13: **end if**

Ausgabe: Funktionen $f_0, \dots, f_n: V \rightarrow \mathbb{Z}$ und $g: V \rightarrow V$

Satz 2.2.2:

$$f_k(v) = \min \{l(P) : P \text{ ist eine } s\text{-}v\text{-Kantenfolge mit } \leq k \text{ Kanten}\}$$

$$\stackrel{\text{Satz 2.2.1}}{=} \min \{l(P) : P \text{ ist ein } s\text{-}v\text{-Weg mit } \leq k \text{ Kanten}\}$$

Beweis:

Offensichtliche Induktion über k .

□

Satz 2.2.3:

$$f_n = f_{n-1}$$

$\iff$ Alle von s aus erreichbaren gerichteten Kreise haben nicht negative Länge

Beweis:

„ $\Rightarrow$ “: Angenommen es gilt $f_n = f_{n-1}$. Wir zeigen zunächst, dass dann für jede Kante $e = (u, v) \in A$ gilt, dass $l(e) \geq f_{n-1}(v) - f_{n-1}(u)$. Da eine s - u -Kantenfolge aus $\leq n-1$ Kanten zusammen mit der Kante e eine s - v -Kantenfolge aus $\leq n$ Kanten ergibt, haben wir $f_{n-1}(u) + l(e) \geq f_n(v) = f_{n-1}(v)$, was nach Satz 2.2.2 das Gewünschte zeigt.

Für einen Kreis $C = (v_0, a_1, v_1, a_2, \dots, a_{m-1}, v_{m-1}, a_m, v_m)$ mit $v_m = v_0$ gilt also:

$$l(C) = \sum_{i=1}^m l(a_i) \geq \sum_{i=1}^m f_{n-1}(v_{i-1}) - f_{n-1}(v_i) = 0$$

„ $\Leftarrow$ “: Nehmen wir an, dass alle gerichteten Kreise, die von s aus erreichbar sind, nicht-negative Länge besitzen. Für einen Knoten v vergleichen wir $f_{n-1}(v)$ und $f_n(v)$. Nach Satz 2.2.2 sind beide Werte Minima, wobei wir für $f_n(v)$ über eine größere Menge minimieren, die zusätzlich auch s - v -Kantenfolgen, die n Kanten enthalten, beinhaltet.

Sei P eine s - v -Kantenfolge mit n Kanten. P muss dann mindestens einen Knoten zweimal besuchen und damit einen gerichteten Kreis C enthalten. Lassen wir C aus, erhalten wir eine s - v -Kantenfolge P' , die weniger Kanten als P und damit $\leq n-1$ Kanten enthält. Da der gerichtete Kreis C von s erreichbar ist, hat er nicht-negative Länge. Wir haben also:

$$l(P) = l(P') + l(C) \geq l(P')$$

Damit haben wir gezeigt, dass es für jede s - v -Kantenfolge mit n Kanten eine mit $n-1$ Kanten gibt, die kürzere oder gleich große Länge besitzt. Das Hinzunehmen der Kantenfolgen aus n Kanten ändert also das Minimum nicht. Damit gilt $f_n = f_{n-1}$.

□

Beobachtungen zum Bellman-Ford-Algorithmus

$$\text{Laufzeit ist proportional zu: } |V| \cdot \underbrace{|A|}_{\substack{\text{maximal} \\ 2 \cdot \binom{|V|}{2} \approx |V|^2}} \leq |V|^3$$

Insbesondere:

- $f_{n-1}(v) = \text{dist}(s, v)$
- $v, g(v), g(g(v)), \dots, s$ ist eine Umkehrung eines kürzesten s - v -Weges

2.3. Geometrische Modellierung: Potentiale

23.04.2014
4. Vorlesung

Definition 2.3.1 (Potential):

Sei $D = (V, A)$ ein gerichteter Graph und $l: A \rightarrow \mathbb{Z}$ die zugehörige Längenfunktion. Eine Funktion $p: V \rightarrow \mathbb{R}$ heißt **Potential** oder **Potenzialfunktion** für D und l , falls

$$\forall a = (u, v) \in A: l(a) \geq p(v) - p(u)$$

Satz 2.3.1:

Sei $D = (V, A)$ ein gerichteter Graph und $l: A \rightarrow \mathbb{Z}$ die zugehörige Längenfunktion. Es gilt:

$$\exists \text{ Potential } p: V \rightarrow \mathbb{R} \text{ für } D \iff \forall \text{ gerichteten Kreise } C \text{ in } D \text{ gilt } l(C) \geq 0$$

Beweis:

„ $\Leftarrow$ “: Füge zu $D = (V, A)$ in V einen neuen Knoten s hinzu, der mit allen anderen Knoten verbunden ist, wobei die Länge der neuen Kanten $(s, t) \forall t \in V$ gleich 0 ist. Definiere für $t \in V$: $p(t) = \text{dist}(s, t)$. Dies ist eine Potentialfunktion, weil $\forall a = (u, v) \in A$ gilt

$$p(v) - p(u) = \text{dist}(s, v) - \text{dist}(s, u) \leq l(a) \Leftrightarrow \text{dist}(s, v) \leq \text{dist}(s, u) + l(a)$$

Dies gilt, weil $\overset{\text{dist}(s, u)}{\overbrace{s \cdots u}} \overset{l(a)}{\overbrace{u \cdots v}}$ eine s - v -Kantenfolge der Länge $\text{dist}(s, u) + l(a)$ ist und weil alle gerichteten Kreise nicht-negative Länge haben, gibt es eine kürzeste s - v -Kantenfolge die ein Weg ist, also

$$\text{dist}(s, v) \leq \text{dist}(s, u) + l(a)$$

„ $\Rightarrow$ “: Sei $C = (v_0, a_1, v_1, \dots, a_m, v_m)$ ein gerichteter Kreis, d. h. $v_0 = v_m$. Sei weiterhin p ein Potenzial. Dann

$$l(C) = \sum_{i=1}^m l(a_i) \geq \underbrace{\sum_{i=1}^m p(v_i) - p(v_{i-1})}_{\text{Teleskopsumme}} = 0$$

□

Satz 2.3.2 (geometrische Modellierung kürzester Wege):

Sei $D = (V, A)$ ein gerichteter Graph und $l: A \rightarrow \mathbb{Z}$ die zugehörige Längenfunktion. D enthält keine gerichteten Kreise negativer Länge. Dann gilt für $s, t \in V$, welche durch einen s - t -Weg verbunden sind:

$$\begin{aligned} \text{dist}(s, t) &= \min_{P \text{ s-t-Weg}} l(P) \\ &= \max p(t) - p(s) \end{aligned}$$

mit $p: V \rightarrow \mathbb{R}$ und $l(a) \geq p(v) - p(u) \quad \forall a = (u, v) \in A$.

Beweis:

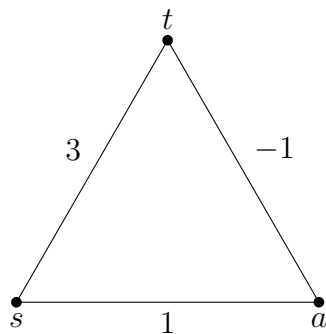
„ $\min \geq \max$ “: Sei $P = (s = v_0, a_1, v_1, \dots, a_m, v_m = t)$ eine s - t -Kantenfolge und sei $p: V \rightarrow \mathbb{R}$ ein Potenzial. Dann gilt

$$\begin{aligned} l(P) &= \sum_{i=1}^m l(a_i) \\ &\geq \sum_{i=1}^m p(v_i) - p(v_{i-1}) \\ &= p(t) - p(s) \end{aligned}$$

„ $\min \leq \max$ “: Setze $p(v) = \text{dist}(s, v)$, falls v zu einer s - v -Kantenfolge gehört und $p(v) = 0$ sonst. Dann ist p ein Potenzial und es gilt

$$p(t) - p(s) = \text{dist}(s, t) - 0 = \text{dist}(s, t)$$

□

Beispiel 2.3.1 (geometrische Modellierung kürzester Wege):

$$\begin{aligned} \text{dist}(s, t) &= \max p_3 - p_1 \\ &\quad p_1, p_2, p_3 \in \mathbb{R} \\ &\quad p_2 - p_1 \leq 1 \\ &\quad p_3 - p_1 \leq 3 \\ &\quad p_3 - p_2 \leq -1 \\ &= \max(-1, 0, 1)p \\ &\quad p \in \mathbb{R}^3 \\ &\quad (-1, 1, 0)p \leq 1 \\ &\quad (-1, 0, 1)p \leq 3 \\ &\quad (0, -1, 1)p \leq -1 \end{aligned}$$

2.4. Das Rucksackproblem

Beispiel 2.4.1:

Der Rucksack hat ein Volumen von acht Litern und es gibt fünf nützliche Gegenstände.

Gegenstand i	Volumen a_i	Nutzwert c_i
1	5	4
2	3	7
3	2	3
4	2	5
5	1	4

Aufgabe: Finde eine Auswahl der Gegenstände 1 bis 5, so dass diese in den Rucksack passen und ihr summierter Nutzwert maximal ist.

Formulierung als Kürzeste-Wege-Problem (dynamische Programmierung)

Knoten: $(i, x) \in V, \quad i = 0, \dots, 6, \quad x = 0, \dots, 8$

Kanten: – $((i-1, x), (i, x))$ mit Länge 0 für $i = 1, \dots, 5$

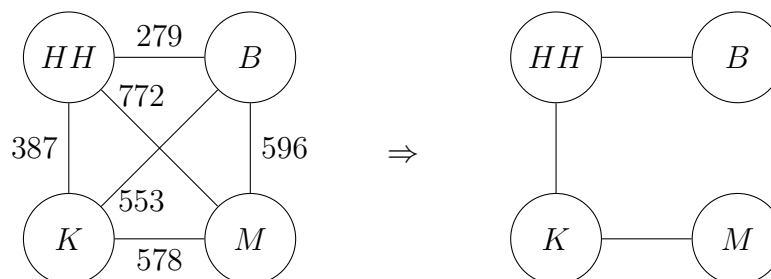
– $((i-1, x), (i, x + a_i))$ mit Länge $-c_i$ für $i = 1, \dots, 5$

– $((5, x), (6, 8))$ mit Länge 0 $\forall x$

Dann ergeben, im Beispiel 2.4.1, die kürzesten $(0, 0)$ - $(6, 8)$ -Wege die optimale Auswahl.

3. Spannbäume

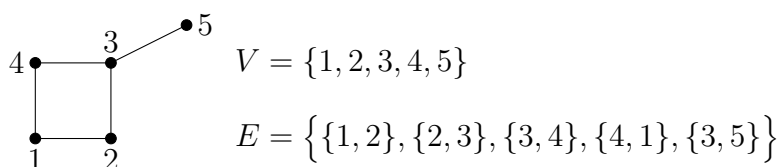
Problem: Effizientes Design eines Netzwerks. Verlege dabei möglichst wenig Kabel, Rohre, etc.

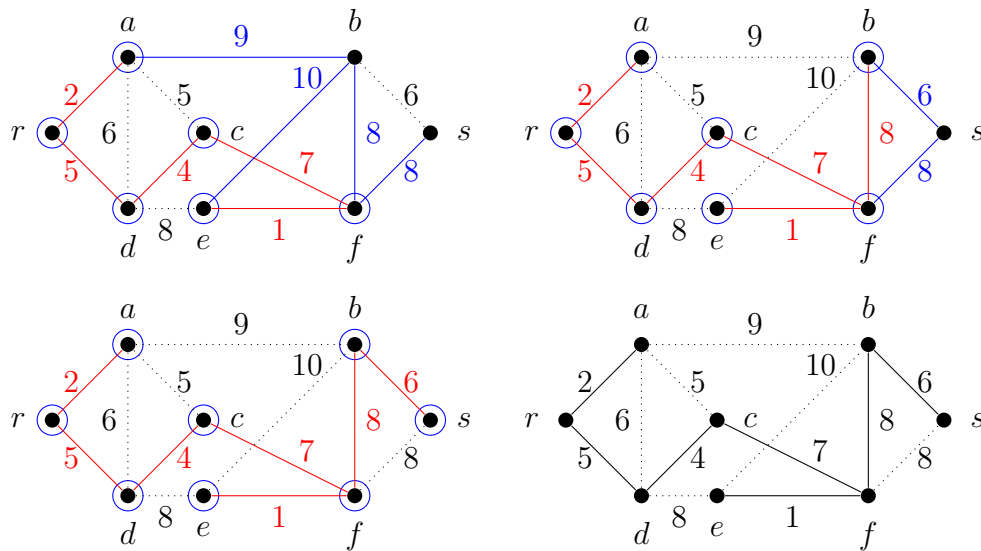


Notation:

- $G = (V, E)$ **ungerichteter Graph**
- V = endliche Menge von **Knoten**
- $E \subseteq \{\{u, v\} : u, v \in V, u \neq v\}$ Menge der **ungerichteten Kanten**
- $l: E \rightarrow \mathbb{Z}$ **Kantenlängen**
- $F \subseteq E: l(F) = \sum_{e \in F} l(e)$ **Länge** von F
- **Kantenfolge, Weg, Kreis** wie bei gerichtetem Graphen
- Sei $v \in V$, dann heißt $w \in V$ **Nachbar** von v , falls $\{v, w\} \in E$
- $u, v \in V$ heißen **wegzusammenhängend**, falls ein u - v -Weg existiert
Die Relation „wegzusammenhängend“ ist eine Äquivalenzrelation und ihre Äquivalenzklassen heißen **Zusammenhangskomponenten** (siehe Definition 3.1.3)
- G heißt **zusammenhängend**, wenn $\forall s, t \in V$ mit $s \neq t$ ein s - t -Weg existiert, d. h. es gibt nur eine Zusammenhangskomponente
- G heißt **Wald**, wenn G keine Kreise enthält
- G heißt **Baum**, wenn G ein Wald und zusammenhängend ist
- $H = (V, T)$ heißt **Spannbaum** von $G = (V, E)$, falls $T \subseteq E$ und H ein Baum ist

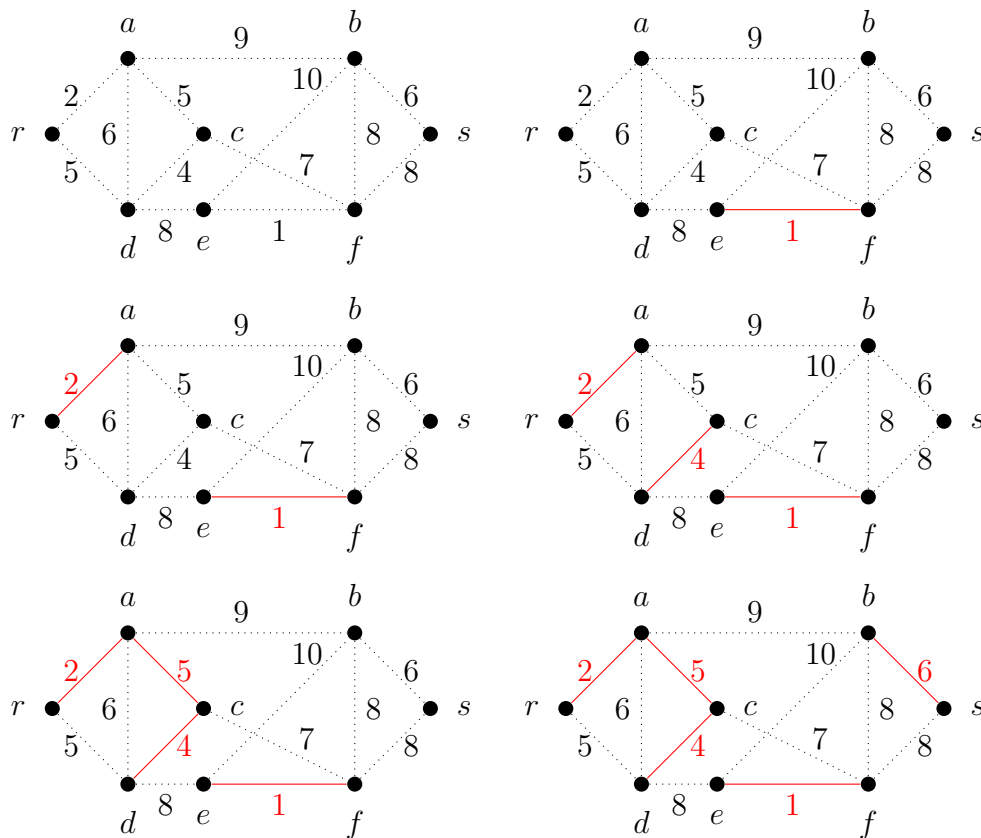
Beispiel 3.1:

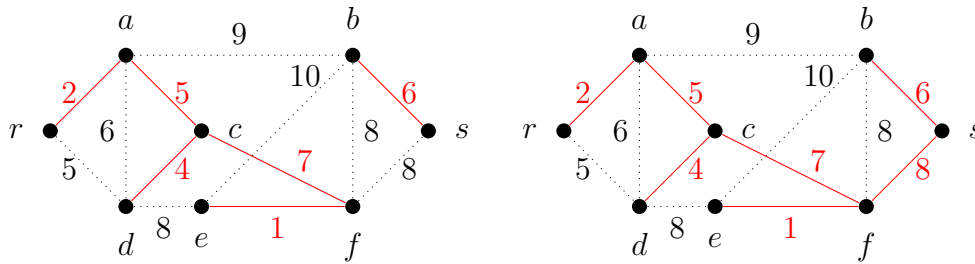




Algorithmus 3.1.2 Algorithmus von Kruskal

- 1: $T = \emptyset$
 - 2: **while** $H = (V, T)$ kein Spannbaum **do**
 - 3: **choose** $e \in E$ **mit** $H = (V, T \cup \{e\})$ ist Wald und $l(e)$ ist minimal mit dieser Eigenschaft
 - 4: $T = T \cup \{e\}$
 - 5: **end while**
-

Beispiel 3.1.2:

**Definition 3.1.2 (guter Wald):**

Sei $G = (V, E)$ ein ungerichteter Graph, $l: E \rightarrow \mathbb{Z}$. Ein Wald (V, F) mit $F \subseteq E$ heißt **gut**, falls es einen minimalen Spannbaum (V, T) von G gibt, mit $F \subseteq T$, d. h. ein guter Wald lässt sich zu einem minimalen Spannbaum ergänzen.

Satz 3.1.1 (Erweiterung guter Wälder):

Sei (V, F) ein guter Wald und sei $e \in E \setminus F$. Dann

$$(V, F \cup \{e\}) \text{ guter Wald} \Leftrightarrow \exists \text{ Schnitt } C \text{ mit } C \cap F = \emptyset, e \in C \text{ und } l(e) = \min_{e' \in C} l(e')$$

Beweis:

„ $\Rightarrow$ “: Übung

„ $\Leftarrow$ “: Sei (V, T) ein minimaler Spannbaum, der (V, F) enthält. Seien $v, w \in V$ die Knoten, die inzident zu e sind, d. h. $e = \{v, w\}$. Sei P der eindeutige Weg in (V, T) , der v mit w verbindet. Sei $U \subseteq V$ mit $\delta(U) = C$. Dann ist $v \in U$, $w \notin U$ oder umgekehrt. Also enthält P eine Kante f mit $f \in C$.

Behauptung:

$(V, T \setminus \{f\} \cup \{e\})$ ist wieder ein Spannbaum.

Beweis:

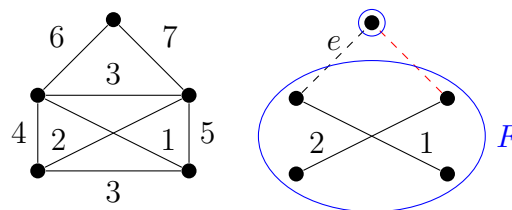
1. Fall: $e = f$

Keine Änderung

2. Fall: $e \neq f$

Das Hinzufügen von e zu P erzeugt einen Kreis. Das löschen von f zerstört diesen Kreis wieder, aber $(V, T \setminus \{f\} \cup \{e\})$ ist zusammenhängend. Nach Voraussetzung ist $l(e) \leq l(f)$, weil $e, f \in C$, also $l(T \setminus \{f\} \cup \{e\}) \leq l(T)$, d. h. $(V, T \setminus \{f\} \cup \{e\})$ ist wieder ein minimaler Spannbaum und $F \cup \{e\} \subseteq T \setminus \{f\} \cup \{e\}$.

□

Beispiel 3.1.3:

Korollar 3.1.1.a:

Der Algorithmus von Prim berechnet einen minimalen Spannbaum.

Beweis:

In jeder Iteration ist (V, T) ein guter Wald und $e \in E$ wird minimal im Schnitt $C = \delta(U)$ gewichtet.

□

Definition 3.1.3 (Zusammenhangskomponente):

Sei $G = (V, E)$ ein ungerichteter Graph. Partitioniere V in Äquivalenzklassen, wobei v und w äquivalent sind, falls es einen v - w -Weg in G gibt. Diese Klassen heißen **Zusammenhangskomponente**.

Korollar 3.1.1.b:

Der Algorithmus von Kruskal berechnet einen minimalen Spannbaum.

Beweis:

In jeder Iteration ist (V, T) ein guter Wald und $e \in E$ wird minimal aus dem Schnitt $C = \delta(K)$ gewählt, wobei K eine Zusammenhangskomponente von (V, T) ist, die zu e inzident ist.

□

Satz 3.1.2 (Geometrische Modellierung minimaler Spannbäume):

Seien $G = (V, E)$, $l: E \rightarrow \mathbb{Z}$ gegeben. Dann ist

$$\text{Die Länge eines minimalen Spannbaum} = \min \sum_{\substack{e \in E \\ X_e \in \{0,1\}}} l(e) X_e$$

$$\sum_{e \in E} X_e = |V| - 1 \quad (3.1)$$

$$\sum_{e \in E[U]} X_e \leq |U| - 1 \quad \forall U \subseteq V, U \neq \emptyset \quad (3.2)$$

wobei $E[U] = \{e \in E: |U \cap e| = 2\}$ ist.

Beweis:

Es gilt:

(V, T) ist Spannbaum $\Leftrightarrow$ Die Variablen X_e mit $X_e = \begin{cases} 1, & e \in T \\ 0, & \text{sonst} \end{cases}$ erfüllen die Bedingungen (3.1) und (3.2).

Grund: (3.1) $G = (V, E)$ ist Baum $\Leftrightarrow G$ ist Wald und $|E| = |V| - 1$

(3.2) G ist Wald $\Leftrightarrow \forall U \subseteq V, U \neq \emptyset: |E[U]| \leq |U| - 1$

Rest: Übung

□

4. Matchings in bipartiten Graphen

25.04.2014
5. Vorlesung

4.1. Matchings mit maximaler Kardinalität	23
4.2. Matchings mit maximalem Gewicht	25
4.3. Das Matchingpolytop	27

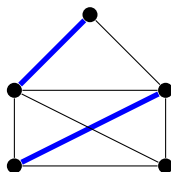
4.1. Matchings mit maximaler Kardinalität

Definition 4.1.1 (Matching):

Sei $G = (V, E)$ ein ungerichteter Graph. Ein **Matching** $M \subseteq E$ in G ist eine Teilmenge disjunkter Kanten, d. h. $\forall e, f \in M$ gilt

$$e \cap f = \emptyset, \text{ falls } e \neq f$$

Beispiel 4.1.1:



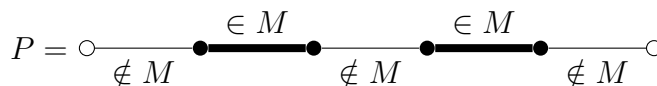
Im Folgenden fassen wir einen Weg $P = (v_0, e_1, v_1, \dots, e_m, v_m)$ als Teilmenge $P = \{e_1, \dots, e_m\} \subseteq E$ auf.

Definition 4.1.2 (M-augmentierend):

Sei $M \subseteq E$ ein Matching in G . Ein Weg $P \subseteq E$ heißt **M-augmentierend**, also **M-verbessernd**, falls

1. P hat ungerade Länge
2. Seine Enden werden nicht durch M überdeckt
3. Seine Kanten sind alternierend in M und nicht in M

Beispiel 4.1.2:



Klar: Falls P M -augmentierend ist, dann ist $M' = M \Delta P$ ein Matching mit Kardinalität $|M'| = |M| + 1$. Hierbei bezeichnet Δ die **symmetrische Differenz** von M und P . Diese ist definiert als: $M \Delta P = (M \setminus P) \cup (P \setminus M)$.

Satz 4.1.1:

Sei $M \subseteq E$ ein Matching in G . Entweder hat M maximale Kardinalität oder es gibt einen M -augmentierenden Weg.

Beweis:

Klar: M hat maximale Kardinalität $\Rightarrow \nexists M$ -augmentierender Weg

Andererseits: Sei M' ein Matching mit $|M'| > |M|$. Betrachte $G' = (V, M \cup M')$ und seine Zusammenhangskomponenten. Da jeder Knoten von G' höchstens zwei Nachbarn hat, bestehen die Zusammenhangskomponenten von G' aus Wegen (eventuell mit Länge 0) oder aus Kreisen. Da $|M'| > |M|$, muss es eine Zusammenhangskomponente von G' geben, die mehr Kanten aus M' als Kanten aus M enthält. Diese Zusammenhangskomponente bildet einen M -augmentierenden Weg.

□

Algorithmus 4.1.1 Bestimmung von Matchings mit maximaler Kardinalität

```

1:  $M = \emptyset$ 
2: while  $\exists M$ -augmentierender Weg  $P$  do
3:    $M \leftarrow M \Delta P$ 
4: end while
    
```

Wie findet man M -augmentierende Wege?

- Einfacher Algorithmus für bipartite Graphen
- Edmonds „Blütenschrumpf“-Algorithmus für allgemeine Graphen

Definition 4.1.3 (chromatische Zahl):

Sei $G = (V, E)$ ein ungerichteter Graph. Eine Menge $U \subseteq V$ heißt **unabhängig**, falls $\forall x, y \in U: \{x, y\} \notin E$. Die **chromatische Zahl** $X(G)$ ist definiert als

$$X(G) = \min\{k \in \mathbb{N}: \exists \text{ Partition } U_1, \dots, U_k \text{ von } V, \text{ wobei } U_1, \dots, U_k \text{ unabhängig}\}$$

Ein Graph heißt **bipartit**, falls $X(G) = 2$.

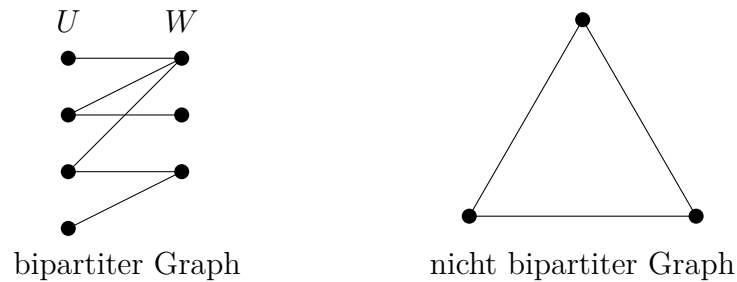
Definition 4.1.4 (bipartiter Graph):

$G = (V, E)$ heißt **bipartit**, falls

$$\exists U, W \subseteq V: V = U \dot{\cup} W$$

und $\forall e \in E: |e \cap U| = |e \cap W| = 1$.

Hierbei bezeichnet $\dot{\cup}$ die **disjunkte Vereinigung** von U und W , d. h. $U \cup W = V$ und $U \cap W = \emptyset$.

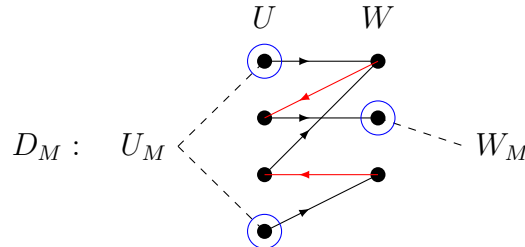
Beispiel 4.1.3:

Definition 4.1.5 (Residualgraph für Matchings):

Sei $G = (V, E)$ bipartit mit Partition $V = U \dot{\cup} W$ und sei $M \subseteq E$ ein Matching in G . Wir definieren den

gerichteten Residualgraph $D_M = (V, A_M)$

$$\text{wobei } A_M = \underbrace{\left\{ (u, w) : e = \{u, w\} \in E \setminus M \right\}}_{\substack{\in U \times W \\ \text{Vorwärtskanten}}} \cup \underbrace{\left\{ (w, u) : e = \{u, w\} \in E \cap M \right\}}_{\substack{\in W \times U \\ \text{Rückwärtskanten}}}.$$

Seien $U_M \subseteq U$, $W_M \subseteq W$ die Knoten, die nicht von M überdeckt werden. Dann ist jeder gerichtete Weg von einem Knoten in U_M zu einem Knoten in W_M ein M -augmentierender Weg und umgekehrt.

Beispiel 4.1.4:

Algorithmische Umsetzung

Finde je zwei Knoten in U_M und W_M mit dem Bellman-Ford-Algorithmus, wobei $l: A_M \rightarrow \mathbb{Z}$, $l(a) = 1$ gesetzt wird. [Dies ist eine einfache, aber auch suboptimale Strategie, besser wäre z. B. die Verwendung der Tiefensuche]

4.2. Matchings mit maximalem Gewicht

30.04.2014
6. Vorlesung

Sei $G = (V, E)$ ein ungerichteter Graph und $w: E \rightarrow \mathbb{R}$ eine Gewichtsfunktion. Für $M \subseteq E$ definieren wir $w(M) = \sum_{e \in M} w(e)$.

Gesucht: $\nu_w(G) = \max\{w(M) : M \subseteq E \text{ Matching in } G\}$.

Definition 4.2.1 (extremes Matching):

Ein Matching $M \subseteq E$ heißt **extrem**, falls $\forall M' \subseteq E$ Matching gilt

$$|M'| = |M| \Rightarrow w(M') \leq w(M)$$

Definition 4.2.2 (Längenfunktion):

Sei $M \subseteq E$ ein Matching in G . Wir definieren die **Längenfunktion** $l: E \rightarrow \mathbb{R}$ durch

$$l(e) = \begin{cases} w(e), & e \in M \\ -w(e), & e \notin M \end{cases}$$

Für $P \subseteq E$ definiere $l(P) = \sum_{e \in P} l(e)$.

Satz 4.2.1:

Sei $M \subseteq E$ ein extremes Matching und sei $P \subseteq E$ ein M -augmentierender Weg mit minimaler Länge bezüglich l . Dann ist $M' = M \Delta P$ ebenfalls extrem.

Beweis:

Sei $N \subseteq E$ ein extremes Matching mit $|N| = |M| + 1$. Da $|N| > |M|$ enthält der Graph $(V, M \cup N)$ eine Zusammenhangskomponente Q die M -augmentierender Weg ist. Dann ist $l(Q) \geq l(P)$. Es ist $N \Delta Q$ ein Matching mit Kardinalität $|N \Delta Q| = |M|$, also ist $w(N \Delta Q) \leq w(M)$. Zusammen ergibt sich

$$\begin{aligned} w(N) &= w(N \Delta Q) - \underbrace{l(Q)}_{\geq l(P)} \\ &\leq w(M) - l(P) \\ &= w(M') \end{aligned}$$

Das heißt $w(N) \underset{\substack{\text{weil } N \\ \text{extrem}}}{=} w(M')$, d. h. M' extrem.

□

Algorithmus 4.2.1 „Ungarische Methode“ zur Bestimmung gewichteter Matchings mit maximaler Kardinalität, 1931

Eingabe: G bipartit, $w: E \rightarrow \mathbb{R}$

1: $M_0 = \emptyset$, $k = 1$

2: **while** $\exists M_{k-1}$ -augmentierender Weg **do**

3: **choose** M_{k-1} -augmentierender Weg P **mit** minimaler Länge

4: $M_k \leftarrow M_{k-1} \Delta P$

5: $k \leftarrow k + 1$

6: **end while**

Ausgabe: $\max\{w(M_i) : i = 0, \dots, k - 1\}$

Wie findet man M -augmentierende Wege mit minimaler Länge in bipartiten Graphen?

Betrachte dazu wieder den Residualgraph $D_M = (V, A_M)$ mit

$$A_M = \left\{ \underbrace{(u, w)}_{\in U \times W} : e = \{u, w\} \in E \setminus M \right\} \cup \left\{ \underbrace{(w, u)}_{\in W \times U} : e = \{u, w\} \in E \cap M \right\}$$

und der Längenfunktion $l: A_M \rightarrow \mathbb{R}$ mit $l((a, b)) = l(\{a, b\})$.

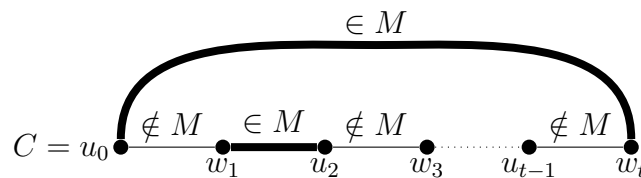
- Finde kürzesten Weg von U_M nach W_M im Graphen von D_M (vgl. Beispiel 4.1.4)
- Bellman-Ford-Algorithmus (Algorithmus 2.2.1) ist wegen Satz 4.2.2 anwendbar

Satz 4.2.2:

Sei $M \subseteq E$ ein extremes Matching im bipartiten Graph $G = (V, E)$. Dann enthält D_M keinen gerichteten Kreis mit negativer Länge.

Beweis:

Angenommen $C \subseteq E$ ist ein gerichteter Kreis in D_M mit $l(C) < 0$.



Dann ist $M' = M \Delta C$ ein Matching mit $|M'| = |M|$. Es gilt $w(M') = w(M) - l(C) > w(M)$.
 $\nrightarrow$ zur Extremalität von M .

□

4.3. Das Matchingpolytop

Achtung:

Dieser Abschnitt verwendet Erkenntnisse aus der Polyedertheorie, machen Sie sich also vorher mit Kapitel 6 vertraut.

Definition 4.3.1 (Inzidenzvektor):

Der **Inzidenzvektor** von M ist definiert als $\chi^M: E \rightarrow \mathbb{R}$

$$\chi^M(e) = \begin{cases} 1, & e \in M \\ 0, & \text{sonst} \end{cases}$$

Notation:

- $\mathbb{R}^E = \{f: E \rightarrow \mathbb{R}\}$ Vektorraum
- $\mathbb{R}^V = \{g: V \rightarrow \mathbb{R}\}$ Vektorraum

Definition 4.3.2 (Matchingpolytop):

Das **Matchingpolytop** von einem Graphen $G = (V, E)$ ist

$$P_{\text{matching}}(G) = \text{conv}\{\chi^M: M \text{ Matching}\} \subseteq \mathbb{R}^E$$

Sei $w: E \rightarrow \mathbb{R}$ und $w \in \mathbb{R}^E$.

$\max w^\top x$, wobei $x \in P_{\text{matching}}(G)$ = Gewicht eines maximalen Matchings in G .

Das Matching Polytop ist ein Polyeder.

Ziel: Beschreibe P_{matching} durch lineare Ungleichungen.

Zuerst: Beschreibe $P_{\text{perfect matching}}$ durch lineare Ungleichungen.

Definition 4.3.3 (perfektes Matching):

Sei $M \subseteq E$ ein Matching. M heißt **perfekt**, falls $V = \bigcup_{e \in M} e$.

$$P_{\text{perfect matching}}(G) = \text{conv}\{\chi^M : M \text{ perfektes Matching}\}$$

Theorem 4.3.1 (Theorem von Birkhoff, 1946):

Sei $G = (V, E)$ bipartit. Dann

$$P_{\text{perfect matching}}(G) = \left\{ x \in \mathbb{R}^E : x_e \geq 0 \quad \forall e \in E, \sum_{e: e \ni v} x_e = 1 \quad \forall v \in V \right\}$$

Beweis:

„ $\subseteq$ “: Klar, weil die Inzidenzvektoren χ^M die Bedingung erfüllen.

„ $\supseteq$ “: Die rechte Seite ist ein beschränkter Polyeder, also ein Polytop. Daher reicht es zu zeigen, dass jede Ecke von der Form χ^M ist, wobei M perfektes Matching. Sei $x \in \mathbb{R}^E$ eine Ecke. Sei $F = \{e \in E : x_e > 0\}$.

Angenommen F enthält einen Kreis C . Weil G bipartit ist, ist C gerade. Des weiteren gilt $C = M \dot{\cup} N$ für disjunkte Matchings M, N . Für genügend kleine $\varepsilon > 0$ sind $x' = x + \varepsilon(\chi^M - \chi^N)$ und $x'' = x - \varepsilon(\chi^M - \chi^N)$ Elemente der rechten Seite. Dann gilt $x = \frac{1}{2}x' + \frac{1}{2}x''$, d. h. x ist keine Ecke $\nmid \Rightarrow (V, F)$ ist Wald.

Behauptung:

$$x \in \{0, 1\}^E$$

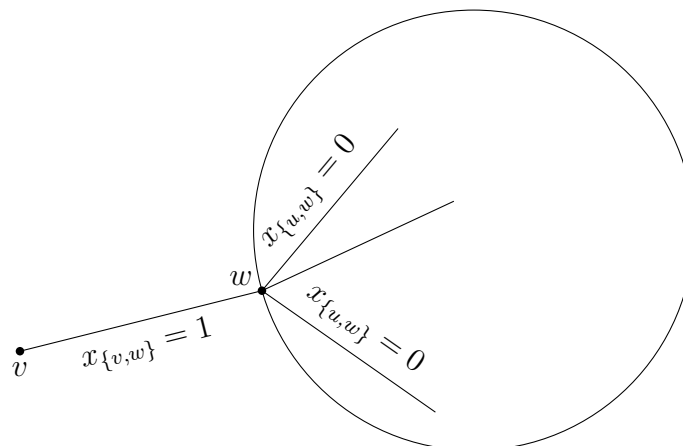
Beweis (per Induktion über $|E|$):

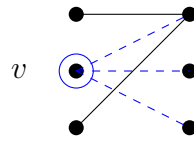
1. Fall: $|E| = 1$

Klar, weil $x_e = 1$

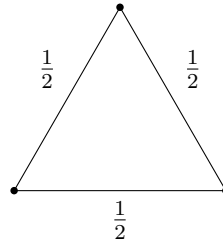
2. Fall: $|E| > 1$

Weil F Wald, gibt es Knoten $v \in V$, die nur zu einer Kante $\{v, w\} \in F$ gehört. Dann $x_{\{v, w\}} = 1$ und $x_h = 0$ für alle $h \in E \setminus \{\{v, w\}\}$, wobei $H = \{\{u, v\} \in E : u \in V\}$. Nun wende die Induktionsvoraussetzung auf $(V, E \setminus H)$ an. Jedes $x \in \{0, 1\}^E$, das zur rechten Seite gehört, ist Inzidenzvektor eines perfekten Matchings.



Beispiel 4.3.1:

Bemerkung 4.3.1:

Wie das Beispiel unten zeigt, ist [Theorem 4.3.1](#) (Theorem von Birkhoff) im Allgemeinen, für nicht bipartite Graphen, falsch.


 02.05.2014
7. Vorlesung

Satz 4.3.1:

$$P_{\text{matching}}(G) = \text{conv}\{\chi^M : M \subseteq E \text{ Matching}\}$$

$$= \left\{ x \in \mathbb{R}^E : x_e \geq 0 \quad \forall e \in E \quad \sum_{e: e \ni v} x_e \leq 1 \quad \forall v \in V \right\}$$

Beweis:

Übung.

Das bedeutet, dass das maximale Gewicht eines Matchings im bipartiten Graph $G = (V, E)$ mit der Gewichtsfunktion $w \in \mathbb{R}^E$ durch das lineare Programm

$$\begin{aligned} \max \quad & w^\top x \\ & x \in \mathbb{R}^E \\ & x_e \geq 0 \quad \forall e \in E \\ & \sum_{e: e \ni v} x_e \leq 1 \quad \forall v \in V \end{aligned}$$

bestimmt werden kann.

Man kann also den Simplexalgorithmus benutzen, z. B. mit Startecke $\chi^{\{e\}}$ für $e \in E$.

Anwendung des Dualitätstheorems liefert den [Satz 4.3.2](#).

Satz 4.3.2 (Satz von Egeváry):

Sei $G = (V, E)$ ein bipartiter Graph und $w: E \rightarrow \mathbb{R}$ eine Gewichtsfunktion. Das maximale Gewicht eines Matchings in G ist gleich

$$\begin{aligned} \min \quad & \sum_{v \in V} y_v \\ & y \in \mathbb{R}^V \\ & y_v \geq 0 \text{ für } v \in V \\ & y_u + y_v \geq w(e) \text{ für } e = \{u, v\} \in E \end{aligned}$$

Beweis:

Sei $A \in \mathbb{R}^{V \times E}$ mit $A_{v,e} = \begin{cases} 1, & \text{falls } v \in E \\ 0, & \text{sonst} \end{cases}$ und $b \in \mathbb{R}^V$ mit $b_v = 1$.

Das maximale Gewicht eines Matchings ist

$$\begin{aligned} \max\{w^\top \chi^M : M \text{ Matching in } G\} &= \max\{w^\top x : x \in P_{\text{matching}}\} \\ &= \max\{w^\top x : x \in \mathbb{R}^E, x \geq 0, Ax \leq b\} \\ &\stackrel{\text{Theorem 6.7.1}}{=} \min\{y^\top b : y \in \mathbb{R}^V, y \geq 0, y^\top A \geq w\} \\ &= \min \begin{aligned} &\sum_{v \in V} y_v \\ &y \in \mathbb{R}^V, y_v \geq 0 \quad \forall v \in V \\ &y_u + y_v \geq w(\{u, v\}) \quad \forall \{u, v\} \in E \end{aligned} \end{aligned}$$

□

Satz 4.3.3:

Sei $G = (V, E)$ bipartit, $w: E \rightarrow \mathbb{Z}$. Dann gibt es eine optimale Lösung $y \in \mathbb{Z}^V$.

Beweis:

Angenommen $y \in \mathbb{R}$ ist optimale Lösung und y ist so gewählt, dass die Anzahl der y_v mit $y_v \notin \mathbb{Z}$ minimal ist. Setze $X = \{v \in V : y_v \notin \mathbb{Z}\}$.

Angenommen $X \neq \emptyset$. Dann ist Ohne Einschränkung $|X \cap U| \geq |X \cap W|$. Sei $u \in X \cap U$ mit $\varepsilon = y_u - \lfloor y_u \rfloor$ so klein wie möglich. Definiere $\tilde{y} \in \mathbb{R}^V$ durch

$$\tilde{y}_v = \begin{cases} y_v - \varepsilon, & v \in X \cap U \\ y_v + \varepsilon, & v \in X \cap W \\ y_v, & \text{sonst} \end{cases}$$

Dann ist $\tilde{y}$ ebenfalls eine gültige Lösung, wegen der Ganzzahligkeit von W . Außerdem

$$\begin{aligned} \sum_{v \in V} \tilde{y}_v &= \sum_{v \in V} y_v - \varepsilon |X \cap U| + \varepsilon |X \cap W| \\ &\leq \sum_{v \in V} y_v \end{aligned}$$

Das bedeutet $\tilde{y}$ ist optimal, da $\tilde{y}_u \in \mathbb{Z}$ weniger nicht ganzzahlige Einträge hat als $\tilde{y}$. $\nmid$

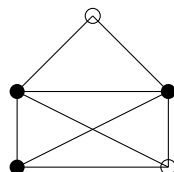
□

Definition 4.3.4 (Knotenüberdeckung):

Sei $G = (V, E)$ ein ungerichteter Graph. Eine Teilmenge $U \subseteq V$ heißt **Knotenüberdeckung**, falls

$$\forall e \in E: |e \cap U| \geq 1$$

Beispiel 4.3.2:



Korollar 4.3.3.a (Matchingtheorem von König, 1931):

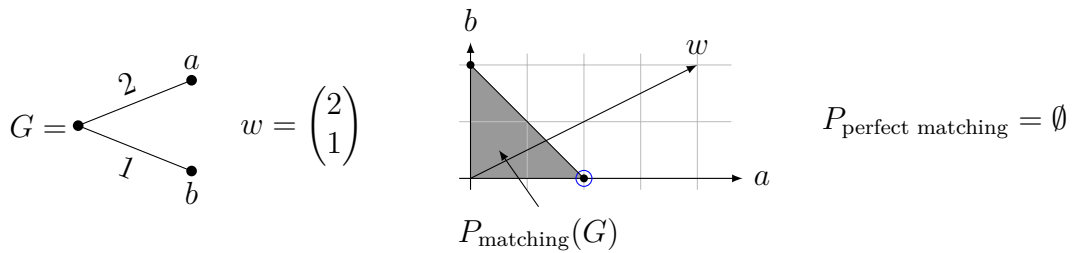
Sei $G = (V, E)$ bipartit. Dann gilt

$$\begin{aligned}\nu(G) &= \max\{|M| : M \subseteq E \text{ Matching in } G\} \\ &= \min\{|U| : U \subseteq V \text{ Knotenüberdeckung von } G\} \\ &= \tau(G)\end{aligned}$$

Beweis:

Wende die [Sätze 4.3.2 und 4.3.3](#) an und wähle jeweils $w = 1$. Die optimale Lösung $y \in \mathbb{Z}^V$ gibt eine Knotenüberdeckung.

□

Beispiel 4.3.3:


$$\begin{aligned}P_{\text{matching}}(G) &= \text{conv} \left\{ \chi^\emptyset, \chi^a, \chi^b \right\} \subseteq \mathbb{R}^E \\ &= \text{conv} \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}\end{aligned}$$

5. Netzwerkflüsse

07.05.2014
8. Vorlesung

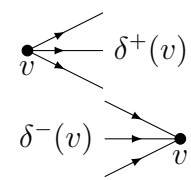
5.1. Das Max-Flow-Min-Cut-Theorem	32
5.2. Zirkulationen	37
5.3. Flüsse mit minimalen Kosten	39

5.1. Das Max-Flow-Min-Cut-Theorem

Sei $D = (V, A)$ ein gerichteter Graph (**Netzwerk**) mit $s, t \in V$. Der Knoten s heißt **Quelle** (engl. *source*) und t heißt **Senke** (engl. *terminal*).

Zur Erinnerung:

$$\delta^+(v) = \delta^+(\{v\}) = \{(v, w) \in A : v \in \{v\}, w \notin \{v\}\}$$

$$\delta^-(v) = \delta^-(\{v\}) = \{(v, w) \in A : v \notin \{v\}, w \in \{v\}\}$$


Definition 5.1.1 (*s-t*-Fluss):

Eine Funktion $f: A \rightarrow \mathbb{R}_{\geq 0}$ heißt ***s-t*-Fluss**, falls

$$\forall v \in V \setminus \{s, t\}: \sum_{a \in \delta^-(v)} f(a) = \sum_{a \in \delta^+(v)} f(a) \quad (\text{Flusserhaltung})$$

Definition 5.1.2 (Wert):

Der **Wert** eines *s-t*-Flusses f ist

$$\text{value}(f) = \sum_{a \in \delta^+(s)} f(a) - \sum_{a \in \delta^-(s)} f(a)$$

Definition 5.1.3 (Kapazitätsfunktion):

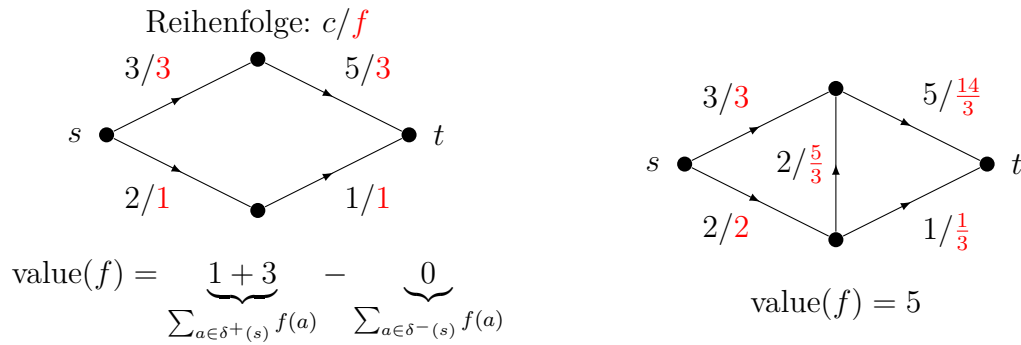
Ein *s-t*-Fluss $f: A \rightarrow \mathbb{R}_{\geq 0}$ heißt durch die **Kapazitätsfunktion** $c: A \rightarrow \mathbb{R}_{\geq 0}$ **beschränkt**, falls

$$\forall a \in A: f(a) \leq c(a)$$

Notation: $f \leq c$

Definition 5.1.4 (maximaler *s-t*-Fluss):

Ein **maximaler *s-t*-Fluss** ist ein *s-t*-Fluss f , der durch c beschränkt ist und maximalen Wert hat.

Beispiel 5.1.1:

Ziel: Finde einen maximalen s - t -Fluss

$$\begin{aligned}
 &\max \text{ value } f \\
 &f: A \rightarrow \mathbb{R}_{\geq 0} \\
 &f \text{ } s\text{-}t\text{-Fluss} \\
 &f(a) \leq c(a) \quad \forall a \in A
 \end{aligned}$$

Lemma 5.1.1:

Sei $D = (V, A)$ ein gerichteter Graph, $s, t \in V$ und $c: A \rightarrow \mathbb{R}_{\geq 0}$. Weiterhin sei f ein s - t -Fluss, der durch c beschränkt ist und $U \subseteq V$ mit $s \in U$, $t \notin U$, so dass $\delta^+(U)$ ein s - t -Schnitt ist. Dann gilt

$$\text{value}(f) \leq c(\delta^+(U)) = \sum_{a \in \delta^+(U)} c(a)$$

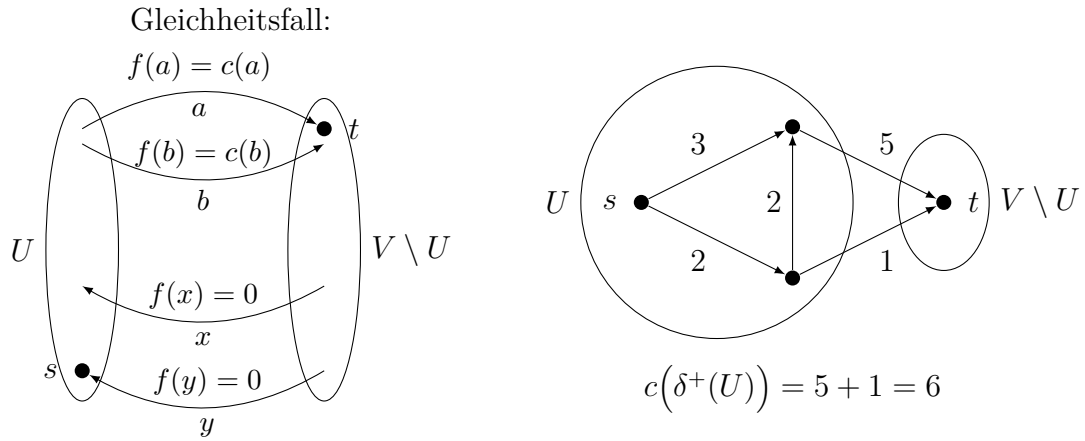
Dabei gilt Gleichheit genau dann, wenn

$$\begin{aligned}
 f(a) &= c(a) & \forall a \in \delta^+(U) \\
 f(a) &= 0 & \forall a \in \delta^-(U)
 \end{aligned}$$

Beweis:

$$\begin{aligned}
 \text{value}(f) &= \sum_{a \in \delta^+(s)} f(a) - \sum_{a \in \delta^-(s)} f(a) \\
 &\stackrel{\text{Fluss-erhaltung}}{=} \sum_{v \in U} \left(\sum_{a \in \delta^+(v)} f(a) - \sum_{a \in \delta^-(v)} f(a) \right) \\
 &\stackrel{\text{Fluss-erhaltung}}{=} \sum_{a \in \delta^+(U)} f(a) - \sum_{a \in \delta^-(U)} f(a) \\
 &\leq \sum_{a \in \delta^+(U)} c(a) - 0 \\
 &= c(\delta^+(U))
 \end{aligned}$$

□

Beispiel 5.1.2:

Definition 5.1.5 (Residualgraph für Flüsse):

Sei $D = (V, A)$ ein gerichteter Graph und $a = (u, v) \in A$. Definiere die Kante $a^{-1} = (v, u)$ und $A^{-1} = \{a^{-1} : a \in A\}$. Sei f ein s - t -Fluss und c eine Kapazitätsfunktion mit $f \leq c$. Dann ist

$D_f = (V, A_f)$ der **Residualgraph**

wobei $A_f = \underbrace{\{a \in A : f(a) < c(a)\}}_{\text{Vorwärtskanten}} \cup \underbrace{\{a^{-1} \in A^{-1} : f(a) > 0\}}_{\text{Rückwärtskanten}}.$

Lemma 5.1.2:

Sei f ein s - t -Fluss in D , der durch c beschränkt ist. Angenommen D_f enthält keinen gerichteten s - t -Weg. Seien $U \subseteq V$ die Knoten, die in D_f von s aus erreichbar sind. Dann gilt

$$\text{value}(f) = c(\delta^+(U))$$

Insbesondere ist der Wert von f maximal.

Beweis:

Für $a \in \delta^+(U)$ gilt $a \notin A_f$, d. h. $f(a) = c(a)$. Für $a \in \delta^-(U)$ gilt $a^{-1} \notin A_f$, d. h. $f(a) = 0$. Damit gilt $\text{value}(f) = c(\delta^+(U))$ und nach **Lemma 5.1.1** ist f maximaler s - t -Fluss. $\square$

Notation:

Sei P ein gerichteter Weg in D_f . Definiere $\chi^P \in \mathbb{R}^A$ durch

$$\chi^P(a) = \begin{cases} 1, & \text{falls } P \text{ die gerichtete Kante } a \text{ durchläuft} \\ -1, & \text{falls } P \text{ die gerichtete Kante } a^{-1} \text{ durchläuft} \\ 0, & \text{sonst} \end{cases}$$

Theorem 5.1.1 (max-Flow=min-Cut-Theorem, Ford-Fulkerson, 1954):

Sei $D = (V, A)$ ein gerichteter Graph, $s, t \in V$ und $c: A \rightarrow \mathbb{R}_{\geq 0}$. Dann ist der maximale Wert eines s - t -Flusses, der durch c beschränkt ist, gleich der minimalen Kapazität eines s - t -Schnittes.

$$\max_{\substack{f \text{ s-t-Fluss} \\ 0 \leq f \leq c}} \text{value}(f) = \min_{\substack{U \subseteq V \\ s \in U, t \notin U}} c(\delta^+(U))$$

Beweis:

„ $\max \leq \min$ “: Siehe [Lemma 5.1.1](#)

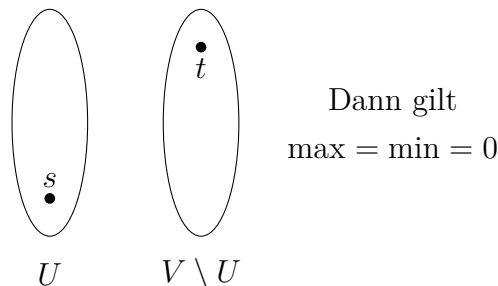
„ $\max \geq \min$ “: Sei f ein maximaler s - t -Fluss der durch c beschränkt ist. Es genügt zu zeigen, dass es einen s - t -Schnitt $\delta^+(U)$ gibt mit $\text{value}(f) = c(\delta^+(U))$ für ein $U \subseteq V$. Betrachte dazu den Residualgraphen D_f .

1. Fall: Falls es keinen s - t -Weg in D_f gibt, dann sind wir fertig nach [Lemma 5.1.2](#).
2. Fall: Angenommen es gibt einen s - t -Weg P in D_f . Dann ist $f' = f + \varepsilon \chi^P$ für genügend kleines $\varepsilon > 0$ ebenfalls ein s - t -Fluss (weil χ^P s - t -Fluss), der durch c beschränkt ist. Dann ist $\text{value}(f') = \text{value}(f) + \varepsilon$.
 $\nmid$ zur Maximalität von $\text{value}(f)$.

□

Bemerkung 5.1.1:

Angenommen in D gibt es keinen gerichteten Weg zwischen s und t .

**Folgerung 5.1.1:**

Falls $c: A \rightarrow \mathbb{Z}$ ganzzahlig ist, dann gibt es auch einen ganzzahligen maximalen s - t -Fluss $f: A \rightarrow \mathbb{Z}$.

Beweis:

Wähle $\varepsilon = 1$ im Beweis von [Theorem 5.1.1](#).

Algorithmus 5.1.1 Ford-Fulkerson-Algorithmus zur Berechnung maximaler Flüsse

Eingabe: $D = (V, A)$, $s, t \in V$, $c: A \rightarrow \mathbb{R}_{\geq 0}$

- 1: $f = 0$
- 2: **while** $\exists$ gerichteter s - t -Weg P in Residualgraph D_f **do**
- 3: $f = f + \varepsilon \chi^P$ ▷ wobei $\varepsilon > 0$ maximal gewählt ist, so dass $0 \leq f + \varepsilon \chi^P \leq c$
- 4: **end while**

Ausgabe: s - t -Fluss $f: A \rightarrow \mathbb{R}_{\geq 0}$, welcher maximal ist für $f \leq c$

09.05.2014
9. Vorlesung

Satz 5.1.1:

Falls $c(a) \in \mathbb{Q}$ für alle $a \in A$, dann terminiert der Algorithmus 5.1.1 (Ford-Fulkerson) in endlich vielen Schritten. Sonst im Allgemeinen nicht und es können ∞ -viele Schritte erfolgen.

Beweis:

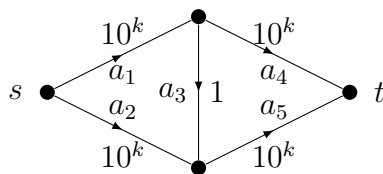
$\exists K \in \mathbb{N}: K \cdot c(a) \in \mathbb{N} \forall a \in A$, d. h. in jeder Iteration ist ε ein Vielfaches von $\frac{1}{K}$ und der Wert des Flusses vergrößert sich mindestens um $\frac{1}{K}$. Da der Wert $c(\delta^+(s))$ nicht überschritten werden kann, terminiert der Algorithmus nach höchstens $K \cdot c(\delta^+(s))$ Schritten.

□

Bemerkung 5.1.2:

Es gibt Beispiele mit $c(a) \in \mathbb{R}$, so dass der Algorithmus 5.1.1 nicht terminiert. [siehe 1, Kapitel 10.4a]

Verbesserung des Ford-Fulkerson-Algorithmus



Offensichtlich ist
 $f = (10^k, 10^k, 0, 10^k, 10^k)$
 ein maximaler s - t -Fluss mit
 $\text{value}(f) = 2 \cdot 10^k$

Eine ungünstige Wahl im Ford-Fulkerson-Algorithmus kann dazu führen, dass der Fluss in jeder Iteration nur um den Wert 1 erhöht wird $\rightarrow 2 \cdot 10^k$ Iterationen.

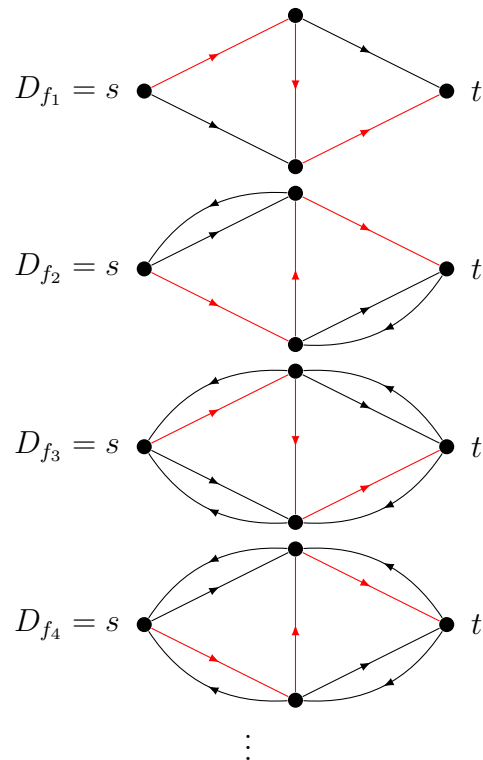
$$f_1 = (0, 0, 0, 0, 0)$$

$$f_2 = (1, 0, 1, 0, 1)$$

$$f_3 = (1, 1, 0, 1, 1)$$

$$f_4 = (2, 1, 1, 1, 2)$$

$\vdots$



Problem: Die gewählten s - t -Wege in D_{f_k} waren nicht die kürzesten.

Lösung: Dinitz (1970), Edmonds und Karp (1972)

Falls ein s - t -Weg mit minimaler Länge gewählt wird, dann kann diese „ungünstige Wahl“ nicht passieren. [siehe 2, Kapitel 4.4]

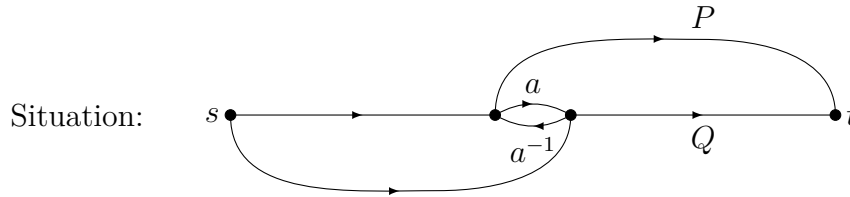
Lemma 5.1.3:

$D = (V, A)$, $s, t \in V$, $\mu(D) := \text{dist}(s, t)$. Sei $\alpha(D) \subseteq A$ Menge von Kanten, die in einem kürzesten s - t -Weg vorkommen. Definiere $D' = (V, A \cup \alpha(D)^{-1})$. Dann gilt:

$$\mu(D') = \mu(D) \text{ und } \alpha(D') = \alpha(D)$$

Beweis:

Es genügt zu zeigen, dass $\mu(D)$ und $\alpha(D)$ sich nicht verändern, wenn wir eine Kante $a^{-1} \in \alpha(D)^{-1}$ zu D hinzufügen. Angenommen dies ist nicht so, dann gibt es einen s - t -Weg $P \subseteq A \cup \{a^{-1}\}$, der durch a^{-1} geht und eine Länge von höchstens $\mu(D)$ hat. Da $a \in \alpha(D)$ gibt es einen s - t -Weg $Q \subseteq A$, der durch a geht und Länge $\mu(D)$ hat.



Dann enthält $P \cup Q \setminus \{a, a^{-1}\}$ einen s - t -Weg der Länge kleiner $\mu(D)$. $\nexists$

□

Satz 5.1.2:

Falls in jeder Iteration des Ford-Fulkerson-Algorithmus ein kürzester s - t -Weg in D_f gewählt wird, dann ist die Anzahl der Iterationen kleiner gleich $|V| \cdot |A|$. Dies ist unabhängig von der Kapazitätsfunktion, d. h. es funktioniert für ein beliebiges c .

Beweis:

Falls wir einen Fluss f entlang eines kürzesten s - t -Weges P in D_f verbessern und einen neuen Fluss f' erhalten, dann ist die Kantenmenge von $D_{f'}$ in der Kantenmenge von $D' = (V, A_f \cup \alpha(D_f)^{-1})$ enthalten. Nach Lemma 5.1.3 gilt $\mu(D_f) = \mu(D')$ und $\alpha(D_f) = \alpha(D')$. Weiterhin gilt $\mu(D_{f'}) \geq \mu(D')$.

Falls $\mu(D_{f'}) = \mu(D_f)$, dann $\alpha(D_{f'}) \subseteq \alpha(D') = \alpha(D_f)$. Da wenigstens eine Kante aus P zu D_f aber nicht zu $D_{f'}$ gehört, haben wir strikte Inklusion $\alpha(D_{f'}) \subset \alpha(D_f)$. Diese Situation, $\mu(D_{f'}) = \mu(D_f)$, kann maximal $|A|$ -mal vorkommen, danach muss $\mu(D_{f'}) > \mu(D_f)$ sein. Da $\mu(D_f)$ sich höchstens $|V|$ -mal vergrößern kann, folgt die Behauptung.

□

5.2. Zirkulationen

Definition 5.2.1 (Zirkulation):

Sei $D = (V, A)$ ein gerichteter Graph, $f: A \rightarrow \mathbb{R}$ heißt **Zirkulation**, falls Flusserhaltung in jedem Knoten gilt

$$\forall v \in V: \sum_{a \in \delta^+(v)} f(a) = \sum_{a \in \delta^-(v)} f(a)$$

Satz 5.2.1 (Hoffman, 1960):

Sei $D = (V, A)$ ein gerichteter Graph und $d, c: A \rightarrow \mathbb{R}$ mit $d(a) \leq c(a) \forall a \in A$. Es gibt eine Zirkulation $f: A \rightarrow \mathbb{R}$ mit

$$\forall a \in A: d(a) \leq f(a) \leq c(a) \quad (5.1)$$

$$\Leftrightarrow \forall U \subseteq V: \sum_{a \in \delta^-(U)} d(a) \leq \sum_{a \in \delta^+(U)} c(a) \quad (5.2)$$

Beweis:

„ $\Rightarrow$ “: Angenommen f ist Zirkulation, die (5.1) erfüllt. Dann

$$\sum_{a \in \delta^-(U)} d(a) \leq \sum_{a \in \delta^-(U)} f(a) = \sum_{a \in \delta^+(U)} f(a) \leq \sum_{a \in \delta^+(U)} c(a)$$

„ $\Leftarrow$ “: Für $f: A \rightarrow \mathbb{R}$ definiere $\text{loss}_f: V \rightarrow \mathbb{R}$ durch

$$\text{loss}_f(v) = \sum_{a \in \delta^+(v)} f(a) - \sum_{a \in \delta^-(v)} f(a)$$

Sei f so gewählt, dass $d \leq f \leq c$ gilt und die 1-Norm $\|\text{loss}_f\|_1$ minimal ist, wobei

$$\|\text{loss}_f\|_1 = \sum_{v \in V} |\text{loss}_f(v)|$$

Definiere $S = \{v \in V: \text{loss}_f(v) < 0\}$ und $T = \{v \in V: \text{loss}_f(v) > 0\}$. Falls $S = \emptyset$, dann ist auch $T = \emptyset$ und f ist eine Zirkulation. ✓

Falls $S \neq \emptyset$, betrachte den Graphen $D_f = (V, A_f)$, wobei

$A_f = \{a \in A: f(a) < c(a)\} \cup \{a^{-1} \in A^{-1}: f(a) > d(a)\}$. Falls D_f einen Weg von S nach T enthält, können wir f verändern, um $\|\text{loss}_f\|_1$ zu reduzieren. ✗

Es gibt also keinen Weg von S nach T in D_f . Sei $U \subseteq V$ die Knotenmenge, die von S aus erreichbar ist. Für $a \in \delta^+(U)$ gilt $a \notin A_f$, also $f(a) = c(a)$. Für $a \in \delta^-(U)$ gilt $a^{-1} \notin A_f$, also $f(a) = d(a)$. Zusammen

$$\begin{aligned} \sum_{a \in \delta^+(U)} c(a) - \sum_{a \in \delta^-(U)} d(a) &= \sum_{a \in \delta^+(U)} f(a) - \sum_{a \in \delta^-(U)} f(a) \\ &= \sum_{v \in U} \text{loss}_f(v) \\ &= \sum_{v \in S} \text{loss}_f(v) \\ &< 0 \quad \text{✗} \end{aligned}$$

□

Korollar 5.2.1.a:

Falls c und d ganzzahlig und falls es eine Zirkulation f mit $d \leq f \leq c$ gibt, dann gibt es auch eine ganzzahlige Zirkulation f' mit $d \leq f' \leq c$.

Satz 5.2.2:

Sei $D = (V, A)$ ein gerichteter Graph, $f: A \rightarrow \mathbb{R}_{\geq 0}$. Sei $\mathcal{C}$ die Menge der gerichteten Kreise in D . Für $C \in \mathcal{C}$ definiere

$$\chi^C: A \rightarrow \{0, 1\} \text{ durch } \chi^C(a) = \begin{cases} 1, & \text{falls } a \text{ in } C \text{ durchlaufen wird} \\ 0, & \text{sonst} \end{cases}$$

Dann gilt:

$$f \text{ ist eine Zirkulation} \Leftrightarrow f \in \text{cone}\{\chi^C: C \in \mathcal{C}\}$$

$$\text{wobei } \text{cone}\{\chi^C: C \in \mathcal{C}\} = \left\{ \sum_{C \in \mathcal{C}} \alpha_C \chi^C: \alpha_C \geq 0 \right\}.$$

Beweis:

„ $\Rightarrow$ “: ✓

„ $\Leftarrow$ “: Sei $A' = \{a \in A: f(a) > 0\}$. Per Induktion über $|A'|$

1. Fall: $|A'| = 0$:

Dann ist $f = 0$ und die Aussage ist trivial.

2. Fall: $|A'| > 0$:

Da f eine Zirkulation ist, muss A' einen gerichteten Kreis C enthalten. Sei $\varepsilon = \min\{f(a): a \in C\}$. Setze $f' = f - \varepsilon \chi^C$. Dann ist $f': A \rightarrow \mathbb{R}_{\geq 0}$ eine Zirkulation und wir können die Induktionsvoraussetzung anwenden.

□

5.3. Flüsse mit minimalen Kosten

14.05.2014
10. Vorlesung

Gegeben: $D = (V, A)$ gerichteter Graph, $s, t \in V$, $c: A \rightarrow \mathbb{R}_{\geq 0}$ Kapazitätsfunktion, $k: A \rightarrow \mathbb{R}_{\geq 0}$ Kostenfunktion.

Definition 5.3.1 (Kosten):

Die **Kosten** von $f: A \rightarrow \mathbb{R}_{\geq 0}$ sind definiert als

$$\text{cost}(f) = \sum_{a \in A} k(a)f(a)$$

Gesucht: Maximaler s - t -Fluss f der durch c beschränkt ist und gleichzeitig minimale Kosten hat (minimal unter allen maximalen Flüssen, die durch c beschränkt sind).

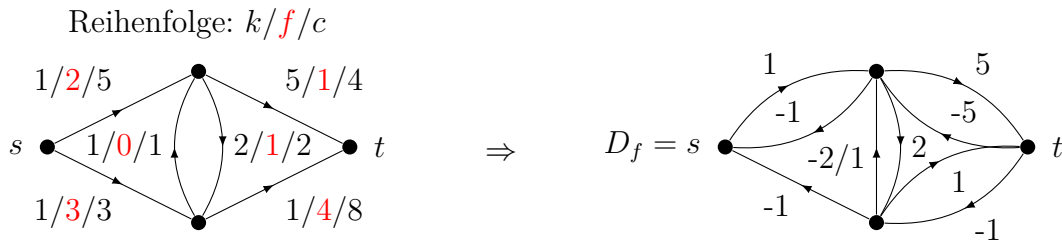
Definition 5.3.2 (extremer Fluss):

Ein s - t -Fluss f mit $f \leq c$ heißt **extrem**, falls $\text{cost}(f) \leq \text{cost}(g)$ für alle s - t -Flüsse g mit $g \leq c$ und $\text{value}(f) = \text{value}(g)$.

Ergänze den Residualgraph $D_f = (V, A_f)$ um die Längenfunktion $l: A_f \rightarrow \mathbb{R}$ die für $a \in A_f$ definiert ist als

$$l(a) = \begin{cases} k(a), & \text{falls } a \in A \\ -k(a^{-1}), & \text{falls } a^{-1} \in A \end{cases}$$

Beispiel 5.3.1:



Satz 5.3.1:

f ist extrem $\Leftrightarrow D_f$ hat keinen gerichteten Kreis negativer Länge (bezüglich l)

Beweis:

„ $\Rightarrow$ “: Angenommen $C = (a_1, \dots, a_k)$ ist ein Kreis negativer Länge in D_f , dann gilt $l(C) = l(a_1) + \dots + l(a_k) < 0$. Definiere für $i = 1, \dots, k$

$$\sigma_i = \begin{cases} c(a_i) - f(a_i), & \text{für } a_i \in A \\ f(a_i^{-1}), & \text{für } a_i^{-1} \in A \end{cases}$$

Es gilt immer $\sigma_i > 0$. Sei $\alpha = \min\{\sigma_1, \dots, \sigma_k\}$. Definiere für $a \in A$

$$g(a) = \begin{cases} f(a) + \alpha, & \text{wenn } a \in C \\ f(a) - \alpha, & \text{wenn } a^{-1} \in C \\ f(a), & \text{sonst} \end{cases}$$

Dann ist g ebenfalls ein durch c beschränkter s - t -Fluss mit $\text{value}(f) = \text{value}(g)$. Außerdem $\text{cost}(g) = \text{cost}(f) + \alpha l(C) < \text{cost}(f)$. Also ist f nicht extrem.

„ $\Leftarrow$ “: Sei g ein Fluss mit $\text{value}(g) = \text{value}(f)$. Definiere

$$h: A_f \rightarrow \mathbb{R}_{\geq 0} \text{ durch } \begin{cases} h(a) &= g(a) - f(a), & \text{falls } g(a) > f(a) \\ h(a^{-1}) &= f(a) - g(a), & \text{falls } g(a) < f(a) \\ h(a) &= 0, & \text{für alle anderen } a \in A_f \end{cases}$$

Behauptung:

Dann ist h eine Zirkulation in D_f .

Beweis:

Nach Voraussetzung gilt, wegen Flusserhaltung bzw. $\text{value}(f) = \text{value}(g)$, $\forall v \in V$

$$0 = \sum_{a \in \delta_D^+(v)} g(a) - \sum_{a \in \delta_D^-(v)} g(a) - \left(\sum_{a \in \delta_D^+(v)} f(a) - \sum_{a \in \delta_D^-(v)} f(a) \right)$$

Es ist zu zeigen, dass

$$0 = \sum_{a \in \delta_{D_f}^+(v)} h(a) - \sum_{a \in \delta_{D_f}^-(v)} h(a)$$

Dies geschieht mit einer Bijektion der Summanden.

1. Fall: $a \in \delta_D^+(v)$:

(α) $g(a) > f(a)$:

Dann $a \in \delta_{D_f}^+(v)$ und $h(a) = g(a) - f(a)$

(β) $g(a) = f(a)$:

Dann $h(a) = 0$ falls $a \in D_f$ bzw. $h(a^{-1}) = 0$ falls $a^{-1} \in D_f$

(γ) $g(a) < f(a)$:

Dann $a^{-1} \in D_f$, also $a^{-1} \in \delta_{D_f}^-(v)$ und $h(a^{-1}) = f(a) - g(a)$

2. Fall: $a \in \delta_D^-(v)$:

analog

Nach Satz 5.2.2 ist

$$h \in \text{cone}\{\chi^C : C \text{ gerichteter Kreis in } D_f\}$$

Also ist $h = \sum_{C \in \mathcal{C}} \lambda_C \chi^C$ mit $\lambda_C \geq 0$. Dann gilt $\text{cost}(g) - \text{cost}(f) = \sum_{C \in \mathcal{C}} \lambda_C l(C) \geq 0$, also $\text{cost}(g) \geq \text{cost}(f)$ und f ist extrem.

□

Satz 5.3.2:

Sei $D = (V, A)$ ein gerichteter Graph, $c: A \rightarrow \mathbb{R}_{\geq 0}$ eine Kapazitätsfunktion, $k: A \rightarrow \mathbb{R}_{\geq 0}$ eine Kostenfunktion und f ein s - t -Fluss, welcher extrem und durch c beschränkt ist. Sei P ein gerichteter s - t -Weg in D_f mit minimaler Länge. Sei $f' = f + \varepsilon \chi^P$ für $\varepsilon > 0$ ebenfalls durch c beschränkt. Dann ist f' extrem.

Beweis:

Sei f'' ein s - t -Fluss, der durch c beschränkt ist, mit $\text{value}(f') = \text{value}(f'')$. Dann gilt

$$f'' - f = \sum_{i=1}^n \mu_i \chi^{P_i} + \sum_{j=1}^m \lambda_j \chi^{C_j}$$

wobei $\mu_1, \dots, \mu_n > 0$, $\lambda_1, \dots, \lambda_m > 0$ und $P_1, \dots, P_n$ sind s - t -Wege in D_f , $C_1, \dots, C_m$ sind gerichtete Kreise in D_f . Des weiteren ist $\sum_{i=1}^n \mu_i = \varepsilon$. Dann gilt

$$\text{cost}(f'' - f) = \underbrace{\sum_{i=1}^n \mu_i \underbrace{l(P_i)}_{\geq l(P)}}_{\geq \varepsilon l(P)} + \sum_{j=1}^m \lambda_j \underbrace{l(C_j)}_{\geq 0} \geq \varepsilon l(P)$$

Da $\text{cost}(f' - f) = \varepsilon l(P)$ gilt, ist $\text{cost}(f'') \geq \text{cost}(f')$.

□

Algorithmus 5.3.1 zur Berechnung maximaler Flüsse mit minimalen Kosten

- 1: $f = 0$
 - 2: **while** $\exists$ gerichteter s - t -Weg P in Residualgraph D_f mit minimaler Länge **do**
 - 3: $f = f + \varepsilon \chi^P$ wobei $\varepsilon > 0$ maximal gewählt wird, so dass $0 \leq f + \varepsilon \chi^P \leq c$ gilt
 - 4: **end while**
-

Wieder ist der Bellman-Ford-Algorithmus (Algorithmus 2.2.1) anwendbar, da D_f keine Kreise negativer Länge hat.

Theorie der linearen Optimierung

6. Polyedertheorie	43
6.1. Motivation	43
6.2. Konvexe Mengen	44
6.3. Trenn- und Stütz-Hyperebenen	49
6.4. Extrempunkte & Ecken	54
6.5. Theorem von Minkowski-Weyl	56
6.6. Lemma von Farkas	61
6.7. Lineare Programmierung	63
7. Ganzzahlige Optimierung und vollständig-unimodulare Matrizen	67
7.1. Ganzzahlige lineare Programme	67
7.2. Vollständig-unimodulare Matrizen	69
7.3. Vollständig-unimodulare Matrizen und bipartite Graphen	71
7.4. Vollständig-unimodulare Matrizen und gerichtete Graphen	74
8. Ganzzahlige Optimierung, Schnittebenen und vollständig duale ganzzahlige Systeme	77
8.1. LP und ILP	77
8.2. Chvátal-Gomory-Abschluss	79
8.3. Total duale ganzzahlige Systeme und Hilbertbasen	81
9. Spieltheorie	85
9.1. Bimatrixspiele	85
9.2. Matrixspiele	89

6. Polyedertheorie

16.05.2014
11. Vorlesung

6.1. Motivation	43
6.2. Konvexe Mengen	44
6.3. Trenn- und Stütz-Hyperebenen	49
6.4. Extrempunkte & Ecken	54
6.5. Theorem von Minkowski-Weyl	56
6.6. Lemma von Farkas	61
6.7. Lineare Programmierung	63

6.1. Motivation

Viele (Optimierungs-)Probleme des Operations Research lassen sich als LP (linear programm) formulieren.

Definition 6.1.1 (lineares Programm):

Ein **lineares Programm (LP)** in Standardform ist

$$\begin{aligned} \max \quad & c^\top x \\ \text{wobei } x \in \mathbb{R}^n \\ & a_1^\top x \leq b_1 \\ & \vdots \\ & a_m^\top x \leq b_m \end{aligned} \tag{LP}$$

wobei $c \in \mathbb{R}^n$ die Zielfunktion $x \mapsto c^\top x$ beschreibt und $a_1, \dots, a_m \in \mathbb{R}^n$, $b_1, \dots, b_m \in \mathbb{R}$ beschreiben die linearen Ungleichungen (Nebenbedingungen).

x wird hier als Optimierungsvariable bezeichnet.

Kurzschreibweise: $Ax \leq b$, mit $A = \begin{pmatrix} a_1^\top \\ \vdots \\ a_m^\top \end{pmatrix} \in \mathbb{R}^{m \times n}$, $b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \in \mathbb{R}^m$.

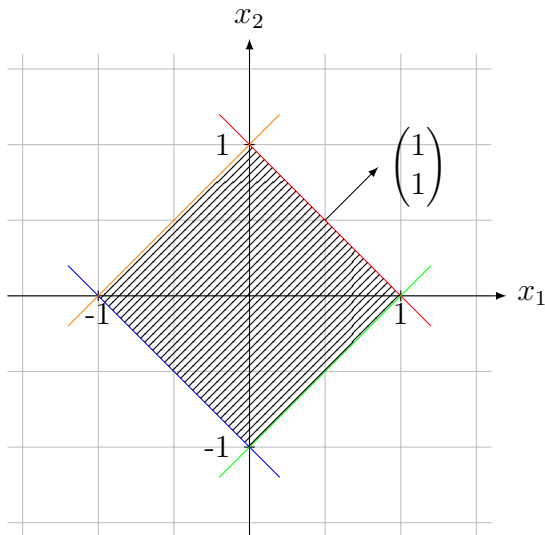
Manchmal soll x zusätzlich Ganzzahligkeitsbedingungen erfüllen: $x \in \mathbb{Z}^n \rightsquigarrow$ Kapitel 7

Dieses Kapitel: Wie sieht die Menge der *zulässigen Lösungen*

$$\{x \in \mathbb{R}^n : Ax \leq b\}$$

geometrisch aus? $\Rightarrow$ (**konvexer**) **Polyeder**

Beispiel 6.1.1:



$$A = \begin{pmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$P = \{x \in \mathbb{R}^2 : Ax \leq b\}$$

$$Ax \leq b \Leftrightarrow \begin{cases} x_1 + x_2 & \leq 1 \\ x_1 - x_2 & \leq 1 \\ -x_1 + x_2 & \leq 1 \\ -x_1 - x_2 & \leq 1 \end{cases}$$

$$a_1^T x = 1$$

$$a_2^T x = 1$$

$$a_3^T x = 1$$

$$a_4^T x = 1$$

Polyedertheorie: Erweiterung der linearen Algebra über den Körper $\mathbb{R}$ durch lineare Ungleichungen.

6.2. Konvexe Mengen

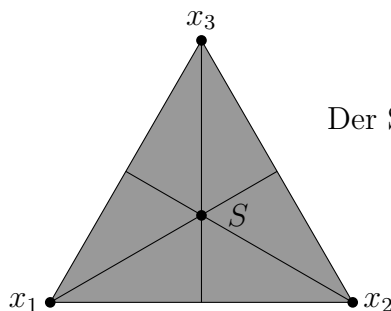
Definition 6.2.1 (Konvexkombination):

Seien $x_1, \dots, x_N \in \mathbb{R}^n$ gegeben. Dann ist $y \in \mathbb{R}^n$ eine **Konvexkombination** von $x_1, \dots, x_N$, falls

$$y = \sum_{i=1}^N \alpha_i x_i$$

mit $\alpha_1, \dots, \alpha_N \geq 0$ und $\sum_{i=1}^N \alpha_i = 1$.

Beispiel 6.2.1:



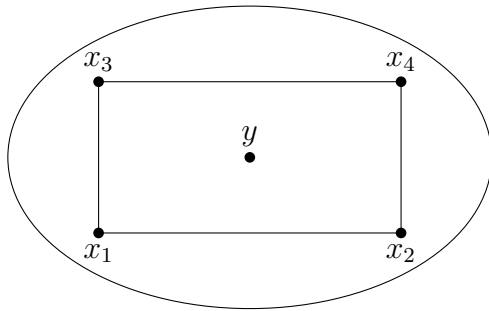
Der Schwerpunkt S ist eine Konvexkombination

$$S = \frac{1}{3}(x_1 + x_2 + x_3)$$

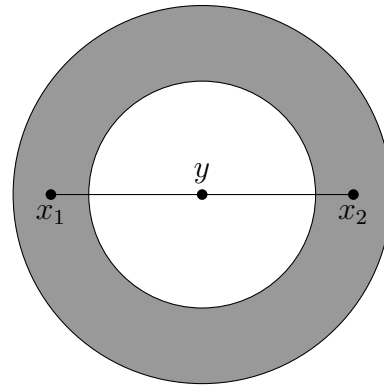
Definition 6.2.2 (konvexe Menge):

Eine Menge $C \subseteq \mathbb{R}^n$ heißt **konvex**, falls sie unter Konvexkombinationen abgeschlossen ist, d. h.

$$\forall N \in \mathbb{N}, \forall x_1, \dots, x_N \in C, \forall \alpha_1, \dots, \alpha_N \geq 0, \sum_{i=1}^N \alpha_i = 1 \Rightarrow \sum_{i=1}^N \alpha_i x_i \in C$$

Beispiel 6.2.2:

y ist innerhalb $\Rightarrow$ konvex



y ist außerhalb $\Rightarrow$ nicht konvex

Notation:

Seien $x, y \in \mathbb{R}^n$ Punkte, dann ist die Menge aller Konvexkombinationen

$$[x, y] = \left\{ \underbrace{(1 - \alpha)}_{=\alpha_1} x + \underbrace{\alpha}_{=\alpha_2} y : \alpha \in [0, 1] \right\}$$

die **Verbindungsstrecke** zwischen x und y .

Satz 6.2.1:

Eine Menge $C \subseteq \mathbb{R}^n$ ist konvex $\iff \forall x, y \in C: [x, y] \subseteq C$

Beweis:

„ $\Rightarrow$ “: Klar, Spezialfall der Definition ($N = 2$)

„ $\Leftarrow$ “: Wird gezeigt per Induktion über N :

Jede Konvexkombination von weniger oder gleich N Punkten liegt wieder in C .

Induktionsanfang: $N = 1$ ✓

Induktionsschritt: $N \mapsto N + 1$

Sei $y = \sum_{i=1}^{N+1} \alpha_i x_i$ mit $\alpha_i \geq 0$, $\sum_{i=1}^{N+1} \alpha_i = 1$ und $x_i \in C$.

1. Fall: $\alpha_{N+1} = 1$

$$\Rightarrow \alpha_1, \dots, \alpha_N = 0$$

$$\Rightarrow y = x_{N+1} \in C$$

2. Fall: $\alpha_{N+1} \neq 1$

Nach Induktionsvoraussetzung ist

$$y' = \sum_{i=1}^N \underbrace{\frac{\alpha_i}{1 - \alpha_{N+1}}}_{\geq 0} x_i \in C, \text{ weil } \frac{\alpha_i}{1 - \alpha_{N+1}} \geq 0 \Rightarrow \sum_{i=1}^N \alpha_i = 1 - \alpha_{N+1}$$

Des weiteren gilt:

$$y = (1 - \alpha_{N+1}) \underbrace{y'}_{\in C} + \alpha_{N+1} \underbrace{x_{N+1}}_{\in C} \in [y', x_{N+1}] \subseteq C$$

□

Definition 6.2.3 (Metrik):

Sei X eine Menge und $d: X \times X \rightarrow \mathbb{R}_{\geq 0}$ eine Abbildung, die je zwei Elementen x und y aus X einen Abstand $d(x, y)$ zuordnet. Die Abbildung d ist eine **Metrik**, falls für alle $x, y, z \in X$ gilt

$$\begin{aligned} d(x, y) &= 0 \iff x = y \\ d(x, y) &= d(y, x) \\ d(x, z) &\leq d(x, y) + d(y, z) \end{aligned}$$

Das Paar (X, d) heißt dann **metrischer Raum**.

Definition 6.2.4 (Minkowski-Metrik):

Die Familie der **Minkowski-Metriken** $L_p(x, y)$ ist folgendermaßen definiert

$$L_p(x, y) := \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}} \text{ für } 1 \leq p < \infty$$

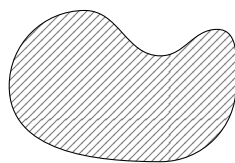
Für $p = 1$ erhalten wir die **Manhattan-Metrik**, für $p = 2$ die **euklidische Metrik**. Für $p = \infty$ ergibt sich $L_\infty(x, y) := \max_{i=1, \dots, n} |x_i - y_i|$ und wird als **Maximum-Metrik** bezeichnet.

Sei $\|\cdot\|: \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ eine Norm. Dann ist die zugehörige Einheitskugel

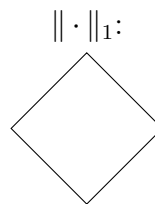
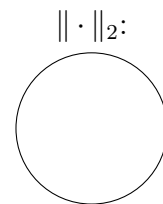
$$B = \{x \in \mathbb{R}^n: \|x\| \leq 1\}$$

konvex. Seien $x, y \in B$ und sei $\alpha \in [0, 1]$. Betrachte

$$\begin{aligned} \|\alpha x + (1 - \alpha)y\| &\leq \|\alpha x\| + \|(1 - \alpha)y\| \\ &= \alpha \|x\| + (1 - \alpha) \|y\| \\ &\leq \alpha + (1 - \alpha) \\ &= 1 \end{aligned}$$

Beispiel 6.2.3:

nicht konvex

 $\|\cdot\|_1$  $\|\cdot\|_2$

konvex

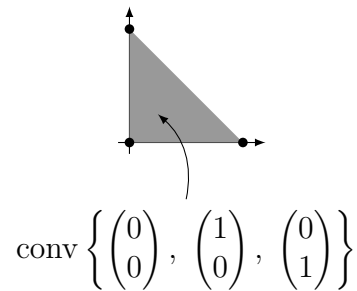
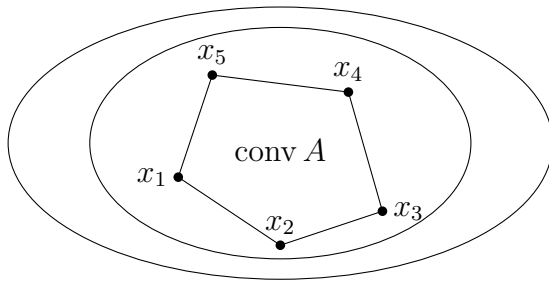
Definition 6.2.5 (konvexe Hülle):

Sei $A \subseteq \mathbb{R}^n$. Die **konvexe Hülle** von A ist

$$\text{conv } A = \bigcap_{\substack{B \supseteq A \\ B \text{ konvex}}} B$$

Der Durchschnitt von konvexen Mengen ist wieder konvex. Bei $\text{conv } A$ handelt es sich also um die inklusionskleinste konvexe Menge, die A enthält.

Beispiel 6.2.4:

**Satz 6.2.2:**

Sei $A \subseteq \mathbb{R}^n$. Dann gilt

$$\text{conv } A = \left\{ y \in \mathbb{R}^n \mid \exists N \in \mathbb{N}, \exists x_1, \dots, x_N \in A, \alpha_i \geq 0, \sum_{i=1}^N \alpha_i = 1 : y = \sum_{i=1}^N \alpha_i x_i \right\}$$

$\text{conv } A$ besteht also aus allen Konvexkombinationen der Menge A .

Beweis:

„ $\subseteq$ “: trivial ✓

„ $\supseteq$ “: Sei B konvex mit $B \supseteq A$. Da B abgeschlossen unter Konvexkombinationen ist, muss für jede Wahl von $N \in \mathbb{N}$, $x_1, \dots, x_N \in A$, $\alpha_i \geq 0$ und $\sum_{i=1}^N \alpha_i = 1$ immer $y = \sum_{i=1}^N \alpha_i x_i \in B$ gelten.

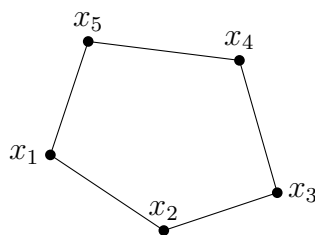
□

Definition 6.2.6 (Polytop):

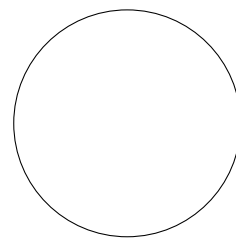
Eine Menge $P \subseteq \mathbb{R}^n$ heißt **(konvexes) Polytop**, wenn es eine Menge von endlich vielen Punkten $A = \{x_1, \dots, x_N\}$ gibt, so dass

$$P = \text{conv } A$$

Beispiel 6.2.5:



Polytop



Kein Polytop

Frage: Wie groß muss N sein, damit $y \in \text{conv } A$ ist?

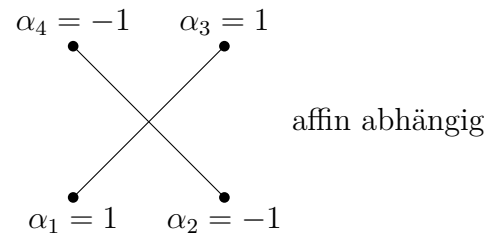
Definition 6.2.7 (affin unabhängig):

Die Punkte $x_1, \dots, x_N \in \mathbb{R}^n$ heißen **affin unabhängig**, wenn

$$\forall \alpha_1, \dots, \alpha_N \in \mathbb{R}: \sum_{i=1}^N \alpha_i = 0, \sum_{i=1}^N \alpha_i x_i = 0 \Rightarrow \alpha_1 = \dots = \alpha_N = 0$$

Beispiel 6.2.6:

• affin unabhängig
 O



Seien $x_1, \dots, x_4 \in \mathbb{R}^2$. Dann

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ aber } \dim \text{Kern} \begin{pmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \end{pmatrix} \geq 1$$

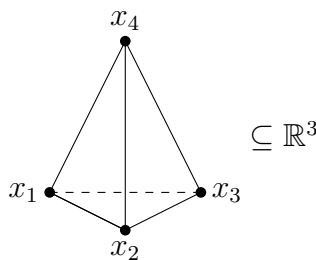
21.05.2014
12. Vorlesung

Definition 6.2.8 (Dimension):

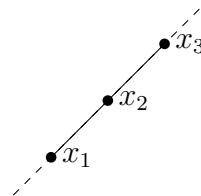
Sei $A \subseteq \mathbb{R}^n$. Dann ist die **Dimension** von A definiert als

$$\dim A = \max \{N - 1 : \exists x_1, \dots, x_N \in A, x_1, \dots, x_N \text{ affin unabhängig}\}$$

Beispiel 6.2.7:



$x_1, \dots, x_4$ sind affin unabhängig
 $\Rightarrow \dim\{x_1, \dots, x_4\} = 3$



$\dim\{x_1, x_2, x_3\} = 1$

$x_1 \bullet$
 $\dim\{x_1\} = 0$

Satz 6.2.3 (Carathéodory):

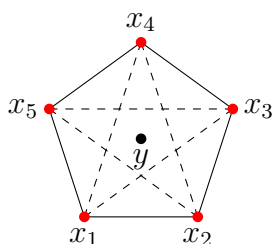
Sei $A \subseteq \mathbb{R}^n$ und sei $y \in \text{conv } A$. Dann $\exists x_1, \dots, x_N \in A$ affin unabhängig mit $y \in \text{conv}\{x_1, \dots, x_N\}$.

Insbesondere ist

$$N \leq \dim A + 1 \leq n + 1$$

Es ist immer $\dim A \leq n$, weil $\text{rang} \begin{pmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_N \end{pmatrix} \leq n + 1$.

Beispiel 6.2.8:



Im Allgemeinen gilt $\dim = 3$

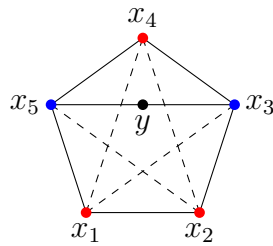
$$\dim\{x_1, x_2, x_4\} = 3$$

$$\dim\{x_2, x_3, x_5\} = 3$$

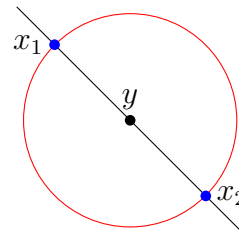
$$\dim\{x_1, x_3, x_4\} = 3$$

$$\dim\{x_2, x_4, x_5\} = 3$$

$$\dim\{x_1, x_3, x_5\} = 3$$



Im Spezialfall, wenn y auf einer Verbindungslinie liegt, gilt $\dim = 2$
 $\dim\{x_3, x_5\} = 2$



Hier gilt immer $\dim = 2$
 Es lassen sich immer zwei Punkte auf dem Rand finden, so dass eine Verbindungslinie existiert

Beweis:

Sei $y \in \text{conv } A$ mit Darstellung $y = \sum_{i=1}^N \alpha_i x_i$, $x_i \in A$, $\alpha_i \geq 0$, $\sum_{i=1}^N \alpha_i = 1$.

Angenommen N ist minimal und $x_1, \dots, x_N$ sind affin abhängig. Seien $\beta_1, \dots, \beta_N \in \mathbb{R}$ mit $\sum_{i=1}^N \beta_i x_i = 0$, $\sum_{i=1}^N \beta_i = 0$ aber $(\beta_1, \dots, \beta_N) \neq (0, \dots, 0)$. Wähle $\beta_j > 0$, so dass $\frac{\alpha_j}{\beta_j}$ minimal.

Betrachte die Neudarstellung $y = \sum_{i=1}^N \alpha_i x_i - \underbrace{\frac{\alpha_j}{\beta_j} \sum_{i=1}^N \beta_i x_i}_{=0}$.

Behauptung:

Dies ist eine Konvexkombination von y der Punkte $x_1, x_2, \dots, \hat{x}_j, \dots, x_N$.

Beweis:

i) $\alpha_i - \frac{\alpha_j}{\beta_j} \beta_i \geq 0$

1. Fall: $\beta_i \leq 0$ Klar

2. Fall: $\beta_i > 0$ Nach Wahl von b_j klar

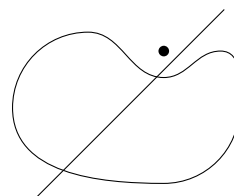
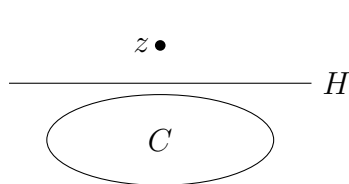
ii) $\sum_{i=1}^N \alpha_i - \frac{\alpha_j}{\beta_j} \beta_i = 1 \quad \nrightarrow$ zur Minimalität von N

iii) $\underbrace{\alpha_i}_{0 \cdot \hat{x}_j = 0} - \frac{\alpha_j}{\beta_j} \beta_i = 0$

□

6.3. Trenn- und Stütz-Hyperebenen

Eine *Fundamentale Eigenschaft* abgeschlossener konvexer Mengen $C \subseteq \mathbb{R}^n$, mit $C \neq \emptyset$ ist, dass jeder Punkt $z \notin C$ durch eine Hyperebene von C getrennt werden kann.



Wichtige Verallgemeinerung in der Funktionalanalysis:
 Satz von Hahn-Banach

Definition 6.3.1 (Hyperebene und Halbraum):

Eine Menge $H \subseteq \mathbb{R}^n$ heißt **(affine) Hyperebene**, falls es einen Vektor $c \in \mathbb{R}^n \setminus \{0\}$ und $\delta \in \mathbb{R}$ gibt, so dass

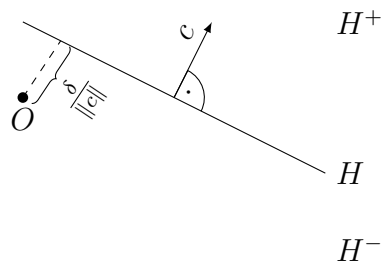
$$H = \{x \in \mathbb{R}^n : c^\top x = \delta\}$$

Die abgeschlossenen und konvexen Mengen

$$H^+ = \{x \in \mathbb{R}^n : c^\top x \geq \delta\}$$

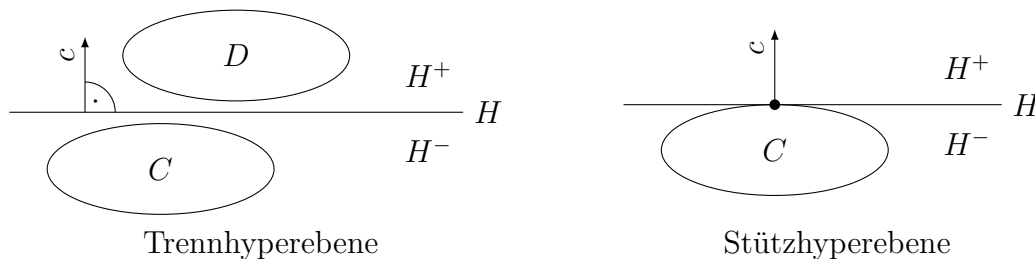
$$H^- = \{x \in \mathbb{R}^n : c^\top x \leq \delta\}$$

heißen **Halbräume**.

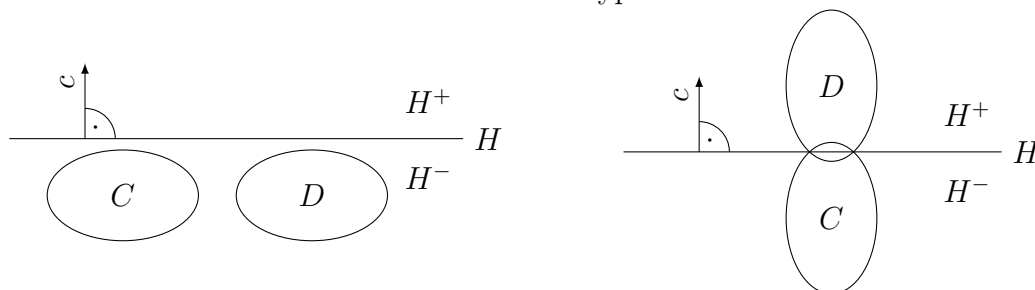
Beispiel 6.3.1:**Definition 6.3.2 (Trenn- und Stütz-Hyperebene):**

Seien die Mengen $C, D \subseteq \mathbb{R}^n$ gegeben. Eine Hyperebene $H \subseteq \mathbb{R}^n$ heißt **Trennhyperebene** von C und D , falls $C \subseteq H^-$ und $D \subseteq H^+$ (oder umgekehrt).

Eine Hyperebene $H \subseteq \mathbb{R}^n$ heißt **Stützhyperebene** von C , falls $C \subseteq H^-$ und $C \cap H \neq \emptyset$.

Beispiel 6.3.2:**Beispiel 6.3.3 (Gegenbeispiel):**

Nicht trennende Hyperebenen



Frage: Wie findet man Trenn- und Stütz-Hyperebenen für eine konvexe Menge $C \subseteq \mathbb{R}^n$?

Antwort: Verwende **metrische Projektion** (also die beste Approximation von z in C bzw. die orthogonale Projektion von z auf C , falls C affin linear).

Notation für metrische Projektion von z auf C : $y = \Pi_C(z)$.

Lemma 6.3.1:

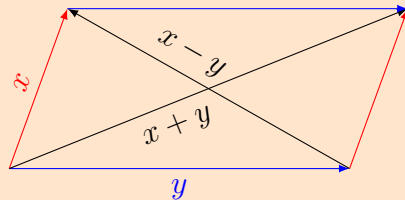
Sei $C \subseteq \mathbb{R}^n$ eine abgeschlossene konvexe Menge mit $C \neq \emptyset$ und $z \notin C$. Dann

$$\exists! y \in C \text{ mit } \|y - z\| = \inf_{y' \in C} \|y' - z\|$$

Zur Erinnerung:

Parallelogrammgesetz:

Für $x, y \in \mathbb{R}^n$: $\|x + y\|^2 + \|x - y\|^2 = 2\|x\|^2 + 2\|y\|^2$



Beweis durch nachrechnen

mit $\|x\| = x^T x$

Beweis (Funktioniert auch in allgemeinen Hilberträumen):

Existenz von y : Sei $d = \inf_{y' \in C} \|y' - z\|$.

Ohne Einschränkung: $z = 0$ (durch Translation).

Sei $(x_n)_{n \in \mathbb{N}}$ eine Folge aus C mit $\lim_{n \rightarrow \infty} \|x_n\| = d$. Es ist zu zeigen, dass $(x_n)_{n \in \mathbb{N}}$ eine Cauchyfolge ist. Nach Parallelogrammgesetz gilt:

$$\underbrace{\left\| \frac{x_n + x_m}{2} \right\|^2}_{\substack{\in C \\ \geq d^2}} + \underbrace{\left\| \frac{x_n - x_m}{2} \right\|^2}_{\rightarrow 0} = \underbrace{\frac{1}{2} \|x_n\|^2 + \frac{1}{2} \|x_m\|^2}_{\substack{\rightarrow d^2 \quad \rightarrow d^2}}$$

Weil $\mathbb{R}^n$ vollständig ist, konvergiert $(x_n)_{n \in \mathbb{N}}$. Weil C abgeschlossen ist, liegt der Grenzwert $\lim_{n \rightarrow \infty} x_n$ ebenfalls in C . Dieser Grenzwert ist ein y mit der gewünschten Eigenschaft.

Eindeutigkeit von y : Angenommen $\exists y, y'$ mit $\|y - z\| = \|y' - z\| = \inf_{y'' \in C} \|y'' - z\|$ mit $y \neq y'$. Dann

$$\begin{aligned} \underbrace{\left\| \frac{y + y'}{2} \right\|^2}_{\in C} &< \left\| \frac{y + y'}{2} \right\|^2 + \underbrace{\left\| \frac{y - y'}{2} \right\|^2}_{> 0 \text{ da } y \neq y'} \\ &= \frac{1}{2} \|y\|^2 + \frac{1}{2} \|y'\|^2 \\ &= d^2 \end{aligned}$$

Dies ist ein Widerspruch zur Minimalität von d .

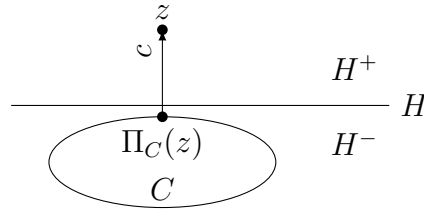
□

Theorem 6.3.1:

Sei $C \subseteq \mathbb{R}^n$ konvex und abgeschlossen. Sei $C \neq \emptyset$ und $z \notin C$. Dann ist die Hyperebene

$$H = \{x \in \mathbb{R}^n : c^\top x = \delta\}$$

wobei $c = z - \Pi_C(z)$ und $\delta = \frac{1}{2}(\|z\|^2 - \|\Pi_C(z)\|^2)$ eine Trennhyperebene von $\{z\}$ und C .

**Beweis:**

Definiere $c = z - \Pi_C(z)$ und $\delta = \frac{1}{2}(\|z\|^2 - \|\Pi_C(z)\|^2)$.

Behauptung:

$H = \{x \in \mathbb{R}^n : c^\top x = \delta\}$ ist eine Trennhyperebene von $\{z\}$ und C .

Beweis:

1. $z \in H^+$:

Es gilt sogar $c^\top z > \delta$, weil

$$\begin{aligned} c^\top z &= (z - \Pi_C(z))^\top z \\ &> (z - \Pi_C(z))^\top z - \frac{1}{2}\|z - \Pi_C(z)\|^2 \\ &= \delta \end{aligned}$$

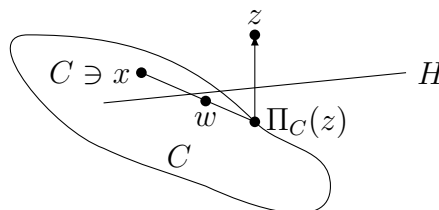
2. $C \subseteq H^-$:

Es ist sogar $c^\top x < \delta \forall x \in C$.

Angenommen es existiert $x \in C : c^\top x \geq \delta$.

Dann gilt $c^\top(x - \Pi_C(z)) > 0$, weil $c^\top \Pi_C(z) < c^\top \Pi_C(z) + \frac{1}{2}\|c\|^2 = \delta$. Also gibt es ein $\lambda \in (0, 1]$ mit der zusätzlichen Eigenschaft

$$\lambda < \frac{2c^\top(x - \Pi_C(z))}{\|x - \Pi_C(z)\|^2} \quad (6.1)$$



Setze $w = \lambda x + (1 - \lambda)\Pi_C(z) \in C$. Dann ist

$$\begin{aligned}
 \|w - z\|^2 &= \left\| \lambda(x - \Pi_C(z)) + \underbrace{\Pi_C(z) - z}_{=-c} \right\|^2 \\
 &= \left\| \lambda(x - \Pi_C(z)) - c \right\|^2 \\
 &= \underbrace{\lambda^2 \|x - \Pi_C(z)\|^2 - 2\lambda c^\top (x - \Pi_C(z)) + \|c\|^2}_{< 0 \text{ wegen (6.1)}} \\
 &\stackrel{(6.1)}{<} \|c\|^2 \\
 &= \|\Pi_C(z) - z\|^2
 \end{aligned}$$

Dies ist ein Widerspruch zur Definition der metrischen Projektion von $\Pi_C(z)$.

□

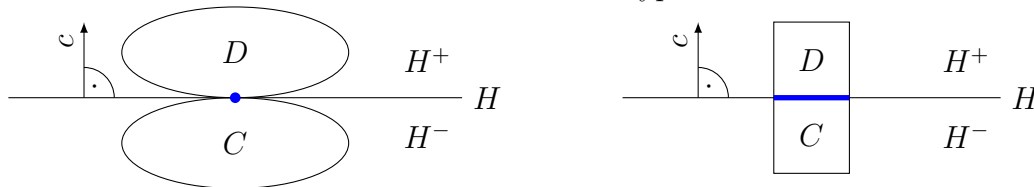
23.05.2014
13. Vorlesung

Bemerkung 6.3.1:

Die im Beweis konstruierte Trennhyperebene ist **strikt**, d. h. $z \in H^+$, $C \subseteq H^-$, $\{z\} \cap H = \emptyset$, $C \cap H = \emptyset$.

Beispiel 6.3.4 (Gegenbeispiel):

Nicht strikt trennende Hyperebenen



Korollar 6.3.1.a:

Sei $C \subseteq \mathbb{R}^n$ eine konvexe und abgeschlossene Menge mit $C \neq \emptyset$. Dann gilt

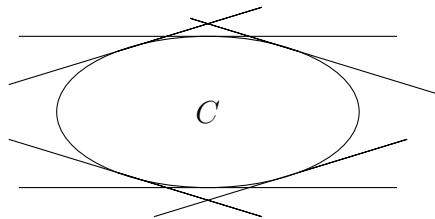
$$C = \bigcap_{H \text{ Stützhyperebene von } C} H^-$$

Beweis:

Übung.

□

Beispiel 6.3.5:

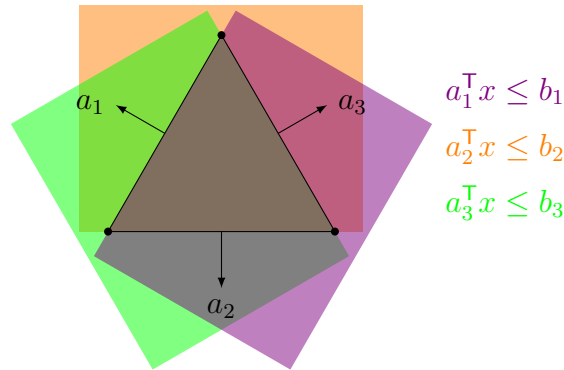


Definition 6.3.3 (Polyeder):

Eine Menge $P \subseteq \mathbb{R}^n$ heißt **Polyeder**, falls es eine Matrix $A \in \mathbb{R}^{m \times n}$ und einen Vektor $b \in \mathbb{R}^m$ gibt, so dass

$$P = \{x \in \mathbb{R}^n : Ax \leq b\}$$

(Man kann Polyeder auch als Durchschnitt endlich vieler Halbräume oder als Lösungsmenge eines Systems linearer Ungleichungen definieren.)

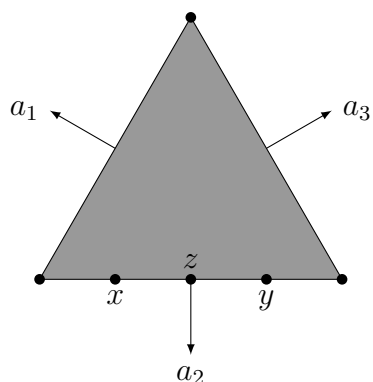
Beispiel 6.3.6:

Jetzt: Verhältnis zwischen Polyedern und Polytopen.

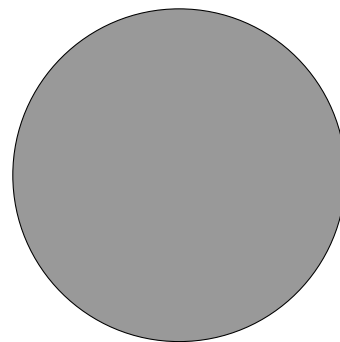
6.4. Extrempunkte & Ecken**Definition 6.4.1 (Extrempunkt):**

Sei $C \subseteq \mathbb{R}^n$ eine konvexe Menge. Ein Punkt $z \in C$ heißt **Extrempunkt** von C genau dann, wenn

$$\forall x, y \in C : z \in (x, y) = \{\alpha x + (1 - \alpha)y : \alpha \in (0, 1)\} \Rightarrow x = z = y$$

Beispiel 6.4.1:

3 Extrempunkte
 z ist kein Extrempunkt



Unendlich viele Extrempunkte

Bemerkung 6.4.1:

Extrempunkte von Polyedern heißen **Ecken**.

Notation: Sei $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ ein Polyeder und $z \in P$. Dann ist A_z die Teilmatrix von A , die genau die Zeilen $a_i^\top$ enthält mit $a_i^\top z = b_i$ („aktive Ungleichungen“).

Satz 6.4.1:

Sei $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ ein Polyeder und $z \in P$. Dann gilt

$$z \text{ ist Ecke von } P \iff \text{rang } A_z = n$$

Beweis:

„ $\Rightarrow$ “: Angenommen z ist eine Ecke, aber $\text{rang } A_z < n$, dann $\exists c \neq 0 : A_z c = 0$. Da $a_i^\top z < b_i$ für alle Zeilen $a_i^\top$ von A , die nicht zu A_z gehören, gibt es $\delta > 0$ mit

$$a_i^\top (z + \delta c) \leq b_i \text{ und } a_i^\top (z - \delta c) \leq b_i$$

Also $A(z + \delta c) \leq b$ und $A(z - \delta c) \leq b$ und damit $z + \delta c, z - \delta c \in P$. Es folgt $z \in (z - \delta c, z + \delta c) \subseteq P$, d. h. z ist keine Ecke. $\nmid \Rightarrow \text{rang } A_z = n$.

„ $\Leftarrow$ “: Es sei $z \in (x, y)$ mit $x, y \in P$, wobei $z = \alpha x + (1 - \alpha)y$ mit $\lambda \in (0, 1)$. Für eine Zeile $a_i^\top$ von A_z gilt

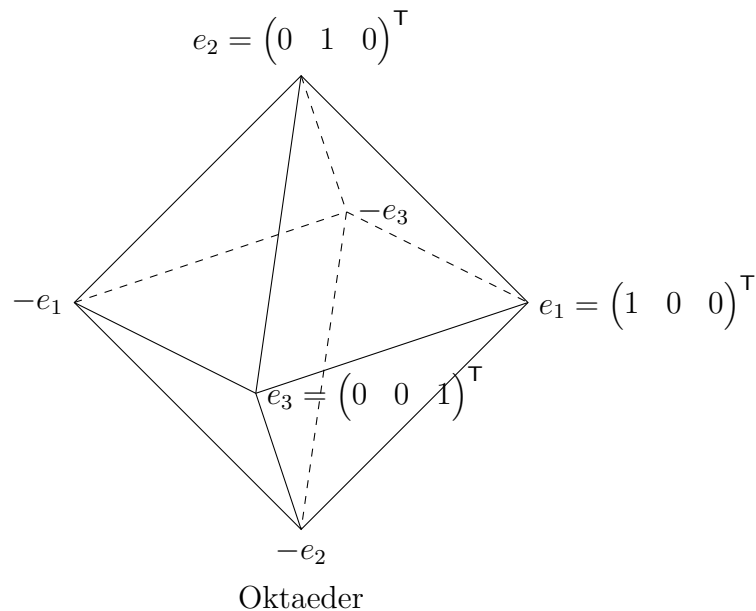
$$\begin{aligned} b_i &= a_i^\top z \\ &= a_i^\top (\alpha x + (1 - \alpha)y) \\ &= \alpha a_i^\top x + (1 - \alpha) a_i^\top y \\ &\leq \alpha b_i + (1 - \alpha) b_i \\ &= b_i \end{aligned}$$

Daraus folgt $a_i^\top x = b_i, a_i^\top y = b_i$, da $\alpha \in (0, 1)$, für alle Zeilen $a_i^\top$, die zu A_z gehören. Weil der Rang von $A_z = n$ ist, muss $x = y = z$ gelten. $\Rightarrow z$ ist ein Extrempunkt von P .

□

Beispiel 6.4.2:

$$P = \left\{ x \in \mathbb{R}^3 : \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & 1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \\ -1 & -1 & -1 \end{pmatrix} x \leq \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right\}$$



$$A_{e_1} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \end{pmatrix}$$

$\text{rang } A_{e_1} = 3 \Rightarrow e_1 \text{ ist Ecke von } P$

Bemerkung 6.4.2:

Der Satz 6.4.1 gibt einen Algorithmus zur Bestimmung aller Eckpunkte von P , indem man alle $n \times n$ Teilmatrizen von A betrachtet.

Korollar 6.4.1.a:

Ein Polyeder $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ mit $A \in \mathbb{R}^{m \times n}$ hat höchstens $\binom{m}{n}$ Ecken.

6.5. Theorem von Minkowski-Weyl

28.05.2014
14. Vorlesung

Definition 6.5.1 (Polare):

Sei $A \subseteq \mathbb{R}^n$. Dann ist die **Polare** von A definiert als

$$A^* = \{y \in \mathbb{R}^n : x^\top y \leq 1 \quad \forall x \in A\}$$

Lemma 6.5.1:

- (1) $(\alpha A)^* = \frac{1}{\alpha} A^*$, wobei $\alpha > 0$ und $\alpha A = \{\alpha a : a \in A\}$
- (2) $A \subseteq B \Rightarrow B^* \subseteq A^*$
- (3) $(B_n)^* = B_n$, wobei $B_n = \{x \in \mathbb{R}^n : \underbrace{\|x\|_2}_{=\sqrt{x^\top x}} \leq 1\}$ (n -dimensionale Einheitskugel)
- (4) $P = \text{conv}\{x_1, \dots, x_t\} \Rightarrow P^* = \{y \in \mathbb{R}^n : x_i^\top y \leq 1, i = 1, \dots, t\}$

Beweis:

(1) und (2): Übung.

(3) „ $\subseteq$ “: Sei $y \in (B_n)^*$. Falls $y = 0$, dann $y \in B_n$. Falls $y \neq 0$, dann $y^\top x \leq 1$ für $x = \frac{y}{\|y\|}$, d.h. $y^\top \frac{y}{\|y\|} \leq 1 \Rightarrow \|y\| \leq 1$.

„ $\supseteq$ “: Seien $x, y \in B_n$. Dann $x^\top y \stackrel{\text{Cauchy-Schwarz}}{\leq} \underbrace{\|x\|}_{\leq 1} \cdot \underbrace{\|y\|}_{\leq 1} \leq 1$.

(4) „ $\supseteq$ “: trivial ✓

„ $\subseteq$ “: $y \in \mathbb{R}^n$ erfülle $y^\top x_i \leq 1$, $i = 1, \dots, t$. Sei $z \in P$. Dann $z = \sum_{i=1}^t \alpha_i x_i$ mit $\alpha_i \geq 0$, $\sum \alpha_i = 1$. Es gilt $y^\top z = \sum_{i=1}^t \alpha_i y^\top x_i \leq \sum_{i=1}^t \alpha_i = 1$ wie gewünscht. $\square$

Theorem 6.5.1 (Theorem von Minkowski-Weyl):Sei $P \subseteq \mathbb{R}^n$. Dann gilt

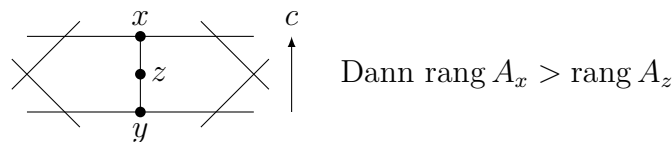
$$P \text{ ist beschränkter Polyeder} \iff P \text{ ist Polytop}$$

Beweis:„ $\Rightarrow$ “: (Theorem von Minkowski)Seien $x_1, \dots, x_t$ Ecken von P .**Behauptung:**

$$P = \text{conv}\{x_1, \dots, x_t\}$$

Beweis:„ $\supseteq$ “: trivial ✓„ $\subseteq$ “: Sei $z \in P$. Zeige $z \in \text{conv}\{x_1, \dots, x_t\}$ per Induktion nach n -rang A_z .**Induktionsanfang:** n -rang $A_z = 0$.Dann ist $\text{rang } A_z = n \Rightarrow z$ ist Ecke von $P \Rightarrow z \in \{x_1, \dots, x_t\}$. Also auch $z \in \text{conv}\{x_1, \dots, x_t\}$.**Induktionsschluss:** n -rang $A_z > 0$.Dann ist $\text{rang } A_z < n \Rightarrow \exists c \neq 0: A_z c = 0$. Seien

$$\mu_0 = \max\{\mu: z + \mu c \in P\} \text{ und } \nu_0 = \min\{\nu: z + \nu c \in P\}$$

 μ_0 und ν_0 existieren, weil P kompakt ist. Setzte $x = z + \mu_0 c$ und $y = z + \nu_0 c$.

Es gilt $a_i^\top(z + \mu c) \leq b_i \forall \mu$ mit $z + \mu c \in P$. Außerdem gilt $\mu \leq \frac{b_i - a_i^\top z}{a_i^\top c}$, falls $a_i^\top c > 0$. Dann

$$\mu_0 = \min \left\{ \frac{b_i - a_i^\top z}{a_i^\top c} : a_i^\top c > 0 \right\} \quad (6.2)$$

In (6.2) sei i_0 die Zeile von A , in der das Minimum angenommen wird. Dann ist

$$(1) \quad A_z x = A_z(z + \mu_0 c) = A_z z + \underbrace{\mu_0 A_z c}_{=0} = A_z z$$

$$(2) \quad a_{i_0}^\top x = a_{i_0}^\top(z + \mu_0 c) = b_{i_0}$$

Somit enthält A_x alle Zeilen von A_z und außerdem die Zeile $a_{i_0}^\top$. Da $A_z c = 0$ aber $a_{i_0}^\top c \neq 0$ gilt $\text{rang } A_x > \text{rang } A_z$. Nach Induktionsvoraussetzung ist $x \in \text{conv}\{x_1, \dots, x_t\}$. Genauso folgt $y \in \text{conv}\{x_1, \dots, x_t\}$. Da $z \in [x, y]$ ist, folgt die Behauptung $z \in \text{conv}\{x_1, \dots, x_t\}$.

„ $\Leftarrow$ “: (Theorem von Weyl)

Sei $P = \text{conv}\{x_1, \dots, x_t\} \subseteq \mathbb{R}^n$.

Ohne Einschränkung sei $0 \in P$ (durch Translation).

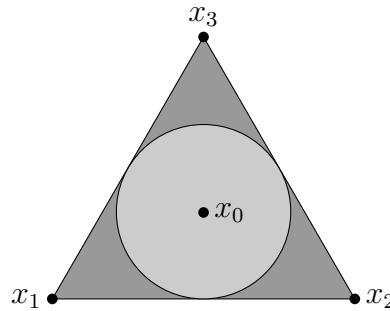
Ohne Einschränkung sei $\dim P = n$ (durch Einschränkung auf den Span von P).

Ohne Einschränkung seien $x_1, \dots, x_{n+1}$ affin unabhängig (durch Ummummerierung).

Für den Schwerpunkt

$$x_0 = \frac{1}{n+1}(x_1, \dots, x_{n+1})$$

gibt es ein $r > 0$, so dass $B(x_0, r) \subseteq P$, wobei $B(x_0, r) = \{y \in \mathbb{R}^n : \|x_0 - y\| \leq r\}$.



Ohne Einschränkung sei $x_0 = 0$ (nach Translation).

Nach den Teilen (1)-(3) von **Lemma 6.5.1** gilt $P^* \subseteq B(x_0, \frac{1}{r})^1$. Nach Teil (4) ist P^* ein beschränkter Polyeder und nach „ $\Rightarrow$ “ ist dies ein Polytop: $P^* = \text{conv}\{y_1, \dots, y_s\}$ für $y_1, \dots, y_s \in \mathbb{R}^n$.

Behauptung:

$$P = \{x \in \mathbb{R}^n : y_1^\top x \leq 1, \dots, y_s^\top x \leq 1\}$$

Beweis:

„ $\subseteq$ “: Klar, wegen der Definition der Polaren (**Definition 6.5.1**)².

„ $\supseteq$ “: Sei $x \in \mathbb{R}^n$ mit $y_1^\top x \leq 1, \dots, y_s^\top x \leq 1$. Zu zeigen: $x \in P$.

Angenommen $x \notin P$, dann existiert eine Trennhyperebene, die $\{x\}$ und P trennt:

$$\exists c \neq 0, \delta \in \mathbb{R} \text{ mit } c^\top x' < \delta \quad \forall x' \in P, \quad c^\top x > \delta$$

Da $0 \in P$ ist $0 < \delta$. Ohne Einschränkung sei $\delta = 1$ (durch Skalierung von c).

Dann ist $c \in P^*$ und somit

$$c = \sum_{j=1}^s \alpha_j y_j, \quad \alpha_j \geq 0, \quad \sum_{j=1}^s \alpha_j = 1$$

¹ $B(x_0, r) \subseteq P \xRightarrow{(2)} P^* \subseteq B(x_0, r)^* = (rB_n)^* \xRightarrow{(1)} P^* \subseteq \frac{1}{r}B_n^* \xRightarrow{(3)} P^* \subseteq \frac{1}{r}B_n$

² $x \in P, y_i^\top x \leq 1 \quad \forall i = 1, \dots, s$, weil $y_i \in P^*$

$$\begin{aligned}
1 &< c^T x \\
&= \left(\sum_{j=1}^s \alpha_j y_j \right)^T x \\
&= \sum_{j=1}^s \alpha_j \underbrace{(y_j^T x)}_{\leq 1} \\
&\leq \sum_{j=1}^s \alpha_j = 1 \quad \nexists
\end{aligned}$$

$\Rightarrow x \in P.$

□

Beispiel 6.5.1:

$$\begin{aligned}
P &= \left\{ x \in \mathbb{R}^3 : \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & 1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \\ -1 & -1 & -1 \end{pmatrix} x \leq \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right\} \\
&= \text{conv}\{\pm e_1, \pm e_2, \pm e_3\} \\
&= \left\{ x \in \mathbb{R}^3 : \|x\|_1 = |x_1| + |x_2| + |x_3| \leq 1 \right\}
\end{aligned}$$

Frage: Wie sehen allgemeine Polyeder aus, die eventuell nicht beschränkt sind?

Exkurs:
SoSe 2015

Definition 6.5.2 (konische Kombination):

Seien $v_1, \dots, v_N \in \mathbb{R}^n$ Vektoren. Dann ist $w \in \mathbb{R}^n$ eine **konische Kombination** von $v_1, \dots, v_N$, falls

$$w = \sum_{i=1}^N \alpha_i v_i, \quad \alpha_i \geq 0$$

Definition 6.5.3 (konvexer, spitzer, ordentlicher Kegel):

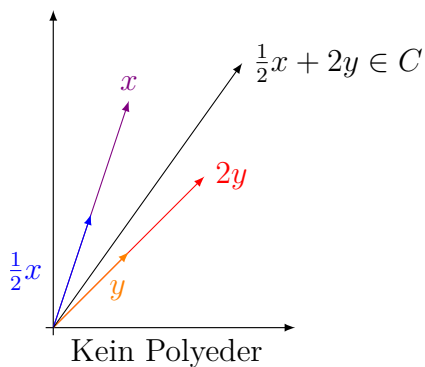
1. Eine Menge $K \subseteq \mathbb{R}^n$, $K \neq \emptyset$ heißt **(konvexer) Kegel**, falls K unter der Bildung von konischen Kombinationen abgeschlossen ist.
2. Ein konvexer Kegel $K \subseteq \mathbb{R}^n$ heißt **spitz**, falls aus $x, -x \in K$ stets $x = 0$ folgt.
3. Ein konvexer Kegel $K \subseteq \mathbb{R}^n$ heißt **ordentlich**, falls er spitz, abgeschlossen (in der Topologie von $\mathbb{R}^n$) und volldimensional ($\dim K = n$) ist.

Ordentliche Kegel sind **Positivitätsbereiche** und definieren eine **partielle Ordnung** auf dem $\mathbb{R}^n$:

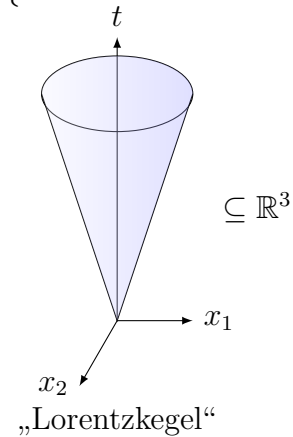
$$x \geq_K y \Leftrightarrow x - y \in K$$

- Falls $K = [0, \infty) \subseteq \mathbb{R}$, dann ist $x \geq_K y$ einfach $x \geq y$
- Es kann passieren, dass weder $x \geq_K y$ noch $y \geq_K x$ gilt. In diesem Fall sind x und y nicht vergleichbar bezüglich K (*partielle* Ordnung).

Beispiel 6.5.2:



$$\mathcal{L}^{n+1} = \{(x, t) \in \mathbb{R}^{n+1} : \|x\| \leq t\}$$

**Bemerkung 6.5.1:**

Ein konvexer Kegel ist tatsächlich konvex.

Definition 6.5.4 (konische Hülle):

Sei $A \subseteq \mathbb{R}^n$. Dann ist die **konische Hülle** von A definiert als

$$\text{cone } A = \bigcap_{\substack{B \supseteq A \\ B \text{ konvexer Kegel}}} B$$

(Analog zur Definition 6.2.5 der konvexen Hülle.)

Satz 6.5.1:

Sei $A \subseteq \mathbb{R}^n$. Dann gilt

$$\text{cone } A = \left\{ w \in \mathbb{R}^n \mid \exists N \in \mathbb{N}, \alpha_1, \dots, \alpha_N \geq 0, v_1, \dots, v_N \in A : w = \sum_{i=1}^N \alpha_i v_i \right\}$$

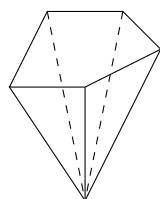
(Analog zum Satz 6.2.2.)

Definition 6.5.5 (endlich erzeugter Kegel):

Eine Menge $K \subseteq \mathbb{R}^n$ heißt **endlich erzeugter Kegel**, falls es Vektoren $v_1, \dots, v_N \in \mathbb{R}^n$ gibt, so dass

$$K = \text{cone}\{v_1, \dots, v_N\}$$

Beispiel 6.5.3:



Für endlich erzeugt muss der Schnitt mit der Hyperebene ein Polyeder bzw. Polytop sein

Satz 6.5.2 (Minkowski-Weyl für Kegel):

Ein konvexer Kegel $K \subseteq \mathbb{R}^n$ ist endlich erzeugt
 $\iff \exists A \in \mathbb{R}^{m \times n}: K = \{x \in \mathbb{R}^n: Ax \leq 0\}$

Beweis:

[Siehe 3, Kapitel 7].

□

30.05.2014
15. Vorlesung**Theorem 6.5.2:**Sei $P \subseteq \mathbb{R}^n$.

$$P \text{ ist Polyeder} \\ \iff \exists x_1, \dots, x_t, v_1, \dots, v_s \in \mathbb{R}^n: P = \underbrace{\text{conv}\{x_1, \dots, x_t\}}_{\text{Polytop}} + \underbrace{\text{cone}\{v_1, \dots, v_s\}}_{\text{konische Hülle}}$$

Wobei hier für Mengen $A, B \subseteq \mathbb{R}^n$ die **Minkowski-Summe** gemeint ist

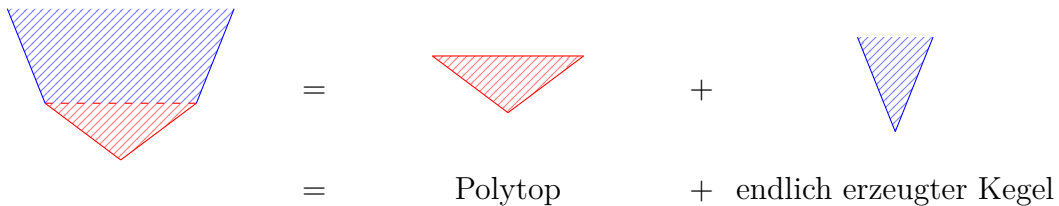
$$A + B = \{a + b: a \in A, b \in B\}$$

Beweis:

[Siehe 3, Kapitel 7].

□

Beispiel 6.5.4:



$$= \text{Polytop} + \text{endlich erzeugter Kegel}$$

6.6. Lemma von Farkas

Satz 6.6.1 ist bereits aus der Linearen Algebra bekannt.

Satz 6.6.1:Für das lineare Gleichungssystem $Ax = b$, mit $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$ und $b \in \mathbb{R}^m$ gilt:

$$\exists x \in \mathbb{R}^n: Ax = b \iff \nexists y \in \mathbb{R}^m: y^T A = 0 \text{ und } y^T b \neq 0$$

Beweis:„ $\Rightarrow$ “: Angenommen $Ax = b$ und $\exists y: y^T A = 0$ und $y^T b \neq 0$. Dann

$$0 = y^T \underbrace{Ax}_{=b} = y^T b \neq 0 \quad \text{!}$$

„ $\Leftarrow$ “: Angenommen $Ax = b$ hat keine Lösung. Dann $\text{rang}(A | b) > \text{rang } A$.Wähle $y \in \text{Kern } A^T$ mit $y^T b \neq 0$.

□

Analoges Resultat für Systeme linearer Ungleichungen $\Rightarrow$ Lemma von Farkas.

Satz 6.6.2 (Lemma von Farkas):

Seien $A \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$. Dann

$$\exists x \in \mathbb{R}^n, x \geq 0: Ax = b \iff \nexists y \in \mathbb{R}^m: y^\top A \geq 0 \text{ und } y^\top b < 0$$

Beweis:

„ $\Rightarrow$ “: Angenommen $\exists x \geq 0$ mit $Ax = b$ und angenommen $\exists y: y^\top A \geq 0$ und $y^\top b < 0$.

$$\text{Dann } y^\top Ax = y^\top b < 0, \text{ aber } \underbrace{y^\top A}_{\geq 0} \underbrace{x}_{\geq 0} \geq 0 \quad \nexists$$

„ $\Leftarrow$ “: Angenommen $\nexists x \geq 0$ mit $Ax = b$.

Seien $a_1, \dots, a_n \in \mathbb{R}^m$ die Spalten von A . Dann ist $b \notin \text{cone}\{a_1, \dots, a_n\}$. Also existiert eine Trennhyperebene von $\{b\}$ und $\text{cone}\{a_1, \dots, a_n\}$:

$$\exists c \neq 0: c^\top b < 0 \text{ und } c^\top x \geq 0 \forall x \in \text{cone}\{a_1, \dots, a_n\}$$

Also erfüllt $y = c$ die Ungleichungen $y^\top A \geq 0$ und $y^\top b < 0$.

□

Interpretation des Lemma von Farkas: Theorem der Alternativen:

Entweder $\exists x \geq 0: Ax = b$
oder $\exists y: y^\top A \geq 0, y^\top b < 0$

Korollar 6.6.2.a (Variante des Lemma von Farkas):

$$\exists x \in \mathbb{R}^n: Ax \leq b \iff \nexists y \geq 0: y^\top A = 0 \text{ und } y^\top b < 0$$

Beweis:

Betrachte die Matrix $A' = \begin{pmatrix} A & -A & I \end{pmatrix} \in \mathbb{R}^{m \times (n+n+m)}$. Sei $x' = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in \mathbb{R}^{n+n+m}$ mit $x_1, x_2 \in \mathbb{R}^n$ und $x_3 \in \mathbb{R}^m$.

$$\begin{aligned} \exists x' \geq 0: A'x' = b &\Leftrightarrow \exists x_1, x_2, x_3 \geq 0: Ax_1 - Ax_2 + x_3 = b \\ &\Leftrightarrow \exists x_1, x_2, x_3 \geq 0: A(x_1 - x_2) + x_3 = b \\ &\Leftrightarrow \exists x \in \mathbb{R}^n: Ax \leq b \end{aligned}$$

Wende das Lemma von Farkas auf $A'x' = b$ an:

$A'x' = b$ hat Lösung $x' \geq 0 \Leftrightarrow \nexists y \in \mathbb{R}^m: y^\top \begin{pmatrix} A & -A & I \end{pmatrix} \geq 0 \in \mathbb{R}^{n+n+m}$ und $y^\top b < 0$.
 Dabei

$$\begin{aligned} y^\top \begin{pmatrix} A & -A & I \end{pmatrix} \geq 0 &\Leftrightarrow \begin{aligned} y^\top A &\geq 0 \\ y^\top (-A) &\geq 0 \\ y^\top I &\geq 0 \end{aligned} \\ &\Leftrightarrow \begin{aligned} y^\top A &\leq 0 \\ y^\top &\geq 0 \end{aligned} \end{aligned}$$

□

Korollar 6.6.2.b:

Angenommen $Ax \leq b$ hat eine Lösung, dann gilt

$$c^T x \leq \delta \quad \forall x: Ax \leq b \iff \exists y \geq 0: y^T A = c^T, y^T b \leq \delta$$

Beweis:

„ $\Leftarrow$ “: $Ax \leq b \Rightarrow y^T Ax \leq y^T b \Rightarrow c^T x \leq \delta$.

„ $\Rightarrow$ “: Angenommen ein solches y gibt es nicht. Dann hat die lineare Gleichung

$$\begin{pmatrix} y^T & \lambda \end{pmatrix} \begin{pmatrix} A & b \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} c^T & \delta \end{pmatrix}$$

keine Lösung mit $\begin{pmatrix} y^T & \lambda \end{pmatrix} \geq \begin{pmatrix} 0 & 0 \end{pmatrix}$. Nach dem Lemma von Farkas $\exists \begin{pmatrix} z \\ \mu \end{pmatrix} \in \mathbb{R}^{n+1}$

mit $\begin{pmatrix} A & b \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z \\ \mu \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ und $\begin{pmatrix} c^T & \delta \end{pmatrix} \begin{pmatrix} z \\ \mu \end{pmatrix} < 0$.

1. Fall: $\mu = 0$:

Dann $Az \geq 0$ und $c^T z < 0$. Nach Voraussetzung hat $Ax \leq b$ eine Lösung x_0 . Dann ist für $\tilde{c}$ groß: $A(x_0 - \tilde{c}z) \leq b$ und $c^T(x_0 - \tilde{c}z) > \delta$. $\nexists$

2. Fall: $\mu > 0$:

Ohne Einschränkung sei $\mu = 1$ (durch skalieren). Dann gilt $Az + b \geq 0$ und $c^T z + \delta < 0$. Also $A(-z) \leq b$ und $c^T(-z) > \delta$. $\nexists$

□

6.7. Lineare Programmierung

04.06.2014
16. Vorlesung

Lineares Programm in Standardform:

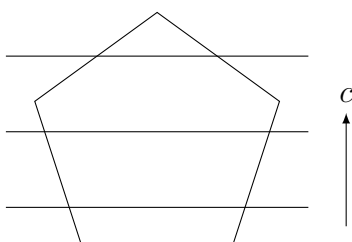
$$\begin{aligned} &\max c^T x \\ &\text{wobei } x \in \mathbb{R}^n \\ &Ax \leq b \end{aligned} \quad (\text{LP})$$

wobei $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$ gegeben sind (siehe Definition 6.1.1).

Geometrische Interpretation

Maximiere die lineare Funktion $x \rightarrow c^T x$ über das Polyeder

$$P = \{x \in \mathbb{R}^n: Ax \leq b\}$$



Das heißt schiebe die zu c orthogonale Hyperebene soweit wie möglich in Richtung c , so dass sie noch P schneidet.

Sprechweise:

- x ist **zulässige** oder **gültige Lösung** $\Leftrightarrow x \in P$
- x ist **optimale Lösung** $\Leftrightarrow x \in P, c^\top x = \max\{c^\top y : y \in P\}$
- (LP) heißt **unbeschränkt** $\Leftrightarrow \sup\{c^\top x : x \in P\} = +\infty$
- (LP) heißt **unzulässig** oder **ungültig** $\Leftrightarrow \nexists x \in P$
 $\Leftrightarrow \sup\{c^\top x : x \in P\} = -\infty$

Satz 6.7.1:

Sei P ein Polytop mit $P \neq \emptyset$. Dann existiert eine Ecke von P , die eine optimale Lösung von (LP) ist.

Beweis:

Seien $x_1, \dots, x_t$ die Ecken von P . Nach dem Theorem von Minkowski ist $P = \text{conv}\{x_1, \dots, x_t\}$. Angenommen keine Ecke ist eine optimale Lösung, dann gilt, unter der Berücksichtigung das P kompakt ist:

$$\exists y \in P: c^\top y > c^\top x_i \quad \forall i = 1, \dots, t$$

Schreibe $y = \sum_{i=1}^t \alpha_i x_i$ mit $\alpha_i \geq 0$ und $\sum_{i=1}^t \alpha_i = 1$. Dann ist

$$c^\top y = c^\top \left(\sum_{i=1}^t \alpha_i x_i \right) = \sum_{i=1}^t \alpha_i c^\top x_i < \sum_{i=1}^t \alpha_i c^\top y = c^\top y$$

Widerspruch. □

Offensichtlicher (aber eventuell sehr langsamer) Algorithmus zur Lösung von (LP), falls P beschränkt ist:

Bestimme alle Ecken $x_1, \dots, x_t$ von P und finde den Index i_0 mit $c^\top x_{i_0} \geq c^\top x_i \quad \forall i = 1, \dots, t$.

Problem: t kann sehr groß werden, z. B. beim Würfel:

$$C_n = \{x \in \mathbb{R}^n : -1 \leq x_i \leq 1 \quad \forall i = 1, \dots, n\}$$

Dies sind $2n$ Ungleichungen, aber 2^n Ecken.

Algorithmen zum Lösen von linearen Programmen

- Fourier-Motzkin-Elimination $\rightsquigarrow$ Kapitel 10
- Simplexverfahren (Dantzig, 1947) $\rightsquigarrow$ Kapitel 11
- Ellipsoid-Methode (Khachiyan, 1979) $\rightsquigarrow$ Kapitel 12
- Innere-Punkte-Methode (Karmarkar, 1984) $\rightsquigarrow$ Kapitel 13
- Methode von Chubanov (Chubanov, 2010)

Definition 6.7.1 (primales und duales lineares Programm):

Seien $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$ gegeben. Wir definieren das **primale LP**

$$\begin{aligned} p^* &= \max c^\top x \\ \text{wobei } x &\in \mathbb{R}^n \\ Ax &\leq b \end{aligned} \quad (\text{PLP})$$

und das **duale LP**

$$\begin{aligned} d^* &= \min y^\top b \\ \text{wobei } y &\in \mathbb{R}^m \\ y &\geq 0 \\ y^\top A &= c^\top \end{aligned} \quad (\text{DLP})$$

Lemma 6.7.1:

Sei $\delta = \sup\{c^\top x : Ax \leq b\}$. Falls $\delta \in (-\infty, +\infty)$, dann existiert eine optimale Lösung, d. h. es existiert ein x , so dass

$$Ax \leq b, \quad c^\top x \geq \delta$$

Beweis:

Zu zeigen: $\exists x: Ax \leq b, \quad c^\top x \geq \delta$. Angenommen ein solches x existiert nicht, d. h. $\nexists x: \begin{pmatrix} A \\ -c^\top \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix}$. Nach dem Lemma von Farkas $\exists \begin{pmatrix} y \\ \lambda \end{pmatrix} \geq 0: \begin{pmatrix} y^\top & \lambda \end{pmatrix} \begin{pmatrix} A \\ -c^\top \end{pmatrix} = 0$ und $\begin{pmatrix} y^\top & \lambda \end{pmatrix} \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0$. Dann

$$\begin{aligned} \lambda \delta &= \sup\{\lambda c^\top x : Ax \leq b\} \\ &= \sup\{y^\top Ax : Ax \leq b\} \\ &\leq y^\top b \\ &< \lambda \delta \quad \nexists \end{aligned}$$

□

Satz 6.7.2 (Optimalitätsbedingung):

Seien x und y zulässig. Dann ist x optimal für (PLP) und y optimal für (DLP) genau dann, wenn $y^\top (Ax - b) = 0$.

Insbesondere gelten dann die **Komplementaritätsbedingungen**:

$$\begin{aligned} y_j \neq 0 &\Rightarrow (Ax - b)_j = 0 & \forall j = 1, \dots, m \\ (Ax - b)_j \neq 0 &\Rightarrow y_j = 0 & \forall j = 1, \dots, m \end{aligned}$$

Beweis:

Übung.

□

Theorem 6.7.1 (Dualität linearer Programme, von Neumann, 1947):

Für das primale lineare Programm $p^* = \max\{c^\top x : Ax \leq b\}$ und das duale lineare Programm $d^* = \min\{y^\top b : y \geq 0, y^\top A = c^\top\}$ gilt, falls beide Mengen gültiger Lösungen nicht leer sind:

$$p^* = d^*$$

Beweis:

„ $p^* \leq d^*$ “: (schwache Dualität)

Sei x zulässig für (PLP) und y zulässig für (DLP), dann ist $c^\top x \leq y^\top b$.

Insbesondere: $p^* \leq d^*$.

Beweis: $y^\top b - c^\top x \geq y^\top (Ax) - (y^\top A)x = 0$.

Jede gültige Lösung des (DLP) gibt also eine obere Schranke für das (PLP) und umgekehrt gibt jede gültige Lösung des (PLP) eine untere Schranke für das (DLP).

„ $p^* \geq d^*$ “: (starke Dualität)

Behauptung: $\exists x: Ax \leq b$ und $c^\top x = p^*$.

Beweis: Da (DLP) gültig ist folgt aus der schwachen Dualität:

$p^* = \sup\{c^\top x : Ax \leq b\} < +\infty$. Genauso: $\inf\{y^\top b : y \geq 0, y^\top A = c^\top\} \geq -\infty$.

Nach **Lemma 6.7.1** existiert eine optimale Lösung x für $\delta = p^*$.

Behauptung: $\exists y \geq 0: y^\top A = c^\top$ und $y^\top b \leq p^*$.

Beweis: Setze $\delta = p^*$ in **Korollar 6.6.2.b**.

□

Korollar 6.7.2.a:

Falls beide Mengen gültiger Lösungen nicht leer sind, dann gilt

$$\max\{c^\top x : x \geq 0, Ax = b\} = \min\{y^\top b : y^\top A \geq c^\top\}$$

Beweis:

Setze $\tilde{A} = \begin{pmatrix} A \\ -A \\ -I \end{pmatrix}$ und $\tilde{b} = \begin{pmatrix} b \\ -b \\ 0 \end{pmatrix}$. Dann

$$\begin{aligned} \max\{c^\top x : x \geq 0, Ax = b\} &= \max\{c^\top x : \tilde{A}x \leq \tilde{b}\} \\ &= \min\{z^\top \tilde{b} : z \geq 0, z^\top \tilde{A} = c^\top\} \\ &\stackrel{z=(u,v,w)^\top}{=} \min\{u^\top b - v^\top b + w^\top 0 : u, v, w \geq 0, u^\top A - v^\top A - w^\top = c^\top\} \\ &\stackrel{y=u-v}{=} \min\{y^\top b : y^\top A \geq c^\top\} \end{aligned}$$

□

7. Ganzzahlige Optimierung und vollständig-unimodulare Matrizen

06.06.2014
17. Vorlesung

7.1. Ganzzahlige lineare Programme	67
7.2. Vollständig-unimodulare Matrizen	69
7.3. Vollständig-unimodulare Matrizen und bipartite Graphen	71
7.4. Vollständig-unimodulare Matrizen und gerichtete Graphen	74

7.1. Ganzzahlige lineare Programme

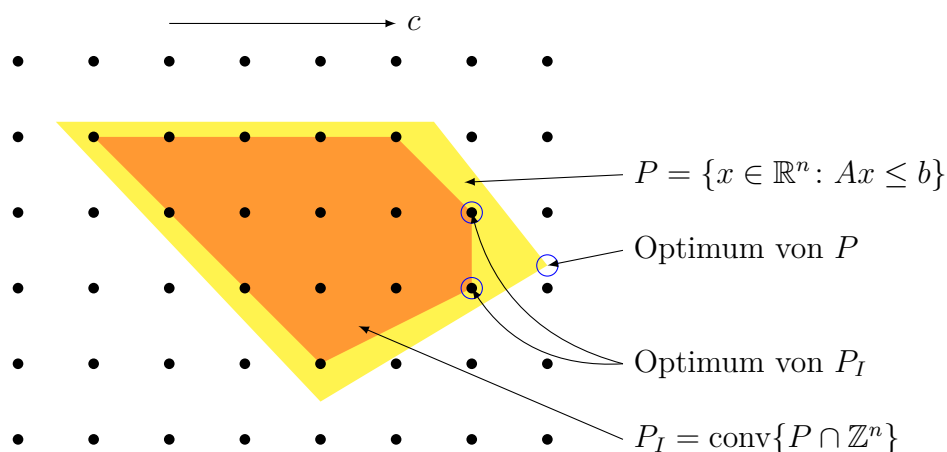
Viele Optimierungsprobleme im Operations Research lassen sich als ILP (integer linear programm) formulieren.

Definition 7.1.1 (ganzzahliges lineares Programm):

Ein **ganzzahliges lineares Programm (ILP oder kurz IP)** in Standardform ist

$$\begin{aligned} \max \quad & c^T x \\ & x \in \mathbb{Z}^n \text{ (also **ganzzahlig**)} \\ & Ax \leq b \end{aligned} \tag{IP}$$

Maximiere eine lineare Funktion über die ganzzahligen Punkte in einem Polyeder.



Beispiel 7.1.1:

Sei $G = (V, E)$ ein ungerichteter Graph und $A \in \mathbb{R}^{V \times E}$ die Inzidenzmatrix definiert durch

$$A_{v,e} = \begin{cases} 1, & \text{falls } v \in e \\ 0, & \text{sonst} \end{cases}$$

Dann ist

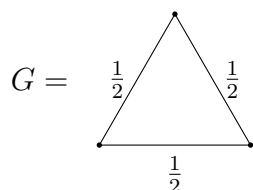
$$\begin{aligned} \nu(G) &= \max \left\{ |M| : M \subseteq E \text{ Matching in } G \right\} \\ &= \max \left\{ \sum_{e \in E} x_e : x_e \geq 0 \forall e \in E, x_e \in \mathbb{Z} \forall e \in E, \sum_{e: e \ni v} x_e \leq 1 \forall v \in V \right\} \\ &= \max \left\{ \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}^\top x : x \geq 0, x \in \mathbb{Z}^E, Ax \leq \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \right\} \end{aligned} \quad (7.1)$$

Wir haben in Abschnitt 4.3 gesehen, dass

$$(7.1) = \max \left\{ \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}^\top x : x \geq 0, x \in \mathbb{R}^E, Ax \leq \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \right\}$$

gilt, falls G *bipartit* ist. Im Allgemeinen, nicht bipartiten Fall, gilt nur „<“, wie das folgende Beispiel 7.1.2 zeigt.

Beispiel 7.1.2:



Hier ist das Maximum im IP gleich 1, aber das Maximum im LP ist $\geq \frac{3}{2}$.

Dualitätsbeziehung

Falls beide Programme zulässige Lösungen haben gilt

$$\begin{aligned} \max \{ c^\top x : Ax \leq b, x \in \mathbb{Z}^n \} &\leq \max \{ c^\top x : Ax \leq b, x \in \mathbb{R}^n \} \\ &= \min \{ y^\top b : y \geq 0, y^\top A = c^\top, y \in \mathbb{R}^m \} \\ &\leq \min \{ y^\top b : y \geq 0, y^\top A = c^\top, y \in \mathbb{Z}^m \} \end{aligned}$$

Im Allgemeinen gilt „<“. Kein wunder: LP kann effizient in polynomieller Zeit gelöst werden. Dagegen ILP wahrscheinlich nicht. Dieses Problem ist äquivalent zum $\mathcal{P}$ - $\mathcal{NP}$ -Problem, auf dessen Lösung 1.000.000 \$ Belohnung ausgelobt sind.

Frage: In welcher Situation gilt „=“?

Definition 7.1.2 (ganzzahliges Polytop):

Ein Polytop $P \subseteq \mathbb{R}^n$ heißt **ganzzahlig**, falls jede Ecke z von P ganzzahlig ist, d. h. $z \in \mathbb{Z}^n$.

Falls P ganzzahlig ist, gilt

$$\max\{c^\top x : x \in P\} = \max\{c^\top x : x \in P, x \in \mathbb{Z}^n\}$$

Frage: Welche Bedingungen müssen für A und b gelten, damit $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ ganzzahlig ist?

7.2. Vollständig-unimodulare Matrizen

Definition 7.2.1 (vollständig-unimodulare Matrix):

$A \in \mathbb{R}^{m \times n}$ heißt **vollständig-unimodular**, falls jeder der Minoren, d. h. die Determinante einer quadratischen Teilmatrix, von A gleich -1 , 0 oder $+1$ ist.

Insbesondere: $A_{i,j} \in \{-1, 0, +1\}$.

Diese Eigenschaft bleibt erhalten, wenn ein Null- oder Einheitsvektor an die Matrix angefügt wird.

$$A \text{ vollständig-unimodular} \Rightarrow \begin{pmatrix} A & & \\ 0 & \dots & 0 \end{pmatrix} \text{ vollständig-unimodular}$$

$$A \text{ vollständig-unimodular} \Rightarrow \begin{pmatrix} A \\ e_i^\top \end{pmatrix} \text{ vollständig-unimodular}$$

Satz 7.2.1:

Sei $A \in \mathbb{R}^{m \times n}$ vollständig-unimodular, d. h. $A \in \{-1, 0, 1\}^{m \times n}$, und sei $b \in \mathbb{Z}^m$. Dann ist jede Ecke des Polyeders $P = \{x : Ax \leq b\}$ ganzzahlig.

Beweis:

Sei $z \in P$ eine Ecke. Dann hat die Teilmatrix A_z (siehe Abschnitt 6.4) vollen Rang n und sie enthält eine reguläre $n \times n$ Teilmatrix A' , d. h. $\text{rang } A' = n$.

Ohne Einschränkung enthalten die ersten n Zeilen von A diese Teilmatrix A' . Setze

$$b' = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \in \mathbb{Z}^n. \text{ Dann gilt } A'z = b' \text{ und } z \in \mathbb{Z}^n, \text{ weil } \det A' = \{\pm 1\}. \text{ Nach der Cramerschen}$$

Regel gilt:

$$z_1 = \frac{\det \begin{pmatrix} b' & A'_2 & \dots & A'_n \end{pmatrix}}{\det A'}$$

Da durch 1 oder -1 geteilt wird ist z_1 ganzzahlig. Analog sind $z_2, \dots, z_n$ ebenfalls ganzzahlig.

□

Bemerkung 7.2.1:

Nicht jedes Polyeder hat eine Ecke.

18.06.2014
18. Vorlesung

Definition 7.2.2 (ganzzahliger Polyeder):

Ein Polyeder $P \subseteq \mathbb{R}^n$ heißt **ganzzahlig**, falls für alle $c \in \mathbb{R}^n$ für die $\max\{c^\top x : x \in P\}$ endlich ist, das Maximum an einem ganzzahligen Vektor angenommen wird.

Satz 7.2.2:

Sei $A \in \mathbb{R}^{m \times n}$ vollständig-unimodular, d. h. $A \in \{-1, 0, 1\}^{m \times n}$, und $b \in \mathbb{Z}^m$. Dann ist $P = \{x: Ax \leq b\}$ ein ganzzahliger Polyeder.

Beweis:

Sei $c \in \mathbb{R}$ gegeben und x^* optimale Lösung von $\max \{c^\top x: Ax \leq b\}$. Wähle $d', d'' \in \mathbb{Z}^n$ mit $d' \leq x^* \leq d''$. Dann ist $Q = \{x \in \mathbb{R}^n: Ax \leq b, d' \leq x \leq d''\}$ ein Polytop. Es ist

$$Q = \left\{ x \in \mathbb{R}^n: \begin{pmatrix} A \\ -I \\ I \end{pmatrix} x \leq \begin{pmatrix} b \\ -d' \\ d'' \end{pmatrix} \right\}$$

und die Matrix $\begin{pmatrix} A \\ -I \\ I \end{pmatrix}$ ist vollständig-unimodular. Dann ist Q nach Satz 7.2.1 ein ganzzahliges Polytop und $\max \{c^\top x: x \in Q\}$ wird an einer ganzzahligen Ecke $\tilde{x}$ angenommen. Dann ist $\tilde{x}$ auch eine optimale Lösung von $\max \{c^\top x: Ax \leq b\}$, weil $A\tilde{x} \leq b$ und $c^\top \tilde{x} \geq c^\top x^*$, da $x^* \in Q$. □

Korollar 7.2.2.a:

Sei $A \in \mathbb{R}^{m \times n}$ vollständig-unimodular, d. h. $A \in \{-1, 0, 1\}^{m \times n}$, $b \in \mathbb{Z}^m$ und $c \in \mathbb{Z}^n$. Dann haben die beiden linearen Programme

$$\max \{c^\top x: Ax \leq b\} = \min \{y^\top b: y \geq 0, y^\top A = c^\top\}$$

ganzzahlige optimale Lösungen, falls beide Optima endlich sind.

Beweis:

Folgt aus Satz 7.2.2, weil die Matrix $\begin{pmatrix} -I & A^\top & -A^\top \end{pmatrix}^\top \in \mathbb{Z}^{(2n+m) \times m}$ vollständig-unimodular ist. □

Ziel: Charakterisierung von vollständig-unimodular mit Hilfe von Ganzzahligkeit von Polyedern.

Satz 7.2.3 (Hoffman, Kruskal, 1956):

$$A \in \mathbb{Z}^{m \times n} \text{ ist vollständig-unimodular} \\ \iff \forall b \in \mathbb{Z}^m: P = \{x: Ax \leq b, x \geq 0\} \text{ ist ganzzahlig}$$

Beweis:

„ $\Rightarrow$ “: Weil $P \subseteq \{x: x \geq 0\}$ wird $\max \{c^\top x: x \in P\}$ in einer Ecke von P angenommen, falls das Maximum endlich ist.

Sei z eine Ecke von P . Betrachte $B = \begin{pmatrix} A \\ -I \end{pmatrix}$, $Bx \leq \begin{pmatrix} b \\ 0 \end{pmatrix}$. Dann enthält B_z eine reguläre Teilmatrix $B' \in \mathbb{Z}^{n \times n}$. Sei $b' \in \mathbb{Z}^n$ der entsprechende Teilvektor von $\begin{pmatrix} b \\ 0 \end{pmatrix}$. Dann ist $z = (B')^{-1}b' \in \mathbb{Z}^n$ aufgrund der Cramerschen Regel und weil A vollständig-unimodular ist.

„ $\Leftarrow$ “: Sei A' reguläre $k \times k$ Teilmatrix von A .

Es ist zu zeigen: $|\det A'| = 1$. Ohne Einschränkung ist A' obere linke Teilmatrix von A . Betrachte $B \in \mathbb{Z}^{m \times m}$ bestehend aus den ersten k und den letzten $m - k$ Spalten von $\begin{pmatrix} A & I_m \end{pmatrix}$. Dann $|\det A'| = |\det B|$.

$$\left(A \mid I_m \right) = \left(\begin{array}{c|c|c} \overset{k}{A'} & & \overset{k}{I_k} \\ \hline & \overset{n-k}{I_{n-k}} & \overset{m-k}{I_{m-k}} \end{array} \right) \begin{array}{c} \overset{k}{k} \\ \hline \overset{m-k}{m-k} \end{array}$$

Ziel: Zeige, dass $B^{-1} \in \mathbb{Z}^{m \times m}$. Dann ist $|\det B^{-1}| \in \mathbb{Z}$. Weil $\det B \cdot \det B^{-1} = 1$ muss $|\det B| = 1$ sein.

Sei $i \in \{1, \dots, m\}$. Wir zeigen, dass die i -te Spalte von B^{-1} ganzzahlig ist. Wähle $y \in \mathbb{Z}^m$ mit $y + B^{-1}e_i \geq 0$. Setze $z = y + B^{-1}e_i$. Dann ist $Bz = By + e_i \in \mathbb{Z}^m$. Setze $b = Bz$. Füge zu z hinter den ersten k Komponenten $n - k + k$ Nullen hinzu und nenne diesen Vektor z' . Dann gilt nach Konstruktion

$$\begin{pmatrix} A & I \end{pmatrix} z' = Bz = b$$

Da $z' \geq 0$ ist, liegt der Vektor z'' , der aus den ersten n Komponenten von z' besteht, in $P = \{x : Ax \leq b, x \geq 0\}$. Es gilt also

$$\begin{pmatrix} A \\ -I \end{pmatrix} z'' \leq \begin{pmatrix} b \\ 0 \end{pmatrix}$$

Die ersten k und die letzten $n - k$ dieser Ungleichungen sind mit Gleichheit erfüllt. Die entsprechenden Zeilen von $\begin{pmatrix} A \\ -I \end{pmatrix}$ sind linear unabhängig, also ist z'' eine Ecke von P . Nach Voraussetzung ist $z'' \in \mathbb{Z}^n$. Dann ist auch $z' \in \mathbb{Z}^{n+m}$, weil auf den ersten n Komponenten z' und z'' übereinstimmen. Die letzten m Komponenten von z' sind durch $b - Az'' \in \mathbb{Z}^m$ gegeben und sind ebenfalls ganzzahlig. Also ist $z \in \mathbb{Z}^m$ und damit auch $B^{-1}e_i = z - y \in \mathbb{Z}^m$.

□

7.3. Vollständig-unimodulare Matrizen und bipartite Graphen

Satz 7.3.1:

Sei $G = (V, E)$ ein ungerichteter Graph.

G ist bipartit $\iff$ Inzidenzmatrix $A \in \mathbb{R}^{V \times E}$ ist vollständig-unimodular

Beweis:

„ $\Rightarrow$ “: Sei B eine $t \times t$ Teilmatrix von A .

Zeige: $\det B \in \{-1, 0, +1\}$ per Induktion über t .

$t = 1$: ✓

$t > 1$: 1. Fall: B enthält eine Nullspalte $\Rightarrow \det B = 0$

2. Fall: B enthält eine Spalte, die genau eine 1 enthält. Dann (nach eventuellen Permutationen von Zeilen und Spalten)

$$B = \begin{pmatrix} 1 & b^T \\ 0 & B' \end{pmatrix}, \quad b \in \mathbb{R}^{t-1}, \quad B' \in \mathbb{R}^{(t-1) \times (t-1)}$$

Nach Induktionsvoraussetzung gilt $\det B' \in \{-1, 0, +1\}$. Also auch $\det B \in \{-1, 0, +1\}$

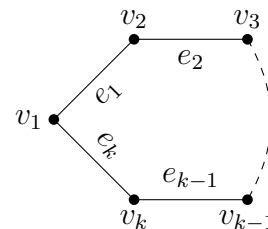
3. Fall: Jede Spalte von B enthält genau zwei Einsen. Da G bipartit ist, kann man (nach eventueller Permutation der Zeilen) B als

$$B = \begin{pmatrix} B' \\ B'' \end{pmatrix}$$

schreiben, wobei jede Spalte von B' genau eine Eins enthält und jede Spalte von B'' genau eine Eins enthält. Aufaddieren aller Zeilen von B' ergibt den Vektor $(1 \ \cdots \ 1)$. Genauso das Aufaddieren aller Zeilen von B'' , d.h. die Zeilen von B sind linear abhängig $\Rightarrow \det B = 0$.

„ $\Leftarrow$ “: Sei A vollständig-unimodular. Angenommen der Graph G ist nicht bipartit. Dann enthält G einen Kreis ungerader Länge mit Knoten $v_1, \dots, v_k$ und Kanten $e_1, \dots, e_k$. Die entsprechende Teilmatrix von A (mit Zeilen $v_1, \dots, v_k$ und Spalten $e_1, \dots, e_k$) ist von der Form:

$$\begin{array}{c} e_1 \ e_2 \ e_3 \ \cdots \ e_{k-2} \ e_{k-1} \ e_k \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{k-2} \\ v_{k-1} \\ v_k \end{array} \end{array} \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 1 \end{pmatrix}$$



Durch z. B. Entwicklung nach der ersten Spalte sieht man, dass die Determinante dieser Matrix gleich $(+1) \cdot 1 + (+1) \cdot 1 = 2$ ist (für ungerades k). Also kann A nicht vollständig-unimodular sein. □

Korollar 7.3.1.a (Matchingtheorem von König, siehe Korollar 4.3.3.a):

Sei $G = (V, E)$ bipartit. Dann gilt

$$\begin{aligned} \nu(G) &= \max\{|M| : M \subseteq E \text{ Matching in } G\} \\ &= \min\{|U| : U \subseteq V \text{ Knotenüberdeckung von } G\} \\ &= \tau(G) \end{aligned}$$

Beweis:

Die Inzidenzmatrix $A \in \mathbb{R}^{V \times E}$ ist vollständig-unimodular. Nach dem Satz von Hoffman-Kruskal (Satz 7.2.3) ist

$$\begin{aligned} \max\{1^\top x : x \geq 0, Ax \leq 1, x \in \mathbb{R}^E\} &= \max\{1^\top x : x \geq 0, Ax \leq 1, x \in \mathbb{Z}^E\} \\ &= \max \left\{ \sum_{e \in E} x_e : x_e \geq 0 \forall e \in E, \sum_{e: e \in v} x_e \leq 1 \forall v \in V, x_e \in \mathbb{Z} \right\} \\ &= \nu(G) \end{aligned}$$

Dualisieren der LPs:

$$\begin{array}{ll} \max\{dx + ey + fz : x \geq 0, z \leq 0, & = \min\{ua + vb + wc : u \geq 0, w \leq 0, \\ Ax + By + Cz \leq a & uA + vD + wG \geq d \\ Dx + Ey + Fz = b & uB + vE + wH = e \\ Gx + Hy + Kz \geq c\} & uC + vF + wK \leq f\} \end{array}$$

mit $d = 1^\top$, $a = 1$, $A = A$, Rest = 0.

$$\begin{aligned} \text{Also } \max\{1^\top x : x \geq 0, Ax \leq 1, x \in \mathbb{R}^E\} &= \min\{u1 : u \geq 0, uA \geq 1^\top, u \in \mathbb{R}^V\} \\ &= \min \left\{ \sum_{v \in V} y_v : y \geq 0, y \in \mathbb{R}^V, \right. \\ &\quad \left. y_u + y_v \geq 1, \text{ falls } \{u, v\} \in E \right\} \\ &\stackrel{A \text{ ist vollständig-unimodular}}{=} \min\{1^\top y : y \geq 0, y \in \mathbb{Z}^V, y^\top A \geq 1\} \\ &= \tau(G) \end{aligned}$$

Ein ganzzahliger Vektor y , der das Minimum erzielt, hat nur $\{0, 1\}$ -Komponenten. Dann $y_v = 1 \Leftrightarrow v$ ist Knoten in der minimalen Knotenüberdeckung. □

Korollar 7.3.1.b:

Sei G bipartit. Dann

$$\text{conv}\{\chi^M : M \subseteq E \text{ Matching in } G\} = \{x \in \mathbb{R}^E : x \geq 0, Ax \leq 1\}$$

wobei A Inzidenzmatrix von G .

Beweis:

„ $\subseteq$ “: ✓

„ $\supseteq$ “: $Q = \{x \in \mathbb{R}^E, x \geq 0, Ax \leq 1\}$ ist beschränkt und somit ein Polytop. Damit ist Q die konvexe Hülle seiner Ecken. Da A vollständig-unimodular ist, sind alle Ecken von Q ganzzahlig. Jeder ganzzahlige Vektor in Q ist Inzidenzvektor eines Matchings von G . □

Korollar 7.3.1.c (Ganzzahlige Version von Egerváry Satz 4.3.2):

Sei G bipartit und $w: E \rightarrow \mathbb{Z}_{\geq 0}$ eine Gewichtsfunction. Dann ist das maximale Gewicht eines Matchings in G gleich

$$\begin{aligned} \min \quad & \sum_{v \in V} y_v \\ & y \in \mathbb{Z}^V \\ & y_v \geq 0 \text{ für } v \in V \\ & y_u + y_v \geq w(e) \text{ für } e = \{u, v\} \in E \end{aligned}$$

Beweis:

Weil die Inzidenzmatrix $A \in \mathbb{R}^{V \times E}$ vollständig-unimodular ist gilt

$$\max\{w^T x: x \geq 0, Ax \leq 1\} = \min\{y^T 1: y \geq 0, y^T A \geq w\}$$

wobei beide Probleme optimale ganzzahlige Lösungen haben. □

7.4. Vollständig-unimodulare Matrizen und gerichtete Graphen

25.06.2014
20. Vorlesung

Definition 7.4.1 (Inzidenzmatrix):

Sei $D = (V, A)$ ein gerichteter Graph und $M \in \mathbb{R}^{V \times A}$ die **Inzidenzmatrix** von D mit

$$M_{v,a} = \begin{cases} +1, & a \in \delta^-(v) \\ -1, & a \in \delta^+(v) \\ 0, & \text{sonst} \end{cases}$$

Jede Spalte von M enthält genau eine $+1$ und genau eine -1 .

Satz 7.4.1:

Die Inzidenzmatrix M eines gerichteten Graphen ist vollständig-unimodular.

Beweis:

Sei $B \in \mathbb{R}^{t \times t}$ eine $t \times t$ Teilmatrix von M . Es ist zu zeigen, dass $\det B \in \{-1, 0, +1\}$ per Induktion über t .

$t = 1$: ✓

$t > 1$: 1. Fall: B enthält eine Nullspalte. Dann $\det B = 0$.

2. Fall: B enthält eine Spalte, die genau ein Element verschieden von 0 enthält.

Dann ist (nach Vertauschen von Zeilen und Spalten) $B = \begin{pmatrix} \pm 1 & b^T \\ 0 & B' \end{pmatrix}$

mit $b \in \mathbb{R}^{t-1}$ und $B' \in \mathbb{R}^{(t-1) \times (t-1)}$. Nach Induktionsvoraussetzung ist $\det B' \in \{-1, 0, +1\}$. Also $\det B = (\pm 1) \cdot \det B' \in \{-1, 0, +1\}$.

3. Fall: Jede Spalte von B enthält genau 2 Elemente verschieden von 0. Addieren aller Zeilen von B ergibt 0, d. h. $\det B = 0$. □

Bemerkung 7.4.1:

M sei Inzidenzmatrix von $D = (V, A)$. Dann ist $f: A \rightarrow \mathbb{R}$ (bzw. $f \in \mathbb{R}^A$) eine Zirkulation genau dann, wenn $Mf = 0 \in \mathbb{R}^V$. Für $v \in V$ gilt nämlich

$$(Mf)_v = \sum_{a \in \delta^-(v)} f(a) - \sum_{a \in \delta^+(v)} f(a) = 0$$

Folgerung 7.4.1:

$D = (V, A)$ gerichteter Graph, $c: A \rightarrow \mathbb{Z}$, $d: A \rightarrow \mathbb{Z}$. Falls es eine Zirkulation $f: A \rightarrow \mathbb{R}$ mit $c \leq f \leq d$ gibt, dann gibt es auch eine ganzzahlige Zirkulation $f': A \rightarrow \mathbb{Z}$ mit $c \leq f' \leq d$.

Beweis:

Nach Voraussetzung ist das Polytop $P = \{x \in \mathbb{R}^A: Mx = 0, c \leq x \leq d\} \neq \emptyset$. Also hat P eine Ecke f' . Weil M vollständig-unimodular ist, ist auch $\begin{pmatrix} I & -I & M \end{pmatrix}^\top$ vollständig-unimodular. Damit ist f' ganzzahlig, da $c, d \in \mathbb{Z}^A$. □

Korollar 7.4.1.a (max-Flow=min-Cut-Theorem, Theorem 5.1.1):

Sei $D = (V, A)$ ein gerichteter Graph, $s, t \in V$ mit s Start- und t Zielknoten und $c: A \rightarrow \mathbb{Z}_{\geq 0}$. Dann ist der maximale Wert eines s - t -Flusses, der durch c beschränkt ist, gleich der minimalen Kapazität eines s - t -Schnittes.

$$\max_{\substack{f \text{ s-t-Fluss} \\ 0 \leq f \leq c}} \text{value}(f) = \min_{\substack{U \subseteq V \\ s \in U, t \notin U}} c(\delta^+(U))$$

Beweis:

„min $\geq$ max“: ✓

„min $\leq$ max“: Sei $M \in \mathbb{R}^{V \times A}$ die Inzidenzmatrix von D und sei M' die Matrix, die man aus M erhält, wenn man die zu s und t gehörenden Zeilen streicht. $f \in \mathbb{R}^A$ ist ein s - t -Fluss genau dann, wenn $M'f = 0$. Sei $w^\top \in \mathbb{R}^A$ die zu t gehörende Zeile von M . Dann ist $\text{value}(f) = w^\top f$, also

$$\max_{\substack{f \text{ s-t-Fluss} \\ 0 \leq f \leq c}} \text{value}(f) = \max\{w^\top f: 0 \leq f \leq c, M'f = 0\}$$

Nach der LP-Dualität gilt

$$\begin{aligned} \max\{w^\top f: 0 \leq f \leq c, M'f = 0\} &= \min\{y^\top c: y^\top \geq 0, z \in \mathbb{Z}^{V \setminus \{s, t\}}, y^\top + z^\top M' \geq w^\top\} \\ &= \min\left\{y^\top c: \begin{pmatrix} y^\top & z^\top \end{pmatrix} \begin{pmatrix} I & I \\ 0 & M' \end{pmatrix} \geq \begin{pmatrix} 0 \\ w \end{pmatrix}\right\} \end{aligned}$$

Die Matrix $\begin{pmatrix} I & I \\ 0 & M' \end{pmatrix}$ ist vollständig-unimodular und $\begin{pmatrix} 0 \\ w \end{pmatrix}$ ist ganzzahlig, also wird das Minimum an ganzzahligen Vektoren y^*, z^* angenommen. Erweitere z^* zu $z^* \in \mathbb{Z}^V$ durch $z_t^* = -1, z_s^* = 0$. Dann gilt

$$(y^*)^\top + (z^*)^\top M \geq 0$$

Setze $U = \{v \in V: z_v^* \geq 0\}$. Dann ist $s \in U$, aber $t \notin U$. Also definiert U einen s - t -Schnitt $\delta^+(U)$.

Behauptung:

$$\delta^+(U) = \sum_{a \in \delta^+(U)} c(a) \leq (y^*)^\top c \quad (= \max \text{value}(f))$$

Beweis:

Es genügt zu zeigen: Falls $a \in \delta^+(U)$, dann $y_a^* \geq 1$. Da für $a = (u, v) \in \delta^+(U)$ immer $z_u^* \geq 0$ und $z_v^* \leq -1$ gilt, haben wir wegen $(y^*)^\top + (z^*)^\top M \geq 0$:

$$y_a^* + z_v^* - z_u^* \geq 0 \quad \text{also} \quad y_a^* \geq \underbrace{z_u^*}_{\geq 0} - \underbrace{z_v^*}_{\leq -1} \geq 1$$

□

LP-Dualität

$$\begin{aligned} \max \{ & dx + ey + fz : x \geq 0, z \leq 0, \\ & Ax + By + Cz \leq a \\ & Dx + Ey + Fz = b \\ & Gx + Hy + Kz \geq c \} &= \min \{ & ua + vb + wc : u \geq 0, w \leq 0, \\ & uA + vD + wG \geq d \\ & uB + vE + wH = e \\ & uC + vF + wK \leq f \} \end{aligned}$$

mit $d = w^\top$, $x = f$, $A = I$, $a = c$, $D = M'$, $\text{Rest} = 0$.

8. Ganzzahlige Optimierung, Schnittebenen und vollständig duale ganzzahlige Systeme

27.06.2014
21. Vorlesung

8.1. LP und ILP	77
8.2. Chvátal-Gomory-Abschluss	79
8.3. Total duale ganzzahlige Systeme und Hilbertbasen	81

8.1. LP und ILP

ILP: Gegeben: $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$

Gesucht:

$$\max\{c^T x : Ax \leq b, x \in \mathbb{Z}^n\} \quad (\text{ILP})$$

(ILP) ist im Allgemeinen ein schwieriges Problem.

Komplexitätstheorie: (ILP) ist $\mathcal{NP}$ -vollständig.

Zugehöriges LP (LP Relaxierung):

$$\max\{c^T x : Ax \leq b, x \in \mathbb{R}^n\} \quad (\text{LP})$$

(LP) kann effizient gelöst werden. In der Praxis mit dem Simplexalgorithmus ($\rightsquigarrow$ Kapitel 11), in der Theorie mit der Ellipsoid-Methode ($\rightsquigarrow$ Kapitel 12).

Klar: $(\text{LP}) \geq (\text{ILP})$

Es gilt sogar $(\text{LP}) = (\text{ILP})$, falls A vollständig-unimodular, $b \in \mathbb{Z}^m$.

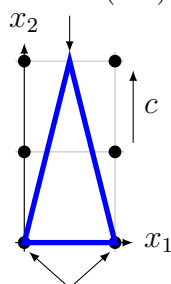
Idee: Verwende (LP) um (ILP) zu lösen.

Problem: Es kann passieren, dass $(\text{LP}) \gg (\text{ILP})$ ist.

Beispiel 8.1.1:

$$\max\left\{x_2 : x_1, x_2 \in \mathbb{Z}, x_2 \geq 0, -x_1 + \frac{1}{2r}x_2 \leq 0, x_1 + \frac{1}{2r}x_2 \leq 1\right\}$$

Optimum für (LP) = 2



Dann ist $r = (\text{LP})$, aber $0 = (\text{ILP})$

Optimum für (ILP) = 0

Definition 8.1.1 (ganzzahlige Hülle):

Sei $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ ein rationaler Polyeder (d.h. $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$). Die **ganzzahlige Hülle** von P ist

$$P_I = \text{conv}(P \cap \mathbb{Z}^n)$$

Falls P ein Polytop ist, dann ist $|P \cap \mathbb{Z}^n|$ endlich. Also ist P_I ein Polytop.

Satz 8.1.1:

$A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$ definieren ein rationales Polyeder $P = \{x \in \mathbb{R}^n : Ax \leq b\}$. Dann ist P_I ebenfalls ein (rationales) Polyeder.

Das ist gut, weil

$$\max\{c^\top x : Ax \leq b, x \in \mathbb{Z}^n\} = \max\{c^\top x : x \in P_I\}$$

wobei P_I rationales Polyeder $P_I = \{x \in \mathbb{R}^n : A'x \leq b'\}$

Beweis:

Klar, falls P beschränkt ist.

Angenommen: P ist unbeschränkt. Nach Theorem 6.5.2 (allgemeine Form) ist $P = Q + C$, wobei Q ein Polytop ist und C ein endlich erzeugter Kegel, d.h. $C = \text{cone}\{y_1, \dots, y_s\}$, wobei $y_1, \dots, y_s \in \mathbb{Q}^n$.

Ohne Einschränkung $y_1, \dots, y_s \in \mathbb{Z}^n$ (durch Skalierung mit kgV). Definiere

$$B = \left\{ \sum_{i=1}^s \lambda_i y_i : \lambda_i \in [0, 1] \right\}. \text{ Dann ist } B \text{ ein Polytop. } B = \text{conv} \left\{ \sum_{i \in I} y_i : I \subseteq \{1, \dots, s\} \right\}.$$

Behauptung:

$$P_I = \underbrace{(Q + B)_I}_{\text{Polytop}} + \underbrace{C}_{\substack{\text{endlich} \\ \text{erzeugter Kegel}}}, \text{ d.h. } P_I \text{ ist Polyeder nach Theorem 6.5.1.}$$

Beweis:

„ $\subseteq$ “: Sei $p \in P \cap \mathbb{Z}^n$. Dann $p = q + c$, mit $q \in Q$, $c \in C$. Dann $c = b + c'$ mit $b \in B$, $c' \in C \cap \mathbb{Z}^n$.

$$c = \sum_{i=1}^s \mu_i y_i = \underbrace{\sum_{i=1}^s (\mu_i - \lfloor \mu_i \rfloor) y_i}_{=b \in B} + \underbrace{\sum_{i=1}^s \lfloor \mu_i \rfloor y_i}_{=c' \in C \cap \mathbb{Z}^n}$$

Also $p = q + c = q + b + c' \in (Q + B)_I + C$, weil $q + b = p - c' \in \mathbb{Z}^n$.

„ $\supseteq$ “: $(Q + B)_I + C \subseteq P_I + C = P_I + C_I = (P + C)_I \underbrace{=}_{C=C+C} P_I$.

□

Korollar 8.1.1.a:

$$\max\{c^\top x : Ax \leq b, x \in \mathbb{Z}^n\} = \max\{c^\top x : x \in P_I\}$$

falls $\max < +\infty$.

Beweis:

„ $\leq$ “: ✓

„ $\geq$ “: Sei z eine optimale Lösung von $\max\{c^T x : x \in P_I\}$, mit $\alpha = c^T z$. Betrachte die Stütz-Hyperebene $H_\alpha = \{x \in \mathbb{R}^n : c^T x = \alpha\}$ und $F = H_\alpha \cap P_I$. Dann $z \in F$.

Behauptung:

F enthält ganzzahlige Punkte.

Beweis:

Angenommen es sein nicht so. Dann gilt für alle $y \in P \cap \mathbb{Z}^n : c^T y < \alpha$. Nun ist $z \in \text{conv}(P \cap \mathbb{Z}^n)$, d.h. $\exists x_1, \dots, x_t \in P \cap \mathbb{Z}^n$ mit

$$z = \sum_{i=1}^t \lambda_i x_i, \text{ mit } \lambda_i \geq 0, \sum_{i=1}^t \lambda_i = 1$$

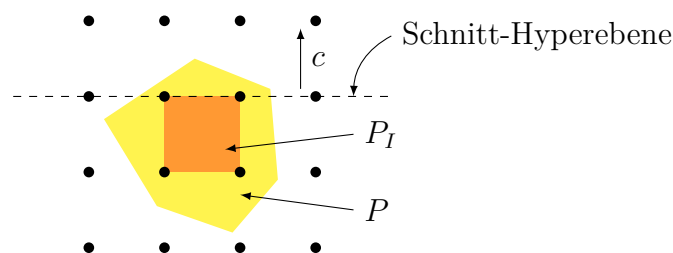
$$\text{Dann } \alpha = c^T z = c^T \left(\sum_{i=1}^t \lambda_i x_i \right) = \sum_i \lambda_i \underbrace{c^T x_i}_{< \alpha} < \alpha \sum_{i=1}^t \lambda_i = \alpha. \quad \nexists$$

□

Moral: Jedes (ILP) kann als (LP) geschrieben werden.

Problem: Eine Ungleichungsbeschreibung von P_I ist im Allgemeinen nicht einfach (bzw. effizient) zu bestimmen.

Idee: Füge (systematisch) Schnitt-Hyperebenen hinzu, um P Schritt für Schritt zu P_I zu verkleinern.



8.2. Chvátal-Gomory-Abschluss

02.07.2014
22. Vorlesung

Konstruiere Polyeder P' von P mit

$$P \supseteq P' \supseteq P_I$$

und iteriere $P \supseteq P' \supseteq P'' \supseteq \dots \supseteq P^{(t)} = P_I$.

Definition 8.2.1 (Chvátal-Gomory-Abschluss):

Sei P ein rationales Polyeder. Der **Chvátal-Gomory-Abschluss** (CG-Abschluss) von P ist

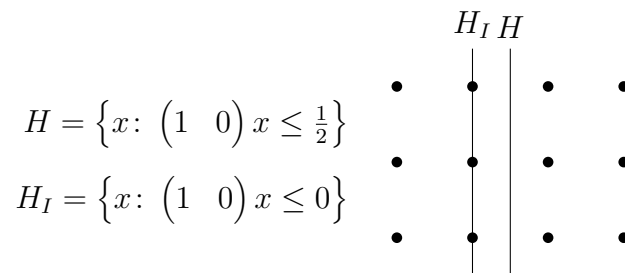
$$P' = \bigcap_{\substack{H \supseteq P \\ H \text{ rationaler Halbraum}}} H_I$$

wobei $H_I = \text{conv}\{H \cap \mathbb{Z}^n\}$. Weil aus $P \subseteq H$ auch $P_I \subseteq H_I$ folgt, ist $P \supseteq P' \supseteq P_I$.

Berechnung von H_I

Sei $H = \{x \in \mathbb{R}^n : a^\top x \leq b\}$ mit $a \in \mathbb{Q}^n \setminus \{0\}$, $b \in \mathbb{Q}$ ein rationaler Halbraum. Ohne Einschränkung sei $a \in \mathbb{Z}^n \setminus \{0\}$, $\text{ggT}(a_1, \dots, a_n) = 1$. Dann $H_I = \{x \in \mathbb{R}^n : a^\top x \leq \lfloor b \rfloor\}$.

Beispiel 8.2.1:



In Worten:

- Schiebe H parallel entgegen der Richtung a so lange, bis der Rand ganzzahlige Punkte enthält
- Um P' zu erhalten, mache dies für jeden rationalen Halbraum, der P enthält

Fragen:

- Ist P' wieder ein Polyeder? (In Definition 8.2.1 werden unendlich viele Halbräume verwendet)
- Falls ja, wie berechnet man P' ?
- Konvergiert $P', P'', P''', \dots$ nach P_I in endlich vielen Schritten?

Theorem 8.2.1:

Sei P ein rationales Polyeder.

- Dann ist P' ebenfalls ein rationales Polyeder
- $\exists t \in \mathbb{N} : P^{(t)} = P_I$.

Das kleinste t mit $P^{(t)} = P_I$ heißt **Chvátal-Rang** von P .

Beweis:

In der Vorlesung nur für Teil a), für Teil b) [siehe 3, Kapitel 23].

□

Definition 8.2.2 (vollständig dual ganzzahliges System):

Sei $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$. Das System $Ax \leq b$ heißt **vollständig dual ganzzahlig** (TDI = totally dual integral), falls das Minimum in der LP-Dualität

$$\max\{c^\top x : Ax \leq b\} = \min\{y^\top b : y \geq 0, y^\top A = c^\top\}$$

von einem ganzzahligen Vektor y für jedes $c \in \mathbb{Z}^n$ angenommen wird, falls es endlich ist.

Achtung:

Diese Eigenschaft bezieht sich auf das System $Ax \leq b$ und nicht auf das Polyeder $\{x: Ax \leq b\}$.

Beispiel:

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \text{ ist TDI,}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ aber nicht.}$$

Satz 8.2.1:

Sei $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Q}^m$. Angenommen $Ax \leq b$ ist TDI. Sei $P = \{x: Ax \leq b\}$, dann gilt $P' = \{x: Ax \leq \lfloor b \rfloor\}$. Insbesondere ist P' ein Polyeder (falls wir P mit TDI-System beschreiben können).

Beweis:

Angenommen $P \neq \emptyset$.

„ $\subseteq$ “: Klar, nach Definition von H_I .

„ $\supseteq$ “: Sei $H = \{x \in \mathbb{R}^n: c^\top x \leq \delta\}$ ein rationaler Halbraum mit $P \subseteq H$ und $H_I = \{x: c^\top x \leq \lfloor \delta \rfloor\}$, wobei $c \in \mathbb{Z}^n \setminus \{0\}$.

Behauptung:

$$H_I \supseteq \{x: Ax \leq \lfloor b \rfloor\}$$

Beweis:

Es gilt $\delta \geq \max\{c^\top x: Ax \leq b\} = \min\{y^\top b: y^\top A = c^\top, y \geq 0\}$. Da $Ax \leq b$ TDI wird das Minimum bei $y_0 \in \mathbb{Z}^m$ angenommen. Angenommen für $x \in \mathbb{R}^n$ gilt $Ax \leq \lfloor b \rfloor$. Dann $c^\top x = y_0^\top Ax \leq y_0^\top \lfloor b \rfloor \leq \underbrace{y_0^\top b}_{\text{da } y_0 \in \mathbb{Z}^m} \leq \lfloor \delta \rfloor$, also $x \in H_I$. □

Noch zu klären: Wie beschreibt man ein rationales Polyeder mit einem TDI-System?

8.3. Total duale ganzzahlige Systeme und Hilbertbasen

Definition 8.3.1 (Hilbertbasis):

Seien $a_1, \dots, a_t \in \mathbb{Q}^n$ gegeben.

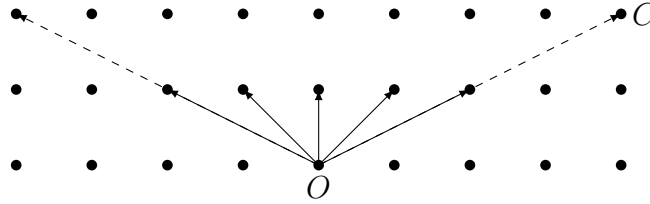
a) Diese Vektoren bilden eine **Hilbertbasis**, wenn

$$\forall b \in \text{cone}\{a_1, \dots, a_t\} \cap \mathbb{Z}^n \exists \lambda_1, \dots, \lambda_t \in \mathbb{Z}: b = \sum_{i=1}^t \lambda_i a_i$$

b) Diese Vektoren bilden eine **ganzzahlige Hilbertbasis**, falls $a_1, \dots, a_t \in \mathbb{Z}^n$ und $a_1, \dots, a_t$ sind Hilbertbasis.

Beispiel 8.3.1:

$C = \text{cone} \left\{ \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix} \right\}$. Dann ist $\begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}$ eine Hilbertbasis.



Satz 8.3.1:

Es sei $C = \{x \in \mathbb{R}^n : Ax < 0\}$ ein rationaler Kegel, der endlich erzeugt ist. Dann gibt es eine ganzzahlige Hilbertbasis $a_1, \dots, a_t \in \mathbb{Z}^n$ mit $C = \text{cone}\{a_1, \dots, a_t\}$.

Beweis:

Nach Theorem 6.5.1 gibt es $b_1, \dots, b_s \in \mathbb{Z}^n$ mit $C = \text{cone}\{b_1, \dots, b_s\}$. Betrachte das Polytop $B = \left\{ \sum_{i=1}^s \alpha_i b_i : \alpha_i \in [0, 1] \right\}$. Dann ist $\{a_1, \dots, a_t\} = B \cap \mathbb{Z}^n$ eine ganzzahlige Hilbertbasis.

□

04.07.2014
23. Vorlesung

Definition 8.3.2 (Polyederseite):

Sei $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ ein Polyeder und $H = \{x \in \mathbb{R}^n : c^T x = \delta\}$ eine Stützhyperebene, d. h. $P \subseteq H^-$, $P \cap H \neq \emptyset$. Dann heißt $F = P \cap H$ eine **Seite** von P .

Satz 8.3.2:

$$F \subseteq P \text{ eine Seite} \Leftrightarrow F \neq \emptyset \text{ und } F = \{x \in P : A'x = b'\}$$

wobei $A'x \leq b'$ ein Teilsystem von $Ax \leq b$ ist.

Beweis:

Nicht schwer. „ $\Rightarrow$ “ benutzt LP-Dualität.

Satz 8.3.3:

Sei $Ax \leq b$ ein rationales System, was Polyeder $P = \{x : Ax \leq b\}$ definiert. Für jede Seite $F \subseteq P$ gelte: Die Zeilen von A , die aktiv bei F sind, sind eine Hilbertbasis, dann ist $Ax \leq b$ TDI.

Beweis:

Sei $c \in \mathbb{Z}^n$ mit $\max\{c^T x : Ax \leq b\} = \min\{y^T b : y^T A = c^T, y \geq 0\}$, wobei das Minimum endlich ist. Es sei F eine inklusionsminimale Seite von P , an der das Maximum angenommen wird. Ohne Einschränkung seien die ersten t Zeilen von A $a_1^T, \dots, a_t^T$ bei F aktiv. Dann ist $c \in \text{cone}\{a_1, \dots, a_t\}$, weil:

Seien x_0, y_0 optimal. Dann sind die letzten $m - t$ Komponenten von y_0 gleich Null, weil wegen der LP-Dualität gilt

$$c^T x_0 = y_0^T A x_0 = \tilde{y}_0^T \begin{pmatrix} a_1^T \\ \vdots \\ a_t^T \end{pmatrix} x_0$$

wobei $\tilde{y}_0$ die ersten t Komponenten von y_0 enthält.

Da $a_1, \dots, a_t$ nach Voraussetzung eine Hilbertbasis ist, $\exists \lambda_1, \dots, \lambda_t \in \mathbb{Z}: c = \sum_{i=1}^t \lambda_i a_i$. Sei $y = (\lambda_1 \ \dots \ \lambda_t \ 0 \ \dots \ 0)^\top \in \mathbb{Z}^m$. Dann $y^\top b = y^\top A x_0 = c^\top x_0$, d.h. $y \in \mathbb{Z}^m$ ist ein ganzzahliger Vektor, der das Minimum annimmt. Damit ist $Ax \leq b$ TDI. $\square$

Satz 8.3.4:

Sei P ein rationales Polyeder. Dann gibt es $A \in \mathbb{Z}^{m \times n}$ und $b \in \mathbb{Q}^m$, so dass $Ax \leq b$ TDI ist und $P = \{x \in \mathbb{R}^n: Ax \leq b\}$ gilt.

Beweis:

Sei F eine minimale Seite von P . Definiere den rationalen, endlich erzeugten Kegel

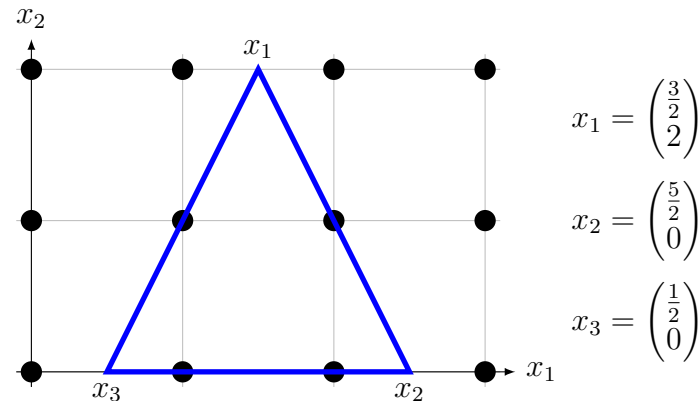
$$C_F = \left\{ c \in \mathbb{R}^n: F = \left\{ x \in P: c^\top x = \max\{c^\top y: y \in P\} \right\} \right\}$$

Dies ist der Kegel aller Zielfunktionen für die das Maximum an der Seite F angenommen wird. Da C_F rational ist, gibt es eine ganzzahlige Hilbertbasis $a_1, \dots, a_s$ mit $C_F = \text{cone}\{a_1, \dots, a_s\}$. Definiere β_i durch $\beta_i = a_i^\top x_0$, wobei $x_0 \in F$. Betrachte das System $(\sum_F): a_i^\top x \leq \beta_i, i = 1, \dots, s$.

Es sei $Ax \leq b$ das System, das man erhält, wenn man alle Systeme $(\sum_F)$, F minimale Seite von P , hintereinander schreibt. Dies ist nach Satz 8.3.3 TDI und $P = \{x: Ax \leq b\}$. $\square$

Beispiel 8.3.2:

$$P = \{x \in \mathbb{R}^2: x_2 \geq 0, 2x_1 - x_2 \geq 1, 2x_1 + x_2 \leq 5\}$$



$\underbrace{\{x_1\}}_{=F_1}, \underbrace{\{x_2\}}_{=F_2}, \underbrace{\{x_3\}}_{=F_3}$ sind minimale Seiten von P

$$\begin{aligned} \text{Dann: } C_{F_1} &= \text{cone} \left\{ \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix} \right\} \\ C_{F_2} &= \text{cone} \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right\} \\ C_{F_3} &= \text{cone} \left\{ \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right\} \end{aligned}$$

Die ganzzahlige Hilbertbasis von C_{F_1} (siehe Beispiel 8.3.1):

$$\begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

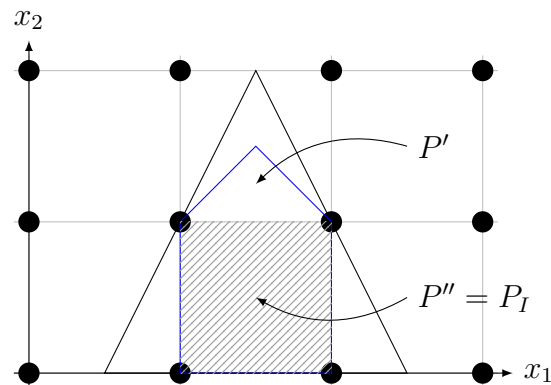
Die ganzzahlige Hilbertbasis von C_{F_2} :



Die ganzzahlige Hilbertbasis von C_{F_3} (C_{F_2} gespiegelt):

$$\begin{pmatrix} -2 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$P = \left\{ x \in \mathbb{R}^2 : \underbrace{\begin{pmatrix} -2 & 1 \\ -1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 2 & 1 \\ 2 & 1 \\ 1 & 0 \\ 0 & -1 \\ -2 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} x \leq \begin{pmatrix} -1 \\ \frac{1}{2} \\ 2 \\ \frac{7}{2} \\ 5 \\ 5 \\ \frac{5}{2} \\ 0 \\ -1 \\ -\frac{1}{2} \\ 0 \end{pmatrix}}_{\text{TDI-System}} \right\} \quad P' = \left\{ x \in \mathbb{R}^2 : \begin{pmatrix} -2 & 1 \\ -1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 2 & 1 \\ 2 & 1 \\ 1 & 0 \\ 0 & -1 \\ -2 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} x \leq \begin{pmatrix} -1 \\ 0 \\ 2 \\ 3 \\ 5 \\ 5 \\ 2 \\ 0 \\ -1 \\ -1 \\ 0 \end{pmatrix} \right\}$$



9. Spieltheorie

9.1. Bimatrixspiele	85
9.2. Matrixspiele	89

Ein Teilgebiet des Operations Research ist die (mathematische) Spieltheorie. Sie hat das Ziel, mathematische Methoden bereitzustellen, mit denen man das Entscheidungsverhalten von mehreren „Spielern“ in Konfliktsituationen analysieren kann. Sie wurde von JOHN VON NEUMANN und OSKAR MORGENSTERN durch das Buch „Theory of games and economic behaviour“ [siehe 4] wissenschaftlich begründet. Auch hier ist die Polyedertheorie und die Theorie der linearen Programmierung ein wichtiges Hilfsmittel, was wir am Beispiel von Matrixspielen sehen werden.

9.1. Bimatrixspiele

Beispiel 9.1.1 (Prisoner's Dilemma):

Es modelliert die Situation zweier Gefangener, die eines gemeinsamen Verbrechens beschuldigt werden. Jeder Gefangene hat unabhängig voneinander die Möglichkeit zu schweigen oder zu gestehen. Das individuelle Strafmaß richtet sich danach, wie die beiden Gefangenen zusammengenommen ausgesagt haben. Schweigen beide, erhalten sie niedrige Strafen, gestehen beide, erhalten sie hohe Strafen. Gesteht jedoch nur einer der Gefangenen, so geht dieser als Kronzeuge straffrei aus, während der andere die Höchststrafe bekommt.

Die Ergebnisse der Möglichen Verhaltensweisen lassen sich in einem Matrix-Paar darstellen:

$$\begin{array}{cc} & \begin{array}{cc} 2 \text{ sagt aus} & 2 \text{ schweigt} \end{array} \\ \begin{array}{c} 1 \text{ sagt aus} \\ 1 \text{ schweigt} \end{array} & \begin{pmatrix} 5 & 0 \\ 6 & 1 \end{pmatrix} \end{array}$$

Entsprechend sieht die Matrix für Gefangenen 2 wie folgt aus:

$$\begin{array}{cc} & \begin{array}{cc} 2 \text{ sagt aus} & 2 \text{ schweigt} \end{array} \\ \begin{array}{c} 1 \text{ sagt aus} \\ 1 \text{ schweigt} \end{array} & \begin{pmatrix} 5 & 6 \\ 0 & 1 \end{pmatrix} \end{array}$$

Insgesamt schreiben wir das Spiel als

$$(A, B) = \begin{pmatrix} 5, 5 & 0, 6 \\ 6, 0 & 1, 1 \end{pmatrix}$$

Der Gefangene 1 ist der *Zeilenspieler* Z ; der Gefangene 2 ist der *Spaltenspieler* S . Beide wollen einen möglichst kleinen Eintrag erreichen, unter der Bedingung der Wahl des anderen.

Allgemein: Sei $(A, B) = (A_{i,j}, B_{i,j})_{i=1,\dots,m, j=1,\dots,n} \in \mathbb{R}^{m \times n}$ die Auszahlungsmatrix für Zeilen-
spieler Z und Spaltenspieler S .

- Z versucht den maximalen Wert $A_{i,j}$ zu bekommen unter der Bedingung das S die Spalte festlegt
- S versucht den maximalen Wert $B_{i,j}$ zu bekommen unter der Bedingung das Z die Zeile festlegt

Beide Spieler werden versuchen die Wahl des anderen zu erraten. Dies modellieren wir wie folgt: Wir stellen uns vor, dass das Spiel sehr häufig gespielt wird, und dass die Spieler ihre Wahl *nicht* vom Ausgang vorheriger Spiele abhängig machen. Sie wählen also jede Zeile/Spalte mit einer gewissen Wahrscheinlichkeit.

Definition 9.1.1 (gemischte Strategie):

Sei u_i die Wahrscheinlichkeit, dass Z die Zeile i wählt und sei v_j die Wahrscheinlichkeit, dass S die Spalte j wählt. Dann sind

$$u = (u_1 \quad \cdots \quad u_m), \quad v = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} \quad \text{mit } u, v \geq 0, \quad u e = 1, \quad e^T v = 1$$

gemischte Strategien für Z respektive S .

Hierbei steht e für den Vektor mit Einsen der passenden Länge.

Für Strategie v_0 von S ist u_0 eine **beste Antwort**, falls $u_0 A v_0 = \max\{u A v_0 : u \geq 0, u e = 1\}$. Entsprechend ist v_0 eine beste Antwort auf u_0 , falls $u_0 B v_0 = \max\{u_0 B v : v \geq 0, v^T e = 1\}$.

Beispiel (Fortsetzung von Beispiel 9.1.1):

Da wir im allgemeinen Fall maximieren, betrachten wir die Bimatrix

$$\begin{pmatrix} -5, -5 & 0, -6 \\ -6, 0 & -1, -1 \end{pmatrix}$$

Sei $v_0 = \begin{pmatrix} q \\ 1-q \end{pmatrix}$ fest mit $0 \leq q \leq 1$. Wollen beste Antwort auf v_0 :

$$\begin{aligned} \max \left\{ \begin{pmatrix} p \\ 1-p \end{pmatrix}^T \begin{pmatrix} -5 & 0 \\ -6 & -1 \end{pmatrix} \begin{pmatrix} q \\ 1-q \end{pmatrix} : 0 \leq p \leq 1 \right\} &= \max \left\{ \begin{pmatrix} p \\ 1-p \end{pmatrix}^T \begin{pmatrix} -5q \\ -5q-1 \end{pmatrix} : 0 \leq p \leq 1 \right\} \\ &= \max\{p - 5q - 1 : 0 \leq p \leq 1\} \end{aligned}$$

Also ist $p = 1$ und somit $\begin{pmatrix} 1 \\ 0 \end{pmatrix}^T$ die eindeutige beste Antwort, egal wie q gewählt war.

Genauso sieht man, dass $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ die beste Antwort für S ist.

Achtung:

Nicht immer ist die beste Antwort unabhängig von der Wahl des Gegners.



Definition 9.1.2 (Gleichgewicht):

Ein **Gleichgewicht** (Nash equilibrium) ist ein Paar u_0, v_0 von Strategien, so dass u_0 beste Antwort auf v_0 ist, und v_0 beste Antwort auf u_0 .

Der **Wert** eines Gleichgewichts ist $(u_0 A v_0, u_0 B v_0)$.

Beispiel 9.1.2:

Gegeben sei das Spiel

$$\begin{pmatrix} 2, -1 & -2, -2 \\ 1, \frac{3}{2} & -1, 2 \end{pmatrix}$$

Beste Antworten:

- Für $0 \leq q \leq 1$ fest:

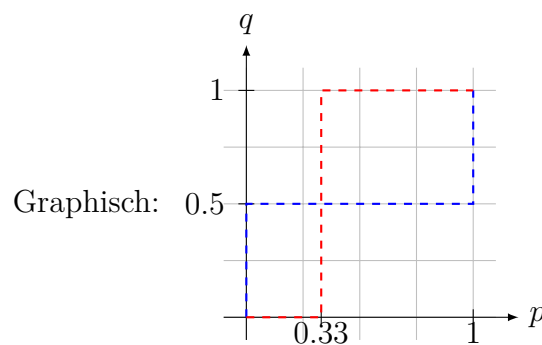
$$\begin{pmatrix} p \\ 1-p \end{pmatrix}^\top \begin{pmatrix} 2 & -2 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} q \\ 1-q \end{pmatrix} = \begin{pmatrix} p \\ 1-p \end{pmatrix}^\top \begin{pmatrix} 4q-2 \\ 2q-1 \end{pmatrix} \\ = p(2q-1) + 2q-1$$

$\Rightarrow$ für $q < \frac{1}{2}$ wähle $p = 0$, für $q > \frac{1}{2}$ wähle $p = 1$

- Für $0 \leq p \leq 1$ fest:

$$\begin{pmatrix} p \\ 1-p \end{pmatrix}^\top \begin{pmatrix} -1 & -2 \\ \frac{3}{2} & 2 \end{pmatrix} \begin{pmatrix} q \\ 1-q \end{pmatrix} = \begin{pmatrix} -\frac{5}{2}p + \frac{3}{2} \\ -4p + 2 \end{pmatrix}^\top \begin{pmatrix} -5q \\ -5q-1 \end{pmatrix} \\ = q\left(\frac{3}{2}p - \frac{1}{2}\right) + 2 - 4p$$

$\Rightarrow$ für $p < \frac{1}{3}$ wähle $q = 0$, für $p > \frac{1}{3}$ wähle $q = 1$



Es gibt 3 Schnittpunkte und jeder ist ein Gleichgewicht.

- Gleichgewicht $\begin{pmatrix} 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ hat Wert $(-1, 2)$
- Gleichgewicht $\begin{pmatrix} \frac{1}{3} & \frac{2}{3} \end{pmatrix}, \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$ hat Wert $\frac{1}{6}(2, -1) + \frac{1}{6}(-2, -2) + \frac{2}{6}(1, \frac{3}{2}) + \frac{2}{6}(-1, 2) = (0, \frac{4}{6}) = (0, \frac{2}{3})$
- Gleichgewicht $\begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ hat Wert $(2, -1)$

Bemerkung 9.1.1:

Da die Werte der Gleichgewichte sehr unterschiedlich sind, ist nicht klar, dass sie im Spiel tatsächlich erreicht werden.

Z wird versuchen, den Wert $(2, -1)$ zu erreichen. S wird versuchen, den Wert $(-1, 2)$ zu erreichen. Wenn beide entsprechend spielen, landen sie bei $(-2, -2)$.

Definition 9.1.3 (korreliertes Gleichgewicht):

Sei $P \in \mathbb{R}^{m \times n}$ mit $P_{i,j} \geq 0$ und $\sum_{i=1}^m \sum_{j=1}^n P_{i,j} = 1$. Dann heißt P **korreliertes Gleichgewicht** (oder **Korrelationsmatrix**) für (A, B) falls gilt:

Wenn mit Wahrscheinlichkeit $P_{i,j}$ Z geraten wird, i zu wählen und S geraten wird, j zu wählen (und ihnen nicht gesagt wird, was dem Anderen geraten wurde), dann ist es für Beide am besten, dem Rat zu folgen.

Der Wert des korrelierten Gleichgewichts ist $\sum_{i,j} P_{i,j}(A_{i,j}, B_{i,j})$.

Beispiel (Fortsetzung von Beispiel 9.1.2):

Für das Spiel

$$\begin{pmatrix} 2, -1 & -2, -2 \\ 1, \frac{3}{2} & -1, 2 \end{pmatrix}$$

ist ein korreliertes Gleichgewicht gegeben durch

$$P = \begin{pmatrix} 0.2 & 0 \\ 0.3 & 0.5 \end{pmatrix}$$

Denn:

- Raten wir Z Zeile 1 zu wählen, dann weiß Z , dass wir mit Wahrscheinlichkeit 1 S raten, Spalte 1 zu wählen, also Strategie $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ zu spielen.
Beste Antwort: $(1, 0)$. Also wählt Z am besten Zeile 1.
- Raten wir Z Zeile 2 zu wählen, dann weiß Z , dass wir mit Wahrscheinlichkeit $\frac{0.3}{0.3+0.5} = \frac{3}{8}$ S raten, Spalte 1 zu wählen, und mit Wahrscheinlichkeit $\frac{5}{8}$ Spalte 2. Da $\frac{3}{8} < \frac{1}{2}$, ist die beste Antwort auf $\begin{pmatrix} \frac{3}{8} \\ \frac{5}{8} \end{pmatrix}$ also $(0, 1)$.
- Raten wir S zu Spalte 1, dann beraten wir Z nach Strategie $(\frac{2}{5}, \frac{3}{5})$. Da $\frac{2}{5} > \frac{1}{3}$, ist die beste Antwort $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$.
- Raten wir S zu Spalte 2, dann beraten wir Z nach Strategie $(0, 1)$. Die beste Antwort ist $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

Der Wert ist $\frac{2}{10}(2, -1) + \frac{3}{10}(1, \frac{3}{2}) + \frac{5}{10}(-1, 2) = (\frac{1}{5}, \frac{5}{4})$.

9.2. Matrixspiele

Das bekannte Spiel *Schere-Stein-Papier* kann man mit Hilfe einer Matrix beschreiben:

	Schere	Stein	Papier
Schere	0	+1	-1
Stein	-1	0	+1
Papier	+1	-1	0

Dabei liegt folgende Interpretation zugrunde: Zwei Spieler, Z und S wählen gleichzeitig (schnick-schnack-schnuck) eine der drei Optionen Schere, Stein oder Papier aus. Falls z. B. Z Schere wählt und S Papier, dann sagt Eintrag $(1, 3)$ der Matrix, dass Z minus einen Eurocent an S bezahlen muss, d. h. Z bekommt einen Eurocent von S .

Merkregeln: Schere schneidet Papier, Stein macht Schere stumpf, Papier wickelt Stein ein. Mit gesundem Menschenverstand kommt man zur Überzeugung, dass es eine gute, wenn nicht sogar eine optimale Strategie ist, jede der drei Alternativen mit einer Wahrscheinlichkeit von jeweils $\frac{1}{3}$ zu wählen. Die Dualitätstheorie der linearen Programme ermöglicht es, dies zu beweisen, und auch dieses Spiel in einem viel allgemeineren Rahmen zu sehen.

Allgemein betrachten wir ein Spiel mit zwei Spielern. Spieler Z , der Zeilenspieler, hat die Möglichkeiten aus den Optionen $\{1, \dots, m\}$ zu wählen. Spieler S , der Spaltenspieler, kann aus $\{1, \dots, n\}$ wählen. Die Auszahlungsmatrix

$$A = (A_{i,j})_{\substack{i=1,\dots,m \\ j=1,\dots,n}} \in \mathbb{R}^{m \times n}$$

bestimmt den Spielausgang. Falls Z Option i wählt und S Option j , dann bezahlt Z an S den Betrag $A_{i,j}$. Wie schon gesehen, kann dieser Betrag auch negativ sein. Das Ziel für Z ist also, dass die Bezahlung minimiert wird, während S die Bezahlung maximieren möchte. Wir gehen davon aus, dass Z und S das Spiel sehr oft gegeneinander spielen möchten bzw. müssen. Deswegen ist es sinnvoll, dass die Spieler ihre Strategien in Abhängigkeit einer Wahrscheinlichkeitsverteilung wählen. So ist u_i die Wahrscheinlichkeit, dass Z die Option i wählt, und v_j die von S für Option j . Allgemeine Vektoren

$$u = (u_1 \ \cdots \ u_m), \ v = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} \text{ mit } u, v \geq 0, \ u e = 1, \ e^\top v = 1$$

wobei $e = (1, \dots, 1)^\top$ der Einsvektor ist, heißen *gemischte Strategien*. Die Standardbasisvektoren $e_i^\top$, mit $i = 1, \dots, m$, bzw. e_j , mit $j = 1, \dots, n$, heißen *reine Strategien*. Der Erwartungswert der Bezahlung von Z an S ist somit

$$uAv = \sum_{i=1}^m \sum_{j=1}^n u_i A_{i,j} v_j$$

Nun bestimmen wir die optimale Strategie u des Zeilenspielers Z unter der (pessimistischen) Annahme, dass seine Strategie dem Spaltenspieler S bekannt ist. S wählt v so, dass uAv maximiert wird:

$$\sup\{uAv : v \geq 0, \ e^\top v = 1\} = \max_{j=1,\dots,n} (uA)_j$$

Nun ist das beste, was Z machen kann, dieses Maximum zu minimieren:

$$\begin{aligned} \min \quad & \max_{j=1,\dots,n} (uA)_j \\ & u \geq 0 \\ & ue = 1 \end{aligned}$$

Das kann man durch Hinzufügen einer Variablen λ als ein lineares Programm modellieren:

$$\begin{aligned} \min \quad & \lambda \\ & u \geq 0 \\ & ue = 1 \\ & uA \leq \lambda e \end{aligned}$$

Das dazu duale lineare Programm ist nach Aufgabe 6.3 (vollziehen Sie diesen wichtigen Schritt in aller Ruhe nach!)

$$\begin{aligned} \max \quad & \mu \\ & z \leq 0 \\ & -e^T z = 1 \\ & e\mu + Az \leq 0 \end{aligned}$$

bzw.

$$\begin{aligned} \max \quad & \mu \\ & v \geq 0 \\ & e^T v = 1 \\ & Av \geq \mu e \end{aligned}$$

Dies bedeutet aber (warum?), dass das duale lineare Programm die optimale Strategie v für den Spaltenspieler S bestimmt, wieder unter der pessimistischen Annahme, dass der Zeilenspieler die Strategie von S kennt. Nach dem Dualitätstheorem – die Mengen gültiger Lösungen sind bei beiden linearen Programmen nicht leer – stimmen nun beide Werte überein. Dieser Wert heißt auch der *Wert* des durch die Matrix A bestimmten Matrixspieles.

Hinweis:

Die Notizen zu „Matrixspielen“ der neueren Vorlesung von 2015 befindet sich im [Anhang](#).



Algorithmen der linearen Optimierung

10. Das Eliminationsverfahren von Fourier und Motzkin	92
11. Das Simplexverfahren	94
11.1. Der Algorithmus	95
12. Die Ellipsoidmethode	97
12.1. Grundlegendes zu Ellipsoiden	97
12.2. Separieren und optimieren	99
13. Die Innere-Punkte-Methode	103
13.1. Der zentrale Pfad	103
13.2. Der Algorithmus und seine Analyse	108

10. Das Eliminationsverfahren von Fourier und Motzkin

09.07.2014
24. Vorlesung

Bislang haben wir die Theorie der Systeme linearer Ungleichungen vor allem geometrisch betrachtet, ohne auf Algorithmen einzugehen. Auch haben wir mehrmals Parallelen zur linearen Algebra aufgezeigt. Der wohl wichtigste Algorithmus in der linearen Algebra ist das Gaußsche Eliminationsverfahren zur Lösung von Systemen linearer Gleichungen. Das Eliminationsverfahren von Fourier und Motzkin ist eine Variante des Gaußschen Eliminationsverfahren, mit dem man Systeme linearer Ungleichungen lösen kann.

Wir stellen uns die folgende algorithmische Aufgabe: Gegeben seien eine Matrix $A \in \mathbb{R}^{m \times n}$ und ein Vektor $b \in \mathbb{R}^m$. Es ist ein Vektor $x \in \mathbb{R}^n$ gesucht, der das System linearer Ungleichungen $Ax \leq b$ löst, bzw. der Algorithmus soll mit Sicherheit sagen, dass es einen solchen Vektor nicht gibt.

Fourier-Motzkin Elimination

Die Fourier-Motzkin Elimination geht diese Aufgabe wie folgt an: Durch Multiplikation der Zeilen der Matrix A und der entsprechenden rechten Seiten b_i mit einer positiven Konstante kann man annehmen, dass die Einträge der ersten Spalte von A entweder 1, -1 oder 0 sind. Nach Umnummerierung der Zeilen von A erhält man so

$$\begin{pmatrix} 1 & a_1'^T \\ \vdots & \vdots \\ 1 & a_r'^T \\ -1 & a_{r+1}'^T \\ \vdots & \vdots \\ -1 & a_s'^T \\ 0 & a_{s+1}'^T \\ \vdots & \vdots \\ 0 & a_m'^T \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \leq \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \quad (10.1)$$

wobei $a_i'^T \in \mathbb{R}^{1 \times (n-1)}$ die i -te Zeile von A ist, in der das erste Element gelöscht wurde. Nun sind der ersten $r + s$ Bedingungen äquivalent zu

$$\max_{r+1 \leq j \leq s} a_j'^T x' - b_j \leq x_1 \leq \min_{1 \leq i \leq r} b_i - a_i'^T x' \quad (10.2)$$

wobei $x' = (x_2, \dots, x_n)^T$. Also kann man die Variable x_1 eliminieren und das System (10.1) ist äquivalent zu

$$\begin{aligned} a_j'^T x' - b_j &\leq b_i - a_i'^T x', & i = 1, \dots, r, & j = r + 1, \dots, s \\ a_i'^T x' &\leq b_i, & i = s + 1, \dots, m \\ \text{bzw. zu } (a_j' + a_i')^T x' &\leq b_i + b_j, & i = 1, \dots, r, & j = r + 1, \dots, s \\ a_i'^T x' &\leq b_i, & i = s + 1, \dots, m \end{aligned}$$

Dieses System besteht aus $r(s - r) + m - (r + s)$ Ungleichungen und $n - 1$ Variablen. Geometrisch ist die Lösungsmenge eine Projektion des Polyeders $\{x \in \mathbb{R}^n : Ax \leq b\}$ entlang der x_1 -Achse. Jedes seiner Lösungen x' kann zu einer Lösung $\begin{pmatrix} x_1 \\ x' \end{pmatrix}$ von (10.1) erweitert werden, falls x_1 die Ungleichung (10.2) erfüllt.

Im nächsten Schritt kann man genauso x_2 eliminieren, etc. Wenn man so bei x_n angelangt ist, ist es offensichtlich, ob das System eine Lösung besitzt oder nicht, da es sich um ein System mit nur einer Variablen handelt. Falls es eine Lösung gibt, dann kann man sie zu einer Lösung des ursprünglichen Systems erweitern.

Weitere Anwendungen

Genauso wie das Gaußsche Eliminationsverfahren kann man die Fourier-Motzkin Elimination auch für andere algorithmische Fragestellungen benutzen. Wir schauen uns zwei an: die Lösung von linearen Optimierungsproblemen und ein alternativer, algorithmischer Beweis für das Lemma von Farkas.

Lineare Optimierungsprobleme

Man kann die Fourier-Motzkin Elimination verwenden, um eine optimale Lösung des linearen Optimierungsproblems

$$\max\{c^T x : Ax \leq b\}$$

zu finden. Dazu betrachtet man das System

$$\begin{aligned} Ax &\leq b \\ -c^T x + \lambda &\leq 0 \end{aligned}$$

wobei $\lambda \in \mathbb{R}$ eine zusätzliche Variable ist. Wir möchten also einen Vektor $\begin{pmatrix} x \\ \lambda \end{pmatrix}$ finden, der dieses System löst, und bei dem λ so groß wie möglich ist. Dazu eliminieren wir sukzessive die Variablen $x_1, x_2, \dots, x_n$ bis λ die einzige Variable ist, die übrigbleibt. Diese wird nun, wie gesagt, möglichst groß gewählt.

Lemma von Farkas

Angenommen das System $Ax \leq b$ hat keine Lösung. Nach dem Lemma von Farkas gibt es einen Vektor $y \in \mathbb{R}^m$, so dass $y \geq 0$, $y^T A = 0$ und $y^T b < 0$. Einen solchen Vektor y kann man mit Hilfe der Fourier-Motzkin Elimination per Induktion konstruieren:

Falls $Ax \leq b$ keine Lösung hat, dann hat auch das System $A'x' \leq b'$ keine Lösung, in dem die Variable x_1 eliminiert worden ist. Nach Induktion ist nun der Vektor $(0, \dots, 0, -1)$ eine nichtnegative Linearkombination der Zeilen der Matrix $\begin{pmatrix} A' & b' \end{pmatrix}$. Nach Konstruktion sind die Zeilen der Matrix $\begin{pmatrix} 0 & A' & b' \end{pmatrix}$ die Summe von Zeilen von $\begin{pmatrix} A & b \end{pmatrix}$. Also ist der Vektor $(0, \dots, 0, -1)$ eine nichtnegative Linearkombination der Zeilen von $\begin{pmatrix} A & b \end{pmatrix}$ und die Koeffizienten dieser Linearkombination ergeben den gesuchten Vektor y .

11. Das Simplexverfahren

Ziel: Löse $p^* = \max\{c^T x : x \in \mathbb{R}^n, Ax \leq b\}$ bei gegebenen $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$.

Zunächst:

Annahme: $P = \{x : Ax \leq b\}$ ist ein Polytop. Wir wissen das Maximum von P wird in einer Ecke angenommen. Sei x_0 eine uns bekannte Ecke von P .

- Wähle quadratisches Teilsystem $A_0 x \leq b_0$ von $Ax \leq b$, wobei $A_0 x_0 = b_0$ und $\text{rang } A_0 = n$
- Teste ob x_0 eine optimale Lösung ist
- Bestimme $u \in \mathbb{R}^m$ mit $c^T = u^T A$ mit $u_i = 0$, falls Zeile i von A in A_0 nicht vorkommt (berechne $c^T A_0^{-1}$ und füge Nullen hinzu)

Fall 1. $u \geq 0$

Dann x_0 optimal, weil u optimale duale Lösung:

$$\begin{aligned} c^T x_0 &= u^T A_0 x_0 \\ &= u^T b \\ &\geq \min\{y^T b : y \geq 0, y^T A = c^T\} \\ &= \max\{c^T x : Ax \leq b\} \end{aligned}$$

Fall 2. $u \not\geq 0$

Sei i kleinster Index mit $u_i < 0$. Wähle $y \in \mathbb{R}^n$ mit $a_i^T y = -1$ und $a^T y = 0 \forall$ anderen Zeilen a^T von A_0 mit $a^T \neq a_i^T$ (y ist die entsprechende Spalte von $-A_0^{-1}$).

Fall 2.a) $a^T y \leq 0 \forall$ Zeilen a^T von A

Dann gilt $x_0 + \lambda y \in P \forall \lambda \geq 0$ und

$$c^T(x_0 + \lambda y) = c^T x_0 + \lambda \underbrace{c^T y}_{=u^T A y} \xrightarrow{\lambda \rightarrow +\infty} +\infty$$

Also ist das Problem unbeschränkt.

Fall 2.b) $a^T y > 0$ für eine Zeile a^T von A

Setze $\lambda_0 = \max\{\lambda : x_0 + \lambda y \in P\}$

$$\xRightarrow[\text{Minkowski-Weyl}]{\text{Theorem 6.5.1}} \min \left\{ \frac{b_j - a_j^T x_0}{a_j^T y} : j = 1, \dots, m, a_j^T y > 0 \right\}$$

Sei j der kleinste Index an dem das Minimum angenommen wird

Definiere die Matrix A_1 = ersetze in A_0 die Zeile a_i^T durch a_j^T

Definiere $x_1 = x_0 + \lambda_0 y$

Gehe zum Anfang mit A_1 und x_1 an stelle von A_0 und x_0 . Wiederhole das ganze bis $u \geq 0$ gilt oder der Fall 2.a) eintritt.

11.1. Der Algorithmus

Algorithmus 11.1.1 Simplexverfahren

Eingabe: $A_0 \in \mathbb{R}^{n \times n}$, x_0 Ecke von $P = \{x: Ax \leq b\}$

1: **for** $k \leftarrow 0$ **to** $+\infty$ **do**

2: Bestimme $u \in \mathbb{R}^m$

3: **end for**

Ausgabe:

11.07.2014
25. Vorlesung

Satz 11.1.1:

Das Simplexverfahren terminiert nach endlich vielen Schritten.

Beweis:

Variablen im k -ten Schritt: A_k , x_k , n_k , y_k . Es gilt

$$c^\top x_0 \leq c^\top x_1 \leq c^\top x_2 \leq \dots$$

wobei $c^\top x_k = c^\top x_{k+1}$ genau dann, wenn $x_k = x_{k+1}$, weil $c^\top x_{k+1} = c^\top (x_k + \lambda_0 y_k)$ mit $\lambda_0 \geq 0$ und $c^\top y_k = u_k^\top A y_k = -(u_k)_i > 0$. Angenommen das Verfahren landet in einer Endlosschleife, dann $\exists k, l$ mit $k < l$ so dass $A_k = A_l$. Dann gilt

$$x_k = x_{k+1} = \dots = x_l$$

Sei r der größte Index, so dass die Zeile $a_r^\top$ in einer Iteration aus A_t genommen wird, wobei $t = k, k+1, \dots, l-1$. Dies passiere im Schritt p .

Weil $A_k = A_l$, existiert ein Schritt q , so dass $a_r^\top$ wieder zu A_q hinzugenommen wird. Dann $k \leq p < q < l$.

Dann gilt für $j > r$:

$$a_j^\top \text{ kommt in } A_p \text{ vor} \Leftrightarrow a_j^\top \text{ kommt in } A_q \text{ vor}$$

Es gilt: $u_p^\top A y_q = c^\top y_q > 0$, also ist $(u_p^\top)_j (a_j^\top y_q) > 0$, also ist $(u_p^\top)_j (a_j^\top y_q) > 0$ für ein j , aber

Fall 1. $a_j^\top$ gehört nicht zu A_p . $(u_p^\top)_j = 0$ $\nmid$

Fall 2. $a_j^\top$ gehört zu A_p

Fall 2.a) $j > r$: $a_j^\top y_q = 0$ $\nmid$

Fall 2.b) $j = r$: $(u_p)_j < 0$ und $a_j^\top y_q > 0$ $\nmid$

Fall 2.c) $j < r$: $(u_p)_j \geq 0$ und $(a_j^\top y_q) \leq 0$ $\nmid$

Wie verwendet man den Simplexalgorithmus, wenn man keine Ecke von P kennt bzw. wenn es gar keine Ecke gibt?

Ohne Einschränkung: Das lineare Programm ist von der Form $\max\{c^\top x \mid x \geq 0, Ax \leq b\}$. Dies ist gleich $\max\{c^\top x \mid x \geq 0, A_1 x \leq b_1, A_2 x \geq b_2\}$ wobei $b_1 \geq 0$, $b_2 > 0$. Betrachte das lineare Programm $\max\{e^\top (A_2 x - \tilde{x}) : x, \tilde{x} \geq 0, A_1 x \leq b_1, A_2 x - \tilde{x} \leq b_2\}$, wobei

$e = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ der Einsvektor ist. Dann ist $x = 0$, $\tilde{x} = 0$ eine Ecke. Wir können also den

Simplexalgorithmus starten. Falls der optimale Wert $e^\top b_2$ ist, der an Ecke $x, \tilde{x}$ angenommen wird, dann ist x eine Ecke des Originalproblems und wir können das Originalproblem mit

dem Simplexalgorithmus lösen. Falls der optimale Wert kleiner als $e^T b_2$ ist, dann ist das Originalproblem ungültig. $\forall x, \tilde{x} \geq 0, A_1 x \leq b_1, A_2 x - \tilde{x} < b_2$, d. h. $\nexists x \geq 0: A_1 x \leq b_1, A_2 x \geq b_2$.

□

Effizienz des Simplexalgorithmus

Vorteile:

- + In der Praxis ist der Simplexalgorithmus eines der besten bzw. schnellsten Verfahren
- + Es gibt sehr gute Implementationen des Simplexalgorithmus
- + Spielman und Teng (2004): „smoothed analysis“
Der Simplexalgorithmus ist polynomiell, falls die Eingabedaten leicht zufällig „gestört“ sind

Nachteile:

- In der Theorie gibt es Beispiele, bei denen der Simplexalgorithmus exponentiell viele Schritte benötigt (Klee-Minty-Cube, 1972)
- Offenes Problem („polynomielle Hirsch-Vermutung“):
Ist der maximale Abstand zwischen zwei Ecken polynomiell in n, m ?

12. Die Ellipsoidmethode

16.07.2014
26. Vorlesung

12.1. Grundlegendes zu Ellipsoiden	97
12.2. Separieren und optimieren	99

12.1. Grundlegendes zu Ellipsoiden

Definition 12.1.1 (Ellipsoid):

Sei $A \in \mathbb{R}^{n \times n}$ eine positiv definite Matrix und $x \in \mathbb{R}^n$ ein Vektor. Als **Ellipsoid** bezeichnen wir dann

$$\mathcal{E}(A, x) = \{y \in \mathbb{R}^n : (y - x)^\top A^{-1} (y - x) \leq 1\}$$

Beispiel 12.1.1:

$$\begin{aligned} \mathcal{E}(r^2 I_n, 0) &= y^\top \underbrace{(r^2 I_n)^{-1}}_{\left(\frac{1}{r^2} I_n\right)} y \leq 1 \\ &\Leftrightarrow y^\top y \leq r^2 \\ &\Leftrightarrow \|y\|^2 \leq r^2 \\ &= r \cdot B_n \end{aligned}$$

wobei $B_n = \{y \in \mathbb{R}^n : \|y\| \leq 1\}$ die n -dimensionale Einheitskugel ist.

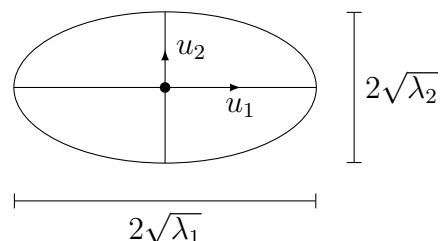
Satz 12.1.1 (Hauptachsentransformation/Spektralzerlegung):

Sei $A \in \mathbb{R}^{n \times n}$ symmetrisch. Es gibt positive Eigenwerte $\lambda_1, \dots, \lambda_n \in \mathbb{R}$ von A und eine Orthonormalbasis aus den zugehörigen Eigenvektoren $u_1, \dots, u_n \in \mathbb{R}^n$ mit

$$Au_i = \lambda_i u_i \text{ und } A = \sum_{i=1}^n \lambda_i \underbrace{\begin{pmatrix} u_i & u_i^\top \end{pmatrix}}_{n \times n \text{-Matrix}}$$

Beispiel 12.1.2:

u_i : Hauptachsenrichtung
 $2\sqrt{\lambda_i}$: Länge der Hauptachse



Volumen von $\mathcal{E}(A, x)$

$\text{vol } \mathcal{E}(A, x) = \sqrt{\det A} \text{vol } B_n$. Dabei ist $\text{vol } B_n = \frac{\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2}+1)}$, wobei die Gammafunktion rekursiv definiert wird als $\Gamma(\frac{1}{2}) = \sqrt{\pi}$, $\Gamma(1) = 1$, $\Gamma(x+1) = x \cdot \Gamma(x)$, d. h. $n! = \Gamma(n+1)$.

Satz 12.1.2:

Sei $K \subseteq \mathbb{R}^n$ eine konvexe und kompakte Menge. Dann existiert genau ein Ellipsoid $\mathcal{E}(K)$ mit den Eigenschaften

1. $K \subseteq \mathcal{E}(K)$
2. $\mathcal{E}(K)$ hat minimales Volumen

$\mathcal{E}(K)$ heißt **Loewner-John-Ellipsoid** von K .

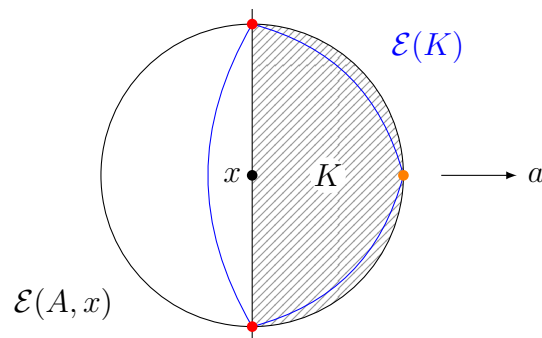
Beweis:

↪ Nächstes Semester in der Vorlesung „Nichtlineare Optimierung“.

□

Hier: Spezialfall $K = \mathcal{E}(A, x) \cap H^+ = \{y \in \mathbb{R}^n : a^\top y \geq a^\top x\}$ mit $a \neq 0$.

Beispiel 12.1.3:



Drei Punkte auf dem Rand definieren ein eindeutiges Ellipsoid. Zwei Punkte vom Schnitt der Kugel mit der Hyperebene („Äquator“, hier rot) und als drittes der Punkt, der am weitesten weg ist (hier orange).

Lemma 12.1.1:

Sei $A \in \mathbb{R}^{n \times n}$ positiv definit, $x \in \mathbb{R}^n$ und $a \in \mathbb{R}^n \setminus \{0\}$. Dann

$$\mathcal{E}(\mathcal{E}(A, x) \cap \{y \in \mathbb{R}^n : a^\top y \geq a^\top x\}) = \mathcal{E}(A', x')$$

wobei $A' = \frac{n^2}{n^2-1} \left(A - \frac{2}{n+1} b b^\top \right)$, $x' = x + \frac{1}{n} b$, $b = \frac{1}{\sqrt{a^\top A a}} A a$ und $\frac{\text{vol } \mathcal{E}(A', x')}{\text{vol } \mathcal{E}(A, x)} \leq e^{-\frac{1}{2(n+1)}} < 1$.

Beweis:

Übung.

□

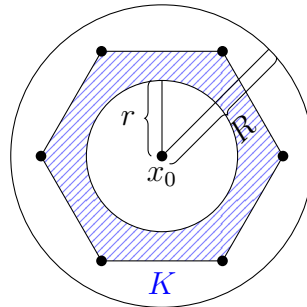
12.2. Separieren und optimieren

Voraussetzung:

Sei $\mathcal{K}$ eine Klasse von konvexen und kompakten Mengen. Angenommen wir kennen für jedes $K \in \mathcal{K}$ einen Vektor $x_0 \in \mathbb{R}^n$ und $r, R > 0$ mit

$$x_0 + rB_n \subseteq K \subseteq x_0 + RB_n$$

Beispiel 12.2.1:

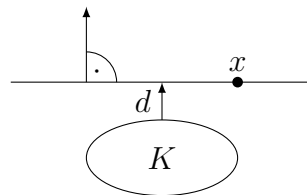


Separationsproblem

Eingabe: $K \in \mathcal{K}$, $x \in \mathbb{R}^n$

Ausgabe: „ $x \in K$ “ oder $d \in \mathbb{R}^n \setminus \{0\}$ mit $d^\top x > \max_{y \in K} d^\top y$

Die Hyperebene $\{y \in \mathbb{R}^n : d^\top y = d^\top x\}$ trennt $\{x\}$ von K .



Optimierungsproblem

Eingabe:

1. Optimierungsrichtung: $c \in \mathbb{R}^n$ mit $\|c\| = 1$
2. Genauigkeit: $\varepsilon > 0$
3. $K \in \mathcal{K}$

Ausgabe: $x \in K$ mit $c^\top x \geq \max_{y \in K} c^\top y - \varepsilon$

Beispiel 12.2.2:

Falls K ein (beschränkter) Polyeder P ist, gegeben durch $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, dann ist das Separationsproblem einfach zu lösen:

Überprüfe für gegebenes x alle linearen Ungleichungen $a_i^\top x \leq b_i$, $i = 1, \dots, m$. Dann ist $x \in K$ genau dann, wenn alle Ungleichungen erfüllt sind. Falls $a_i^\top x > b_i$, dann setze $d = a_i$.

Theorem 12.2.1 (Grötschel, Lovász, Schrijver, 1981, basierend auf der Ellipsoidmethode):

Für die Klasse $\mathcal{K}$ lässt sich das Optimierungsproblem durch N -faches lösen des Separationsproblems lösen, wobei N so gewählt ist, dass folgende Ungleichung gilt:

$$2 \cdot \frac{R^2}{r} \cdot e^{-\frac{N}{2(n+1)n}} \leq \varepsilon$$

Beweis:

Siehe [Bemerkung 12.2.1](#).

□

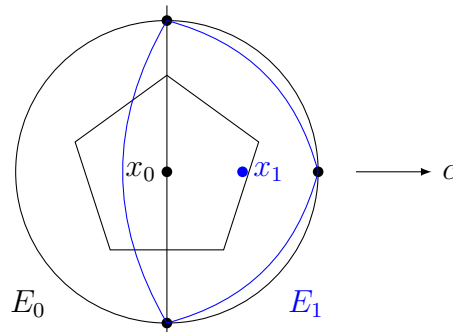
Algorithmus 12.2.1 Ellipsoidmethode

```

1:  $E_0 = \mathcal{E}(R^2 I_n, x_0)$   $\triangleright E_0 = x_0 + RB_n \supseteq K$ 
2: for  $k \leftarrow 0$  to  $N - 1$  do
3:   Löse das Separationsproblem für den Mittelpunkt  $x_k$  von  $E_k$ 
4:   if  $x_k \in K$  then  $\triangleright k$  gültiger Index
5:      $a = c$ 
6:   else
7:      $a = -d$   $\triangleright d$  gegeben durch das Separationsproblem
8:   end if
9:    $E_{k+1} = \mathcal{E}(E_k \cap \{y \in \mathbb{R}^n : a^\top y \geq a^\top x_k\})$ 
10: end for

```

Beispiel 12.2.3:

**Lemma 12.2.1:**

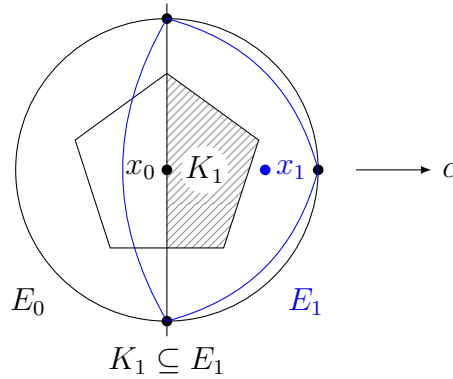
Sei $\xi_k = \max\{c^\top x_j : 0 \leq j < k, x_j \in K\}$, wobei x_j ein Mittelpunkt aus Algorithmus 12.2.1 ist. Sei $K_k \subseteq \mathbb{R}^n$ definiert durch $K_k = K \cap \{x \in \mathbb{R}^n : c^\top x \geq \xi_k\}$. Dann gilt

$$K_k \subseteq E_k$$

Beweis (per Induktion über k):

Übung (siehe Aufgabe 8.1).

□

Beispiel 12.2.4:

Lemma 12.2.2:

Sei $k \in \mathbb{N}$ und j sei ein gültiger Index ($0 \leq j < k$, $x_j \in K$) mit $\xi_k = c^\top x_j$. Dann gilt

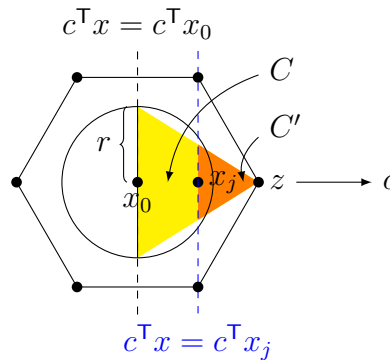
$$\xi_k = c^\top x_j \geq \max_{y \in K} c^\top y - \underbrace{2 \cdot \frac{R^2}{r} \cdot e^{-\frac{k}{2(n+1)n}}}_{\text{abhängig von } k} \leq \varepsilon \quad \text{für } k \text{ groß}$$

Beweis:

Wir wissen:

$$\frac{\text{vol } E_{k+1}}{\text{vol } E_k} \leq e^{-\frac{1}{2(n+1)}} < 1 \Rightarrow \frac{\text{vol } E_k}{\text{vol } E_0} \leq e^{-\frac{k}{2(n+1)}}$$

Sei $z \in K$ eine optimale Lösung $c^\top z = \max_{y \in K} c^\top y$. Betrachte den (runden) Kegelstumpf C mit Spitze z und Basis $(x_0 + rB_n) \cap \{y \in \mathbb{R}^n : c^\top y = c^\top x_0\}$.



Sei $C' = C \cap \{x \in \mathbb{R}^n : c^\top x \geq c^\top x_j\}$. Dann ist

$$\text{vol } C' = \frac{r^{n-1} \text{vol } B_{n-1}}{n} (c^\top z - c^\top x_0) \left(\frac{c^\top z - c^\top x_j}{c^\top z - c^\top x_0} \right)^n$$

Nach Lemma 12.2.1 ist $C' \subseteq K_k \subseteq E_k$ also

$$\begin{aligned} \text{vol } C' &\leq \text{vol } E_k \\ &\leq e^{-\frac{k}{2(n+1)}} \text{vol } E_0 \\ &= R^n e^{-\frac{k}{2(n+1)}} \text{vol } B_n \end{aligned}$$

Betrachte $c^\top z - c^\top x_0$. Nach Cauchy-Schwarz gilt:

$$|c^\top z - c^\top x_0| = |c^\top(z - x_0)| \leq \underbrace{\|c\|}_{=1} \cdot \underbrace{\|z - x_0\|}_{\leq R} \leq R$$

Zusammen:

$$\begin{aligned} (c^\top z - c^\top x_j)^n &\leq e^{-\frac{k}{2(n+1)}} R^n \operatorname{vol} B_n \frac{n}{r^{n-1} \operatorname{vol} B_{n-1}} \underbrace{(c^\top z - c^\top x_0)^{n-1}}_{\leq R^{n-1}} \\ &= \left(\frac{n \operatorname{vol} B_n}{\operatorname{vol} B_{n-1}} \right) \frac{R^{2n-1}}{r^{n-1}} e^{-\frac{k}{2(n+1)}} \\ \Rightarrow c^\top z - c^\top x_j &\leq \underbrace{\left(\frac{n \operatorname{vol} B_n}{\operatorname{vol} B_{n-1}} \right)^{\frac{1}{n}}}_{\leq 2} \cdot \frac{R^2}{r} \cdot e^{-\frac{k}{2(n+1)n}} \end{aligned}$$

□

Bemerkung 12.2.1:

Aus [Lemma 12.2.2](#) folgt unmittelbar [Theorem 12.2.1](#).

Bemerkung 12.2.2:

Allerdings haben wir an einem wichtigen Punkt idealisiert. Wir können nicht mit unendlicher Genauigkeit rechnen, weil in der Definition von $b = \frac{1}{\sqrt{a^\top A a}} A a$ Wurzeln vorkommen, welche sich in Rechnern nicht exakt darstellen lassen.

Mit einigem technischen Aufwand kann dies zum Teil gelöst werden.

Bemerkung 12.2.3:

In der Praxis ist die Ellipsoidmethode nur effizient, falls weniger als 1000 Variablen verwendet werden, sonst ist das Simplexverfahren besser.

13. Die Innere-Punkte-Methode

18.07.2014
27. Vorlesung

13.1. Der zentrale Pfad	103
13.2. Der Algorithmus und seine Analyse	108

Vorteile:

- + Löst effizient lineare Programme (in Theorie und Praxis)
- + erweiterbar (zu einer größeren Klasse von Optimierungsproblemen)
- + einfach zu implementieren

13.1. Der zentrale Pfad

LP-Dualität

$$\begin{array}{ll}
 \min c^T x & = \max b^T y \\
 Ax = b & A^T y \leq c \\
 x \geq 0 & (\Leftrightarrow y^T A \leq c^T)
 \end{array} \quad (\text{LP})$$

Führe die Variable $s \in \mathbb{R}^n$, $s \geq 0$ ein:

$$\begin{array}{ll}
 A^T y + s = c \text{ so dass } \min c^T x & = \max b^T y \\
 Ax = b & A^T y \leq c \\
 x \geq 0 & s \geq 0
 \end{array}$$

Optimalitätskriterium (siehe Kapitel 6: Polyedertheorie)

$$\underbrace{(x^*)}_{\text{primal}}, \underbrace{(y^*, s^*)}_{\text{dual}} \text{ ist optimal} \Leftrightarrow A^T y^* + s^* = c$$

$$\begin{array}{l}
 Ax^* = b \\
 x_i^* s_i^* = 0, \quad i = 1, \dots, n \\
 x^* \geq 0 \\
 s^* \geq 0
 \end{array}$$

Idee: Betrachte die (nichtlineare) Abbildung

$$F: \mathbb{R}^{2n+m} \rightarrow \mathbb{R}^{2n+m}$$

$$(x, y, s) \mapsto \begin{pmatrix} A^T y + s - c \\ Ax - b \\ XSe \end{pmatrix}$$

wobei $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $s \in \mathbb{R}^n$ und $X = \text{diag}(x_1, \dots, x_n)$, $S = \text{diag}(s_1, \dots, s_n)$ und $e = (1 \ \dots \ 1)^T$, also $A^T y + s - c \in \mathbb{R}^n$, $Ax - b \in \mathbb{R}^m$, $XSe \in \mathbb{R}^n$. Finde (x^*, y^*, s^*) mit $F(x^*, y^*, s^*) = 0$, z. B. mit dem Newton-Verfahren.

Problem:

Es gibt viele (x^*, y^*, s^*) mit $F(x^*, y^*, s^*) = 0$, die nicht zur Menge der **zulässigen Lösungen**

$$\mathcal{F} = \{(x, y, s): A^T y + s = c, Ax = b, x \geq 0, s \geq 0\}$$

gehören. Sei

$$\mathcal{F}^0 = \{(x, y, s) \in \mathcal{F}: x > 0, s > 0\}$$

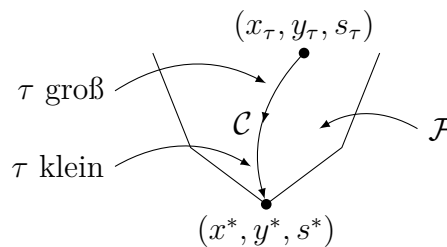
die Menge der **strikt zulässigen Lösungen**, d. h. x und s liegen im *Innern* des nicht-negativen Orthanten $\mathbb{R}_{\geq 0}^n = \{z \in \mathbb{R}^n: z \geq 0\}$.

Sei $\tau > 0$. Suche (x_τ, y_τ, s_τ) mit $F(x_\tau, y_\tau, s_\tau) = \begin{pmatrix} 0 \\ 0 \\ \tau e \end{pmatrix}$, d. h. $(x_\tau)_i (s_\tau)_i = \tau$. Wir zeigen gleich, dass $(x_\tau, y_\tau, s_\tau) \in \mathcal{F}^0$ existieren und eindeutig bestimmt sind.

Definition 13.1.1 (zentrale Pfad):

Die Menge $\mathcal{C} = \{(x_\tau, y_\tau, s_\tau): \tau > 0\}$ heißt der **zentrale Pfad** von (LP).

Beispiel 13.1.1:



Lemma 13.1.1:

$$\lim_{\tau \rightarrow 0} c^T x_\tau - b^T y_\tau = 0$$

Beweis:

$$\begin{aligned} c^T x_\tau - b^T y_\tau &= (A^T y_\tau + s_\tau)^T x_\tau - b^T y_\tau \\ &= y_\tau^T A x_\tau + s_\tau^T x_\tau - b^T y_\tau \\ &= \cancel{y_\tau^T b} + s_\tau^T x_\tau - \cancel{b^T y_\tau} \\ &= n\tau \xrightarrow{\tau \rightarrow 0} 0 \end{aligned}$$

□

Definition 13.1.2 (Barrierefunktion):

Definiere $H^0 = \{(x, s) \in \mathbb{R}^{2n} \mid \exists y: (x, y, s) \in \mathcal{F}^0\}$ und die **Barrierefunktion** $f_\tau: H^0 \rightarrow \mathbb{R}$ durch

$$f_\tau(x, s) = \frac{1}{\tau} x^\top s - \sum_{i=1}^n \ln(x_i s_i)$$

(Falls $(x, s) \rightarrow \text{Rand von } H^0 \Rightarrow f_\tau(x, s) \rightarrow +\infty$)

Ziel: An (x_τ, s_τ) nimmt f_τ sein Minimum an.

Lemma 13.1.2:

f_τ ist strikt konvex.

Beweis:

1. Der Term $x^\top s$ ist linear in H^0 . Sei $\bar{x}$ mit $A\bar{x} = b$. Dann gilt für $(x, s) \in H^0$:

$$\begin{aligned} x^\top s &= x^\top (c - A^\top y) \\ &= x^\top c - b^\top y \\ &= x^\top c - (A\bar{x})^\top y \\ &= x^\top c - \bar{x}^\top A^\top y \\ &= x^\top c - \bar{x}^\top (c - s) \end{aligned}$$

2. Die Funktion $t \rightarrow -\ln t$ ist strikt konvex, weil $\frac{\partial^2}{\partial t^2} -\ln t = \frac{1}{t^2} > 0$.

Aus 1. und 2. folgt die Behauptung. □

Lemma 13.1.3:

$$\exists C: f_\tau(x, s) \geq C \quad \forall (x, s) \in H^0$$

Beweis:

Hierzu formen wir zunächst $f_\tau(x, s)$ wie folgt um:

$$\begin{aligned} f_\tau(x, s) &= \frac{1}{\tau} x^\top s - \sum_{i=1}^n \ln(x_i s_i) \\ &= \sum_{i=1}^n \frac{x_i s_i}{\tau} - \ln(x_i s_i) \\ &= \sum_{i=1}^n \frac{x_i s_i}{\tau} - \left(\ln\left(\frac{x_i s_i}{\tau}\right) + \ln(\tau) \right) \\ &= \sum_{i=1}^n \left(\frac{x_i s_i}{\tau} - \ln\left(\frac{x_i s_i}{\tau}\right) \right) - n \ln(\tau) \end{aligned}$$

Da nun $x, s \in \mathbb{R}_{>0}^{2n}$ für $(x, s) \in H^0$ und auch $\tau > 0$, betrachten wir die Funktion $g: \mathbb{R}_{>0} \rightarrow \mathbb{R}$ gegeben durch $g(y) = y - \ln(y)$. Dann ist g konvex und es ist leicht zu sehen, dass das globale Minimum für $y = 1$ angenommen wird, wobei $g(1) = 1$. Also können wir jeden Summanden nach unten mit 1 abschätzen und erhalten

$$f_\tau(x, s) \geq n - n \ln(\tau)$$

Alternativ reicht es auch aus zu sehen das die Summanden alle positiv sind, um die etwas schwächere Schranke $-n \ln(\tau)$ zu erhalten. □

Lemma 13.1.4:

Für $K \geq 0$ ist die Menge $\{(x, s) \in H^0: x^\top s \leq K\}$ beschränkt.

Beweis:

Übung. □

Lemma 13.1.5:

Sei $\kappa \in \mathbb{R}$. Dann ist $\{(x, s) \in H^0: f_\tau(x, s) \leq \kappa\}$ in einer kompakten Menge enthalten.

Beweis:

Für $g(t) = t - \ln t - 1$ ist $f_\tau(x, s) = \sum_{i=1}^n g\left(\frac{x_i s_i}{\tau}\right) + n - n \ln \tau$.

Dann $f_\tau(x, s) \leq \kappa \Leftrightarrow \sum_{i=1}^n g\left(\frac{x_i s_i}{\tau}\right) \leq \bar{\kappa} = \kappa - n + n \ln \tau$.

Für Index $i = 1, \dots, n$ gilt

$$g\left(\frac{x_i s_i}{\tau}\right) \leq \bar{\kappa} - \sum_{\substack{j=1 \\ j \neq i}}^n g\left(\frac{x_j s_j}{\tau}\right) \leq \bar{\kappa}$$

weil $g(t) \geq 0$ für $t \geq 0$. Weil $g(t) \rightarrow +\infty$ für $t \rightarrow 0$ und $t \rightarrow \infty$ gilt:

$$\exists M: \frac{1}{M} \leq x_i s_i \leq M, \quad i = 1, \dots, n$$

Also $x^\top s \leq nM$. Nach **Lemma 13.1.4** existiert eine Schranke M_u , so dass $x_i \leq M_u$, $s_i \leq M_u$. Also $\frac{1}{M} \leq x_i s_i \leq M_u s_i \Rightarrow s_i \geq \frac{1}{MM_u}$, genauso $x_i \geq \frac{1}{MM_u}$, d. h. $\frac{1}{MM_u} \leq x_i s_i \leq M_u$ für $i = 1, \dots, n$. □

Theorem 13.1.1 (Extrema mit Nebenbedingungen):

Sei $U \subseteq \mathbb{R}^n$ offen. Seien $f: U \rightarrow \mathbb{R}$ und $g_1, \dots, g_N: U \rightarrow \mathbb{R}$ stetig differenzierbare Funktionen. Falls $x^* \in \mathbb{R}^n$ ein lokales Extremum von f unter der Nebenbedingung

$$g_1(x^*) = \dots = g_N(x^*) = 0$$

ist, dann $\exists \lambda_1, \dots, \lambda_N \in \mathbb{R}$ (Lagrange-Multiplikatoren) mit $\nabla f(x^*) = \sum_{i=1}^N \lambda_i \nabla g_i(x^*)$.

Beweis:

Bereits bekannt aus der Analysis. □

Theorem 13.1.2:

Der zentrale Pfad existiert und ist eindeutig. Falls $\mathcal{F}^0 \neq \emptyset$ und $\tau > 0$ dann

$$\exists!(x_\tau, y_\tau, s_\tau) \in \mathcal{F}^0: F(x_\tau, y_\tau, s_\tau) = \begin{pmatrix} 0 \\ 0 \\ \tau e \end{pmatrix}, \quad x_\tau > 0, \quad s_\tau > 0$$

Beweis:

Behauptung:

Das eindeutige Minimum (x^*, s^*) von f_τ ist (x_τ, s_τ) .

Beweis (Existenz):

(x^*, s^*) löst das Minimierungsproblem (in den Variablen x, y, s)

$$\begin{aligned} \min f_\tau(x, s) \\ Ax &= b \\ A^\top y + s &= c \\ x &> 0 \\ s &> 0 \end{aligned}$$

Insbesondere ist (x^*, s^*) ein lokales Minimum von

$$\begin{aligned} \min f_\tau(x, s) \\ Ax &= b \\ A^\top y + s &= c \end{aligned}$$

Nach Anwendung von [Theorem 13.1.1](#) folgt hier:

$$\nabla f_\tau(x, s) = \begin{pmatrix} \frac{\partial}{\partial x} f_\tau(x, s) \\ \frac{\partial}{\partial y} f_\tau(x, s) \\ \frac{\partial}{\partial s} f_\tau(x, s) \end{pmatrix} = \begin{pmatrix} \frac{1}{\tau} s - X^{-1}e \\ 0 \\ \frac{1}{\tau} x - S^{-1}e \end{pmatrix}$$

$$\begin{aligned} g_1(x, y, s) &= (Ax - b)_1 \\ \vdots &= \vdots \\ g_m(x, y, s) &= (Ax - b)_m \\ g_{m+1}(x, y, s) &= (A^\top y + s - c)_1 \\ \vdots &= \vdots \\ g_{m+n}(x, y, s) &= (A^\top y + s - c)_n \end{aligned}$$

$\nabla g_1(x, y, s) = a_1$ wobei a_1 die erste Zeile von A ist. Also $\exists \lambda \in \mathbb{R}^m$ und $\mu \in \mathbb{R}^n$.

1. $\frac{1}{\tau} s - X^{-1}e = A^\top \lambda$
2. $0 = A\mu$
3. $\frac{1}{\tau} x - S^{-1}e = \mu$

Aus den Punkten 2. und 3. folgt $A\left(\frac{x}{\tau} - S^{-1}e\right) = 0$. Punkt 1. multipliziert mit $\frac{x}{\tau} S^{-1}e$ ergibt:

$$\underbrace{\left(\frac{x}{\tau} S^{-1}e\right)^\top \left(\frac{s}{\tau} X^{-1}e\right)}_{\left(\frac{1}{\tau} X e - S^{-1}e\right)^\top \left(\frac{1}{\tau} S e - X^{-1}e\right)} = \underbrace{\left(\frac{x}{\tau} S^{-1}e\right)^\top A^\top \lambda}_{=0}$$

Definiere $X^{\frac{1}{2}} = \text{diag}(\sqrt{x_1}, \dots, \sqrt{x_n})$ und $S^{\frac{1}{2}} = \text{diag}(\sqrt{s_1}, \dots, \sqrt{s_n})$. Dann

$$\begin{aligned} 0 &= \left(\frac{1}{\tau} X e - S^{-1} e \right)^{\top} \underbrace{\left(X^{-\frac{1}{2}} S^{\frac{1}{2}} \right) \left(X^{\frac{1}{2}} S^{-\frac{1}{2}} \right)}_{=I} \left(\frac{1}{\tau} S e - X^{-1} e \right) \\ &= \left\| \frac{1}{\tau} (X S)^{\frac{1}{2}} e - (X S)^{-\frac{1}{2}} e \right\|^2 \end{aligned}$$

$$\begin{aligned} \Rightarrow \frac{1}{\tau} (X S)^{\frac{1}{2}} e - (X S)^{-\frac{1}{2}} e &= 0 \\ \Rightarrow X S e &= \tau e \end{aligned}$$

□

Definition 13.1.3 (Dualitätsmaß):

Sei $(x, y, s) \in \mathcal{F}^0$. Das **Dualitätsmaß** von (x, y, s) ist $\mu = \frac{x^{\top} s}{n} (= c^{\top} x - b^{\top} y)$

Definition 13.1.4 (Nachbarschaft):

Sei $\theta \in [0, 1)$. Die **Nachbarschaft** des zentralen Pfads ist

$$N(\theta) = \left\{ (x, y, s) \in \mathcal{F}^0 : \|X S e - \mu e\| \leq \theta \mu \right\}$$

Lemma 13.1.6:

1. $(x, y, s) \in \mathcal{C} \Leftrightarrow (x, y, s) \in N(0)$
2. Falls $\|X S e - \mu e\| < 1 \Rightarrow x > 0, s > 0$

Beweis:

1. trivial ✓
2. Kontraposition:
Falls $x_i = 0$ oder $s_i = 0$, dann $\|X S e - \mu e\| \geq |x_i s_i - \mu| = \mu \Rightarrow$ Behauptung.

□

13.2. Der Algorithmus und seine Analyse

Definition 13.2.1 (modifizierter Newton-Schritt):

Für den Zentrierungsparameter $\sigma \in [0, 1]$ definiere den **modifizierten Newton-Schritt**

$$(\Delta x, \Delta y, \Delta s) \in \mathbb{R}^{2n+m}$$

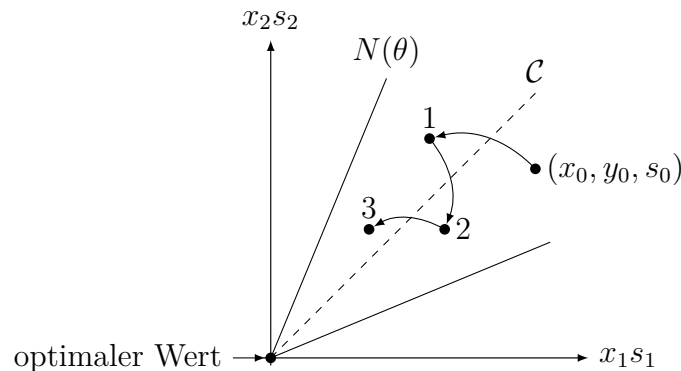
durch das lineare Gleichungssystem

$$\underbrace{\begin{pmatrix} 0 & A^{\top} & I \\ A & 0 & 0 \\ S & 0 & X \end{pmatrix}}_{\nabla F(x, y, s)} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta s \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X S e + \sigma \mu e \end{pmatrix}$$

Algorithmus 13.2.1 Innere-Punkte-Methode**Eingabe:** $\theta = 0.4$, $\sigma = 1 - \frac{0.4}{\sqrt{n}}$, $N = \text{Anzahl der Iterationen}$ Startpunkt: $(x_0, y_0, s_0) \in N(\theta)$ 1: **for** $k \leftarrow 0$ **to** $N - 1$ **do**2: Bestimme $(\Delta x_k, \Delta y_k, \Delta s_k)$ durch

3:

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S_k & 0 & X_k \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta y_k \\ \Delta s_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X_k S_k e + \sigma \mu_k e \end{pmatrix}$$

4: $(x_{k+1}, y_{k+1}, s_{k+1}) = (x_k, y_k, s_k) + (\Delta x_k, \Delta y_k, \Delta s_k)$ 5: **end for****Beispiel 13.2.1:****Geschichte der Algorithmen zum lösen von linearen Programmen**

1947	Dantzig:	Simplexalgorithmus
1972		Klee-Minty-Cube
1979	Khachiyan:	Ellipsoidmethode
1984	Karmarkar:	Innere-Punkte-Methode (Start der „Innere-Punkte-Revolution“)
1994	Nesterov & Nemirovski:	Erweiterung der Innere-Punkte-Methode auf nichtlineare Programme

↔ Vergleiche mit Algorithmenliste auf Seite 64

Lemma 13.2.1:Falls $(x_k, y_k, s_k) \in \mathcal{F}^0$, dann gilt

1. $(\Delta x_k)^T (\Delta s_k) = 0$

$$2. \mu_k = \underbrace{\left(1 - \frac{0.4}{\sqrt{n}}\right)}_{=\sigma} \mu_{k-1}$$

Beweis:

1. Aus der ersten „Zeile“ des Systems aus [Definition 13.2.1](#) entnehmen wir, dass $A^\top \Delta y + \Delta s = 0$ und aus der zweiten, dass $A \Delta x = 0$. Somit ist

$$(\Delta x)^\top (\Delta s) = (\Delta x)^\top (-A^\top \Delta y) = -(A \Delta x)^\top (\Delta y) = 0^\top \Delta y = 0$$

2. $\mu_k = \frac{x_k^\top s_k}{n}$

$$\begin{aligned} x_k^\top s_k &= (x_{k-1} + \Delta x_{k-1})^\top (s_{k-1} + \Delta s_{k-1})^\top \\ &= x_{k-1}^\top s_{k-1} + \underbrace{\Delta x_{k-1}^\top s_{k-1} + x_{k-1}^\top \Delta s_{k-1}}_{s_{k-1} \Delta x_{k-1} + x_{k-1} \Delta s_{k-1}} + \underbrace{\Delta x_{k-1}^\top \Delta s_{k-1}}_{\stackrel{!}{=} 0} \\ &= -X_{k-1} S_{k-1} e + \sigma \mu_{k-1} e \end{aligned}$$

$$\begin{aligned} \xRightarrow[\text{der Einträge}]{\text{Aufsummieren}} (\Delta x_{k-1})^\top s_{k-1} + x_{k-1}^\top \Delta s_{k-1} &= -x_{k-1}^\top s_{k-1} + \sigma n \mu_{k-1} \\ &= (-1 + \sigma) x_{k-1}^\top s_{k-1} \end{aligned}$$

$$\begin{aligned} \Rightarrow x_k^\top s_k &= \sigma x_{k-1}^\top s_{k-1} \\ \Rightarrow \mu_k &= \sigma \mu_{k-1} \end{aligned}$$

□

Beispiel 13.2.2:

Seien $\mu_0 = 10^6$, $n = 10000$ und $N = 5000$. Dann

$$\begin{aligned} \mu_N &= \sigma^N \mu_0 \\ &= \left(1 - \frac{0.4}{\sqrt{n}}\right)^N \cdot \mu_0 \\ &= \left(1 - \frac{0.4}{100}\right)^{5000} \cdot 10^6 \\ &\leq 0.02 \end{aligned}$$

Es bleibt zu zeigen, dass $(x_k, y_k, s_k) \in \mathcal{F}^0$. Dafür werden wir zeigen, dass $(x_k, y_k, s_k) \in N(\theta)$.

Lemma 13.2.2:

Sei $u, v \in \mathbb{R}^n$, $u^\top v \geq 0$. Dann gilt

$$\|U V e\| \leq 2^{-\frac{3}{2}} \|u + v\|^2$$

Beweis (nach [5]):

Es ist

$$\begin{aligned} 0 \leq u^\top v &= \sum_{u_i v_i > 0} u_i v_i + \sum_{u_i v_i < 0} u_i v_i \\ &= \sum_{i \in P} |u_i v_i| - \sum_{i \in M} |u_i v_i| \end{aligned}$$

wobei $P = \{i: u_i v_i > 0\}$ und $M = \{i: u_i v_i < 0\}$, also

$$\sum_{i \in M} |u_i v_i| \leq \sum_{i \in P} |u_i v_i| \quad (13.1)$$

Somit

$$\begin{aligned}
\|UVe\| &= \left(\|(u_i v_i)_{i \in P}\|^2 + \|(u_i v_i)_{i \in M}\|^2 \right)^{\frac{1}{2}} \\
&\stackrel{(1)}{\leq} \left(\|(u_i v_i)_{i \in P}\|_1^2 + \|(u_i v_i)_{i \in M}\|_1^2 \right)^{\frac{1}{2}} \\
&\stackrel{(13.1)}{\leq} \left(2 \|(u_i v_i)_{i \in P}\|_1^2 \right)^{\frac{1}{2}} \\
&= \sqrt{2} \|(u_i v_i)_{i \in P}\|_1 \\
&\stackrel{(2)}{\leq} \sqrt{2} \left\| \left(\frac{1}{4} (u_i + v_i)^2 \right)_{i \in P} \right\|_1 \\
&= 2^{-\frac{3}{2}} \sum_{i \in P} (u_i + v_i)^2 \\
&\leq 2^{-\frac{3}{2}} \|u + v\|^2
\end{aligned}$$

Dabei gilt (1), weil $\|\cdot\|_2 \leq \|\cdot\|_1$. (2) gilt, weil für $\alpha, \beta \in \mathbb{R}$ immer $\alpha\beta \leq \frac{1}{4}(\alpha + \beta)^2$ erfüllt ist. □

Lemma 13.2.3:

Sei $(x_k, y_k, s_k) \in N(0.4)$. Dann gilt

$$\|X_{k+1}S_{k+1}e - \mu_{k+1}e\| \leq 0.2\mu_k$$

d. h. insbesondere $(x_{k+1}, y_{k+1}, s_{k+1}) \in N(0.4)$.

Beweis:

Behauptung:

$$\|X_{k+1}S_{k+1}e - \mu_{k+1}e\| = \|\Delta X_k \Delta S_k e\|$$

Beweis:

Durch einfaches nachrechnen.

Behauptung:

$$\|\Delta X_k \Delta S_k e\| \leq 0.2\mu_k$$

Beweis:

Definiere $D_k = X_k^{\frac{1}{2}} S_k^{-\frac{1}{2}}$

$$\begin{aligned}
\|\Delta X_k \Delta S_k e\| &= \left\| \underbrace{D_k^{-1} \Delta X_k}_u \underbrace{D_k \Delta S_k}_v e \right\| \\
&\stackrel{\text{Lemma 13.2.2}}{\leq} 2^{-\frac{3}{2}} \|D_k^{-1} \Delta X_k + D_k \Delta S_k\|^2
\end{aligned}$$

Nun ist $D_k^{-1}\Delta X_k + D_k\Delta S_k = (X_k S_k)^{-\frac{1}{2}}(-X_k S_k e + \sigma \mu_k e)$ weil $S_k \Delta X_k + X_k \Delta S_k = -X_k S_k e + \sigma \mu_k e$ gilt, und dies multipliziert mit $(X_k S_k)^{-\frac{1}{2}}$ liefert die obige Gleichung. Also

$$\begin{aligned}\|\Delta X_k \Delta S_k e\| &\leq 2^{-\frac{3}{2}} \sum_{i=1}^n \frac{(-x_{k,i} s_{k,i} + \sigma \mu_k)^2}{x_{k,i} s_{k,i}} \\ &\leq 2^{-\frac{3}{2}} \sum_{i=1}^n \frac{(x_{k,i} s_{k,i} - \sigma \mu_k)^2}{\min_{j=1,\dots,n} x_{k,j} s_{k,j}}\end{aligned}$$

Da $(x_k, y_k, s_k) \in N(0.4)$ ist, gilt $\min_{j=1,\dots,n} x_{k,j} s_{k,j} \geq (1-\theta)\mu_k$, weil $\|X_k S_k e - \mu_k e\| \leq \mu_k \theta$
 $\Rightarrow |x_{k,j} \cdot s_{k,j} - \mu_k| \leq \mu_k \theta$
 $\Rightarrow x_{k,j} s_{k,j} - \mu_k \geq -\mu_k \theta$
 Des Weiteren gilt $e^\top (X_k S_k e - \mu_k e) = x_k^\top s_k - n\mu_k = 0$. Also

$$\begin{aligned}\sum_{i=1}^n (x_{k,i} s_{k,i} - \sigma \mu_k)^2 &= \|X_k S_k e - \sigma \mu_k e\|^2 \\ &= \|(X_k S_k e - \mu_k e) + (1-\sigma)\mu_k e\|^2 \\ &= \|X_k S_k e - \mu_k e\|^2 + 2 \underbrace{\left((1-\sigma)\mu_k e \right)^\top (X_k S_k e - \mu_k e)}_{=0} + (1-\sigma)^2 \mu_k^2 e^\top e \\ &\leq \theta^2 \mu_k^2 + (1-\sigma)^2 \mu_k^2 n\end{aligned}$$

$$\text{Zusammen: } \|\Delta X_k \Delta S_k e\| \leq 2^{-\frac{3}{2}} \frac{\theta^2 \mu_k^2 + (1-\theta)^2 \mu_k^2 n}{(1-\theta)\mu_k} \leq 0.2\mu_k$$

□

Verzeichnisse

A. Bibliografie	I
B. Index	II
C. Algorithmen	V
D. Definitionen	VI
E. Theoreme	VIII

A. Bibliografie

A.a. Literaturverzeichnis

- [1] Alexander Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Berlin, Heidelberg: Springer, 2003. ISBN: 978-3-540-44389-6 (siehe S. 36).
- [2] Alexander Schrijver. *Kombinatorische Optimierung*. Vorlesungsskript (im [Anhang](#)). Institut für Mathematik, 2013 (siehe S. 36).
- [3] Alexander Schrijver. *Theory of Linear and Integer Programming*. 1. Auflage. New York: John Wiley & Sons, 1998. ISBN: 978-0-471-98232-6 (siehe S. 61, 80).
- [4] John von Neumann und Oskar Morgenstern. *Theory of Games and Economic Behaviour*. New York: Oxford University Press, 1947. URL: <https://archive.org/details/theoryofgamesand030098mbp> (besucht am 22.05.2014) (siehe S. 85).
- [5] Stephen J. Wright. *Primal-Dual Interior-Point Methods*. Philadelphia: SIAM, 1997. ISBN: 978-0-898-71382-4 (siehe S. 110).
- [6] William J. Cook u. a. *Combinatorial Optimization*. New York: John Wiley & Sons, 2011. ISBN: 978-1-118-03139-1.
- [7] George Bernard Dantzig. *Linear Programming and Extensions*. Freiburg i.B., 1963. URL: <http://www.rand.org/pubs/reports/R366.html> (besucht am 22.05.2014).
- [8] Bernhard H. Korte und Jens Vygen. *Kombinatorische Optimierung - Theorie Und Algorithmen*. 2. deutsche Auflage. Berlin: Springer, 2012. ISBN: 978-3-540-76919-4.
- [9] Christos H. Papadimitriou und Kenneth Steiglitz. *Combinatorial optimization - algorithms and complexity*. Unabridged. New York: Dover Publications, 1998. ISBN: 978-0-486-40258-1.

B. Index

-Symbole-

$\mathcal{K}$ „konvexe & kompakte Menge“ 99

-A-

Abstand 10
 affin unabhängig 47
 Antwort
 beste 86

-B-

Barrierefunktion 105
 Baum 18

-C-

chromatische Zahl 24
 Chvátal-Gomory-Abschluss 79
 Chvátal-Rang 80

-D-

Differenz
 symmetrisch 23
 Dimension 48
 Dualitätsmaß 108

-E-

Ellipsoid 97
 Loewner-John-Ellipsoid 98
 Extrempunkt 54

-F-

Fluss
 s - t -Fluss 32
 beschränkt 32
 extrem 39
 maximaler s - t -Fluss 32

-G-

Gleichgewicht 87
 korreliert 88
 Wert 87
 Graph
 bipartit 24
 gerichtet 9
 ungerichtet 18
 zusammenhängend 18

-H-

Hülle

konisch 60
 konvex 46
 Halbraum 50
 Hilbertbasis 81
 ganzzahlig 81
 Hyperebene 50
 Stützhyperebene 50
 Trennhyperebene 50
 strikt 53

-I-

Inzidenzmatrix 74
 Inzidenzvektor 27

-K-

Kante
 gerichtet 9
 ungerichtet 18
 Kantenfolge 9, 18
 Kantenlänge 18
 Kapazitätsfunktion 32
 Kegel 59
 endlich erzeugt 60
 ordentlich 59
 spitz 59
 Knoten 9, 18
 Endknoten 9
 Startknoten 9
 Knotenüberdeckung 30
 Kombination
 konisch 59
 konvex 44
 Korollar
 max-Flow=min-Cut-Theorem, [Theorem 5.1.1](#) 75
 Ganzzahlige Version von Egerváry [Satz 4.3.2](#) 74
 Matchingtheorem von König, 1931 31
 Matchingtheorem von König, siehe [Korollar 4.3.3.a](#) 72
 Variante des Lemma von Farkas .. 62
 Korrelationsmatrix 88
 Kosten 39
 Kreis 18
 gerichtet 13

-L-

Länge 9, 18

Längenfunktion	9, 26
Lösung	
gültig	64
optimal	64
zulässig	64, 104
strikt	104
lineares Programm	43
dual	65
ganzzahlig	67
primal	65
unbeschränkt	64
ungültig	64
unzulässig	64

-M-

M -augmentierend	23
Matching	3, 23
extrem	25
nicht stabil	3
perfekt	28
stabil	3
Matchingpolytop	27
Matrix	
vollständig-unimodular	69
Menge	
konvex	44
unabhängig	24
Metrik	46
Euklidisch	46
Manhattan	46
Maximum	46
Minkowski-Metrik	46
Minkowski-Summe	61

-N-

Nachbar	18
Nachbarschaft	108
Netzwerk	32
Newton-Schritt	
modifiziert	108

-O-

Ordnung	
partiell	59

-P-

Pfad	
zentral	104
Polare	56
Polyeder	54
ganzzahlig	69
Seite	82

Polytop	47
ganzzahlig	68
Positivitätsbereich	59
Potential	15
Potenzialfunktion	15
Projektion	
metrisch	50

-Q-

Quelle	32
--------------	----

-R-

Raum	
Halb~	50
metrisch	46
Residualgraph	
Fluss	34
Matching	25

-S-

Satz	
min-max-Charakterisierung kürzester Wege, Robacker, 1956	10
Carathéodory	48
Erweiterung guter Wälder	21
geometrische Modellierung kürzester Wege	16
Geometrische Modellierung minimaler Spannbäume	22
Hauptachsentransformation/Spektralzerlegung	97
Hoffman, 1960	38
Hoffman, Kruskal, 1956	70
Lemma von Farkas	62
Minkowski-Weyl für Kegel	61
Optimalitätsbedingung	65
Satz von Egeváry	29
Schnitt	
s - t -Schnitt	10
Senke	32
Spannbaum	18
minimal	19
Strategie	
gemischt	86

-T-

Theorem	
max-Flow=min-Cut-Theorem, Ford-Fulkerson, 1954	34
Dualität linearer Programme, von Neumann, 1947	66
Extrema mit Nebenbedingungen	106

Grötschel, Lovász, Schrijver, 1981, basierend auf der
 Ellipsoidmethode 100
 Theorem von Birkhoff, 1946 28
 Theorem von Minkowski-Weyl 57

-V-

Verbindungsstrecke 45
 Vereinigung
 disjunkt 24
 vollständig dual ganzzahlig 80

-W-

Wald 18
 gut 21
 Weg 9, 18
 v_0 - v_m -Weg 9
 zusammenhängend 18
 Wert 32

-Z-

Zirkulation 37
 Zusammenhangskomponente 18, 22

C. Algorithmen

I. Operations Research

i. Kombinatorische Graphen-Algorithmen 2

1. Algorithmus 1.2.1 (Gale-Shapley-Algorithmus)	4
2. Algorithmus (Gale-Shapley-Algorithmus (Fortsetzung))	5
3. Algorithmus 2.1.1 (Dijkstra-Algorithmus)	11
4. Algorithmus 2.2.1 (Bellman-Ford-Algorithmus)	13
5. Algorithmus 3.1.1 (Algorithmus von Prim)	19
6. Algorithmus 3.1.2 (Algorithmus von Kruskal)	20
7. Algorithmus 4.1.1 (Bestimmung von Matchings mit maximaler Kardinalität)	24
8. Algorithmus 4.2.1 („Ungarische Methode“ zur Bestimmung gewichteter Matchings mit maximaler Kardinalität, 1931)	26
9. Algorithmus 5.1.1 (Ford-Fulkerson-Algorithmus zur Berechnung maximaler Flüsse)	35
10. Algorithmus 5.3.1 (zur Berechnung maximaler Flüsse mit minimalen Kosten)	41

ii. Theorie der linearen Optimierung 42

iii. Algorithmen der linearen Optimierung 91

11. Algorithmus 11.1.1 (Simplexverfahren)	95
12. Algorithmus 12.2.1 (Ellipsoidmethode)	100
13. Algorithmus 13.2.1 (Innere-Punkte-Methode)	109

D. Definitionen

I. Operations Research

i. Kombinatorische Graphen-Algorithmen 2

1. Definition 1.1.1 (Matching)	3
2. Definition 1.1.2 (stabiles und nicht stabiles Matching)	3
3. Definition 2.1.1 (Abstand)	10
4. Definition 2.1.2 (s - t -Schnitt)	10
5. Definition 2.2.1 (gerichteter Kreis)	13
6. Definition 2.3.1 (Potential)	15
7. Definition 3.1.1 (Minimaler Spannbaum)	19
8. Definition 3.1.2 (guter Wald)	21
9. Definition 3.1.3 (Zusammenhangskomponente)	22
10. Definition 4.1.1 (Matching)	23
11. Definition 4.1.2 (M -augmentierend)	23
12. Definition 4.1.3 (chromatische Zahl)	24
13. Definition 4.1.4 (bipartiter Graph)	24
14. Definition 4.1.5 (Residualgraph für Matchings)	25
15. Definition 4.2.1 (extremes Matching)	25
16. Definition 4.2.2 (Längenfunktion)	26
17. Definition 4.3.1 (Inzidenzvektor)	27
18. Definition 4.3.2 (Matchingpolytop)	27
19. Definition 4.3.3 (perfektes Matching)	28
20. Definition 4.3.4 (Knotenüberdeckung)	30
21. Definition 5.1.1 (s - t -Fluss)	32
22. Definition 5.1.2 (Wert)	32
23. Definition 5.1.3 (Kapazitätsfunktion)	32
24. Definition 5.1.4 (maximaler s - t -Fluss)	32
25. Definition 5.1.5 (Residualgraph für Flüsse)	34
26. Definition 5.2.1 (Zirkulation)	37
27. Definition 5.3.1 (Kosten)	39
28. Definition 5.3.2 (extremer Fluss)	39

ii. Theorie der linearen Optimierung 42

29. Definition 6.1.1 (lineares Programm)	43
30. Definition 6.2.1 (Konvexkombination)	44
31. Definition 6.2.2 (konvexe Menge)	44
32. Definition 6.2.3 (Metrik)	46
33. Definition 6.2.4 (Minkowski-Metrik)	46
34. Definition 6.2.5 (konvexe Hülle)	46
35. Definition 6.2.6 (Polytop)	47
36. Definition 6.2.7 (affin unabhängig)	47

37. Definition 6.2.8 (Dimension)	48
38. Definition 6.3.1 (Hyperebene und Halbraum)	50
39. Definition 6.3.2 (Trenn- und Stütz-Hyperebene)	50
40. Definition 6.3.3 (Polyeder)	54
41. Definition 6.4.1 (Extrempunkt)	54
42. Definition 6.5.1 (Polare)	56
43. Definition 6.5.2 (konische Kombination)	59
44. Definition 6.5.3 (konvexer, spitzer, ordentlicher Kegel)	59
45. Definition 6.5.4 (konische Hülle)	60
46. Definition 6.5.5 (endlich erzeugter Kegel)	60
47. Definition 6.7.1 (primales und duales lineares Programm)	65
48. Definition 7.1.1 (ganzzahliges lineares Programm)	67
49. Definition 7.1.2 (ganzzahliges Polytop)	68
50. Definition 7.2.1 (vollständig-unimodulare Matrix)	69
51. Definition 7.2.2 (ganzzahliger Polyeder)	69
52. Definition 7.4.1 (Inzidenzmatrix)	74
53. Definition 8.1.1 (ganzzahlige Hülle)	78
54. Definition 8.2.1 (Chvátal-Gomory-Abschluss)	79
55. Definition 8.2.2 (vollständig dual ganzzahliges System)	80
56. Definition 8.3.1 (Hilbertbasis)	81
57. Definition 8.3.2 (Polyederseite)	82
58. Definition 9.1.1 (gemischte Strategie)	86
59. Definition 9.1.2 (Gleichgewicht)	87
60. Definition 9.1.3 (korreliertes Gleichgewicht)	88

iii. Algorithmen der linearen Optimierung 91

61. Definition 12.1.1 (Ellipsoid)	97
62. Definition 13.1.1 (zentrale Pfad)	104
63. Definition 13.1.2 (Barrierefunktion)	105
64. Definition 13.1.3 (Dualitätsmaß)	108
65. Definition 13.1.4 (Nachbarschaft)	108
66. Definition 13.2.1 (modifizierter Newton-Schritt)	108

E. Theoreme

I. Operations Research

i. Kombinatorische Graphen-Algorithmen	2
1. Satz 1.2.1	6
1. Lemma 1.2.1	6
2. Satz 1.2.2	7
1. Korollar 1.2.2.a	7
3. Satz 1.3.1	8
4. Satz 2.1.1 (min-max-Charakterisierung kürzester Wege, Robacker, 1956) . .	10
5. Satz 2.1.2	11
6. Satz 2.2.1	13
7. Satz 2.2.2	14
8. Satz 2.2.3	14
9. Satz 2.3.1	15
10. Satz 2.3.2 (geometrische Modellierung kürzester Wege)	16
11. Satz 3.1.1 (Erweiterung guter Wälder)	21
2. Korollar 3.1.1.a	22
3. Korollar 3.1.1.b	22
12. Satz 3.1.2 (Geometrische Modellierung minimaler Spannbäume)	22
13. Satz 4.1.1	23
14. Satz 4.2.1	26
15. Satz 4.2.2	27
1. Theorem 4.3.1 (Theorem von Birkhoff, 1946)	28
16. Satz 4.3.1	29
17. Satz 4.3.2 (Satz von Egeváry)	29
18. Satz 4.3.3	30
4. Korollar 4.3.3.a (Matchingtheorem von König, 1931)	31
2. Lemma 5.1.1	33
3. Lemma 5.1.2	34
2. Theorem 5.1.1 (max-Flow=min-Cut-Theorem, Ford-Fulkerson, 1954)	34
5. Folgerung 5.1.1	35
19. Satz 5.1.1	35
4. Lemma 5.1.3	37
20. Satz 5.1.2	37
21. Satz 5.2.1 (Hoffman, 1960)	38
6. Korollar 5.2.1.a	38
22. Satz 5.2.2	39
23. Satz 5.3.1	40
24. Satz 5.3.2	41

ii. Theorie der linearen Optimierung 42

25. Satz 6.2.1	45
26. Satz 6.2.2	47
27. Satz 6.2.3 (Carathéodory)	48
5. Lemma 6.3.1	51
3. Theorem 6.3.1	52
7. Korollar 6.3.1.a	53
28. Satz 6.4.1	55
8. Korollar 6.4.1.a	56
6. Lemma 6.5.1	56
4. Theorem 6.5.1 (Theorem von Minkowski-Weyl)	57
29. Satz 6.5.1	60
30. Satz 6.5.2 (Minkowski-Weyl für Kegel)	61
5. Theorem 6.5.2	61
31. Satz 6.6.1	61
32. Satz 6.6.2 (Lemma von Farkas)	62
9. Korollar 6.6.2.a (Variante des Lemma von Farkas)	62
10. Korollar 6.6.2.b	63
33. Satz 6.7.1	64
7. Lemma 6.7.1	65
34. Satz 6.7.2 (Optimalitätsbedingung)	65
6. Theorem 6.7.1 (Dualität linearer Programme, von Neumann, 1947)	66
11. Korollar 6.7.2.a	66
35. Satz 7.2.1	69
36. Satz 7.2.2	70
12. Korollar 7.2.2.a	70
37. Satz 7.2.3 (Hoffman, Kruskal, 1956)	70
38. Satz 7.3.1	71
13. Korollar 7.3.1.a (Matchingtheorem von König, siehe Korollar 4.3.3.a)	72
14. Korollar 7.3.1.b	73
15. Korollar 7.3.1.c (Ganzzahlige Version von Egerváry Satz 4.3.2)	74
39. Satz 7.4.1	74
16. Folgerung 7.4.1	75
17. Korollar 7.4.1.a (max-Flow=min-Cut-Theorem, Theorem 5.1.1)	75
40. Satz 8.1.1	78
18. Korollar 8.1.1.a	78
7. Theorem 8.2.1	80
41. Satz 8.2.1	81
42. Satz 8.3.1	82
43. Satz 8.3.2	82
44. Satz 8.3.3	82
45. Satz 8.3.4	83

iii. Algorithmen der linearen Optimierung 91

46. Satz 11.1.1	95
47. Satz 12.1.1 (Hauptachsentransformation/Spektralzerlegung)	97

48. Satz 12.1.2	98
8. Lemma 12.1.1	98
8. Theorem 12.2.1 (Grötschel, Lovász, Schrijver, 1981, basierend auf der Ellipsoidmethode)	100
9. Lemma 12.2.1	100
10. Lemma 12.2.2	101
11. Lemma 13.1.1	104
12. Lemma 13.1.2	105
13. Lemma 13.1.3	105
14. Lemma 13.1.4	106
15. Lemma 13.1.5	106
9. Theorem 13.1.1 (Extrema mit Nebenbedingungen)	106
10. Theorem 13.1.2	106
16. Lemma 13.1.6	108
17. Lemma 13.2.1	109
18. Lemma 13.2.2	110
19. Lemma 13.2.3	111