Vorlesung Optimierung I und II
WS 1998/99 und SS 1999

Inhaltsverzeichnis

Literatur

[1] Bollobas, B.: Graph Theory, Springer, New York 1979.

[2] Burkhard, R.E.: Methoden der ganzzahligen Optimierung, Springer, Wien-New York 1972.

[3] Collatz, L., Wetterling, W.: Optimierungsaufgaben, Springer, Berlin-New York 1971.

[4] Dantzig, G.B.: Lineare Programmierung und Erweiterungen, Springer, Berlin-New York 1966.

[5] Franklin, J.N.: Methods of mathematical economics : linear and nonlinear programming; fixed point theorems, Springer, New York, 1980.

[6] Gaede, K.W., Heinhold,J.: Grundzüge des Operations Research, Carl-Hanser-Verlag, München-Wien 1976.

[7] Gass, S.I.: Linear Programming, Methods and Applications, Mac Graw-Hill, New York 1958.

[8] Grötschel, M., Lovasz, L., Schrijver, A.: Geometric algorithms and Combinatorial Optimization, Springer, Berlin-New York 1988.

[9] Hadley, G.: Linear Programming, Addison Wesley Publ. Comp., London 1972.

[10] Hu, T.Chiang: Ganzzahlige Programmierung und Netzwerkflüsse, Oldenburg, München 1972.

[11] Kall, P.: Mathematische Methoden des Operations Research, Teubner, Stuttgart 1976.

[12] Marti, J.: Konvexe Analysis, Birkhäuser, Basel 1977.

[13] Neumann, K.: Operations-Research-Verfahren I-III, Carl-Hanser-Verlag, München-Wien 1975.

[14] Nozicka, F.: Geometrie konvexer Mengen und konvexe Analysis, Akadem. Verl., Berlin 1988.

[15] Owen, G.: Spieltheorie, Springer, Berlin-New York 1971.

[16] Papadimitriou, C.H., Steiglitz, K.: Combinatorial Optimization, Prentice-Hall Inc., New Jerssey 1982.

[17] Rauhut, B., Schmitz, N., Zachow, E.-W.: Spieltheorie, Teubner, Stuttgart 1979.

[18] Roberts, A.W., Verberg, D.E.: Convex Functions, Academic Press, New York 1973.

[19] Rockafellar, R.: Convex Analysis, Princeton University Press, Princeton 1972.

[20] Schrijver, A.: Theory of Linear and Integer Programming, Wiley, Chichester 1986.

[21] Stoer, Witzgall: Convexity and Optimization in finite dimensions, Springer 1970.

[22] Suchowitzki, S.I., Andejewa, L.I.: Lineare und konvexe Optimierung, Oldenburg, München 1969.

[23] Szép, J., Foró, F.: Einführung in die Spieltheorie, Harri Deutsch, Thun, Frankfurt, 1983.

[24] Turau, V.: Algorithmische Graphentheorie, Addison-Wesley, 1996.

[25] van Tiel, J.: Convex Analysis, Wiley, Chichester 1984.

[26] Valentine, F.A.: Konvexe Mengen, Bibliograph. Inst., Mannheim 1968.

[27] Vogel, W.: Lineares Optimieren, Akad. Verlagsges., Leipzig 1967.

[28] Vorob'ev, N.N.: Game Theory, Springer, New York 1977.

Index

1 Einführung

1.1 Allgemeines und Historie

Probleme der linearen und kombinatorischen Optimierung sind in den 40er Jahren aus Anforderungen der Wirtschaft und militärischen Logistik im stärkeren Umfang behandelt worden. Die ersten Beispiele reichen weit in die klassische Mathematik zurück. Als eigenständige Disziplin wird das Optimieren mit Georg Dantzigs Entdeckung des Simplex–Algorithmus 1947 geschaffen. Etwa gleichzeitig entwickelte von Neumann das Dualitätsprinzip. Weitere wichtige Namen der Gründerzeit sind Motzkin, Leontief, Kantorowicz, Gale, Kuhn, Tucker, Gomory, Klee, … . Die schnelle Entwicklung des Gebietes liegt an der besonders engen Verflechtung von Theorie und Anwendung, vor allem aber direkter und vielfältiger Anwendung. Mathematische Verbesserungen, die effizientere Verfahren versprachen, gingen durch die Weltpresse, so z. B. Khachians Ellipsoid–Methode (1979) oder der Algorithmus von Karmarkar (1984). Optimierungsprobleme gab es schon in der Geometrie der alten Griechen, vor allem aber in der mathematischen Physik (Prinzipien von Newton, Euler, Lagrange, d'Alembert, Gauß, etc.). Aber diese Methoden der Analysis paßten nicht auf die Probleme der Ökonomie. Das führte auf die Methoden der linearen, konvexen, diskreten und kombinatorischen Optimierung, deren mathematische Grundlagen vor allem aus der Linearen Algebra und der Konvexgeometrie, aber auch der Kombinatorik, Graphentheorie und Geometrie der Zahlen kommen. Insofern ist das Optimieren eines der wichtigsten Anwendungsgebiete der genannten mathematischen Disziplinen. Außerdem ist für eine effektive und kostengünstige Anwendung der verschiedenen Methoden die Frage nach schnellen und effektiven Algorithmen entscheidend, d. h. die Verbindung zur Informatik, Computational Geometry und Computer Algebra.

1.2 Problemstellung

Das allgemeine Optimierungsproblem läßt sich folgendermaßen beschreiben. Sei F eine Menge, M eine geordnete Menge (z. B. $\mathbb{R}, \mathbb{Z}$) und $c : F \to M$ eine Abbildung. Gesucht wird ein $x_0 \in F$ mit $c(x_0) \leq c(x)$ für alle $x \in F$.
In dieser Allgemeinheit ist das Problem praktisch unangreifbar. Darum wendet man sich bestimmten Klassen von Problemen zu.

Definition 1.1 *Ein Beispiel einer* Optimierungsaufgabe *ist ein Paar* (F, c) *mit* F Menge der zulässigen Punkte $c : F \to M$ Kostenfunktion. *Gesucht ist* $x \in F$ *mit*

$$c(x) \leq c(y), \text{ für alle } y \in F.$$

x heißt global optimale Lösung *des gegebenen Beispiels oder, wenn keine Verwechslungen zu befürchten sind, einfach optimale Lösung.*

Definition 1.2 *Ein* Optimierungsproblem *ist eine Menge von Beispielen eines Optimierungsproblems.*

Ein wichtiger Spezialfall ist $F \subset \mathbb{R}^n, \quad M = \mathbb{R}$, wobei F in der Form beschrieben werden kann $F = \{x \in \mathbb{R}^n \,|\, g_i(x) \geq 0, i = 1, \ldots, m, h_j(x) = 0, j = 1, \ldots, p\}$.

Mit diesem Problem befaßt sich die *nichtlineare Optimierung*, die unter passenden Voraussetzungen an c, die g_i und h_j Bedingungen für ein Minimum angibt, sowie iterative Algorithmen, die es gestatten, das Minimum zu berechnen. Sind c und alle g_i, h_j konvex (linear) so sprechen wir von *konvexer (linearer) Optimierung.*

Beispiel 1.1 (Lineare Optimierungsaufgabe, Linear programming, LP) *Seien* $m, n \in \mathbb{N}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ *und* $A \in \mathbb{R}^{m \times n}$. *Ein Beispiel von LP ist definiert durch*

$$F = \{x \,|\, x \in \mathbb{R}^n, Ax = b, x \geq 0\}, \quad c : x \mapsto c^\top x.$$

Ist die Menge F endlich, so sprechen wir von *kombinatorischer Optimierung.* Das Problem ist im Prinzip trivial in dem Sinne, daß wir als Lösungsmöglichkeit haben, alle Punkte aus F zu untersuchen und dann den passenden auszuwählen. Nur hat F im allgemeinen so viele Punkte, daß man sich um effizientere Algorithmen bemühen muß. Dabei stellt sich natürlich das Problem zu sagen, was ein effizienter Algorithmus ist. Auf dieses Problem werden wir später zurückkommen.

Beispiel 1.2 (Traveling SalesmanProblem (TSP)) *In einem Beispiel des TSP ist der Abstand zwischen jedem Paar von n Städten gegeben. Gesucht ist eine Rundreise minimaler Gesamtlänge. F ist hier die Menge von zyklischen Permutationen. Die Kostenfunktion c ist durch eine $n \times n$–Matrix (a_{ij}) bestimmt mit* $c(i_1, \dots, i_n) = a_{i_1 i_2} + a_{i_2 i_3} + \dots + a_{i_{n-1} i_n} + a_{i_n i_1}$.

Prinzipiell ist LP ein stetiges Optimierungsproblem, es kann aber auch als kombinatorisches Problem angesehen werden. Dazu zunächst nur ein Beispiel, gegeben durch $m = 1, n = 3$ und $A = (1, 1, 1)$, $b = (2)$. F ist der Durchschnitt einer Ebene mit dem ersten Oktanten des $\mathbb{R}^3$, also ein Dreieck. Gesucht wird das Minimum von $c^\top x$. Dies wird offensichtlich stets an einer Ecke angenommen. Damit genügt die Bestimmung von $c^T x$ für jede Ecke. Für größere m, n kann man entsprechend vorgehen. Die Aufgabe wird dann schnell sehr groß, bleibt aber endlich.

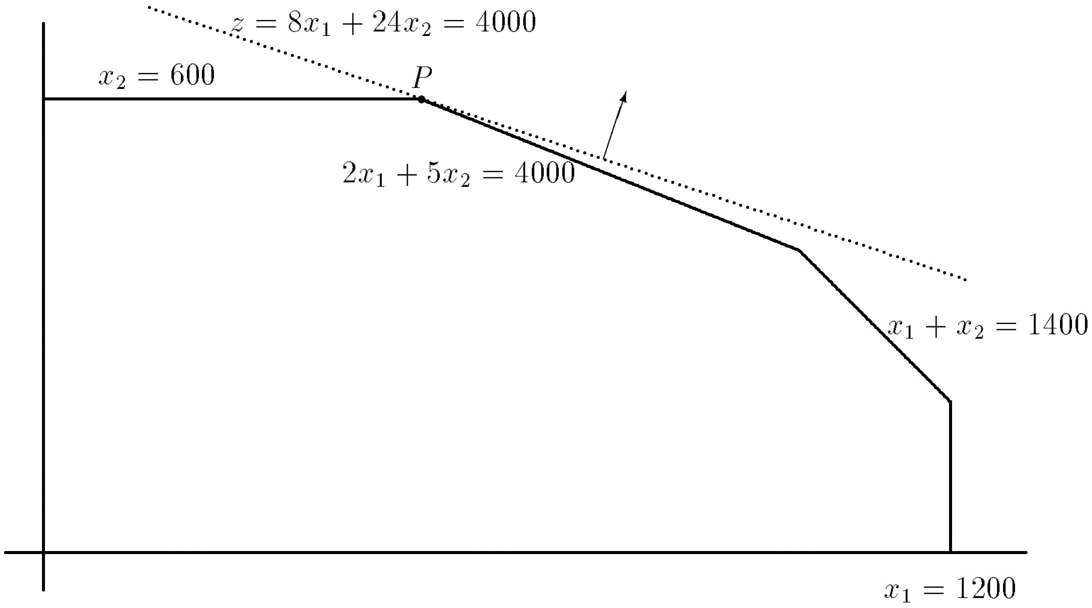
1.3 Anwendungsbeispiele

1.3.1 Ein Produktionsproblem

In einem Textilbetrieb werden 2 Tuche gewebt und anschließend gefärbt.

	Aufwand in Stunden		Gewinn	Kapazitäten an
	Weben	Färben	pro Einheit	Tucheinheiten
Tuch 1	2	1	8	1200
Tuch 2	5	1	24	600
Kapazität in Std	4000	1400	–	–

Sei x_i die Einheiten an Tuch i, $i = 1, 2$. Dann ergeben sich die Restriktionen
$x_1, x_2 \geq 0,\ x_1 \leq 1200,\ x_2 \leq 600,\ 2x_1 + 5x_2 \leq 4000,\ x_1 + x_2 \leq 1400.$
Die Zielfunktion $z = 8x_1 + 24x_2$ ist zu maximieren.
Da nur zwei Variablen vorhanden sind, ist eine graphische Darstellung möglich.



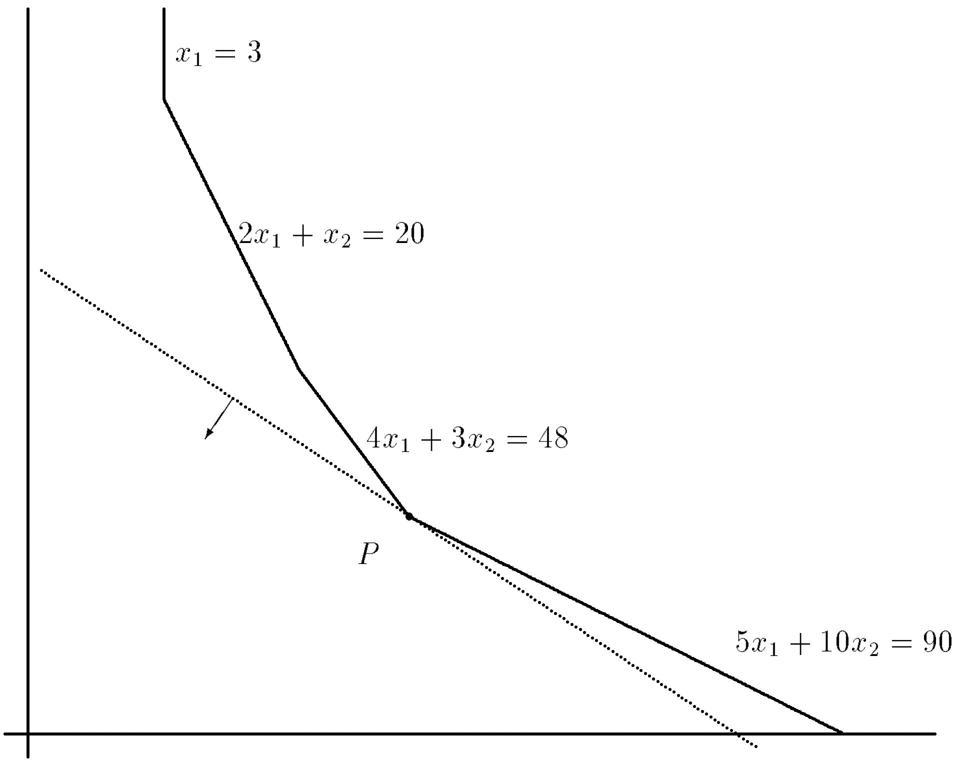
Der Bereich der zulässigen (feasible) Vektoren (Punkte, Lösungen) ist ein konvexes Sechseck. z = const ist eine Parallelenschar, der z–Wert wächst in $\rightarrow$ Richtung. Die optimale Lösung ist $P = (500, 600)$. Der maximale Gewinn ist $8 \cdot 500 + 24 \cdot 600 = 18400$. Die Kapazität der Weberei (4000) wird voll ausgenutzt, ebenso das Material (600) für Tuch 2. Dagegen wird die Kapazität der Weberei nicht voll ausgenutzt, ebenso das Material für Tuch 1. Ein weiteres Problem bestände darin, ungenutzte Kapazitäten (Materialien) zu minimieren.

1.3.2 Ein Fütterungsproblem

In einer Hühnerfarm werden 2 Futtersorten F_1, F_2 mit 4 Nährstoffen $N_1, \ldots, N_4$ verwendet. Folgende Daten liegen zugrunde.

	Gehalt in Einheiten		Bedarf pro Huhn
	F_1	F_2	und Zeiteinheit
N_1	5	10	90
N_2	4	3	48
N_3	0.5	0	1.5
N_4	2	1	20
Kosten pro Einheit	2	3	

Ist x_i die Anzahl der Einheiten von Futter F_i, $i = 1, 2$, die ein Huhn pro Zeiteinheit bekommt, so ergeben sich die Restriktionen
$5x_1 + 10x_2 \geq 90, \quad 4x_1 + 3x_2 \geq 48, \quad 0.5x_1 \geq 1.5, \quad 2x_1 + x_2 \geq 20, \quad x_1, x_2 \geq 0.$
Die Zielfunktion $z = 2x_1 + 3x_2$ (die Kosten) ist zu minimieren.



Der Bereich der zulässigen Lösungen ist unbeschränkt (polyedrisch). Aber es existiert ein Minimum, da z in $\rightarrow$ Richtung abnimmt. Die optimale Lösung ist $P = (8.4, 4.8)$ mit minimalen Kosten $z = 2 \cdot 8.4 + 3 \cdot 4.8 = 31.2$. Man beachte, daß nur N_1 und N_2 voll ausgenutzt werden.

1.3.3 Ein Transportproblem

Es soll Ware von Lagerhäusern L_1, L_2 zu Verteilerstationen $V_1, \ldots, V_5$ transportiert werden. Dabei sind Vorrat, Bedarf und Transportkosten gegeben (Transportkosten ∞ heißt, daß ein Transport nicht möglich ist).

	V_1	V_2	V_3	V_4	V_5	Vorrat in L_i
L_1	3	∞	5	∞	4	10
L_2	2	4	∞	6	5	12
Bedarf von V_i	4	3	4	5	6	$\rightarrow \sum \stackrel{!}{=} 22$

Ist x_{ij} die transportierte Menge vom Lager L_i zum Verteiler V_j, so ergeben sich die Restriktionen
$x_{11} + x_{13} + x_{15} \leq 10, \quad x_{21} + x_{22} + x_{24} + x_{25} \leq 12$
$x_{11} + x_{21} \geq 4, x_{22} \geq 3, x_{13} \geq 4, x_{24} \geq 5, x_{15} + x_{25} \geq 6$ und $x_{ij} \geq 0$, für alle i, j.
Die Zielfunktion
$z = 3x_{11} + 5x_{13} + 4x_{15} + 2x_{21} + 4x_{22} + 6x_{24} + 5x_{25}$
ist zu minimieren. Da der Gesamtvorrat gleich dem Gesamtbedarf (=22) ist, kann jeder Verteiler nur genau seinen Bedarf erhalten und jedes Lager muß seine Kapazität voll ausschöpfen,

d.h. alle Ungleichungen gelten mit Gleichheit. Insbesondere $x_{22} = 3, x_{13} = 4, x_{24} = 5$. Der zugehörige Kostenanteil ist $z_0 = 12 + 20 + 30 = 62$. Daher ergibt sich ein reduziertes Problem:

$z = 3x_{11} + 4x_{15} + 2x_{21} + 5x_{25} + 62 \to \min$ unter

$x_{11} + x_{15} = 6, x_{21} + x_{25} = 4,\ x_{11} + x_{21} = 4, x_{15} + x_{25} = 6$ und $x_{ij} \geq 0$.

Hier ist wegen der 4 Variablen keine graphische Lösung möglich. Jedoch können weitere Variable eliminiert werden: $x_{15} = 6 - x_{11},\ x_{25} = x_{11},\ x_{21} = 4 - x_{11}$. Wegen $0 \leq x_{11} \leq 4$ ist der zulässige Bereich also eine Strecke. Das Minimum von $z = 3x_{11} + 4(6 - x_{11}) + 2(4 - x_{11}) + 5x_{11} + 62 = 2x_{11} + 94$ wird offensichtlich für $x_{11} = 0$ angenommen mit $z_{\text{min}} = 94$.

2 Lineare Optimierung

2.1 Der Fundamentalsatz über lineare Ungleichungen

Definition 2.1 *Seien $a^1, \ldots, a^m, b \in \mathbb{R}^n$. Dann heißt b* Linearkombination *von $a^1, \ldots, a^m$, falls es $\lambda_1, \ldots, \lambda_m \in \mathbb{R}$ gibt, mit $b = \lambda_1 a^1 + \ldots + \lambda_m a^m$.*
Die Linearkombination heißt nichtnegativ*, falls $\lambda_i \geq 0, \quad i = 1, \ldots, m$.*

Bezeichnung: (1) Für $X \subset \mathbb{R}^n$ bezeichnet $\mathrm{lin}X$ die lineare Hülle von X.
(2) Für $a^1, \ldots, a^m \in \mathbb{R}^n$ bezeichnet $\mathrm{rank}\{a^1, \ldots, a^m\} = \dim \mathrm{lin}\{a^1, \ldots, a^m\}$ den Rang, d.h. die Dimension des von $a^1, \ldots, a^m$ aufgespannten Unterraums.

Satz 2.1 *Seien $a^1, \ldots, a^m, b \in \mathbb{R}^n, \quad t = \mathrm{rank}\{a^1, \ldots, a^m, b\}$. Dann gilt entweder*

1. *b ist eine nichtnegative Linearkombination linear unabhängiger Vektoren aus $\{a^1, \ldots, a^m\}$.*

   oder

2. *Es existiert eine Hyperebene $\{x \mid c^\top x = 0\}$, die $t-1$ linear unabhängige Vektoren aus $\{a^1, \ldots, a^m\}$ enthält und $c^\top b < 0$ und $c^\top a^1, \ldots, c^\top a^m \geq 0$ genügt.*

Beweis: Sei $L = \mathrm{lin}(a^1, \ldots, a^m)$. Gilt $b \notin L$, so gilt sicher (2) und nicht (1). Für $b \in L$ genügt es die Behauptung für L statt $\mathbb{R}^n$ zu beweisen (sonst fasse L als $\mathbb{R}^{n'}$ auf mit $n' = \dim L$). Sei daher o.B.d.A. $L = \mathbb{R}^n$.
Die beiden Fälle (1), (2) schließen sich aus, denn aus $b = \lambda_1 a^1 + \ldots + \lambda_m a^m$ mit $\lambda_i \geq 0$ folgt

$$c^\top b = \lambda_1 c^\top a^1 + \ldots + \lambda_m c^\top a^m \geq 0.$$

Zeige: Es gilt (1) oder (2). Dazu sei $D = \{a^{i_1}, \ldots, a^{i_n}\}$ eine Basis des $\mathbb{R}^n$. Jetzt wende man folgende Iteration an:

i. Schreibe $b = \lambda_{i_1} a^{i_1} + \ldots + \lambda_{i_n} a^{i_n}$. Gilt $\lambda_{i_1}, \ldots, \lambda_{i_n} \geq 0$, so gilt Fall (1).

ii. Wähle unter $i_1, \ldots, i_n$ den kleinsten Index h mit $\lambda_h < 0$. Sei $\{x \mid c^\top x = 0\}$ die von $D \setminus \{a^h\}$ aufgespannte Hyperebene. Es sei o.B.d.A. $c^\top a^h = 1$ und daher $c^\top b = \lambda_h$.

iii. Gilt $c^\top a^1, \ldots, c^\top a^m \geq 0$, so gilt Fall (2).

iv. Ansonsten wähle das kleinste s mit $c^\top a^s < 0$. Dann ersetze man D durch $(D \setminus \{a^h\}) \cup \{a^s\}$ und setze mit (i) fort.

Es bleibt zu zeigen, daß die Iteration abbricht. Sei D_k die Basis der k-ten Iteration. Da nur endlich viele Basen existieren, hat man im Fall des Nichtabbruchs $D_k = D_l$, für $k < l$. Sei r der größte Index, für den a^r aus D in den Iterationen $k, \ldots, l-1$ eliminiert wird. Dies sei die Iteration p. Wegen $D_k = D_l$ ist a^r für eine Iteration q mit $k \leq q < l$ zu D hinzugefügt worden. Nach Definition von r gilt:

$$D_p \cap \{a^{r+1}, \ldots, a^m\} = D_q \cap \{a^{r+1}, \ldots, a^m\}. \tag{2.1}$$

Sei jetzt $D_p = \{a^{i_1}, \dots, a^{i_n}\}$, $\quad b = \lambda_{i_1} a^{i_1} + \ldots + \lambda_{i_n} a^{i_n}$ und sei c_q der Vektor c aus der Iteration q, dann gilt

$$0 > {c_q}^\top b = {c_q}^\top (\lambda_{i_1} a^{i_1} + \ldots + \lambda_{i_n} a^{i_n}) = \lambda_{i_1} {c_q}^\top a^{i_1} + \ldots + \lambda_{i_n} {c_q}^\top a^{i_n} > 0.$$

Dabei gilt die erste Ungleichung nach (ii) aus der Iteration q. Die zweite Ungleichung ergibt sich aus

a) Für $i_j < r$ gilt nach (ii), Iteration p, $\lambda_{i_j} \geq 0$ und nach (iv), Iteration q, $\quad {c_q}^\top a^{i_j} \geq 0$.

b) Für $i_j = r$ gilt nach (ii), Iteration p, $\lambda_{i_j} < 0$ und nach (iv), Iteration q, $\quad {c_q}^\top a^{i_j} < 0$.

c) Für $i_j > r$ gilt ${c_q}^\top a^{i_j} = 0$ nach (ii) und Gleichung (2.1).

□

Bemerkung: Der Beweis ist konstruktiv und liefert einen Algorithmus zur Bestimmung der Fälle. Dies wird dem Simplexalgorithmus mit der Blandschen Antizykelregel entsprechen.

Beispiel 2.1 *Seien* $a^1 = (1,0,0)^\top$, $a^2 = (0,1,0)^\top$, $a^3 = (0,0,1)^\top$, $a^4 = (1,1,-1)^\top$, $b = (-1,-1,-1)^\top$. *b ist keine nichtnegative Linearkombination von a^1, a^2, a^3, a^4, da $c = (1,0,0)^\top$ der Bedingung (2) genügt.*

2.2 Konvexe Kegel

Definition 2.2 *$K \subset \mathbb{R}^n$ heißt* konvex, *falls für $x, y \in K$ und $\lambda \in [0,1]$ stets $\lambda x + (1-\lambda) y \in K$ gilt (d.h. mit je zwei Punkten ist auch die Verbindungsstrecke in K enthalten).*

Definition 2.3 *$C \subset \mathbb{R}^n$ heißt* (konvexer) Kegel, *falls für $x, y \in C$ und $\lambda, \mu \geq 0$ stets $\lambda x + \mu y \in C$ gilt.*
Ein Kegel C heißt endlich erzeugt, *falls $x^1, \dots, x^m \in \mathbb{R}^n$ existieren, so daß*

$$C = \operatorname{pos}\{x^1, \dots, x^m\} = \{\lambda_1 x^1 + \ldots + \lambda_m x^m \mid \lambda_1, \dots, \lambda_m \geq 0\}.$$

Ein Kegel heißt polyedrisch, *falls $C = \{x \mid Ax \leq 0\}$, für eine Matrix A gilt.*

Bemerkung: Eine Ungleichung zwischen Vektoren wird stets komponentenweise verstanden. Bezeichnet man eine Menge $\{x \mid a^\top x \leq 0\}$ mit $a \in \mathbb{R}^n$, $\quad a \neq 0$, als *Halbraum*, so ist ein polyedrischer Kegel der Durchschnitt endlich vieler Halbräume.

Definition 2.4 *Sei $M \subset \mathbb{R}^n$. Die zu M* polare Menge *$M^\star$ ist definiert durch*

$$M^\star = \{y \in \mathbb{R}^n \mid y^\top x \leq 0 \textit{ für alle } x \in M\}.$$

Satz 2.2 *Ist C ein Kegel, so ist $C^\star$ ebenfalls ein Kegel.*

Beweis: Übung.

□

Satz 2.3 *1. Ist C der von $x^1, \ldots, x^m$ erzeugte Kegel, so ist $C^\star$ der polyedrische Kegel $C^\star = \{y \in \mathbb{R}^n \mid y^\top x^i \leq 0,\ i = 1, \ldots, m\}$.*

2. Ist C der durch $C = \{y \in \mathbb{R}^n \mid y^\top x^i \leq 0,\ i = 1, \ldots, m\}$ gegebene Kegel, so ist $C^\star$ der von $x^1, \ldots, x^m$ erzeugte Kegel.

Bemerkung: Insbesondere gilt $(C^\star)^\star = C$, wenn C endlich erzeugt oder polyedrisch ist.

Beweis: (1) Sei $D = \{y \in \mathbb{R}^n \mid y^\top x^i \leq 0,\ i = 1, \ldots, m\}$. Sei $y \in D,\ x = \lambda_1 x^1 + \ldots + \lambda_m x^m \in C$. Es folgt $y^\top x = \lambda_1 y^\top x^1 + \ldots + \lambda_m y^\top x^m \leq 0$, also $D \subset C^\star$. Sei $y \in C^\star$. Dann gilt $y^\top x^i \leq 0,\ i = 1, \ldots, m$, also $y \in D$.
(2) Sei $D = \mathrm{pos}\{x^1, \ldots, x^m\}$. Sei $x = \lambda_1 x^1 + \ldots + \lambda_m x^m \in D,\ y \in C$. Es folgt $y^\top x = \lambda_1 y^\top x^1 + \ldots + \lambda_m y^\top x^m \leq 0$, also $D \subset C^\star$. Sei jetzt $b \notin D$. Nach Satz 2.1 existiert ein c mit $c^\top b < 0,\ c^\top x^i \geq 0,\ i = 1, \ldots, m$. Also gilt $-c \in C$ und damit $b \notin C^\star$. □

Satz 2.4 *Ein konvexer Kegel ist genau dann polyedrisch, wenn er endlich erzeugt ist.*

Beweis: a) Seien $x^1, \ldots, x^m \in \mathbb{R}^n$. $C = \mathrm{pos}\{x^1, \ldots, x^m\}$ ist polyedrisch: Sei zunächst $\mathbb{R}^n = \mathrm{lin}\{x^1, \ldots, x^m\}$. Zu einer Hyperebenen $H_c = \{x : c^\top x = 0\}$ sei $H_c^+ = \{x : c^\top x \geq 0\}$ einer der beiden zugehörigen Halbräume. Setze

$$Q := \bigcap_{x^1, \ldots, x^m \in H_c^+,\ H_c = \mathrm{lin}(x^{i_1}, \ldots, x^{i_{n-1}})} H_c^+.$$

Q ist sicher polyedrischer Kegel. Bleibt $C = Q$ zu zeigen. Es gilt $b \notin Q$, genau dann, wenn $\exists H_c$ mit $x^1, \ldots, x^m \in H_c^+,\ H_c = \mathrm{lin}(x^{i_1}, \ldots, x^{i_{n-1}})$ und $b \notin H_c^+$ bzw. $\exists c$ mit $c^\top x^i \geq 0, i = 1, \ldots, m$, $c^\top x^{i_1} = \ldots = c^\top x^{i_{n-1}} = 0$ und $c^\top b < 0$. Dies ist genau dann der Fall, wenn (2) in Satz 2.1, also nicht (1) in Satz 2.1 gilt. D.h. genau dann, wenn $b \notin C$.
Sei jetzt $L = \mathrm{lin}\{x^1, \ldots, x^m\}$ unterdimensional: Zu jedem Halbraum $H \subset L$ existiert ein Halbraum $H' \subset \mathbb{R}^n$ mit $H = H' \cap L$ und L ist selbst Durchschnitt von endlich vielen Halbräumen. Nach dem vorhergehenden gilt die Behauptung im Raum L und damit auch in $\mathbb{R}^n$.
b) Sei $C = \{x \mid a^{1\top} x \leq 0, \ldots, a^{m\top} x \leq 0\}$ ein polyedrischer Kegel. Nach Satz 2.3 gilt $C^\star = \mathrm{pos}\{a^1, \ldots, a^m\}$. Nach a) ist $C^\star = \{y \mid y^\top b^i \leq 0,\ i = 1, \ldots, t\}$. Nach Satz 2.3 gilt $C^{\star\star} = C$ und $C^{\star\star}$ ist ein endlich erzeugter Kegel. □

2.3 Das Lemma von Farkas und der Dualitätsssatz

Eine Formulierung des Fundamentalsatzes in der Sprache der linearen Algebra gibt

Satz 2.5 (Lemma von Farkas) *Sei $A \in \mathbb{R}^{n \times m}$, $\quad b \in \mathbb{R}^n$. Dann existiert ein $x \in \mathbb{R}^m$ mit $x \geq 0, \quad Ax = b$ genau dann, falls für jedes $y \in \mathbb{R}^n$ mit $y^\top A \geq 0$ gilt $\quad y^\top b \geq 0$.*

Beweis: Die Notwendigkeit der Bedingung ist klar, da für alle x, y mit $x \geq 0, \quad y^\top A \geq 0$ und $Ax = b$ sofort $y^\top b = y^\top A x \geq 0$ folgt.
Die Bedingung ist hinreichend: Es gebe kein $x \geq 0$ mit $Ax = b$. Seien $a^1, \ldots, a^m$ die Spalten von A. Dann gilt $b \notin \mathrm{pos}\{a^1, \ldots, a^m\}$ und daher existiert nach Satz 2.1 ein $y \in \mathbb{R}^n$ mit $y^\top b < 0$ und $y^\top A \geq 0$. □

Ein LP kann in verschiedenen Varianten vorliegen. Z.B. lieferten die Kapazitätsbedingungen in den Anwendungsbeispielen Restriktionen in der Form von Ungleichungen. Diese können aber durch die Einführung neuer (künstlicher) Variablen in Gleichungen überführt werden:
Die Bedingung $Ax \leq b$ bedeutet $A^{i\top} x \leq b_i,\ i = 1, \ldots, n$, falls A^i die Zeilen von A sind. Substituiere

$$x_{m+i} := b_i - A^{i\top} x,\ i = 1, \ldots, n.$$

Diese x_{m+i} werden *Schlupfvariable* genannt. Sie bilden zusammen einen Vektor $\tilde{x} \geq 0$. Mit dieser Substitution ist $Ax \leq b,\ x \in \mathbb{R}^m$ äquivalent zu $(A \,|\, I)\binom{x}{\tilde{x}} = b$ *und* $\tilde{x} \geq 0$.

Die entsprechende Variante des Lemmas von Farkas zu der Form $\{x \,|\, Ax \leq b\}$ ist:

Satz 2.6 *Sei* $A \in \mathbb{R}^{n\times m}, \quad b \in \mathbb{R}^n$. *Dann existiert ein* $x \in \mathbb{R}^m$ *mit* $Ax \leq b$ *genau dann, wenn* $y^\top b \geq 0$ *für jedes* $y \geq 0$ *und* $y^\top A = 0$ *gilt.*

Beweis: Durch die Einführung von Schlupfvariablen ist die Existenz eines $x \in \mathbb{R}^m$ mit $Ax \leq b$ äquivalent zu der Existenz eines $(x, \tilde{x}) \in \mathbb{R}^{m+n}$ mit $(A \,|\, I)\binom{x}{\tilde{x}} = b,\ \tilde{x} \geq 0$. Dies ist äquivalent zur Lösbarkeit von $(A \,|\, -A \,|I)(x', x'', \tilde{x})^\top = b, \quad x', x'' \in \mathbb{R}^m, \tilde{x} \in \mathbb{R}^n\ x', x'', \tilde{x} \geq 0$ (Augabe), d.h. zu $\exists z \in \mathbb{R}^{2m+n}, z \geq 0$ und $(A| - A|I)z = b$.
Nach Satz 2.5 hat dies genau dann eine Lösung, falls $y^\top b \geq 0$, für jedes y mit $y^\top A \geq 0, \quad y^\top(-A) \geq 0, \quad y^\top I \geq 0$ gilt. $\square$

Definition 2.5 *(1) Ein Menge* $P \subset \mathbb{R}^n$ *der Form* $P = \{x \,|\, Ax \leq b\}$ *mit* $A \in \mathbb{R}^{m\times n}, \quad b \in \mathbb{R}^m$ *heißt* Polyeder.
(2) Ein beschränktes Polyeder heißt Polytop.

Die *lineare Optimierung, linear programming, LP* beschäftigt sich mit der Minimierung bzw. Maximierung linearer Funktionen auf einem Polyeder. Beispiele sind

$$\max\{c^\top x \,|\, Ax \leq b\} \qquad \min\{c^\top x \,|\, x \geq 0,\ Ax \leq b\}.$$

Ein erster Satz ist

Satz 2.7 (Dualitätssatz der linearen Optimierung) *Sei* $A \in \mathbb{R}^{n\times m}, \quad b \in \mathbb{R}^n, \quad c \in \mathbb{R}^m$. *Dann gilt*

$$\max\{c^\top x \,|\, x \in \mathbb{R}^m,\ Ax \leq b\} = \min\{y^\top b \,|\, y \in \mathbb{R}^n,\ y \geq 0,\ y^\top A = c^\top\}, \tag{2.2}$$

unter der Voraussetzung, daß beide Mengen in (2.2) nichtleer sind.

Beweis: Gilt $Ax \leq b, \quad y \geq 0$ und $y^\top A = c^\top$, so gilt $c^\top x = y^\top Ax \leq y^\top b$. Damit gilt sicher stets $\max \leq \min$. Es bleibt die Existenz von x, y mit $Ax \leq b, \quad y \geq 0, \quad y^\top A = c^\top, \quad c^\top x \geq y^\top b$ zu zeigen, d.h. die Existenz von x, y mit

$$\begin{pmatrix} A & 0 \\ -c^\top & b^\top \\ 0 & A^\top \\ 0 & -A^\top \\ 0 & -I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \leq \begin{pmatrix} b \\ 0 \\ c \\ -c \\ 0 \end{pmatrix}. \tag{2.3}$$

Nach Satz 2.6 ist dies äquivalent zu: Sind $u, \lambda, v, w, z \geq 0$ mit $u^\top A - \lambda c^\top = 0$ und $\lambda b^\top + v^\top A^\top - w^\top A^\top - z^\top = 0$, dann gilt $u^\top b + v^\top c - w^\top c \geq 0$.
Um dies zu zeigen, mögen u, λ, v, w, z den Voraussetzungen genügen. Gilt $\lambda > 0$ so folgt

$$\begin{aligned} u^\top b &= \lambda^{-1}\lambda b^\top u = \lambda^{-1}((w^\top - v^\top)A^\top + z^\top)u \\ &\geq \lambda^{-1}\lambda(w^\top - v^\top)c. \end{aligned}$$

Gilt $\lambda = 0$, so seien x^0, y^0 so gewählt, daß $Ax^0 \leq b$ und $y^0 \geq 0, \quad y^{0\top} A = c^\top$ gilt (nach Voraussetzung möglich). Es folgt

$$u^\top b \geq u^\top A x^0 = 0 = ((w^\top - v^\top)A^\top + z^\top)y^0 \geq (w^\top - v^\top)c.$$

$\square$

Beispiel 2.2 *Sei*

$$A = \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{pmatrix}, \; b = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \; c = \begin{pmatrix} 2 \\ 1 \end{pmatrix},$$

dann zeigen $x^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \; y^0 = \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix}$

$$2 = \max\{c^\top x \,|\, Ax \leq b\} = \min\{y^\top b \,|\, y \geq 0, \; y^\top A = c^\top\}.$$

Setzt man wie üblich $\inf \emptyset = +\infty$, $\sup \emptyset = -\infty$, so läßt sich der Dualitätssatz noch etwas verschärfen zu:

Satz 2.8 *Sei* $A \in \mathbb{R}^{n \times m}, \quad b \in \mathbb{R}^n, \quad c \in \mathbb{R}^m$. *Dann gilt*

$$\sup\{c^\top x \,|\, Ax \leq b\} = \inf\{y^\top b \,|\, y \geq 0, \; y^\top A = c^\top\} \tag{2.4}$$

unter der Voraussetzung, daß eine Menge in (2.4) nichtleer ist.

Beweis: Nach Satz 2.7 bleibt zu zeigen:

$$\{y \,|\, y \geq 0, \; y^\top A = c^\top\} = \emptyset \;\Rightarrow\; \sup\{c^\top x \,|\, Ax \leq b\} = \infty. \tag{2.5}$$

und

$$\{x \,|\, Ax \leq b\} = \emptyset \;\Rightarrow\; \inf\{y^\top b \,|\, y \geq 0, \; y^\top A = c^\top\} = -\infty. \tag{2.6}$$

(1) zu (2.5): Sei $D = \{y \,|\, y \geq 0, \; y^\top A = c^\top\} = \emptyset$. Dann ist nach Voraussetzung $\{x \,|\, Ax \leq b\} \neq \emptyset$. Angenommen, das Supremum ist endlich. Mit $M(\gamma) = \{x \,|\, -c^\top x \leq -\gamma, \; Ax \leq b\}$ gibt es dann $\gamma_1 < \gamma_2$ mit $M(\gamma_1) \neq \emptyset$ und $M(\gamma_2) = \emptyset$. Nach Satz 2.6 ist $M(\gamma) \neq \emptyset$ genau dann, wenn aus $(y_0, y^\top) \geq 0, y^\top A - y_0 c^\top = 0$ folgt: $y^\top b - y_0\gamma \geq 0$. Mit $\gamma = \gamma_2$ folgt also die Existenz eines $\binom{y_0}{y} \in \mathbb{R}^{n+1}$ mit $y_0 \geq 0, y \geq 0, y^\top A = y_0 c^\top$ und $y_0\gamma_2 > y^\top b$. Mit $\gamma = \gamma_1$ gilt aber für dieses

$(y_0, y^\top)$ nun $y_0\gamma_1 \leq y^\top b$. Somit $y_0\gamma_2 > y_0\gamma_1$ und darum $y_0 > 0$. Dann ist aber $z := 1/y_0 \cdot y \in D$. Widerspruch.
(2) zu (2.6): Sei $\{x \mid Ax \leq b\} = \emptyset$. Nach Voraussetzung existiert ein y^0 mit $y^0 \geq 0,\ y^{0\top} A = c^\top$. Nach Satz 2.6 gibt es ein $y^1 \in \mathbb{R}^n$ mit $y^1 \geq 0,\ y^{1\top} A = 0$ aber $y^{1\top} b < 0$. Die Betrachtung von $y^0 + \lambda y^1,\ \lambda \geq 0$ zeigt sofort, daß das Infimum in 2.6 $-\infty$ ist. □

Bemerkung: Es ist möglich, daß beide Mengen in Satz 2.8 leer sind (Aufgabe!).

Es gibt für LP viele Formulierungen, die in dem Sinn äquivalent sind, daß sie leicht ineinander überführt werden können, z. B.

1. $\max\{c^\top x \mid Ax \leq b\}$,
2. $\max\{c^\top x \mid x \geq 0,\ Ax \leq b\}$ (kanonische Form),
3. $\max\{c^\top x \mid x \geq 0,\ Ax = b\}$,
4. $\min\{c^\top x \mid Ax \geq b\}$,
5. $\min\{c^\top x \mid x \geq 0,\ Ax \geq b\}$,
6. $\min\{c^\top x \mid x \geq 0,\ Ax = b\}$ (Standardform).

(2) wird in (1) überführt, da $x \geq 0,\ Ax \leq b$ äquivalent zu $\binom{-I}{A} x \leq \binom{0}{b}$. (3) wird in (2) überführt, indem $Ax = b$ durch $Ax \leq b,\ -Ax \leq -b$ ersetzt wird. (1) wird in (3) überführt, indem $Ax \leq b$ durch $x', x'', \tilde{x} \geq 0, Ax' - Ax'' + \tilde{x} = b$ ersetzt wird. Weiter sind (3) und (6) durch Austauschen von c und $-c$ ineinander überführbar.
Jedes der obigen Probleme, ebenso wie Mischformen heißt *allgemeines LP-Problem.*
Ebenso hat der Dualitätssatz äquivalente Formulierungen. So gilt z.B.

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

falls beide Mengen nichtleer sind.
Eine allgemeine Formulierung ist: Seien A, B, C, D Matrizen und a, b, c, d Vektoren. Dann gilt

$$\begin{aligned}&\max\{c^\top x + d^\top y \mid x \geq 0,\ Ax + By \leq a,\ Cx + Dy = b\} = \\ &\min\{u^\top a + v^\top b \mid u \geq 0,\ u^\top A + v^\top C \geq c^\top,\ u^\top B + v^\top D = d^\top\},\end{aligned}$$

falls beide Mengen nichtleer sind. (Man beachte, daß in den beiden Formeln $\geq$ und $\leq$ vertauscht sind.)
Das Maximum in der Formel ist äquivalent zu:

$$\max\{c^\top x + d^\top y \mid -Ix \leq 0,\ Ax + By \leq a,\ Cx + Dy \leq b,\ -Cx - Dy \leq -b\}.$$

Hierauf wendet man den Dualitätssatz an und erhält einen zum zweiten Ausdruck äquivalenten Ausdruck (Aufgabe!).
Beziehungen wie die obige bzw. im Satz 2.7 heißen Dualitätsgleichungen. Sie drücken aus, daß der optimale Wert eines gewissen Maximierungsproblems gleich dem Optimum eines anderen

Minimierungsproblems ist. Das Minimierungsproblem heißt das *duale* Problem des Maximierungsproblems, das dann *primal* heißt. Da sich Minimierungs– und Maximierungsprobleme ineinander überführen lassen, läßt sich ebenso das Minimierungsproblem als primales Problem auffassen. Man sieht leicht, daß das duale Problem eines dualen Problems wieder das primale Problem ist (Aufgabe!).
Anhand der folgenden Tabelle kann man ablesen, welche Fälle für das primale Problem **P** und das Duale **D** möglich sind.

$\mathbf{P} \setminus \mathbf{D}$	$\emptyset$	∞	opt
$\emptyset$	x	x	–
∞	x	–	–
opt	–	–	x

Die Bedingungen $Ax \leq b$, $x \geq 0$, $Cx = d$ usw. heißen *Restriktionen.* Ein Punkt x, der den Restriktionen genügt, heißt *zulässig.* Die Menge der zulässigen Punkte wird als *zulässiger Bereich* bezeichnet. Ist der zulässige Bereich nichtleer, so heißt das Problem *zulässig*, andernfalls *unzulässig.*
Die Funktion $x \mapsto c^\top x$ heißt *Zielfunktion* oder *Kostenfunktion.* $c^\top x$ heißt *Zielwert* oder *Kosten.* Ein zulässiger Punkt, dessen Kosten gleich dem Maximum sind, heißt *Optimum* oder *optimale Lösung.*

Satz 2.9 (Komplementäre Schlupfbedingungen) *Gegeben sei das LP*

$$\max\{c^\top x \,|\, Ax \leq b\}$$

und sein duales

$$\min\{y^\top b \,|\, y \geq 0,\ y^\top A = c^\top\}.$$

Sei x^0 zulässige Lösung des primalen und y^0 zulässige Lösung des dualen Problems. Dann ist äquivalent:

1. *x^0, y^0 sind optimale Lösungen,*
2. *$c^\top x^0 = y^{0\top} b$,*
3. *Ist eine Komponente von y^0 positiv, dann ist die entsprechende Ungleichung von $Ax \leq b$ mit Gleichheit erfüllt, d.h. $y^{0\top}(b - Ax^0) = 0$.*

Beweis: Die Äquivalenz von (1) und (2) folgt sofort aus Satz 2.7. Die Äquivalenz von (2) und (3) folgt, da in $c^\top x^0 = y^{0\top} A x^0 \leq y^{0\top} b$ Gleichheit gelten muß. $\square$

2.4 Der Simplexalgorithmus

Das älteste und wohl immer noch wichtigste Verfahren für LP ist der Simplexalgorithmus. In diesem Abschnitt wird seine Theorie dargestellt, im nächsten seine praktische Durchführung. Es wird ein Problem in Standardform

$$\min\{c^\top x \mid Ax = b,\ x \geq 0\}, \quad A \in \mathbb{R}^{m\times(n+m)},\ c \in \mathbb{R}^{n+m},\ b \in \mathbb{R}^m \tag{2.7}$$

betrachtet. Nach dem Vorhergehenden ist dies keine Einschränkung. Weiter lassen sich mit linearer Algebra (Gaußelimination) linear abhängige Zeilen von A entfernen. Daher sei o.B.d.A. $\text{rank} A = m$. Offensichtlich bleibt das Problem gleich, wenn für eine nichtsinguläre Matrix $Q \in \mathbb{R}^{m\times m}$ und ein $u \in \mathbb{R}^m$ A ersetzt wird durch QA, b ersetzt wird durch Qb und $c^\top$ ersetzt wird durch $c^\top + u^\top QA$ (es ändert sich nur der Wert des Minimums um $u^\top QAb$ wegen $u^\top QAx = u^\top Qb$).

Beispiel 2.3

$$A = \begin{pmatrix} 2 & 3 & 1 & 0 & 0 \\ 1 & 5 & 1 & 1 & 0 \\ 5 & 2 & 1 & 0 & 1 \end{pmatrix}, b = \begin{pmatrix} 1 \\ 3 \\ 4 \end{pmatrix}, c = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

Mit $P = \{x \in \mathbb{R}^{n+m} \mid Ax = b,\ x \geq 0\}$ wird das Polyeder der zulässigen Punkte bezeichnet. Das duale Problem ist $\min\{y^\top b \mid y^\top A \leq c^\top\}$. Schließlich sei wie üblich e^i der i–te Einheitsvektor.

Definition 2.6 *x heißt* Basislösung (basic solution, bs) *von (2.7), falls $Ax = b$ gilt und e^{i_j}, $j = 1, \dots, n$ existieren mit $e^{i_j\top} x = 0$ und* rank $\begin{pmatrix} e^{i_1\top} \\ \vdots \\ e^{i_n\top} \\ A \end{pmatrix} = n + m$*. Für eine Basislösung x und eine zugehörige Menge von Indizes $M = \{i_1, \dots, i_n\}$ heißt die geordnete Menge $B = \{k_1, \dots, k_m\}$ eine* zu x gehörige Basis*, falls $B = \{1, \dots, n+m\} \setminus M$.*
x heißt Ecke (zulässige Basislösung, basic feasible solution, bfs) *von (2.7), falls zusätzlich $x \geq 0$.*

Bemerkung: Man konstruiert eine Basislösung, indem man eine Menge von m linear unabhängigen Spaltenvektoren von A findet (deren Indizes eine Basis bilden), die Variablen, die zu den übrigen Spalten gehören (Nichtbasis–Variablen) auf Null setzt, und das so entstandene, eindeutig lösbare lineare Gleichungssystem löst. Sind alle Koordinaten der Lösung nichtnegativ, so hat man eine Ecke gefunden.

Beispiel 2.3 (Fortsetzung) *$x = (0, 0, 1, 2, 3)^\top$ ist eine Ecke und $B = \{3, 4, 5\}$ eine zu x gehörige Basis.*

Sei zunächst eine Ecke x^0 gegeben. Es wird eine Folge von Ecken konstruiert, bis entweder ein optimaler Punkt gefunden ist oder die Unbeschränktheit des Problems gezeigt wird.
Für x^0 sind die Spalten A^{k_j} mit $k_j \in B$ linear unabhängig. Daher existiert eine Matrix Q, so daß für $j = 1, \dots, m$ in QA in der k_j–ten Spalte der Vektor e^j steht.

Beispiel 2.3 (Fortsetzung) *Es ist*

$$Q = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \quad QA = \begin{pmatrix} 2 & 3 & 1 & 0 & 0 \\ -1 & 2 & 0 & 1 & 0 \\ 3 & -1 & 0 & 0 & 1 \end{pmatrix}, \quad Qb = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

Setze $u^\top = {c_B}^\top = (c_{k_1}, \dots, c_{k_m})$ und

$$\tilde{c}^\top = c^\top - u^\top QA.$$

Dann gilt $\tilde{c}^\top e^{k_j} = c_{k_j} - c_{k_j} = 0$, für $j = 1, \dots, m$. Ferner gilt für $\tilde{c}^\top$

$$\tilde{c}^\top x^0 = 0, \quad \tilde{c}^\top x = c^\top x - u^\top QAx = c^\top x - u^\top Qb = c^\top x - c^\top x^0, \quad \forall x \text{ mit } Ax = b.$$

(letzte Gleichung mit $x = x^0$). $\tilde{c}^\top x$ gibt also die Veränderung der Kosten bei Übergang von x^0 zu x an. Daher wird $\tilde{c}$ als *Vektor der relativen Kosten* bezeichnet.

Beispiel 2.3 (Fortsetzung) $\tilde{c}^\top = (-3, -3, 0, 0, 0)$.

Fallunterscheidung:
Fall 1: $\tilde{c} \geq 0$: Dann ist $u^\top QA \leq c^\top$, d.h. $u^\top Q$ ist dual zulässig und weiter

$$c^\top x^0 = c^\top x^0 - \tilde{c}^\top x^0 = u^\top QAx^0 = u^\top Qb \leq \max\{y^\top b \,|\, y^\top A \leq c^\top\} = \min\{c^\top x \,|\, x \geq 0,\ Ax = b\}$$

nach Satz 2.7. Damit ist x^0 optimal.
Fall 2: Wähle den kleinsten Index $i^\star$ mit $\tilde{c}^\top e^{i^\star} < 0$ und setze

$$y = e^{i^\star} - \sum_{j=1}^{m} (QA)_{ji^\star} e^{k_j}.$$

Es gilt $QAy = 0$ (wegen $QAe^{k_j} = e^j$ ergibt sich zweimal die $i^\star$–te Spalte von QA) und damit auch $Ay = 0$. Setzt man

$$x(\lambda) = x^0 + \lambda y,$$

so folgt $Ax(\lambda) = Ax^0 = b$, d.h. $x(\lambda)$ erfüllt die Gleichungen für alle $\lambda > 0$. Weiter gilt wegen $\tilde{c}^\top e^{k_j} = 0$

$$c^\top x(\lambda) = \tilde{c}^\top x(\lambda) + u^\top QAx(\lambda) = \tilde{c}^\top x^0 + \lambda \tilde{c}^\top \left(e^{i^\star} - \sum_{j=1}^{m} (QA)_{ji^\star} e^{k_j} \right) + u^\top Qb = \lambda \tilde{c}_{i^\star} + u^\top Qb.$$

Also $\tilde{c}^\top y < 0$ und die Zielfunktion fällt in λ. Wieder sind zwei Fälle zu unterscheiden:
Fall 2a: $(QA)_{ji^\star} \leq 0$, für alle j. Dann ist $x(\lambda)$ für alle $\lambda \geq 0$ zulässig und damit das Optimum unbeschränkt.
Fall 2b: $(QA)_{ji^\star} > 0$ für mindestens ein j. Sei jetzt

$$\lambda_0 = \min\left\{ \frac{x^0_{k_j}}{(QA)_{ji^\star}} \,|\, (QA)_{ji^\star} > 0 \right\}.$$

Weiter sei $k^\star$ der kleinste Index k_j, für den das Minimum angenommen wird (*nicht* die erste Zeile!). Dann ist $x(\lambda_0)$ Ecke von P und $(B \setminus \{k^\star\}) \cup \{i^\star\}$ eine Basis von $x(\lambda_0)$ (die $i^\star$–te Spalte von QA ist wegen $(QA)_{ji^\star} > 0$ nicht linear abhängig zu den Spalten zu $B \setminus \{k^\star\}$). Mit $x(\lambda_0)$ und dieser Basis wird das Verfahren fortgesetzt.

Beispiel 2.3 (Fortsetzung) $i^\star = 1,\ k^\star = 3,\ \lambda_0 = \min\{1/2, 1/1\} = 1/2$, $y = (1, 0, -2, 1, -3),\ x(\lambda_0) = (1/2, 0, 0, 5/2, 3/2),\ c^\top x(\lambda_0) = 9/2$.

Bemerkung: $\lambda_0 = 0$ und damit $x(\lambda_0) = x^0$ ist durchaus möglich. Dazu müssen aber einige (mindestens eine) Basisvariable 0 sein. Zu x^0 gibt es dann i.a. verschiedene Basen. Geometrisch bedeutet dies für das Polyeder P der zulässigen Lösungen mit Dimension n (im Lösungsraum von $Ax = b$, die Ungleichungen $x \geq 0$ liefern Halbräume), daß eine entartete Ecke vorliegt, d.h. eine Ecke liegt in mehr als n begrenzenden Hyperebenen (Facetten von P). Es könnte sein, daß im Laufe des Verfahrens ein *Zykel* auftritt, d.h. die Ecke ändert sich nicht, nur die Basen werden periodisch vertauscht. Durch die spezielle Wahl von $i^\star$ und $k^\star$ (hier: Blandsche Antizykelregel) wird dies aber ausgeschlossen.

Satz 2.10 *Der Algorithmus bricht stets mit Fall 1 oder Fall 2a ab.*

Beweis: Die Größen im j–ten Schritt werden mit $x^j,\ B_j, \tilde{c}^{j\top}, y^j, u^j, Q^j$ bezeichnet. Nach Konstruktion gilt $c^\top x^{j+1} \leq c^\top x^j$ und Gleichheit gilt nur falls $x^{j+1} = x^j$. Widerspruchsannahme: Der Algorithmus bricht nicht ab. Es gibt nur endlich viele Basen und so existieren $k, l \in \mathbb{N}$ mit $k < l$ und $B_k = B_l$. Da eine Basis eine Ecke eindeutig festlegt, gilt $x^k = x^l$ und damit $x^k = x^{k+1} = \ldots = x^l =: x$. Sei jetzt r der größte Index, so daß r in den Schritten $k, \ldots, l-1$ aus einer Basis eliminiert wurde, etwa im Schritt p. Wegen $B_k = B_l$ ist r in einem Schritt q mit $k \leq q < l$ der Basis hinzugefügt worden. Nun gilt für $k \leq j < l$:

$$\tilde{c}^{j\top} y^p = c^\top y^p - u^{j\top} Q^j A y^p = c^\top y^p,$$

also insbesondere $\tilde{c}^{q\top} y^p = \tilde{c}^{p\top} y^p < 0$. Darum gibt es einen Index a, so daß $\tilde{c}^q_a \cdot y^p_a < 0$.
(i) Ist $a > r$, dann gilt nach Definition von r entweder $a \in B_p \cap B_q$ oder $a \notin B_p$. Dann folgt aber $\tilde{c}^q_a = 0$ oder $y^p_a = 0$ (wegen $i^\star_p \leq r$).
(ii) Ist $a = r$, dann gilt wegen $r = i^\star_q$: $\tilde{c}^q_r < 0$ und wegen $r = k^\star_p$: $y^p_r < 0$.
(iii) Ist $a < r$, so gilt im Falle $x_a > 0$ dann $a \in B_q$ und daher $\tilde{c}^q_a = 0$. Sei also $x_a = 0$. Wegen $r = i^\star_q$ folgt $\tilde{c}^q_a \geq 0$. Angenommen, $y^p_a < 0$. Dann gilt $a = k_j$, für ein j mit $(QA)_{ji^\star} > 0$. Also $\lambda_0 = 0$ und $r = k^\star_p \leq a$. Widerspruch und darum $y^p_a \geq 0$.
In jedem Falle gilt also $\tilde{c}^q_a \cdot y^p_a \geq 0$. Widerspruch. $\square$

Der allgemeine Fall:
Gegeben sei das Problem $\min\{c^\top x \,|\, x \geq 0,\ Ax = b\}$. O.B.d.A. sei $b \geq 0$. Ist keine Ecke bekannt, so sei $\tilde{x} \in \mathbb{R}^m$ und $d \in \mathbb{R}^{n+2m}$ gegeben durch $d^\top = (0, \ldots, 0, 1, \ldots, 1)$ mit m Einsen. Es wird jetzt das neue Problem (Phase I–Problem)

$$\min\left\{ d^\top \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} \,|\, \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} \geq 0,\ (A, I)\begin{pmatrix} x \\ \tilde{x} \end{pmatrix} = b \right\}$$

betrachtet. Dies hat die Ecke $\binom{0}{b}$. Weiter ist das Minimum genau dann 0, wenn das ursprüngliche Problem zulässig ist. Sei x^0 die optimale Lösung des Hilfsproblems. Dann bilden die ersten $n + m$ Koordinaten von x^0 eine Ecke von P: Sei etwa B_0 die zu x^0 gehörige Basis. Gilt für alle $j \in B_0 \quad j \leq n + m$, so ist alles klar. Sei daher $j > n + m$. Dann gilt wegen $d^\top x^0 = 0$ notwendigerweise $x^0_j = 0$. Sei Q_0 die zu B_0 gehörige Matrix aus dem Algorithmus, so daß für

passendes i gilt $(Q_0A)_{ij} = 1$. Da die ersten $n+m$ Spalten von A den Rang m haben, existiert sicher ein $k \leq n+m$ mit $(Q_0A)_{ik} \neq 0$. Damit ist aber $B_0 \setminus \{j\} \cup \{k\}$ Basis. Nach endlich vielen Schritten ergibt sich so eine Basis zu x^0 als Ecke von P.
Hat A nicht Höchstrang, so kann man dies mit der Phase I feststellen: Das "Herausdrehen"der künstlichen Variablen aus der Basis kann nur fehlschlagen, wenn die i–te Zeile von Q_0A eine Nullzeile ist (bis zur $(n+m)$–ten Spalte). Wegen $x_j^0 = 0$ ist dann $(Qb)e^i = 0$. Die i–te Zeile ist also überflüssig (redundant) und kann gestrichen werden. Dreht man jeden Basisindex $j > n+m$ entweder heraus oder streicht die entsprechende Zeile, dann hat die so entstandene Matrix Höchstrang, der identisch mit dem Rang von A ist.
Zusammengefaßt ergibt sich:

Satz 2.11 *Sei das LP* $\min\{c^T x \mid x \geq 0,\ Ax = b\}$ *gegeben und A habe Höchstrang. Bezeichnet F die Menge der zulässigen Punkte, so gilt eine der drei folgenden Möglichkeiten:*

1) $F = \emptyset$.

2) $c^\top x$ ist auf F unbeschränkt.

3) Das Optimum wird in einer Ecke angenommen.

2.5 Das Simplextableau

Die Durchführung des Simplexalgorithmus erfolgt in einer $(m+1) \times (m+n+1)$-Matrix, dem sogenannten Simplextableau, dessen Elemente im folgenden mit ξ_{ij} bezeichnet werden:

$-c^\top x$	$\tilde{c}^\top$
Qb	QA

Zu Beginn wird die Berechnung von QA z.B. mit dem Gaußschen Eliminationsverfahren durchgeführt und $\tilde{c}^\top, -c^\top x, Qb$ berechnet. Also wird Beispiel 2.3 zu

Beispiel 2.4

0	1	1	1	1	1
1	2	3	1	0	0
3	1	5	1	1	0
4	5	2	1	0	1

Eine Basis ist $\{3,4,5\}$. Daraus ergibt sich das Starttableau

-6	-3	-3	0	0	0
1	2	3	1	0	0
2	-1	2	0	1	0
3	3	-1	0	0	1

Im Tableau entspricht jede Zeile j dem Basiselement k_j, in dessen Spalte die 1 der Einheitsuntermatrix von QA steht. Schließlich gibt ξ_{j0} den Wert der Basisvariablen x_{k_j} an.

Als nächstes erfolgt die Wahl einer Spalte $i^\star$ gemäß $i^\star = \min\{i \geq 1 \mid \xi_{0i} < 0\}$. λ_0 ergibt sich aus

$$\lambda_0 = \min\left\{\frac{\xi_{j0}}{\xi_{ji^\star}} \mid \xi_{ji^\star} > 0\right\}.$$

Schließlich ergibt sich das Element, das die Basis verläßt aus:

$$k^\star = \min\left\{k_j \mid \frac{\xi_{j0}}{\xi_{ji^\star}} = \lambda_0\right\}.$$

Die zugehörige Zeile ist $j^\star$, also $k^\star = k_{j^\star}$.

Beispiel 2.4 (Fortsetzung) *Gemäß dem vorigen Abschnitt ist* $\lambda_0 = 1/2$, $i^\star = 1$, $k^\star = 3$, $j^\star = 1$.

Es bleibt die Aufstellung des Tableaus für den nächsten Schritt. Dies geschieht vorteilhaft mit Hilfe des alten Tableaus. Im folgenden werden daher alte Größen mit einem "$'$" bezeichnet, also Q', ξ'_{lt} usw. Zunächst ist $B = (B' \setminus \{k^\star\}) \cup \{i^\star\}$, nun also $k_{j^\star} = i^\star$ statt $k^\star$.

Satz 2.12 (Transformationsgesetz) *Für* $0 \leq l \leq m$, $0 \leq t \leq n+m$ *gilt*

$$\xi_{lt} = \begin{cases} \xi'_{lt} - \frac{\xi'_{j^\star t}\xi'_{li^\star}}{\xi'_{j^\star i^\star}} & l \neq j^\star \\ \frac{\xi'_{lt}}{\xi'_{j^\star i^\star}} & l = j^\star \end{cases}. \tag{2.8}$$

Beweis: Die Berechnung von QA bedeutet die Inversion einer Matrix. Diese vereinfacht sich erheblich durch Berechnung von $(QQ'^{-1})(Q'A)$. Die Transformation (2.8) entspricht der Multiplikation mit einer Matrix R, die bis auf die $j^\star$–te Spalte die Einheitsmatrix ist (elementare Zeilenumformungen: Zeile $j^\star$ wird durch $\xi'_{j^\star i^\star}$ dividiert, zu der l–ten Zeile, $l \neq j^\star$, wird $\xi'_{li^\star}/\xi_{j^\star i^\star}$ mal die $j^\star$–te Zeile abgezogen). Es folgt $R = Q(Q')^{-1}$, wenn $(\xi_{lt}) \cdot e^{k_k} = e^j$, für $k_j \in B$, da Q hierdurch eindeutig bestimmt ist. Dies rechnet man für $t = k_j \in B' \backslash \{k^\star\}$ und für $t = i^\star = k_{j^\star}$ anhand (2.8) leicht nach.
Also gilt die Formel für $1 \leq l \leq m$, $1 \leq t \leq n+m$.
Wegen $Qb = RQ'b$ gilt sie auch für $t = 0, 1 \leq l \leq m$.
Für $l = t = 0$ folgt mit $(\tilde{c}')^\top x = c^\top(x - x')$

$$\xi_{00} = -c^\top x = -c^\top x' - \tilde{c}'^\top x = \xi'_{00} - \tilde{c}'_{i^\star} x_{i^\star} = \xi'_{00} - \frac{\xi'_{0i^\star}\xi'_{j^\star 0}}{\xi'_{j^\star i^\star}},$$

da $M' \cap B = \{i^\star\}$.
Für den Rest der 0–ten Zeile gilt

$$\tilde{c}^\top - \tilde{c}'^\top = c_{B'}{}^\top Q'A - c_B{}^\top QA = (c_{B'}{}^\top - c_B{}^\top R)Q'A$$

$c_{B'}$ und c_B unterscheiden sich nur in der $j^\star$–ten Stelle und I und R unterscheiden sich nur in der $j^\star$–ten Spalte. Es folgt, daß $\tilde{c}^\top - \tilde{c}'^\top = \mu(Q'A)^{j^\star}$, also ein Vielfaches der $j^\star$–ten Spalte des alten Tableaus ist. Wegen $\tilde{c}_{i^\star} = 0$ folgt $\mu = -(\xi'_{0i^\star})/(\xi'_{j^\star i^\star})$ und darum für $1 \leq t \leq n+m$: $\xi_{0t} = \xi'_{0t} - \frac{\xi'_{j^\star t}\xi'_{0i^\star}}{\xi'_{j^\star i^\star}}$. □

Üblicherweise bezeichnet man die $j^\star$–te Zeile als *Pivotzeile*, die $i^\star$–te Spalte als *Pivotspalte*, $\xi'_{j^\star i^\star}$ als *Pivotelement*, die Transformation als *Pivotschritt*. Damit läßt sich der Pivotschritt auch beschreiben durch: Man teile die Pivotzeile durch das Pivotelement und ziehe von den übrigen Zeilen Vielfache der Pivotzeile ab, so daß die übrigen Elemente der Pivotspalte Null werden.

Beispiel 2.4 (Fortsetzung) *Der Pivotschritt ergibt*

$-9/2$	0	$3/2$	$3/2$	0	0
$1/2$	1	$3/2$	$1/2$	0	0
$5/2$	0	$7/2$	$1/2$	1	0
$3/2$	0	$-11/2$	$-3/2$	0	1

Da alle Elemente von $\tilde{c}$ nichtnegativ sind, ist damit das Optimum bestimmt, es ist $(1/2, 0, 0, 5/2, 3/2)$*, sein Wert ist* $9/2$.

Zusammenfassend ergibt sich, daß so bei einem Pivotschritt $O(m(n+m))$ (Anzahl der Elemente in dem Tableau) Rechenschritte benötigt werden, eine völlige Neuberechnung jedoch $O(m^2(n+m))$ Rechenschritte benötigt.

Algorithmus 2.1 (Twophase)
Input: Ein LP in Standardform.
Output: Unzulässigkeit, Unbeschränktheit oder eine optimale Ecke.

```
Procedure Simplex
begin
  opt := nein, unbesch := nein;
  while opt = nein and unbesch = nein do
    if c̃_i ≥ 0, für alle i then opt := ja, stop ; else
    begin
      wähle ein i* mit c̃_{i*} < 0;
      if ξ_{ji*} ≤ 0, für alle j then unbesch := ja, stop ; else
      begin
        bestimme λ_0 = min{ ξ_{j0}/ξ_{ji*} | ξ_{ji*} > 0 } = ξ_{j*0}/ξ_{j*i*};
        pivotiere mit Pivotelement ξ_{j*i*};
      end;
    end;
end;

begin
  unzul := nein,  redund := nein;
  (*              Phase I              *)
  führe eine künstliche Basis x̃ ein;
  call simplex mit den Kosten d = ∑ x̃_j;
  if d_opt > 0 in Phase I then unzul := ja, stop ; else
  begin
    if eine künstliche Variable in der Basis, die nicht eliminiert werden kann then
```

```
    begin
      redund := ja;
      streiche die entsprechende Zeile;
    end;
    (*              Phase II              *)
    call Simplex mit den ursprünglichen Kosten;
  end;
end.
```

2.6 Der revidierte Simplexalgorithmus

Zur Durchführung eines Schrittes des Simplexalgorithmus ist die Kenntnis der Komponenten von $\tilde{c}^\top$ und nach Wahl der Spalte $i^\star$ die Kenntnis von $QA^{i^\star}$, wobei A^i die i–te Spalte von A bezeichnet, und Qb notwendig. Wir setzen voraus, daß das Starttableau eine Einheitsmatrix enthält und die zugehörigen c_i Null sind. Dies ist z.B. der Fall in der kanonischen Form nach Einführung von Schlupfvariablen oder wenn eine Ecke bekannt ist (dann sei o.E. $c = \tilde{c}$). Der entsprechende Ausschnitt im Starttableau ist

0	$0 \ \cdots \ 0$
b	I

.

Im Laufe des Verfahrens wird dies zu

$-c^\top x$	$-c_B Q$
Qb	Q

.

Hierin ist also alle nötige Information enthalten. Zunächst erzeuge man $\tilde{c}$ durch $\tilde{c}_i = c_i(c_B Q)A^i$ und wähle $i^\star$. Dann bestimme man die Pivotspalte $(QA)^{i^\star} = Q(A^{i^\star})$ und $j^\star$ wie gewohnt mit Hilfe der Pivotzeile und Qb. Die Umrechnung des Ausschnittes kann wie üblich erfolgen, da die Pivotspalte ja bekannt ist. Wird zusätzlich die Basis B als geordnete Menge gespeichert, so ergibt sich x aus $x_k = (Qb)_j$, falls k das j–te Element von B ist und $x_k = 0$, sonst.
Als Vorteil des revidierten Simplexalgorithmus ergibt sich zunächst, daß weniger Speicherplatz benötigt wird. Ferner sind für die Erneuerung des Tableaus nur $O(m^2)$ statt $O(m(m+n))$ Operationen notwendig. Als zusätzlicher Aufwand ist die Berechnung von $QA^{i^\star}$ und der $\tilde{c}_i$ dazugekommen. Der eine erfordert $O(m^2)$ Operationen, der andere einen Aufwand zwischen $O(m)$ und $O(m(n+m))$ Operationen. Damit ist der Aufwand des revidierten Algorithmus i.a. nicht kleiner als der Aufwand im normalen Tableau.
Der revidierte Algorithmus hat jedoch zwei weniger offensichtliche Vorteile: Bei manchen (kombinatorischen) Problemen ist n sehr groß gegen m und A nur als Vorschrift zur Spaltenerzeugung bekannt, so daß nur der revidierte Algorithmus praktikabel ist. Weiter enthält A oft vor allem

Nullen (schwach besetzte Matrix), QA jedoch keinesfalls, so daß der Aufwand beim revidierten Algorithmus viel geringer ist. Schließlich läßt sich mit speziellen Techniken der Aufwand zur Berechnung der $QA^{i^\star}$, $c_B{}^\top Q$ vermindern.

2.7 Pivotwahl und Antizykelalgorithmen

Die Beschreibung des Simplexalgorithmus in Abschnitt 2.4 bzw. seine Durchführung in Abschnitt 2.5 zeigen, daß für den Übergang von einem Tableau zum anderen nur die Wahl der Spalte $i^\star$ gemäß $\tilde{c}_{i^\star} < 0$ und der Zeile $j^\star$ gemäß $\frac{\xi_{j^\star 0}}{\xi_{j^\star i^\star}} = \lambda_0$ wichtig sind. Die spezielle Methode diente nur zum Nachweis der Endlichkeit. In der Praxis wird insbesondere Wert auf mehr Freiheit bei der Wahl der Spalte gelegt. So sind einige Möglichkeiten für die Spaltenwahl:

1. Man wählt das kleinste $\tilde{c}_i$. Diese Methode ist am weitesten verbreitet und entspricht dem steilsten Anstieg im Raum der Nichtbasisvariablen, da $\tilde{c}_i$ als Ableitung der Kostenfunktion nach der Nichtbasisvariablen x_i angesehen werden kann.

2. Man wählt die Spalte so, daß die Kosten im betreffenden Schritt maximal vermindert werden. Dies erfordert einen erheblichen Aufwand bei der Berechnung.

3. Ein Zuwachs der Nichtbasisvariablen x_i um 1 ändert x gemäß

$$x_k = \begin{cases} +1 & k = i \\ -\xi_{ji} & k = k_j,\ j = 1, \dots, m \\ 0 & sonst \end{cases} .$$

   Damit ergibt sich als Ableitung der Kosten nach dem Abstand im Raum aller Variablen $\tilde{c}_i / \sqrt{1 + \sum_{j=1}^m \xi_{ji}^2}$ und man wählt die Spalte mit der minimalen Ableitung.

Ist $\lambda_0 \neq 0$, so ändern sich die Kosten und damit ist eine Wiederholung der Basis unmöglich. Dies kann für $\lambda_0 = 0$ der Fall sein:

Beispiel 2.5 *Gegeben sei das Tableau*

3	$-3/4$	20	$-1/2$	6	0	0	0
0	$1/4$	-8	-1	9	1	0	0
0	$1/2$	-12	$-1/2$	3	0	1	0
1	0	0	1	0	0	0	1

Man verwende folgende Pivotsuche:
Spaltenwahl: Wähle das kleinste $\tilde{c}_i$.
Zeilenwahl: Wähle die erste Zeile, in der λ_0 angenommen wird.
Nach sechs Pivotschritten ist die erste Basis wieder erreicht.

Diese Erscheinung heißt Zykeln. Gesucht ist daher eine Methode, die bei freier Wahl der Spalte das Zykeln verhindert. Die einfachste Methode ist es, die Zeile nach einem Zufallsprinzip auszuwählen. Sie führt mit Wahrscheinlichkeit 1 zum Erfolg. Zur Einführung einer deterministischen Regel dient

Definition 2.7 *$v \in \mathbb{R}^n$ heißt lexikographisch positiv (lex–positiv), falls seine erste von Null verschiedene Komponente positiv ist. Ist $v = 0$, so heißt v lex–null, und v heißt lex–negativ, falls $-v$ lex–positiv ist.*
Schreibweisen: $v \overset{L}{>} 0$, $v \overset{L}{=} 0$, $v \overset{L}{<} 0$.
Entsprechend lassen sich lex–größer , lex–kleiner und lex–gleich definieren. Weiter sind die Begriffe lex–min *und* lex–max *in der offensichtlichen Weise definiert.*

Beispiel 2.6 $(0, 0, 1, -3) \overset{L}{>} (0, 0, 0, 2)$.

Zur Beschreibung des folgenden Verfahrens wird die i–te Zeile des Tableaus mit a_i bezeichnet. Zur Lösung von Mehrdeutigkeiten bei der Zeilenwahl beim Simplexalgorithmus wird jetzt die Zeile statt durch $\min\{\xi_{j0}/\xi_{ji^\star} \,|\, \xi_{ji^\star} > 0\}$ durch

$$\mathrm{lex} - \min\{a_j/\xi_{ji^\star} \,|\, \xi_{ji^\star} > 0\}$$

festgelegt. Da zwei Zeilen nie gleich sein können, ist damit eine eindeutige Pivotwahl gesichert. Weiter läßt sich durch passende Numerierung stets erreichen, daß die Zeilen des Tableaus zu Beginn des Algorithmus lex–positiv sind, da das Tableau eine Einheitsmatrix enthält.

Satz 2.13 (Lexikographische Antizykelregel) *Zu Beginn des Simplexalgorithmus gelte $a_i \overset{L}{>} 0$, $i = 1, \ldots, m$. Dann implizieren die folgenden Pivotregeln, daß die Zeilen lex–positiv bleiben, die Zeile Null lexikographisch wächst und der Simplexalgorithmus nach endlich vielen Schritten endet:*
a) Wähle eine Spalte $i^\star$ mit $\tilde{c}_{i^\star} < 0$.
b) Wähle die Zeile gemäß

$$a_{j^\star}/\xi_{j^\star i^\star} = \textit{lex–min}\{a_j/\xi_{ji^\star} \,|\, \xi_{ji^\star} > 0\}.$$

Beweis: Die $a_j, j = 1, \ldots, m$ bleiben nach dem Pivotschritt lex–positiv: Für $j = j^\star$ gilt $a_{j^\star} = a'_{j^\star}/\xi'_{j^\star i^\star}$ und so $a_{j^\star} \overset{L}{>} 0$. Für $j \neq j^\star$ und $\xi'_{ji^\star} > 0$ gilt nach der lexikographischen Wahl von $j^\star$:

$$a_j = a'_j - \frac{\xi'_{ji^\star}}{\xi'_{j^\star i^\star}} a'_{j^\star} = \xi'_{ji^\star}\left(\frac{a'_j}{\xi'_{ji^\star}} - \frac{a'_{j^\star}}{\xi'_{j^\star i^\star}}\right) \overset{L}{>} 0.$$

Schließlich ist für $\xi'_{ji^\star} \leq 0$

$$a_j = a'_j - (\xi'_{ji^\star}/\xi'_{j^\star i^\star}) \cdot a'_{j^\star} = a'_j + |\xi'_{ji^\star}/\xi'_{j^\star i^\star}| \cdot a'_{j^\star} \overset{L}{>} 0.$$

Für die 0-te Zeile gilt

$$a_0 = a'_0 - (\xi'_{0i^\star}/\xi'_{j^\star i^\star}) \cdot a'_{j^\star} \overset{L}{>} a'_0.$$

Damit wächst die 0–te Zeile lexikographisch streng monoton. Da die 0–te Zeile durch die Basis festgelegt ist, kann sich keine Basis wiederholen und die Endlichkeit des Algorithmus folgt aus der endlichen Anzahl von Basen. □

2.8 Der duale Simplexalgorithmus

Zu (irgend-) einer Basislösungen gemäß Definition 2.6 kann man ein Tableau aufstellen. I.a. wird weder $x \geq 0$ oder äquivalent $Qb \geq 0$ sein, noch $\tilde{c} \geq 0$ gelten. Der (primale) Simplexalgorithmus beginnt mit einer zulässigen Basis, d.h. $Qb \geq 0$. In Pivotoperationen wird $x \geq 0$ erhalten und $\tilde{c} \geq 0$ erreicht. Der duale Simplexalgorithmus beginnt mit einer optimalen, aber unzulässigen Lösung, d.h. $\tilde{c} \geq 0$. Anschließend werden Pivotoperationen durchgeführt, die $\tilde{c} \geq 0$ erhalten und $x \geq 0$ erreichen.

Beispiel 2.7 *Beispiel 2.3 mit der Basis* $\{1, 4, 2\}$ *ergibt das Tableau*

$$\begin{array}{r|cccccc}
-\frac{45}{11} & 0 & 0 & \frac{12}{11} & 0 & \frac{3}{11} \\ \hline
\frac{10}{11} & 1 & 0 & \frac{1}{11} & 0 & \frac{3}{11} \\
\frac{38}{11} & 0 & 0 & -\frac{5}{11} & 1 & \frac{7}{11} \\
-\frac{3}{11} & 0 & 1 & \frac{3}{11} & 0 & -\frac{2}{11}
\end{array}$$

Ist $\xi'_{j^\star i^\star}$ das Pivotelement, so gilt für die Pivotzeile $a_{j^\star} = a'_{j^\star}/\xi'_{j^\star i^\star}$. Also wird $j^\star$ so gewählt, daß $\xi'_{j^\star 0} < 0$ und $\xi'_{j^\star i^\star} < 0$ gilt. Weiter ist $i^\star$ so zu wählen, daß $\xi_{0i} \geq 0$, für $1 \leq i \leq n+m$. Dies wird gewährleistet durch die Wahl der Pivotspalte $i^\star$ mit $\xi'_{0i^\star}/\xi'_{j^\star i^\star} = \max\{\xi'_{0i}/\xi'_{j^\star i} \,|\, \xi'_{j^\star i} < 0\}$.

Beispiel 2.7 (Fortsetzung) $\xi_{3,5}$ *ergibt sich als Pivotelement. Der Pivotschritt ergibt*

$$\begin{array}{r|ccccc}
-\frac{9}{2} & 0 & \frac{3}{2} & \frac{3}{2} & 0 & 0 \\ \hline
\frac{1}{2} & 1 & \frac{3}{2} & \frac{1}{2} & 0 & 0 \\
\frac{5}{2} & 0 & \frac{7}{2} & \frac{1}{2} & 1 & 0 \\
\frac{3}{2} & 0 & -\frac{11}{2} & -\frac{3}{2} & 0 & 1
\end{array}$$

Damit ist das Tableau optimal.

Ist das Problem nicht entartet, also $\xi'_{0i^\star}/\xi'_{j^\star i^\star} < 0$, so folgt

$$\xi_{00} = \xi'_{00} - \frac{\xi'_{j^\star 0}\xi'_{0i^\star}}{\xi'_{j^\star i^\star}} < \xi'_{00}.$$

Damit kann sich keine Basis wiederholen und das Verfahren ist endlich. Im Fall der Entartung wird die Endlichkeit durch eine geeignete Modifizierung eines Antizykelalgorithmus erreicht.
Zum Start wird vorausgesetzt, daß alle Spalten lexikographisch positiv sind. Dies kann evtl. durch Zeilenvertauschungen und -umformungen geschehen. Eine sichere Methode ist es, die zusätzliche Restriktion

$$x_1 + \ldots + x_{n+m} \leq M$$

einzuführen, wobei M so groß ist, daß dies für alle zulässigen Punkte, die in Frage kommen, erfüllt ist, (M kann etwa aus der Problemstellung erraten werden oder auch systematisch bestimmt werden). Versehen mit einer Schlupfvariablen wird die Restriktion als erste Zeile hinzugefügt, so daß wegen $\tilde{c} \geq 0$ alle Spalten lexikographisch positiv sind.

Satz 2.14 *Beim Beginn des dualen Simplexalgorithmus seien alle Spalten nach der Nullten lexikographisch positiv,* $A_i \overset{L}{>} 0,\ i = 1, \ldots, n+m$*. Dann implizieren die folgenden Pivotregeln, daß diese Spalten lexikographisch positiv bleiben und die Spalte* 0 *lexikographisch streng fällt* ($A_0 \downarrow_{\mathrm{L}}$) *und daß der duale Simplexalgorithmus nach einer endlichen Anzahl von Schritten abbricht:*

(a) Wähle eine Zeile $j^\star$ *mit* $\xi_{j^\star 0} < 0$.

(b) Wähle die Spalte $i = i^\star$ *gemäß* $A_{i^\star}/\xi_{j^\star i^\star} = \mathrm{lex} - \max\{\mathrm{A_i}/\xi_{\mathrm{j^\star i}} \,|\, \xi_{\mathrm{j^\star i}} < 0\}$.

Beweis: Das Pivotieren bedeutet für die i–te Spalte

$$A_i = A'_i - \frac{\xi'_{j^\star i}}{\xi'_{j^\star i^\star}} \left(A'_{i^\star} - e^{j^\star} \right).$$

Da $A'_{i^\star} \overset{L}{>} 0$ und $\xi'_{j^\star i^\star} < 0$, muß $A'_{i^\star} - e^{j^\star} \overset{L}{>} 0$ gelten. Für Spalten mit $\xi'_{j^\star i} \geq 0$ folgt $A_i \overset{L}{\geq} A'_i \overset{L}{>} 0$. Für Spalten mit $\xi'_{j^\star i} < 0$ folgt nach Wahl von $i^\star$

$$A_i = \xi'_{j^\star i} \left(\frac{A'_i}{\xi'_{j^\star i}} - \frac{A'_{i^\star}}{\xi'_{j^\star i^\star}} \right) + \frac{\xi'_{j^\star i}}{\xi'_{j^\star i^\star}} e^{j^\star} \overset{L}{>} 0.$$

Für $i = 0$ gilt $A_0 = A'_0 - \frac{\xi'_{j^\star 0}}{\xi'_{j^\star i^\star}} \left(A'_{i^\star} - e^{j^\star} \right) \overset{L}{<} A'_0$. Darum kann keine Basis doppelt vorkommen, und das Verfahren bricht ab.

□

Die Wahl der Pivotspalte ist eindeutig, wenn alle Spalten paarweise verschieden sind. Dies kann man aber vor dem Start dadurch erreichen, daß Variable mit identischen Spalten zu einer zusammengefaßt werden.

Es bleibt die Interpretation des Algorithmus. Diese ergibt sich aus

$$c^\top = u^\top Q A + \tilde{c}^\top.$$

Die Bedingung an die Basis heißt also $c^\top \geq u^\top Q A$, d.h. $u^\top Q$ ist dual zulässig. Bezeichnet man mit A_k die Spalten von A, so gilt für $k \in B$ Gleichheit in $c_k \geq u^\top Q A_k$, d.h. $u^\top Q$ liegt in den m unabhängigen Ebenen $\{y \,|\, y^\top A_k = c_k,\ k \in B\}$, damit ist $y^\top = u^\top Q$ eine Ecke des zum dualen Problem gehörigen Polyeders. Also löst der duale Simplexalgorithmus das duale Problem durch Konstruktion einer Folge von dualen Ecken, für die die Zielfunktion $y^\top b$ wächst:

$$(u^\top Q) b = c^\top x = -\xi_{00}.$$

Diese Betrachtungen zeigen auch, daß unter gewissen Bedingungen, die optimale Lösung des dualen Problems aus dem Schlußtableau des primalen Problems abgelesen werden kann: Enthält das Starttableau eine Einheitsmatrix I und sind die zugehörigen Komponenten von c im Starttableau c', so enthält das Schlußtableau gerade oberhalb von I $\tilde{c}' = c' - u^\top Q$, d.h. die Lösung y des dualen Problems ergibt sich zu $y = c' - \tilde{c}'$.

	$c'^\top$	
	I	

Starttableau

	$\tilde{c}' = c'^\top - y^\top$	
	Q	

Endtableau

3 Ganzzahlige lineare Optimierung

3.1 Einführung

Bei vielen Optimierungsproblemen sind nur ganzzahlige Lösungen von Interesse, z.B. Stückzahlen von Flugzeugen, etc. Darum betrachten wir :

Definition 3.1 *Ganzzahlige lineare Optimierung (Integer linear programming, ILP):*

$$\min\{c^\top x \mid Ax = b,\ x \geq 0,\ x \in \mathbb{Z}^n\},$$

für $c \in \mathbb{Z}^n, \quad A \in \mathbb{Z}^{m\times n}, \quad b \in \mathbb{Z}^m$.
ILP kann ebenso in (wieder äquivalenter) kanonischer Form definiert werden.

Eine naheliegende Lösungsmöglichkeit wäre es, die Ganzzahligkeitsbedingung fallenzulassen, das normale LP zu lösen, und dann die optimale Lösung zu runden. Jedoch kann der Abstand der nähesten ganzzahligen Lösung sehr groß sein.

Beispiel 3.1 *Die optimale Lösung von* $\max\{x_1 \mid x_1, x_2 \geq 0,\ x_1 - 20x_2 \leq 0,\ x_1 + 20x_2 \leq 20\}$ *ist* $(10, 1/2)$. *Es gibt aber nur die beiden ganzzahligen Lösungen* $(0,0)$ *und* $(0,1)$.

Ferner muß die gerundete Lösung nicht optimal sein.

Beispiel 3.2 *Betrachte* $\min\{2x_1 - 3x_2 \mid x_1, x_2 \geq 0,\ -x_1 + 2x_2 \leq 3\}$. *Die optimale Lösung ist* $(0, 3/2)$ *mit Zielfunktionswert* -4.5. *Die gerundete Lösung ist* $(0,1)$ *mit Zielfunktionswert* -3, *aber* $(1,2)$ *ist zulässig mit Zielfunktionswert* -4.

Desweiteren ist ILP durch die Beschreibung kombinatorischer Bedingungen als Ganzzahligkeitsbedingungen motiviert. Ein Runden würde hierbei die Struktur des Problems verletzen.

Beispiel 3.3 *A) Das TSP als ILP: Sei ein TSP mit* $n+1$ *Städten* $0, 1, \ldots, n$ *und den Abständen* c_{ij} *gegeben. Sei* T *eine Tour, d.h. eine Rundreise, die alle Städte genau einmal durchläuft. Es wird* $x_{ij} = 1$ *gesetzt, falls von Stadt* i *zu Stadt* j *gegangen wird. Damit wird TSP:*

$$\min\{z = \sum_{i,j=0}^{n} c_{ij}x_{ij} \mid \qquad 0 \leq x_{ij} \leq 1 \text{ für alle } i,j, \tag{3.1}$$

$$x_{ij} \in \mathbb{Z} \text{ für alle } i,j, \tag{3.2}$$

$$\sum_{i=0}^{n} x_{ij} = 1, \quad j = 0, \ldots, n, \tag{3.3}$$

$$\sum_{j=0}^{n} x_{ij} = 1, \quad i = 0, \ldots, n\} \tag{3.4}$$

Gleichungen (3.3) liefern, daß in jede Stadt genau einmal hineingegangen wird. (3.4) liefert, daß aus jeder Stadt genau einmal herausgegangen wird.
Man beachte, daß TSP nicht vollständig erfaßt ist, da jede Menge disjunkter Kreise den obigen Restriktionen genügt. Dies kann aber durch zusätzliche Bedingungen verhindert werden.

B) Logische Ausdrücke (falls $x, y \geq 0$):

$$(x \geq a \textit{ oder } y \geq b) \quad \Leftrightarrow \quad x \geq \delta a;\; y \geq (1-\delta)b;\; \delta \in \mathbb{Z};\; 0 \leq \delta \leq 1.$$

Ersetzen einer Disjunktion durch MILP-Bedingung (MILP heißt mixed integer linear programming, da ganzzahlige Variable zusammen mit reellen Variablen auftreten).
C) Diskrete Variablen. Es sei eine Restriktion der Form $x \in \{s_1, \ldots, s_m\}$ gegeben. Darstellung: $x = s_1\delta_1 + \ldots + s_m\delta_m$ mit

$$\delta_1 + \ldots + \delta_m = 1,\; \delta_j \in \mathbb{Z},\; \delta_j \geq 0,\; j = 1, \ldots, m.$$

Zu einem ILP

$$\min\{c^\top x \mid Ax = b,\; x \geq 0,\; x \in \mathbb{Z}^n\} \tag{3.5}$$

kann man die zugehörige *LP–Relaxation*

$$\min\{c^\top x \mid Ax = b,\; x \geq 0\} \tag{3.6}$$

betrachten. Anhand der folgenden Tabelle kann man ablesen, welche Beziehungen zwischen (3.5) und (3.6) bestehen können.

ILP \ LP	$\emptyset$	∞	opt
$\emptyset$	x	x	x
∞	–	x	–
opt	–	–	x

Die erste Spalte erklärt sich von selbst. Genauso die zweite Zeile. Für die erste Zeile verwende man das Beispiel $\max\{x_1 \mid 3x_2 \geq 1,\; 3x_2 \leq 2,\; x_1 \geq 0\}$ bzw. mit zusätzlicher Restriktion $x_1 \leq 1$. Es bleibt nur zu zeigen, daß im Falle der Unbeschränktheit von (3.6) das ganzzahlige Problem entweder unzulässig oder unbeschränkt ist. Hierzu geben wir zunächst ein Kriterium mit Hilfe des Lemmas von Farkas.

Satz 3.1 *Es gilt genau dann $\min\{c^\top x \mid Ax = b,\; x \geq 0\} = -\infty$, falls ein $x \in \mathbb{R}^n$ mit $Ax = b,\; x \geq 0$ und ein $y \in \mathbb{R}^n$ mit $Ay = 0,\; c^\top y = -1,\; y \geq 0$ existiert.*

Beweis: Existieren x, y mit den genannten Eigenschaften, so ist das LP sicher unbeschränkt. Sei jetzt das LP unbeschränkt. Existiert kein x, so gilt $\min\{c^\top x \mid Ax = b,\; x \geq 0\} = \infty$. Existiert kein y, so folgt nach Satz 2.5, angewandt auf $A' = \binom{A}{c^\top}$ und $b' = \binom{0}{-1}$, daß es ein z gibt mit $z^\top\binom{A}{c^\top} \geq 0$ und $z^\top\binom{0}{-1} < 0$. Die letzte Koordinate von z sei daher o.B.d.A. 1. Sei jetzt z' der Vektor, den man aus z durch Weglassen der letzten Koordinate erhält, so folgt $z'^\top A \geq -c^\top$ oder $-z'^\top A \leq c^\top$, d.h. das duale Problem ist zulässig und so ein Widerspruch nach Satz 2.8. $\square$

Satz 3.2 *Ist (3.6) unbeschränkt und hat (3.5) eine zulässige Lösung, dann ist (3.5) ebenfalls unbeschränkt.*

Beweis: Sei x eine zulässige Lösung von (3.5). Offensichtlich ist x eine zulässige Lösung von (3.6). Weiter existiert nach Satz 3.1 eine Lösung y^0 von $Ay^0 = 0,\ c^\top y^0 = -1,\ y^0 \geq 0$.
Wir können o.B.d.A. annehmen, daß alle Koordinaten von y rational sind: Der Lösungsraum L von $Ay = 0, c^\top y = -1$, zusammen mit $y_i = 0$, für die $y_i^0 = 0$, hat eine rationale Basis. Somit kann man y^0 mit rationalen $y^n \in L$ approximeren, so daß auch für die übrigen Koordinaten $y_i^n > 0$ gilt.
Für ein geeignetes Vielfaches $z = \alpha y^0$ gilt also $z \in \mathbb{Z}^n,\ Az = 0,\ c^\top z < 0,\ z \geq 0$. Somit liefert $\{x + nz \mid n = 0, 1, \dots\}$ eine unbeschränkte Folge von ganzzahligen zulässigen Lösungen für (3.5).

□

3.2 Ein Schnittebenenverfahren von Gomory

Beim folgenden Schnittebenenverfahren wird für eine nichtganzzahlige Lösung des LP der zulässige Bereich weiter eingeschränkt, bis sich eine ganzzahlige Lösung ergibt. Sei das ILP (3.5) und die zugehörige LP–Relaxation (3.6) gegeben. Weiter seien F bzw. F^* die zulässigen Mengen von (3.5) bzw. (3.6).
Ist x^* eine optimale Ecke von (3.6), dann gilt:

1. Gilt $x^* \in \mathbb{Z}^n$, dann ist x optimal für (3.5).
2. $c^\top x^*$ ist eine untere Schranke für die Kosten in (3.5).

Ist $x^* \notin \mathbb{Z}^n$, so kann man versuchen, aus x^* und (3.6) neue lineare Restriktionen zu erzeugen, die

1. x^* abschneiden,
2. kein $x \in F$ abschneiden.

Eine solche Restriktion, die keine ganzzahlige Lösung ausschließt, heißt Schnittebene oder einfach Schnitt. Ein entsprechender Algorithmus stammt von R.E. Gomory. Dazu sei ein Beispiel von ILP gegeben. Eine Lösung von (3.6) mit dem Simplexalgorithmus ergibt eine Ecke x^* mit zugehöriger Basis $B = \{k_1, \dots, k_m\}$. Die Gleichungen im abschließenden Tableau haben die Gestalt

$$x_{k_i} + \sum_{j \notin B} \xi_{ij} x_j = \xi_{i0}, \quad 0 \leq i \leq m.$$

(Sind z die Kosten, so fasse man x_{k_0} als $-z$ auf. Der Fall $i = 0$ entspricht dann der Gleichung $\tilde{c}^\top x = c^\top x - c^\top x^*$.)
Aus $x \geq 0$ folgt für alle $x \in F^*$

$$\sum_{j \notin B} \lfloor \xi_{ij} \rfloor x_j \leq \sum_{j \notin B} \xi_{ij} x_j,$$

wobei $\lfloor \alpha \rfloor$ die größte ganze Zahl $\leq \alpha$ bedeutet. Damit wird aus der Gleichung des Tableaus

$$x_{k_i} + \sum_{j \notin B} \lfloor \xi_{ij} \rfloor x_j \leq \xi_{i0}. \tag{3.7}$$

Für jedes $x \in F$ ist die linke Seite von (3.7) in $\mathbb{Z}$. Daher gilt für $x \in F$

$$x_{k_i} + \sum_{j \notin B} \lfloor \xi_{ij} \rfloor x_j \leq \lfloor \xi_{i0} \rfloor.$$

Subtraktion von der ursprünglichen Gleichung ergibt: $\sum_{j \notin B}(\xi_{ij} - \lfloor \xi_{ij} \rfloor)x_j \geq \xi_{i0} - \lfloor \xi_{i0} \rfloor$. Bezeichnet $f_{ij} = \xi_{ij} - \lfloor \xi_{ij} \rfloor$ den gebrochenen Teil von ξ_{ij}, so ergibt sich die Restriktion

$$\boxed{\sum_{j \notin B} f_{ij} x_j \geq f_{i0}.} \tag{3.8}$$

Diese heißt Gomory-Schnitt zur erzeugenden Zeile i und wird dem Tableau hinzugefügt. Damit x Basislösung bleibt, wird (3.8) mit -1 multipliziert und die Schlupfvariable s hinzugefügt:

$$\sum_{j \notin B} -f_{ij} x_j + s = -f_{i0}.$$

Der Effekt dieses Prozesses für ein optimales Tableau ist:

Satz 3.3 *Wird ein Gomoryschnitt dem optimalen Tableau eines LP hinzugefügt, dann werden keine ganzzahligen zulässigen Punkte ausgeschlossen und das neue Tableau ist ein Basistableau, das primal unzulässig ist, wenn ξ_{i0} nicht ganzzahlig ist, und dual zulässig ist.*

Beweis: Nach Konstruktion wird keine ganzzahlige zulässige Lösung ausgeschlossen. s ist eine neue Basisvariable und bildet zusammen mit der optimalen Basis B wieder eine Basis. Ist ξ_{i0} nicht ganzzahlig, dann gilt $f_{i0} > 0$ und die Basislösung schließt $s = -f_{i0}$ ein und ist damit primal unzulässig. Sie bleibt dual zulässig, da die nullte Zeile nicht geändert wird. □

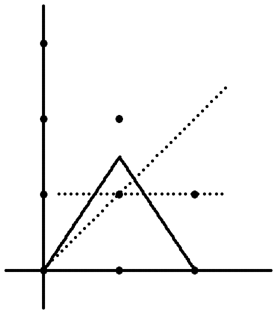
Es ergibt sich

Algorithmus 3.1

Procedure Gebrochen–Dual (GDA);
begin
  löse (3.6) mit optimaler Lösung x^*;
  Zulässig:='Ja';
  while (x^* nicht ganzzahlig) **and** (Zulässig='Ja') **do**
  begin
    Wähle eine erzeugende Zeile i;
    Füge den erzeugten Gomory-Schnitt und eine entsprechende Basisvariable s zu (3.6) hinzu;
    Wende den dualen Simplexalgorithmus an;
    if (duales Problem ist unbeschränkt) **then** Zulässig:='Nein';
    Sei x^* das neue Optimum;
  end;
end.

Der Algorithmus heißt gebrochen–dualer Algorithmus, da gebrochene Anteile auftreten und die duale Zulässigkeit aufrechterhalten wird.

Beispiel 3.4 *Gegeben sei das ILP*



$$\max x_2$$
$$3x_1 + 2x_2 \leq 6$$
$$-3x_1 + 2x_2 \leq 0$$
$$x \geq 0, x \text{ ganzzahlig}$$

Mit den Schlupfvariablen x_3 und x_4 wird die Standardform

		x_1	x_2	x_3	x_4
$-z =$	0	0	-1	0	0
$x_3 =$	6	3	2	1	0
$x_4 =$	0	-3	2	0	1

Das optimale Tableau folgt mit zwei Pivotschritten

		x_1	x_2	x_3	x_4
$-z =$	3/2	0	0	1/4	1/4
$x_1 =$	1	1	0	1/6	$-1/6$
$x_2 =$	3/2	0	1	1/4	1/4

Dies entspricht $x = (1, 3/2)$ mit den Kosten $z = -3/2$. Wegen $x \notin \mathbb{Z}^2$ werden Schnitte erzeugt: Die Nullte Gleichung ist

$$-z + 1/4x_3 + 1/4x_4 = 3/2.$$

Dies ergibt den Schnitt

$$1/4x_3 + 1/4x_4 \geq 1/2.$$

Ersetzen von x_3 und x_4 durch die Gleichungen des Starttableaus zeigt, daß dies mit $x_2 \leq 1$ äquivalent ist.
(Die Wahl der ersten bzw. zweiten Zeile als erzeugende Zeile hätte als Schnitt

$$1/6x_3 + 5/6x_4 \geq 0 \quad \textit{bzw.} \quad 1/4x_3 + 1/4x_4 \geq 1/2$$

ergeben. Dies ist zu $x_2 \leq x_1 + 1/2$ bzw. $x_2 \leq 1$ äquivalent.)
Hinzufügen von $-1/4x_3 - 1/4x_4 + s_1 = -1/2$ ergibt:

		x_1	x_2	x_3	x_4	s_1
$-z =$	3/2	0	0	1/4	1/4	0
$x_1 =$	1	1	0	1/6	$-1/6$	0
$x_2 =$	3/2	0	1	1/4	1/4	0
$s_1 =$	$-1/2$	0	0	$\mathbf{-1/4}$	$-1/4$	1

Der angegebene duale Pivotschritt ergibt

		x_1	x_2	x_3	x_4	s_1
$-z =$	1	0	0	0	0	1
$x_1 =$	2/3	1	0	0	$-1/3$	2/3
$x_2 =$	1	0	1	0	0	1
$x_3 =$	2	0	0	1	1	-4

Die erste Zeile ergibt den nichttrivialen Schnitt $2/3x_4 + 2/3s_1 \geq 2/3$ (äquivalent zu $x_1 \geq x_2$). Hinzufügen von $-2/3x_4 - 2/3s_1 + s_2 = -2/3$ zum Tableau ergibt

		x_1	x_2	x_3	x_4	s_1	s_2
$-z =$	1	0	0	0	0	1	0
$x_1 =$	2/3	1	0	0	$-1/3$	2/3	0
$x_2 =$	1	0	1	0	0	1	0
$x_3 =$	2	0	0	1	1	-4	0
$x_6 =$	$-2/3$	0	0	0	$-2/3$	$-2/3$	1

Optimieren mit dem dualen Simplexalgorithmus ergibt das optimale Tableau:

		x_1	x_2	x_3	x_4	s_1	s_2
$-z =$	1	0	0	0	0	1	0
$x_1 =$	1	1	0	0	0	1	$-1/2$
$x_2 =$	1	0	1	0	0	1	0
$x_3 =$	1	0	0	1	0	-5	3/2
$x_4 =$	1	0	0	0	1	1	$-3/2$

Satz 3.4 *Sei der GDA mit folgenden Regeln gegeben:*

(a) Die erzeugende Zeile sei die erste mit $\xi_{i0} \notin \mathbb{Z}$.

(b) Man benutze die lexikographische Version des dualen Simplexalgorithmus.

Dann gibt der GDA nach einer endlichen Zahl von Schritten die optimale Lösung oder die Unzulässigkeit des ILP.

Beweis: Die Hauptschleife im Algorithmus heiße Reoptimierung und das Tableau nach der l–ten Reoptimierung sei A^l mit den Spalten A^l_j. Da Zeilen immer am Ende des Tableaus angefügt werden, folgt $A^1_0 \overset{L}{>} A^2_0 \overset{L}{>} A^3_0 \overset{L}{>} \dots$. Ist das ILP zulässig, so ist die erste Komponente $\xi^l_{00} = -z$ nach unten beschränkt. Wegen der Monotonie konvergiert also ξ^l_{00} gegen $w_{00} = \lfloor w_{00} \rfloor + f_{00}$. Für hinreichend großes $l = k$ gilt somit

$$\xi^k_{00} = \lfloor w_{00} \rfloor + f^k_{00}.$$

Nach der Regel (a) ist 0 die nächste erzeugende Zeile und der Schnitt

$$-f^k_{00} = -\sum_{j \notin B} f^k_{0j} x_j + s$$

wird dem Tableau hinzugefügt. Der nächste Schritt im dualen Algorithmus habe p als Pivotspalte, die in die Basis kommt. Nach diesem Pivotschritt wird

$$\xi^{k+1}_{00} = \xi^k_{00} - \frac{\xi^k_{0p}}{f^k_{0p}} f^k_{00}. \tag{3.9}$$

In einem optimalen Tableau des dualen Simplexalgorithmus gilt $\xi^k_{0p} \geq 0$ und somit ist ξ^k_{0p} größer als sein gebrochener Anteil: $\xi^k_{0p} \geq f^k_{0p}$. Aus (3.9) folgt dann

$$\xi^{k+1}_{00} \leq \xi^k_{00} - f^k_{00} = \lfloor \xi^k_{00} \rfloor = \lfloor w_{00} \rfloor.$$

Wegen der Monotonie gilt $\xi_{00}^l = \lfloor w_{00} \rfloor$, für $l \geq k+1$. Es folgt nun, daß ξ_{01}^l monoton fallend und beschränkt (nämlich ≥ 0) ist. Aus $\xi_{00}^{l+1} = \xi_{00}^l$ folgt $\xi_{0p}^l = 0$ und da A_p^l lex–positiv ist, gilt $\xi_{1p}^l \geq 0$. Wie oben ist dann auch ξ_{10}^l für hinreichend große l konstant und ganzzahlig.
Dieses Argument kann wiederholt werden, bis entweder alle Variablen ganzzahlig sind oder der duale Algorithmus die Unbeschränktheit des dualen Problems und damit die Unzulässigkeit des ILP liefert.

□

Beim Algorithmus in dieser Form wird die Zahl der Gleichungen immer größer. Dies läßt sich vermeiden, indem die Schlupfvariable s_i mit der zugehörigen Zeile aus dem Tableau entfernt wird, wenn diese Variable in die Basis kommt. Auf diese Weise wird die Zahl der Zeilen nie größer als die Zahl der ursprünglichen Variablen. Es läßt sich zeigen, daß der Algorithmus auch bei dieser Variante abbricht und daß die Zahl der Schritte exponentiell beschränkt ist.

4 Konvexe Optimierung

4.1 Einleitung

Die lineare Optimierung ist durch Dantzigs Simplexverfahren sowohl theoretisch als auch praktisch im Prinzip vollständig erschlossen. Denn das Verfahren liefert immer eine Antwort und das mit relativ geringem Aufwand. Wie in anderen mathematischen Disziplinen bieten lineare Probleme oft nur ein idealisiertes Bild der realen Probleme, z.B. sind Kosten i.a. nicht proportional zu Stückzahlen (Mengenrabatte, Rationalisierung etc.). Wird dies bei den mathematischen Modellen berücksichtigt, gelangt man zu nichtlinearen Optimierungsaufgaben, die allgemein sowohl theoretisch als auch praktisch kein Patentverfahren zulassen. Der allgemeine Typ lautet:

Minimiere $F(x)$ unter den Nebenbedingungen

$$f_i(x) \leq 0,\ i = 1, \ldots, m \quad \text{und} \quad g_j(x) = 0,\ j = 1, \ldots, k.$$

Dabei sind F und die f_i, g_j stetige, reellwertige Funktionen von $x \in \mathbb{R}^n$. Ist F etwa die Höhe des Luftdrucks, so zeigt ein Blick auf die Wetterkarte, daß es viele lokale Minima und Maxima gibt. Damit wird das Auffinden eines absoluten Extremums höchst schwierig. Wenn man jedoch von den Restriktionen f_i, g_j und der Zielfunktion einschränkend Konvexität verlangt, dann kann man die Aufgaben theoretisch vernünftig behandeln und hat auch praktisch eine Fülle von Methoden zur Verfügung. Es wird sich zeigen, daß die Zielfunktion, wenn überhaupt, ihr Minimum auf einer konvexen Punktmenge annimmt und daß dieses Minimum absolut ist und keine (weiteren) relativen Minima existieren.

4.2 Konvexe Funktionen

Definition 4.1 *Sei $K \subset \mathbb{R}^n$ konvex. Eine Funktion $f : K \to \mathbb{R}$ heißt* konvex, *wenn*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y), \quad \text{für alle } x, y \in K \text{ und } \lambda \in (0,1). \tag{4.1}$$

f heißt streng konvex, *wenn für $x \neq y$ in (4.1) sogar $<$ gilt. Eine Funktion f heißt (streng)* konkav, *wenn $-f$ (streng) konvex ist.*

Insbesondere sind lineare Funktionen sowohl konvex als auch konkav.
Eine einfache Eigenschaft ist

Lemma 4.1 *Sei $K \subset \mathbb{R}^n$ konvex, $f, g : K \to \mathbb{R}$ konvexe Funktionen und $\alpha, \beta \geq 0$. Dann ist $\alpha f + \beta g$ konvex.*

Beweis: Nachrechnen. □

Eine Verbindung zwischen konvexen Mengen und konvexen Funktionen gibt

Lemma 4.2 *Sei $K \subset \mathbb{R}^n$ konvex. Die Funktion $f : K \to \mathbb{R}$ ist genau dann konvex, wenn die Teilmenge $\{(\mu, x) : x \in K, \mu \in \mathbb{R}, \mu \geq f(x)\}$ des $\mathbb{R}^{n+1}$ konvex ist.*

Beweis: Aufgabe. $\square$

Lemma 4.3 (Jensensche Ungleichung) *Sei $K \subset \mathbb{R}^n$ konvex und $f : K \to \mathbb{R}$ konvexe Funktion. Dann gilt für $x^1, \ldots, x^m \in K$ und $\lambda_1, \ldots, \lambda_m \geq 0$ mit $\sum_{i=1}^m \lambda_i = 1$*

$$f\left(\sum_{i=1}^m \lambda_i x^i\right) \leq \sum_{i=1}^m \lambda_i f(x^i).$$

Beweis: Die Menge $M = \{(\mu, x) \in \mathbb{R}^n \,|\, x \in K, \mu \geq f(x)\}$ ist konvex, da f konvex ist. Damit ist jede Konvexkombination von Punkten aus M wieder in M. Dies ist äquivalent zur Ungleichung. $\square$

Lemma 4.4 *Sei $K \subset \mathbb{R}^n$ offen und konvex und $f : K \to \mathbb{R}$ konvex. Dann ist f stetig auf K.*

Beweis: Sei $x^0 \in K$. Dann gibt es einen Würfel $W = \text{conv}\{x^1, \ldots, x^{2^n}\}$ mit Mittelpunkt x^0 und $W \subset K$. Für alle $x \in W$ gilt nach Lemma 4.3 $f(x) \leq \max_{1 \leq i \leq 2^n} f(x^i) =: M$. Sei $B_r(x^0)$ eine Kugel mit Radius $r > 0$ um x^0, so daß $B_r(x^0) \subset W$ und somit $f(x) \leq M$, für $x \in B_r(x^0)$. Sei nun $\epsilon > 0$ beliebig und (o.E. $f(x^0) < M$) $\delta := \min\{r, \epsilon r/(M - f(x^0))\} > 0$. Für $y \in B_\delta(x^0)$ sei

$$x^1 = x^0 + \frac{r}{\delta}(y - x^0) \quad \text{und} \quad x^2 = x^0 - \frac{r}{\delta}(y - x^0).$$

Dann gilt $x^1, x^2 \in B_r(x^0)$ und

$$y = \frac{\delta}{r}x^1 + (1 - \frac{\delta}{r})x^0 \quad \text{und} \quad x^0 = \frac{\delta}{\delta + r}x^2 + \frac{r}{\delta + r}y.$$

Aus der Konvexität von f folgt, daß $f(y) - f(x^0) \leq \delta/r(M - f(x^0)) \leq \epsilon$ und $f(x^0) - f(y) \leq \delta/r(M - f(x^0)) \leq \epsilon$. Also ist f in x^0 stetig. $\square$

Bemerkung: Für allgemeine konvexe Mengen gilt Lemma 4.4 nicht (vgl. Übung).

Definition 4.2 *Sei A eine $(n \times n)$–Matrix. Ist $x^\top Ax \geq 0$, für alle $x \in \mathbb{R}^n$, so heißt A* positiv semidefinit. *Gilt für $x \neq 0$ sogar $x^\top Ax > 0$, dann heißt A* positiv definit.

Satz 4.1 *Sei A eine positiv definite $(n \times n)$–Matrix. Dann ist $f(x) = x^\top Ax$ streng konvex im $\mathbb{R}^n$. Ist A positiv semidefinit, dann ist f konvex.*

Beweis: Sei A positiv definit und $0 < \lambda < 1$. Dann gilt für $x \neq y$

$$\begin{aligned} f(\lambda x + (1 - \lambda)y) &= f(y + \lambda(x - y)) = \{y + \lambda(x - y)\}^\top A\{y + \lambda(x - y)\} \\ &= \lambda^2(x - y)^\top A(x - y) + \lambda(x - y)^\top Ay + \lambda y^\top A(x - y) + y^\top Ay. \end{aligned}$$

Wegen $\lambda^2 < \lambda$ folgt

$$\begin{aligned} f(\lambda x + (1 - \lambda)y) &< \lambda(x - y)^\top A(x - y) + \lambda(x - y)^\top Ay + \lambda y^\top A(x - y) + y^\top Ay \\ &= \lambda(x^\top Ax - y^\top Ay) + y^\top Ay = \lambda f(x) + (1 - \lambda)f(y). \end{aligned}$$

Ist A positiv semidefinit, so gilt nur $\leq$ statt $<$. $\square$

Satz 4.2 *Sei $K \subset \mathbb{R}^n$ konvex und $f : K \to \mathbb{R}$ partiell differenzierbar. Dann gilt:*
(a) f ist konvex, genau dann, wenn für alle $x, y \in K$ gilt

$$f(y) \geq f(x) + (y - x)^\top \operatorname{grad} f(x).$$

(b) f ist streng konvex, genau dann, wenn für alle $x, y \in K, x \neq y$ gilt

$$f(y) > f(x) + (y - x)^\top \operatorname{grad} f(x).$$

Beweis: (1) Es sei $\lambda \in (0, 1)$ und $z = \lambda x + (1 - \lambda)y$. Dann gilt $f(x) \geq f(z) + (x - z)^\top \operatorname{grad} f(z)$ und $f(y) \geq f(z) + (y - z)^\top \operatorname{grad} f(z)$. Multiplikation mit λ bzw. $(1 - \lambda)$ und Addieren ergibt $\lambda f(x) + (1 - \lambda)f(y) \geq f(z)$, d.h. f ist konvex. Analog für streng konvex mit $<$.
(2) Sei f konvex und $x, y \in K$. Für $\lambda \in [0, 1]$ setze

$$F(\lambda) = (1 - \lambda)f(x) + \lambda f(y) - f((1 - \lambda)x + \lambda y).$$

Offensichtlich gilt $F(\lambda) \geq 0$ und $F(0) = F(1) = 0$. Für $\lambda \leq 1/2$ gilt $(1 - \lambda)x + \lambda y = (1 - 2\lambda)x + 2\lambda(x + y)/2$ und darum

$$F(\lambda) \geq (1-\lambda)f(x)+\lambda f(y)-(1-2\lambda)f(x)-2\lambda f((x+y)/2) = \lambda(f(x)+f(y)-2f((x+y)/2)).$$

Mit $c := f(x) + f(y) - 2f((x + y)/2) \geq 0$ gilt also $F(\lambda) \geq c\lambda$ und somit $F'(0) \geq c$. Aus $F'(0) = f(y) - f(x) - (y - x)^\top \operatorname{grad} f(x)$ folgt die Behauptung (beachte $c > 0$, wenn f streng konvex).

$\square$

Satz 4.3 *Sei $K \subset \mathbb{R}^n$ konvex und $f : K \to \mathbb{R}$ sei 2–mal stetig partiell differenzierbar. Ist die (Hessesche) Matrix $A(x) = \left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j}\right)_{i,j=1,\ldots n}$ für alle $x \in K$ positiv semidefinit (positiv definit), dann ist f konvex (streng konvex) auf K.*

Beweis: Entwickle f nach Taylor: Zu $x, y \in K$ gibt es ein ξ auf der Verbindungsstrecke mit

$$f(y) = f(x) + (y - x)^\top \operatorname{grad} f(x) + \frac{1}{2}(y - x)^\top A(\xi)(y - x).$$

Es ist $A(\xi)$ positiv semidefinit, also

$$f(y) \geq f(x) + (y - x)^\top \operatorname{grad} f(x).$$

Nach Satz 4.2 ist f also konvex. Analog mit $>$, wenn $A(\xi)$ positiv definit ist. $\square$

Bei der Linearen Optimierung war die Menge der Minimallösungen eine *Seite* des zulässigen Polyeders. Hier ist diese Menge zumindest konvex.

Satz 4.4 *Sei $K \subset \mathbb{R}^n$ konvex und $f : K \to \mathbb{R}$ konvex. Dann gilt*
(a) Jedes relative Minimum von f ist absolutes Minimum.
(b) Die Menge der Minimallösungen ist konvex.

Beweis: (a) Sei x relatives Minimum und $y \in K$. Für $\lambda > 0$ hinreichend klein gilt dann also $f((1-\lambda)x + \lambda y) \geq f(x)$. Also

$$f(x) \leq f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y),$$

woraus $f(x) \leq f(y)$ folgt.
(b) Sind $x, y \in K$ Minimalpunkte mit $f(x) = f(y) = M$, so gilt für $\lambda \in (0,1)$ einerseits $f(\lambda x + (1-\lambda)y) \geq M$. Andererseits gilt wegen der Konvexität $f(\lambda x + (1-\lambda)y) \leq M$. □

Satz 4.5 *Sei $K \subset \mathbb{R}^n$ konvex und $f : K \to \mathbb{R}$ streng konvex. Dann gibt es höchstens einen Minimalpunkt.*

Beweis: Seien $x \neq y$ Minimalpunkte mit $f(x) = f(y) = M$. Dann gilt $f((x+y)/2) < 1/2 f(x) + 1/2 f(y) = M$, Widerspruch. □

4.3 Trennungssätze

Der Fundamentalsatz über lineare Ungleichungen kann geometrisch wie folgt interpretiert werden. Liegt ein Punkt b außerhalb eines Kegels K, so gibt es eine (Trenn–) Hyperebene H, so daß b und K auf verschiedenen Seiten von H liegen. Solche *Trennungssätze* gelten aber viel allgemeiner.

Satz 4.6 *Seien $K_1, K_2 \subset \mathbb{R}^n$ konvex, K_1 sei abgeschlossen, K_2 kompakt und $K_1 \cap K_2 = \emptyset$. Dann gibt es eine* (echte) Trennhyperebene *zu K_1 und K_2, d.h. es gibt ein $c \in \mathbb{R}^n \setminus \{0\}$ und ein $\gamma \in \mathbb{R}$ mit $c^\top x < \gamma < c^\top y$, für alle $x \in K_1, y \in K_2$.*

Beweis: Sei $d := \inf\{\|x-y\| \mid x \in K_1, y \in K_2\}$. Das Infimum wird angenommen:
Es gibt Folgen $x^\nu \in K_1, y^\nu \in K_2$ mit $\|x^\nu - y^\nu\| \to d$, für $\nu \to \infty$. $\{y^\nu\}$ hat nach Bolzano–Weierstraß eine konvergente Teilfolge, o. E. sei $y^\nu \to v \in K_2$. Die x^ν sind o. E. alle aus $K_1 \cap B_{d+1}(v)$ (kompakt). Also hat auch $\{x^\nu\}$ eine konvergente Teilfolge, o. E. sei $x^\nu \to u \in K_1$. Es folgt $\|u - v\| = \lim \|x^\nu - y^\nu\| = d$.
Somit gilt $\|x - y\| \geq \|u - v\|$, für alle $x \in K_1, y \in K_2$. Es sei nun $x \in K_1$, dann gilt für die Funktion

$$d(\lambda) = \|((1-\lambda)u + \lambda x) - v\|^2 = \|u - v + \lambda(x-u)\|^2$$

$d'(0) \geq 0$. Nach der Kettenregel ist $d'(0) = 2(u-v)^\top(x-u)$ und mit $c = (v-u)$ folgt, daß $c^\top x \leq c^\top u$. Analog gilt $c^\top y \geq c^\top v > c^\top u$, für alle $v \in K_2$ und mit $\gamma = c^\top(u+v)/2$ folgt die Behauptung. □

Satz 4.7 *Sei $K \subset \mathbb{R}^n$ konvex und abgeschlossen und $p \in \mathrm{bd}(K)$ ein Randpunkt von K. Dann gibt es eine* Stützhyperebene *zu K, die p enthält, d.h. es gibt ein $c \in \mathbb{R}^n \setminus \{0\}$ mit $c^\top x \leq c^\top p$, für alle $x \in K$.*

Beweis: Es gibt eine Folge von Punkten $\{p^\nu\}$ mit $p^\nu \to p$ und $p^\nu \notin K$. Nach Satz 4.6 gibt es Vektoren c^ν mit $\|c^\nu\| = 1$ mit $c^{\nu\top} x < c^{\nu\top} p^\nu$, für alle $x \in K$. O.E. sei c^ν konvergent gegen c mit $\|c\| = 1$ (Bolzano–Weierstraß). Dann gilt für alle $x \in K$

$$c^\top x = \lim c^{\nu\top} x \leq \lim c^{\nu\top} p^\nu = c^\top p.$$

□

.

Bemerkung: Die Bedingung K_2 ist notwendig, wie das Beispiel $K_1 = \{(x_1, x_2) \,|\, x_1 \geq 0, x_2 \leq\}$ und $K_2 = \{(x_1, x_2) \,|\, x_2 \geq 1/x_1\}$ zeigt.

Satz 4.8 (Trennungssatz) *Es seien $K_1, K_2 \subset \mathbb{R}^n$ konvex, $K_1, K_2 \neq \emptyset$ und $K_1 \cap K_2 = \emptyset$. Dann gibt es ein $c \in \mathbb{R}^n \backslash \{0\}$ mit $c^\top x \leq c^\top y$, für alle $x \in K_1, y \in K_2$. Ist eine der beiden Mengen offen, so gilt sogar $<$.*

Beweis: Es ist $K = K_1 - K_2$ konvex und $0 \notin K$. Gilt $0 \notin \overline{K}$, dann existiert nach Satz 4.6 ein c mit $c^\top x < 0$, für alle $x \in \overline{K}$. Gilt $0 \in \overline{K}$, dann ist 0 Randpunkt von $\overline{K}$ (vgl. Übung). Nach Satz 4.7 gibt es also ein c mit $c^\top x \leq 0$, für alle $x \in \overline{K}$. Nach Definition von K folgt dann die Behauptung.
Ist eine der Mengen offen, so ist K offen. Aus $c^\top x = 0$, für ein $x \in K$, würde dann $c^\top y > 0$ für ein y in der Nähe von x gelten. □

4.4 Der Sattelpunktsatz von Kuhn–Tucker

Im folgenden betrachten wir

Definition 4.3 *Allgemeines Konvexes Optimierungsproblem:*

$$\min\{F(x) \,|\, f_i(x) \leq 0, i = 1, \ldots, m, \; x \geq 0\}, \tag{4.2}$$

für konvexe Funktionen F, $f_1, \ldots, f_m : \mathbb{R}^n \to \mathbb{R}$.

Man beachte, daß die Menge M der zulässigen Punkte konvex ist. Dies ist i.a. nicht mehr der Fall, wenn konvexe Gleichungsrestriktionen hinzugefügt werden. Lineare Gleichungsrestriktionen $Ax - b = 0$ können aber ohne weiteres hinzugenommen werden. Die Nichtnegativitätsbedingung $x \geq 0$ ist keine echte Einschränkung, da man bei sinnvoll gestellten Problemen die Menge der zulässigen Punkte in den ersten Orthanten verschieben kann.

Definition 4.4 *Zu dem konvexen Optimierungsproblem (4.2) mit Zielfunktion F heißt*

$$\Phi(x, u) = F(x) + \sum_{i=1}^{m} u_i f_i(x) = F(x) + u^\top f(x)$$

die Lagrangefunktion *zu (4.2) und die u_i heißen* Lagrange–Multiplikatoren.

Definition 4.5 $\binom{x^0}{u^0} \in \mathbb{R}^{n+m}$ *mit* $x^0 \geq 0$, $u^0 \geq 0$ *heißt* Sattelpunkt *von* $\Phi(x,u)$ *(bezüglich des 1. Orthanten von* $\mathbb{R}^{n+m}$*), wenn für alle* $x \geq 0$, $u \geq 0$ *gilt*

$$\Phi(x^0,u) \leq \Phi(x^0,u^0) \leq \Phi(x,u^0).$$

Lemma 4.5 *Ist* $\binom{x^0}{u^0}$ *ein Sattelpunkt von* $\Phi(x,u)$*, so ist* x^0 *Minimallösung von (4.2).*

Beweis: Nach Voraussetzung gilt für alle $x \geq 0$ und $u \geq 0$

$$F(x^0) + u^\top f(x^0) \leq F(x^0) + u^{0\top} f(x^0) \leq F(x) + u^{0\top} f(x).$$

Also $u^\top f(x^0) \leq u^{0\top} f(x^0)$, für alle $u \geq 0$. Wählt man etwa $u = u^0 + e^i$, so folgt $f(x^0) \leq 0$. Also ist x^0 zulässiger Punkt und wegen $u^0 \geq 0$ folgt $u^{0\top} f(x^0) \leq 0$. Andererseits folgt mit $u = 0$ auch $u^{0\top} f(x^0) \geq 0$, also $u^{0\top} f(x^0) = 0$. Somit gilt für alle $x \geq 0$ zunächst $F(x^0) \leq F(x) + u^{0\top} f(x)$ und für zulässige x folglich $F(x^0) \leq F(x)$. Darum ist x^0 Minimalpunkt. □

Lemma 4.5 gilt für beliebige Funktionen F, f_i. Für die Umkehrung wird jedoch ausgenutzt, daß F und die f_i konvexe Funktionen sind.

Satz 4.9 (Kuhn-Tucker, 1951) *Es gebe ein zulässiges* $x \in \mathbb{R}^n$ *mit* $f_i(x) < 0$*, für* $i = 1, \ldots, m$*. Dann ist* $x^0 \geq 0$ *genau dann Minimallösung von (4.2), wenn es ein* $u^0 \in \mathbb{R}^m$ *mit* $u^0 \geq 0$ *gibt, so daß* $\binom{x^0}{u^0}$ *ein Sattelpunkt von* $\Phi(x,u)$ *ist.*

Beweis: Die eine Richtung gilt nach Lemma 4.5. Sei nun x^0 Minimalpunkt und

$$K_1 = \{y \in \mathbb{R}^{m+1} \,|\, \exists x \geq 0 \text{ mit } y_0 \geq F(x) \text{ und } y_i \geq f_i(x), i = 1, \ldots, m\}$$

und

$$K_2 = \{y \in \mathbb{R}^{m+1} \,|\, y_0 < F(x^0),\ y_i < 0,\ i = 1, \ldots, m\}.$$

K_1 und K_2 sind nichtleer. K_2 ist offen und konvex.
K_1 ist konvex: Seien $v, w \in K_2$ und $\lambda \in (0,1)$. Dann gilt für $z = \lambda v + (1-\lambda)w$

$$z_0 = \lambda v_0 + (1-\lambda)w_0 \geq \lambda F(x^v) + (1-\lambda)F(x^w) \geq F(\lambda x^v + (1-\lambda)x^w)$$

und für $i = 1, \ldots, m$

$$z_i = \lambda v_i + (1-\lambda)w_i \geq \lambda f_i(x^v) + (1-\lambda)f_i(x^w) \geq f_i(\lambda x^v + (1-\lambda)x^w).$$

Ferner gilt $K_1 \cap K_2 = \emptyset$, da x^0 Minimalpunkt. Somit gibt es nach Satz 4.8 ein $c = (c_0, \ldots, c_m)^\top \neq 0$ mit

$$c^\top v > c^\top w, \quad \forall v \in K_1, w \in K_2.$$

Mit $v \in K_1$ ist auch $v + \lambda e^i \in K_1$, für $\lambda \geq 0$ und $i = 0, \ldots, m$. Darum ist $c \geq 0$.
Für $x \geq 0$ ist $v = (F(x), f_1(x), \ldots, f_m(x))^\top \in K_1$ und $w = (F(x^0), 0, \ldots 0)^\top \in \overline{K_2}$, also $c^\top v \geq c^\top w$, d.h.

$$c_0 F(x) + \sum_{i=1}^{m} c_i f_i(x) \geq c_0 F(x^0).$$

Angenommen, $c_0 = 0$. Dann folgt $\sum_{i=1}^m c_i f_i(x) \geq 0$, für alle $x \geq 0$, insbesondere für das x mit $f(x) < 0$. Mit $c \geq 0$ folgt dann $c_1 = \ldots = c_m = 0$, Widerspruch.
Also gilt $c_0 > 0$. Setze

$$u^0 := \frac{1}{c_0}(c_1, \ldots, c_m)^\top.$$

Dann ist also $u^0 \geq 0$. Aus obiger Gleichung folgt

$$F(x) + u^{0\top} f(x) \geq F(x^0), \quad \forall x \geq 0.$$

Insbesondere für $x = x^0$ folgt $u^{0\top} f(x^0) \geq 0$. Wegen $u^0 \geq 0$ und $f(x^0) \leq 0$ folgt dann $u^{0\top} f(x^0) = 0$. Zusammen mit $u^\top f(x^0) \leq 0$, für alle $u \geq 0$, folgt dann für alle $x \geq 0$ und $u \geq 0$

$$F(x^0) + u^\top f(x^0) \leq F(x^0) + u^{0\top} f(x^0) \leq F(x) + u^{0\top} f(x).$$

□

Bemerkungen: (1) Die Voraussetzung im Satz von Kuhn–Tucker heißt *Slater–Bedingung* und ist äquivalent zu der Bedingung: Zu jedem $i = 1, \ldots, m$ gibt es ein zulässiges x^i mit $f_i(x^i) < 0$.
(2) Für das Minimum der Zielfunktion kann man Schranken angeben. Sei x^0 Minimallösung. Ist einerseits x^1 zulässiger Punkt, so gilt $F(x^0) \leq F(x^1)$. Ist andererseits $u^\star \in \mathbb{R}^m, u^\star \geq 0$ fest und x^2 Lösung von $F(x) + u^{\star\top} f(x) \to \min$, für $x \geq 0$ (ohne Nebenbedingungen), so ist

$$F(x^2) + u^{\star\top} f(x^2) \leq F(x^0) + u^{\star\top} f(x^0) \leq F(x^0).$$

(3) Kuhn–Tuckers Satz ist eine Weiterführung der klassischen Lösung von Extremalproblemen mit Nebenbedingungen, wobei jetzt Ungleichungen erlaubt sind.

4.5 Lokale Kuhn–Tucker–Bedingungen

Im Satz von Kuhn–Tucker wird die Lösung der konvexen Optimierungsaufgabe durch eine globale Sattelpunktbedingung charakterisiert. Sind F und die f_i differenzierbar, so kann man die globale Bedingung durch verschiedene lokale Bedingungen ersetzen. Wie im letzten Abschnitt sei $\Phi(x, u) = F(x) + u^\top f(x)$. Ferner setzen wir abkürzend

$$\begin{aligned} \Phi_x &= \left(\frac{\partial \Phi}{\partial x_1}, \ldots, \frac{\partial \Phi}{\partial x_n}\right)^\top \qquad \text{und} \\ \Phi_u &= \left(\frac{\partial \Phi}{\partial u_1}, \ldots, \frac{\partial \Phi}{\partial u_m}\right)^\top = f(x). \end{aligned}$$

Satz 4.10 *Es gebe ein zulässiges x mit $f_i(x) < 0, i = 1, \ldots, m$. $x^0 \geq 0$ ist genau dann Minimallösung von (4.2), wenn es ein $u^0 \geq 0$ gibt mit*

$$\Phi_x(x^0, u^0) \geq 0 \quad , \quad x^{0\top} \Phi_x(x^0, u^0) = 0 \tag{4.3}$$
$$\Phi_u(x^0, u^0) \leq 0 \quad , \quad u^{0\top} \Phi_u(x^0, u^0) = 0. \tag{4.4}$$

Beweis: (1) Es gebe ein $u^0 \geq 0$ mit (4.3) und (4.4). Da F und die f_i konvex sind und $u^0 \geq 0$, ist auch $\Phi(x, u^0)$ konvex. Nach Satz 4.2 gilt dann für alle $x \geq 0$

$$\Phi(x, u^0) \geq \Phi(x^0, u^0) + (x - x^0)^\top \Phi_x(x^0, u^0).$$

Weiter ist $\Phi(x^0, u)$ als Funktion von u linear (affin), d.h.

$$\Phi(x^0, u) = \Phi(x^0, u^0) + (u - u^0)^\top \Phi_u(x^0, u^0).$$

Mit (4.3) und (4.4) folgt $\Phi(x, u^0) \geq \Phi(x^0, u^0)$, für alle $x \geq 0$ und $\Phi(x^0, u) \leq \Phi(x^0, u^0)$, für alle $u \geq 0$, d.h. (x^0, u^0) ist Sattelpunkt.
(2) Sei (x^0, u^0) Sattelpunkt, also

$$\Phi(x^0, u) \leq \Phi(x^0, u^0) \leq \Phi(x, u^0), \quad \forall x \geq 0,\ u \geq 0.$$

Wäre $\frac{\partial \Phi}{\partial x_i} < 0$, für ein i, so wäre $x = x^0 + \epsilon e^i \geq 0$ mit $\Phi(x, u^0) < \Phi(x^0, u^0)$, für ein $\epsilon > 0$. Widerspruch. Also $\Phi_x(x^0, u^0) \geq 0$.
Wegen $x^0 \geq 0$ folgt $x^{0\top} \Phi_x(x^0, u^0) \geq 0$. Angenommen, für ein i gilt $\frac{\partial \Phi}{\partial x_i}(x^0, u^0) > 0$ und $x_i^0 > 0$. Dann ist $x = x^0 - \epsilon e^i \geq 0$ und $\Phi(x, u^0) < \Phi(x^0, u^0)$, für ein $\epsilon > 0$. Widerspruch. Somit gilt $x^{0\top} \Phi_x(x^0, u^0) = 0$.
Analog zeigt man (4.4) für u mit umgekehrten Vorzeichen. □

Ein zulässiger Punkt x von (4.2) erfüllt die Restriktionen $f_i(x) \leq 0,\ i = 1, \ldots, m$ und $g_j(x) := -x_j \leq 0,\ j = 1, \ldots, n$. Die Restriktionen, für die Gleichheit gilt, nennen wir *aktive* Restriktionen.

Satz 4.11 *Es gebe ein zulässiges x mit $f_i(x) < 0, i = 1, \ldots, m$. Ein zulässiger Punkt x^0 ist genau dann Minimallösung von (4.2), wenn $-\operatorname{grad} F(x^0)$ nichtnegative Linearkombination der Gradienten zu den aktiven Restriktionen ist. Gibt es keine aktive Restriktion, so gilt $\operatorname{grad} F(x^0) = 0$.*

Beweis: (1) Es sei $x^0 \geq 0$ Minimallösung, d.h. Φ hat einen Sattelpunkt $\binom{x^0}{u^0}$ mit $x^0 \geq 0$ und $u^0 \geq 0$. Nach Satz 4.10 gilt dann auch (4.3) und (4.4). Es seien P^0 und Q^0 die Menge der aktiven Restriktionen f_i bzw. g_j. Es gilt $\Phi_u(x^0, u^0) = f(x^0)$. Nach (4.4) ist $u^{0\top} f(x^0) = 0$, also kann nicht gleichzeitig $f_i(x^0) < 0$ und $u_i^0 > 0$ sein. Somit gilt $u_i^0 \geq 0$, für $i \in P^0$ und $u_i^0 = 0$, für $i \notin P_0$. Sei

$$v^0 := \Phi_x(x^0, u^0) = \operatorname{grad} F(x^0) + \sum_{i=1}^{m} u_i^0 \operatorname{grad} f_i(x^0).$$

Nach (4.3) in Satz 4.10 ist $v^0 \geq 0$ und $x^{0\top} v^0 = 0$. Somit kann nicht gleichzeitig $x_j^0 > 0$ und $v_j^0 > 0$ sein. Also $v_j^0 \geq 0$, für $j \in Q^0$ und $v_j^0 = 0$, für $j \notin Q^0$.
Mit der Darstellung

$$v^0 = -\sum_{j=1}^{n} v_j^0 \operatorname{grad} g_j(x^0) = -\sum_{j \in Q^0} v_j^0 \operatorname{grad} g_j(x^0)$$

wird

$$-\operatorname{grad} F(x^0) = \sum_{i \in P^0} u_i^0 \operatorname{grad} f_i(x^0) + \sum_{j \in Q^0} v_j^0 \operatorname{grad} g_j(x^0),$$

d.h. $-\operatorname{grad} F(x^0)$ ist nichtnegative Linearkombination der Gradienten zu aktiven Restriktionen. Insbesondere ist $\operatorname{grad} F(x^0) = 0$, wenn es keine aktiven Restriktionen gibt.
(2) Sei $-\operatorname{grad} F(x^0)$ nichtnegative Linearkombination der zu den aktiven Restriktionen gehörenden Gradienten, d.h. es gelte

$$-\operatorname{grad} F(x^0) = \sum_{i \in P^0} \alpha_i \operatorname{grad} f_i(x^0) + \sum_{j \in Q^0} \beta_j \operatorname{grad} g_j(x^0),$$

mit $\alpha_i, \beta_j \geq 0$. Setze $u_i^0 = \alpha_i$, für $i \in P^0$, und $u_i^0 = 0$, sonst und analog sei $v_j^0 = \beta_j$, für $j \in Q^0$, und $v_j^0 = 0$, sonst. Dann folgt $u^0 \geq 0, v^0 \geq 0, u^{0\top} f(x^0) = 0$ und $x^{0\top} v^0 = 0$. Ferner gilt

$$v^0 = \operatorname{grad} F(x^0) + \sum_{i=1}^{m} u_i^0 \operatorname{grad} f_i(x^0) = \Phi_x(x^0, u^0).$$

Also gelten mit $\Phi_u(x^0, u^0) = f(x^0) \leq 0$ genau die Bedingungen (4.3) und (4.4) aus Satz 4.10. (Im Falle $P^0 = Q^0 = \emptyset$ gilt dies alles mit $\operatorname{grad} F(x^0) = 0, u^0 = 0, v^0 = 0$.) $\square$

Die Menge der Minimallösungen läßt sich wie folgt charakterisieren.

Satz 4.12 *Ist x^0 Minimallösung von (4.2), so ist die Menge der Minimallösungen die Menge der zulässigen z mit*

$$\operatorname{grad} F(z) = \operatorname{grad} F(x^0) \quad \textit{und} \quad (z - x^0)^\top \operatorname{grad} F(x^0) = 0. \tag{4.5}$$

Beweis: (1) Sei z zulässig und es gelte $\operatorname{grad} F(z) = \operatorname{grad} F(x^0)$ und $(z - x^0)^\top \operatorname{grad} F(x^0) = 0$. Da F konvex, gilt nach Satz 4.2

$$F(x^0) \geq F(z) + (x^0 - z)^\top \operatorname{grad} F(z) \geq F(z),$$

also ist z Minimallösung.
(2) Sei z Minimallösung. Dann ist die Funktion $g(\lambda) = F(x^0 + \lambda(z - x^0))$ auf $[0,1]$ konstant, also nach der Kettenregel

$$(z - x^0)^\top \operatorname{grad} F(x^0) = \frac{dg}{d\lambda}(0) = 0.$$

Setze

$$H(y) = F(y) - (y - x^0)^\top \operatorname{grad} F(x^0).$$

Dann ist H konvex und $H(z) = F(z) = F(x^0) = H(x^0)$ und $\operatorname{grad} H(z) = \operatorname{grad} F(z) - \operatorname{grad} F(x^0)$. Nach Satz 4.2 für F (mit Basis x^0) eingesetzt in der Definition von H ergibt sich

$$H(y) \geq F(x^0) = H(z), \quad \forall y.$$

Somit hat H in z ein absolutes Minimum und es folgt $\operatorname{grad} H(z) = 0$, also $\operatorname{grad} F(z) = \operatorname{grad} F(x^0)$. $\square$

4.6 Quadratische Optimierung

Unter den Verfahren der konvexen Optimierung nimmt die quadratische Optimierung eine Sonderstellung ein, weil es im wesentlichen auf das Simplexverfahren zurückgeführt werden kann.

Definition 4.6 *Es sei $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $p \in \mathbb{R}^n$ und $C \in \mathbb{R}^{m\times m}$ eine symmetrische positiv semidefinite Matrix. Dann lautet das Quadratische Optimierungsproblem*

$$\min\{Q(x) = p^\top x + x^\top Cx \mid Ax \leq b,\, x \geq 0\}. \tag{4.6}$$

In Anlehnung an (4.2) sind hier die linearen Restriktionen in Form von Ungleichungen gegeben. Natürlich kann man das Problem äquivalent in der Standardform formulieren.

Satz 4.13 *Sei $F : \mathbb{R}^n \to \mathbb{R}$ differenzierbar und $f(x) = Ax - b$, d.h. es treten nur lineare Restriktionen auf. Dann kann die Slater–Bedingung in den Sätzen 4.9, 4.10 und 4.11 entfallen.*

Beweis: Die Sätze 4.10 und 4.11 geben nur äquivalente Umschreibungen der Sattelpunktbedingung. Ferner wurde die Slater–Bedingung in Satz 4.9 nur in einer Richtung benötigt. Es reicht also die Notwendigkeit der Bedingung in Satz 4.11 zu zeigen.
Sei also x^0 Minimallösung. Ferner seien $a^{i_1}, \ldots, a^{i_p}, e^{j_1}, \ldots e^{j_q}$ die Vektoren zu den aktiven Restriktionen zu x^0, d. h. $a^{i_k\top}x^0 = b_{i_k}$ bzw. $x^0_{j_k} = 0$, und

$$B = \left(-a^{i_1}, \ldots, -a^{i_p}, e^{j_1}, \ldots, e^{j_q}\right) \in \mathbb{R}^{n\times(p+q)}.$$

Sei $y^\top B \geq 0$, d. h. $y^\top a^{i_k} \leq 0$ und $y^\top e^{j_k} \geq 0$. Dann ist $x^0 + \lambda y$ für hinreichend kleine $\lambda > 0$ zulässig und somit $y^\top \operatorname{grad} F(x^0) \geq 0$. Nach Satz 2.5 existiert dann ein $z = (\alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_q) \geq 0$ mit $Bz = \operatorname{grad} F(x^0)$, d.h.

$$\operatorname{grad} F(x^0) = \sum_{i=1}^{p} -\alpha_i a^i + \sum_{j=1}^{q} \beta_j e^j.$$

Damit ist $-\operatorname{grad} F(x^0)$ nichtnegative Linearkombination der Gradienten zu den aktiven Restriktionen.

□

Bemerkung: Man kann zeigen, daß für Satz 4.13 die Bedingung F differenzierbar weggelassen werden kann. Treten sowohl lineare als auch nichtlineare Restriktionen auf, so gilt der Satz von Kuhn–Tucker, wenn die Slater–Bedingung für die nichtlinearen Restriktionen erfüllt ist.

Die Anwendung auf das quadratische Optimierungsproblem ergibt

Satz 4.14 *$x^0 \geq 0$ ist genau dann Lösung von (4.6), wenn gilt*
Es gibt $u^0 \in \mathbb{R}^m$, $v^0 \in \mathbb{R}^n, y^0 \in \mathbb{R}^m$ mit

$$u^0 \geq 0, v^0 \geq 0, y^0 \geq 0, \quad Ax^0 + y^0 = b, \quad v^0 - 2Cx^0 - A^\top u^0 = p \tag{4.7}$$

und

$$x^{0\top}v^0 + u^{0\top}y^0 = 0. \tag{4.8}$$

Beweis: Es ist $\Phi(x,u) = p^\top x + x^\top Cx + u^\top(Ax-b)$. Setze

$$v = \Phi_x = p + 2Cx + A^\top u, \quad y = -\Phi_u = -Ax + b.$$

Nach Satz 4.10 bzw. Satz 4.13 ist x^0 Minimallösung genau dann, wenn es ein $u^0 \geq 0$ gibt mit

$$\begin{aligned} v^0 = \Phi_x(x^0,u^0) &\geq 0, \qquad & x^{0\top}v^0 &= 0 \\ y^0 = -\Phi_u(x^0,u^0) &\geq 0, \qquad & u^{0\top}y^0 &= 0. \end{aligned}$$

Da in $x^{0\top}v^0$ und $u^{0\top}y^0$ nur nichtnegative Summanden stehen, ist $x^{0\top}v^0 = 0$ und $u^{0\top}y^0 = 0$ äquivalent zu $x^{0\top}v^0 + u^{0\top}y^0 = 0$. □

Man beachte, daß (4.8) die einzige nichtlineare Bedingung ist. Sie gibt zugleich eine Fehlerabschätzung ab:

Satz 4.15 (Einschließungssatz) *Sei $x^0 \geq 0, v^0, u^0, y^0$ Lösung von (4.7) und x^1 sei Minimallösung von (4.6). Dann gilt*

$$Q(x^0) - (x^{0\top}v^0 + y^{0\top}u^0) \leq Q(x^1) \leq Q(x^0).$$

Beweis: Sei $x^0 \geq 0, v^0 \geq 0, u^0 \geq 0, y^0 \geq 0$ Lösung von (4.7). Wegen $y^0 \geq 0$ ist $Ax^0 \leq b$. Also ist x^0 zulässig und die rechte Ungleichung folgt. Da Q konvex, folgt aus Satz 4.2 für zulässiges x

$$\begin{aligned} Q(x) - Q(x^0) &\geq (x-x^0)^\top \operatorname{grad} Q(x^0) = (x-x^0)^\top(p + 2Cx^0) \\ &= (x-x^0)^\top(v^0 - A^\top u^0) \\ &= x^\top v^0 - x^{0\top}v^0 - (Ax-b)^\top u^0 + (Ax^0-b)^\top u^0 \\ &\geq -x^{0\top}v^0 - y^{0\top}u^0. \end{aligned}$$

Mit $x = x^1$ folgt die Behauptung. □

Beispiel 4.1 Gesucht ist $x = (x_1, x_2)^\top \in \mathbb{R}^2$ mit $x_1, x_2 \geq 0,\ x_1 + x_2 \leq 8,\ x_1 \leq 6,\ x_1 + 3x_2 \leq 18$ und minimalem $Q(x) = 2x_1^2 + x_2^2 - 48x_1 - 40x_2$. Also

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 3 \end{pmatrix}, \quad C = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 8 \\ 6 \\ 18 \end{pmatrix}, \quad p = \begin{pmatrix} -48 \\ -40 \end{pmatrix}.$$

Wählt man mit $x^0 = (3,5)^\top$ eine Ecke des zulässigen Polyeders, so wird $y^0 = b - Ax^0 = (0,3,0)^\top$. Mit $v^0 = 0$ findet man als eine nichtnegative Lösung von $v^0 - 2Cx^0 - A^\top u^0 = p$ den Vektor $u^0 = (30,6,0)^\top$. Mit $Q(x^0) = -301$ folgt aus Satz 4.15 folgende Einschließung für das Minimum $Q(x^1)$: $-319 \leq Q(x^1) \leq -301$. Minimallösung ist, wie im nächsten Abschnitt gezeigt wird oder wie aus Einsetzen in Satz 4.14 hervorgeht: $x^1 = (4,4)^\top$ mit $Q(x^1) = -304$.

Mit Satz 4.12 kann die Menge der Minimallösungen charakterisiert werden.

Satz 4.16 *Sei M die Menge der zulässigen Punkte für (4.6). Mit $x^0 \in M$ ist $x^0 + y \in M$ genau dann Minimallösung, wenn $Cy = 0$ und $p^\top y = 0$. Ist C positiv definit, so ist $y = 0$ die einzige Lösung.*

Beweis: Sei x^0 Minimallösung. Nach Satz 4.12 ist dann $x^0 + y$ genau dann Minimallösung, wenn $\operatorname{grad} Q(x^0) = \operatorname{grad} Q(x^0 + y)$ und $y^\top \operatorname{grad} Q(x^0) = 0$. Mit $\operatorname{grad} Q(x) = p + 2Cx$ ist dies ist äquivalent zu $Cy = 0$ und $p^\top y + 2y^\top C x^0 = 0$. Da C symmetrisch, folgt die Behauptung. Ist C positiv definit, dann ist $Cy = 0$ genau dann, wenn $y = 0$. $\square$

Die Menge der Minimallösungen ist also der Durchschnitt von M mit einem affinen Unterraum.

Falls die Restriktionen als Gleichungen gegeben sind, wird (4.6) zu

$$\min\{Q(x) = p^\top x + x^\top C x \mid Ax = b,\, x \geq 0\}. \tag{4.9}$$

Anstelle von Satz 4.14 gilt dann

Satz 4.17 *$x^0 \geq 0$ ist genau dann Lösung von (4.9), wenn $Ax^0 = b$ gilt und es $u^0 \in \mathbb{R}^m$, $v^0 \in \mathbb{R}^n$ gibt mit*

$$v^0 - 2Cx^0 - A^\top u^0 = p, \quad v^0 \geq 0,\ (u^0 \textit{ beliebig}), \quad x^{0\top} v^0 = 0. \tag{4.10}$$

Beweis: Wende Satz 4.14 auf $\begin{pmatrix} A \\ -A \end{pmatrix} x \leq \begin{pmatrix} b \\ -b \end{pmatrix}$ an: x^0 ist genau dann Minimallösung, wenn es $u', u'', y', y'' \in \mathbb{R}^m$ und ein $v^0 \in \mathbb{R}^n$ gibt mit $u', u'', y', y'' \geq 0, v^0 \geq 0$ und

$$\begin{pmatrix} A \\ -A \end{pmatrix} x^0 + \begin{pmatrix} y' \\ y'' \end{pmatrix} = \begin{pmatrix} b \\ -b \end{pmatrix}, \quad v^0 - 2Cx^0 - (A^\top, -A^\top) \begin{pmatrix} u' \\ u'' \end{pmatrix} = p$$

und

$$x^{0\top} v^0 + (u'^\top, u''^\top) \begin{pmatrix} y' \\ y'' \end{pmatrix} = 0.$$

Notwendigerweise ist $y' = -y''$, also sind beide 0. Setzt man $u^0 = u' - u''$ (nun also ohne Vorzeichenrestriktion), so sind die Bedingungen äquivalent zu $Ax^0 = b$ und denen in (4.10). $\square$

4.7 Das Verfahren von Wolfe

Unter den vielen Lösungsalgorithmen für quadratische Optimierung ist das bekannteste das von Wolfe (1959), das eine Variante des Simplexverfahrens mit einer Zusatzvorschrift ist, und das aus der Kuhn–Tucker–Theorie durch Spezialisierung folgt.
Ausgangspunkt des Verfahrens von Wolfe ist das Problem (4.9) mit positiv semidefiniter symmetrischer Matrix C. Wir setzen voraus, daß das Problem nicht unbeschränkt ist. Dies ist insbesondere der Fall, wenn C positiv definit ist oder wenn C positiv semidefinit und $p = 0$. Nach Satz 4.17 ist $x \geq 0$ genau dann Minimallösung von (4.9), wenn (ohne Indizes)

$$Ax = b, \quad -2Cx + v - A^\top u = p, \quad x \geq 0,\ v \geq 0,\ u \in \mathbb{R}^m, \quad x^\top v = 0. \tag{4.11}$$

In einem ersten Schritt sucht man einen zulässigen Punkt $\overline{x}$ von (4.9), d.h. mit $A\overline{x} = b, \overline{x} \geq 0$. Dies kann wie üblich durch Erraten oder durch die Phase–I im Simplexverfahren geschehen. Gibt es keinen zulässigen Punkt, so ist (4.9) nicht lösbar, ansonsten erhält man eine Ecke $\overline{x}$ des zulässigen Polyeders. Nach Voraussetzung gibt es dann eine Lösung von (4.9) und damit von (4.11). Zur Lösung von (4.11) setzt man $h = p + 2C\overline{x}$ und löst das Hilfsproblem

$$\min\{\zeta \mid Ax = b,\ -2Cx + v - A^\top u + h\zeta = p,\ x \geq 0,\ v \geq 0,\ u \in \mathbb{R}^m,\ \zeta \geq 0, \quad x^\top v = 0\}. \tag{4.12}$$

Wegen $x^\top v = 0$ ist das kein lineares Problem. Man kann es jedoch deuten als lineares Programm für den $(2n+m+1)$–dimensionalen Vektor $(x^\top, v^\top, u^\top, \zeta)$, mit der zusätzlichen Bedingung, daß zu jedem $i = 1, \ldots, n$ immer $x_i = 0$ oder $v_i = 0$ ist, da sonst $x^\top v > 0$. Das Problem (4.12) ist so konstruiert, daß $x = \overline{x}, v = 0, u = 0, \zeta = 1$ zulässiger Punkt ist und für $\zeta = 0$ (4.11) dasteht. Die Matrix zu (4.12) ist $\begin{pmatrix} A & 0 & 0 & 0 \\ -2C & I & -A^\top & h \end{pmatrix}$, d.h. eine $(2n + m + 1) \times (m + n)$–Matrix vom Rang $n + m$. Folgende $(m + n)$ Spaltenvektoren sind linear unabhängig.

(a) Die m Spaltenvektoren von A, die zu der Basis B von $\overline{x}$ gehören, ergänzt durch die entsprechenden Spalten von $-2C$.

(b) Alle Spaltenvektoren von $\begin{pmatrix} 0 \\ -A^\top \end{pmatrix}$ (beachte $\text{rank}(A) = \text{rank}(A^\top) = m$).

(c) Die $n - m$ Spaltenvektoren von $\begin{pmatrix} 0 \\ I \end{pmatrix}$ zu Komponenten v_i mit $i \notin B$.

(Die Vektoren aus (b) und (c) sind linear unabhängig nach Definition der Basis bzw. der Ecke $\overline{x}$.) Diese $(m + n)$ linear unabhängigen Spalten sind noch keine Basis zum Startpunkt, da $\zeta = 1$. Darum muß einer der Vektoren zu den $u_i = 0$ oder $v_i = 0$ durch den Vektor $\begin{pmatrix} 0 \\ h \end{pmatrix}$ ersetzt werden, und zwar so, daß die Menge linear unabhängig bleibt (es sei o.E. $h \neq 0$, sonst ist man fertig). Nun löst man (4.12) als lineares Programm wie beim Simplexverfahren, allerdings mit der Zusatzvorschrift

$$\begin{array}{c}\text{Verbleibt bei einem Austauschschritt } x_i \text{ (bzw. } v_i\text{) in der Basis,} \\ \text{so darf } v_i \text{ (bzw. } x_i\text{) nicht in die Basis hineingenommen werden.}\end{array} \tag{4.13}$$

Dann ist immer $x^\top v = 0$ erfüllt. Das Verfahren endet, wenn $\zeta = 0$ erreicht wird, oder wenn bei $\zeta > 0$ kein zulässiger Austauschschritt mehr möglich ist.

Lemma 4.6 *Sei C positiv semidefinit und $x^\top Cx = 0$. Dann gilt $Cx = 0$.*

Beweis: Für $y \in \mathbb{R}^n$ beliebig gilt

$$0 \leq (x + \lambda y)^\top C(x + \lambda y) = x^\top Cx + 2\lambda y^\top Cx + \lambda^2 y^\top Cy =: \varphi(\lambda).$$

Wegen $\varphi(0) = 0$ ist 0 minimal und daher $\varphi'(0) = 0$. Also $y^\top Cx = 0$, für alle $y \in \mathbb{R}^n$ und damit $Cx = 0$.

□

Satz 4.18 *Das Simplexverfahren mit der Zusatzvorschrift (4.13) führt zur Minimallösung $\zeta = 0$ von (4.12), wenn eine der drei folgenden Bedingungen erfüllt ist:*

(a) C ist positiv definit.

(b) C ist positiv semidefinit und $p = 0$.

(c) Ein Schritt, der zu $\zeta < 1$ führt, ist möglich.

Beweis: Angenommen das Verfahren bricht mit $\hat{\zeta} > 0$ ab. Es sei also in $\hat{x}, \hat{v}, \hat{u}, \hat{\zeta}$ kein erlaubter Pivotschritt mehr möglich. Dann hat das LP

$$\begin{array}{rcl} \zeta \to \min & & \\ Ax & = & b \\ -2Cx + v - A^\top u + h\zeta & = & p \\ \hat{v}^\top x + \hat{x}^\top v & = & 0 \\ x \geq 0, v \geq 0, u \in \mathbb{R}^m, \zeta \geq 0 & & \end{array}$$

in $\hat{x} \geq 0, \hat{v} \geq 0, \hat{u}, \hat{\zeta}$ eine optimale Lösung:
Nach jedem Pivotschritt muß für ein i sowohl x_i als auch v_i in der Basis sein. Sei o.E. vorher x_i in der Basis. Da die relativen Kosten zu v_i negativ sind, muß $\hat{x}_i > 0$ sein, da sonst der Tausch v_i gegen x_i erlaubt wäre (auch bei negativem Pivotelement). Wegen der zusätzlichen Voraussetzung kommt also v_i mit Wert 0 in die Basis und die Zielfunktion bleibt unverändert. Somit verbessert kein Pivotschritt die Zielfunktion.
Das hierzu duale LP

$$\begin{array}{rcl} b^\top y + p^\top w \to \max & & \\ A^\top y - 2Cw + \hat{v}\xi & \leq & 0 \\ w + \hat{x}\xi & \leq & 0 \\ Aw & = & 0 \\ h^\top w & \leq & 1 \\ y \in \mathbb{R}^m, w \in \mathbb{R}^n, \xi \in \mathbb{R} & & \end{array}$$

hat eine Lösung $\hat{y}, \hat{w}, \hat{\xi}$ mit

$$b^\top \hat{y} + p^\top \hat{w} = \hat{\zeta} > 0.$$

Wegen $\hat{\zeta} > 0$ ist nach dem komplementären Schlupf $h^\top \hat{w} = 1$.
Wegen $\hat{x}^\top \hat{v} = 0$ tritt für jedes $i = 1, \ldots, n$ genau einer der drei folgenden Fälle auf:

1. $\hat{x}_i > 0, \hat{v}_i = 0$, dann ist $(A^\top \hat{y} - 2C\hat{w})_i = 0$,

2. $\hat{x}_i = 0, \hat{v}_i > 0$, dann ist $\hat{w}_i = 0$,

3. $\hat{x}_i = \hat{v}_i = 0$, dann ist $(A^\top \hat{y} - 2C\hat{w})_i \leq 0$ und $\hat{w}_i \leq 0$,

(jeweils mit komplementärem Schlupf). Also gilt $\hat{w}^\top(A^\top\hat{y} - 2C\hat{w}) \geq 0$.
Wegen $A\hat{w} = 0$ folgt $\hat{w}^\top C\hat{w} \leq 0$.
Ist C positiv definit, dann folgt $\hat{w} = 0$ im Widerspruch zu $h^\top\hat{w} = 1$. Also folgt (a).
Ist C positiv semidefinit, so folgt nur $C\hat{w} = 0$. Dann ist $1 = h^\top\hat{w} = (p + 2C\overline{x})^\top\hat{w} = p^\top\hat{w}$, im Widerspruch zu $p = 0$. Also folgt (b).
Schließlich ist $b^\top\hat{y} = \hat{x}^\top A^\top\hat{y} = 0$, denn mit $\hat{x}_i > 0$ ist $\hat{v}_i = 0$ und mit komplementärem Schlupf dann $(A^\top\hat{y})_i = 0$. Es folgt $\hat{\zeta} = 1$ und damit (c). □

Algorithmus 4.1
Procedure Wolfe ;
begin
  Man finde mit Phase–I eine Ecke $\overline{x}$ von $Ax = b, x \geq 0$;
  Setze $h = p + 2C\overline{x}$;
  Man konstruiere eine Basis zur Lösung $(\overline{x}, 0, 0, 1)$ von (4.12);
  Man löse (4.12) mit dem Simplexverfahren und der Vorschrift (4.13).
end.

Beispiel 4.2 Das Besipiel 4.1 wird nach Einführung von Schlupfvariablen zu

$$\begin{aligned} 2x_1^2 + x_2^2 - 48x_1 - 40x_2 &\rightarrow \min \\ x_1 + x_2 + x_3 &= 8, \\ x_1 + x_4 &= 6, \\ x_1 + 3x_2 + x_5 &= 18, \\ x &\geq 0. \end{aligned}$$

Jetzt ist also $C = \operatorname{diag}(2, 1, 0, 0, 0)$. Startecke ist $\overline{x} = (0, 0, 8, 6, 18)^\top$. Dann ist $h = p + 2C\overline{x} = (-48, -40, 0, 0, 0)^\top$. Das zweite Gleichungssystem in (4.12) wird zu

$$\begin{aligned} -4x_1 + v_1 - u_1 - u_2 - u_3 - 48\zeta &= -48 \\ -2x_2 + v_2 - u_1 - 3u_3 - 40\zeta &= -40 \\ v_3 - u_1 &= 0 \\ v_4 - u_2 &= 0 \\ v_5 - u_3 &= 0 \end{aligned}$$

Die letzten 3 Gleichungen sind so gebaut, daß die nicht vorzeichenbeschränkten u_i eliminiert werden können. Dadurch reduziert sich das Gleichungssystem auf 5 Gleichungen. Das Problem hat dann Standardform und man erhält folgendes Tableau:

		x_1	x_2	x_3	x_4	x_5	v_1	v_2	v_3	v_4	v_5	ζ
z	0	0	0	0	0	0	0	0	0	0	0	1
x_3	8	1	1	1	0	0	0	0	0	0	0	0
x_4	6	1	0	0	1	0	0	0	0	0	0	0
x_5	18	1	3	0	0	1	0	0	0	0	0	0
v_1	−48	−4	0	0	0	0	1	0	−1	−1	−1	−48
v_2	−40	0	−2	0	0	0	0	1	−1	0	−3	−40

Durch einen Pivotschritt (negatives Pivotelement) wird v_2 durch ζ ausgetauscht und man erhält das Basistableau

		x_1	x_2	x_3	x_4	x_5	v_1	v_2	v_3	v_4	v_5	ζ
z	-1	0	$-1/20$	0	0	0	0	$1/40$	$-1/40$	0	$-3/40$	0
x_3	8	1	1	1	0	0	0	0	0	0	0	0
x_4	6	1	0	0	1	0	0	0	0	0	0	0
x_5	18	1	3	0	0	1	0	0	0	0	0	0
v_1	0	-4	$12/5$	0	0	0	1	$-6/5$	$1/5$	-1	$13/5$	0
ζ	1	0	$1/20$	0	0	0	0	$-1/40$	$1/40$	0	$3/40$	1

Da v_2 nicht in der Basis ist, kann x_2 hineingenommen werden.

		x_1	x_2	x_3	x_4	x_5	v_1	v_2	v_3	v_4	v_5	ζ
z	-1	$-1/12$	0	0	0	0	$1/48$	0	$-1/48$	$-1/48$	$-1/48$	0
x_3	8	$8/3$	0	1	0	0	$-5/12$	$1/2$	$-1/12$	$5/12$	$-13/12$	0
x_4	6	1	0	0	1	0	0	0	0	0	0	0
x_5	18	6	0	0	0	1	$-5/4$	$3/2$	$-1/4$	$5/4$	$-13/4$	0
x_2	0	$-5/3$	1	0	0	0	$5/12$	$-1/2$	$1/12$	$-5/12$	$13/12$	0
ζ	1	$1/12$	0	0	0	0	$-1/48$	0	$1/48$	$1/48$	$1/48$	1

v_1 ist nicht in der Basis, also ist x_1 erlaubt.

		x_1	x_2	x_3	x_4	x_5	v_1	v_2	v_3	v_4	v_5	ζ
z	$-3/4$	0	0	$1/32$	0	0	$1/128$	$1/64$	$-3/128$	$-1/128$	$-7/128$	0
x_1	3	1	0	$3/8$	0	0	$-5/32$	$3/16$	$-1/32$	$5/32$	$-13/32$	0
x_4	3	0	0	$-3/8$	1	0	$5/32$	$-3/16$	$1/32$	$-5/32$	$13/32$	0
x_5	0	0	0	$-9/4$	0	1	$-5/16$	$3/8$	$-1/16$	$5/16$	$-13/16$	0
x_2	5	0	1	$5/8$	0	0	$5/32$	$-3/16$	$1/32$	$-5/32$	$13/32$	0
ζ	$3/4$	0	0	$-1/32$	0	0	$-1/128$	$-1/64$	$3/128$	$1/128$	$7/128$	1

Schließlich darf man v_3 in die Basis nehmen und man erhält

		x_1	x_2	x_3	x_4	x_5	v_1	v_2	v_3	v_4	v_5	ζ
z	0	0	0	0	0	0	0	0	0	0	0	1
x_1	4	1	0	$1/3$	0	0	$-1/6$	$1/6$	0	$1/6$	$-1/3$	$4/3$
x_4	2	0	0	$-1/3$	1	0	$1/6$	$-1/6$	0	$-1/6$	$1/3$	$-4/3$
x_5	2	0	0	$-7/3$	0	1	$-1/3$	$1/3$	0	$1/3$	$-2/3$	$8/3$
x_2	4	0	1	$2/3$	0	0	$1/6$	$-1/6$	0	$-1/6$	$1/3$	$-4/3$
v_3	32	0	0	$-4/3$	0	0	$-1/3$	$-2/3$	1	$1/3$	$7/3$	$128/3$

Wegen $\zeta = 0$ ist man bei der Lösung der Aufgabe angelangt. Es wird $x_1 = 4, x_2 = 4$ und $Q_{\min} = -304$.

4.8 Ein Verfahren zulässiger Richtungen

In diesem und dem nächsten Abschnitt werden zwei häufig benutzte iterative Verfahrenstypen beschrieben, die allgemein für konvexe Optimierung brauchbar sind. Für die quadratische Optimierung sind diese nur mittelmäßig, da sie im allgemeinen nicht nach endlich vielen Schritten abbrechen.
Bei dem Verfahren der zulässigen Richtungen wird folgendes Problem betrachtet.

$$\min\{F(x) \mid Ax = b,\ x \geq 0\}. \tag{4.14}$$

Dabei ist F konvex und stetig differenzierbar. Ferner wird vorausgesetzt, daß die Menge $M = \{x \in \mathbb{R}^n \mid Ax = b,\ x \geq 0\}$ nichtleer und beschränkt ist.

Definition 4.7 *Sei $x \in M$. Dann heißt $s \in \mathbb{R}^n$* zulässige Richtung *in x, wenn*

$$\max\{\lambda \mid x + \lambda s \in M\} > 0 \quad \textit{und} \quad s^\top \operatorname{grad} F(x) < 0.$$

Satz 4.19 *$x \in M$ ist genau dann Lösung von (4.14), wenn es in x keine zulässigen Richtungen gibt.*

Beweis: (1) Es gebe in x keine zulässigen Richtungen. Dann gilt für beliebiges $y \in M, y \neq x$ und $s = y - x$: $\max\{\lambda | x + \lambda s \in M\} \geq 1$ und $s^\top \operatorname{grad} F(x) \geq 0$. Nach Satz 4.2 ist dann

$$F(y) - F(x) \geq s^\top \operatorname{grad} F(x) \geq 0,$$

also $F(y) \geq F(x)$. Da $y \in M$ beliebig war, ist x Minimalpunkt.
(2) Sei x Minimallösung und s zulässige Richtung in x. Dann ist die Funktion $\varphi(\lambda) = F(x + \lambda s)$ für ein Intervall $[0, \delta]$ definiert und ihre Ableitung in 0 ist nach der Kettenregel $\varphi'(0) = s^\top \operatorname{grad} F(x) < 0$. Für kleine $\lambda > 0$ ist also $x + \lambda s$ zulässig mit $F(x + \lambda s) < F(x)$, d.h. x ist kein Minimalpunkt.
$\square$

Das Verfahren verläuft so:
Man sucht einen (zulässigen) Startpunkt $x^0 \in M$ und eine zulässige Richtung s^0 in x^0. Gibt es kein solches s^0, so ist x^0 nach Satz 4.19 Minimalpunkt von (4.14). Andernfalls bestimmt man λ_0 aus $\min\{F(x^0 + \lambda s^0) \mid \lambda \geq 0, x^0 + \lambda s^0 \in M\}$. Mit $x^1 = x^0 + \lambda_0 s^0$ hat man den ersten Iterationspunkt usw.
Allgemein bestimmt man λ_k mit $F(x^k + \lambda_k s^k) = \min\{F(x^k + \lambda s^k) \mid \lambda \geq 0, x^k + \lambda s^k \in M\}$ und erhält $x^{k+1} = x^k + \lambda_k s^k$.
D.h. man bestimmt eine Folge von Näherungslösungen von (4.14), indem man eine Folge von Minimierungsaufgaben in einer Variablen löst. Aus dem Beweis von Satz 4.19 folgt, daß die Folge der Zielfunktionswerte streng monoton fällt. Weil F stetig und M kompakt ist, konvergiert das Verfahren, jedoch nicht unbedingt gegen den Minimalwert von (4.14), sofern man die zulässigen Richtungen nicht einschränkt.

Beispiel 4.3 Minimiere $F(x) = x_1 + x_2$ unter $x_1 + x_2 \leq 2,\ x_1, x_2 \geq 0$. Sei $x^1 = (2, 0)^\top$ und $s^k = (-1 - 1/k, 1 + 1/(k+1))^\top$, für k ungerade und $s^k = (1 + 1/(k+1), -1 - 1/k)^\top$, für k gerade. Dann ist $x^k = (1 + 1/k, 0)^\top$, für k ungerade und $x^k = (0, 1 + 1/k)^\top$, für k gerade. Mit $\operatorname{grad} F = (1, 1)^\top$ ist $s^{k\top} \operatorname{grad} F(x^k) = 1/(k+1) - 1/k < 0$, also sind alle s^k zulässig. Trotzdem konvergiert $F(x^k) = 1 + 1/k$ gegen 1, obwohl $F_{\min} = 0$.

Daher müssen die zulässigen Richtungen eingeschränkt werden. Eine (von vielen) Vorschriften ist gegeben durch

Lemma 4.7 *Es sei y^k Lösung des linearen Programms* $\min\{x^\top \operatorname{grad} F(x^k) \mid Ax = b, x \geq 0\}$. *Dann gilt*

(a) Ist $(y^k - x^k)^\top \operatorname{grad} F(x^k) = 0$, so ist x^k Lösung von (4.14).

(b) Andernfalls ist $s^k = y^k - x^k$ zulässige Richtung.

Beweis: (a) Nach Definition von y^k gilt $(y^k - x)^\top \operatorname{grad} F(x^k) \leq 0$, für alle $x \in M$. Also gilt nach Satz 4.2 für alle $x \in M$

$$F(x) - F(x^k) \geq (x - x^k)^\top \operatorname{grad} F(x^k) \geq (y^k - x^k)^\top \operatorname{grad} F(x^k) = 0.$$

Darum ist x^k Minimalpunkt.
(b) Da $x^k \in M$, folgt nach Definition von y^k, daß $(y^k - x^k)^\top \operatorname{grad} F(x^k) < 0$. Wegen $x^k, y^k \in M$ ist $s^k = y^k - x^k$ dann also zulässige Richtung. □

Satz 4.20 *Werden die zulässigen Richtungen wie in Lemma 4.7 bestimmt, so konvergiert die Folge $\{F(x^k)\}$ gegen den Minimalwert von (4.14) und jede konvergente Teilfolge von $\{x^k\}$ konvergiert gegen eine Lösung.*

Beweis: Bricht das Verfahren nach endlichen vielen Schritten ab, so folgt die Behauptung aus Lemma 4.7. Ist das Verfahren nicht endlich, so ist $F(x^k)$ eine streng monoton fallende Folge, die nach Voraussetzung beschränkt ist. Also konvergiert sie gegen einen Wert γ. Sei α der Minimalwert von (4.14), dann gilt $\alpha \leq \gamma$.
Angenommen $\alpha < \gamma$. Die in Lemma 4.7 ermittelten y^k sind als Lösungen eines linearen Programms Ecken von M. Da M nur endlich viele Ecken hat, gibt es eine Ecke y^*, so daß $y^{k_j} = y^*$, für eine Teilfolge gilt. Da M kompakt ist, hat x^{k_j} eine konvergente Teilfolge, o.E. sei $x^* = \lim x^{k_j} \in M$ (mit $F(x^*) = \gamma$, da F stetig). Es sei $\overline{x}$ eine Lösung von (4.14), also $F(\overline{x}) = \alpha$. Es gilt nach Definition von y^k: $(y^{k_j} - \overline{x})^\top \operatorname{grad} F(x^{k_j}) \leq 0$. Durch Grenzübergang $(j \to \infty)$ und wegen der Stetigkeit von $\operatorname{grad} F$ folgt $(y^* - \overline{x})^\top \operatorname{grad} F(x^*) \leq 0$, also mit Satz 4.2:

$$\beta := (y^* - x^*)^\top \operatorname{grad} F(x^*) \leq (\overline{x} - x^*)^\top \operatorname{grad} F(x^*) \leq F(\overline{x}) - F(x^*) = \alpha - \gamma < 0.$$

Also existiert ein ϵ mit $0 < \epsilon < 1/2\|x^* - y^*\|$ und $(y^* - x)^\top \operatorname{grad} F(x) \leq \beta/2$, für alle $x \in U_1 = \{x \mid x \in M, \|x - x^*\| \leq \epsilon\}$. Sei $U_2 = \{x \mid x \in M, \|x - x^*\| \leq \epsilon/2\}$. Wegen $x^{k_j} \to x^*$ sind alle bis auf endlich viele der x^{k_j} in U_2 enthalten. Für $x^{k_j} \in U_2$ und $y^{k_j} = y^*$ sei z^{k_j} der Punkt auf der Verbindungsstrecke von x^{k_j} und y^*, der auf dem Rand von U_1 liegt. Dann gilt $z^{k_j} - x^{k_j} = \lambda(y^* - z^{k_j})$ mit $\lambda = \frac{\|z^{k_j} - x^{k_j}\|}{\|y^* - z^{k_j}\|} \geq \frac{\epsilon/2}{\|y^* - x^*\| + \epsilon} =: \mu$. Nach Definition von x^{k_j+1} und Satz 4.2 gilt (beachte $\beta < 0$)

$$\begin{aligned} F(x^{k_j+1}) &\leq F(z^{k_j}) \leq F(x^{k_j}) + (z^{k_j} - x^{k_j})^\top \operatorname{grad} F(z^{k_j}) \\ &= F(x^{k_j}) + \lambda(y^* - z^{k_j})^\top \operatorname{grad} F(z^{k_j}) \\ &\leq F(x^{k_j}) + \mu\beta/2. \end{aligned}$$

Darum wird im Laufe des Verfahrens der Zielfunktionswert unendlich oft um $|\mu\beta/2|$ verringert. Widerspruch zu F beschränkt.
Also gilt $\lim F(x^k) = \gamma = \alpha$. Konvergiert eine Teilfolge der x^k gegen ein $\overline{x}$, so folgt mit der Stetigkeit von F, daß $F(\overline{x}) = \gamma = \alpha$. $\square$

Algorithmus 4.2
Procedure Zulässige Richtungen ;
begin
opt = 'Nein';
Man bestimme ein zulässiges x^0;
$k = 0$;
repeat
Bestimme Lösung y^k von $\min\{x^\top \operatorname{grad} F(x^k) \mid Ax = b, x \geq 0\}$ mit dem Simplexalgorithmus;
$s^k = y^k - x^k$;
if $s^{k\top} \operatorname{grad} F(x^k) = 0$ **then** opt = 'Ja', STOP **else**
begin
Bestimme λ_k mit $F(x^k + \lambda_k s^k) = \min\{F(x^k + \lambda s^k) \mid \lambda \geq 0, x^k + \lambda s^k \in M\}$;
Setze $x^{k+1} = x^k + \lambda_k s^k$;
$k = k + 1$;
end;
until opt = 'Ja' **or** x^k ist hinreichend gut;
end.

Beispiel 4.4 Minimiere $F(x) = x_1^2 + x_2^2 - 4x_1 - 2x_2$ unter $x_1 + 2x_2 \leq 8,\ x_1 + x_2 \leq 5,\ x_1, x_2 \geq 0$. Die Ecken sind $w^1 = (0,0)^\top, w^2 = (5,0)^\top, w^3 = (2,3)^\top, w^4 = (0,4)^\top$. Weiter ist $\operatorname{grad} F(x) = (2x_1 - 4, 2x_2 - 2)^\top$. Da F streng konvex ist, existiert eine eindeutige Lösung und die ist $\overline{x} = (2,1) \in M$ und $\operatorname{grad} F(\overline{x}) = (0,0)^\top$. Also $F_{\min} = F(\overline{x}) = -5$.

(1) Start in $x^0 = w^1 = (0,0)^\top$. Dann ist $F(x^0) = 0$ und $\operatorname{grad} F(x^0) = (-4,-2)^\top$. Durch Vergleich der Werte in den Ecken erhält man mit Lemma (4.7) $y^0 = w^2 = (5,0)^\top$. Mit $x(\lambda) = \lambda x^0 + (1-\lambda)y^0 = (5(1-\lambda), 0)^\top$ und $\varphi(\lambda) = F(x(\lambda)) = 25(1-\lambda)^2 - 20(1-\lambda)$ wird das Minimum von $\varphi(\lambda)$ gesucht, $0 \leq \lambda \leq 1$. $\varphi'(\lambda) = 0$ liefert $\lambda_0 = 3/5$.

(2) Also $x^1 = x(\lambda_0) = (2,0)^\top$ und $F(x^1) = -4$ und $\operatorname{grad} F(x^1) = (0,-2)^\top$. Durch Vergleich der Werte in den Ecken erhält man mit Lemma (4.7) $y^1 = w^4 = (0,4)^\top$. Mit $x(\lambda) = \lambda x^1 + (1-\lambda)y^1 = (2\lambda, 4(1-\lambda))^\top$ und $\varphi(\lambda) = F(x(\lambda)) = 20\lambda^2 - 32\lambda + 8$ wird das Minimum von $\varphi(\lambda)$ gesucht, $0 \leq \lambda \leq 1$. $\varphi'(\lambda) = 0$ liefert $\lambda_1 = 4/5$.

(3) Also $x^2 = x(\lambda_1) = (8/5, 4/5)^\top$ und $F(x^2) = -4.8$ und $\operatorname{grad} F(x^2) = (-4/5, -2/5)^\top$. Durch Vergleich der Werte in den Ecken erhält man mit Lemma (4.7) $y^2 = w^2 = (5,0)^\top$. Mit $x(\lambda) = \lambda x^2 + (1-\lambda)y^2 = (5 - 17/5\lambda, 4/5\lambda)^\top$ und $\varphi(\lambda) = F(x(\lambda)) = 5 - 22\lambda + 61/5\lambda^2$ wird das Minimum von $\varphi(\lambda)$ gesucht, $0 \leq \lambda \leq 1$. $\varphi'(\lambda) = 0$ liefert $\lambda_2 = 55/61$.

(4) Also $x^3 = x(\lambda_2) = (118/61, 44/61)^\top$ und $F(x^3) = -4.92\ldots$ usw.

4.9 Ein Schnittebenenverfahren

Sind $F, f_1, \dots, f_m$ konvexe Funktionen, so betrachten wir das Problem

$$\min\{F(x) \mid f_i(x) \le 0,\ i = 1, \dots, m\}. \tag{4.15}$$

Zusätzlich setzen wir voraus

(1) Die Menge der zulässigen Punkte $M = \{x \mid f_i(x) \le 0,\ i = 1, \dots, m\}$ seien in einem Polytop $S^0 = \{x \mid Ax \le b\}$ enthalten (S^0 und damit auch M sind also beschränkt).

(2) Auf S^0 seien die F und die f_i konvex und differenzierbar und $\|\operatorname{grad} f_i(x)\| \le K,\ i = 1, \dots, m$.

Man darf annehmen, daß die Zielfunktion linear ist, denn ansonsten füge man die zusätzliche Restriktion $f_{m+1}(x) = F(x) - x_{n+1} \le 0$ ein und minimiere x_{n+1}.
Weiter sei $f(x) = \max\{f_1(x), \dots, f_m(x)\}$ für $x \in S^0$. f ist konvex und stetig aber nicht überall differenzierbar. In Punkten, wo das Maximum mehrfach angenommen wird, kann man einen der betreffenden Gradienten auswählen, so daß $\operatorname{grad} f$ (in einem gewissen Sinne) definiert und beschränkt ist.
Wegen $f(x) \le 0$ genau dann, wenn alle $f_i(x) \le 0$, ist das Problem (4.15) also äquivalent zu

$$\min\{c^\top x \mid f(x) \le 0\}. \tag{4.16}$$

Das Verfahren (von Kelley, 1960) besteht nun aus einer (im allgemeinen unendlichen) Folge von linearen Programmen. Für $k = 0, 1, 2, \dots$ sei x^k Minimallösung von $\min\{c^\top x \mid x \in S^k\}$ und

$$S^{k+1} = S^k \cap \{x \in \mathbb{R}^n \mid f(x^k) + (x - x^k)^\top \operatorname{grad} f(x^k) \le 0\}. \tag{4.17}$$

Bemerkungen: (1) S^{k+1} entsteht also aus S^k durch Abschneiden mit der Hyperebenen $f(x^k) + (x - x^k)^\top \operatorname{grad} f(x^k) = 0$. Daher nennt man die Methode Schnittebenenverfahren. Diese Schnitthyperebene ist der Schnitt der Tangentialebene (bzw. Stützebene) von f in x^k (im $\mathbb{R}^{n+1}$) mit dem n–dimensionalen Raum $y = x_{n+1} = 0$.
(2) Da S^0 Polytop ist, gilt dies auch für alle S^k.

Lemma 4.8 *(a) $M \subset S^k$, für alle k.*

(b) Entweder ist x^k Minimallösung von (4.16) oder $x^k \notin M$ und $x^k \in S^k \backslash S^{k+1}$.

Beweis: (a) Mit vollständiger Induktion. Nach Voraussetzung ist $M \subset S^0$. Sei $M \subset S^{k-1}$. Für alle zulässigen $x \in M$ gilt mit Satz 4.2

$$0 \ge f(x) \ge f(x^k) + (x - x^k)^\top \operatorname{grad} f(x^k),$$

d.h. x wird nicht abgeschnitten. Da $x \in M$ beliebig, folgt $M \subset S^k$.
(b) Sei x^k Minimalpunkt bzgl. S^k. Gilt $x^k \in M$, so ist x^k wegen $M \subset S^k$ erst recht Minimalpunkt bzgl. M. Ist aber $x^k \notin M$, so ist $f(x^k) > 0$ und die Ungleichung $f(x^k) + (x - x^k)^\top \operatorname{grad} f(x^k) \le 0$ ist für $x = x^k$ nicht erfüllt. Also $x^k \notin S^{k+1}$ aber $x^k \in S^k$. □

Im Laufe des Verfahrens sind drei Fälle möglich.

(1) Ein $S^k = \emptyset$. Nach Lemma 4.8 ist dann auch $M = \emptyset$ und (4.16) hat keine Lösung.

(2) Ein $x^k \in M$. Nach Lemma 4.8 ist dann x^k Minimallösung.

(3) Man erhält eine unendliche Folge $\{x^k\}$ und zugehörige $\{S^k\}$ mit $S^{k+1} \subset S^k$.

In Fall (3) gilt

Satz 4.21 *Jeder Häufungspunkt $\overline{x}$ von $\{x^k\}$ ist Minimalpunkt von (4.16).*

Beweis: Es sei x^{k_j} eine Teilfolge, die gegen $\overline{x}$ konvergiert. Dann gilt für alle $x \in M$, daß $f(x) \geq f(x^{k_j})$. Wegen der Stetigkeit von f gilt also $f(\overline{x}) \leq f(x)$, für alle $x \in M$. Damit reicht es zu zeigen, daß $\overline{x} \in M$.
Angenommen, $\overline{x} \notin M$. Dann gilt $f(\overline{x}) = \eta > 0$. Da $\overline{x}$ Häufungspunkt ist und f stetig, existiert ein k mit $\|x^k - \overline{x}\| < \eta/(2K)$ und $f(x^k) > \eta/2$ (dabei ist K mit $\|\operatorname{grad} f(x)\| \leq K$, für alle $x \in S^0$). Damit wird

$$f(x^k) + (\overline{x} - x^k)^\top \operatorname{grad} f(x^k) > \eta/2 - (\eta/(2K))K = 0.$$

Also ist $\overline{x} \notin S^{k+1}$. Da S^{k+1} abgeschlossen ist, gilt sogar für eine ganze Umgebung $B = B(\overline{x}, \epsilon)$, daß $B \cap S^{k+1} = \emptyset$. Wegen $x^l \in S^{k+1}$, für $l \geq k+1$, ist dies ein Widerspruch. □

Bemerkung: Im 1–dimensionalen Fall stimmt die Methode mit dem klassischen Newton–Verfahren überein.

Algorithmus 4.3
Procedure Kelley ;
begin
  leer = 'Nein'; opt = 'Nein';
  Man bestimme ein Polytop S^0 mit $M \subset S^0$;
  $k = 0$;
  repeat
    Bestimme Lösung x^k von $\min\{c^\top x \mid x \in S^k\}$ mit dem Simplexalgorithmus;
    if $S^k = \emptyset$ **then** leer = 'Ja', STOP **else**
    if $x^k \in M$ **then** opt = 'Ja', STOP **else**
    begin
      $S^{k+1} = S^k \cap \{x \mid f(x^k) + (x - x^k)^\top \operatorname{grad} f(x^k) \leq 0\}$;
      $k = k + 1$;
    end;
  until (leer = 'Ja') **or** (opt = 'Ja') **or** (x^k ist 'hinreichend zulässig');
end.

Beispiel 4.5 Minimiere $F(x) = x_1 - x_2$ unter $f(x) = 3x_1^2 - 2x_1x_2 + x_2^2 - 1 \leq 0$. M ist Ellipse und $x^\star = (0,1)^\top$ ist Minimalpunkt mit $F_{\min} = -1$. Ferner $\operatorname{grad} f(x) = (6x_1 - 2x_2, -2x_1 + 2x_2)^\top$. Man wähle $S^0 = \{x \mid |x_1|, |x_2| \leq 2\}$.

(1) Das Minimmum von F bzgl. S^0 liegt in $x^0 = (-2,2)^\top$ mit $F(x^0) = -4$. Es ist $f(x^0) = 23 > 0$, also $x^0 \notin M$. Wegen $\operatorname{grad} f(x^0) = (-16, 8)^\top$ ist die Schnittebene $-16x_1 + 8x_2 - 25 \leq 0$. Also $S^1 = \{x \mid |x_1|, |x_2| \leq 2,\ -16x_1 + 8x_2 \leq 25\}$.

(2) Das Minimmum von F bzgl. S^1 liegt in $x^1 = (-9/16, 2)^\top$ mit $F(x^1) = -41/16$. Wegen $f(x^1) > 0$ ist $x^1 \notin M$. $\operatorname{grad} f(x^1) = 1/8(-59, 41)^\top$. Also ist die Schnittebene $-59/8x_1 + 41/8x_2 - 2099/256 \leq 0$.

Also $S^2 = \{x \mid |x_1|, |x_2| \leq 2,\ -16x_1 + 8x_2 \leq 25,\ -59/8x_1 + 41/8x_2 \leq 8.2\}$.
Und so weiter. Man erhält eine Folge von Schnittebenen, die sich der Tangente an M in $x^\star$ annähern.

Die Nachteile des Verfahrens sind

- Im allgemeinen unendlich viele Schritte.
- Langsame Konvergenz.
- Keines der x^k gehört zu M.
- Wachsende Anzahl von Restriktionen.
- Im Grenzfall schleifende Schnitte.

Daher ist das Verfahren selbst inzwischen überholt. Es dient aber als Ausgangspunkt für diverse Varianten.

4.10 Ganzzahlige konvexe Optimierung

Natürlich kann auch bei konvexer Optimierung die Ganzzahligkeit der Lösungen notwendig sein. Seien also konvexe Funktionen $F, f_1, \ldots, f_m$ gegeben. Dann betrachten wir das Problem

$$\min\{F(x) \mid f_i(x) \leq 0,\ i = 1, \ldots, m,\ x \geq 0,\ x \in \mathbb{Z}^n\}. \tag{4.18}$$

Man kann wieder o.E. annehmen, daß $F(x) = c^\top x$ linear ist. Zur Lösbarkeit des Problems ist notwendig:

(1) Die Menge der zulässigen Punkte der Relaxation $M = \{x \in \mathbb{R}^n \mid f_i(x) \leq 0,\ i = 1, \ldots, m,\ x \geq 0\}$ liegt in einem Polytop $S^0 = \{x \in \mathbb{R}^n \mid Ax \leq b,\ x \geq 0\}$.

(2) Die f_i seien auf S^0 konvex und differenzierbar und und ihre partiellen Ableitungen seien auf S^0 beschränkt.

Man löst das Problem wie beim Schnittebenenverfahren, nur daß nun statt des Simplexalgorithmus das Gomory–Verfahren angewandt wird.

Algorithmus 4.4
Procedure Ganzzahlige Konvexe Optimierung ;
begin
leer = 'Nein'; opt = 'Nein';
Man bestimme ein Polytop S^0 mit $M \subset S^0$;
$k = 0$;

repeat
    Bestimme Lösung x^k von $\min\{c^\top x \mid x \in S^k, x \in \mathbb{Z}^n\}$ mit dem GDA;
    if $S^k \cap \mathbb{Z}^n = \emptyset$ **then** leer = 'Ja', STOP **else**
    if $x^k \in M \cap \mathbb{Z}^n$ **then** opt = 'Ja', STOP **else**
    begin
        Bestimme p mit $f_p(x^k) = \max\{f_i(x^k) \mid i = 1, \ldots, m\}$;
        $S^{k+1} = S^k \cap \{x \mid f_p(x^k) + (x - x^k)^\top \operatorname{grad} f_p(x^k) \leq 0\}$;
        $k = k + 1$;
    end;
until (leer = 'Ja') **or** (opt = 'Ja');
end.

Satz 4.22 *Der Algorithmus 4.4 liefert nach endlich vielen Schritten eine Lösung von (4.18) oder meldet, daß das Problem keine zulässigen Punkte hat.*

Beweis: Es gibt nur endlich viele ganzzahlige Lösungen. In jeder Iteration ist das ganzzahlige Optimum entweder zulässig oder es wird durch den Kelley–Schnitt eliminiert. Also muß nach endlich vielen Schritten das Optimum erreicht sein oder das Verfahren bricht mit der Meldung ab, daß es keine ganzzahlige zulässige Lösung gibt. □

Beispiel 4.6 Minimiere $F(x) = x_1 - x_2$ unter $f(x) = 4x_1^2 - 3x_1x_2 + x_2^2 - 8 \leq 0,\ x \geq 0, x \in \mathbb{Z}^2$. M ist eine Ellipse und liegt in $S^0 = \{x \mid 0 \leq x_1 \leq 3,\ 0 \leq x_2 \leq 5\}$. Es gilt $\operatorname{grad} f(x) = (8x_1 - 3x_2, -3x_1 + 2x_2)^\top$.

(1) Das Minimum von F in S^0 ist in $x^0 = (0, 5)^\top$ mit $F_{\min} = -5$. Wegen $f(x^0) = 17 > 0$ ist x^0 nicht zulässig. Wegen $\operatorname{grad} f(x^0) = (-15, 10)^\top$ ist der Kelley–Schnitt dann $-15x_1 + 10x_2 - 33 \leq 0$.

(2) Ein Minimum von F in $S^1 \cap \mathbb{Z}^2$ liegt in $x^1 = (2, 5)^\top$ (Gomory) mit $F(x^1) = -3$. Wegen $f(x^1) = 3 > 0$ ist x^1 nicht zulässig. Wegen $\operatorname{grad} f(x^1) = (1, 4)^\top$ ist der Kelley–Schnitt $x_1 + 4x_2 - 19 \leq 0$.

(3) Ein Minimum von F in $S^2 \cap \mathbb{Z}^2$ liegt in $x^2 = (0, 3)^\top$ (Gomory) mit $F(x^2) = -3$. Wegen $f(x^2) = 1 > 0$ ist x^2 nicht zulässig. Wegen $\operatorname{grad} f(x^1) = (-9, 6)^\top$ ist der Kelley–Schnitt $-9x_1 + 6x_2 - 17 \leq 0$.

(4) Das Minimum von F in $S^3 \cap \mathbb{Z}^2$ liegt in $x^3 = (1, 4)^\top$ (Gomory) mit $F(x^3) = -3$. Wegen $f(x^3) = 0$ ist x^3 zulässig und damit ganzzahlige optimale Lösung.

5 Komplexität und das Ellipsoid–Verfahren

5.1 Zeitschranken

Für die praktische Anwendung von Optimierungsverfahren ist der Aufwand, den sie zur Lösung eines Problems benötigen, entscheidend. Ein erstes Kriterium ist die Endlichkeit oder Nicht–Endlichkeit eines Verfahrens. Bei letzterem ist die Frage, nach wievielen Schritten eine brauchbare Näherung erreicht ist. Das führt zu Approximationsproblemen, auf die hier nicht eingegangen wird. Bei endlichen Verfahren spielt der reale Aufwand die entscheidende Rolle. Dieser Aufwand wird in der Regel als die Zeit betrachtet, die ein Algorithmus zur Lösung eines Problems benötigt (der Speicheraufwand spielt heute nur noch eine untergeordnete Rolle). Das Zeitmaß hängt natürlich vom jeweiligen Computer ab, ist aber bei ein und demselben Computer proportional zu der Anzahl der ausgeführten elementaren Operationen bzw. Schritte. Technischer Fortschritt verbessert die Konstanten sehr schnell, so daß im wesentlichen nur die Größenordnung interessiert. Dazu werden die klassischen Landau–Symbole benutzt.

Definition 5.1 *Für $f, g : \mathbb{N} \to \mathbb{R}$ schreibt man*

(1) $f(n) = O(g(n))$, falls $c > 0$ und $n_0 \in \mathbb{N}$ existieren mit $f(n) \le cg(n)$, für $n > n_0$.

(2) $f(n) = \Omega(g(n))$, falls $c > 0$ und $n_0 \in \mathbb{N}$ existieren mit $f(n) \ge cg(n)$, für $n > n_0$.

(3) $f(n) = \Theta(g(n))$, falls (1) und (2) gelten.

Eine andere Schreibweise für $f(n) = \Theta(g(n))$ ist $f(n) \asymp g(n)$. $\asymp$ stellt offensichtlich eine Äquivalenzrelation dar. Die Äquivalenzklasse von $f(n)$ heißt Wachstumsrate *von $f(n)$.*

Wir gehen davon aus, daß der Zeitaufwand für jede elementare Instruktion (arithmetische Instruktionen, Vergleiche) proportional zur Länge der Darstellung der Zahlen in einem Zahlensystem ist. Zur Behandlung eines Problems mit dem Computer wird der Input in einem endlichen Alphabet beschrieben (z. B. $\{0, 1, \ldots 9\}$ oder $\{0, 1\}$). Die Größe eines Beispiels ist die Anzahl der Zeichen in dieser Beschreibung.

Beispiel 5.1 *Der Input sei eine natürliche Zahl n. In einem Zahlensystem mit Basis B ist die Zahl der Zeichen für die Darstellung $\lceil \log_B(n+1) \rceil$. Damit ist Größe der Darstellung $\Theta(\log n)$ wegen $\log_B n = (\log n)/\log B$. Für negative Zahlen wird ein weiteres Zeichen für das Vorzeichen benötigt. Die Größe einer rationalen Zahl $r = p/q$ ist die Summe der Größen von p und q.*

Beispiel 5.2 *Größe eines LP : Die Elemente von A, b, c seien ganze Zahlen. Zur Trennung von Zahlen dienen gewisse Symbole. Die Größe eines LP ist $\Theta(mn + \log|Z|)$, wobei Z das Produkt aller von Null verschiedenen Koeffizienten ist.*

Die Zahl der ausgeführten Schritte hängt auch vom Typ der eingegebenen Daten ab. Ist z. B. beim Simplexverfahren $\min\{c^\top x \mid Ax \le b, x \ge 0\}$ zu lösen und ist $c \ge 0$, so ist der Startpunkt $x = 0$ schon die Lösung. Daher fragt man bei vorgegebener Zahl eingegebener Daten nach dem ungünstigsten Fall (worst case) für den Algorithmus. Hat der Input die Länge

n, so ist diese größtmögliche Anzahl eine Funktion von n, die die *Komplexität* des Verfahrens genannt wird (z. B. $2n^3$, $3n^2 + 4\log n$, $3^n + 6n^4$ etc.).
Im folgenden werden meist die Kosten einer Operation unabhängig von der Größe der Zahl angenommen. Damit wird die Zeitkomplexität eines Algorithmus einfach die Anzahl der benötigten elementaren Operationen. Dabei wird implizit benutzt, daß die Zahlen, die während des Algorithmus auftauchen, nicht wesentlich größer als die des Input sind.
Es kann jedoch der wesentlich kompliziertere Fall auftreten, daß die Zahl der arithmetischen Operationen von der Größe des Input abhängt.

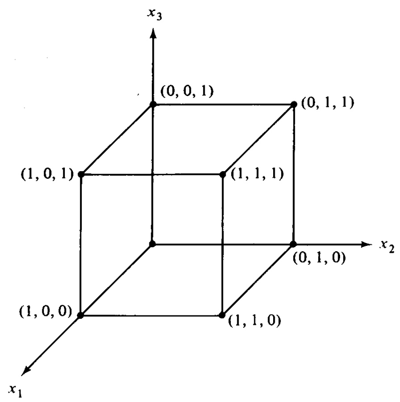
Beispiel 5.3 *Der Simplexalgorithmus besteht aus einer Initialisierungsphase für die Aufstellung des Starttableaus und einer Anzahl von Iterationen. Für eine $m \times n$–Matrix A erfordert die Initialsierung $O(mn)$ arithmetischer Operationen. Ein Iterationsschritt erfordert ebenfalls $O(mn)$ Operationen. Bei Zykelvermeidung könnte der Simplexalgorithmus im schlimmsten Fall alle $\binom{n}{m}$ Basen durchlaufen. Der Simplexalgorithmus erfordert also höchstens $O(mn\binom{n}{m})$ Operationen. In der Tat kann er sehr viele Operationen erfordern.*

Definition 5.2 *Ein Algorithmus heißt* polynomial, *wenn es ein Polynom in der Größe des Input gibt, daß den Aufwand des Algorithmus abschätzt. Die übrigen Algorithmen heißen* exponentiell.

Beispiel 5.4 *Der Gaußsche Algorithmus zur Auflösung eines Gleichungssystems $Ax = b$, mit $A \in \mathbb{Z}^{n\times n}$, $b \in \mathbb{Z}^n$, ist polynomial.*

5.2 Der Simplexalgorithmus ist nicht polynomial

Um zu zeigen, daß der Simplexalgorithmus nicht polynomial ist, genügt es, eine Serie von Beispielen zu geben, bei der der Input quadratisch wächst, die Zahl der Pivotschritte sich aber bei vorgeschriebener Spaltenwahl verdoppelt.
Beim Simplexalgorithmus wird in jedem Schritt eine neue Ecke des zugehörigen Polytops bestimmt, die mit der vorigen durch eine Kante verbunden ist.
Gesucht ist eine Serie von Polytopen, bei der die Eckenzahl exponentiell mit der Seitenzahl wächst. Eine solche Serie besteht aus den d–dimensionalen Würfeln im $\mathbb{R}^d$ $W_d = \{(x_1, \dots, x_d) \,|\, 0 \leq x_j \leq 1\}$. Der d–dimensionale Würfel hat 2^d Ecken: j Koordinaten Null und die übrigen Eins.

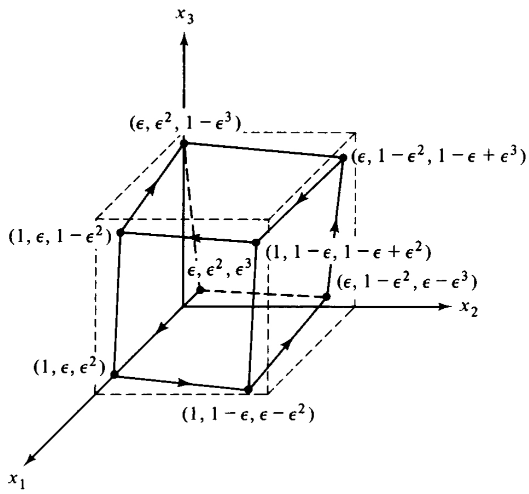


Für die Zielfunktion $\max x_d$ haben 2^{d-1} Ecken den Wert Null und die übrigen den Wert Eins. Folgende Änderung des Würfels ergibt das gesuchte Beispiel.
Das Polytop P sei für ein ϵ mit $0 < \epsilon < 1/2$ gegeben durch

$$1 \geq x_1 \geq \epsilon, \quad 1 - \epsilon x_{j-1} \geq x_j \geq \epsilon x_{j-1}, \quad j = 2, \ldots, d.$$

Die Überführung in Standardform geschieht durch d Schlupfvariablen s_j und d Überschußvariablen r_j. Also $m = 2d$ und $n = 3d$ und das vollständige LP ist:

$$\begin{aligned} \min\{-x_d \quad | \quad & x_1 - r_1 = \epsilon, \; x_1 + s_1 = 1 \\ & x_j - \epsilon x_{j-1} - r_j = 0, \; j = 2, \ldots, d, \\ & x_j + \epsilon x_{j-1} + s_j = 1, \; j = 2, \ldots, d, \\ & x_j, s_j, r_j \geq 0\} \end{aligned} \tag{5.1}$$



Lemma 5.1 *Die Menge der zulässigen Basen von (5.1) ist die Menge der Teilmengen von* $\{x_1, \ldots, x_d, r_1, \ldots, r_d, s_1, \ldots, s_d\}$*, die alle* x_j *und genau ein Element von* $\{r_j, s_j\}$*, für jedes* $j = 1, \ldots, d$ *enthalten. Alle Ecken sind nichtentartet.*

Beweis: Wegen $x_1 \geq \epsilon$ und $x_{j+1} \geq \epsilon x_j$ für $j = 1, \ldots, d-1$ ist in jeder zulässigen Lösung $x_j \geq \epsilon^j > 0$. Alle zulässigen Basen von (5.1) enthalten daher $x_1, \ldots, x_d$.
Angenommen, für ein j gelte $r_j = s_j = 0$. Für $j = 1$ ergibt sich der Widerspruch $\epsilon = x_1 = 1$. Für $j \geq 2$ ergibt die Subtraktion der beiden zugehörigen Restriktionen $2\epsilon x_{j-1} = 1$, also einen Widerspruch.
Da eine Basis außer $x_1, \ldots, x_d$ noch d weitere Elemente enthält, ist in jeder Ecke ist $r_j > 0$ oder $s_j > 0$, für $j = 1, \ldots, d$. Da $x_j > 0$ für alle $j = 1, \ldots, d$ gilt, sind alle Ecken nichtentartet.

$\square$

Jede Ecke von (5.1) hat die Darstellung x^S, mit $S \subset \{1, \ldots, d\}$ und $j \in S \iff r_j > 0$. Der Wert von x_j in x^S wird mit x_j^S bezeichnet.

Lemma 5.2 *(a) Sei* $d \in S$ *aber* $d \notin S'$*, dann gilt* $x_d^S > x_d^{S'}$.
(b) Für $S' = S \setminus \{d\}$ *gilt* $x_d^{S'} = 1 - x_d^S$.

Beweis: Wegen $d \in S$ gilt $s_d = 0$ und die vierte Restriktion in (5.1) ergibt $x_d^S = 1 - \epsilon x_{d-1}^S$. Es ist aber $x_{d-1}^S \leq 1$ und $\epsilon < 1/2$ und damit $x_d^S > 1/2$. Andererseits gilt wegen $d \notin S'$, daß $r_d = 0$ und die dritte Restriktion in (5.1) ergibt $x_d^{S'} = \epsilon x_{d-1}^{S'} < 1/2$. Da sich beim Übergang von S zu S' nur x_d ändert ($x_1, \ldots, x_{d-1}$ sind durch die ersten $d-1$ Gleichungen schon bestimmt), gilt insbesondere $x_{d-1}^{S'} = x_{d-1}^S =: x_{d-1}$. Mit $x_d^{S'} - \epsilon x_{d-1} = 0$ und $x_d^S + \epsilon x_{d-1} = 1$ folgt $x_d^{S'} = 1 - x_d^S$.

□

Lemma 5.3 *Seien die Teilmengen von $\{1, \ldots, d\}$ so aufgezählt, daß $x_d^{S_i} \leq \ldots \leq x_d^{S_j}$ für $i < j$ gilt. Dann gilt überall Ungleichheit und die Ecke x^{S_j} und $x^{S_{j+1}}$ sind benachbart für $j = 1, \ldots, 2^d - 1$.*

Beweis: Induktion nach d. Für $d = 1$ gibt es die Ecken $(x_1, r_1, s_1) = (\epsilon, 0, 1-\epsilon)$ und $(1, 1-\epsilon, 0)$. Der Satz sei jetzt für den d–Würfel bewiesen und die entsprechende Aufzählung sei $S_1, \ldots, S_{2^d}$. Weiter sind $S_1, \ldots, S_{2^d}$ auch Teilmengen von $\{1, \ldots, d+1\}$ und es gilt $x_{d+1}^{S_j} = \epsilon x_d^{S_j} < 1/2$, da $r_{d+1}^{S_j} = 0$. Damit gilt nach der Induktionsvoraussetzung $x_{d+1}^{S_1} < \ldots < x_{d+1}^{S_{2^d}}$. Die übrigen 2^d Teilmengen von $\{1, \ldots, d+1\}$ ergeben sich durch $S'_j = S_j \cup \{d+1\}$, $j = 1, \ldots, 2^d$. Nach Lemma 5.2 gilt $x_{d+1}^{S'_j} = 1 - x_{d+1}^{S_j}$ und somit

$$x_{d+1}^{S_1} < \ldots < x_{d+1}^{S_{2^d}} < 1/2 < x_{d+1}^{S'_{2^d}} < \ldots < x_{d+1}^{S'_1}.$$

Nach Induktionsannahme sind x^{S_j} und $x^{S_{j+1}}$ benachbart und damit auch $x^{S'_j}$ und $x^{S'_{j+1}}$. Schließlich sind auch $x^{S_{2^d}}$ und $x^{S'_{2^d}}$ benachbart, da die letzte Basis aus der vorigen durch Hinzunahme von r_{d+1} und Weglassen von s_{d+1} entsteht.

□

Satz 5.1 (Klee–Minty) *Für jedes $d > 1$ gibt es ein LP mit $2d$ Gleichungen, $3d$ Variablen und ganzzahligen Koeffizienten, deren Absolutbetrag durch 4 beschränkt ist, so daß der Simplexalgorithmus $2^d - 1$ Iterationen brauchen kann, um das Optimum zu finden.*

Beweis: Man wähle $\epsilon = 1/4$ und multipliziere alle Gleichungen in (5.1) mit 4, so daß alle Koeffizienten ganze Zahlen sind. Damit hat die Kette benachbarter Ecken aus Lemma 5.3 abnehmende Kosten.

□

Bemerkung: Die Größe des Input in Satz 5.1 ist $\Theta(6d^2 + \log(4^{5(d-1)+6}))$, also $\Theta(d^2)$, und damit ist der Simplexalgorithmus i. a. nicht polynomial. Jedoch sind Beispiele wie das von Klee–Minty sehr 'selten'. Borgwardt zeigte 1982, daß der Simplexalgorithmus 'fast immer' polynomial ist. Daher sind in der Praxis in der Regel Varianten des Simplexverfahrens effizienter als Verfahren, die theoretisch besser sind, wie etwa das polynomiale Ellipsoidverfahren, das im folgenden behandelt wird.

5.3 Komplexität von LP, LI und LSI

Es wird gezeigt, daß zur Lösung der folgenden drei scheinbar ungleich schweren Probleme im wesentlichen der gleiche Rechenaufwand benötigt wird.

1. Lineares Programm (LP) in der Standardform:

   Für $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ finde man eine Lösung von

$$\min\{c^\top x \mid Ax = b, x \geq 0\} \tag{5.2}$$

   oder zeige, daß es keine Lösung gibt oder das Problem unbeschränkt ist (o. E. sei $n \geq m$).

2. Lineare Ungleichungen (LI):

   Für $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$ finde man eine Lösung von $Ax \leq b$ oder zeige, daß es keine Lösung gibt.

3. Lineare strenge Ungleichungen (LSI):

   Für $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$ finde man eine Lösung von $Ax < b$ oder zeige, daß es keine Lösung gibt.

Satz 5.2 *Es gibt einen polynomialen Algorithmus für LP genau dann, wenn es einen polynomialen Algorithmus für LI gibt.*

Beweis: (i) Hat man einen polynomialen Algorithmus für LP, so kann man ein Beispiel von LI zunächst auf Standardform bringen (durch Einführung von Schlupfvariablen und Beseitigung der nichtrestringierten Variablen durch Substitution). Setze c beliebig und bestimme mit Hilfe des Verfahrens für LP eine Lösung oder stelle fest, daß keine existiert.
(ii) Es stehe nun ein polynomialer Algorithmus $\mathcal{A}$ für LI zur Verfügung. Zu lösen ist (5.2).
(1) Unser Algorithmus ruft $\mathcal{A}$ zunächst mit $Ax \leq b$, $-Ax \leq -b$ und $-Ix \leq 0$ auf, um festzustellen, ob es zulässig ist.
(2) Falls es zulässig ist, so ist x genau dann eine optimale Lösung, wenn es ein $y \in \mathbb{R}^m$ gibt mit

$$Ax = b,\ x \geq 0,\ y^\top A \leq c^\top,\ c^\top x = y^\top b$$

(Dualitätssatz). Falls es kein solches Paar (x, y) gibt, dann ist das Problem unbeschränkt. Man rufe also $\mathcal{A}$ auf mit

$$Ax \leq b,\ -Ax \leq -b,\ -Ix \leq 0,\ A^\top y \leq c^\top,\ c^\top x - b^\top y \leq 0,\ -c^\top x + b^\top y \leq 0.$$

Ist (x, y) eine Lösung, dann ist x Lösung des LP. Gibt es keine Lösung, so ist das Problem unbeschränkt. Ist schließlich L die Größe des LP, so haben die obigen LI eine Größe von höchstens $O(L^2)$. Darum ist das Verfahren polynomial. $\square$

Bemerkung: Der Satz besagt, daß das scheinbar viel schwierigere LP im wesentlichen die gleiche Komplexität hat wie LI, d. h. allein das Suchen einer Startecke (Phase I) hat im wesentlichen die gleiche Komplexität wie das ganze Problem.

Lemma 5.4 *Das LI $Ax \leq b$ hat genau dann eine Lösung, wenn das LSI $Ax < b + \epsilon(1, \ldots, 1)^\top$, mit $\epsilon = 2^{-2L}$ eine Lösung hat.*

Beweis: Hat das LI eine Lösung, so ist diese auch eine des LSI. Umgekehrt soll nun aus einer Lösung x^0 des LSI eine des LI konstruiert werden. Es sei I die Menge von Zeilenvektoren a_i von A mit $b_i \leq a_i{}^\top x^0 < b_i + \epsilon$. Es kann angenommen werden, daß sich die übrigen Zeilenvektoren linear durch die in I kombinieren lassen. Ansonsten, ist ein a_j linear unabhängig zu denen in I, so hat das LGS $a_i{}^\top z = 0$, für $i \in I$ und $a_j{}^\top z = 1$ eine Lösung z^0. Für hinreichend kleines $\lambda > 0$ ist dann $x^1 = x^0 + \lambda z^0$ eine Lösung des LSI mit einem Element mehr in I. Dies bricht aber spätestens nach $k = \text{rank}(A) \leq m$ Schritten ab. Darum gilt $a_j = \sum_{a_i \in I'} \beta_{ji} a_i$, für eine linear unabhängige Teilmenge I' von I. Nach der Cramerschen Regel (o. E. sei $\text{lin}(I') = \text{lin}(I) = \mathbb{R}^k$, sonst Drehung) sind alle β_{ji} Quotienten von Determinanten mit Betrag $\leq 2^L$: $\beta_{ji} = D_{ji}/D$. (Es gilt $k = |I'| = \text{rank}(A) \leq \min(n, m)$. Darum sind die Determinanten, als Summe von $k!$ Summanden $\leq |Z|$ höchstens $k!|Z| \leq 2^{k^2}|Z| \leq 2^{mn}|Z| = L$.) Es sei $\overline{x}$ die Lösung der linearen Gleichungen $a_i{}^\top x = b_i$, für $a_i \in I'$. O. E. sei $D > 0$. Es gilt für alle j:

$$\begin{aligned} D(a_j{}^\top \overline{x} - b_j) &= \sum_{a_i \in I'} D_{ji} a_i{}^\top \overline{x} - D b_j = \sum_{a_i \in I'} D_{ji} b_i - D b_j \\ &= -\sum_{a_i \in I'} D_{ji}(a_i{}^\top x^0 - b_i) + D(a_j{}^\top x^0 - b_j) \\ &< \epsilon \left(\sum_{a_i \in I'} |D_{ji}| + D \right) < 2^{-2L}(m+1)2^L < 1 \end{aligned}$$

(mit der Definition von $\overline{x}$, Addieren und Subtrahieren von $D a_j{}^\top x^0$, $|a_i{}^\top x^0 - b_i| < \epsilon$, für $i \in I'$ und $a_j{}^\top x^0 - b_j| < \epsilon$, für alle j). Also gilt für alle j, daß $D(a_j{}^\top \overline{x} - b_j) < 1$. Ferner sind alle Nenner der Komponenten von $\overline{x}$ Teiler von D (Cramersche Regel, wie oben) und darum ist $D(a_j{}^\top \overline{x} - b_j)$ eine ganze Zahl. Es folgt $a_j{}^\top \overline{x} - b_j \leq 0$, für alle j und damit ist $\overline{x}$ die gewünschte Lösung des LI. □

Satz 5.3 *Gibt es einen polynomialen Algorithmus für LSI, dann auch für LI.*

Beweis: Nach Lemma 5.4 ist das LI (mit Größe L) äquivalent zu dem LSI

$$2^{2L} a_i{}^\top x < 2^{2L} b_i + 1, \quad i = 1, \ldots, m.$$

Sei Z' das Produkt aller Koeffizienten $\neq 0$ in dem LSI. Dann gilt $\log|Z'| \leq 6L \log|Z|$ und die Größe dieses Problems ist höchstens $O(L^2)$. Darum werden nur polynomial viele Schritte benötigt ($p(L^2)$ ist wieder ein Polynom). □

5.4 Ellipsoide

Definition 5.3 $E \subset \mathbb{R}^n$ *heißt* Ellipsoid, *wenn es ein* $d \in \mathbb{R}^n$ *und eine symmetrische positiv definite* $(n \times n)$*–Matrix* Q *gibt mit*

$$E = E(Q, d) = \{x \in \mathbb{R}^n \,|\, (x - d)^\top Q^{-1}(x - d) \leq 1\}.$$

Man beachte, daß die Bedingung 'Q symmetrisch' keine Einschränkung darstellt. Mit Methoden der Linearen Algebra und Analysis erhält man:

Satz 5.4 *Es sei Q eine (symmetrische) positiv definite $(n \times n)$–Matrix, $d \in \mathbb{R}^n$ und $E = E(Q,d)$ das zugehörige Ellipsoid. Dann gilt*

(a) Q^{-1} ist positiv definit.

(b) Es gibt eine eindeutig definierte symmetrische positiv definite Matrix R mit $Q = R^\top R$.

(c) $E(Q,d) = RB^n + d$, wobei B^n die (Einheits–) Kugel um 0 mit Radius 1 ist.

(d) Das Volumen eines Ellipsoids ist $V(E) = \sqrt{\det Q} \cdot \kappa_n$, wobei $\kappa_n = \frac{\pi^{n/2}}{\Gamma(n/2+1)} = V(B^n)$.

(e) Der Quotient der Volumina zweier Ellipsoide ist invariant unter (bijektiven) affinen Abbildungen.

(f) Sind $\lambda_1, \ldots, \lambda_n$ die Eigenwerte von Q, so sind $\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}$ die Halbachsen von $E(Q,d)$.

Beweis: (a) Es sei $x \in \mathbb{R}^n \backslash \{0\}$. Dann gibt es ein $y \neq 0$ mit $x = Qy$ und es folgt $x^\top Q^{-1} x = y^\top Q^\top Q^{-1} Q y = y^\top Q y > 0$.
(b) Aus der Linearen Algebra ist bekannt, daß die Eigenwerte $\lambda_1, \ldots, \lambda_n$ von Q reell und positiv sind. Weiter gibt es eine Orthonormalbasis von Eigenvektoren, d. h. Q ist diagonalisierbar. Also gibt es eine orthogonale Matrix T, so daß $T^\top Q T = \operatorname{diag}(\lambda_1, \ldots, \lambda_n) =: \Lambda$. Setze $R := T^\top \sqrt{\Lambda} T$. Dann ist R symmetrisch, positiv definit und es gilt $R^\top R = T^\top \sqrt{\Lambda} T T^\top \sqrt{\Lambda} T = Q$. Jede Matrix R mit den geforderten Eigenschaften hat die gleichen Eigenvektoren wie Q und die Eigenwerte $\sqrt{\lambda_i}$. Hierdurch ist R eindeutig bestimmt.
(c) Nach (b) gilt

$$E(Q,d) = d + \{y \mid y^\top R^{-1} R^{-1} y \leq 1\} = d + \{Rx \mid \|x\| \leq 1\} = d + RB^n.$$

(d) und (e) folgen aus (c) mit $V(A(X)) = \det A \cdot V(X)$, für affine Abbildungen A und meßbare Mengen X.
(f) E ist bis auf Translation und Drehung $\sqrt{\Lambda} \cdot B^n = \{x \mid (x_1/\sqrt{\lambda_1})^2 + \ldots + (x_n/\sqrt{\lambda_n})^2 \leq 1\}$.
$\square$

In dem Ellipsoidverfahren werden Ellipsoide fortgesetzt halbiert.

Definition 5.4 *Sei $E(Q,d)$ ein Ellipsoid und $a \in \mathbb{R}^n \backslash \{0\}$. Dann sei*

$$\mathrm{HE}\,(\mathrm{Q}, \mathrm{d}, \mathrm{a}) = \mathrm{E}(\mathrm{Q}, \mathrm{d}) \cap \{\mathrm{x} \in \mathbb{R}^\mathrm{n} \mid \mathrm{a}^\top \mathrm{x} \leq \mathrm{a}^\top \mathrm{d}\},$$

also eine Hälfte des Ellipsoids, das durch den Zentralschnitt geteilt wird.

Satz 5.5 *Sei $E = E(Q,d)$ Ellipsoid, $a \in \mathbb{R}^n \backslash \{0\}$ und*

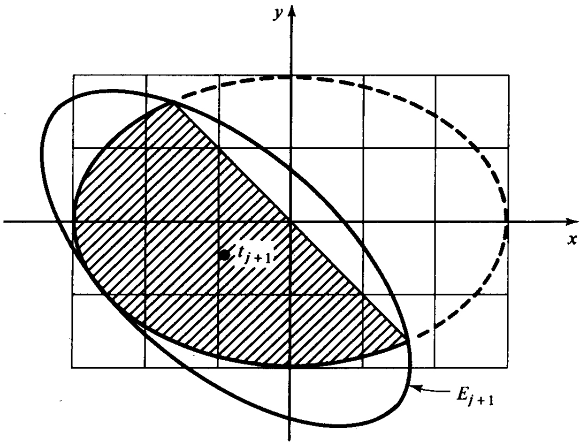
$$d' = d - \frac{1}{n+1} \frac{Qa}{\sqrt{a^\top Q a}} \quad \text{und} \quad Q' = \frac{n^2}{n^2-1} \left(Q - \frac{2}{n+1} \frac{(Qa)(Qa)^\top}{a^\top Q a} \right). \tag{5.3}$$

Dann gilt

(a) $E' = E(Q', d')$ ist ein Ellipsoid.

(b) $\mathrm{HE}(\mathrm{Q}, \mathrm{d}, \mathrm{a}) \subset \mathrm{E}'$.

(c) $V(E')/V(E) < e^{-1/(2(n+1))} < 1$.



Beweis: (i) Sei zunächst $Q = I,\ d = 0$ und $a = e^1$. Dann ist $d' = -e^1/(n+1)$ und

$$Q' = \frac{n^2}{n^2-1}\left(I - \frac{2}{n+1}\operatorname{diag}(1,0,\dots,0)\right) = \operatorname{diag}\left(\frac{n^2}{(n+1)^2}, \frac{n^2}{n^2-1}, \dots, \frac{n^2}{n^2-1}\right) =: W^2.$$

Offensichtlich ist Q' positiv definit, also ist E' ein Ellipsoid. Sei $x \in \mathrm{HE}(\mathrm{I}, 0, \mathrm{e}^1)$, d. h. $\|x\| \le 1$ und $x_1 \le 0$. Es ist zu zeigen, daß $(x-d')^\top Q^{-1}(x-d') \le 1$, d. h. $q := \|W^{-1}(x+e^1/(n+1))\|^2 \le 1$. Es gilt aber

$$\begin{aligned} q &= \|\left(\frac{(n+1)x_1+1}{n}, x_2\frac{\sqrt{n^2-1}}{n}, \dots, x_n\frac{\sqrt{n^2-1}}{n}\right)\|^2 \\ &= \frac{n^2-1}{n^2}(x_2^2+\dots+x_n^2) + \frac{(n+1)^2}{n^2}x_1^2 + \frac{2(n+1)x_1}{n^2} + \frac{1}{n^2} \\ &= \frac{n^2-1}{n^2}\|x\|^2 + \frac{1}{n^2} + \frac{2(n+1)(x_1^2+x_1)}{n^2} \le 1, \end{aligned}$$

wegen $-1 \le x_1 \le 0$. Das Volumen von E ist κ_n, das von E' ist $\kappa_n\sqrt{\det Q'} = \kappa_n \frac{n}{n+1}\left(\frac{n^2}{n^2-1}\right)^{(n-1)/2}$. Also

$$\frac{V(E')}{V(E)} = \left(1 - \frac{1}{n+1}\right)\left(1 + \frac{1}{n^2-1}\right)^{(n-1)/2}.$$

Mit $1 - x \le e^{-x}$ und $1 + x \le e^x$, für $x \ge 0$, ist der Quotient also nicht größer als $e^{-1/(n+1)}e^{1/(n^2-1)\cdot(n-1)/2} = e^{-1/(2(n+1))}$.

(ii) Nun wird der allgemeine Fall auf (i) zurückgeführt. O. E. sei $\|Ra\| = 1$. Es gibt eine Drehung T mit $Te^1 = Ra$. Man definiere die affine Abbildung U durch $U(x) = d + RTx$, dann gilt

(α) $E(Q, d) = U(E(I, 0)) = U(B^n)$.

(β) $E(Q',d') = U(E(W^2, -e^1/(n+1)))$.

(γ) $\mathrm{HE}\,(\mathrm{Q}, \mathrm{d}, \mathrm{a}) = \mathrm{U}(\,\mathrm{HE}\,(\mathrm{I}, 0, \mathrm{e}^1))$.

(α) folgt aus $E(Q,d) = d + RB^n = d + RTB^n = U(B^n)$.
(β) Sei $Q = RR = RTT^\top R$. Dann folgt

$$\begin{aligned}
Q' &= \frac{n^2}{n^2-1}\left(Q - \frac{2}{n+1} Qaa^\top Q\right)\\
&= \frac{n^2}{n^2-1}\left(RTT^\top R - \frac{2}{n+1} RTT^\top Raa^\top RTT^\top R\right)\\
&= \frac{n^2}{n^2-1}\left(RTT^\top R - \frac{2}{n+1} RT\,\mathrm{diag}\,(1,0,\ldots,0)T^\top R\right)\\
&= RT\left(\frac{n^2}{n^2-1}\left(I - \frac{2}{n+1}\,\mathrm{diag}\,(1,0,\ldots,0)\right)T^\top R\right)\\
&= RTW^2T^\top R = (RTW)(RTW)^\top.
\end{aligned}$$

Es folgt

$$\begin{aligned}
U(E(W^2, -e^1/(n+1))) &= d + RT(-e^1/(n+1)) + RTWB^n = d - \frac{Qa}{n+1} + RTWB^n\\
&= d' + RTWB^n = E(Q', d').
\end{aligned}$$

(γ) Durch Substitution $x = d + RTz$ erhält man

$$\begin{aligned}
\mathrm{HE}\,(\mathrm{Q}, \mathrm{d}, \mathrm{a}) &= \{x \mid x \in d + RB^n,\ a^\top x \le a^\top d\}\\
&= \{d + RTz \mid d + RTz \in d + RB^n,\ z^\top T^\top Ra \le 0\}\\
&= d + RT\{z \mid z \in B^n,\ z^\top e^1 \le 0\} = U(E(I, 0, e^1)).
\end{aligned}$$

Also ist $Q' = RTW(RTW)^\top$ positiv definit und $E(Q', d')$ ein Ellipsoid. Mit (i) folgt

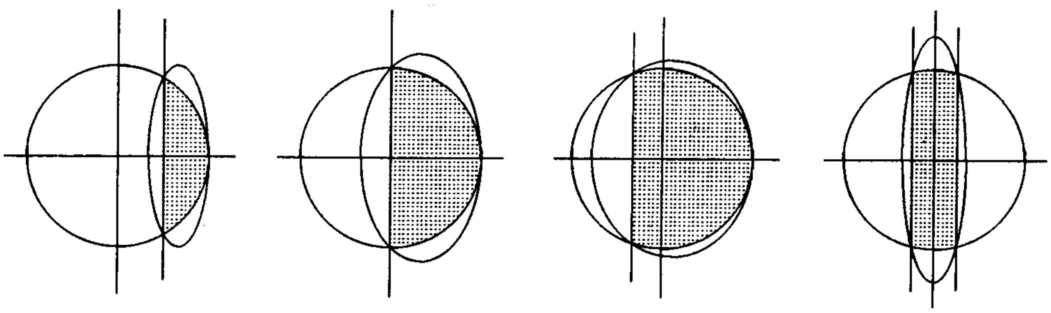
$$\mathrm{HE}\,(\mathrm{Q}, \mathrm{d}, \mathrm{a}) = \mathrm{U}(\,\mathrm{HE}\,(\mathrm{I}, 0, \mathrm{e}^1)) \subset \mathrm{U}(\mathrm{E}(\mathrm{W}^2, -\mathrm{e}^1/(\mathrm{n}+1)) = \mathrm{E}(\mathrm{Q}', \mathrm{d}')$$

und mit Satz 5.4 (e) gilt schließlich

$$\frac{V(E')}{V(E)} = \frac{V(U(E(W^2, -e^1/(n+1)))}{V(U(B^n))} < e^{-1/(2(n+1))}.$$

$\square$

Bemerkungen: (1) Das Ellipsoid in Satz 5.5 ist das sogenannte Löwner–John–Ellipsoid zu der Ellipsoidhälfte. Der Satz von Löwner–John besagt, daß es zu einer konvexen kompakten Menge K ein eindeutig bestimmtes (Löwner–John–) Ellipsoid E gibt, das K enthält und minimales Volumen hat. Ferner ist $1/nE$ in K enthalten und für zentralsymmetrisches K (d. h. $K = -K$) kann $1/n$ durch $1/\sqrt{n}$ ersetzt werden.
(2) Es gibt Varianten des Ellipsoid–Verfahrens, die mit ‘flachen’ oder ‘tiefen’ Schnitten arbeiten.



5.5 Das Ellipsoidverfahren

Gegeben sei das LSI: Finde eine Lösung von $Ax < b$ oder stelle fest, daß es keine Lösung gibt. Dabei sei $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$ und o. E. sei $m \geq n$. Die Größe des LSI ist $L = mn + \log(|Z|)$, wobei Z das Produkt aller Einträge $\neq 0$ ist. Die Menge der Lösungen eines LSI ist ein Polyeder P.
Die Idee des Ellipsoidverfahrens ist sehr einfach. Man konstruiert eine Folge von immer kleiner werdenden Ellipsoiden E_j, für die garantiert ist, daß sie eine Lösung enthalten (falls überhaupt eine existiert). Ist nach einer (nicht zu großen) Anzahl von Schritten der Mittelpunkt nicht in P enthalten, so wird garantiert, daß es keine Lösung gibt.
Zunächst wird eine große Kugel mit Radius $n2^L$ als Startellipsoid gewählt. Im folgenden wird $P' = P \cap (n2^L)B^n$ statt P betrachtet. Dieses P' ist, falls P nicht leer, nicht zu klein.

Lemma 5.5 *Ist $P \neq \emptyset$, so ist das Volumen von P' mindestens $2^{-(n+2)L}$.*

Beweis: Hat das LSI $Ax < b$ eine Lösung, so gibt es eine Ecke x^0 von $\{x \mid Ax \leq b\}$. Diese erfüllt mindestens $n \leq m$ der Ungleichungen mit Gleichheit. Sei $\tilde{A}$ die $(n \times n)$–Matrix aus den zugehörigen Zeilen und $\tilde{A}_i$ die Matrix $\tilde{A}$, in der die i–te Spalte durch b ausgetauscht wurde. Nach der Cramerschen Regel gilt $x_i^0 = \det \tilde{A}_i / \det \tilde{A}$ und weil die Determinanten ganzzahlig sind, folgt $|x_i| \leq \det \tilde{A}_i \leq n!|Z| < 2^{n^2}|Z| \leq 2^L$. Somit hat das Polytop $\tilde{P} = \{x \mid Ax \leq b, |x_i| \leq 2^L,\ i = 1, \ldots, n\}$ einen inneren Punkt und es gilt $\operatorname{aff}\tilde{P} = \operatorname{aff}\{x \mid x \text{ Ecke von } \tilde{P}\} = \mathbb{R}^n$. Es gibt daher $n+1$ affin unabhängige Ecken $v^0, \ldots, v^n$ von $\tilde{P}$. Sei S die konvexe Hülle dieser $n+1$ Ecken (ein Simplex), dann gilt $S \subset (n2^L)B^n$ und

$$V(P') \geq V(S) = \frac{1}{n!}\left|\det\begin{pmatrix} | & & | \\ v^1 - v^0 & \ldots & v^n - v^0 \\ | & & | \end{pmatrix}\right| = \frac{1}{n!}\left|\det\begin{pmatrix} 1 & 1 & \ldots & 1 \\ v^0 & v^1 & \ldots & v^n \end{pmatrix}\right| \neq 0$$

(Scherung in $\{x \mid x_1 = 1\}$). Die v^i sind Lösungen eines Linearen Gleichungssystems mit einer Matrix bestehend aus Zeilen von A und Einheitsvektoren. Wie oben gilt dann $v^i = u^i/D_i$, mit u^i ganzzahlig und D_i ganz und $\leq 2^L$. Also

$$V(S) \geq \frac{1}{n!}\frac{1}{D_0 \cdot \ldots \cdot D_n} \geq \frac{1}{n!}(2^L)^{-(n+1)} > (2^L)^{-(n+2)},$$

da $n! < 2^{n^2} \leq 2^L$. $\square$

Falls in einem Schritt $d_j \in P'$ gilt, so ist d_j eine Lösung. Andernfalls gibt es eine Ungleichung ${a_k}^\top x < b_k$, die von d_j verletzt wird. Dann ist P in dem Halbellipsoid $\mathrm{HE}(\mathrm{Q_j}, \mathrm{d_j}, \mathrm{a^k})$ und damit nach Satz 5.5 in $E_{j+1} := E({Q_j}', {d_j}')$ enthalten. Also $P' \subset E_{j+1} \subset E_j$. Andererseits fällt das Volumen der Ellipsoide nach Satz 5.3 schnell genug.

Lemma 5.6 *Ist $E_0 = E(Q_0, d_0)$ mit $Q_0 = n^2 2^{2L} \cdot I$ und $d_0 = 0$ das Startellipsoid und wird der Zentralschnitt $N = 10n(n+1)L$ mal angewandt, so gilt $V(E_N) < 2^{-(n+2)L}$.*

Beweis: Es ist $V(E_0) = \kappa_n (n2^L)^n \leq (2n)^n 2^{Ln}$, da B^n in dem Würfel mit Kantenlänge 2 enthalten ist. Nach Satz 5.5 (c) gilt

$$\frac{V(E_N)}{V(E_0)} < \left(e^{-\frac{1}{2(n+1)}}\right)^{10n(n+1)L} = e^{-5nL} < 2^{-5nL}.$$

Also $V(E_N) < (2n)^n 2^{-4nL} < 2^{-3nL} \leq 2^{-(n+2)L}$. □

Algorithmus 5.1 (Ellipsoidverfahren für LSI)

Input: Ein $(m \times n)$–LSI der Größe L.
Output: eine Lösung des LSI oder die Antwort 'Nein'.

begin
  $j := 0$;
  $d_0 := 0$;
  $Q_0 := n^2 2^{2L} \cdot I$;
  $N := 10n(n+1)L$;
  while $j \leq N$ **and** d_j ist keine Lösung von $Ax < b$ **do**
  begin
    bestimme eine Ungleichung aus $Ax < b$, die von d_j verletzt wird, etwa ${a_k}^\top d_j \geq b_k$;
    $a := a_k$;
    $d_{j+1} := d_j - \frac{1}{n+1} \frac{Q_j a}{\sqrt{a^\top Q_j a}}$;
    $Q_{j+1} := \frac{n^2}{n^2-1} \left(Q_j - \frac{2}{n+1} \frac{(Q_j a)(Q_j a)^\top}{a^\top Q_j a}\right)$;
    $j := j + 1$;
  end;
  if $j < N$ **then** d_j ist Lösung **else** 'Keine Lösung'.
end.

Satz 5.6 *Das Ellipsoidverfahren entscheidet korrekt, ob ein LSI eine Lösung besitzt.*

Beweis: Erfüllt im Laufe des Verfahrens ein d_j das LSI $Ax < b$, so hat man eine Lösung gefunden. Es breche das Verfahren mit $j = N$ ab. Dann gilt nach obigen Betrachtungen $P' \subset E_j$, für $j = 0, \ldots, n$, insbesondere $P' \subset E_N$. Wäre $P \neq \emptyset$, so würde nach Lemma 5.5 und Lemma 5.6 gelten $V(E_N) < V(P')$, Widerspruch. Also gibt es in diesem Fall keine Lösung. □

Der Start benötigt $O(mn)$ Schritte. Das gleiche trifft für alle Schritte in einer Iteration zu, von denen es höchstens $O(n^2 L)$ viele gibt. Insgesamt hat man die polynomiale Schranke $O(mn^3 L)$.

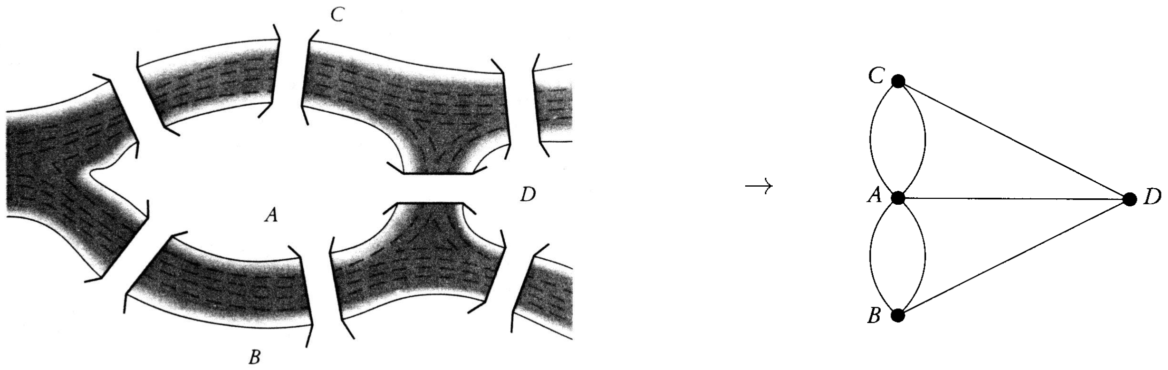
Es gibt aber Probleme bei der Implementierung, da in jedem Schritt eine Wurzel bestimmt werden muß. Diese ist i. a. nicht rational, es muß also gerundet werden. Es ist jedoch möglich, dies numerisch in den Griff zu bekommen, ohne daß die Polynomialität verletzt wird. Zusammen mit den Sätzen 5.2 und 5.3 erhalten wir also abschließend.

Korollar 5.1 *Es gibt einen polynomialen Algorithmus für LP.*

6 Kombinatorische Optimierung

6.1 Begriffe aus der Graphentheorie

Zur Beschreibung vieler Optimierungsprobleme eignet sich besonders die Sprache der Graphentheorie. Diese wurde 1736 von Euler ins Leben gerufen (Königsberger Brückenproblem):

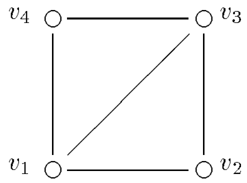


Es gibt keinen Rundweg, in dem alle Brücken genau einmal überschritten werden, denn sonst müsste von jeder Ecke eine gerade Anzahl von Kanten ausgehen.

Definition 6.1 Ein *Graph* G ist ein Paar $G = (V, E)$, wobei V eine endliche Menge ist und die Elemente von E zweielementige Teilmengen von V sind. V heißt die Menge der *Knoten* oder *Ecken* von G, die Elemente von E heißen *Kanten* von G.

Beispiel 6.1

$$G = (\{v_1, v_2, v_3, v_4\}, \{[v_1, v_2], [v_2, v_3], [v_3, v_4], [v_4, v_1], [v_1, v_3]\})$$

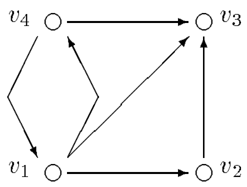


Beispielsweise durch Einbahnstraßen motiviert, wird die Richtung von Kanten beachtet:

Definition 6.2 Ein *gerichteter Graph* G ist ein Paar $G = (V, E)$, wobei V eine endliche Menge ist und $E \subset V \times V$. Die Elemente von E heißen *(gerichtete) Kanten* von G.

Beispiel 6.2

$$G = (\{v_1, v_2, v_3, v_4\}, \{(v_1, v_2), (v_2, v_3), (v_4, v_3), (v_4, v_1), (v_1, v_3), (v_1, v_4)\})$$



Definition 6.3 1. Ein *Kantenzug* der *Länge* $k - 1$ in einem Graphen ist eine Folge $w = (v_1, \ldots, v_k)$, $k > 1$ von Knoten, so daß $[v_j, v_{j+1}] \in E$ für $j = 1, \ldots, k - 1$ ist. Der Kantenzug heißt *geschlossen*, falls $v_k = v_1$ gilt. Der Kantenzug heißt *Weg* falls $v_i \neq v_j$ für $i \neq j$. Ein geschlossener Kantenzug mit $v_i \neq v_j$ für $i \neq j$ und $1 \leq i, j \leq k - 1$ heißt *Kreis*.

2. Ein Graph heißt *zusammenhängend*, wenn es zu jedem Paar (v_i, v_j) einen Kantenzug gibt, der v_i und v_j enthält.

3. Ein Graph heißt *Baum*, falls er zusammenhängend ist und keinen Kreis enthält.

4. Ein Graph $G_1 = (V_1, E_1)$ heißt *Teilgraph* von $G = (V, E)$, falls $V_1 \subset V$ und $E_1 \subset E$ gilt.

5. T heißt *spannender Baum* von G, falls T Teilgraph von G ist, die Knotenmengen gleich sind und T ein Baum ist.

6. Ein Graph $G = (V, E)$ zusammen mit einer Funktion $w : E \to \mathbb{R}$ heißt *gewichteter Graph*. Das Gewicht der Kante $[u, v]$ wird mit $w(u, v)$ oder w_{uv} bezeichnet. (Die Gewichte haben häufig passende Namen wie Kosten oder Abstände.)

Anmerkungen:

1. Die obigen Definitionen von Weg und Kreis lassen sich unmittelbar auf gerichtete Graphen übertragen. Man definiert in diesem Fall gerichtete Wege und gerichtete Kreise, d.h. die Richtung des Weges bzw. Kreises muß mit der Richtung der Kanten übereinstimmen.

2. Ein Kantenzug (Weg, Kreis) läßt sich auch als Folge $(e_1, \ldots, e_n)$ von Kanten darstellen.

3. Für einen gewichteten Graphen $G = (V, E)$ und einen Weg $(v_1, \ldots, v_n)$ bezeichnet man $\sum_{i=1}^{n-1} w(v_i, v_{i+1})$ auch als Länge des Weges. Die obige Längendefinition ergibt sich durch Gewichte der Größe 1.

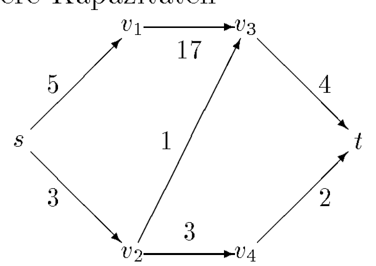
Beispiel 6.3 (Traveling SalesmanProblem (TSP)) In einem Beispiel des TSP ist der Abstand zwischen jedem Paar von n Städten gegeben. Gesucht ist eine Rundreise minimaler Gesamtlänge. Die symmetrische $n \times n$ Abstandsmatrix (d_{ij}) kann als gewichteter vollständiger Graph $G = (\{v_1, \ldots, v_n\}, K_n)$ mit $K_n = \{[v_i, v_j], 1 \leq i < j \leq n\}$ aufgefaßt werden. Die Menge F der zulässigen Punkte ist dann die Menge aller Kreise der Länge n.

Beispiel 6.4 (Minimal spanning tree (MST)) Sei G der Graph aus dem vorigen Beispiel. Gesucht ist ein spannender Baum, für den die Summe der Kantenlängen minimal ist. Also

$$F = \{\text{alle spannenden Bäume } (V, E) \text{ mit } V = \{v_1, \ldots, v_n\}\},$$
$$c : (V, E) \to \sum_{[v_i, v_j] \in E} d_{ij}$$

Definition 6.4 Ein *Netzwerk* $N = (s, t, V, E, c, b)$ ist ein gerichteter Graph (V, E) mit einer *Quelle* $s \in V$, so daß $(x, s) \notin E$ für $x \in V$, einem *Ziel* oder *Senke* $t \in V$, so daß $(t, x) \notin E$ für $x \in V$, einer *unteren Kapazität* $c : E \to \mathbb{R}$ und einer *oberen Kapazität* $b : E \to \mathbb{R}$, so daß $c \leq b$ gilt. Ist die untere Kapazität nicht angegeben, so ist sie identisch Null.

Beispiel 6.5 Netzwerk ohne untere Kapazitäten



Definition 6.5 Ein *Fluß* in einem Netzwerk N ist eine Abbildung $f : E \to \mathbb{R}$ (bzw. ein Vektor des $\mathbb{R}^{|E|}$) mit

(1) $c(u,v) \le f(u,v) \le b(u,v)$ für jedes $(u,v) \in E$.

(2) $\sum_{(u,v)\in E} f(u,v) = \sum_{(v,u)\in E} f(v,u)$ für alle $v \in V \setminus \{s,t\}$.

Der *Wert* $|f|$ des Flusses ist die Grösse $|f| = \sum_{(s,v)\in E} f(s,v)$.

6.2 Das Problem des kürzesten Wegs

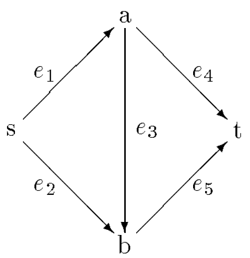
Definition 6.6 Für einen gerichteteten, gewichteten Graphen ist ein Beispiel des Problems der *kürzesten Wege (shortest path problem, SP)* einen gerichteten Weg minimaler Länge von einem ausgezeichneten Quellknoten s zu einem ausgezeichneten Zielknoten t zu finden. Die minimale Länge wird mit $d(s,t)$ bezeichnet.

Als Optimierungsproblem ist die zulässige Menge
$F = \{P = (e_{j_1}, \dots, e_{j_k}) \mid P \text{ ist ein gerichteter Weg von } s \text{ nach } t\}$
und die Kostenfunktion $c(P) = \sum_{i=1}^{k} c_{j_i}$.
Zur Formulierung von SP als LP wird die (Knoten–Kanten–) Inzidenzmatrix $A = (a_{ij})$ des Graphen G durch

$$a_{ij} = \begin{cases} +1 & \text{wenn die Kante } e_j \text{ aus dem Knoten } i \text{ herausführt} \\ -1 & \text{wenn die Kante } e_j \text{ in den Knoten } i \text{ hineinführt} \\ 0 & \text{sonst} \end{cases}$$

$i = 1, \dots, |V|, \quad j = 1, \dots, |E|$

Beispiel 6.6 G :



$$A = \begin{array}{c} \\ s \\ t \\ a \\ b \end{array} \begin{array}{c} \begin{array}{ccccc} e_1 & e_2 & e_3 & e_4 & e_5 \end{array} \\ \left| \begin{array}{ccccc} +1 & +1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 \\ -1 & 0 & +1 & +1 & 0 \\ 0 & -1 & -1 & 0 & +1 \end{array} \right| \end{array}$$

$c(e_1) = c(e_5) = 1, \ c(e_2) = c(e_3) = 2, \ c(e_4) = 3.$

Mit 1 als trivialen Kapazitäten ist ein Weg ein spezieller Fluß mit etwa $s^\top f = -t^\top f = 1$. Die Flußbedingungen ergeben daher für einen Weg das Gleichungssystem

$$Af = (1, -1, 0, \dots, 0)^\top,$$

wobei die "1" zu s und die "-1" zu t gehört. Im Allgemeinen sind die Koeffizienten von f keine ganzen Zahlen, aber für den Fluß mit den geringsten Kosten

$$\min\{c^\top f \mid Af = (1, -1, 0, \dots, 0)^\top, \ f \ge 0\}$$

existiert eine optimale Lösung, in der jedes f_i 0 oder 1 ist und so einen kürzesten Weg von s nach t in G darstellt (siehe Übung).
Weiter sind die $|V|$ Gleichungen des Problems redundant, da die Flußerhaltung an $|V| - 1$ Knoten die Flußerhaltung am ausgelassenen Knoten sichert. Man kann daher eine Gleichung z.B. für t auslassen.

Beispiel 6.6 (Forts.) Weglassen von t ergibt das Tableau:

0	1	2	2	3	1
1	1	1	0	0	0
0	-1	0	1	1	0
0	0	-1	-1	0	1

Addition der Zeile 1 zur Zeile 2 ergibt die Basis $\{1,4,5\}$. Das Starttableau wird so:

-4	0	-1	0	0	0
1	1	1	0	0	0
1	0	1	1	1	0
0	0	-1	-1	0	1

Eine Basis stellt eine Menge von $|V|-1$ Kanten dar, von denen eine Teilmenge einem Weg von s nach t mit den gegebenen Kosten entspricht. Die Kanten, die nicht auf dem Weg liegen, stellen entartete Komponenten der Basis dar. Im Beispiel stellt die Basis den Weg (e_1, e_4) mit den Kosten 4 und die entartete Kante e_5 dar.

Beispiel 6.6 (Forts.) Ein Pivotschritt mit ξ_{22} ergibt das optimale Tableau

-3	0	0	1	1	0
0	1	0	-1	-1	0
1	0	1	1	1	0
1	0	0	0	1	1

Dies entspricht dem optimalen Weg (e_2, e_5) mit Kosten 3 und der entarteten Kante e_1. Eine Kante ist zur Basis hinzugefügt worden und eine andere weggenommen, um einen neuen Weg mit geringeren Kosten zu finden.

Für den gerichteten Graph $G = (V, E)$ mit der nichtnegativen Gewichtsfunktion c und $v \in V$ wird die Länge der kürzesten Wege von v zu jedem anderen $u \in V$ gesucht. O.B.d.A. sei für jedes $u, w \in V$ $(u, w) \in E$ (setze gegebenenfalls $c_{uw} = \infty$).
Für $W \subset V$ und $x \in V$ bezeichne $\rho_W(x)$ die Länge des kürzesten Weges von v nach x, der nur Punkte aus W als Zwischenpunkte benutzt. Dies wird sukzessive bestimmt durch

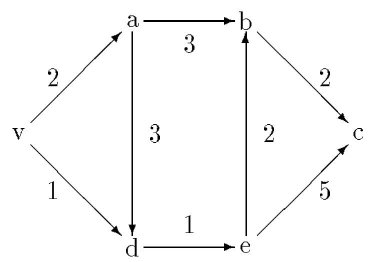
Algorithmus 6.1 (Dijkstras Algorithmus)
Eingabe: Ein gerichteter Graph $G = (V, E)$, $c : E \to \mathbb{R}$, $v \in V$.
Ausgabe: Die Länge der kürzesten Wege von v zu den übrigen Knoten.
begin
  $W := \{v\}$; $\rho_W(v) := 0$;
  for all $y \in V - \{v\}$ **do** $\rho_W(y) := c_{vy}$;
  while $W \neq V$ **do**
  begin
    finde x mit $\rho_W(x) = \min\{\rho_W(y) \mid y \in V \setminus W\}$;
    $W := W \cup \{x\}$;
    for all $y \in V \setminus W$ **do**

```
      begin
        ρ_W(y) := min{ρ_W(y), ρ_W(x) + c_xy};
      end;
    end;
end.
```

Satz 6.1 *Dijkstras Algorithmus bestimmt die Länge der kürzesten Wege von v zu den übrigen Knoten korrekt.*

Beweis: Mit Induktion nach der Schrittzahl wird zeigt, daß $\rho_W(y)$ den kürzesten Weg von v nach y mit Zwischenpunkten aus W angibt.
Wird x in W aufgenommen, so ist $\rho_W(x)$ nach Induktionsvoraussetzung die Länge des kürzesten Weges in W nach x. Ferner sei $W' := W \cup \{x\}$.
Angenommen, der kürzeste Weg (überhaupt) nach x enthielte Ecken, die nicht in W liegen. Sei etwa $y \notin W$ die erste solche Ecke auf dem kürzesten Weg. Nach Wahl von x hat der Weg schon bei y eine Länge, die nicht kleiner ist als $\rho_W(x)$. Also ist $\rho_W(x)$ die Länge des kürzesten Weges nach x überhaupt.
Zur Berechnung von $\rho_{W'}(y)$: Entweder, ein kürzester Weg in W' nach y führt nicht über x, dann gilt $\rho_{W'}(y) = \rho_W(y)$.
Führen aber alle über x, dann gehen sie von x aus direkt nach y, da sonst eine Ecke $w \in W$ dazwischen liegt. Der kürzeste Weg (überhaupt) nach w liegt aber nach Induktionsvoraussetzung ganz in W, d.h. es gibt dann doch einen kürzesten Weg nach y, der nicht über x führt. Widerspruch. Darum ist $\rho_{W'}(y) = \rho_W(x) + c_{xy}$. □

Beispiel 6.7



(0) $W = \{v\},\ \rho(v) = 0,$
$\rho(d) = 1,\ \rho(a) = 2,\ \rho(b) = \rho(e) = \rho(c) = \infty.$

(1) Markiere d. $W = \{v, d\},\ \rho(v) = 0,\ \rho(d) = 1,$
$\rho(e) = 1 + 1,\ \rho(a) = 2,\ \rho(b) = \rho(c) = \infty.$

(2) Markiere e. $W = \{v, d, e\},\ \rho(v) = 0,\ \rho(d) = 1,\ \rho(e) = 2,$
$\rho(b) = 2 + 2,\ \rho(c) = 2 + 5,\ \rho(a) = 2.$

(3) Markiere a. $W = \{v, d, e, a\},\ \rho(v) = 0,\ \rho(d) = 1,\ \rho(e) = 2,\ \rho(a) = 2,$
$\rho(b) = \min\{4, 2 + 3\} = 4,\ \rho(c) = 7.$

(4) Markiere b. $W = \{v, d, e, a, b\},\ \rho(v) = 0,\ \rho(d) = 1,\ \rho(e) = 2,\ \rho(a) = 2,\ \rho(b) = 4,$
$\rho(c) = \min\{7, 4 + 2\} = 6.$

Somit gilt $d(v, v) = 0,\ d(v, d) = 1,\ d(v, c) = d(v, a) = 2,\ d(v, b) = 4,\ d(v, c) = 6.$

Aufwand des Algorithmus: Für $|V| = n$ erfordert die Initialisierung $n+1$ Operationen. Die Schleife wird $n-1$ mal durchlaufen und erfordert jeweils $O(n)$ Operationen. Damit ist der Gesamtaufwand $O(n^2)$ Operationen.

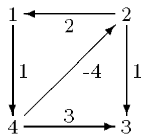
Der folgende Floyd–Warshall–Algorithmus arbeitet auch mit negativen Gewichten und erkennt Kreise negativer Länge. Er berechnet die Abstände aller Knoten gleichzeitig. Gegeben sei ein vollständiger gerichteter Graph als $n \times n$–Matrix (c_{ij}), wobei $c_{ii} = \infty$ für alle i gesetzt ist. Für einen Kantenzug $\omega = (v_1, \dots, v_s)$ in einem (gerichteten) gewichteten Graphen bezeichnet $l(\omega)$ die Länge von ω.

Definition 6.7 Sei $(d_{ij}) \in \mathbb{R}^{n \times n}$. Eine *Dreiecksoperation* für $j \in \{1, \dots, n\}$ ist definiert durch

$$d_{ik} = \min\{d_{ik}, d_{ij} + d_{jk}\}$$

für alle $i, k \in \{1, \dots, n\} \setminus \{j\}$ ($k = i$ ist also möglich).

Beispiel 6.8



$$(d_{ij}) = \begin{pmatrix} \infty & \infty & \infty & 1 \\ 2 & \infty & 1 & \infty \\ \infty & \infty & \infty & \infty \\ \infty & -4 & 3 & \infty \end{pmatrix}.$$

Dreiecksoperation mit $j = 4$ ergibt $(d'_{ij}) = \begin{pmatrix} \infty & \mathbf{-3} & \mathbf{4} & 1 \\ 2 & \infty & 1 & \infty \\ \infty & \infty & \infty & \infty \\ \infty & -4 & 3 & \infty \end{pmatrix}.$

Satz 6.2 *Sei $G = (V, E)$ ein gerichteter gewichteter Graph. Existiert in G ein geschlossener Kantenzug negativer Länge, so existiert in G ein Kreis negativer Länge.*

Beweis: Sei $\omega = (v_1, \dots, v_{n-1}, v_1)$ ein Kantenzug negativer Länge mit einer Minimalzahl von Kanten. Angenommen, eine Ecke $v_i, i \neq 1$, käme doppelt vor. Dann ist der Ausschnitt $(v_i, \dots, v_i)$ ein geschlossener Kantenzug. Hat er negative Länge, dann haben wir einen Widerspruch zur Wahl von ω. Hat er nichtnegative Länge, dann können wir ihn eliminieren, und der restliche Kantenzug hat dann negative Länge und weniger Kanten als ω. Widerspruch. $\square$

Satz 6.3 *Sei $G = (V, E)$ ein gerichteter gewichteter Graph, der keinen Kreis negativer Länge enthält. Dann gilt für jeden Kantenzug $\omega = (v_1, \dots, v_k) \quad l(\omega) \geq d(v_1, v_k)$.*

Beweis: Annahme: Sei $l(\omega) < d(v_1, v_k)$ und die Eckenzahl von ω sei minimal. Dann ist ω kein Weg. Daher existiert ein $v_i \in \omega$ mit $\omega = (v_1, \dots, v_i, \dots, v_{i+l} = v_i, \dots, v_k)$ und $v_p \neq v_q$ für $i \leq p, q \leq i + l - 1, p \neq q$. Dann ist $\kappa = (v_i, \dots, v_{i+l})$ ein Kreis, $\bar{\omega} = (v_1, \dots, v_i, v_{i+l+1}, \dots, v_k)$ ein Kantenzug mit weniger Ecken als ω und $l(\bar{\omega}) = l(\omega) - l(\kappa) \leq l(\omega) < d(v_1, v_k)$. Widerspruch.

$\square$

Satz 6.4 *In (d_{ij}) werde für $k = 1, \dots, n$ nacheinander eine Dreiecksoperation durchgeführt. Das Ergebnis nach k Schritten werde mit d_{ij}^k bezeichnet.*
1) Es gibt einen Kreis negativer Länge genau dann, wenn $d_{ii}^k < 0$ für ein i und ein k gilt.
2) Gilt $d_{ii}^k \geq 0$ für alle i und k, so ist d_{ij}^k die Länge des kürzesten Weges von i nach j, der als Zwischenknoten nur Elemente aus $\{1, \dots, k\}$ benutzt.

Beweis: Mit vollständiger Induktion wird gezeigt:
Gilt $d_{ii}^r \geq 0$, für alle i und $r = 0, 1, \dots, k$, so ist d_{ij}^k die Länge des kürzesten Weges von i nach j, der nur Zwischenknoten aus $\{1, \dots, k\}$ benutzt. (Im Falle $i = j$ ist dies die Länge des kürzesten Kreises, der i und sonst nur Knoten aus $\{1, \dots, k\}$ enthält.)
Der Induktionsanfang $k = 0$ ist klar. Sei die obige Behauptung bewiesen für $k-1$. Für $i, j \in V$ sei jetzt $\omega = (i, \dots, j)$ der kürzeste Weg, der i und j verbindet und Zwischenknoten nur in $\{1, \dots, k\}$ hat.

(i) ω enthalte nicht k. Seien $\omega_1 = (i, \dots, k)$ und $\omega_2 = (k, \dots, j)$ die zu d_{ik}^{k-1} bzw. d_{kj}^{k-1} gehörigen Wege und $\omega' = \omega_1 + \omega_2$. Angenommen, $l(\omega') < l(\omega)$. Dann enthält ω' einen Kreis κ negativer Länge (vgl. Beweis von Satz 6.3). κ enthält k und die übrigen Knoten sind aus $\{1, \dots, k-1\}$. Sei s eine dieser Ecken aus κ, dann folgt $d_{ss}^k \leq d_{sk}^{k-1} + d_{ks}^{k-1} \leq l(\kappa) < 0$. Widerspruch. Also wird $d_{ij}^{k_0} = d_{ij}^{k_0-1}$ gesetzt.

(Im Fall von Kreisen negativer Länge ist es möglich, daß ω' kleinere Länge hat und $d_{ij}^k = d_{ik}^{k-1} + d_{kj}^{k-1}$ gesetzt wird. Dann kann also d_{ij}^k Kantenzüge mit Kreisen repräsentieren).

(ii) Ist k Zwischenknoten von ω, so sind beide Teilstücke kürzeste Wege und es gilt $d_{ij}^k = d_{ik}^{k-1} + d_{kj}^{k-1}$.

Insbesondere ist (2) gezeigt. Nun wird (1) gezeigt:
Es gelte erstmalig $d_{ii}^k < 0$, für ein i. Dann gilt $d_{ik}^{k-1} + d_{ki}^{k-1} < 0$. Seien ω_1 und ω_2 die zu d_{ik}^{k-1} bzw. zu d_{ki}^{k-1} gehörigen Wege. Dann ist $\omega_1 + \omega_2$ ein geschlossener Kantenzug negativer Länge und nach Satz 6.2 gibt es damit auch einen Kreis negativer Länge.
Es existiere umgekehrt ein Kreis κ negativer Länge und angenommen, es gelte $d_{ii}^k \geq 0$, für alle i und k. Sei r der höchste vorkommende Index und s der zweithöchste. Dann gilt $d_{rr}^s \leq d_{rs}^{s-1} + d_{sr}^{s-1} \leq l(\kappa) < 0$. Widerspruch. □

Nach Satz 6.4 liefert daher eine Anwendung von Dreiecksoperationen entweder einen Kreis negativer Länge oder nach n Schritten die Abstandsmatrix. Zur Bestimmung der Wege dient die Matrix (e_{ij}) in der für den kürzesten Weg von i nach j der Vorgänger von j eingetragen wird.

Algorithmus 6.2 (Floyd–Warshall–Algorithmus)
Eingabe: Eine $(n \times n)$ Abstandsmatrix (c_{ij})
Ausgabe: Feststellung eines Kreises negativer Länge oder die Abstandsmatrix d_{ij} des vollständigen Graphen, sowie für jedes i, j der Vorgänger von j auf dem kürzesten Weg von i nach j

begin
  for all i **do**
  begin
    $d_{ij} := c_{ij}$; $e_{ij} := i$; $d_{ii} := \infty$;
  end;
  Negativer_Kreis := 'Nein';
  $j := 1$;
  while (Negativer_Kreis = 'Nein') **and** $j \leq n$ **do**
  begin
    for $k = 1$ **to** n **do**
    for $l = 1$ **to** n **do**
    begin
      if $d_{kl} > d_{kj} + d_{jl}$ **then**
      begin
        $d_{kl} := d_{kj} + d_{jl}$;
        $e_{kl} := e_{jl}$;
        if $(k = l)$ **and** $d_{kk} < 0$ **then** Negativer_Kreis := 'Ja';
      end;
    end;
    $j := j + 1$;
  end;
end.

Beispiel 6.8 (Forts.)

$$(d_{ij}^0) = \begin{pmatrix} \infty & \infty & \infty & 1 \\ 2 & \infty & 1 & \infty \\ \infty & \infty & \infty & \infty \\ \infty & -4 & 3 & \infty \end{pmatrix} \qquad (e_{ij}^0) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{pmatrix}$$

$$(d_{ij}^1) = \begin{pmatrix} \infty & \infty & \infty & 1 \\ 2 & \infty & 1 & \mathbf{3} \\ \infty & \infty & \infty & \infty \\ \infty & -4 & 3 & \infty \end{pmatrix} \qquad (e_{ij}^1) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & \mathbf{1} \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{pmatrix}$$

$$(d_{ij}^2) = \begin{pmatrix} \infty & \infty & \infty & 1 \\ 2 & \infty & 1 & 3 \\ \infty & \infty & \infty & \infty \\ \mathbf{-2} & -4 & \mathbf{-3} & \mathbf{-1} \end{pmatrix} \qquad (e_{ij}^2) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 \\ 3 & 3 & 3 & 3 \\ \mathbf{2} & 4 & \mathbf{2} & \mathbf{1} \end{pmatrix}$$

Abschätzung des Aufwands für den Floyd–Warshall–Algorithmus: Die beiden 'for'–Schleifen erfordern einen Aufwand von $O(n^2)$, die 'while'–Schleife wird höchstens n Mal durchlaufen. Damit ist der Gesamtaufwand $O(n^3)$.

6.3 Minimal spannende Bäume

Definition 6.8 Gegeben sei die symmetrische $|V| \times |V|$–Abstandsmatrix positiver Zahlen c_{ij}. Dann ist das Problem *minimaler spannender Bäume* (minimal spanning tree, MST) die Suche nach einem kürzesten spannenden Baum des vollständigen Graphen $K_{|V|}$ bezüglich der Abstände c_{ij}.

Bezeichnungen: Eine endliche Vereinigung $F = \{(U_1, T_1), \ldots, (U_n, T_n)\}$ eckendisjunkter Bäume heißt *Wald*. Ein Wald F *spannt* den Graphen $G = (V, E)$ auf, falls $V = U_1 \cup \ldots \cup U_n$.
Für einen Baum T sei $c(T) = \sum_{[u,v] \in T} c(u, v)$ die Länge des Baums.
Die Algorithmen zur Lösung von MST benutzen

Satz 6.5 *Sei $\{(U_1, T_1), \ldots, (U_n, T_n)\}$ ein Wald, der V aufspannt, und $v \in U_1, u \notin U_1$ mit $c(v, u) = \min\{c(x, y) \mid x \in U_1, y \notin U_1\}$. Dann ist unter allen Bäumen, die alle Kanten aus $T = \bigcup_{j=1}^n T_j$ enthalten, ein minimaler, der $[v, u]$ enthält.*

Beweis: Es sei $\tilde{T} = (V, F)$ ein spannender Baum, der T enthält und minimale Kosten hat. Gilt $[v, u] \in F$, so sind wir fertig, also sei $[v, u] \notin F$. In $F \cup [v, u]$ gibt es einen Kreis K, der $[v, u]$ enthält. Wegen $u \notin U_1$ existiert eine weitere Kante $[v', u'] \neq [v, u]$ auf K mit $u' \in U_1, v' \notin U_1$ und damit $[v', u'] \notin T$. Nach Wahl von $[v, u]$ gilt $c(v, u) \leq c(v', u')$. (V, F') mit $F' = F + [v, u] - [v', u']$ ist ein spannender Baum, der T enthält mit $[v, u] \in F'$ und $c(F') \leq c(F)$. □

Der Satz liefert folgenden

Algorithmus 6.3 (MST1)
Eingabe: V und die Abstände c_{ij};
Ausgabe: Der kürzeste spannende Baum (V, T)

```
begin
   (* Initialsierung *)
   U := {v1};
   T := ∅;
   for all v ∈ V − {v1} do Nächster[v] := v1;
   while U ≠ V do
   begin
      (* Finde die zu U benachbarte Ecke *)
      min := ∞;
      for all v ∈ V − U do
      begin
         if c(v,Nächster[v]) < min then
         begin
            min := c(v,Nächster[v]);
            Neue_Ecke := v;
         end;
      end;
      U := U∪ Neue_Ecke;
      T := T ∪ { [Neue_Ecke, Nächster[Neue_Ecke]]};
```

```
      (* Nächster aktualisieren *)
      for all v ∈ V − U do
      begin
         if c(v, Nächster[v])> c(v, Neue_Ecke) then Nächster[v] := Neue_Ecke;
       end;
   end;
end.
```

Nächster vermeidet überflüssige Arbeit bei Suche nach der kürzesten Kante, die aus U herausgeht.

Satz 6.6 *MST1 löst MST in einer Zeit $O(|V|^2)$.*

Beweis: Es sei B das Ergebnis des Algorithmus. Da immer nur Kanten hinzugefügt werden, die aus U herauslaufen, treten keine Kreise auf. Wegen $U = V$ bei Abbruch ist B ein spannender Baum. Durch Induktion nach der Mächtigkeit von U folgt die Existenz eines optimalen spannenden Baumes, der $T(U)$ enthält: Dies ist sicher der Fall für $U = \{v_1\}$ und $T = \emptyset$. Sei die Behauptung richtig für $|U| = j, 1 \leq j < |V|$. Es seien $v_1, \ldots, v_r$ die Ecken aus $V \backslash U$, dann ist $\{(U, T(U)), (\{v_1\}, \emptyset), \ldots, (\{v_r\}, \emptyset)\}$ ein spannender Wald. Nach Satz 6.5 existiert unter allen Bäumen, die $T(U)$ enthalten, ein minimaler Baum $\hat{T}$, der die kürzeste Kante e enthält, die aus U herausgeht. Nach Induktionsannahme gibt es einen (global) minimal spannenden Baum T^*, der $T(U)$ enthält. Es gilt $c(T^*) \leq c(\hat{T})$, da T^* global minimal und $c(\hat{T}) \leq c(T^*)$, da T^* bei der Minimierung vorkommt, bzgl. derer $\hat{T}$ minimal ist. Somit gilt $c(\hat{T}) = c(T^*)$ und $\hat{T}$ ist ein minimal spannender Baum, der $T(U) + e$ enthält. Also gilt die Behauptung auch für die nächste Stufe.
Zur Komplexität beachte man, daß der Algorithmus $|V| - 1$ Stufen durchläuft, in denen je eine Kante hinzugefügt wird. Die Initialisierung benötigt $O(|V|)$ Schritte und das Auffinden der neuen Ecke und das Aktualisieren von Nächster benötigt ebenfalls je $O(|V|)$ Schritte. □

MST1 ist im Allgemeinen bestmöglich, da für MST jede Kante betrachtet werden muß. Für Graphen mit wenigen Kanten ($c(u, v) = \infty$ für viele Kanten) läßt sich MST1 durchaus verbessern. Der Algorithmus MST2 benutzt eine Hilfsprozedur zur Bestimmung der sogenannten Zusammenhangskomponenten. In einem ungerichteten Graphen $G = (V, E)$ sind dies die Äquivalenzklassen bezüglich der Äquivalenzrelation

$u \sim v \Leftrightarrow$ es gibt einen Weg von u nach v

auf der Menge der Ecken. Die Zusammenhangskomponenten können mit der folgenden Modifikation des Verfahrens von Dijkstra bestimmt werden.

Algorithmus 6.4 (Zusammenhangskomponenten)
Eingabe: Ein (ungerichteter) Graph $G = (V, E)$.
Ausgabe: Die Zusammenhangskomponenten $S_1, \ldots, S_k$ von G.

```
begin
   i := 1;
   while V ≠ ∅ do
   begin
      s := erste Ecke von V;
      Z := {s};
      Liste := {s};
      while Liste ≠ ∅ do
      begin
         v := erste Ecke von Liste;
         für alle w, die zu v benachbart sind do
         if w ∉ Z then
         begin
            Z := Z ∪ {w};
            Liste := Liste ∪{w};
          end;
         Liste := Liste \{v};
       end;
      S_i := Z;
      V := V\Z;
      i := i + 1;
    end;
 end.
```

Satz 6.7 *Der Algorithmus 6.4 bestimmt die Zusammenhangskomponenten von $G = (V, E)$ in einer Zeit von $O(|E|)$.*

Beweis: Es reicht zu zeigen, daß innerhalb der inneren While–Schleife die Zusammenhangskomponente von s bestimmt wird.
Gibt es keinen Weg von s nach v, so kann v nicht in Z aufgenommen werden (sonst gäbe es einen Weg).
Angenommen, es gibt eine Ecke v, die mit s verbunden ist und nicht in Z aufgenommen wird. Sei der Abstand von v zu s dabei minimal gewählt. v ist zu einem w benachbart, das einen kürzeren Abstand zu s hat und damit in Z liegt. Dann war aber w auch in Liste und damit wird v in Z aufgenommen, wenn w bearbeitet wird. Widerspruch.
Zur Komplexität beachte man: Jede Kante $[v, w]$ wird im Verfahren höchstens zweimal betrachtet (beim Bearbeiten von v und von w). Darum ist der Aufwand insgesamt $O(|E|)$. $\square$

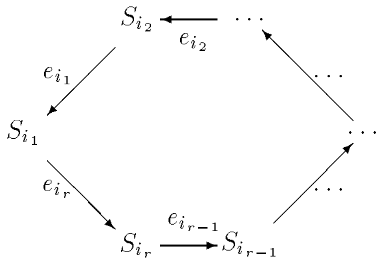
Der Algorithmus MST2, der ebenfalls auf Satz 6.5 beruht, arbeitet in Stufen: Auf jeder Stufe werden die Zusammenhangskomponenten $S_1, \ldots, S_k$ von (V, T) mit Algorithmus 6.4 bestimmt und anschließend jeder Komponente eine neue Kante hinzugefügt. Zur Bestimmung dieser Kanten dient das Array KÜRZESTER.

Algorithmus 6.5 (MST2)
Eingabe: Ein zusammenhängender Graph $G = (V, E)$ und für jedes $[u, v] \in E$ ein Abstand $c(u, v)$.
Ausgabe: Der kürzeste spannende Baum (V, T) von G.

begin
  (* Initialisierung; C ist die Menge der Zusammenhangskomponenten von (V, T) *)
  $T := \emptyset$; $C := \{\{v_1\}, \ldots, \{v_n\}\}$;
  while $|C| \neq 1$ **do**
  begin
    for all $S \in C$ **do** min [S] := ∞;
    for all $[u, v] \in E$ **do**
    begin
      S1, S2 := Komponenten, die v und u enthalten;
      if S1 $\neq$ S2 **then**
      begin
        if $c(u, v)$ <min[S1] **then**
        begin
          min [S1] := $c(u, v)$;
          Kürzester[S1] := $[u, v]$;
        end;
        if $c(u, v) <$ min[S2] **then**
        begin
          min[S2] := $c(u, v)$;
          Kürzester[S2] := $[u, v]$;
        end;
      end;
    end;
    for all $S \in C$ **do** $T := T \cup \{$ Kürzester[S] $\}$;
    finde die Menge C der Zusammenhangskomponenten von (V, T);
  end;
end.

Satz 6.8 *MST2 löst MST für (V, E) und c in einer Zeit $O(|E| \log |V|)$.*

Beweis: Eine Ausführung der while–Schleife heiße eine Stufe. In einer Stufe gebe es k Komponenten und $e_1, \ldots, e_k$ seien die zugehörigen kürzesten Kanten aus $S_1, \ldots, S_k$. O. E. sei $c(e_1) \leq \ldots \leq c(e_k)$. e_k ist die kürzeste Kante aus S_k heraus. Damit folgt nach Satz 6.5, daß die Hinzunahme erlaubt ist. Nun werden die Komponenten S_k und $S_{k'}$, die e_k verbindet zusammengefasst. $e_{k'}$ ist dann die kürzeste Kante aus $S_k \cup S_{k'}$ heraus. Nach Satz 6.5 ist die Hinzunahme erlaubt, usw. Das Verfahren bricht ab, wenn eine Kante die kürzeste für beide angrenzenden Komponenten ist, oder ein Kreis entsteht. Tritt der zweite Fall nicht auf, so kann man entsprechend fortfahren, bis alle Kanten hinzugefügt sind. Es bleibt also zu zeigen, daß keine Kreise auftreten. Angenommen, doch. Dann liegt folgende Situation vor:



Es folgt dann $c(e_{i_1}) \geq c(e_{i_2}) \geq \ldots \geq c(e_{i_r}) \geq c(e_{i_1})$. Darum sind alle $c(e_{i_j})$ gleich. Die Kante e_{i_2} muß vor e_{i_1} betrachtet worden sein, e_{i_3} vor e_{i_2}, .. e_{i_1} vor e_{i_r}, also auch vor e_{i_2}. Widerspruch. Zur Komplexität: Eine Stufe besteht aus der Berechnung von Kürzester und C. Das letztere wird durch den Algorithmus 6.4 in einer Zeit $O(|T|) = O(|E|)$ erledigt. Dabei stellt Algorithmus 6.4 ein Array Komponente auf, so daß für $v \in V$ Komponente$[v]$ die Zusammenhangskomponente von v angibt. Mit Hilfe von Komponente läßt sich Kürzester in $O(|E|)$ aufstellen, indem die Kanten von G nacheinander abgearbeitet werden. Somit ist der Aufwand für eine Stufe $O(|E|)$. Auf jeder Stufe wird die Anzahl der vorhandenen Komponenten mindestens halbiert: Jede Komponente enthält mindestens zwei Komponenten der vorherigen Stufe, da jede Komponente mit einer anderen durch die kürzeste Kante verbunden wird. Da auf der ersten Stufe $k = |V|$ gilt, bricht der Algorithmus nach höchstens $\log |V|$ Stufen ab mit der Bestimmung von MST.
□

Definition 6.9 *Gegeben sei ein Graph $G = (V, E)$ und eine Gewichtsfunktion $w_{ij} \geq 0$ für jedes $[v_i, v_j] \in E$. Gesucht wird ein Wald F mit möglichst großem Gesamtgewicht $w(F) = \sum_{e \in F} w(e)$ (maximum weighted forest, MWF).*

MWF ist eng mit MST verwandt. Sei G zunächst zusammenhängend. Wegen $w_{ij} \geq 0$ ist ein optimaler Wald dann ein spannender Baum von G. Sei jetzt für $W = \max\{w_{ij}\}$ $\quad c_{ij} = W - w_{ij}$ für alle $[v_i, v_j] \in E$. Damit ist für jeden spannenden Baum T $c(T)$ mit $w(T)$ durch $w(T) = (|V| - 1)W - c(T)$ verknüpft und T ist maximal bezüglich w genau dann, wenn T minimal ist bezüglich c (d. h. ein MST ist).
Für unzusammenhängendes G erfolgt Rückführung auf die Zusammenhangskomponenten:

Satz 6.9 *Der Wald maximalen Gesamtgewichts von $G = (V, E)$ unter w ist die Vereinigung der MST der Zusammenhangskomponenten von G unter dem oben definierten c.*

Beweis: Jeder Wald kann in Teilgraphen zerlegt werden, die in den einzelnen Komponenten von G liegen. Der Wald ist genau dann maximal, wenn alle Teilgraphen maximal sind.
□

Das Gegenstück zum Satz 6.5 für das MWF–Problem ist:

Satz 6.10 *Sei $\{(U_1, T_1), \ldots, (U_k, T_k)\}$ ein Wald, der V aufspannt und sei $[v, u]$ diejenige aus U_1 herausragende Kante mit maximalem Gewicht. Dann ist unter allen Wäldern, die $T = \bigcup_{i=1}^{k} T_k$ enthalten, ein optimaler, der $[v, u]$ enthält.*

Beweis: In jeder Komponente S_j von G soll der MST bzgl. c bestimmt werden, unter der Bedingung, daß die in S_j enthaltenen T_i enthalten sind. Sei S_r die Komponente, die die Kante $[u, v]$ enthält. Dann folgt die Behauptung mit Satz 6.5 für S_r.
□

Damit gilt der folgende natürliche Algorithmus für MWF:

Algorithmus 6.6 (Der gierige Algorithmus)
Eingabe: $G = (V, E)$ mit einer Gewichtsfunktion w_{ij};
Ausgabe: Der Wald F maximalen Gesamtgewichts von G;
begin
  $F := \emptyset$;
  $C := \{\{v_1\}, \ldots, \{v_n\}\}$;
  while $E \neq \emptyset$ **do**
  begin
    $[u, v]$ Kante maximalen Gewichts in E;
    $E := E - \{[u, v]\}$;
    if u und v sind nicht in derselben Komponente von (V, F) **then**
    begin
      $F := F + \{[u, v]\}$;
      fasse die Komponenten von u und v zusammen;
    end;
  end;
end.

Der gierige Algorithmus schnappt also stets nach dem größten erreichbaren Brocken. Trotzdem löst er das Problem:

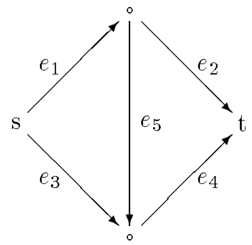
Satz 6.11 *Der gierige Algorithmus löst das MWF–Problem.*

Beweis: Da nur Kanten $[u, v]$ zugelassen werden, für die u und v in verschiedenen Komponenten liegen, entstehen keine Kreise und das Ergebnis ist ein Wald (der spannend ist von Anfang an). Satz 6.10 gewährleistet, daß alle Hinzunahmen von Kanten erlaubt sind (d. h. eine Ergänzung zu einem MWF ist möglich). □

6.4 Maximale Flüsse und der Algorithmus von Ford–Fulkerson

Definition 6.10 Zu einem Netzwerk $N = (s, t, V, E, b)$ mit $n = |V|$ Knoten und $m = |E|$ Kanten, ist ein Beispiel des *Problems des größten Flusses (max flow problem, MFP)* die Aufgabe den Fluß $f \in \mathbb{R}^m$ mit maximalem Wert zu finden.

Beispiel 6.9 Ein Netzwerk mit $n = 4$ Knoten und $m = 5$ Kanten. Der maximale Fluß ist 2.



$b(e_i) = 1, \; i = 1, \ldots, 5.$

Formulierung des Maximalflußproblems als LP: Gegeben sei das Netzwerk $N = (s, t, V, E, b)$. Sei $n = |V|$ und $m = |E|$. A sei die Inzidenzmatrix des Graphen und $f(x, y)$ der Fluß in der Kante (x, y). Dann ist ein s–t Fluß f mit Wert v durch die Restriktionen

$$Af + dv = 0, f \leq b, f \geq 0$$

gegeben, wobei $d \in \mathbb{R}^n$, $\quad d_i = \begin{cases} -1, & i = s \\ +1, & i = t \\ 0, & \text{sonst} \end{cases}$. v ist zu maximieren.

Definition 6.11 Ein *s–t–Schnitt* ist eine Partition (S, T) von V (d.h. $S, T \subset V, S \cup T = V$, und $S \cap T = \emptyset$) in S, T, so daß $s \in S, t \in T$ gilt. Die *Kapazität* eines s–t–Schnittes ist

$$C(S, T) = \sum_{\substack{(i,j) \in E \\ i \in S, j \in T}} b(i, j)$$

Lemma 6.1 *Sei $N = (s, t, V, E, b)$ ein Netzwerk, (S, T) ein Schnitt und f ein Fluß. Dann gilt*

$$|f| = \sum_{\substack{(i,j) \in E \\ i \in S, j \in T}} f(i, j) - \sum_{\substack{(i,j) \in E \\ i \in T, j \in S}} f(i, j)$$

und daher insbesondere $|f| \leq C(S, T)$.

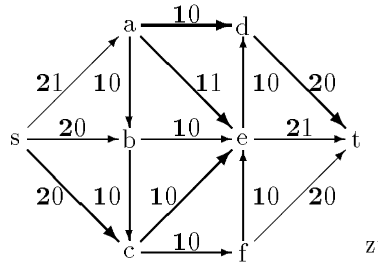
Beweis: Addition der Flußerhaltungsbedingung (2) für alle $v \in S \backslash \{s\}$ ergibt:

$$\begin{aligned} |f| &= \sum_{(s,j) \in E} f(s, j) = \sum_{v \in S} \left(\sum_{(v,j) \in E} f(v, j) - \sum_{(j,v) \in E} f(j, v) \right) \\ &= \sum_{\substack{(i,j) \in E \\ i \in S, j \in S}} f(i, j) + \sum_{\substack{(i,j) \in E \\ i \in S, j \in T}} f(i, j) - \sum_{\substack{(i,j) \in E \\ i \in S, j \in S}} f(i, j) - \sum_{\substack{(i,j) \in E \\ i \in T, j \in S}} f(i, j). \end{aligned}$$

Die 1. und 3. Summe ergeben zusammen 0. Für die Abschätzung beachte man, daß $f(i, j) \leq b(i, j)$, für Kanten (i, j) mit $i \in S, j \in T$ und $f(i, j) \geq 0$, für Kanten (i, j) mit $i \in T, j \in S$. $\square$

Definition 6.12 Zu einem Netzwerk $N = (s, t, V, E, b)$ und einem zulässigen s–t–Fluß f ist ein *zunehmender Weg* ω ein s–t–Weg in folgendem gerichteten Graphen: Sei $G' = (V, E')$ mit $(i, j) \in E'$ genau dann, wenn $(i, j) \in E$ und $f(i, j) < b(i, j)$ (Vorwärtskante) oder $(j, i) \in E$ und $f(j, i) > 0$ (Rückwärtskante).

Beispiel 6.10



zunehmender Weg (s, c, e, a, d, t)

Satz 6.12 *Ein Fluß f auf einem Netzwerk $N = (s,t,V,E,b)$ ist genau dann maximal, wenn es keinen bzgl. f zunehmenden Weg gibt.*

Beweis: Zunächst sei f ein maximaler Fluß. Angenommen, es gäbe einen zunehmenden Weg ω zu f. Dann setze

$$d := \min\{\ \min\{b(e) - f(e) \mid e \text{ Vorwärtskante}\},\ \min\{b(e) \mid e \text{ Rückwärtskante}\}\ \} > 0$$

und

$$f'(e) := \begin{cases} f(e) + d, & \text{für Vorwärtskanten } e \\ f(e) - d, & \text{für Rückwärtskanten } e \\ f(e), & \text{sonst} \end{cases}$$

f' ist ein Fluß mit dem Wert $|f| + d > |f|$. Dies ist ein Widerspruch zur Maximalität von f. Umgekehrt setzen wir voraus, daß es in N keinen bzgl. f zunehmenden Weg gibt. Es sei S die Menge, für die ein zunehmender Weg von s nach v existiert (einschließlich s) und $T = V \backslash S$. Nach Voraussetzung ist dann (S,T) ein Schnitt von N. Jede Kante (i,j) mit $i \in S$ und $j \in T$ muß 'gesättigt' sein, d.h. $f(i,j) = b(i,j)$ und jede Kante (i,j) mit $i \in T$ und $j \in S$ muß 'leer' sein, d.h. $f(i,j) = 0$. Nach Lemma 6.1 gilt dann aber $|f| = C(S,T)$, weswegen f maximal ist.

□

Beispiel 6.10 (Fortsetzung) $d = 1$.

Korollar 6.1 *Ein Fluß f auf einem Netzwerk N ist genau dann maximal, wenn $|f| = C(S,T)$ gilt, wobei S die Menge aller von s aus auf einem zunehmenden Weg erreichbaren Punkte sei.*

Satz 6.13 *Ist N ein Netzwerk mit nur ganzzahligen Kapazitäten, dann gibt es einen ganzzahligen maximalen Fluß auf N.*

Beweis: Durch $f_0(e) = 0$, für alle Kanten e wird ein ganzzahliger Fluß f_0 vom Wert 0 auf N definiert. Ist dieser nicht maximal, so gibt es einen zunehmenden Weg bzgl. f_0. Die im Beweis von Satz 6.12 auftretende Zahl d ist hier eine positive ganze Zahl. Somit können wir – wie im Beweis von Satz 6.12 beschrieben – einen ganzzahligen Fluß f_1 vom Wert d konstruieren. Dieses Verfahren wird entsprechend fortgesetzt. Die Folge $\{|f_i|\}$ der Flußwerte ist ganzzahlig, monoton wachsend und wegen Lemma 6.1 nach oben beschränkt. Darum erhalten wir nach endlich vielen Schritten einen Fluß f, für den es keinen zunehmenden Weg gibt. Dieser ist ganzzahlig und nach Satz 6.12 maximal. □

Satz 6.14 (Maximalfluß–Minimalschnitt) *Der maximale Wert eines Flusses auf einem Netzwerk N ist gleich der minimalen Kapazität eines Schnittes in N.*

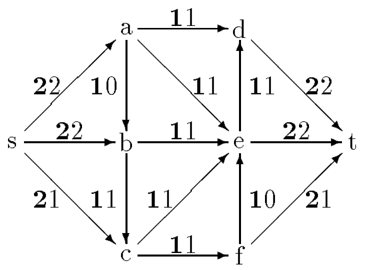
Beweis: Nach Lemma 6.1 gilt $\max|f| = \min C(S,T)$. Es reicht zu zeigen, daß ein maximaler Fluß existiert, denn dann gibt es nach Korollar 6.1 einen Schnitt (S,T) mit $|f| = C(S,T)$.
(1) Sind alle Kapazitäten ganzzahlig, so existiert nach Satz 6.13 ein maximaler Fluß f.
(2) Der Fall rationaler Kapazitäten wird durch Multiplikation mit dem Hauptnenner aller auftretenden Zahlen auf den ganzzahligen Fall reduziert.

(3) Für den irrationalen Fall nehme man eine Folge von Netzwerken $\{N_n\}$ mit rationalen Kapazitäten, die gegen die vorgegebenen irrationalen von unten monoton konvergieren. Für die N_n gibt es maximale Flüsse f_n und minimale Schnitte (S_n, T_n) mit $|f_n| = C(S_n, T_n)$. Die Folge $\{f_n\}$ konvergiert gegen einen Fluß f in N. Da es nur endlich viele Schnitte gibt, sei o. E. $(S_n, T_n) = (S, T)$ konstant. Es folgt

$$|f| = \lim |f_n| = \lim C_{N_n}(S_n, T_n) = \lim C_{N_n}(S, T) = C_N(S, T).$$

$\square$

Beispiel 6.10 (Forts.)



Ein Minimalschnitt besteht aus $\{s, b, c\}$. Der Wert des Schnitts und des Maximalflusses ist 5.

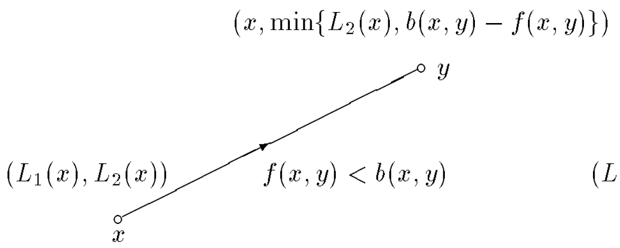
Bemerkungen:

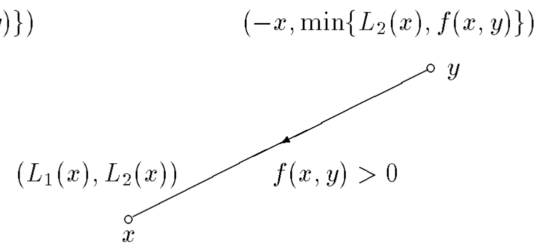
(1) In der Praxis spielen irrationale Kapazitäten in Netzwerken keine Rolle, da im Computer nur rationale Zahlen darstellbar sind.

(2) Die Sätze 6.13 und 6.14 liefern einen Algorithmus, der einen Fluß mit maximalem Wert findet. Für ganzzahliges b ist dieser Fluß (und natürlich sein Wert) sogar ganzzahlig.

(3) Die Sätze machen keine Eindeutigkeitsaussage, d.h. es kann mehrere Flüsse mit maximalem Wert geben, und für ganzzahliges b ist mindestens einer von ihnen ganzzahlig.

Es bleibt noch ein systematisches Verfahren anzugeben, um einen zunehmenden Weg zu finden, falls er existiert. Dies erreichen wir durch Markieren von Knoten beginnend bei s bis wir t erreichen oder steckenbleiben. Jeder Knoten x erhält einen zweiteiligen Label $(L_1(x), L_2(x))$. L_1 gibt an, *von wo* x markiert wurde und $L_2(x)$ gibt den Betrag von zusätzlichem Fluß an, der von s nach x gebracht werden kann. Der Prozeß des Markierens aus einem Knoten x heraus wird "x Abtasten" genannt und wir führen eine Liste, genannt *LISTE*, von markierten aber nicht abgetasteten Knoten.





Es gibt zwei Fälle. Ist Knoten y unmarkiert und folgt x nach, so wird y markiert, falls $f(x, y) < b(x, y)$, und wir setzen

$$\begin{aligned} L_1(y) &:= x \\ L_2(y) &:= \min\{L_2(x), b(x, y) - f(x, y)\} \end{aligned}$$

Das hält fest, daß y von x aus markiert wurde und daß der Fluß um das Minimum von $L_2(x)$ und $b(x, y) - f(x, y)$ vergrößert werden kann. Im Falle, daß y unmarkiert ist und vor x kommt, wird y von x markiert, falls $f(y, x) > 0$, und wir setzen

$$\begin{aligned} L_1(y) &:= -x \\ L_2(y) &:= \min\{L_2(x), f(x, y)\} \end{aligned}$$

Man beachte, daß das Minuszeichen in L_1 besagt, daß die Markierung über eine Rückwärtskante erfolgte.
Das Verfahren beginnt mit dem Abtasten von s und dem Hinzufügen in die *LISTE* von allen Knoten, die von s markiert wurden. Dieses Verfahren wird wiederholt – ein Knoten x wird aus der Liste gewählt und abgetastet und alle Knoten, die von x markiert wurden, werden zur *LISTE* hinzugefügt. Das Verfahren bricht in zwei Fällen ab: entweder wird t markiert oder die *LISTE* wird geleert. Im ersten Fall kann man mittels L_1 von t aus rückwärts einen zunehmenden Weg konstruieren und anschließend den Fluß entlang dieses Weges vergrößern. Im zweiten Fall gibt es keinen zunehmenden Weg und wir haben einen optimalen Fluß erreicht.

Algorithmus 6.7 (Ford–Fulkerson–Algorithmus)
Eingabe: Ein Netzwerk $N = (s, t, V, E, b)$
Ausgabe: Ein maximaler Fluß f in N

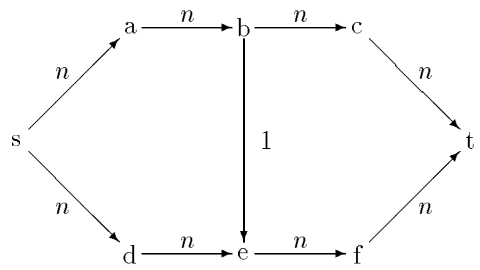
```
begin
  f := 0; (* Fluß initialisieren *)
  repeat
    Setze alle Markierungen auf 0, Setze LISTE := {s},
    (* Suche für zunehmenden Weg initialisieren *)
    while LISTE ≠ ∅ and t nicht markiert do
    begin
      wähle einen Knoten x aus LISTE;
      lösche x aus LISTE;
      abtasten (x);
      if t ist markiert then
      begin
        rekonstruiere den zunehmenden Weg;
        vergrößere den Fluß entlang diesen Weges;
      end;
    end;
  until t ist nicht markiert
end;
```

procedure abtasten
begin
Markiere alle unmarkierten Knoten y, die mit x
durch eine ungesättigte Kante (x, y) verbunden sind, mit $+x$
und nehme sie in *LISTE* auf.

Markiere alle unmarkierten Knoten y, die mit x
durch eine Kante (y, x) mit positivem Fluß verbunden sind, mit $-x$
und nehme sie in *LISTE* auf.
end;

Satz 6.15 *Bricht der Ford–Fulkerson–Algorithmus ab, dann ist der zugehörige Fluß optimal.*

Der Algorithmus von Ford–Fulkerson hat in seiner ursprünglichen Version Nachteile. Im Falle irrationaler Kapazitäten kann er bei ungeschickter Auswahl der Knoten aus *LISTE* versagen. Es gibt ein Beispiel (ohne Beweis), so daß das Verfahren nicht nur nicht abbricht, sondern gegen einen Wert konvergiert, der nur 1/4 des maximal möglichen Flußwertes beträgt. Ferner ist er selbst bei ganzzahligen Kapazitäten nicht polynomial, wie das folgende Beispiel zeigt.
Betrachte das Netzwerk



Wählt man (s, a, b, e, f, t) und (s, d, e, b, c, t) abwechselnd als zunehmende Wege (was bei geeigneter Reihenfolge bei der Auswahl aus *LISTE* möglich ist), so wird der Flußwert in jedem Schritt nur um 1 erhöht und wir benötigen insgesamt $2n$ Iterationen. Andererseits ist der Input $O(\log n)$.
Wir werden nun eine Möglichkeit angeben, wie man zunehmende Wege effektiv auswählen kann. Hält man nämlich *LISTE* als Warteschlange, d.h. es wird der Knoten entnommen, der als erster aufgenommen wurde (FIFO), so wird immer ein Vergrößerungsweg von minimaler Länge ausgewählt und der Algorithmus ist auch für irrationale Kapazitäten polynomial:

Satz 6.16 (Edmonds, Karp) *Entnimmt man die Knoten aus* LISTE *nach dem Prinzip FIFO, so benötigt der Algorithmus 6.7 einen Aufwand von $O(|V||E|^2)$.*

Beweis: Die durch die Markierung bestimmten zunehmenden Wege sind kürzeste Wege, d. h. für $s = v_0, v_1, \ldots, v_r = v)$ gilt (in G') $d(s, v_j) = j$ (siehe Übung).
Wir bezeichnen nun die Folge der im Laufe des Algorithmus konstruierten Flüsse mit f_0 (= der Null–Fluß aus Schritt (1)), $f_1, f_2, \ldots, f_k \ldots$ und mit $d_k(v)$ die kürzeste Länge eines

zunehmenden Weges von s nach v bzgl. f_k, d.h. der Abstand in $G_k = G(f_k)$ von s zu v. Wir behaupten zunächst die Gültigkeit der Abschätzung

$$(*) \qquad d_{k+1}(v) \geq d_k(v), \text{ für alle } k \text{ und } v.$$

Angenommen, $(*)$ werde für ein Paar (v, k) verletzt; o.E. sei dabei v so gewählt, daß $d_{k+1}(v)$ unter allen Werten $d_{k+1}(w)$, die $(*)$ verletzen, minimal ist. Wir betrachten die letzte Kante e in einem zunehmenden Weg von s nach v bzgl. f_{k+1}.
Es sei e eine Vorwärtskante (Rückwärtskante), etwa $e = (u, v)$ ($e = (v, u)$); dann gilt $f_{k+1}(e) < b(e)$ ($f_{k+1}(e) > 0$). Nun ist $d_{k+1}(v) = d_{k+1}(u) + 1$; nach Wahl von v gilt daher $d_{k+1}(u) \geq d_k(u)$, also $d_{k+1}(v) \geq d_k(u) + 1$. Es gilt $f_k(e) = b(e)$ ($f_k(e) = 0$), da sonst $d_k(v) \leq d_k(u) + 1$ und dann doch $d_{k+1}(v) \geq d_k(v)$. Also muß e bei der Abänderung von f_k zu f_{k+1} als Rückwärtskante (Vorwärtskante) verwendet worden sein. Da der zunehmende Weg bzgl. f_k kürzeste Länge hatte, folgt $d_k(u) = d_k(v) + 1$ und daher $d_{k+1}(v) \geq d_k(v) + 2$. Widerspruch.
Bezeichnet $g_k(v)$ die Länge des kürzesten zunehmenden Weges von v nach t bzgl. f_k, so gilt, wenn wir in $(*)$ s durch v und v durch t ersetzen,

$$(**) \qquad g_{k+1}(v) \geq g_k(v), \text{ für alle } k \text{ und } v.$$

Bei jeder Abänderung des Flusses ist mindestens eine Kante des zunehmenden Weges kritisch, d. h. der Fluß durch diese Kante wird entweder bis zur Kapazität erhöht oder auf 0 gesenkt. Es sei $e = (u, v)$ eine kritische Kante im zunehmenden Weg bzgl. f_k. Dieser Weg besteht aus $d_k(v) + g_k(v) = d_k(u) + g_k(u)$ Kanten. Wenn e das nächste Mal in einem zunehmenden Weg (etwa bzgl. f_h) verwendet wird, muß e in umgekehrter Richtung verwendet werden: Wenn e in f_k eine Vorwärtskante war, ist es für f_h eine Rückwärtskante, und umgekehrt. Angenommen, e war für f_k eine Vorwärtskante. Dann folgen $d_k(v) = d_k(u) + 1$ und $d_h(u) = d_h(v) + 1$. Nach $(*)$ und $(**)$ gelten $d_h(v) \geq d_k(v)$ und $g_h(u) \geq g_k(u)$ und somit

$$d_h(u) + g_h(u) = d_h(v) + 1 + g_h(u) \geq d_k(v) + 1 + g_k(u) = d_k(u) + g_k(u) + 2.$$

Der zunehmende Weg bzgl. f_h ist also um mindestens zwei Kanten länger als der bzgl. f_k. Diese Aussage gilt ganz analog auch für den Fall, daß e bzgl. f_k eine Rückwärtskante war (Vertauschen der Rollen von u und v). Trivialerweise kann kein zunehmender Weg mehr als $|V| - 1$ Kanten enthalten. Jede Kante kann also in höchstens $(|V| - 1)/2$ Flußabänderungen kritisch sein, weswegen der Fluß höchstens $O(|V||E|)$ mal geändert werden kann. (Das Verfahren bricht also ab, selbst wenn die Kapazitäten irrational sein sollten.) Das Auffinden eines zunehmenden Weges und die entsprechende Änderung des Flusses erfordern aber jeweils nur $O(|E|)$ Schritte, da dafür jede Kante höchstens zweimal untersucht wird (einmal in der Markierungsphase und einmal beim Abändern). Insgesamt ergibt sich damit ein Aufwand von $O(|V||E|^2)$. $\square$

6.5 Kostenminimale Flüsse

Definition 6.13 Sei $N = (s,t,V,E,b)$ ein Netzwerk, $c_{ij} \in R^+$ für $(i,j) \in E$ eine Wichtung der Kanten und f ein Fluß. Die *Kosten* des Flusses sind $c(f) = \sum_{e\in E} c(e)f(e)$.

Definition 6.14 Sei $N = (s,t,V,E,b)$ ein Netzwerk, $c_{ij} \in R^+$ für $(i,j) \in E$ eine Wichtung der Kanten und $v_0 \in \mathbb{R}^+$. Das Problem *kostenminimaler Flüsse* ist die Bestimmung des Flusses f mit Wert v_0 und minimalen Kosten.

In der Sprache von LP ist dies

$$\min c^\top f, \quad Af + v_0 d = 0, \quad f \le b,\ f \ge 0,$$

wobei A die Inzidenzmatrix ist und $d_s = -1,\ d_t = 1$ und $d_i = 0$, für $i \ne s,t$.

Definition 6.15 Ein zulässiger Fluß mit $Af = 0$ heißt *Zirkulation*.

Zur Lösung des Problems wird folgende Hilfskonstruktion benötigt:

Definition 6.16 Zu einem Fluß f in einem gewichteten Netzwerk $N = (s,t,V,E,b)$ mit Kostenfunktion c wird das gewichtete *Vergrößerungsnetzwerk* $N'(f) = (s,t,V,E',b')$ mit Kostenfunktion c' definiert durch: $(x,y) \in E' \iff$

1. $(x,y) \in E$ und $f(x,y) < b(x,y)$ mit $b'(x,y) = b(x,y) - f(x,y)$ und $c'(x,y) = c(x,y)$
2. $(y,x) \in E$ und $f(y,x) > 0$ mit $b'(x,y) = f(y,x)$ und $c'(x,y) = -c(y,x)$.

Man beachte, daß in $N'(f)$ auch Kanten nach s und aus t heraus möglich sind.

Satz 6.17 *Ein s–t–Fluß f in einem Netzwerk N ist genau dann kostenoptimal, wenn in $N'(f)$ keine Kreise mit negativen Kosten existieren.*

Beweis: Es sei K ein Kreis in $N'(f)$ mit negativen Kosten. Setze

$$\delta := \min\{\, b'(e) \;:\; e \in K\} > 0,$$

dann ist f', definiert durch

$$f'(e) := \begin{cases} f(e) + \delta, & e \text{ Vorwärtskante in } K \\ f(e) - \delta, & e \text{ Rückwärtskante in } K \\ f(e), & \text{sonst} \end{cases}$$

ein Fluß mit Wert $|f'| = |f|$: Ist s nicht in K enthalten, so ist dies klar. Ansonsten ist die Kante in K, die nach s führt, eine Rückwärtskante und die Kante, die herausführt, eine Vorwärtskante. Ferner ändern sich die Kosten um

$$\sum_{e\in K \text{ Vorw.k.}} \delta c(e) - \sum_{e\in K \text{ Rückw.k.}} \delta c(e) = \delta \sum_{e\in K} c'(e) < 0$$

und darum ist f nicht optimal.
Sei nun f Fluß in N mit Wert v_0 und f nicht optimal. Dann gibt es einen Fluß g in N mit Wert v_0 und geringeren Kosten.
Definiere in $N'(f)$ nun h wie folgt:

(1) $0 \leq f(u,v) < g(u,v) \leq b(u,v) \;\mapsto\; h(u,v) := g(u,v) - f(u,v)$,

(2) $0 \leq g(u,v) < f(u,v) \leq b(u,v) \;\mapsto\; h(v,u) := f(u,v) - g(u,v)$,

(3) $h(e) := 0$, sonst.

h ist eine Zirkulation:
(1) Nach Definition gilt $0 \leq h \leq b'$.
(2) h erfüllt in N' die Flußbedingung an allen Knoten (auch in s und t): Für einen Knoten seien E_1, E_2, E_3 die Menge der Kanten, die hineinführen mit $f < g$, $f > g$, bzw. $f = g$ und entsprechend R_1, R_2, R_3 die Menge der Kanten mit $f < g$, $f > g$ bzw. $f = g$, die hinausführen. Dann fließt bzgl. h in den Knoten hinein

$$H^+ = \sum_{e \in E_1} (g(e) - f(e)) + \sum_{e \in R_2} (f(e) - g(e))$$

und hinaus fließt

$$H^- = \sum_{e \in E_2} (f(e) - g(e)) + \sum_{e \in R_1} (g(e) - f(e))$$

Aus der Flußerhaltung von f und g bzgl. N in diesem Knoten folgt

$$\sum_{e \in E_1 \cup E_2 \cup E_3} f(e) = \sum_{e \in R_1 \cup R_2 \cup R_3} f(e), \qquad \sum_{e \in E_1 \cup E_2 \cup E_3} g(e) = \sum_{e \in R_1 \cup R_2 \cup R_3} g(e).$$

Hieraus folgt $\sum_{e \in E_1 \cup E_2} (f(e) - g(e)) = \sum_{e \in R_1 \cup R_2} (f(e) - g(e))$ und damit $H^+ = H^-$.
Die Kosten der Zirkulation sind

$$\begin{aligned}
\sum c'(e) h(e) &= \sum_{f<g} c(e)(g(e) - f(e)) + \sum_{f>g} (-c(e))(f(e) - g(e)) \\
&= \sum_{f(e) \neq g(e)} c(e)(g(e) - f(e)) = c(g) - c(f) < 0.
\end{aligned}$$

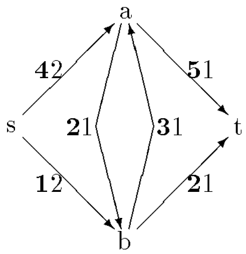
Also ist h eine zulässige Zirkulation in $N'(f)$ mit negativen Kosten. Darum gibt es auch einen Kreis in $N'(f)$ mit negativen Kosten (siehe Übung). □

Kreise negativer Kosten lassen sich mit dem Floyd–Warshall–Algorithmus bestimmen, so daß sich folgender Algorithmus für kostenminimale Flüsse ergibt:

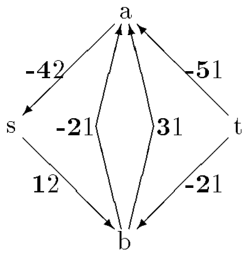
Algorithmus 6.8

begin
  Finde einen Fluß f mit Wert v_0 mit dem Maxflußalgorithmus
  while Es gibt einen Kreis C negativer Kosten in (V, E') **do**
  begin
    Vergrößere f längs C
  end;
end.

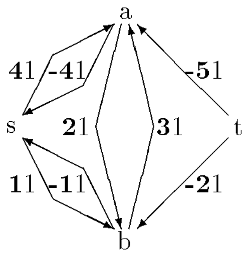
Beispiel 6.11 Gesucht ist ein kostenminimaler s–t–Fluß mit Wert 2 in folgendem Netzwerk, in dem die Kosten fett gedruckt sind.



Ein zulässiger Fluß f ist gegeben durch $f(s,a) = 2$, $f(a,b) = f(a,t) = f(b,t) = 1$, $f(b,a) = f(s,b) = 0$.
Das zugehörige Vergrößerungsnetzwerk N' ist



Dieses Netzwerk enthält den Kreis (s,b,a,s) mit Kosten -5 und einer zugehörigen Zirkulation $\bar{f}$ vom Wert 1. Addition dieser Zirkulation ergibt einen Fluß $f(s,a) = f(a,t) = f(s,b) = f(b,t) = 1, f((a,b) = f(b,a) = 0$. Das zugehörige Vergrößerungsnetzwerk N' ist jetzt



Dieses Netzwerk hat keine Kreise negativer Kosten, so daß die optimale Lösung gefunden ist.

6.6 Branch and Bound Verfahren für TSP

Branch and Bound Verfahren für ein Problem A mit zulässiger Menge G benötigen nur zwei allgemeine Voraussetzungen.

1. Verzweigen (Branch): Für jedes $F \subset G$ ist $F = \bigcup F_i$ darstellbar mit disjunkten F_i. Betrachtet man diese F_i als Knoten eines Graphen, erhält man so einen (gerichteten) Baum.

2. Beschränken nach unten (Bound): Es gibt ein einfaches Problem B, so daß $\min B(F) \leq \min A(F)$ für jedes $F \subset G$ gilt. (B liefert untere Schranken für die Knoten des Baums).

Bezeichnet man wie üblich die Nachfolger eines Knotens in einem gerichteten Baum als Kinder, dann hat der Algorithmus die folgende allgemeine Form: Aktivmenge ist die Menge der Knoten, die noch genauer untersucht werden muß, O ist eine obere Schranke für die Lösung, Bestlösung ist die beste gefundene Lösung

Algorithmus 6.9 (branch_and_bound)

```
begin
   Aktivmenge := {G}; (* G ist das ursprüngliche Problem *)
   O := ∞;
   while Aktivmenge ≠ ∅ do
   begin
      wähle einen Verzweigungsknoten k ∈ Aktivmenge;
      Aktivmenge := Aktivmenge\{k};
      Erzeuge die Kinder von Knoten k, Kind[i], i = 1,...,n_k;
      Erzeuge die unteren Schranken z_i = min B(i), i = 1,...,n_k;
      for i = 1 to n_k do
      begin
         if z_i < O then
         begin
            if Kind[i] ist vollständige Lösung then
            begin
               if A(i) < O then
               begin
                  O := A(i);
                  Bestlösung := Kind[i];
                end;
             end else Aktivmenge := Aktivmenge ∪ {Kind[i]};
          end else lösche Kind[i];
       end;
    end;
 end.
```

Spezielle Möglichkeiten: Für TSP mit Abstandsmatrix (d_{ij}):

1. Formuliere TSP als ganzzahliges Optimierungsproblem

$$\begin{aligned}\min z = \{ \sum_{i,j=1}^{n} d_{ij} x_{ij} \quad | \quad & \sum_{i=1}^{n} x_{ij} = 1, j = 1, \ldots, n; \\ & \sum_{j=1}^{n} x_{ij} = 1, i = 1, \ldots, n; \\ & 0 \le x_{ij} \le 1, \quad x_{ij} \in \mathbb{Z}; \}\end{aligned}$$

   Ist die Lösung eine Tour, dann ist sie sicher optimal. Es können jedoch auch Lösungen auftreten, die Subtouren der Form $(x_{12}, x_{23}, \ldots, x_{k1})$ mit $x_{12} = x_{23} = \ldots = x_{k1} = 1$ enthalten. In einer Lösung von TSP muß mindestens eine dieser Variablen Null sein, so daß die Lösungsmenge in k Kinder mit $x_{i,i+1} = 0$ beim i–ten Kind aufgeteilt wird. Bound wird durch das jeweilige Optimum des ILP geliefert.

2. Bei Transformationen der Form $d_{ij} \to d_{ij} + \pi_i$ für alle i bzw. $d_{ij} \to d_{ij} + \pi_j$ für alle j ändert sich eine optimale Tour nicht (ihre Länge ändert sich um π_i bzw. π_j). Damit wird

$\min_j d_{ij} = \min_i d_{ij} = 0$ und eine Abschätzung der Länge der Tour erreicht. Bilden die zu den d_{ij} mit $d_{ij} = 0$ gehörigen Kanten eine Tour, ist man fertig, andernfalls werden Kanten entweder aufgenommen oder ausgeschlossen (Branch). Wird eine Kante ausgeschlossen, so wird das entsprechende $d_{ij} = \infty$ gesetzt. Wird eine Kante in die Tour aufgenommen, so werden alle dadurch ausgeschlossenen $d_{ij} = \infty$ gesetzt. Ferner müssen (kleine) Kreise vermieden werden, indem geeignete $d_{ij} = \infty$ gesetzt werden. Anschließend kann wie oben weiter reduziert werden (Bound).

Beispiel 6.12 Gegeben sei das folgende Problem mit 5 Städten:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	257	659	505	488
Bi	257	∞	511	618	262
Dr	659	511	∞	728	502
Fr	505	618	728	∞	759
Ha	488	262	502	759	∞

Zeilenoperationen ergeben:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	0	402	248	231
Bi	0	∞	254	461	5
Dr	157	9	∞	226	0
Fr	0	113	223	∞	254
Ha	226	0	240	497	∞

Schranke: 1783

Spaltenoperationen ergeben:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	0	179	22	231
Bi	0	∞	31	235	5
Dr	157	9	∞	0	0
Fr	0	113	0	∞	254
Ha	226	0	17	271	∞

Schranke: 2232

Fall I: (Ac Bi)   Fall II: $\neg$(Ac Bi)

Fall I: Zeile Ac, Spalte Bi, sowie (Bi Ac) werden auf ∞ gesetzt:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	31	235	5
Dr	157	∞	∞	0	0
Fr	0	∞	0	∞	254
Ha	226	∞	17	271	∞

Zeilenoperationen ergeben:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	26	230	0
Dr	157	∞	∞	0	0
Fr	0	∞	0	∞	254
Ha	209	∞	0	254	∞

Schranke: 2254

Fall Ia: (Bi Ha)   Fall Ib: $\neg$(Bi Ha)

Fall Ia: Zeile Bi, Spalte Ha, sowie (Ha Ac) werden auf ∞ gesetzt:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	∞	∞	∞
Dr	157	∞	∞	0	∞
Fr	0	∞	0	∞	∞
Ha	∞	∞	0	254	∞

Schranke: 2254

Fall Iai: (Ha Dr)   Fall Iaii: $\neg$(Ha Dr)

Fall Iai: Zeile Ha, Spalte Dr, sowie (Dr Ac) werden auf ∞ gesetzt:

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	∞	∞	∞
Dr	∞	∞	∞	0	∞
Fr	0	∞	∞	∞	∞
Ha	∞	∞	∞	∞	∞

Schranke: 2254

Damit ergibt sich als Reise (Ac Bi Ha Dr Fr Ac) mit der Länge 2254.

Fall Iaii: (Ha Dr) = ∞.

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	∞	∞	∞
Dr	157	∞	∞	0	∞
Fr	0	∞	0	∞	∞
Ha	∞	∞	∞	254	∞

Schranke: 2508

Fall Ib: (Bi Ha) = ∞.

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	∞	∞	∞
Bi	∞	∞	26	230	∞
Dr	157	∞	∞	0	0
Fr	0	∞	0	∞	254
Ha	209	∞	0	254	∞

Schranke: 2280

Fall II: (Ac Bi) = ∞.

	Ac	Bi	Dr	Fr	Ha
Ac	∞	∞	179	22	231
Bi	0	∞	31	235	5
Dr	157	9	∞	0	0
Fr	0	113	0	∞	254
Ha	226	0	17	271	∞

Schranke: 2254

Damit ist (Ac Bi Ha Dr Fr Ac) die optimale Route, es sind 4 statt 12 Fälle untersucht worden.

6.7 Heuristiken und lokale Suche

Zur Bestimmung oberer Schranken für TSP dienen Heuristiken, die sich mit lokalen Methoden verbessern lassen.

Beispiel 6.13 Far insert: Sei v eine beliebige Stadt, die Tour $T = \{v\}$. Jetzt sei für $T = (v_{i_1}, \dots, v_{i_k})$ v' die Stadt für die der Abstand zur Tour $\min_{1 \le j \le k}(d_{v', v_{j_i}})$ maximal wird. Dann wird diese Stadt in die Tour eingefügt: $T' = (v_1, \dots, v_l, v', v_{l+1}, \dots, v_k)$, wobei l so gewählt wird, daß die Länge von T' minimal wird.

Algorithmus 6.10 (far_insert)
begin
T := (1,1); (* T Tour *)
L := 0; (* L Länge der Tour *)
for i := 2 **to** n **do**
begin
d[i] := d_{1i}; (* d[i] gibt den Abstand zur Tour an *)
end;
for i := 1 **to** n-1 **do**
begin
wähle j mit d[j] = $\max\{d[l] \mid l = 2, \dots, n\}$;
for l := 1 **to** i **do** c[l] := $d_{T_l,j} + d_{j,T_{l+1}} - d_{T_l,T_{l+1}}$;
(* T_l Stadt an der l–ten Stelle der Tour *)
wähle m mit c[m] = $\min\{c[l] \mid l = 1, \dots, i\}$;
füge j an der Stelle m in die Tour ein;
d[j] := 0;
for $x \in \{2, \dots, n\} \setminus T$ **do** d[x] := $\min(d[x], d_{xm})$;
end;
end.

Zur Lösung bzw. Näherungslösung von Optimierungsproblemen läßt sich oft die Suche auf Umgebungen einsetzen.

Definition 6.17 Für ein Optimierungsproblem mit Beispielen (F, c) ist eine *Umgebung* eine für jedes Beispiel definierte Abbildung

$$N : F \to 2^F.$$

Beispiel 6.14 Für TSP ist eine Umgebung *2–Austausch* definiert durch

$$N_2(f) = \{\, g \quad | \quad g \in F;\ g \text{ entsteht aus } f \text{ durch Entfernen zweier Kanten} \\ \text{und Einfügen zweier Kanten}\}$$

2–Austausch läßt sich offensichtlich auf *k–Austausch* verallgemeinern.

Beispiel 6.15 LP läßt sich als kombinatorisches Optimierungsproblem auf der Menge $\mathcal{B}$ aller Basen ansehen. Einen Umgebungsbegriff für $\mathcal{B}$ ist gegeben durch

$$N(B) = \{B' \in \mathcal{B} \mid \quad |B \setminus B'| = 1\}.$$

Durch Umgebungen sind in natürlicher Weise lokale Minima bestimmt:

Definition 6.18 Für ein Beispiel (F, c) eines Optimierungsproblems und eine Umgebung N heißt $f \in F$ *lokal optimal* (in Bezug auf N), falls

$$c(f) \leq c(g), \quad \text{für alle} \quad g \in N(f).$$

Der Umgebungsbegriff liefert einen trivialen Algorithmus zur Bestimmung lokaler Minima:

Algorithmus 6.11
Sei A ein Optimierungsproblem mit Beispielen (F, c). Weiter sei eine Funktion 'verbessere' auf F gegeben mit

$$\text{verbessere}(t) = \begin{cases} s \in N(t), \quad c(s) < c(t), \text{ falls ein solches } s \text{ exitiert} \\ \text{'nein' sonst} \end{cases}$$

verbessere kann durch einen Algorithmus oder auch durch vollständige Suche auf der Umgebung gegeben sein.

procedure Lokal;
begin
  t := beliebiges Element aus F;
  while verbessere(t) $\neq$ 'nein' **do**
  begin
    t := verbessere(t);
  end;
end.

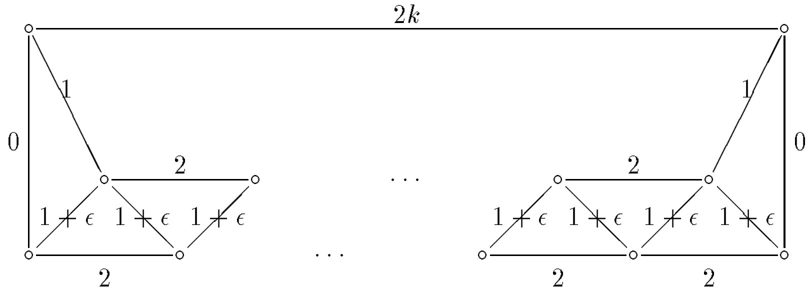
Der Simplexalgorithmus ist ein Beispiel für diesen Algorithmus, wobei 'verbessere' durch den Pivotschritt gegeben ist.
Für TSP sind 2–Austausch und 3–Austausch einfache Algorithmen, die oft nicht zu schlechte Ergebnisse liefern.

Definition 6.19 Gegeben sei ein Optimierungsproblem mit zulässigen Mengen F und Umgebungen N. Die Umgebungen heißen *exakt*, falls aus der Optimalität von $f \in F$ in Bezug auf N stets die globale Optimalität folgt.

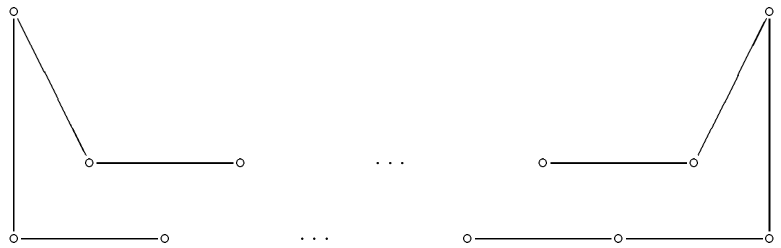
Nach den Ergebnissen über lineare Optimierung ist für (nichtentartete) LP die Umgebung aus Beispiel 6.15 exakt. Ferner ist für die konvexe Optimierung folgender Umgebungsbefriff exakt: Für $x \in F \subset \mathbb{R}^n$ sei $N(x) = B_\epsilon(x) \cap F$. Dagegen gilt

Satz 6.18 *Für das TSP mit n Städten ist $(n-3)$–Austausch keine exakte Umgebung.*

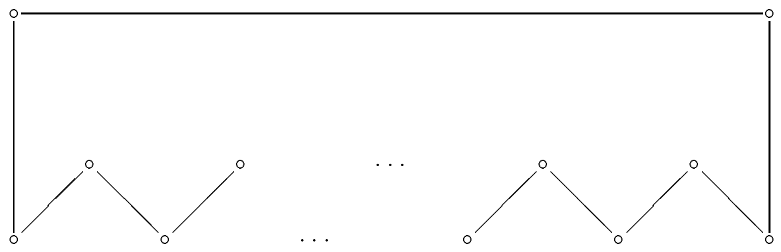
Beweis: Man betrachte den folgenden Graphen mit den angegebenen Gewichten (nicht eingezeichnete Kanten haben Gewicht ∞).



Hat die unterste Reihe $k+1$ Knoten, so ist $n = 2k+3$. Ferner sei $0 < \epsilon < 1/(2k)$. Offensichtlich hat jede Tour Länge ≥ 4 (man betrachte die Projektionen auf die x–Achse). Darum ist die folgende Tour T_0 optimal:



T_1 sei die folgende Tour mit Länge $2k + 2k(1+\epsilon) < 4k+1$.



Nun sei T eine beliebige Tour, die sich von T_1 durch höchstens $n-3 = 2k$ Kanten unterscheidet. Es reicht zu zeigen, daß es eine vertikale Gerade gibt, die T mindestens dreimal schneidet, denn dann folgt $d(T) \geq 4k+1$ (wieder Betrachtung der Projektionen). Die einzige mögliche

Tour ohne 2–Kanten ist T_1 und die einzig mögliche ohne $(1+\epsilon)$–Kanten ist T_0. Also hat T eine 2–Kante, die zu einer $(1+\epsilon)$–Kante benachbart ist. Ist die $2k$–Kante enthalten, so ist die Geradenbedingung erfüllt. Ansonsten muß in den beiden unteren Reihen ein Z–Muster entstehen, so daß auch in diesem Fall die Bedingung erfüllt ist. $\square$

Besonders effektiv scheinen Varianten von Algorithmus 6.11 zu sein, bei denen man

1. auch gelegentliche Verschlechterungen zuläßt
2. die Nachbarpunkte durch ein Zufallsprinzip erzeugt.

Durch die Verschlechterungen ist es möglich, lokale Minima wieder zu verlassen. Weiter lassen sich solche Algorithmen auch verwenden, zulässige Lösungen zu finden, indem man Punkte, die Restriktionen nicht erfüllen, nicht ausschließt, sondern durch eine Straffunktion als schlechte Lösungen charakterisiert.
Einige Beispiele:

1. Versuche zufällige Austauschschritte (bei TSP etwa 2–Austausch) bis eine Zahl von Versuchen in der Größenordnung einer vollständigen Umgebung keine Verbesserung ergibt, dann lasse auch Verschlechterungen zu, wobei ein Auswahlmechanismus dafür sorgt, daß die Verschlechterungen im Allgemeinen nicht zu groß sind.

   Eine Variante davon ist das Simulated Annealing (simuliertes Ausglühen), bei dem am Anfang größere Verschlechterungen akzeptiert werden, am Ende nur noch kleine.

2. Threshold Accepting (Toleranzschwellenalgorithmus): Man wähle eine Toleranzschwelle und lasse beliebige Schritte zu, solange nur die Verschlechterung in einem Schritt kleiner als die Toleranzschwelle ist. Ergibt der Algorithmus eine Zeitlang keine Verbesserungen mehr, so reduziert man die Toleranzschwelle, bis am Ende die Toleranzschwelle 0 zu lokaler Suche führt.

3. Sintflutalgorithmus: Der Algorithmus akzeptiert jeden Austauschschritt, solange eine gewisse Qualität der Lösung (Wasserstand) nicht unterschritten wird, bei jedem akzeptierten Austauschschritt wird der Wasserstand etwas erhöht, bis am Ende ein lokales Optimum erreicht ist.

Ein Problem all dieser Algorithmen besteht darin, die Parameter günstig zu wählen. Anschließend liefern sie oft sehr gute Lösungen.

6.8 Ein approximierender Algorithmus

Definition 6.20 Sei A ein Optimierungsproblem mit ganzer positiver Kostenfunktion c und sei $\mathcal{A}$ ein Algorithmus, der zu jedem Beispiel I von A eine zulässige Lösung $f_{\mathcal{A}}(I)$ berechnet. Weiter sei mit $\hat{f}(I)$ die optimale Lösung von I bezeichnet. Dann heißt $\mathcal{A}$ ϵ–approximierender Algorithmus für A mit $\epsilon \geq 0$, falls

$$\frac{|c(f_{\mathcal{A}}(I)) - c(\hat{f}(I))|}{c(\hat{f}(I))} \leq \epsilon$$

für alle Beispiele I gilt.

Definition 6.21 Das *metrische TSP–Problem* (ΔTSP) ist TSP eingeschränkt auf Matrizen, die der Dreiecksungleichung $d_{ik} < d_{ij} + d_{jk}$ für alle i, j, k genügen.

Definition 6.22 Ein Multigraph (V, E) (ein Graph, bei dem die Wiederholung von Kanten erlaubt ist) heißt *Eulersch*, falls er einen Eulerschen Kreis besitzt. Dabei ist ein *Eulerscher Kreis* ein geschlossener Kantenzug, der alle Ecken mindestens einmal und alle Kanten genau einmal durchläuft.

Satz 6.19 (Euler, ältester Satz der Graphentheorie) *Ein Multigraph $G = (V, E)$ ist genau dann Eulersch, wenn*

(a) G ist zusammenhängend,

(b) Alle Knoten von G haben einen geraden Grad.

Beweis: Offensichtlich sind beide Bedingungen notwendig. Induktiv folgt, daß sie hinreichend sind: Die Behauptung gilt für den Graphen mit einer Ecke und ohne Kanten. $G = (V, E)$ genüge daher (a) und (b), und (V', E') mit $|E'| < |E|$, die (a) und (b) genügen, seien Eulersch. Sei jetzt $v \in V$. Wähle einen Knoten v als Startpunkt und gehe so auf Kanten, daß keine zweimal durchlaufen wird (dies geht nach (b)), bis man wieder nach v zurückkommt. Erhalte so einen geschlossenen Kantenzug w mit Anfangspunkt v, bei dem keine Kante wiederholt wird. Jetzt entfernt man die Kanten von w aus G. Danach entsteht eine Anzahl von Zusammenhangskomponenten G_i. Jedes G_i genügt (a) und (b) und besitzt so einen Eulerschen Kreis. Man erhält jetzt einen Eulerschen Kreis auf G, indem man auf w von einem G_i zum nächsten geht, den Eulerschen Kreis dieser Komponente durchläuft und weiter auf w geht. $\square$

Der Beweis liefert sofort folgenden rekursiven Algorithmus, der in jedem Multigraphen (V, E), der (a) und (b) genügt, einen Eulerschen Kreis in einer Zeit $O(|E|)$ liefert.

Algorithmus 6.12

```
procedure Euler(v; w); (* v Ecke, w geschlossener Weg, G Multigraph *)
begin
   if G hat keine Kanten then
   begin
      w = (v);
    end else
   begin
      w = ();
      ausgehend von v konstruiere man einen geschlossenen Kantenzug
      p = (u_1, ..., u_k) in G, der keine Kante zweimal benutzt;
      (* u_k = u_1 *)
      entferne (u_1, u_2), ..., (u_{k-1}, u_1) aus G;
      for i := 1 to k do
      begin
         Euler(u_i, v);
         w := w + v;
       end;
    end;
 end.
```

Definition 6.23 Sei auf dem vollständigen Graphen mit Ecken $\{1, \ldots, n\}$ (d_{ij}) eine Abstandsmatrix gegeben, die der Dreiecksungleichung genügt. Ein *Eulerscher spannender Graph* ist ein Eulerscher Multigraph $G = (V, E)$ mit $V = \{1, \ldots, n\}$. Die *Kosten* von G sind $c(G) = \sum_{(i,j) \in E} d_{ij}$.

Satz 6.20 *Ist $G = (V, E)$ ein Eulerscher spannender Graph, dann kann man in einer Zeit $O(|E|)$ eine Tour τ von V finden, die $c(\tau) \leq c(G)$ genügt.*

Beweis: Nach Voraussetzung hat G einen Eulerschen Kreis w. Da w alle Ecken von V enthält, kann man $w = (i_1, \alpha_1, i_2, \ldots, i_n, \alpha_n, i_1)$ schreiben, wobei $\tau = (i_1, \ldots, i_n)$ eine Tour ist und $\alpha_1, \ldots, \alpha_n$ (möglicherweise leere) Folgen von Zahlen aus $\{1, \ldots, n\}$ sind. Die Dreiecksungleichung garantiert, daß das Weglassen der α_i die Länge nicht vergrößert. □

Damit erhält man den folgenden Baum–Algorithmus für eine Tour:

Algorithmus 6.13 (baum)

```
begin
   Finde den minimalen spannenden Baum T bezüglich (d_ij);
   Erzeuge einen einen Multigraph G aus T durch
   Verdopplung der Kanten aus T;
   Suche einen Eulerschen Kreis in G;
   Finde eine eingebette Tour;
 end.
```

Satz 6.21 *Der Baum–Algorithmus ist ein 1–approximierender Algorithmus für ΔTSP.*

Beweis: G ist Eulersch, da T zusammenhängend ist, und wegen der Verdopplung der Kanten jede Ecke gerade Valenz hat. Sind c_0 die Kosten der kürzesten Tour, dann reicht es nach Satz 6.20 zu zeigen, daß $c(G) \leq 2 \cdot c_0$ genügt. Für die Kosten $c(T)$ eines minimalen spannenden Baumes gilt $c(G) = 2 \cdot c(T)$. Eine Tour enthält immer einen spannenden Baum, also gilt $c(T) \leq c_0$, woraus die Behauptung folgt. □

Bemerkungen:

1. Es gibt ungünstige Beispiele, so daß die Schranke $\epsilon = 1$ für den oben angegebenen Algorithmus asymptotisch erreicht wird.

2. Der Baum–Algorithmus hat die Komplexität $O(n^2)$.

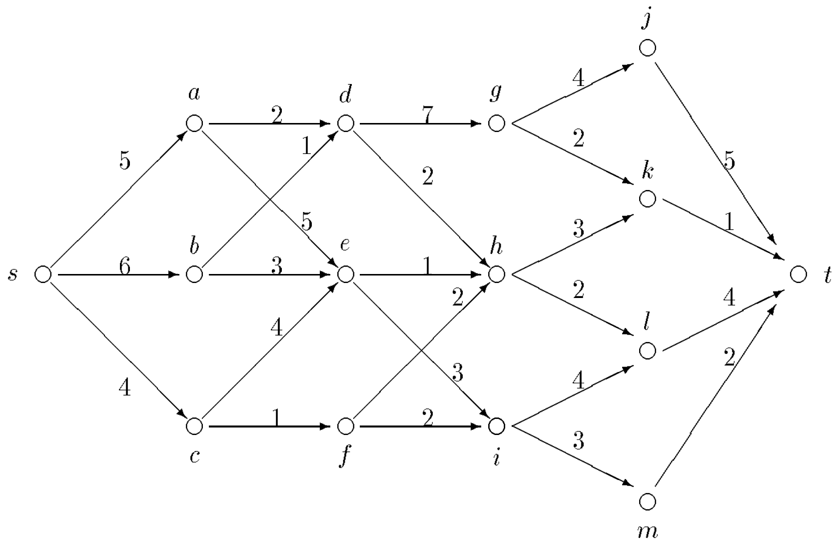
3. Man kann den Baum–Algorithmus so modifizieren, daß ϵ auf 1/2 verbessert wird. Dies geschieht durch geschickte Definition des Eulerschen Graphen aus dem MST, in dem schon einige Knoten geraden Grad haben.

6.9 Dynamische Programmierung

Dynamische Programmierung ist für folgenden Aufgabentyp geeignet: Zur Lösung eines Optimierungsproblems sei eine Folge von n (unbekannten) Entscheidungen $D_1, \ldots, D_n$ notwendig. Weiter sei die Entscheidung D_k in gewisser Weise von den vorher getroffenen Entscheidungen unabhängig (z.B. nur von ihrer Menge, aber nicht ihrer Reihenfolge abhängig, oder nur von einem gewissen Zustand abhängig). Dann braucht man für die Entscheidung D_k nur die Folge von Entscheidungen $D_1, \ldots, D_{k-1}$ zu betrachten, die zu diesem Zustand führt und optimal ist. Die Methode soll an drei Beispielen näher erklärt werden.

(1) Gegeben sei ein *geschichtetes* Netzwerk, d. h. die Eckenmenge kann in disjunkte Mengen $V_0 = \{s\}, V_1, \ldots, V_k, V_{k+1} = \{t\}$ zerlegt werden, so daß nur Kanten zwischen benachbarten V_i, V_{i+1} vorkommen. Gesucht ist ein kürzester Weg von s nach t.
Beginnend bei V_k wird in einer Tabelle festgehalten, wie ein kürzester Weg von jedem $v \in V_i$ aus nach t verläuft. Dazu wird für jedes v der Nachfolger aus V_{i+1} auf einem kürzesten Weg und die zugehörige Länge festgehalten. Aus der Tabelle für V_k kann nun *rückwärts* die Tabelle für V_{k-1} bestimmen usw. bis man bei V_0 angekommen ist. Nun kann man anhand der Tabellen *vorwärts* den kürzesten Weg rekonstruieren.

Beispiel 6.16



$$\text{Tabelle}(4) = \left\{ \begin{array}{lcccc} \text{Knoten} & j & k & l & m \\ \text{Nachfolger} & t & t & t & t \\ \text{Länge} & 5 & 1 & 4 & 2 \end{array} \right. \quad \Rightarrow \quad \text{Tabelle}(3) = \left\{ \begin{array}{lccc} \text{Knoten} & g & h & i \\ \text{Nachfolger} & k & k & m \\ \text{Länge} & 3 & 4 & 5 \end{array} \right.$$

$$\Rightarrow \quad \text{Tabelle}(2) = \left\{ \begin{array}{lccc} \text{Knoten} & d & e & f \\ \text{Nachfolger} & h & h & h \\ \text{Länge} & 6 & 5 & 6 \end{array} \right. \quad \Rightarrow \quad \text{Tabelle}(1) = \left\{ \begin{array}{lccc} \text{Knoten} & a & b & c \\ \text{Nachfolger} & d & d & f \\ \text{Länge} & 8 & 7 & 7 \end{array} \right.$$

$$\Rightarrow \quad \text{Tabelle}(0) = \left\{ \begin{array}{lc} \text{Knoten} & s \\ \text{Nachfolger} & c \\ \text{Länge} & 11 \end{array} \right.$$

Der kürzeste Weg ist dann gegeben durch (s, c, f, h, k, t) mit Länge 11.

(2) Ein Behälter (z. B. Rucksack) mit begrenzter Kapazität soll mit Gegenständen, verschieden an Größe und Wert, möglichst sinnvoll gepackt werden.

Definition 6.24 Gegeben seien zwei Folgen $c_1, \ldots, c_k$ und $d_1, \ldots, d_k$ sowie ein M mit $c_i, d_i, M \in \mathbb{N}$. Das *Ranzenproblem (Knapsackproblem)* besteht darin,

$$\max\{c_{i_1} + \ldots + c_{i_j} \mid \{i_1, \ldots, i_j\} \subset \{1, \ldots, k\}; \quad d_{i_1} + \ldots + d_{i_j} \leq M\}$$

zu bestimmen.

Das Ranzenproblem läßt sich mit folgendem dynamischen Programmierungsverfahren lösen:

Algorithmus 6.14 (knapsack)

```
begin
   (* Opt[i] ist die (vorläufig) beste Möglichkeit für i *)
   (* Beachte: Änderungen von Opt werden erst nach Durchlauf der
       i–Schleife permanent *)
   for i := 1 to M do Opt[i] := 0;
   for i := 1 to k do if d[i] ≤ M then
   begin
      if Opt[d[i]]<c[i] then
      begin
         Opt[d[i]] := c[i];
         Weg[d[i]] := {i};
      end;
      for n := 1 to M do
      begin
         if (Opt[n] > 0) and (n+d[i] ≤ M) then
         begin
            if Opt[n]+c[i] > Opt[n+d[i]] then
            begin
               Opt[n+d[i]] := Opt[n]+c[i];
               Weg[n+d[i]] := Weg[n]∪{i};
            end;
         end;
      end;
   end;
   Max := 0;
   for i := 1 to M do
   begin
      if Opt[i]>Max then
      begin
         Max := Opt[i];
         Bestweg := Weg[i];
      end;
   end;
end.
```

Satz 6.22 *Algorithmus 6.14 löst das Ranzenproblem in einer Zeit* $O(Mk^2)$.

Beweis: Mit Induktion nach der Anzahl der Stücke j wird gezeigt, daß die Opt[i] die beste Belegung für die Größe i und das Problem mit Stücken $1, \ldots, j$ angibt. Für $j = 1$ ist die Aussage klar. Es gelte die Aussage nun für die Opt[i] nach j Schritten. Es seien Opt'[i] die optimalen Werte für das Problem mit den Stücken $1, \ldots, j+1$. Dann gilt

$$\text{Opt'[i]} = \max\{\text{Opt[i]}, \text{Opt}[i - d[j+1]] + c[j+1]\}.$$

Es werden $O(kM)$ Stufen durchlaufen. Der zweite Faktor k in der Aussage über die Laufzeit folgt aus den Speicheroperationen für 'Weg'. □

Bemerkung: Der Algorithmus ist wegen des Faktors M nicht polynomial.

Beispiel 6.17 Sei das folgende Ranzenproblem gegeben:

j	1	2	3	4	5
c_j	6	11	17	3	9
d_j	1	1	3	2	2

$M = 5.$

Es ergeben sich folgende Werte für Opt:

0	0	0	0	0	0
1	$6, \{1\}$	$11, \{2\}$	$11, \{2\}$	$11, \{2\}$	$11, \{2\}$
2	0	$17, \{1, 2\}$	$17, \{1, 2\}$	$17, \{1, 2\}$	$17, \{1, 2\}$
3	0	0	$17, \{3\}$	$17, \{3\}$	$20, \{2, 5\}$
4	0	0	$28, \{2, 3\}$	$28, \{2, 3\}$	$28, \{2, 3\}$
5	0	0	$34, \{1, 2, 3\}$	$34, \{1, 2, 3\}$	$34, \{1, 2, 3\}$

(3) Auch das TSP kann mit Hilfe von dynamischer Optimierung gelöst werden.
Für TSP sei eine optimale Tour gesucht, die in 1 anfängt. Dazu sei zu $S \subset \{2, \ldots, n\}$ und $k \in S$ $C(S, k)$ die optimalen Kosten einer Tour, die in 1 anfängt, alle Städte aus S besucht und in k endet. Für $|S| = 1$ ergibt sich $C(\{k\}, k) = d_{1k}$. $C(S, k)$ erhält man offensichtlich durch

$$C(S, k) = \min_{m \in S \setminus \{k\}} (C(S \setminus \{k\}, m) + d_{mk})$$

Zur Fortsetzung des Verfahrens muß dies für alle S gegebener Mächtigkeit und für jedes $k \in S$ berechnet werden. (Zur Rekonstruktion der Tour muß außerdem das m, für das das Minimum angenommen wird, abgespeichert werden.) Zählt man jeden Wert von $C(S, k)$ als einen Speicherplatz, ergibt sich ein Platzbedarf von

$$\sum_{k=1}^{n-1} k \binom{n-1}{k} = O(n 2^n).$$

Zur Bestimmung jedes $C(S, k)$ werden $O(n)$ Operationen benötigt. Darum ergibt sich ein Zeitaufwand von $O(n^2 2^n)$. Dieser Aufwand wächst exponentiell, ist aber andererseits klein im Vergleich zu $(n-1)!$.

7 Spieltheorie

7.1 Allgemeines und Historie

Gesellschaftsspiele wie Schach, Mühle, Skat, Knobeln, etc. haben gemeinsam, daß mehrere Parteien teilnehmen, die entgegengesetzte, oder zumindest konfligierende Ziele haben. Jeder Spieler trifft eine Entscheidung und das Ergebnis des Spiels ist von der Gesamtheit all dieser Entscheidungen abhängig.
Den gleichen Sachverhalt findet man auch in einigen Wissenschaften und sonstigen Lebensbereichen. Einige Beispiele sind

- Entscheidungen über Preise, Mengen und Produktdifferenzierung;
- Verhandlungssituationen, z. B. über Tarifverträge oder Abrüstung;
- Bietverhalten bei Auktionen;
- strategisches Verhalten in Streiks, Handelskriegen oder militärischen Auseinandersetzungen;
- Wettbewerb und Kooperation verschiedener Regierungen/Zentralbanken in einem gemeinsamen Wirtschaftsraum (Geld–, Steuer–, Außenhandelspolitik).

Aufgrund der vielen Anwendungsmöglichkeiten entwickelte sich in den 40er und 50er Jahren schnell eine Theorie (vor allem durch von Neumann, Morgenstern und Nash), die sowohl in der Mathematik als auch in den Wirtschaftswissenschaften von Interesse ist. In den Jahren 1994 und 1996 wurde der Nobelpreis für Wirtschaftswissenschaften an Spieltheoretiker vergeben.

7.2 Grundlegende Begriffe

Es sollen hier nur *Strategiespiele* betrachtet werden, d. h. solche Spiele, in denen die Teilnehmer zumindest teilweise den Spielverlauf beeinflussen können (der Zufall, in Form eines Würfels, kann als weiterer Spieler angesehen werden).
Es gibt Situationen, in denen es vorteilhaft ist, zumindest teilweise zu kooperieren. Wir beschränken uns hier aber zunächst auf *nichtkooperative* Spiele.
Zwar sind sowohl vom theoretischen als auch vom praktischen Standpunkt auch Spiele mit unendlich vielen Spielern von Interesse. Wir werden hier aber annehmen, daß die Menge $I = \{1, \ldots, n\}$ der Spieler endlich ist.
Jeder Spieler i hat die Möglichkeit, zwischen verschiedenen (mindestens zwei) *Strategien* $s_i \in S_i$ zu wählen. In einem Spiel (oder einer Runde) wird hieraus eine *Situation* $s = (s_1, \ldots, s_n)$ zusammengesetzt. Das kartesische Produkt $S = \prod_{i \in I} S_i$ wird als *Strategienraum* bezeichnet.
Man beachte, daß die Mengen S_i sehr groß sein können. Z. B. im Schach ist eine Strategie die Menge der favorisierten Züge für jede mögliche Spielsituation.
In jeder Situation $s \in S$ erhalten die Spieler eine Auszahlung $H_i(s)$, $i = 1, \ldots, n$. Die H_i werden als *Auszahlungsfunktionen* bezeichnet.

Definition 7.1 Es sei I eine endliche Menge, S_i, $i \in I$ Mengen und $H_i : S \to \mathbb{R}$ Funktionen. Dann heißt das System

$$\Gamma = \langle I, \{S_i\}_{i \in I}, \{H_i\}_{i \in I} \rangle$$

ein *nichtkooperatives Spiel* (in Normalform).

Beispiel 7.1 *Gefangenendilemma*: Zwei Gefangene werden getrennt voneinander verhört. Sie werden beschuldigt, gemeinsam eine Straftat begangen zu haben. Die Indizien reichen nicht für eine Verurteilung aus. Jeder der beiden hat die Möglichkeit, die Tat zu leugnen oder ein Geständnis abzulegen. Gestehen beide, so werden sie zu 8 Jahren Haft verurteilt. Leugnen beide, so können sie wegen geringerer Delikte nur zu einem Jahr verurteilt werden. Gesteht einer und einer leugnet, so wird der Geständige freigesprochen und der andere zu 10 Jahren Haft verurteilt. Die Auszahlungsfunktionen können hier durch zwei Matrizen dargestellt werden:

2\1	L	G
L	(-1,-1)	(0,-10)
G	(0,-10)	(-8,-8)

Definition 7.2 Ein Spiel heißt *Nullsummenspiel*, wenn für alle $s \in S$ gilt $\sum_{i \in I} H_i(s) = 0$.

Beispiel 7.2 (1) *Knobeln*: Es wird eine Münze geworfen. Falls "Wappen" erscheint, gewinnt der erste Spieler, ansonsten der zweite.
(2) *Stein–Schere–Papier*: Zwei Spieler geben gleichzeitig ein Zeichen, das Stein, Schere oder Papier bedeutet. Die Auszahlungsfunktion für Spieler 1 sei durch folgende Matrix gegeben, die für Spieler 2 durch die negative.

2\1	St	Sch	Pa
St	0	-1	+1
Sch	+1	0	-1
Pa	-1	+1	0

In diesen Beispielen werden meist viele Runden gespielt. Dabei ändert man natürlich häufiger seine Startegie, denn sonst kann sich der der Gegner leicht darauf einstellen. Es kommt darum nicht auf die Wahl einer einzelnen Strategie, sondern vielmehr auf die Gewichtung aller Strategien mit Wahrscheinlichkeiten an.
Bei endlichen Spielen bedeutet das, daß einem Spieler mit *reinen Strategien* $S_i = \{1, \dots, m\}$ nun die Möglichkeiten

$$\{x = (x_1, \dots, x_m) \in \mathbb{R}^m \,:\, 0 \le x_i \le 1,\; x_1 + \ldots + x_m = 1\}$$

zur Verfügung stehen, wobei Strategie i mit Wahrscheinlichkeit x_i gewählt wird. Der Vektor x wird als *gemischte Strategie* bezeichnet (im allgemeinen ist eine gemischte Strategie des Spielers i eine Wahrscheinlichkeitsverteilung σ_i auf der Menge S_i der reinen Strategien). Die Menge der gemischten Strategien des Spielers i sei mit Σ_i bezeichnet. Die Wahrscheinlichkeit, daß die Situation $s = (s_1, \dots, s_n)$ eintritt, ist

$$\sigma(s) := \sigma_1(s_1) \cdot \ldots \cdot \sigma_n(s_n).$$

Diese Wahrscheinlichkeitsverteilungen σ auf S werden die Situationen des Spiels in gemischten Strategien genannt. Die Auszahlungsfunktionen ergeben sich nun durch den Erwartungswert der Auszahlungen:

$$H_i(\sigma) = \sum_{s \in S} H_i(s)\sigma(s) = \sum_{s_1 \in S_1} \dots \sum_{s_n \in S_n} H_i(s_1, \dots, s_n) \prod_{i=1}^{n} \sigma_i(s_i). \tag{7.1}$$

(Im allgemeinen Fall ist der Erwarungswert der Auszahlungen ein Integral.)

Definition 7.3 Das Spiel

$$\Gamma^* = \langle I, \{\Sigma_i\}_{i\in I}, \{H_i\}_{i \in I}\rangle,$$

mit H_i wie in (7.1) heißt *gemischte Erweiterung* des Spiels Γ.

Das Verhalten der Spieler wird sich im Laufe des Spieles, abhängig vom Verhalten der Gegner, natürlich ändern. Ein Spieler wird seine Strategie immer dann ändern, wenn ihm dies bei gleichbleibendem Verhalten der Gegner einen Vorteil verschafft. Ist dies für keinen Spieler der Fall, so tritt ein Gleichgewicht ein.
Ist $s = (s_1, \dots, s_n) \in S$, so schreiben wir für die Situation $(s_1, \dots, s_{i-1}, s_i', s_{i+1}, \dots, s_n)$ im folgenden kurz $s \| s_i'$.

Definition 7.4 Sei $\Gamma = \langle I, \{S_i\}_{i\in I}, \{H_i\}_{i \in I}\rangle$ ein Spiel. Eine Situation $s = (s_1, \dots, s_n) \in S$ heißt *zulässig* für Spieler i, wenn

$$H_i(s\|s_i') \leq H_i(s), \quad \text{für alle } s_i' \in S_i.$$

Ist eine Situation zulässig für alle Spieler, so heißt sie *Gleichgewichtspunkt*. Ein *Gleichgewichtspunkt in gemischten Strategien* ist ein Gleichgewichtspunkt in Γ^*.

Eine wesentliche Frage in der Spieltheorie ist, ob solche Gleichgewichtspunkte existieren und wie man sie bestimmen kann.

7.3 Endliche Matrixspiele

Endliche Nullsummenspiele mit 2 Spielern sind charakterisiert durch die Matrix der Auszahlungen eines Spielers. Darum werden sie auch kurz Matrixspiele genannt.
Gilt $|S_1| = m$ und $|S_2| = n$, so sei für $i = 1, \dots, m$ und $j = 1, \dots, n$ a_{ij} die Auszahlung an Spieler 1 bei Eintreffen der Situation (i, j). Dann ist das Spiel gegeben durch die Matrix $A = (a_{ij})$.

Definition 7.5 Sei A eine $(m \times n)$–Matrix. Dann heißt (i^*, j^*) ein *Sattelpunkt* der Matrix, wenn

$$a_{ij^*} \leq a_{i^*j^*} \leq a_{i^*j}, \quad \text{für } i = 1, \dots, m,\ j = 1, \dots, n.$$

Offensichtlich ist ein Sattelpunkt der Matrix gleichbedeutend mit einem Gleichgewichtspunkt des zugehörigen Spiels (in reinen Strategien).

Satz 7.1 *Die Matrix A hat genau dann einen Sattelpunkt, wenn*

$$\max_i \min_j a_{ij} = \min_j \max_i a_{ij}.$$

Beweis: Zunächst gilt in jedem Falle "$\leq$", da für alle i gilt $\min_j a_{ij} \leq \min_j \max_i a_{ij}$.
(i) Sei (i^*, j^*) Sattelpunkt. Dann gilt nach Definition $\max_i a_{ij^*} = \min_j a_{i^*j} = a_{i^*j^*}$. Darum folgt

$$\max_i \min_j a_{ij} \geq \min_j a_{i^*j} = a_{i^*j^*} = \max_i a_{ij^*} \geq \min_j \max_i a_{ij}.$$

(ii) Es gelte $\max_i \min_j a_{ij} = \min_j \max_i a_{ij}$. Setze $b(i) = \min_j a_{ij}$ und $c(j) = \max_i a_{ij}$. Dann gilt

$$b(i) \leq a_{ij} \leq c(j), \quad \text{für alle } i, j.$$

Sei i^* mit $b(i^*) = \max_i b(i)$ und j^* mit $c(j^*) = \min_j c(j)$. Nach Voraussetzung gilt $b(i^*) = c(j^*)$, also mit $i = i^*$ und $j = j^*$ in obiger Gleichung

$$a_{i^*j^*} = b(i^*) = c(j^*) = \min_j a_{i^*j} = \max_i a_{ij^*},$$

woraus die Behauptung folgt. □

Die linke Seite ist der maximale Mindestgewinn für Spieler 1 und die rechte Seite der minimale Maximalverlust für Spieler 2.

Beispiel 7.3 Die Matrix $\begin{pmatrix} 5 & 1 & 3 \\ 3 & 2 & 4 \\ -3 & 0 & 1 \end{pmatrix}$ hat einen Sattelpunkt in der zweiten Zeile und zweiten Spalte, während $\begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$ keinen Sattelpunkt besitzt.

Anders verhält es sich bei der gemischten Erweiterung von Matrixspielen. Hier können die Spieler aus den Mengen

$$X = \{x = (x_1, \ldots, x_m) : 0 \leq x_i \leq 1, \sum_i x_i = 1\}, Y = \{y = (y_1, \ldots, y_n) : 0 \leq y_j \leq 1, \sum_j y_j = 1\}$$

auswählen. Geometrisch sind dies Simplizes in $\mathbb{R}^m$ bzw. $\mathbb{R}^n$.
Die Auszahlung in einer gemischten Situation (x, y) ist nach (7.1)

$$\sum_{i=1}^{m} \sum_{j=1}^{n} x_i y_j a_{ij} = x^\top A y.$$

Wieder will Spieler 1 den Mindestgewinn maximieren und Spieler 2 den Maximalverlust minimieren. Dies führt zu den beiden Werten

$$v_1 = \max_{x \in X} \min_{y \in Y} x^\top A y \quad \text{und} \quad v_2 = \min_{y \in Y} \max_{x \in X} x^\top A y.$$

Bemerkung: Es gilt immer $v_1 \leq v_2$.

Satz 7.2 (Minimax–Theorem) *Für ein endliches Matrixspiel mit $(m \times n)$–Matrix A gilt*

$$\max_{x\in X}\min_{y\in Y} x^\top Ay = \min_{y\in Y}\max_{x\in X} x^\top Ay = v(A).$$

$v(A)$ wird der Wert *des Spiels genannt.*

Beweis: Für jedes $x \in X$ gilt

$$\min_{y\in Y} x^\top Ay = \min_{y\in Y}(x^\top A)y = \min_{1\le j\le n} x^\top A^j,$$

wobei A^j die j–te Spalte von A ist. Führe Variable α ein, dann ist die Bestimmung von v_1 äquivalent zu

$$\alpha \to \max, \qquad x \ge 0,\ x_1 + \ldots + x_m = 1, \quad x^\top A^j - \alpha \ge 0,\ j = 1, \ldots, n.$$

Analog ist die Bestimmung von v_2 äquivalent zu

$$\beta \to \min, \qquad y \ge 0,\ y_1 + \ldots + y_n = 1, \quad A_i y - \beta \le 0,\ i = 1, \ldots, m.$$

Beide Probleme sind zueinander dual (vgl. Optimierung I). Da beide zulässige Mengen nicht leer sind, folgt die Behauptung aus dem Dualitätssatz der Linearen Optimierung. $\square$

Bemerkung: (1) Das Paar (x_0, y_0) ist genau dann ein Gleichgewichtspunkt des Spiels, wenn x_0 optimal für Spieler 1 und y_0 optimal für Spieler 2 ist. In diesem Falle gilt dann ${x_0}^\top Ay_0 = v(A)$ und die Umkehrung gilt nicht (Übung).
(2) Der Beweis zeigt, daß der Wert eines Spieles und ein Gleichgewichtspunkt mittels eines LP bestimmt werden kann. Das LP für Spieler 1 ist

$$\alpha \to \max, \qquad x \ge 0,\ x_1 + \ldots + x_m = 1, \quad x^\top A^j \ge \alpha,\ j = 1, \ldots, n.$$

wobei A^j die j–te Spalte von A ist. Durch Substitution $\alpha = x_{m+1} - x_{m+2}$ mit $x_{m+1}, x_{m+2} \ge 0$ und Einführung der Schlupfvariablen $x_{m+3}, \ldots, x_{m+n+2}$ für die n Ungleichungen bringt man das Problem in Standardform. Das (unzulässige) Tableau ist dann

$$\begin{array}{|c|ccc|cc|ccc|}
\hline
0 & 0 & \ldots & 0 & -1 & 1 & 0 & \ldots & 0 \\
\hline
0 & & & & 1 & -1 & & & \\
\vdots & & -A^\top & & \vdots & \vdots & & I & \\
0 & & & & 1 & -1 & & & \\
\hline
1 & 1 & \ldots & 1 & 0 & 0 & 0 & \ldots & 0 \\
\hline
\end{array}$$

Als Basis können nun die n Schlupfvariable und eine (etwa die i–te) Spalte von $-A^\top$ gewählt und das Tableau entsprechend umgeformt werden. Dabei erscheint die i–te Zeile von A als 0–te Spalte. Ist diese ≥ 0, so hat man ein zulässiges Starttableau. Ist dies für keine Zeile von A der Fall, so kann man das äquivalente Problem mit $A + qE$ betrachten, wobei E eine Matrix ist, deren Einträge alle 1 sind.

Entsprechend dem komplementären Schlupf erhalten wir

Satz 7.3 *Sei v der Wert eines Spiels mit Matrix A.*

(a) Ist $x = (x_1, \dots, x_m)$ eine optimale Lösung für Spieler 1 und gilt für eine Spalte A^j von A $x^\top A^j > v$, dann gilt für jede optimale Lösung y für Spieler 2 $y_j = 0$.

(b) Ist $y = (y_1, \dots, y_n)$ eine optimale Lösung für Spieler 2 und gilt für eine Zeile A_i von A $A_i y < v$, dann gilt für jede optimale Lösung x für Spieler 1 $x_i = 0$.

Beweis: Nach dem komplementären Schlupf gilt für optimale Lösungen (x, α) bzw. (y, β) in den LP's aus dem Beweis von Satz 7.2

$$y_j(x^\top A^j - \alpha) = 0 \quad \text{bzw.} \quad x_i(\beta - A_i y) = 0.$$

Da im Optimalfall $\alpha = \beta = v$ gilt, folgt die Behauptung. □

Definition 7.6 Das Spiel zu einer Matrix $A = \operatorname{diag}(a_1, \dots, a_n)$ mit $a_i > 0$, für $i = 1, \dots, n$, heißt *Diagonalspiel.*

Lemma 7.1 *Für ein Diagonalspiel Γ gilt*

(a) Der Wert ist positiv.

(b) Für jede optimale Lösung x von Spieler 1 gilt $x_i > 0$, für $i = 1, \dots, n$.

Beweis: (a)

$$\begin{aligned} v(A) &= \max_{x \in X} \min_{y \in Y} x^\top A y \geq \min_{y \in Y} (1/n, \dots, 1/n) A y \\ &= \min_{y \in Y} 1/n (a_1, \dots, a_n) y \geq 1/n \min\{a_1, \dots, a_n\} \sum_{i=1}^{n} y_i > 0. \end{aligned}$$

(b) Angenommen, $x_i = 0$, für ein optimales x. Da x optimal, gilt $v = \min_{y \in Y} x^\top A y$. Mit $y = e_i$ folgt dann

$$v \leq x^\top A e_i = x^\top a_i e_i = a_i x_i = 0,$$

im Widerspruch zu (a). □

Satz 7.4 *In einem Diagonalspiel mit Matrix* $\operatorname{diag}(a_1, \dots, a_n)$ *ist der Wert* $v = (\sum_{i=1}^{n} (1/a_i))^{-1}$ *und die optimalen Lösungen für Spieler 1 und 2 sind eindeutig bestimmt mit*

$$x = y = v(1/a_1, \dots, 1/a_n).$$

Beweis: Nach Lemma 7.1 (b) gilt für jede optimale Lösung x für Spieler 1 $x_i > 0$, $i = 1, \dots, n$. Somit gilt für alle optimalen y nach Satz 7.3 $A_i y = v$. Es folgt $y = v(1/a_1, \dots, 1/a_n)$. Wegen $\sum_{i=1}^{n} y_i = 1$ gilt $v = (\sum_{i=1}^{n} 1/a_i)^{-1}$. Wieder mit Satz 7.3 folgt wegen $y_i > 0$, $i = 1, \dots, n$, daß $x^\top A^i = v$, für alle i und damit $x = y$. □

Definition 7.7 Ein Spiel heißt *symmetrisch*, wenn die zugehörige $(n \times n)$–Matrix A schiefsymmetrisch ist, d. h. $A^\top = -A$.

Satz 7.5 *Der Wert eines symmetrischen Spiels ist 0. Ferner sind die Mengen der optimalen Strategien für Spieler 1 und 2 identisch.*

Beweis: Sei $x \in X$ beliebig. Dann gilt $x^\top Ax = x^\top A^\top x = -x^\top Ax$. Also $x^\top Ax = 0$, für alle $x \in X$. Es folgt

$$v = \max_x \min_y x^\top Ay \le \max_x x^\top Ax = 0$$

und analog

$$v = \min_y \max_x x^\top Ay \ge \min_y y^\top Ay = 0.$$

Also gilt $v = 0$. Ferner gilt: x_0 optimal für Spieler 1 $\Leftrightarrow$ $\min_{x \in X} {x_0}^\top Ax = v = 0 \Leftrightarrow \min_{x \in X} x^\top A^\top x_0 = 0 \Leftrightarrow \min_{x \in X} -x^\top Ax_0 = 0 \Leftrightarrow$ $\max_{x \in X} x^\top Ax_0 = 0 \Leftrightarrow x_0$ optimal für Spieler 2. □

Es ist möglich, daß gewisse Strategien überhaupt nicht in Frage kommen.

Definition 7.8 Sei A eine $(m \times n)$–Matrix.
(1) Eine Zeile A_i *dominiert* die Zeile A_k, falls für alle $j = 1, \ldots, n$ gilt $a_{ij} \ge a_{kj}$.
(2) Eine Spalte A^j *dominiert* die Spalte A^k, falls für alle $i = 1, \ldots, m$ gilt $a_{ij} \le a_{ik}$.

Vollkommen plausibel ist dann

Satz 7.6 *Es seien die Zeilen $A_{i_1}, \ldots, A_{i_k}$ dominiert von anderen Zeilen und die A_{i_j} seien paarweise verschieden. Dann gibt es eine optimale Lösung x für Spieler 1 mit $x_{i_1} = \ldots = x_{i_k} =$ 0.*

Beweis: Es werde die Zeile A_k von A_i dominiert und es sei $x \in X$ optimal mit $x_k > 0$. Dann setze $x' \in X$ mit $x'_k = 0$ und $x'_i = x_i + x_k \le 1$. Für alle $y \in Y$ gilt dann

$$(x' - x)^\top Ay = x_k A_i y - x_k A_k y = x_k(A_i - A_k)y \ge 0.$$

Also $\min_y (x')^\top Ay \ge \min_y x^\top Ay = v(A)$. Damit ist auch x' optimal mit einem x_{i_k} weniger. Nacheinander kann man so die x_{i_j} eliminieren. Dabei fahre man jeweils mit einer Zeile fort, die keine der verbleibenden dominiert. □

Bemerkung: (1) Eine entsprechende Aussage gilt natürlich für dominierte Spalten und Spieler 2. Dominierte Zeilen oder Spalten können somit weggelassen werden und so läßt sich das Problem reduzieren.
(2) Der Begriff der Dominanz läßt sich auf Konvexkombinationen von Zeilen, d. h. gemischte Strategien erweitern.

Beispiel 7.4 Gegeben sei das Spiel mit der Matrix $\begin{pmatrix} 2 & 0 & 1 & 4 \\ 1 & 2 & 5 & 3 \\ 4 & 1 & 3 & 2 \end{pmatrix}$. Offenbar dominiert die zweite Spalte die vierte. Nach Streichen der vierten Spalte ergibt sich die Matrix $\begin{pmatrix} 2 & 0 & 1 \\ 1 & 2 & 5 \\ 4 & 1 & 3 \end{pmatrix}$, in der die dritte Zeile die erste dominiert. In der so erhaltenen Matrix $\begin{pmatrix} 1 & 2 & 5 \\ 4 & 1 & 3 \end{pmatrix}$ dominiert die zweite Spalte die dritte und übrig bleibt $\begin{pmatrix} 1 & 2 \\ 4 & 1 \end{pmatrix}$.

Besonders einfach zu lösen sind (2×2)–Spiele. Besitzt die zugehörige Matrix einen Sattelpunkt, so bilden die beiden zugehörigen reinen Strategien einen Gleichgewichtspunkt. Ansonsten gilt

Satz 7.7 *Hat die (2×2)–Matrix A keinen Sattelpunkt, so sind die eindeutig bestimmten optimalen Strategien und der Wert des Spiels gegeben durch*

$$x = \frac{e^\top A^*}{e^\top A^* e}, \quad y = \frac{A^* e}{e^\top A^* e}, \quad v = \frac{\det A}{e^\top A^* e},$$

wobei $e = (1,1)^\top$ und A^* die Adjungierte von A ist.

Beweis: Es gebe keinen Sattelpunkt. Seien x, y optimale Lösungen.
Angenommen, ein $x_i = 0$. Dann gilt $x_{i^*} = 1$ mit $i^* \neq i$. Wäre auch ein $y_j = 0$, dann wäre mit $j^* \neq j$ $\;(i^*, j^*)$ ein Sattelpunkt: $v = x^\top A y = a_{i^*j^*}$ und

$$a_{ij^*} = {e_i}^\top A e_{j^*} \leq {e_{i^*}}^\top A e_{j^*} = a_{i^*j^*} \leq {e_{i^*}}^\top A e_j = a_{i^*j}.$$

Also gilt $y_1, y_2 > 0$ und damit nach Satz 7.3 ${e_{i^*}}^\top A = (v, v)$, also $A_{i^*} = (v, v)$. Da kein Sattelpunkt existiert, gilt $A_i = (v + \epsilon, v + \delta)$, mit $\epsilon, \delta > 0$. Damit würde $\min_y e_i A y = \min(v + \epsilon, v + \delta) > v$ gelten, im Widerpruch zur Maximalität von e_{i^*}.
Also gilt $x_1, x_2 > 0$ und analog $y_1, y_2 > 0$. Nach Satz 7.3 folgt $x^\top A^j = v,\ A_j y = v,\ j = 1, 2$. In Matrixschreibweise also $A^\top x = ve$ und $Ay = ve$. Zusammen mit $x^\top e = y^\top e = 1$ ergibt sich

$$\begin{pmatrix} a_{11} & a_{21} & -1 \\ a_{12} & a_{22} & -1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \qquad \begin{pmatrix} a_{11} & a_{12} & -1 \\ a_{21} & a_{22} & -1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Die beiden Matrizen haben Determinante $D := a_{11} + a_{22} - a_{12} - a_{21}$.
Angenommen $D = 0$, d. h. $a_{11} + a_{22} = a_{12} + a_{21}$. O. E. sei a_{11} maximal. Dann ist a_{22} minimal und $a_{22} \leq a_{12} \leq a_{11}$ und darum $(1, 2)$ ein Sattelpunkt. Widerspruch.
Also ist $D \neq 0$ und die Lösungen sind eindeutig bestimmt. Die Cramersche Regel liefert

$$x_1 = \frac{a_{22} - a_{21}}{D},\ x_2 = \frac{a_{11} - a_{12}}{D},\ v = \frac{\det A}{D}, \quad y_1 = \frac{a_{22} - a_{12}}{D},\ x_2 = \frac{a_{11} - a_{21}}{D}.$$

Mit $e^\top A^* e = D$ folgt die Behauptung. $\square$

Neben den (2×2)–Spielen lassen sich $(2 \times n)$ und $(m \times 2)$–Spiele sehr einfach lösen. Wir betrachten hier nur $(2 \times n)$–Spiele, die Behandlung der $(m \times 2)$–Spiele verläuft analog. Aufgabe des ersten Spielers ist es, $v(x) = \min_j \{a_{1j}x_1 + a_{2j}x_2\}$ zu maximieren, also ist wegen $x_2 = 1 - x_1 =: t$ die Aufgabe, die Funktion

$$v(t) = \min_j \{(a_{2j} - a_{1j})t + a_{1j}\}$$

in dem Intervall $t \in [0,1]$ zu maximieren. Die n linearen Funktionen kann man graphisch darstellen und die Funktion v als stückweise lineare Funktion bestimmen. Das Maximum kann man dann anschließend ablesen.
Eine weitere Lösungsmethode für Matrixpiele ist das *fiktive Auspielen.* Bei einem Spiel mit vielen Runden merkt sich jeder Spieler die Entscheidungen seines Gegners. Dabei setzt Spieler 1 die $y_1, \ldots, y_n$ gemäß der relativen Häufigkeiten der Strategien von Spieler 2 bis zu diesem Zeitpunkt. Dann wählt er seine (reine) Startegie i so, daß $A_i y$ maximal ist. Entsprechend verhält sich Spieler 2. Man kann zeigen, daß jeder Häufungspunkt der so erhaltenen Folgen von relativen Häufigkeiten ein Gleichgewichtspunkt ist.

7.4 Existenz von Gleichgewichtspunkten

In diesem Abschnitt betrachten wir Spiele, die nicht notwendig Nullsummenspiele sind. Es wird gezeigt, daß für eine große Klasse von Spielen ein Gleichgewichtspunkt existiert. Dazu müssen einige Hilfsmittel zur Verfügung gestellt werden.

Lemma 7.2 *Sei $X \subset \mathbb{R}^n$ abgeschlossen und $f : X \to X$ eine Abbildung. Gibt es ein $\alpha < 1$ mit*

$$\|f(x) - f(y)\| \leq \alpha \|x - y\|, \quad \textit{für alle } x, y \in X$$

(d. h. f ist kontrahierend), so hat f einen Fixpunkt, d. h. es gibt ein $x \in X$ mit $f(x) = x$.

Beweis: Sei $x^0 \in X$ beliebig und $x^{n+1} := f(x^n)$, für $n = 0, 1, 2, \ldots$. Es wird gezeigt, daß $\{x^n\}$ eine Cauchy–Folge ist: Sei $\epsilon > 0$. Für $n > m$ gilt

$$\begin{aligned}
\|x^n - x^m\| &\leq \|x^n - x^{n-1}\| + \|x^{n-1} - x^{n-2}\| + \ldots + \|x^{m+1} - x^m\| \\
&\leq \|x^{m+1} - x^m\| + \alpha\|x^{m+1} - x^m\| + \ldots + \alpha^{n-m-1}\|x^{m+1} - x^m\| \\
&\leq \|x^{m+1} - x^m\|(1 + \alpha + \alpha^2 + \ldots) = \frac{1}{1-\alpha}\|x^{m+1} - x^m\| \\
&\leq \frac{\alpha}{1-\alpha}\|x^m - x^{m-1}\| \leq \ldots \leq \frac{\alpha^m}{1-\alpha}\|x^1 - x^0\| < \epsilon,
\end{aligned}$$

für $m > m_0$. Damit ist $\{x^n\}$ konvergent gegen ein $x \in X$, da X abgeschlossen. f ist (Lipschitz–) stetig und darum folgt

$$x = \lim x^{n+1} = \lim f(x^n) = f(\lim x^n) = f(x).$$

□

Im folgenden sei $S^{n-1} = \mathrm{bd} B^n = \{x \in \mathbb{R}^n : \|x\| = 1\}$. Ein *Tangentialfeld* ist eine Abbildung $v : S^{n-1} \to \mathbb{R}^n$ mit $x^\top v(x) = 0$.

Satz 7.8 *Ist n ungerade, dann hat jedes stetige Tangentialfeld v auf S^{n-1} einen Nullpunkt.*

Bemerkung: (1) Für gerade Dimension gibt es solche Tangentialfelder, z. B.

$$v(x_1, \ldots, x_{2k}) = (x_2, -x_1, \ldots, x_{2k}, -x_{2k-1}).$$

(2) Eine populäre Formulierung für $n = 3$ ist "Jeder stetig gekämmte Igel hat einen Glatzpunkt".

Beweis: Es reicht zu zeigen, daß es kein stetig differenzierbares $w : S^{n-1} \to \mathbb{R}^n$ gibt mit $w(x) \neq 0$ und $x^\top w(x) = 0$:
Angenommen, dann gibt es ein *stetiges* $v : S^{n-1} \to \mathbb{R}^n$ mit diesen Eigenschaften. O. E. sei $\|v(x)\| = 1$, für alle $x \in S^{n-1}$ (sonst betrachte man $v/\|v\|$). Setze v auf $\mathbb{R}^n$ fort durch

$$v(x) = \begin{cases} \|x\| v(\frac{x}{\|x\|}), & \text{für } x \neq 0 \\ 0, & \text{für } x = 0. \end{cases}$$

Dann ist v auf dem Würfel $W = [-1, 1]^n$ stetig. Da die Polynome dicht in den stetigen Funktionen liegen (Approximationssatz von Weierstraß), gibt es eine Funktion $p : W \to \mathbb{R}^n$ mit $p = (p_1, \ldots, p_n)$, $p_1, \ldots, p_n$ Polynome und $\|v(x) - p(x)\| \leq 1/3$, für alle $x \in W$. Dann setze

$$w(x) := p(x) - (x^\top p(x)) \cdot x, \quad \text{für } x \in S^{n-1}.$$

Es folgt $x^\top w(x) = x^\top p(x) - (x^\top p(x))\|x\| = 0$, für alle $x \in S^{n-1}$ und mit der Dreiecksungleichung und der Ungleichung von Cauchy–Schwarz

$$\|w(x)\| = \|p(x) - (x^\top (p(x) - v(x))) \cdot x\| \geq \|p(x)\| - \|p(x) - v(x)\| \geq 2/3 - 1/3 = 1/3,$$

für alle $x \in S^{n-1}$. Damit ist w ein stetig differenzierbares Tangentialfeld ohne Nullpunkt. Widerspruch.
Sei nun also angenommen, daß es ein stetig differenzierbares $v : S^{n-1} \to \mathbb{R}^n$ gibt mit $v \neq 0$ und $x^\top v(x) = 0$. Wieder sei o. E. $\|v(x)\| = 1$, für alle $x \in S^{n-1}$. Setze

$$A := \{x \mid 1/2 \leq \|x\| \leq 3/2\}$$

und v sei auf A (sogar auf $\mathbb{R}^n \setminus \{0\}$) erweitert durch $v(x) = \|x\| v(x/\|x\|)$. Damit ist v stetig differenzierbar auf A mit $x^\top v(x) = 0$ und $\|v(x)\| = \|x\|$. Es gibt ein $\lambda > 0$ mit

$$\|v(x) - v(y)\| \leq \lambda \|x - y\|, \quad \text{für alle } x, y \in A.$$

Um dies zu zeigen, reicht es, nur $x, y \in A$ mit $\|x - y\| \leq 1/4$ zu betrachten, da v auf A beschränkt ist. Nach dem Mittelwertsatz gibt es ein $\xi \in B = \{x \mid 1/4 \leq \|x\| \leq 3/2\}$, so daß $v(x) - v(y) = \operatorname{grad} v(\xi)^\top \cdot (x - y)$. Damit folgt die Aussage aus der Kompaktheit von B und der Stetigkeit von $\operatorname{grad} f$.
Sei nun $t \in \mathbb{R}$ mit $|t| < \min\{1/3, 1/(2\lambda)\}$. Mit $A_t = \{x \mid 1/2\sqrt{1+t^2} \leq \|x\| \leq 3/2\sqrt{1+t^2}\}$ ist die Funktion $\varphi_t : A \to A_t$, definiert durch

$$\varphi_t(x) := x + tv(x)$$

bijektiv:
(i) Für $x \in A$ gilt

$$\|\varphi_t(x)\|^2 = \|x\|^2 + t^2\|v(x)\|^2 = (1+t^2)\|x\|^2,$$

also $\varphi_t(x) \in A_t$.
(ii) φ_t ist injektiv: Aus $\varphi_t(x) = \varphi_t(y)$ folgt $x - y = t(v(x) - v(y))$ und daraus $\|x-y\| \leq |t| \cdot \lambda \cdot \|x-y\| \leq 1/2\|x-y\|$, also $x = y$.
(iii) φ_t ist surjektiv: Sei $c \in A_t$ und $c' = c/\|c\| \in S^{n-1}$. Die Abbildung $f : A \to \mathbb{R}^n$, definiert durch $f(x) = c' - tv(x)$ bildet A in sich ab, wegen $\|tv(x)\| \leq 1/3 \cdot 3/2 = 1/2$.
Ferner gilt $\|f(x) - f(y)\| = |t| \cdot \|v(x) - v(y)\| \leq |t| \cdot \lambda \cdot \|x-y\| \leq 1/2\|x-y\|$. Nach Lemma 7.2 gibt es ein $x' \in A$ mit $f(x') = x'$, d. h. $\varphi_t(x') = c'$. Somit gilt $1 = \|\varphi_t(x')\| = \sqrt{1+t^2}\,\|x'\|$. Mit $x := \|c\|\, x'$ folgt $\|x\| = \|c\|/\sqrt{1+t^2} \in [1/2, 3/2]$ und $\varphi_t(x) = \|c\|\, x' + t\,\|c\|\, v(x') = \|c\|\, c' = c$.
Nun ist das Volumen von A_t einerseits

$$V(A_t) = V(\sqrt{1+t^2}\, A) = (1+t^2)^{n/2} V(A).$$

Das Differential von φ_t ist $D\varphi_t(x) = I + tDv(x) = (\delta_{ij} + t\alpha_{ij}(x))_{ij}$. Die Funktionaldeterminante ist ein Polynom n–ten Grades in t:

$$\det \varphi_t(x) = 1 + a_1(x)t + \ldots + a_n(x)t^n.$$

Man beachte, daß sie an der Stelle $t = 0$ gleich 1 ist und die $a_1, \ldots, a_n$ stetig sind. Da $\det D\varphi_t(x) > 0$, für kleine t, folgt aus der Substitutionsregel

$$V(A_t) = \int_{A_t} 1\, dy = \int_A \det D\varphi_t(x)\, dx = V(A) + b_1 t + b_2 t^2 + \ldots + b_n t^n.$$

Damit folgt, daß $(1+t^2)^{n/2} = 1 + b_1/V(A)t + \ldots + b_n/V(A)t^n = p(t)$ ein Polynom vom Grad n ist. Sind $z_1, \ldots, z_n \in \mathbb{C}$ die komplexen Nullstellen von p, so folgt $(t+i)^n(t-i)^n = p(t)^2 = (t-z_1)^2 \cdot \ldots \cdot (t-z_n)^2$. Es müssen gleich viele z_j je $+i$ und $-i$ sein. Dies ist aber ein Widerspruch zu n ungerade.
□

Definition 7.9 Eine konvexe kompakte Teilmenge des $\mathbb{R}^n$ heißt *konvexer Körper*.

Lemma 7.3 *Ist $K \subset \mathbb{R}^n$ ein konvexer Körper mit einem inneren Punkt, so gibt es eine bijektive Abbildung $\tau : K \to B^n$, so daß τ und τ^{-1} stetig sind (d. h. τ ist ein Homöomorphismus).*

Beweis: O. E. sei $0 \in \text{int} K$ und $rB^n \subset K \subset RB^n$. Für $x \in \mathbb{R}^n$ setze

$$f(x) := \min\{\lambda \geq 0 \,|\, x \in \lambda K\}.$$

Wegen $x \in \|x\|/r \cdot rB^n \subset \|x\|/r \cdot K$ gilt $f(x) \leq \|x\|/r$ und $f(x) \geq \min\{\lambda \geq 0 \,|\, x \in \lambda \cdot RB^n\} = \|x\|/R$. Ferner gilt nach Definition $f(\alpha x) = \alpha f(x)$, für $\alpha \geq 0$ und $K = \{x \,|\, f(x) \leq 1\}$.
f ist konvex: Für $x, y \in \mathbb{R}^n$ gilt $x \in f(x) \cdot K$ und $y \in f(y) \cdot K$. Damit gilt für alle $\lambda \in [0,1]$ $(\lambda x + (1-\lambda)y) \in (\lambda f(x) + (1-\lambda) f(y)) \cdot K$. Also $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$.
Somit ist f auch stetig (vgl. Optimierung I). Definiere nun

$$\tau : K \to B^n, \quad \tau(x) = \frac{f(x)}{\|x\|} \cdot x, \text{ für } x \neq 0, \; \tau(0) = 0$$

$$\mu : B^n \to K, \quad \mu(x) = \frac{\|x\|}{f(x)} \cdot x, \text{ für } x \neq 0, \; \mu(0) = 0$$

Wegen $x \in K \Rightarrow f(x) \leq 1 \Rightarrow \tau(x) \in B^n$ und $x \in B^n \Rightarrow \|x\| \leq 1 \Rightarrow f(\mu(x)) = \|x\|/f(x) \cdot f(x) = \|x\| \leq 1 \Rightarrow \mu(x) \in K$ sind die Abbildungen wohldefiniert. Weiter gilt nach Definition für $\alpha \geq 0$ $\tau(\alpha x) = \alpha\tau(x)$ und $\mu(\alpha x) = \alpha\mu(x)$. Mit diesen Eigenschaften prüft man leicht nach, daß $\tau \circ \mu = \mathrm{id}$ und $\mu \circ \tau = \mathrm{id}$. Also ist τ bijektiv mit $\tau^{-1} = \mu$.
τ und μ sind für $x \neq 0$ stetig nach Definition. Für $x \to 0$ gilt $\|\tau(x)\| = f(x) \leq \|x\|/r \to 0$ und $\|\mu(x)\| = \|x\|^2/f(x) \leq \|x\|^2/(\|x\|/R) = R\,\|x\| \to 0$. Also sind τ und τ^{-1} stetig. □

Satz 7.9 (Brouwer) *Es sei $K \subset \mathbb{R}^n$ ein konvexer Körper und $f : K \to K$ stetig. Dann hat f einen Fixpunkt, d. h. es gibt ein $x \in K$ mit $f(x) = x$.*

Bemerkung: Für $n = 1$ ist der Fixpunktsatz eine leichte Folgerung aus dem Zwischenwertsatz.

Beweis: O. E. sei $\dim K = n$ (sonst Betrachtung in der affinen Hülle).
Es reicht, den Satz für $K = B^n$ zu zeigen: Sei $f : K \to K$ stetig und $\tau : K \to B^n$ ein Homöomorphismus wie in Lemma 7.3. Dann betrachte $\varphi : B^n \to B^n$ mit $\varphi = \tau \circ f \circ \tau^{-1}$. φ ist stetig, also gibt es ein $x \in B^n$ mit $\varphi(x) = x = \tau \circ f \circ \tau^{-1}(x)$. Damit ist $\tau^{-1}(x) \in K$ mit $f(\tau^{-1}(x)) = \tau^{-1}(x)$.
Sei nun also $f : B^n \to B^n$ stetig und angenommen, f hat keinen Fixpunkt. Für $x \in B^n$ setze

$$w(x) = x - \frac{1 - \|x\|^2}{1 - x^\top f(x)} \cdot f(x).$$

Dann gilt

(1) w ist stetig.

(2) $w(x) = x$, für $\|x\| = 1$.

(3) $w(x) \neq 0$, für $x \in B^n$.

zu (1): Nach Voraussetzung gilt

$$0 < \|x - f(x)\|^2 = \|x\|^2 - 2x^\top f(x) + \|f(x)\|^2 \leq 2 - 2x^\top f(x).$$

Damit ist w wohldefiniert und stetig.
(2) folgt aus der Definition.
zu (3): Für $\|x\| = 1$ gilt $w(x) = x \neq 0$. Sei $\|x\| < 1$ und angenommen, $w(x) = 0$. Dann gilt $(1 - x^\top f(x)) \cdot x = (1 - \|x\|^2) \cdot f(x)$. Also $f(x) = \alpha x$ mit $\alpha(1 - \|x\|^2) = 1 - \alpha\|x\|^2$. Es folgt $\alpha = 1$ und $f(x) = x$. Widerspruch.
Betrachte nun die Abbildung $u : B^n \to S^n \subset B^{n+1}$ definiert durch

$$u(x) = \left(\frac{x}{\lambda(x)}, \frac{\lambda(x) - 1}{\lambda(x)}\right), \quad \text{mit} \quad \lambda(x) = \frac{1}{2}(\|x\|^2 + 1).$$

Man beachte, daß tatsächlich $\|u(x)\|^2 = 1$. (Mit $N = e_{n+1}$ ergibt sich $u(x)$ durch

$$x = (1 - \lambda)N + \lambda u,$$

wobei λ so gewählt wird, daß $\|u\| = 1$; stereographische Projektion).
Für $\|x\| < 1$ setze

$$\mu(t) = x + tw(x), \quad \varphi(t) = u(\mu(t)), \quad v(x) := \varphi'(0) \in \mathbb{R}^{n+1}.$$

Aus $\varphi(t)^\top \varphi(t) \equiv 1$ folgt $\varphi'(t)^\top \varphi(t) = 0$. Also ist $v(x)^\top u(x) = 0$. Differenzieren von $\mu(t) = (1 - \lambda(\mu(t)) \cdot N + \lambda(\mu(t)) \cdot \varphi(t)$ nach t in 0 liefert:

$$w(x) = -(w(x)^\top \operatorname{grad} \lambda(x)) \cdot N + (w(x)^\top \operatorname{grad} \lambda(x))u(x) + \lambda(x)v(x)$$

und mit $\operatorname{grad} \lambda(x) = x$ folgt

$$v(x) = (\lambda(x))^{-1} \left[w(x) + x^\top w(x) \cdot (N - u(x)) \right].$$

Ist $x^\top w(x) = 0$, so ist $v(x) \neq 0$, da $w(x) \neq 0$. Ansonsten ist die letzte Koordinate von $v(x)$ $\neq 0$. In jedem Falle gilt also $v(x) \neq 0$.
Ist x die Umkehrfunktion von u, $x(u) = \left(\frac{-u_{n+1}}{1 - u_{n+1}} \cdot N + \frac{1}{1 - u_{n+1}} \cdot u \right)$, so ist $v(u) := v(x(u))$ ein stetiges Tangentialfeld $\neq 0$ auf der offenen unteren Halbsphäre. Dieses kann stetig auf die abgeschlossene Halbspäre fortgesetzt werden: Mit $u_{n+1} \to 0$ gilt

$$\|x(u)\| \to 1, \ \|u - x(u)\| \to 0, \ \|w(x(u)) - u\| \to 0, \ \lambda(x(u)) \to 1, \quad \text{also } v(u) \to N.$$

Für $u \in S^{n-1}$ setze also $v(u) := N$.
Für u aus der oberen Halbsphäre setze $v(u) := -v(u)$. Damit ist v stetiges Tangentialfeld auf S^n mit $v \neq 0$. Für gerades n ist das ein Widerspruch zu Satz 7.8.
Sei nun n ungerade und $f : B^n \to B^n$ stetig. Für $y \in B^{n+1}$ sei $\pi(y) = (y_1, \ldots, y_n) \in B^n$ und $g : B^{n+1} \to B^n$ $(\subset B^{n+1})$ definiert durch $g(y) = f(\pi(y))$. g ist stetig und $n+1$ gerade. Darum existiert ein $y \in B^{n+1}$ mit $g(y) = y$. Mit $g(y) \in B^n$ folgt $y \in B^n$, d. h. $\pi(y) = y$. Also mit $x := \pi(y) = y \in B^n$ gilt $f(x) = f(\pi(y)) = g(y) = y = x$. □

Satz 7.10 (Nikaido–Isoda) *Es sei* $\Gamma = \langle \{1, \ldots, n\};\ S_1, \ldots, S_n;\ H_1, \ldots, H_n \rangle$ *ein* n–*Personenspiel. Ferner seien die folgenden Bedingungen erfüllt:*

(i) $S_i \subset \mathbb{R}^{d_i}$ *ist ein konvexer Körper,* $i = 1, \ldots, n$.

(ii) $H_i : S = S_1 \times \ldots \times S_n \to \mathbb{R}$ *ist stetig,* $i = 1, \ldots, n$.

(iii) Für jedes $i = 1, \ldots, n$ *und jedes* $x \in S$ *ist die Funktion* $S_i \to \mathbb{R}$, *definiert durch* $x_i \mapsto H_i(x \| x_i)$, *konkav.*

Dann besitzt Γ *mindestens einen Gleichgewichtspunkt.*

Beweis: Angenommen, es existiert kein Gleichgewichtspunkt. Nach Voraussetzung (i) ist $S = S_1 \times \ldots \times S_n$ ein konvexer Körper im $\mathbb{R}^N$ mit $N = \sum_{i=1}^n d_i$. Definiere die Funktion $G : S \times S \to \mathbb{R}$ durch

$$G(x, y) := \sum_{i=1}^n H_i(x \| y_i) \quad (x = (x_1, \ldots, x_n), y = (y_1, \ldots, y_n) \in S).$$

Nach Voraussetzung (ii) ist G stetig. Ferner gilt
x^0 ist Gleichgewichtspunkt $\Leftrightarrow H_i(x^0) = \max_{y_i \in S_i} H_i(x^0 \| y_i)$, für $i = 1, \ldots, n$.
$\Leftrightarrow \sum_{i=1}^n H_i(x^0) = \sum_{i=1}^n \max_{y_i \in S_i} H_i(x^0 \| y_i) = \max_{y \in S} \sum_{i=1}^n H_i(x^0 \| y_i)$
$\Leftrightarrow G(x^0, x^0) = \max_{y \in S} G(x^0, y)$.
Nach Voraussetzung gibt es also zu jedem $x \in S$ ein $y \in S$ mit $G(x,x) < G(x,y)$. Für $y \in S$ sei

$$U(y) := \{x \in S \,|\, G(x,x) < G(x,y)\} \subset S.$$

Wegen der Stetigkeit von G ist $U(y)$ offen (evtl. $U(y) = \emptyset$). Nach obigem gilt aber $S \subset \cup_{y \in S} U(y)$. Da S kompakt ist, kann man nach dem Satz von Heine–Borel aus der offenen Überdeckung eine endliche Teilüberdeckung auswählen. Sei also $S \subset U(y_1) \cup \ldots \cup U(y_k)$. Setze nun für $x \in S$

$$\begin{aligned} d_j(x) &:= \max\{0, G(x, y^j) - G(x,x)\}, \quad j = 1, \ldots, k, \\ g_j(x) &:= \frac{d_j(x)}{\sum_{i=1}^k d_i(x)}, \quad j = 1, \ldots, k \quad \text{und} \\ \varphi(x) &:= \sum_{j=1}^k g_j(x) y^j. \end{aligned}$$

Es gilt $d_j(x) \geq 0$, für alle j, und nach Voraussetzung gilt für mindestens ein j sogar > 0, d. h. $g_j(x)$ ist wohldefiniert. Wegen $\sum_{j=1}^k g_j(x) = 1$ gilt $\varphi(x) \in \mathrm{conv}(y^1, \ldots, y^k) \subset S$. Also $\varphi : S \to S$. Alle d_j und g_j sind stetig und damit auch φ. Nach dem Brouwerschen Fixpunktsatz gibt es somit ein x^0 mit $x^0 = \varphi(x^0) = \sum_{j=1}^k g_j(x^0) y^j$.
Nach Voraussetzung (iii) gilt dann

$$\begin{aligned} G(x^0, x^0) &= \sum_{i=1}^n H_i(x_1^0, \ldots, x_n^0) = \sum_{i=1}^n H_i\left(x_1^0, \ldots, x_{i-1}^0, \sum_{j=1}^k g_j(x^0) y_i^j, x_{i+1}^0, \ldots, x_n^0\right) \\ &\geq \sum_{i=1}^n \sum_{j=1}^k g_j(x^0) H_i(x^0 \| y_i^j) = \sum_{j=1}^k g_j(x^0) G(x^0, y^j) \\ &= \sum_{j=1}^k g_j(x^0)(G(x^0, y^j) - G(x^0, x^0)) + G(x^0, x^0) > G(x^0, x^0), \end{aligned}$$

da mit $g_j(x^0) > 0$ auch $G(x^0, y^j) - G(x^0, x^0) > 0$. Widerspruch. $\square$

Korollar 7.1 (Nash) *Jedes endliche n–Personen–Spiel hat mindestens einen Gleichgewichtspunkt (in gemischten Strategien).*

Beweis: Hat Spieler i d_i reine Strategien zur Verfügung, so ist die Menge der gemischten Strategien

$$\Sigma_i = \{(x_1, \ldots, x_{d_i}) \,|\, x_j \geq 0, \sum_{j=1}^{d_i} x_j = 1\} \subset \mathbb{R}^{d_i}$$

ein konvexer Körper im $\mathbb{R}^{d_i}$.
$H_i(\sigma_1, \ldots, \sigma_n) = \sum_{s_1 \in S_1} \cdots \sum_{s_n \in S_n} H_i(s_1, \ldots, s_n) \cdot \sigma_1(s_1) \cdot \ldots \cdot \sigma_n(s_n)$

ist ein Polynom (homogen vom Grad n), also stetig. Ferner ist die Funktion

$$H_i(\sigma\|\sigma_i) = \sum_{s_i\in S_i}\sigma_i(s_i)\cdot\Bigg(\sum_{s_1\in S_1}\cdots\sum_{s_{i-1}\in S_{i-1}}\sum_{s_{i+1}\in S_{i+1}}\cdots\sum_{s_n\in S_n} H_i(s_1,\dots,s_n)\cdot\sigma_1(s_1)\cdot\ldots\cdot\sigma_{i-1}(s_{i-1})\cdot\sigma_{i+1}(s_{i+1})\cdot\ldots\cdot\sigma_n(s_n)\Bigg)$$

linear, insbesondere also konkav. Mit Satz 7.10 folgt die Existenz eines Gleichgewichtspunktes.

□

7.5 Bimatrixspiele

Im folgenden betrachten wir endliche Spiele mit 2 Spielern. Das Spiel ist komplett durch die beiden Auszahlungsfunktionen bestimmt. Da diese in Matrizen A, B festgehalten werden können, werden solche Spiele kurz *Bimatrixspiele* genannt.
Hier können unangenehme Effekte auftreten. Für 2–Personen–Nullsummenspiele (insbesondere Matrixspiele) gilt die sogenannte *Austauschbarkeit* von Gleichgewichtspunkten: Sind (x^1, y^1) und (x^2, y^2) Gleichgewichtspunkte, dann auch (x^1, y^2) und (x^2, y^1). Ferner sind alle Gleichgewichtspunkte *äquivalent*, d. h. die zugehörigen Auszahlungen sind gleich. Beides trifft auf Nicht–Nullsummenspielen im Allgemeinen nicht zu.

Beispiel 7.5 Das Bimatrixspiel Γ sei gegeben durch die Matrizen

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix}.$$

Hier sind (e_1, e_1) und (e_2, e_2) Gleichgewichtspunkte, aber weder (e_1, e_2) noch (e_2, e_1). Ferner sind die Auszahlungen $(2, 1)$ bzw. $(1, 2)$.

Ferner lassen sich, selbst wenn Austauschbarkeit und Äquivalenz gegeben sind, Gleichgewichtspunkte im Allgemeinen schwerlich als Lösungen des Spiels bezeichnen.

Beispiel 7.6 Das Bimatrixspiel Γ sei gegeben durch die Matrizen

$$A = \begin{pmatrix} 2 & 4 \\ 1 & 3 \end{pmatrix}, \qquad B = \begin{pmatrix} 2 & 1 \\ 4 & 3 \end{pmatrix}.$$

Hier gibt es nur den Gleichgewichtspunkt (e_1, e_1). Aber der Punkt (e_2, e_2) bringt beiden Spielern eine höhere Auszahlung.

Die Existenz eines Gleichgewichtspunktes ist nach dem Satz von Nash schon garantiert. Im folgenden soll eine Methode angegeben werden, wie alle Gleichgewichtspunkte eines Bimatrixspiels bestimmt werden können.

Satz 7.11 *(x, y) ist ein genau dann ein Gleichgewichtspunkt des Bimatrixspiels mit Matrizen A, B, wenn es Zahlen α, β gibt, so daß*

$$\begin{aligned} x^\top Ay - \alpha &= 0 \\ x^\top By - \beta &= 0 \\ Ay - \alpha e &\leq 0 \\ B^\top x - \beta e &\leq 0 \end{aligned}$$

und $x \in X, y \in Y$.

Beweis: Ist (x_0, y_0) ein Gleichgewichtspunkt, dann gilt $x_0 \in X, y_0 \in Y$ und $x^\top Ay_0 \leq {x_0}^\top Ay_0 =: \alpha$, für alle $x \in X$ und ${x_0}^\top By \leq {x_0}^\top By_0 =: \beta$, für alle $y \in Y$. Mit $x = e_i$ bzw. $y = e_j$ folgt $A_i y_0 \leq \alpha$, für $i = 1, \dots, m$ bzw. $B^j x_0 \leq \beta$, für $j = 1, \dots, n$.
Sind umgekehrt die Bedingungen erfüllt, dann gilt
$x^\top Ay_0 - \alpha = x^\top Ay_0 - \alpha x^\top e = x^\top(Ay_0 - \alpha e) \leq 0$, für alle $x \in X$ und
${x_0}^\top By - \beta = {x_0}^\top By - \beta e^\top y = ({x_0}^\top B - \beta e^\top)y \leq 0$, für alle $y \in Y$.
Damit ist (x_0, y_0) ein Gleichgewichtspunkt. □

Bemerkung: Man kann wegen Satz 7.11 auch von dem Gleichgewichtspunkt (x, y, α, β) sprechen. α und β sind die zugehörigen Auszahlungen für Spieler 1 und 2.

Im folgenden seien die Polyeder S, T definiert durch

$$S = \{(x, \beta) \,|\, B^\top x - \beta e \leq 0,\ x \in X\}$$

und

$$T = \{(y, \alpha) \,|\, Ay - \alpha e \leq 0,\ y \in Y\}.$$

Definition 7.10 Das Quadrupel $(x_0, y_0, \alpha_0, \beta_0)$ heißt *extremaler Gleichgewichtspunkt*, wenn (x_0, β_0) eine Ecke von S, (y_0, α_0) eine Ecke von T ist mit ${x_0}^\top(A + B)y_0 = \alpha_0 + \beta_0$.

Bemerkung: Nach Satz 7.11 gilt: $(x_0, y_0, \alpha_0, \beta_0)$ extremaler Gleichgewichtspunkt $\Rightarrow$ (x_0, y_0) Gleichgewichtspunkt.

Satz 7.12 *Das Quadrupel (x, y, α, β) ist ein Gleichgewichtspunkt genau dann, wenn*

$$(x, \beta) = \sum_{i=1}^{r} \lambda_i (x_i, \beta_i), \quad (y, \alpha) = \sum_{j=1}^{s} \mu_j (y_j, \alpha_j), \quad \lambda_i, \mu_j > 0,\ \sum_{i=1}^{r} \lambda_i = \sum_{j=1}^{s} \mu_j = 1,$$

und alle $(x_i, y_j, \alpha_j, \beta_i)$ sind extremale Gleichgewichtspunkte.

Beweis: Nach Satz 7.11 ist (x, y, α, β) genau dann ein Gleichgewichtspunkt, wenn $(x, \beta) \in S$, $(y, \alpha) \in T$ und $x^\top(A + B)y = \alpha + \beta$.
(1) Sei $(\bar{x}, \bar{y}, \bar{\alpha}, \bar{\beta})$ ein Gleichgewichtspunkt. Dann betrachte das LP

$$x^\top(A + B)\bar{y} - \beta - \bar{\alpha} \to \max, \quad \text{unter} \quad (x, \beta) \in S.$$

Dann hat $(\bar{x}, \bar{\beta})$ wegen Satz 7.11 die Zielfunktion 0. Ferner gilt für alle $(x, \beta) \in S$, daß $x^\top(A + B)\bar{y} - \beta - \bar{\alpha} = x^\top(A\bar{y} - \bar{\alpha}e) + \bar{y}^\top(B^\top x - \beta e) \leq 0$. Darum ist $(\bar{x}, \bar{\beta}) \in S$ eine optimale Lösung des LP. Folglich kann man es als Linearkombination der optimalen Ecken von S darstellen

$$(\bar{x}, \bar{\beta}) = \sum_{i=1}^{r} \lambda_i \cdot (x_i, \beta_i), \quad \lambda_i > 0,\ \sum_{i=1}^{r} \lambda_i = 1.$$

Analog zeigt man

$$(\bar{y}, \bar{\alpha}) = \sum_{j=1}^{s} \mu_j \cdot (y_j, \alpha_j), \quad \mu_j > 0,\ \sum_{j=1}^{s} \mu_j = 1,$$

mit Ecken (y_j, α_j) von T. Es folgt

$$\begin{aligned}
\bar{\alpha} + \bar{\beta} &= \bar{x}^\top (A+B)\bar{y} = \left(\sum_{i=1}^{r} \lambda_i x_i\right)^\top (A+B) \left(\sum_{j=1}^{s} \mu_j y_j\right) \\
&= \sum_{i,j} \lambda_i \mu_j \left({x_i}^\top (A+B) y_j\right) \\
&\leq \sum_{i,j} \lambda_i \mu_j (\alpha_j + \beta_i) = \sum_{i,j} \mu_j \alpha_j \lambda_i + \sum_{i,j} \lambda_i \beta_i \mu_j \\
&= \sum_j \mu_j \alpha_j + \sum_i \lambda_i \beta_i = \bar{\alpha} + \bar{\beta}.
\end{aligned}$$

Damit gilt in allen Abschätzungen Gleichheit, d. h. es gilt ${x_i}^\top(A + B)y_j = \alpha_j + \beta_i$, für alle i und j. Also sind alle $(x_i, y_j, \alpha_j, \beta_i)$ extremale Gleichgewichtspunkte.
(2) Es seien (x, β) und (y, α) von der geforderten Gestalt. Da S und T konvex sind, liegen beide damit in S bzw. T. Wie oben gilt dann $x^\top(A + B)y = \alpha + \beta$. □

Beispiel 7.7 Das Bimatrixspiel Γ sei gegeben durch die Matrizen

$$A = \begin{pmatrix} 2 & 1 & 2 \\ 1 & 3 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 & 2 & 2 \\ 3 & 2 & 2 \end{pmatrix}.$$

Durch graphische Darstellung findet man die Ecken von S:
$x_1 = (1, 0)^\top, \beta_1 = 2;\ x_2 = (0, 1)^\top, \beta_2 = 3;\ x_3 = (1/2, 1/2), \beta_3 = 2.$
und die Ecken von T:
$y_1 = (1, 0, 0)^\top, \alpha_1 = 2;\ y_2 = (0, 1, 0)^\top, \alpha_2 = 3;\ y_3 = (0, 0, 1)^\top, \alpha_3 = 2;$
$y_4 = (2/3, 1/3, 0)^\top, \alpha_4 = 5/3;\ y_5 = (0, 1/2, 1/2)^\top, \alpha_5 = 3/2.$
Einsetzen ergibt die extremalen Gleichgewichtspunkte
$(x_1, y_3),\ (x_1, y_5),\ (x_3, y_4),\ (x_3, y_5).$
Die Gesamtheit der Gleichgewichtspunkte besteht dann aus drei Komponenten

$$\{x_1\} \times \mathrm{conv}\{y_3, y_5\}\ \cup\ \{x_3\} \times \mathrm{conv}\{y_4, y_5\}\ \cup\ \mathrm{conv}\{x_1, x_3\} \times \{y_5\}.$$

Beschränkt man sich darauf, nur einen Gleichgewichtspunkt zu finden, so ergibt sich noch ein wirkungsvollerer Zugang.
Es sei E die Matrix, deren Einträge alle 1 sind. Durch Addieren von Vielfachen von E zu A und B erhalten wir offensichtlich ein äquivalentes Spiel. Darum kann man annehmen, daß alle Einträge in A und B negativ sind. Nach Satz 7.11 sind dann auch $\alpha, \beta < 0$.

Satz 7.13 *Sei Γ ein Bimatrixspiel mit Matrizen A, B. Ist $(u, \hat{u}, z, \hat{z})$ eine Lösung von*

$$\min\{u^\top z + \hat{u}^\top \hat{z} \mid u + A\hat{z} = -e,\ u \geq 0,\ \hat{z} \geq 0, \quad \hat{u} + B^\top z = -e,\ \hat{u} \geq 0,\ z \geq 0\},$$

so ist (x, y, α, β) mit $\alpha = -(e^\top \hat{z})^{-1}$, $\beta = -(e^\top z)^{-1}$ und $x = -\beta z,\ y = \alpha\hat{z}$ ein Gleichgewichtspunkt von Γ.

Beweis: Der Zielfunktionswert ist offensichtlich immer ≥ 0. Wir behaupten, daß der Minimalwert 0 angenommen wird. Zum Beweis benutzen wir einen Gleichgewichtspunkt (x, y, α, β) (der nach Nash existiert) von Γ und o. E. sei $\alpha, \beta < 0$. Setze

$$u = \frac{Ay}{\alpha} - e \in \mathbb{R}^m, \quad \hat{u} = \frac{B^\top x}{\beta} - e \in \mathbb{R}^n, \quad z = -\frac{x}{\beta} \in \mathbb{R}^m, \quad \hat{z} = -\frac{y}{\alpha} \in \mathbb{R}^n.$$

Dann ist $(u, \hat{u}, z, \hat{z})$ zulässig für obiges Problem:
$u = 1/\alpha\ (Ay - \alpha e) \geq 0$, $\hat{u} = 1/\beta\ (B^\top x - \beta e) \geq 0$, $z \geq 0, \hat{z} \geq 0$,
$u + A\hat{z} = (1/\alpha)Ay - e + A(-y/\alpha) = -e, \quad \hat{u} + B^\top z = (1/\beta)B^\top x - e + B^\top(-x/\beta) = -e$
mit Zielfunktionswert

$$u^\top z + \hat{u}^\top \hat{z} = -\frac{1}{\beta}\left(x^\top\left(\frac{Ay}{\alpha} - e\right)\right) - \frac{1}{\alpha}\left(\frac{x^\top B}{\beta} - e\right) y = -\frac{1}{\alpha\beta}(x^\top Ay - \alpha + x^\top By - \beta) = 0$$

(alles mit Satz 7.11).
Ist nun $(u, \hat{u}, z, \hat{z})$ optimal, dann folgt $u^\top z = 0$ und $\hat{u}^\top \hat{z} = 0$. Damit sind alle Bedingungen von Satz 7.11 erfüllt:
Wäre $z = 0$ oder $\hat{z} = 0$, dann wäre $u = -e$ oder $\hat{u} = -e$. Widerspruch. Damit gilt $\alpha, \beta < 0$ und somit $x \geq 0, y \geq 0$.
Ferner gilt $e^\top x = -\beta(e^\top z) = 1$ und $e^\top y = -\alpha(e^\top \hat{z}) = 1$.
$x^\top Ay - \alpha = \alpha\beta(z^\top A\hat{z} - 1/\beta) = \alpha\beta(z^\top(-u - e)) - 1/\beta) = \alpha\beta(-e^\top z - 1/\beta) = 0$,
$x^\top By - \beta = \alpha\beta(z^\top B\hat{z} - 1/\alpha) = \alpha\beta(\hat{z}^\top(-\hat{u} - e)) - 1/\alpha) = \alpha\beta(-e^\top \hat{z} - 1/\alpha) = 0$,
$Ay - \alpha e = -\alpha(A\hat{z} + e) = (-\alpha)(-u) \leq 0, \quad B^\top x - \beta e = -\beta(B^\top z + e) = (-\beta)(-\hat{u}) \leq 0.$ □

Damit ist das Problem des Auffindens eines Gleichgewichtspunktes zurückgeführt auf ein quadratisches Optimierungsproblem. Dieses kann etwa mit dem Wolfeschen Verfahren gelöst werden. Dabei ist gewährleistet, daß der Minimalwert 0 angenommen wird. In der Praxis wird allerdings nicht das Verfahren von Wolfe, sondern eine dem speziellen Problem angepaßte Variante angewandt.

7.6 Spiele über dem Einheitsquadrat

Ein wichtiger Typ unendlicher Spiele ist der, bei dem beide Spieler ein Kontinuum von reinen Strategien besitzen, die gewöhnlich als Intervall $[0, 1]$ dargestellt werden. Die Auszahlung ist eine auf dem Einheitsquadrat $[0, 1] \times [0, 1]$ definierte Funktion $A(x, y)$. Eine gemischte Strategie ist - wie bisher - eine Wahrscheinlichkeitsverteilung auf der Menge aller reinen Strategien. In diesem Fall kann eine gemischte Strategie durch eine *Verteilungsfunktion* $F : [0, 1] \to \mathbb{R}$ beschrieben werden. Diese hat die Eigenschaften

(i) F ist monoton wachsend.

(ii) $F(0) = 0$ und $F(1) = 1$.

(iii) F ist rechtsseitig stetig.

$F(x)$ entspricht der Wahrscheinlichkeit, daß die Zahl aus $[0, x]$ gegriffen wird. Die Wahrscheinlichkeit, daß die Zahl aus dem Intervall $[a, b] \subset [0, 1]$ gegriffen wird, ist dann $F(b) - F(a) \geq 0$. Die reine Strategie $x_0 \in [0, 1]$ kann dargestellt werden durch die Verteilungsfunktion $F_{x_0}(x) = 0$, für $0 \leq x < x_0$ und $F_{x_0}(x) = 1$, für $x_0 \leq x \leq 1$.

Lemma 7.4 *Sei $F : [0, 1] \to \mathbb{R}$ eine Verteilungsfunktion. Dann existieren die links– und rechtsseitigen Grenzwerte $F_-(x)$ und $F_+(x)$ für alle $x \in [0, 1]$ und F ist bis auf höchstens abzählbar viele Punkte stetig.*

Beweis: Setze $\alpha := \sup_{t<x} F(t)$ und $\beta := \inf_{t>x} F(t)$. Dann gilt $F_-(x) = \alpha$ und $F_+(x) = \beta$: Sei $\epsilon > 0$, dann gibt es ein $t_1 < x$ mit $F(t_1) > \alpha - \epsilon$. Da F monoton ist, folgt $\alpha \geq F(t) \geq F(t_1) > \alpha - \epsilon$, für alle $t \in (t_1, x)$. Also gilt $F_-(x) = \alpha$ und analog zeigt man $F_+(x) = \beta$.
Ist F in x unstetig, so gilt $F_+(x) > F_-(x)$ und es gibt ein $q(x) \in [0, 1] \cap \mathbb{Q}$ mit $F_+(x) > q(x) > F_-(x)$. Für $x \neq y$ gilt wegen der Monotonie $q(x) \neq q(y)$. Damit ist die Menge der Unstetigkeitsstellen höchstens abzählbar. $\square$

Wählt Spieler 1 eine reine Strategie x und Spieler 2 eine gemischte Strategie G, so kann die erwartete Auszahlung approximiert werden durch

$$\sum_{j=1}^{n} A(x, \eta_j)(G(y_j) - G(y_{j-1})),$$

mit einer Partition $\{y_0 = 0, y_1, \ldots, y_n = 1\}$ von $[0, 1]$ und Zwischenpunkten $\eta_j \in (y_{j-1}, y_j)$. Konvergieren diese Summen gegen einen festen Wert, wenn man die Feinheit der Partition gegen 0 gehen läßt, so wird dieser Wert das *Riemann–Stieltjes–Integral* von $A(x, \cdot)$ bezüglich G genannt. Man beachte, daß das Riemann–Integral ein Spezialfall mit $G(y) = y$ ist. Die erwartete Auszahlung ist also (im Falle der Existenz)

$$E(x, G) = \int_0^1 A(x, y)\, dG(y)$$

Wählen Spieler 1 und 2 die gemischten Strategien F und G, so ist die erwartete Auszahlung das Riemann–Stieltjes Doppelintegral

$$E(F, G) = \int_{[0,1]^2} A(x, y)\, d(F(x), G(y)),$$

das eine Verallgemeinerung des Riemannschen Mehrfachintegrals ist. Im Falle der Existenz, die wie hier stets voraussetzen, gilt analog zum Satz von Fubini für Riemann–Integrale

$$\begin{aligned} E(F, G) &= \int_0^1 E(F, y)\, dG(y) = \int_0^1 \int_0^1 A(x, y)\, dF(x)\, dG(y) \\ &= \int_0^1 \int_0^1 A(x, y)\, dG(y)\, dF(x) = \int_0^1 E(x, G)\, dF(x). \end{aligned}$$

Über Riemann–Stieltjes–Integrale ist aus der Analysis bekannt:

(1) Existieren die Integrale $\int_a^b f_i(x)\,dg(x)$ und $\int_a^b f(x)\,dg_i(x)$, $i = 1,2$, so existieren auch die folgenden Integrale und es gilt

$$\int_a^b (\lambda_1 f_1(x) + \lambda_2 f_2(x))\,dg(x) = \lambda_1 \int_a^b f_1(x)\,dg(x) + \lambda_2 \int_a^b f_2(x)\,dg(x),$$

$$\int_a^b f(x)\,d(\lambda_1 g_1 + \lambda_2 g_2)(x) = \lambda_1 \int_a^b f(x)\,dg_1(x) + \lambda_2 \int_a^b f(x)\,dg_2(x).$$

(2) Existiert $\int_a^b f(x)\,dg(x)$ und ist $a < c < b$, so existieren auch die folgenden Integrale und es gilt

$$\int_a^b f(x)\,dg(x) = \int_a^c f(x)\,dg(x) + \int_c^b f(x)\,dg(x).$$

(3) Ist f auf $[a,b]$ Riemann–integrierbar und g auf $[a,b]$ differenzierbar und g' Riemann–integrierbar, so existiert $\int_a^b f(x)\,dg(x)$ und es gilt

$$\int_a^b f(x)\,dg(x) = \int_a^b f(t)g'(t)\,dt.$$

(4) Eine Funktion $g : [a,b] \to \mathbb{R}$ heißt *von beschränkter Variation*, wenn

$$V(g) := \sup_Z \sum_{i=1}^n |g(x_i) - g(x_{i-1})| \quad < \infty,$$

wobei das Supremum über alle Partitionen Z genommen wird. Insbesondere sind monotone Funktionen und damit auch Verteilungsfunktionen von beschränkter Variation (mit $V(F) = 1$). Ist f auf $[a,b]$ beschränkt, g auf $[a,b]$ von beschränkter Variation und existiert das Integral $\int_a^b f(x)\,dg(x)$, so gilt

$$\left| \int_a^b f(x)\,dg(x) \right| \le \|f\|_\infty V(g).$$

Die Existenz des Integrals ist z. B. gewährleistet, wenn f auf $[a,b]$ stetig und g auf $[a,b]$ von beschränkter Variation ist.

(5) Es gilt die folgende partielle Integration: Mit $\int_a^b f(x)\,dg(x)$ existiert auch $\int_a^b g(x)\,df(x)$ und es gilt

$$\int_a^b f(x)\,dg(x) + \int_a^b g(x)\,df(x) = f(b)g(b) - f(a)g(a).$$

(6) Ist g eine Treppenfunktion auf $[a,b]$, die genau an den Stellen $x_1, \ldots, x_m$ Sprünge der Größe $h_1, \ldots, h_m$ besitzt, dann gilt für Funktionen f, die in $x_1, \ldots, x_m$ stetig sind,

$$\int_a^b f(x)\,dg(x) = \sum_{i=1}^m f(x_i)h_i.$$

(7) Es sei $f_1(x) \le f_2(x)$, für alle $x \in [a,b]$ und g monoton wachsend auf $[a,b]$. Existieren die Integrale $\int_a^b f_i(x)\,dg(x)$, $i = 1,2$, dann gilt

$$\int_a^b f_1(x)\,dg(x) \le \int_a^b f_2(x)\,dg(x).$$

Wieder will Spieler 1 seinen Mindestgewinn maximieren und Spieler 2 seinen Maximalverlust minimieren. Dies führt zu den beiden Werten

$$v_1 = \sup_F \inf_G E(F,G) \qquad \text{und} \qquad v_2 = \inf_G \sup_F E(F,G).$$

Für alle F gilt $\inf_G E(F,G) \le \inf_G \sup_F E(F,G)$, also $v_1 \le v_2$.
Nun sind zwei Fragen von Interesse

(1) Ist $v_1 = v_2$?
(2) Können $\sup\inf$ bzw. $\inf\sup$ durch $\max\min$ bzw. $\min\max$ ersetzt werden ?

Können beide Fragen mit 'ja' beantwortet werden, dann existieren optimale Strategien. Kann nur die erste Frage positiv beantwortet werden, so hat das Spiel zwar einen Wert $v = v_1 = v_2$, aber keine optimalen Strategien. Gesucht sind dann sogenannte ϵ–optimale Strategien F_ϵ und G_ϵ für Spieler 1 und 2, derart, daß

$$E(F_\epsilon, G) > v - \epsilon \quad \text{und} \quad E(F, G_\epsilon) < v + \epsilon,$$

für alle F und G. Das Spiel ist dann in einem gewissen Sinne stabil, obwohl es nicht wohlbestimmt ist wie bei endlichen Spielen.
Zunächst wird das innere Extremum in v_1 und v_2, wie im Falle der Matrixspiele, von reinen Strategien repräsentiert.

Lemma 7.5 $\quad v_1 = \sup_F \inf_y E(F,y) \qquad$ *und* $\qquad v_2 = \inf_G \sup_x E(x,G).$

Beweis: Es reicht zu zeigen, daß $\inf_y E(F,y) = \inf_G E(F,G)$. Dabei ist '$\ge$' klar. Mit $\alpha := \inf_y E(F,y)$ gilt für alle $y \in [0,1]$ $\; E(F,y) \ge \alpha$. Damit gilt für alle Verteilungsfunktionen G

$$E(F,G) = \int_0^1 E(F,y)\,dG(y) \ge \int_0^1 \alpha\,dG(y) = \alpha \int_0^1 1\,dG(y) = \alpha(G(1) - G(0)) = \alpha.$$

Die Behauptung für v_2 zeigt man analog. $\square$

Im folgenden werden Spiele mit stetiger Auszahlungsfunktion $A(x,y)$ (auch *Kern* genannt) betrachtet.

Lemma 7.6 *Ist $A : [0,1]^2 \to \mathbb{R}$ stetig, so gilt für alle Verteilungsfunktionen F, daß die Funktion*

$$y \mapsto E(F,y) = \int_0^1 A(x,y)\,dF(x)$$

stetig in $[0,1]$ ist. Analog ist für alle G die Funktion

$$x \mapsto E(x,G) = \int_0^1 A(x,y)\,dG(y)$$

stetig in $[0,1]$.

Beweis: Sei $\epsilon > 0$. A ist auf $[0,1]^2$ sogar gleichmäßig stetig. Darum gibt es ein $\delta > 0$ mit $|A(x^1,y^1) - A(x^2,y^2)| < \epsilon$, für $\|(x^1,y^1)-(x^2,y^2)\| < \delta$. Für $|y^1 - y^2| < \delta$ gilt dann

$$|E(F,y^1) - E(F,y^2)| = |\int_0^1 (A(x,y^1) - A(x,y^2))\,dF(x)| \leq \epsilon \cdot V(F) = \epsilon.$$

Analog zeigt man die zweite Aussage. $\square$

Definition 7.11 Eine Folge $\{F_n\}$ von Verteilungsfunktionen heißt *im wesentlichen konvergent* gegen eine Verteilungsfunktion F, wenn für alle Stetigkeitsstellen x von F gilt $F_n(x) \to F(x)$, für $n \to \infty$.

Es gelten die folgenden Sätze von Helly.

Satz 7.14 *Jede Folge von Verteilungsfunktionen $\{F_n\}$ enthält eine Teilfolge, die im wesentlichen gegen eine Verteilungsfunktion F konvergiert.*

Beweis: Es sei $\{a_\nu\} = \mathbb{Q} \cap [0,1]$. Die Folge $\{F_n(a_1)\}$ besitzt eine Teilfolge $\{F_n^1(a_1)\}$, die gegen ein $G(a_1) \in [0,1]$ konvergiert. Die Folge $\{F_n^1(a_2)\}$ besitzt eine Teilfolge $\{F_n^2(a_2)\}$, die gegen ein $G(a_2) \in [0,1]$ konvergiert usw. Erhalte so eine Folge $\{F_n^k\}$ von Teilfolgen mit $F_n^k(a_\nu) \to G(a_\nu)$, für $\nu = 1, \ldots, k$. Man betrachte nun die Diagonalfolge $F_1^1(x), F_2^2(x), F_3^3(x), \ldots$. Für festes k ist F_n^n für $n \geq k$ in $\{F_n^k\}$ enthalten, also gilt $F_n^n(a_k) \to G(a_k)$, für alle k. Da 0 und 1 in $\{a_\nu\}$ enthalten sind, gilt $G(0) = 0$ und $G(1) = 1$. Da F_n^n monoton wachsend ist, folgt $G(a_\nu) \leq G(a_k)$, für $a_\nu \leq a_k$. Setze G auf ganz $[0,1]$ fort durch

$$G(x) := \inf\{G(a_k) \,|\, a_k \geq x\}.$$

Per Definition ist G monoton wachsend. Ferner hat G höchstens abzählbar viele Unstetigkeitsstellen (vgl. Lemma 7.4). In diesen muß nicht unbedingt $G(x) = G_+(x)$ gelten. Es sei aber $F(x) := G_+(x)$, für alle $x \in [0,1]$. Dann ist F eine Verteilungsfunktion, die die gleichen Unstetigkeitsstellen hat wie G und in den Stetigkeitsstellen gilt $F(x) = G(x)$. Sei nun x eine Stetigkeitsstelle von G (und somit von F) und $a_\nu < x < a_k$. Dann gilt $F_n^n(a_\nu) \leq F_n^n(x) \leq F_n^n(a_k)$. Grenzübergang $n \to \infty$ liefert $G(a_\nu) \leq \liminf F_n^n(x) \leq \limsup F_n^n(x) \leq G(a_k)$. Mit $a_\nu \nearrow x$ und $a_k \searrow x$ und der Stetigkeit von G (bzw. F) in x folgt $G(x) \leq \liminf F_n^n(x) \leq \limsup F_n^n(x) \leq G(x)$. Also $F_n^n(x) \to G(x) = F(x)$, für $n \to \infty$. $\square$

Satz 7.15 *Sei $\{F_n\}$ eine Folge von Verteilungsfunktionen, die im wesentlichen gegen eine Verteilungsfunktion F konvergiert. Ferner sei $f : [0,1] \to \mathbb{R}$ stetig. Dann gilt*

$$\lim_{n\to\infty} \int_0^1 f(x)\,dF_n(x) = \int_0^1 f(x)\,dF(x).$$

Beweis: f ist gleichmäßig stetig auf $[0,1]$. Damit gibt es zu jedem $\epsilon > 0$ eine Partition $[x_0 = 0, x_1, \ldots, x_N = 1]$, so daß für $x \in [x_k, x_{k+1}]$ gilt $|f(x) - f(x_k)| < \epsilon$. Definiere die Treppenfunktion f_ϵ durch $f_\epsilon(x) = f(x_k)$, für $x_k \leq x < x_{k+1}$. Ferner seien die x_k o. E. so gewählt, daß

F und alle F_n in x_k stetig sind. Wegen der Stetigkeit von f gilt $M := \max_{x\in[0,1]} |f(x)| < \infty$. Da für alle x_k gilt $F_n(x_k) \to F(x_k)$, folgt für hinreichend große n

$$|F(x_k) - F_n(x_k)| < \frac{\epsilon}{MN}, \qquad \text{für alle } k = 0, \dots, N.$$

Mit der Dreiecksungleichung und den Eigenschaften (5) und (6) von Riemann–Stieltjes–Integralen folgt

$$\begin{aligned}
\left|\int_0^1 f(x)\,dF(x)\right. &- \left.\int_0^1 f(x)\,dF_n(x)\right| \\
&\le \left|\int_0^1 f(x)\,dF(x) - \int_0^1 f_\epsilon(x)\,dF(x)\right| + \left|\int_0^1 f_\epsilon(x)\,dF(x) - \int_0^1 f_\epsilon(x)\,dF_n(x)\right| \\
&\quad + \left|\int_0^1 f_\epsilon(x)\,dF_n(x) - \int_0^1 f(x)\,dF_n(x)\right| \\
&\le \|f - f_\epsilon\|_\infty \cdot V(F) + \left|\int_0^1 f_\epsilon(x)\,d(F - F_n)(x)\right| + \|f_\epsilon - f\|_\infty \cdot V(F_n) \\
&\le 2\epsilon + \left|(F - F_n)(1) f_\epsilon(1) - (F - F_n)(0) f_\epsilon(0) - \int_0^1 (F - F_n)(x)\,df_\epsilon(x)\right| \\
&= 2\epsilon + \left|\sum_{k=1}^N (F - F_n)(x_k)(f_\epsilon(x_k) - f_\epsilon(x_{k-1})\right| \\
&\le 2\epsilon + \sum_{k=1}^N |F(x_k) - F_n(x_k)|(|(f(x_k)| + |f(x_{k-1})|) \quad < 4\epsilon.
\end{aligned}$$

Daraus folgt die Behauptung. $\square$

Satz 7.16 *Ist der Kern $A(x,y)$ stetig auf $[0,1]^2$, dann können* $\sup\inf$ *bzw.* $\inf\sup$ *in v_1 bzw. v_2 durch* $\max\min$ *bzw.* $\min\max$ *ersetzt werden.*

Beweis: Nach Lemma 7.6 ist $y \mapsto E(F,y)$ eine stetige Funktion in $[0,1]$, für alle F. Da $[0,1]$ kompakt ist, wird das Minimum $\min_{y\in[0,1]} E(F,y)$ angenommen. Ferner gibt es nach Definition von v_1 eine Folge von Verteilungsfunktionen F_n mit $\min_{y\in[0,1]} \to v_1$. Nach Satz 7.14 gibt es eine Teilfolge von F_n, die im wesentlichen gegen eine Verteilungsfunktion F konvergiert. O. E. sei diese Teilfolge $\{F_n\}$ selbst. Nach Satz 7.15 gilt für alle $y \in [0,1]$

$$E(F,y) = \int_0^1 A(x,y)\,dF(x) = \lim \int_0^1 A(x,y) dF_n(x) = \lim E(F_n, y) \ge \lim \min_y E(F_n, y) = v_1.$$

Damit gilt $\min_y E(F,y) \ge v_1$ und daher gilt Gleichheit und das Maximum wird in F angenommen. Analog zeigt man die Aussage für v_2. $\square$

Satz 7.17 *Ist der Kern $A(x,y)$ stetig auf $[0,1]^2$, dann gilt $v_1 = v_2$.*

Beweis: (Mit Approximation durch Matrixspiele). Für jedes $n \in \mathbb{N}$ betrachte man die $(n+1) \times (n+1)$–Matrix $A_n = (a_{ij}^n)$ mit

$$a_{ij}^n = A(\frac{i}{n}, \frac{j}{n}), \quad i, j = 0, \dots, n.$$

Das Spiel mit der Maqtrix A_n habe den Wert v_n und die optimalen Strategien $x^n = (x_0^n, \dots, x_n^n)$, $y^n = (y_0^n, \dots, y_n^n)$. A ist auf $[0,1]^2$ gleichmäßig stetig. Darum existiert für alle $\epsilon > 0$ ein $\delta > 0$, so daß aus $\|(x,y) - (x', y')\| < \delta$ folgt $|A(x,y) - A(x',y')| < \epsilon$. Sei n so groß, daß $1/n < \delta$. Definiere die Verteilungsfunktion F_n als Treppenfunktion $F_n(x) := \sum_{i=0}^{[nx]} x_i^n$. Für $j = 0, \dots, n$ gilt

$$E(F_n, \frac{j}{n}) = \int_0^1 A(x, \frac{j}{n})\, dF_n(x) = \sum_{i=1}^{n} A(\frac{i}{n}, \frac{j}{n}) \cdot x_i^n = (A_n)^j \cdot x^n \geq v_n,$$

da x^n optimal für A_n. Für jedes $y \in [0,1]$ gibt es ein j mit $|y - j/n| < \delta$ und damit $|A(x,y) - A(x, j/n)| < \epsilon$, für alle $x \in [0,1]$. Es folgt

$$|E(F_n, y) - E(F_n, j/n)| = |\int_0^1 (A(x,y) - A(x, j/n))\, dF_n(x)| \leq \epsilon \cdot V(F_n) = \epsilon.$$

Somit gilt $E(F_n, y) \geq v_n - \epsilon$, für alle $y \in [0,1]$, also $v_1 \geq v_n - \epsilon$. Analog zeigt man $v_2 \leq v_n + \epsilon$. Damit gilt $v_1 \geq v_2 - 2\epsilon$ und da $\epsilon > 0$ beliebig war, folgt die Behauptung. □

Bemerkung: Der Beweis ist in einem gewissen Sinne konstruktiv. Der Wert und die optimalen Lösungen lassen sich durch Lösung eines Matrixspieles annähern. Dabei hängt die Güte der Approximation von der Gestalt des Kerns ab.

Definition 7.12 Ein Spiel auf dem Einheitsquadrat heißt *konkav–konvex*, wenn sein Kern für jeden Wert von y konkav bzgl. x und für jedes x konvex bzgl. y ist.

Satz 7.18 *Das konkav–konvexe Spiel mit Kern $A(x,y)$ sei stetig. Dann besitzt es optimale reine Strategien.*

Beweis: $S_1 = S_2 = [0,1] \subset \mathbb{R}^1$ sind konvexe Körper. Die Auszahlungsfunktionen $H_1(x,y) = A(x,y)$ und $H_2(x,y) = -A(x,y)$ sind stetig. Ferner sind $x \mapsto H_1(x,y)$ und $y \mapsto H_2(x,y)$ konkav. Die Behauptung folgt somit aus Satz 7.10. □

Beispiel 7.8 Man betrachte das Spiel über $[0,1]^2$ mit Kern $A(x,y) = -2x^2 + y^2 + 3xy - x - 2y$. Wegen $A_{xx}(x,y) = -4$ und $A_{yy}(x,y) = 2$ ist das Spiel (streng) konkav–konvex. Für festes x stellt man durch eine einfache Kurvendiskussion fest, daß $A(x,y)$ genau dann minimal wird, wenn

$$y = \varphi(x) = \begin{cases} (2-3x)/2, & \text{für } x \in [0, 2/3] \\ 0, & \text{für } x \in [2/3, 1] \end{cases}$$

Entsprechend wird für festes y $A(x,y)$ genau dann maximal, wenn

$$x = \psi(y) = \begin{cases} 0, & \text{für } y \in [0, 1/3] \\ (3y-1)/4, & \text{für } y \in [1/3, 1] \end{cases}$$

Für optimale Strategien gilt $A(x,\varphi(x)) = A(\psi(y),y)$. Nach Konstruktion ist $A(x,\varphi(x)) \leq A(x,y) \leq A(\psi(y),y)$. Wegen der strengen Konvexität bzw. Konkavität gilt damit $x = \psi(y)$ und $y = \varphi(x)$. Es ergibt sich somit der Gleichgewichtspunkt $x^0 = 4/17$, $y^0 = 11/17$ und der Wert des Spiels $v = A(x^0,y^0) = -13/17$.

Typische Beispiele für Spiele über dem Einheitsquadrat sind solche, in dem jeder Spieler eine Entscheidung über den Zeitpunkt eines gewissen Vorgangs zu treffen hat. Dabei kommt es in der Regel auf die Reihenfolge an, in der beide Spieler handeln (z. B. Schuß bei einem Duell).

Definition 7.13 Ein Spiel über dem Einheitsquadrat heißt *Zeitspiel*, wenn der Kern die Form

$$A(x,y) = \begin{cases} K(x,y), & \text{für } x < y \\ \varphi(x), & \text{für } x = y \\ L(x,y), & \text{für } x > y \end{cases}$$

hat. Dabei sei $K(x,y)$ stetig auf $\Delta_1 = \{(x,y)\,|\,0 \leq x \leq y \leq 1\}$, $L(x,y)$ stetig auf $\Delta_2 = \{(x,y)\,|\,0 \leq y \leq x \leq 1\}$ und φ stetig auf $[0,1]$.

Satz 7.19 *Es sei $0 \leq a < b \leq 1$, $0 \leq c < d \leq 1$ und F und G stetige Verteilungsfunktionen mit $F(x) = 0$, für $x \in [0,a]$ und $F(x) = 1$, für $x \in [b,1]$ bzw. $G(y) = 0$, für $y \in [0,c]$ und $G(y) = 1$, für $y \in [d,1]$. Ferner seien F und G in $[a,b]$ bzw. $[c,d]$ stetig differenzierbar mit $F'(x) > 0$ und $G'(y) > 0$.*
Dann gilt: Für das Zeitspiel mit Kern wie in Definition 7.13 sind F und G genau dann optimale Lösungen für Spieler 1 bzw. 2 mit $\inf_y E(F,y) = \sup_x E(x,G) = v$, wenn

$$\begin{aligned} E(F,y) &= v, \ \textit{für } y \in (c,d), \\ E(F,y) &\geq v, \ \textit{für alle } y, \\ E(x,G) &= v, \ \textit{für } x \in (a,b), \\ E(x,G) &\leq v, \ \textit{für alle } x. \end{aligned}$$

Sind K und L stetig differenzierbar auf Δ_1 bzw. Δ_2, so gilt darüberhinaus

$$(L(y,y) - K(y,y))F'(y) = \int_c^y K_y(x,y)F'(x)\,dx + \int_y^d L_y(x,y)F'(x)\,dx, \ \textit{für alle } y \in (c,d),$$

$$(K(x,x) - L(x,x))G'(x) = \int_a^x L_x(x,y)G'(y)\,dy + \int_x^b K_x(x,y)G'(y)\,dy, \ \textit{für alle } x \in (a,b).$$

Beweis: (1) Seien F und G von der beschriebenen Form und $\inf_y E(F,y) = v = \sup_x E(x,G)$. Dann sind die Bedingungen $E(F,y) \geq v \geq E(x,G)$ für alle y bzw. x erfüllt. Es gilt $E(F,G) = \int_0^1 E(F,y)\,dG(y) \geq \int_0^1 v\,dG(y) = v$ und $E(F,G) = \int_0^1 E(x,G)\,dF(x) \leq \int_0^1 v\,dF(x) = v$, also gilt in beiden Fällen Gleichheit.
Für $\epsilon > 0$ gilt $E(F,y) = \int_0^{y-\epsilon} A(x,y)\,dF(x) + \int_{y-\epsilon}^{y+\epsilon} A(x,y)\,dF(x) + \int_{y+\epsilon}^1 A(x,y)\,dF(x)$. Da der mittlere Summand Betrag $\leq \|A\|_\infty(F(y+\epsilon) - F(y+\epsilon))$ hat, folgt mit $\epsilon \to 0$

$$E(F,y) = \int_0^y K(x,y)\,dF(x) + \int_y^1 L(x,y)F'(x)\,dx = \int_0^y K(x,y)F'(x)\,dx + \int_y^1 L(x,y)F'(x)\,dx.$$

(Man beachte, daß $F'(a)$ und $F'(b)$ evtl. nicht definiert sind. Für das Integral spielt das aber keine Rolle.) Um zu zeigen, daß $y \mapsto E(F,y)$ stetig in $[0,1]$ ist, reicht es, $y \mapsto \int_0^y K(x,y)F'(x)\,dx$ zu betrachten. Sei $\epsilon > 0$. Da F' und K beschränkt sind, gibt es Zahlen P, Q mit $|F'(x)| \leq P$, für alle $x \in [0,1]$ und $|K(x,y)| \leq Q$, für alle $(x,y) \in \Delta_1$. Ferner ist K auf Δ_1 gleichmäßig stetig. Darum gibt es ein $\delta > 0$ mit $|K(x',y') - K(x,y)| < \epsilon/(2P)$, für $\|(x',y') - (x,y)\| < \delta$. Für $y < y'$ mit $|y - y'| < \min\{\delta, \epsilon/(2PQ)\}$ gilt

$$\begin{aligned}\left|\int_0^y K(x,y)F'(x)\,dx \right. &- \left.\int_0^{y'} K(x,y')F'(x)\,dx\right| \leq \int_0^y |K(x,y) - K(x,y')| \cdot |F'(x)|\,dx \\ &+ \int_y^{y'} |K(x,y')| \cdot |F'(x)|\,dx \leq \frac{\epsilon}{2} + (y'-y)PQ < \epsilon.\end{aligned}$$

Analog zeigt man, daß $x \mapsto E(x,G)$ stetig ist. Nun gilt

$$v = \int_0^1 E(F,y)\,dG(y) = \int_0^1 E(F,y)G'(y)\,dy = \int_c^d E(F,y)G'(y)\,dy.$$

Mit $v = \int_c^d vG'(y)\,dy$ folgt also $\int_c^d (E(F,y) - v)G'(y)\,dy = 0$. Der Integrand ist stetig und ≥ 0. Damit ist er konstant $= 0$, d. h. es gilt $E(F,y) = v$, für alle $y \in (c,d)$. Analog zeigt man $E(x,G) = v$, für $x \in (a,b)$.
(2) Die Umkehrung ist klar.
(3) Es seien K und L stetig differenzierbar. Dann sind die Funktionen $y \mapsto E(F,y)$ und $x \mapsto E(x,G)$ sogar differenzierbar in (c,d) bzw. (a,b):
Dabei reicht es wieder $y \mapsto \int_0^y K(x,y)F'(x)\,dx$ zu betrachten. Mit dem Mittelwertsatz aus der Differential– bzw. der Integralrechnung gilt

$$\begin{aligned}&\frac{\int_0^y K(x,y)F'(x)\,dx - \int_0^{y_0} K(x,y_0)F'(x)\,dx}{y - y_0} \\ &\quad = \int_0^{y_0} \frac{K(x,y) - K(x,y_0)}{y - y_0} F'(x)\,dx + \frac{\int_{y_0}^y K(x,y)F'(x)\,dx}{y - y_0} \\ &\quad = \int_0^{y_0} K_y(x,\xi_1)F'(x)\,dx + K(\xi_2,y)F'(\xi_2), \quad \xi_1, \xi_2 \in (y_0, y). \\ &\quad \to \int_0^{y_0} K_y(x,y_0)F'(x)\,dx + K(y_0,y_0) \cdot F'(y_0),\end{aligned}$$

für $y \to y_0$. Analog zeigt man, daß der zweite Summand abgeleitet $\int_y^1 L_y(x,y)F'(x)\,dx - L(y,y) \cdot F'(y)$ ergibt. $E(F,y)$ ist in (c,d) konstant, hat also Ableitung 0. Daraus ergibt sich die erste Gleichung. Die zweite zeigt man analog. □

Haben die Gleichungen keine Lösung, so gibt es keine optimalen Strategien oder sie sind nicht von dem angegebenen Typ.

Korollar 7.2 *Ist in Satz 7.19 der Kern des Spiels schiefsymmetrisch, d. h. $L(x,y) = -K(y,x)$ und $\varphi(x) = 0$, dann gilt $v = 0$ und man kann $F = G$, $c = a$, $d = b$ setzen. Die Bedingungen werden dann zu*

$$\begin{aligned} E(F,y) &= 0, \text{ für } y \in (a,b), \\ E(F,y) &\geq 0, \text{ für alle } y,\end{aligned}$$

$$2L(y,y)F'(y) = \int_a^y K_y(x,y)F'(x)\,dx + \int_y^b L_y(x,y)F'(x)\,dx, \text{ für alle } y \in (a,b).$$

Beweis: Da das Spiel symmetrisch ist, gilt (nach Übung) $v = 0$ und Spieler 1 und 2 haben die gleichen optimalen Strategien. Man kann daher $F = G$ setzen und die Behauptung folgt. □

Beispiel 7.9 Wir betrachten den folgenden Zweikampf. Zwei Männer starten an verschiedenen Orten zum Zeitpunkt $t = 0$, laufen aufeinander zu und treffen sich zum Zeitpunkt $t = 1$. Beide haben je genau eine Pistole und genau einen Schuß und können diesen irgendwann abfeuern. Trifft einer den anderen, so ist das Duell beendet und der Schütze ist Sieger. Trifft keiner der beiden, dann ist das Duell unentschieden. Schießen und treffen beide gleichzeitig, so wird das Duell ebenfalls unentschieden gewertet. Wir machen die folgenden Annahmen. Erstens nehme die Genauigkeit eines Schusses zu, wenn sich beide Spieler einander nähern und zwar so, daß ein Spieler seinen Gegner mit Wahrscheinlichkeit t trifft, wenn er zum Zeitpunkt t schießt. Zweitens verlaufe das Duell völlig lautlos, d. h. der Spieler weiß nicht, ob sein Gegner bereits geschossen hat (es sei denn, er ist schon getroffen). Der Kern des Spiels berechnet sich wie folgt. Entscheiden sich die beiden Duellanten für die Zeitpunkte x und y mit $x < y$, so erhält Spieler 1 mit Wahrscheinlichkeit x die Auszahlung $+1$ und mit Wahrscheinlichkeit $(1-x)y$ die Auszahlung -1. Also haben wir $K(x,y) = x - y + xy$. Da das Spiel symmetrisch ist, gilt $L(x,y) = x - y - xy$ und $\varphi(x) = 0$. Es folgt $K_y(x,y) = x - 1$ und $L_y(x,y) = -1 - x$ und $L(y,y) = -y^2$. Nehmen wir an, daß die optimale Strategie die Form wie in Satz 7.19 hat, so erhalten wir als notwendige Bedingung

$$-2y^2F'(y) = \int_a^y (x-1)F'(x)\,dx + \int_y^b (-1-x)F'(x)\,dx.$$

Diese Integralgleichung kann durch Differenzieren in eine Differentialgleichung verwandelt werden, nämlich $-4yF' - 2y^2F'' = (y-1)F' + (y+1)F'$, oder vereinfacht, $yF'' = -3F'$. Die Lösung dieser Gleichung lautet $F'(y) = ky^{-3}$. Es müssen noch a, b und k bestimmt werden. Zunächst gilt wegen der Stetigkeit von $E(F,y)$, daß $E(F,b) = 0$. Wäre $b < 1$, dann

$$E(F,1) = \int_a^b (x + 1(x-1))F'(x)\,dx < \int_a^b (x + b(x-1))F'(x)\,dx = E(F,b) = 0.$$

Also gilt $b = 1$. Damit folgt $0 = E(F,1) = k\int_a^1 (2x-1)/x^3\,dx = k(-3/2 + 2/a - 1/(2a^2))$. Damit folgt $a = 1$ oder $a = 1/3$, also $a = 1/3$. Schießlich ist F eine Verteilungsfunktion und darum $\int_{1/3}^1 k/x^3\,dx = 1$, woraus $k = 1/4$ folgt. Damit läßt sich die optimale Strategie der beiden Spieler angeben. Es ist die Verteilungsfunktion F mit der Ableitung

$$F'(x) = \begin{cases} 0, & \text{für } x \in [0, 1/3) \\ \frac{1}{4x^3}, & \text{für } x \in (1/3, 1] \end{cases}$$

und darum

$$F(x) = \begin{cases} 0, & \text{für } x \in [0, 1/3] \\ \frac{9-1/x^2}{8}, & \text{für } x \in (1/3, 1] \end{cases}$$

Man prüft leicht nach, daß dies tatsächlich eine Lösung ist.

7.7 Das Oligopolspiel

In der Ökonomie bezeichnet man als *Oligopol* eine Wettbewerbssituation, bei der $n \geq 2$ Produzenten versuchen, in einem Markt ein bestimmtes Produkt abzusetzen. Als Modell für die Konkurrenz unter n Oligopolisten soll im folgenden ein strategisches n–Personenspiel verwendet werden, bei dem zwar einige vereinfachende Annahmen gemacht werden, das sich aber dennoch gut zur exemplarischen Beschreibung der Anwendungsmöglichkeiten der Spieltheorie in den Wirtschaftswissenschaften eignet.

Es wird vorausgesetzt, daß die n Produzenten ein beliebig teilbares Produkt (z. B. Benzin, Heizöl, Bier etc.) herstellen. Dabei möge der Produzent i über eine Produktionskapazität K_i verfügen, $i = 1, \ldots, n$. Seine reinen Strategien bestehen nun darin, eine bestimmte Menge $x_i \in [0, K_i]$ seines Produktes herzustellen und auf dem Markt anzubieten, so daß insgesamt eine Menge von $\sum_{i=1}^n x_i$ angeboten wird. Der Preis pro Einheit richtet sich nach der insgesamt angebotenen Menge und wird mit

$$p = f\left(\sum_{i=1}^n x_i\right)$$

bezeichnet. Die dadurch gegebene Funktion $f : [0, \sum_{i=1}^n K_i] \to \mathbb{R}$ gibt also zu einer angebotenen Menge denjenigen Preis pro Einheit an, bei dem gerade das Angebot mit der Nachfrage übereinstimmt. Deshalb wird f auch als *Nachfragefunktion* bezeichnet. Wenn der Produzent i dann x_i Einheiten seines Produktes anbietet, kann er dafür am Markt einen Gesamtbetrag von

$$x_i \cdot f\left(\sum_{j=1}^n x_j\right)$$

erlösen. Um seinen unternehmerischen Gewinn zu errechnen, sind von diesem Erlös noch die Kosten $k_i(x_i)$ für die Herstellung der x_i Einheiten abzuziehen.

Es werden nun die folgenden, ökonomisch plausiblen, Annahmen gemacht.

(i) f ist eine stetige, in einem Intervall $[0, s] \subset [0, \min_i K_i]$ streng monoton fallende, zweimal stetig differenzierbare Funktion.

(ii) $f(0) > 0$ und $f(x) = 0$, für alle $x \geq s$.

(iii) $f''(x) \leq 0$, für alle $x \in [0, s]$.

(iv) $k_1 = \ldots = k_n = k \ : \ [0, \max_i K_i] \to \mathbb{R}^+$.

(v) k ist eine streng monoton wachsende, zweimal stetig differenzierbare Funktion mit $k''(x) \geq 0$, für alle $x \in [0, \max_i K_i]$.

Wenn die Produzenten nun die Mengen $x_1, \ldots, x_n$ ihres Produktes herstellen, macht der i–te Hersteller einen Gewinn von

$$g_i(x_1, \ldots, x_n) := x_i f\left(\sum_{j=1}^n x_j\right) - k(x_i), \quad i = 1, \ldots, n.$$

Die ökonomische Entscheidungssituation, welche Mengen $x_i \in [0, K_i]$ die einzelnen Produzenten unabhängig voneinander auf dem Markt anbieten sollen, kann man analysieren durch das n–Personen–Spiel

$$\Gamma = \langle \{1, \dots, n\};\ \{[0, K_1], \dots, [0, K_n]\};\ \{g_1, \dots, g_n\} \rangle,$$

welches auch als *Oligopolspiel* bezeichnet wird. Die Frage, wie sich die Oligopolisten rational verhalten sollen, ist dabei eine klassische Fragestellung der Ökonomie. Für den Spezialfall $n = 2$ des *Duopols* ist der Begriff des Gleichgewichtspunktes seit langem (1838) als *Cournotscher Duopolpunkt* bekannt.

Satz 7.20 *Das Oligopolspiel* $\Gamma = \langle \{1, \dots, n\};\ \{[0, K_1], \dots, [0, K_n]\};\ \{g_1, \dots, g_n\} \rangle$ *besitzt unter den Voraussetzungen (i)–(v) genau einen Gleichgewichtspunkt* $(x_1^*, \dots, x_n^*)$. *Für diesen ist*

$$x_1^* = \ldots = x_n^* = x^*,$$

wobei im Fall $k'(0) \geq f(0)$ *gilt* $x^* = 0$. *Im Fall* $k'(0) < f(0)$ *dagegen ist* x^* *die einzige Lösung der Gleichung*

$$f(nx) + xf'(nx) - k'(x) = 0,$$

die im Intervall $[0, s/n]$ *liegt.*

Beweis: Sei $\bar{x} = (x_1^*, \dots, x_n^*)$ ein Gleichgewichtspunkt und $\sigma := \sum_{i=1}^n x_i^*$. Dann gilt $\sigma \leq s$, da sonst $g_i(\bar{x}) = -k(x_i^*) < -k(0) = g_i(\bar{x} \| x_i = 0)$ (Widerspruch). Für $i = 1, \dots, n$ setze $\sigma_i := \sum_{1 \leq j \leq n, j \neq i} x_j^* = \sigma - x_i^*$ und für $x \in [0, K_i]$

$$h_i(x_i) = g_i(\bar{x} \| x_i) = x_i \cdot f(\sigma_i + x_i) - k(x_i).$$

Die Funktionen $h_i : I_i := [0, s - \sigma_i] \to \mathbb{R}$ sind zweimal stetig differenzierbar mit

$$\begin{aligned} h_i'(x_i) &= f(\sigma_i + x_i) + x_i \cdot f'(\sigma_i + x_i) - k'(x_i), \\ h_i''(x_i) &= 2f'(\sigma_i + x_i) + x_i \cdot f''(\sigma_i + x_i) - k''(x_i). \end{aligned}$$

Wegen $h_i''(x_i) < 0$ ist also h_i' im Intervall I_i streng monoton fallend von $h_i'(0) = f(\sigma_i) - k'(0)$ auf $h_i'(s - \sigma_i) = (s - \sigma_i)f'(s) - k'(s - \sigma_i) < 0$. Für $x_i \in [s - \sigma_i, k_i]$ gilt $h_i(x_i) = -k(x_i)$, also ist h_i streng monoton fallend in $[s - \sigma_i, k_i]$.

Fall 1: $k'(0) \geq f(0)$. Dann gilt $h_i'(0) \leq 0$ und damit $h_i'(x_i) < 0$, für alle $x_i \in (0, s - \sigma_i]$. Darum ist h_i streng monoton fallend in $[0, K_i]$. Also $x_i^* = 0$, für alle i. Andererseits ist $(0, \dots, 0)$ in diesem Fall ein Gleichgewichtspunkt.

Fall 2: $k'(0) < f(0)$. Dann gilt $\bar{x} \neq 0$, da sonst $h_i'(0) = f(0) - k'(0) > 0$ und damit das Maximum von h_i nicht in 0 angenommen wird. Also gilt für mindestens ein k $x_k^* > 0$. Ferner gilt wegen $h_k'(s - \sigma_i) = (s - \sigma_i)f'(s) - k'(x_i) < 0$, daß $x_k^* \in (0, s - \sigma_i)$.

Es gilt $x_i^* > 0$, für alle i: Mit $0 = h_k'(x_k^*) = f(\sigma) + x_k^* f'(\sigma) - k'(x_k^*)$ folgt $h_i'(0) = f(\sigma_i) - k'(0) \geq f(\sigma) - k'(x_k^*) = -x_k^* f'(\sigma) > 0$.

Es muß $x_i^* = x_k^*$ gelten, für alle i: Wie oben gilt $0 = h_i'(x_i^*) = f(\sigma) + x_i^* f'(\sigma) - k'(x_i^*)$ und damit $(x_i^* - x_k^*)f'(\sigma) = k'(x_i^*) - k'(x_k^*)$. Angenommen, $x_i^* > x_k^*$, dann ist die linke Seite < 0 und die rechte Seite ≥ 0, da k' monoton wachsend ist. Also $x_1^* = \ldots = x_n^* = x^*$ und $x^* = \sigma/n \leq s/n$. Mit $h_i'(x^*) = 0$ folgt $0 = f(\sigma) + x^* f'(\sigma) - k'(x^*)$. Die Funktion $\varphi : [0, s/n] \to \mathbb{R}$ mit $\varphi(x) = f(nx) + xf'(nx) - k'(x)$ ist streng monoton fallend ($\varphi'(x) = nf'(nx) + f'(nx) +$

$xf''(nx) - k''(x) < 0)$ mit $\varphi(0) = f(0) - k'(0) > 0$ und $\varphi(s/n) = f(s) + s/n\, f'(s) - k'(s/n) < 0$. Nach dem Zwischenwertsatz existiert eine eindeutige Lösung x^* von $\varphi(x) = 0$ in $[0, s/n]$.
Es bleibt noch zu zeigen, daß $(x^*, \ldots, x^*)$ ein Gleichgewichtspunkt ist. $h_i(x_i)$ nimmt ihr Maximum in $[0, s - (n-1)x^*]$ an. Wegen
$h_i'(0) = f((n-1)x^*) - k'(0) > f(nx^*) + x^* f'(nx^*) - k(x^*) = 0$ und
$h_i'(s - (n-1)x^*) = f(s) + (s - (n-1)x^*) f'(s) - k'(s - (n-1)x^*) < 0$
wird das Maximum im Innern des Intervalls angenommen und damit gilt $h_i'(x_i) = 0$. Da h_i' streng monoton fallend ist, gibt es genau einen solchen Punkt. Wegen $h_i'(x^*) = 0$ ist dieser aber x^*.

$\square$

Beispiel 7.10 Bei einem aus drei Produzenten bestehenden Oligopol mit der Nachfragefunktion

$$f(x) = \begin{cases} 124 - x, & \text{für } x \in [0, 124] \\ 0, & \text{für } x \geq 124 \end{cases}$$

und der Kostenfunktion

$$k(x) = 4x$$

erhält man die Gleichgewichtsproduktion x^* als Lösung von

$$(124 - 3x) + x(-1) - 4 = 0,$$

also $x^* = 30$. Bei dieser Gleichgewichtsproduktion erzielt jeder der drei Produzenten einen Gewinn von

$$g_i(30, 30, 30) = 30 \cdot (124 - 90) - 4 \cdot 30 = 900, \quad i = 1, 2, 3.$$

Angenommen, die drei Produzenten bilden ein Kartell und treffen die Absprache, daß jeder von ihnen nur 20 Einheiten produziert. Dann erzielt jeder von ihnen einen Gewinn von $20 \cdot 64 - 4 \cdot 20 = 1200$, also ein Drittel mehr als in dem Gleichgewichtspunkt. Allerdings ist diese Situation nicht stabil. Verdoppelt einer der Spieler seine Produktion, so erhält dieser einen Gewinn von $40 \cdot 44 - 4 \cdot 40 = 1600$, die beiden anderen jedoch nur $20 \cdot 44 - 4 \cdot 20 = 800$.