

Einführung in die Mathematische Optimierung

Rainer E. Burkard
Technische Universität Graz
Institut für Mathematik
Steyrergasse 30
A-8010 Graz, Austria
`burkard@opt.math.tu-graz.ac.at`

Inhaltsverzeichnis

1	Optimierungsmodelle	5
1.1	Einleitung	5
1.2	Optimierungsmodelle	6
1.2.1	Produktionsmodelle	6
1.2.2	Mischungsprobleme	7
1.2.3	Das Ernährungsmodell von Stigler	8
1.2.4	Transportprobleme	8
1.2.5	Flüsse in Netzwerken	9
1.2.6	Nichtlineare Optimierungsaufgaben	10
1.2.7	Ganzzahlige Optimierungsaufgaben	10
1.2.8	Vom Modell zur Lösung	11
2	Graphische Darstellung von Optimierungsproblemen in zwei Variablen	13
3	Der Hauptsatz der linearen Optimierung	19
4	Das Simplexverfahren	25
5	Die Bestimmung einer zulässigen Ausgangslösung	35
5.1	Zweiphasenmethode von Dantzig	35
5.2	Die M -Methode	38
6	Kreisen des Simplexverfahrens	41
7	Spaltenauswahlregeln	49
8	Die Behandlung von Gleichungen, beschränkten Variablen und nicht vorzeichenbeschränkten Variablen im Simplexverfahren	53
8.1	Die Behandlung von Gleichungen	53
8.2	Beschränkte Variable	55
8.3	Die Behandlung von Variablen ohne Vorzeichenbeschränkung	58
8.4	Eine Anwendung: Lineares Ausgleichsproblem	61
9	Simplexinterpretation des Simplexverfahrens	65
10	Das revidierte Simplexverfahren	69

11	Das Simplexverfahren mit LU-Zerlegung	77
12	Duale lineare Programme	85
13	Eine Anwendung der Dualitätstheorie: Der Hauptsatz der Spieltheorie.	91
14	Trennungssätze für konvexe Mengen	95
15	Alternativsätze und der Beweis des Dualitätssatzes	99
16	Das duale Simplexverfahren	105
17	Komplexität der linearen Optimierung	111
17.1	Komplexität des Simplexverfahrens	111
17.2	Ellipsoidverfahren	111
18	Innere Punkte Verfahren: Theorie	119
18.1	Grundidee zu inneren Punkteverfahren	119
18.2	Selbstduale lineare Programme	121
18.3	Endlichkeit des Inneren Punkteverfahrens	127
19	Innere Punkte Verfahren: Algorithmen	131
19.1	Ein generischer Algorithmus	131
20	Ganzzahligkeit der Lösungen linearer Programme	135
21	Transportprobleme	141
22	Maximale Flüsse in Netzwerken	157
22.1	Ein polynomialer max-Fluß Algorithmus	161
23	Rucksackprobleme	163
23.1	Das Rucksackproblem	163
23.2	Ein dynamisches Optimierungsverfahren für das binäre Rucksackproblem	165
23.3	Ein Branch-and-Bound Verfahren für das binäre Rucksackproblem	167
24	Konvexe Programme und Kuhn-Tucker Sätze	169
24.1	Konvexe Funktionen	169
24.2	Kuhn-Tucker Bedingungen	171

Kapitel 1

Optimierungsmodelle

1.1 Einleitung

Die Bestimmung des größten oder kleinsten Wertes einer Funktion zählt zu den Grundaufgaben der Mathematik. Zu deren klassischen Bestand gehören Sätze über die Existenz vom Maximum und Minimum einer stetigen Funktion definiert auf einer kompakten Menge. Mit Hilfe der Differenzialrechnung kann man Extrema differenzierbarer Funktionen berechnen, und die Theorie der Lagrange'schen Multiplikatoren gestattet es, Nebenbedingungen in Form von Gleichungen bei der Extremwertbestimmung mitzubersichtigen.

Relativ spät wurden jedoch Extremwertaufgaben mit *Ungleichungen* als Nebenbedingungen betrachtet. Die Untersuchung und Lösung von Extremwertaufgaben mit Nebenbedingungen in Form von Ungleichungen zählt zum Kernstück der *Mathematischen Optimierung*. Abgesehen von einigen sporadischen Publikationen (wie z.B. der Bestimmung kürzester spannender Bäume durch Borůvka [6]), nahm die Mathematische Optimierung ihren Ausgangspunkt in Arbeiten russischer und amerikanischer Wissenschaftler in den vierziger Jahren des 20. Jahrhunderts. Sie hat, stimuliert durch den Einsatz effizienter elektronischer Rechenanlagen, seither eine stürmische Entwicklung genommen. Moderne Optimierungsverfahren können Probleme mit vielen Tausenden von Variablen und Nebenbedingungen lösen. Zur rasanten Entwicklung dieses Gebietes hat auch beigetragen, dass mathematische Optimierungsmodelle in Bereichen unseres Lebens einsetzbar sind, die zuvor keine mathematische Behandlung erfuhren, wie etwa die Wirtschaftswissenschaften. Interessante neue theoretische Fragen befruchteten ebenfalls dieses Gebiet. So führte die Frage nach der Laufzeit des Simplexverfahrens für lineare Optimierungsprobleme zur Entdeckung von Ellipsoid- und Inneren Punkteverfahren. Während die Ellipsoidverfahren vor allem von theoretischer Bedeutung sind, haben sich Innere Punkteverfahren zu einem wichtigen Werkzeug zur Lösung von linearen Optimierungsaufgaben entwickelt.

Im Jahre 1939 erschien eine grundlegende Studie des russischen Mathematikers und späteren Nobelpreisträgers L. V. Kantorowicz [29] über mathematische Methoden bei der Organisation und Planung von Produktionsprozessen, in der Kantorowicz aufzeigt, dass sich viele Produktionsprobleme mathematisch in einheitlicher Weise formulieren lassen. Ferner zeigte Kantorowicz, dass sich die mathematischen Modelle einer numerischen Behandlung zugänglich erweisen. Unabhängig von Kantorowicz untersuchte zwei Jahre später Hitchcock [26] Transportprobleme, bei denen die Kosten zum Versand von Waren von mehreren Fabriken an mehrere Abnehmer minimiert werden sollten. Stigler [45] betrachtete 1945 ein Ernährungsmodell, in dem jene Nah-

rungsmittel gefunden werden sollten, die einerseits die notwendigen Stoffe für ein gesundes Leben gewährleisten und andererseits möglichst wenig Kosten verursachen. Eine kurze Diskussion des Stigler'schen Ernährungsmodells findet sich in Abschnitt 1.2.3. Derartige Modelle haben auch heute noch eine gewisse Bedeutung für die Tierhaltung bzw. Düngung. Aber erst 1947 gelang es G. B. Dantzig, ein wirkungsvolles Rechenverfahren für all diese Probleme anzugeben, das heute unter dem Namen *Simplexverfahren* weltweit zu den Standardverfahren der Angewandten Mathematik zählt.

Standen infolge des zweiten Weltkrieges anfangs militärische Anwendungen, und damit im Zusammenhang stehende Probleme im Vordergrund, so werden heute lineare Optimierungsaufgaben in vielen Bereichen von Industrie und Wirtschaft angewandt. Im folgenden Abschnitt wollen wir einige Optimierungsmodelle kennenlernen.

1.2 Optimierungsmodelle

1.2.1 Produktionsmodelle

Ein Betrieb kann n ($n \geq 1$) verschiedene Güter erzeugen. Der Gewinn, der beim Verkauf einer Einheit des j -ten Produktes erzielt wird, sei c_j €. Die Produktion kann nicht beliebig ausgeweitet werden, da Arbeitskräfte, Arbeitsmittel und Rohstoffe nur in beschränktem Maße zur Verfügung stehen. Wieviel soll von jedem einzelnen Produkt erzeugt werden, damit der Gesamtgewinn maximal wird?

Bezeichnen wir mit x_j die vom j -ten Produkt erzeugte Menge. Die Gewinnmaximierung führt auf die Maximierung einer Linearform, der sogenannten *Zielfunktion*

$$c_1x_1 + c_2x_2 + \dots + c_nx_n \quad (1.1)$$

unter den *Nebenbedingungen* oder *Restriktionen*

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i \quad (1 \leq i \leq m) \quad (1.2)$$

und den *Vorzeichenbedingungen*

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0. \quad (1.3)$$

Jede einzelne der m Ungleichungen beschreibt eine Kapazitätsbeschränkung. Die Vorzeichenbedingungen (1.3) besagen, daß die vom j -ten Produkt hergestellte Menge nicht negativ ist. Ein Problem, in dem eine lineare Funktion maximiert oder minimiert werden soll unter linearen Gleichungen und Ungleichungen als Nebenbedingungen, bezeichnet man als *lineare Optimierungsaufgabe* oder als *lineares Programm*.

Beispiel 1.1 Eine Firma stellt vier verschiedene Lacksorten L_1, L_2, L_3 und L_4 her. Der Gewinn pro kg beträgt 1.50 € bei L_1 , 1.– € bei L_2 , 2.– € bei L_3 und 1.40 € bei L_4 . Verfahrensbedingt können pro Tag von den Lacksorten L_1 und L_2 zusammen höchstens 1300 kg und von den Lacksorten L_1, L_3 und L_4 zusammen höchstens 2000 kg hergestellt werden. Die Mindestproduktion von L_3 soll 800 kg betragen. Von L_4 weiß man, dass pro Tag nicht mehr als 500 kg benötigt werden. Wieviel kg soll die Firma pro Tag von jeder Lacksorte herstellen, um einen möglichst großen Gewinn zu erzielen?

Ein mathematisches Modell für dieses Problem sieht folgenderweise aus. Bezeichnen wir die gesuchten Lackmengen mit x_1, x_2, x_3 und x_4 . Die Größen x_j ($j = 1, 2, 3, 4$) sollen so gewählt werden, dass der Gewinn

$$z = 1.5x_1 + 1x_2 + 2x_3 + 1.4x_4$$

möglichst groß wird unter den Nebenbedingungen

$$\begin{aligned} x_1 + x_2 &\leq 1300 \\ x_1 + x_3 + x_4 &\leq 2000 \\ x_3 &\geq 800 \\ x_4 &\leq 500 \end{aligned}$$

Da die erzeugten Lackmengen nicht negativ sein können, fordert man ferner

$$x_j \geq 0 \quad (j = 1, 2, 3, 4).$$

1.2.2 Mischungsprobleme

Soll durch Mischung aus mehreren Grundstoffen möglichst billig ein neuer Stoff hergestellt werden, so kann man die dazu benötigten Anteile der Grundstoffe über eine lineare Optimierungsaufgabe ermitteln. Dies soll an einem einfachen Beispiel verdeutlicht werden:

Eine Metallegierung soll $b\%$ Blei enthalten. Als Ausgangslegierungen stehen zur Verfügung:

Legierung	L_1	L_2	L_3	\dots	L_n
Pb-Gehalt in %	a_1	a_2	a_3	\dots	a_n
Kosten pro kg	c_1	c_2	c_3	\dots	c_n

Welche Legierungen L_j ($1 \leq j \leq n$) sollen in welchem Verhältnis gemischt werden, damit die Kosten für die neue Legierung möglichst klein werden?

Das mathematische Modell zu diesem Mischungsproblem lautet folgenderweise: Wir bezeichnen mit x_j den Anteil pro Einheit, mit dem die Legierung L_j in der gesuchten Legierung enthalten ist. Damit erhält man die zwei Restriktionen

$$\begin{aligned} a_1x_1 + a_2x_2 + \dots + a_nx_n &= b \\ x_1 + x_2 + \dots + x_n &= 1. \end{aligned}$$

Die erste dieser Nebenbedingungen gewährleistet den richtigen Bleigehalt der neuen Legierung, die zweite Restriktion besagt, dass sich alle Anteile auf 1 aufsummieren müssen. Für die Variablen x_j gelten natürlich die Vorzeichenbedingungen

$$x_j \geq 0 \quad (j = 1, 2, \dots, n).$$

Die Zielfunktion lautet

$$\text{Minimiere } z = c_1x_1 + c_2x_2 + \dots + c_nx_n.$$

1.2.3 Das Ernährungsmodell von Stigler

Stigler behandelte das Problem, die Kosten für die Ernährung eines durchschnittlichen Mannes unter Gewährleistung seines täglichen Bedarfs an Kalorien und Nährstoffen zu minimieren. Er betrachtete in seiner Studie 77 Lebensmittel und bezog ihre Kosten in US\$ auf das Jahr 1939. Für jedes Lebensmittel betrachtete er dessen Kalorien sowie seinen Gehalt an Eiweiß, Kalzium, Eisen, und an den Vitaminen A , B_1 , B_2 , C und Nikotinsäureamid (Niacin). Die folgende Nährwertliste gibt diese Werte für 10 Nahrungsmittel an, wobei die jeweilige Mengeneinheit für die Nahrungsmittel so gewählt wurde, dass sie genau 1.- \$ kostet.

Nährstoff Maßeinheit	Kalorien kcal	Eiweiß g	Ca g	Fe mg	Vit.A E	Vit.B ₁ mg	Vit.B ₂ mg	Niacin mg	Vit.C mg
Weizenmehl	44.7	1411	2.0	365	–	55.4	33.3	441	–
Maismehl	36.0	897	1.7	99	30.9	17.4	7.9	106	–
Kondensmilch	8.4	422	15.1	9	26.0	3.0	23.5	11	60
Erdsnußbutter	15.7	661	1.0	48	–	9.6	8.1	471	–
Schweinefett	41.7	–	–	–	0.2	–	0.5	5	–
Rinderleber	2.2	333	0.2	139	169.2	6.4	50.8	316	525
Kohl	2.6	125	4.0	36	7.2	9.0	4.5	26	5369
Kartoffeln	14.3	336	1.8	118	6.7	29.4	7.1	198	2522
Spinat	1.1	106	–	138	918.4	5.7	13.8	33	2755
Weißer Bohnen	29.6	1691	11.4	792	–	38.4	24.6	217	–
Durchschnittsbedarf	3000	70	0.8	12	5000	1.8	2.7	18	75

Der Durchschnittsbedarf bezieht sich auf einen 70 kg schweren, mäßig aktiven Mann. Aufgrund der oben vorgenommenen Normierung der Gewichtsmengen der einzelnen Nahrungsmittel kann man das Problem lösen, indem man die Summe der Variablen (= Anteil des jeweiligen Nahrungsmittels in der Kost) minimiert.

Durch plausibles Schließen war es Stigler möglich, eine Kost herauszufinden, die nur \$ 39.93 pro Jahr kostete, also weniger als 11 Cents pro Tag. Diese Kost bestand aus Weizenmehl, Kondensmilch, Kohl, Spinat und Weissen Bohnen. Ersetzt man die Kondensmilch durch Rinderleber, so erhält man eine Optimallösung mit Jahreskosten von \$ 36.64. Es ist interessant festzuhalten, dass in jeder Optimallösung dieses Problems nur die oben angeführten 10 Lebensmittel auftreten. Man könnte sich also schon im vornherein auf sie beschränken. Für eine genauere Diskussion des Modells sei auf Dantzig [14] verwiesen.

Ähnliche Modelle wie das oben angeführte werden heute in der Viehhaltung verwendet.

1.2.4 Transportprobleme

Das Problem, Waren von den Erzeugern auf die billigste Weise an die Abnehmer zu senden, so dass die ganze Produktion versandt wird und der Bedarf der Abnehmer gedeckt wird, bezeichnet man als *Transportproblem*. Nehmen wir dazu an, in m Fabriken werden jeweils die Warenmengen a_i ($1 \leq i \leq m$) erzeugt. Diese Waren sollen an n Abnehmer, die jeweils die Warenmenge b_j ($1 \leq j \leq n$) benötigen, so versandt werden, dass die Gesamttransportkosten minimal werden. Man kann ohne Beschränkung der Allgemeinheit annehmen, dass

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

gilt. Bezeichnen wir mit c_{ij} die Transportkosten pro Wareneinheit von Fabrik i zum Abnehmer j , so lautet das zugehörige Modell, in dem mit x_{ij} die tatsächlich von i nach j versandte Menge bezeichnet:

$$\text{Minimiere } \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

unter den Nebenbedingungen

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= a_i \quad (1 \leq i \leq m) \\ \sum_{i=1}^m x_{ij} &= b_j \quad (1 \leq j \leq n) \\ x_{ij} &\geq 0 \quad (1 \leq i \leq m, 1 \leq j \leq n). \end{aligned}$$

Knödel [34] behandelte 1960 das Problem, einen optimalen Transportplan für den Versand von Zucker aufzustellen, der in 7 Fabriken in Österreich erzeugt wurde und an rund 300 Abnehmer geliefert werden sollte. Innerhalb kürzester Zeit amortisierten sich die Kosten für die Berechnung einer Optimallösung dieses Problems.

Ist $m = n$ und $a_i = b_j = 1$ für alle $1 \leq i \leq m, 1 \leq j \leq n$, so geht das Transportproblem in ein *Zuordnungsproblem* über. Zuordnungsprobleme spielen etwa in der Personaleinsatzplanung eine Rolle.

1.2.5 Flüsse in Netzwerken

Ein Netzwerk besteht aus Knoten, die durch gerichtete Kanten miteinander verbunden sind. Jede Kante habe eine positive Kapazität. Zwei der Knoten des Netzwerkes sind ausgezeichnet, eine *Quelle* s und eine *Senke* t . Ein *Fluss* von s nach t ordnet jeder Kante eine nichtnegative Zahl, den *Flusswert* zu, wobei die folgenden beiden Restriktionen erfüllt sein müssen:

(A) *Kapazitätsrestriktionen:*

Auf jeder Kante muß der Flusswert größer/gleich 0 und kleiner/gleich der Kantenkapazität sein.

(B) *Flusserhaltungsgleichungen:*

In jedem Knoten $\neq s, t$ ist die Summe der ankommenden Flusswerte gleich der Summe der abgehenden Flusswerte.

Gesucht wird ein *maximaler Fluss*, d.h. ein Fluss, bei dem die Summe auf den von der Quelle s ausgehenden (und in die Senke t einmündenden) Kanten maximal ist. Dadurch kann etwa eine Frage der folgenden Art beantwortet werden:

Wieviele Telefongespräche lassen sich gleichzeitig von s nach t führen?

Sind zusätzlich Kosten auf den Kanten gegeben, kann man einen maximalen Fluss mit minimalen Kosten suchen. Die Transportprobleme des vorigen Abschnittes lassen sich etwa in dieser Weise modellieren.

1.2.6 Nichtlineare Optimierungsaufgaben

Die bisher erwähnten Probleme führen stets auf eine lineare Zielfunktion, die unter linearen Nebenbedingungen (Gleichungen und Ungleichungen) zu maximieren oder zu minimieren ist. In manchen Anwendungen ist es aber notwendig, von der Linearität der Funktionen abzugehen. Dies führt auf sogenannte *nichtlineare Optimierungsaufgaben*. Bei nichtlinearen Optimierungsaufgaben spielen neue Phänomene eine Rolle. So können etwa mehrere lokale Extrema auftreten. Sind allerdings alle involvierten Funktionen konvex, d.h. wird eine konvexe Zielfunktion $F(x)$ minimiert unter den Restriktionen

$$f_i(x) \leq 0 \quad (1 \leq i \leq m),$$

wobei alle Funktionen $f_i(x)$ konvex sind, dann liegt eine sogenannte *konvexe Optimierungsaufgabe* vor, von der man zeigen kann, dass jedes lokale Minimum auch eine globale Minimallösung ist.

1.2.7 Ganzzahlige Optimierungsaufgaben

In vielen Anwendungen können die Variablen nur ganzzahlige Werte annehmen. Dies ist etwa der Fall, wenn sie eine Stückanzahl oder eine logische Entscheidung beschreiben. Eine typische ganzzahlige Optimierungsaufgabe ist zum Beispiel das folgende Standortproblem.

Nichtkapazitiertes Standortproblem In einer großen Firma können m neue Produktionsstätten F_1, F_2, \dots, F_m errichtet werden, die n Kunden K_1, K_2, \dots, K_n beliefern sollen. Die Errichtungskosten für die Produktionsstätte F_i , $1 \leq i \leq m$, betragen $c_i \text{ €}$. Wird der Bedarf des Kunden K_j , $1 \leq j \leq n$, von Fabrik F_i gedeckt, so fallen die Transportkosten $d_{ij} \text{ €}$ an. Welche der Produktionsstätten sollen errichtet werden und wer beliefert welchen Kunden, so dass die gesamten Errichtungs- und Transportkosten minimal werden?

Um dieses Problem zu modellieren, führen wir *binäre* Variable y_i , $1 \leq i \leq m$, und x_{ij} , $1 \leq i \leq m$, $1 \leq j \leq n$, ein. Dabei ist $y_i = 1$, wenn die i -te Fabrik errichtet wird. Andernfalls ist $y_i = 0$. Wird der Kunde K_j von Fabrik F_i aus beliefert, so ist $x_{ij} = 1$, anderenfalls ist $x_{ij} = 0$. Die Minimierung der Gesamtkosten führt auf

$$\text{Minimiere} \quad \sum_{i=1}^m c_i y_i + \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij}$$

unter den Nebenbedingungen

$$\begin{aligned} \sum_{i=1}^m x_{ij} &= 1, & 1 \leq j \leq n, \\ y_i - x_{ij} &\geq 0, & 1 \leq i \leq m, \quad 1 \leq j \leq n, \\ y_i &\in \{0, 1\}, & 1 \leq i \leq m, \\ x_{ij} &\in \{0, 1\}, & 1 \leq i \leq m, \quad 1 \leq j \leq n. \end{aligned}$$

Die erste der Nebenbedingungen garantiert, dass jeder Kunde mindestens die benötigte Warenmenge erhält. Die zweite Bedingung garantiert, dass ein Kunde nur dann von der Produktionsstätte F_i beliefert wird, wenn diese auch errichtet wird.

Eine andere wichtige ganzzahlige Optimierungsaufgabe ist das sogenannte **Rucksackproblem**: Ein Bergsteiger möchte auf einer Wanderung höchstens einen b kg schweren Rucksack tragen. Er muss verschiedene Dinge wie Nahrung, Ausrüstung, Kleidung mitnehmen. Von jedem Gegenstand j steht sein Gewicht a_j fest, sowie der ideeller Wert c_j , den dieser Gegenstand für den Wanderer hat. Welche Gegenstände nimmt nun der Wanderer mit, dass der ideelle Wert maximal, aber das Höchstgewicht nicht überschritten wird?

Setzen wir $x_j = 1$, falls der j -te Gegenstand mitgenommen wird, und andernfalls $x_j = 0$, so lautet das zugehörige Modell:

$$\text{Maximiere } \sum_{j=1}^n c_j x_j$$

unter den Nebenbedingungen

$$\sum_{j=1}^n a_j x_j \leq b$$

$$x_j \in \{0, 1\} \quad (1 \leq j \leq n).$$

Will man eine Erbschaft in einem Wertpapierdepot anlegen, so erhält man ein analoges Problem. Zum Kauf von Wertpapieren stehen $b \text{ €}$ zur Verfügung. Das Wertpapier j habe die Rendite c_j und einen Kaufpreis von $a_j \text{ €}$ pro Stück. Die Variablen x_j geben nun die Stückzahl der zu kaufenden Wertpapiere an. Dabei kann auch eine obere Schranke für die Stückzahl des Wertpapiers j vorgegeben sein. Die Stückzahl soll so bestimmt werden, dass der Gesamtgewinn maximiert wird.

1.2.8 Vom Modell zur Lösung

Die Lösung einer Optimierungsaufgabe aus der Praxis ist ein vielschichtiger Prozess. Zunächst wird man versuchen, eine mathematische Formulierung für das vorliegende Problem zu finden. Dabei wird man zu berücksichtigen haben, dass lineare Nebenbedingungen und reelle Variable wesentlich leichter handzuhaben sind als nichtlineare Nebenbedingungen oder Variable, die nur diskrete Werte annehmen können. Die Gewinnung aussagekräftiger Daten stellt meist ebenfalls ein großes praktisches Problem dar. Dann muss ein geeignetes Lösungsverfahren ausgewählt werden. Die damit gewonnene Lösung muss dann im Licht der Daten und der Aufgabenstellung interpretiert werden. Meist schließt sich in der Praxis ein weiterer Schritt an, in dem das Modell variiert wird, um zusätzliche Gesichtspunkte zu berücksichtigen oder noch mehr Information über die Optimallösungen zu erhalten.

Kapitel 2

Graphische Darstellung von Optimierungsproblemen in zwei Variablen

An graphischen Darstellungen können fast alle wesentlichen Züge von Optimierungsaufgaben abgelesen werden. Daher können sie gut zur Erläuterung der wesentlichen Phänomene in der Optimierung herangezogen werden. Da man sich bei graphischen Darstellungen auf 2 Variablen beschränkt und Optimierungsaufgaben in der Praxis in den allermeisten Fällen sehr viele Variable haben, sind graphische Darstellungen zur *Lösung* von Optimierungsaufgaben im Allgemeinen nicht geeignet.

Gegeben sei das folgende *lineare Programm (LP)*:

Maximiere die *Zielfunktion* $c_1x_1 + c_2x_2$ unter den
Nebenbedingungen (*Restriktionen*) $a_{11}x_1 + a_{12}x_2 \leq b_1$
 $a_{21}x_1 + a_{22}x_2 \leq b_2$
 \vdots
 $a_{m1}x_1 + a_{m2}x_2 \leq b_m$
 $x_1 \geq 0, x_2 \geq 0.$

Die Menge M der Paare (x_1, x_2) , die die Restriktionen erfüllen, heißt *Menge der zulässigen Punkte*.

Zur graphischen Lösung dieses linearen Programms sind folgende Überlegungen notwendig:

- Die Menge aller Punkte (x_1, x_2) , die die Ungleichung

$$a_1x_1 + a_2x_2 \leq b \quad (|a_1| + |a_2| > 0)$$

erfüllen, bilden eine abgeschlossene Halbebene, die von der Geraden $a_1x_1 + a_2x_2 = b$ begrenzt wird.

- Die Menge aller Punkte (x_1, x_2) , die zwei Ungleichungen

$$a_{11}x_1 + a_{21}x_2 \leq b_1 \quad (|a_{11}| + |a_{12}| > 0)$$

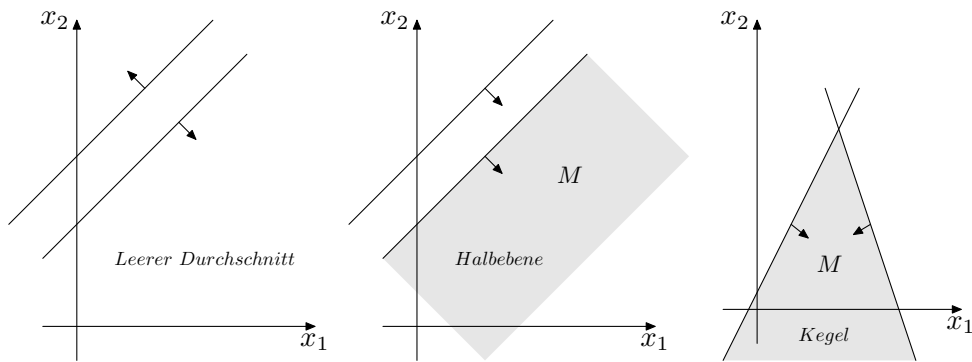


Abbildung 2.1: Mögliche Fälle für den Durchschnitt zweier Halbebenen

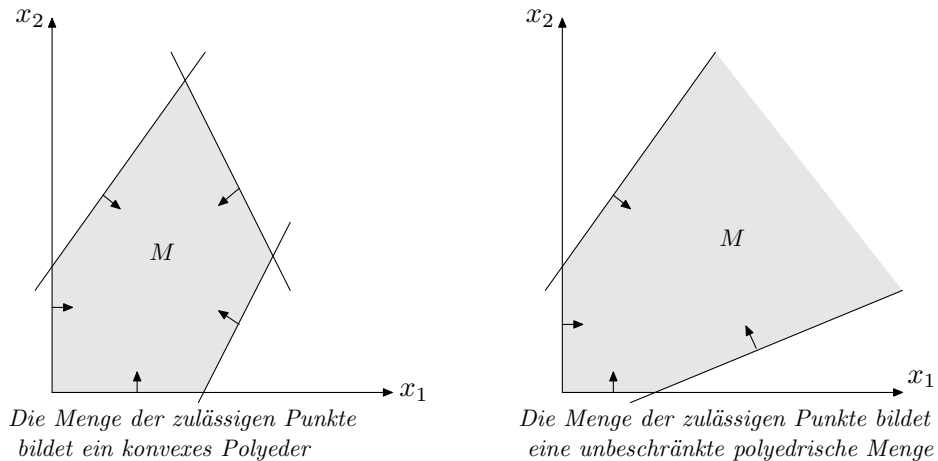


Abbildung 2.2: Menge der zulässigen Punkte einer linearen Optimierungsaufgabe in zwei Variablen. Im linken Bild bildet M ein konvexes Polyeder, während im rechten Bild die Menge der zulässigen Punkte eine unbeschränkte polyedrische Menge bildet.

$$a_{21}x_1 + a_{22}x_2 \leq b_2 \quad (|a_{21}| + |a_{22}| > 0)$$

erfüllt, entspricht dem Durchschnitt zweier Halbebenen. Dieser kann leer sein, oder er ist eine konvexe polyedrische Menge, d.h. eine konvexe¹ Menge, die durch endlich viele Geraden begrenzt wird (vgl. Abbildung 2.1).

- Die Menge aller Punkte (x_1, x_2) , die m Ungleichungen und den Vorzeichenbedingungen genügen, ist daher entweder leer oder eine konvexe polyedrische Menge. Ist diese Menge nicht leer und beschränkt, so bezeichnet man sie auch als konvexes Polyeder (vgl. Abbildung 2.2).
- Deutung der Zielfunktion: Durch

$$c_1x_1 + c_2x_2 = z \quad (|c_1| + |c_2| > 0)$$

¹Eine Menge M heißt *konvex*, wenn sie mit zwei Punkten A und B auch alle Punkte deren Verbindungsstrecke $\lambda A + (1 - \lambda)B$ ($0 \leq \lambda \leq 1$) enthält.

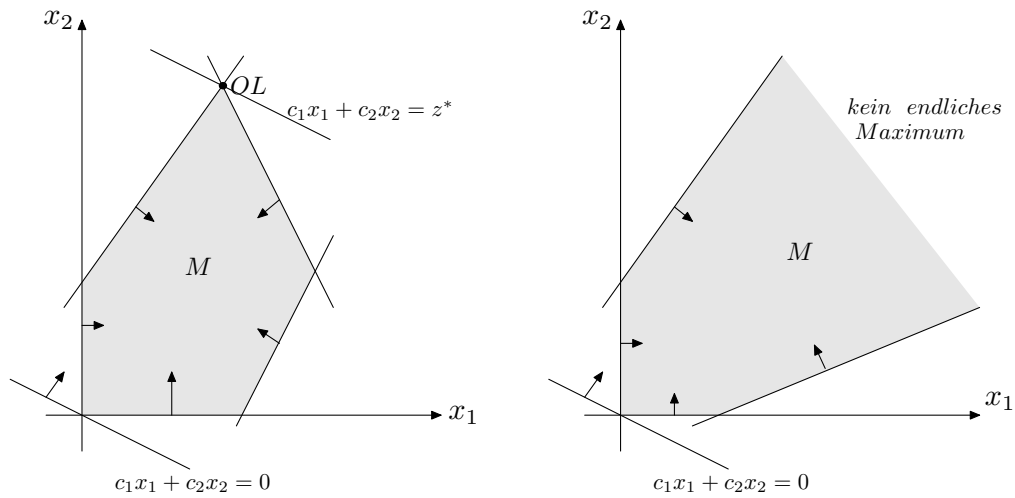


Abbildung 2.3: Graphische Lösung linearer Optimierungsaufgaben. Die Zielfunktion ist durch $z = c_1x_1 + c_2x_2$ gegeben. Während im linken Bild die Zielfunktion eine endliche Optimallösung (OL) annimmt, kann sie im rechten Bild beliebig große Werte erreichen.

wird eine Schar paralleler Geraden beschrieben. Dabei misst z (bis auf einen Normierungsfaktor) den Abstand der Geraden vom Ursprung.

Zur graphischen Lösung der gegebenen linearen Optimierungsaufgabe wird die Gerade $c_1x_1 + c_2x_2 = 0$ so lange parallel verschoben, bis z maximal/minimal wird und (x_1, x_2) noch zulässig ist. Ist die Menge M der zulässigen Punkte nicht beschränkt, so muss das LP keine endliche Optimallösung besitzen (vgl. Abbildung 2.3).

Auch nichtlineare und diskrete Optimierungsaufgaben in zwei Variablen können in analoger Weise graphisch gelöst werden. Dabei treten zum Teil neue Phänomene auf: bei konvexen Optimierungsaufgaben (vgl. Abbildung 2.4), muss das Optimum nicht mehr am Rand des zulässigen Bereichs liegen. Bei nichtkonvexen Optimierungsaufgaben (vgl. Abbildung 2.5) treten lokale Optima auf.

Bei ganzzahligen Optimierungsaufgaben sind nur Gitterpunkte in der Ebene zulässig (vgl. Abbildung 2.6). Konstruiert man deren konvexe Hülle, so können ganzzahlige Programme aufgrund des Hauptsatzes der linearen Optimierung (vgl. Abschnitt 3) auf lineare Programme zurückgeführt werden. Ein allgemeines Verfahren zur Konstruktion der konvexen Hülle von Gitterpunkten im \mathbb{R}^n ist jedoch nicht bekannt. Daher ist diese Lösungsmöglichkeit nur in der Ebene und im Raum möglich.

Da lineare, nichtlineare und ganzzahlige Optimierungsprobleme bei Lösung durch graphische Verfahren denselben Schwierigkeitsgrad haben, muss darauf hingewiesen werden, dass dies im allgemeinen nicht gilt. Mehrdimensionale nichtlineare Probleme sind nur mit einem großen numerischen Aufwand lösbar. Ebenso kann die Lösung ganzzahliger Probleme bereits in wenigen Variablen auf große Schwierigkeiten führen.

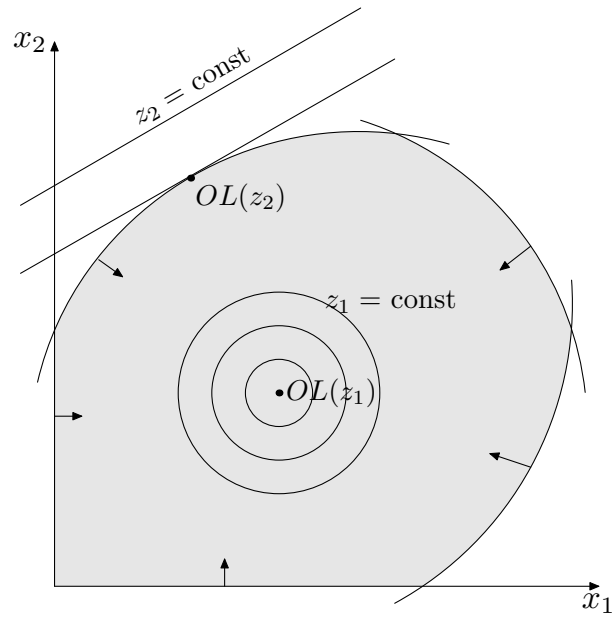


Abbildung 2.4: Konvexe Optimierungsaufgabe

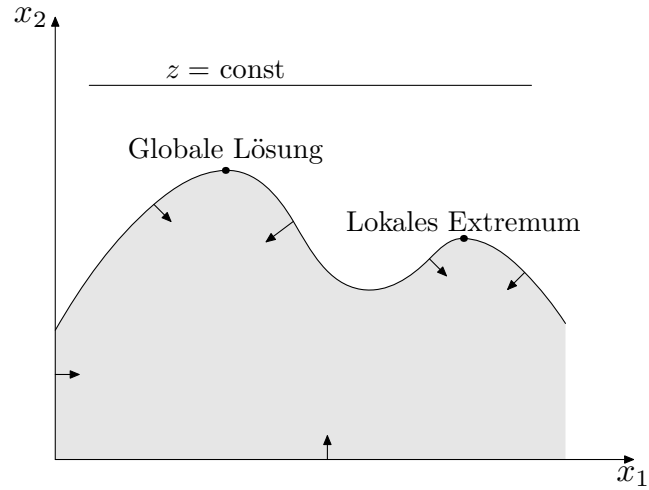


Abbildung 2.5: Nichtkonvexe Optimierungsaufgabe

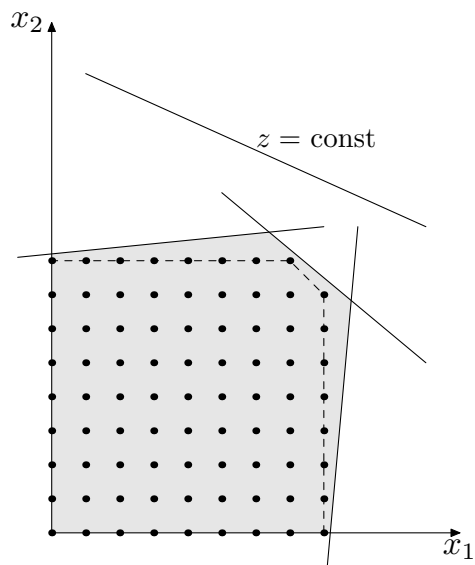


Abbildung 2.6: Lineares ganzzahliges Programm

Kapitel 3

Der Hauptsatz der linearen Optimierung

Anhand graphisch gelöster Beispiele kann man erkennen, dass unter den Optimallösungen eines linearen Programms sich stets eine *Ecke* des zulässigen polyedrischen Bereichs befindet. Wir wollen nun zeigen, dass dies ganz allgemein gilt.

Gegeben sei ein lineares Programm in folgender Form:

$$\begin{array}{ll} \text{Maximiere} & c_1x_1 + c_2x_2 + \dots + c_nx_n \\ \text{unter} & a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1 \\ & a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2 \\ (P) & \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m \\ & x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0. \end{array}$$

Zur Abkürzung fassen wir die Koeffizienten c_1, \dots, c_n zum Vektor c , die Koeffizienten a_{ij} zur Matrix

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$$

und die Größen b_i ($1 \leq i \leq m$) zum Vektor b zusammen. Ferner fassen wir auch die Variablen x_j ($1 \leq j \leq n$) zu einem Vektor x zusammen. Das Skalarprodukt der Vektoren c und x bezeichnen wir mit

$$c'x = c_1x_1 + \dots + c_nx_n.$$

Dann gilt:

Lemma 3.1 *Jedes lineare Programm (P) lässt sich in der folgenden Normalform schreiben:*

$$\begin{array}{ll} (P) \text{ Maximiere} & c'x \quad \text{unter den Restriktionen} \\ & Ax \leq b \\ & x \geq 0 \end{array}$$

Dabei gelten die Ungleichheitszeichen komponentenweise.

Man kann nun leicht zeigen, dass (P) die allgemeinste Form eines linearen Programms ist, denn:

- Die Minimierung von $c'x$ entspricht der Maximierung von $(-c)'x$.
- Eine Restriktion der Form $a'x \geq b$ wird durch Multiplikation mit -1 übergeführt in $(-a)'x \leq (-b)$.
- Eine Gleichung als Restriktion kann in zwei Ungleichungen aufgespalten werden:

$$a'x = b \Leftrightarrow \left\{ \begin{array}{l} a'x \leq b \\ a'x \geq b \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} a'x \leq b \\ (-a)'x \leq (-b) \end{array} \right\}.$$

Eine andere Möglichkeit besteht darin, mit Hilfe der Gleichung eine Variable durch die anderen auszudrücken und sie dann im ganzen Restriktionensystem zu eliminieren.

- Eine nicht vorzeichenbeschränkte Variable kann als Differenz zweier vorzeichenbeschränkter Variablen dargestellt werden:

$$x \text{ nicht vorzeichenbeschränkt} \Leftrightarrow x = \bar{x} - \bar{\bar{x}}, \bar{x} \geq 0, \bar{\bar{x}} \geq 0.$$

Durch diese Transformationen kann jedes lineare Programm auf die Gestalt (P) gebracht werden. ■

Es seien a_i ($1 \leq i \leq m+n$) die Zeilenvektoren des durch die Vorzeichenbedingungen erweiterten Restriktionensystems. Ohne Beschränkung der Allgemeinheit können wir annehmen, dass für alle i stets $a_i \neq 0$ gilt. In diesem Fall wird durch $a_i'x \leq b_i$ ein Halbraum beschrieben, der durch die Hyperebene $H_i := \{x \mid a_i'x = b_i\}$ begrenzt wird.

Im folgenden wollen wir annehmen, dass die Menge der zulässigen Punkte stets in der Form $M = \{x \mid Ax \leq b, x \geq 0\}$ gegeben ist. Somit ist M der Durchschnitt von $n+m$ Halbräumen. Dieser Durchschnitt kann leer sein oder er ist als Durchschnitt konvexer Mengen wieder eine konvexe Menge. Da M von Hyperebenen begrenzt wird, ist M eine polyedrische Menge. Damit erhält man:

Entweder ist M leer oder eine konvexe, polyedrische Menge im \mathbb{R}^n .

Sei nun $x^0 \in \mathbb{R}^n$. Wir definieren:

$$\begin{aligned} I(x^0) &:= \{i \mid a_i'x^0 = b_i\} = \{i \mid x^0 \in H_i\} \\ L(x^0) &:= \{x \mid a_i'x = b_i, i \in I(x^0)\} = \bigcap_{i \in I(x^0)} H_i \end{aligned}$$

d.h. $I(x^0)$ enthält die Indizes jener Hyperebenen, auf denen x^0 liegt. Ferner bezeichnen wir mit $|I(x)|$ die Anzahl der Elemente von $I(x)$. Schneidet man die Hyperebenen H_i , $i \in I(x^0)$, so erhält man einen affinen Unterraum $L(x^0)$ des \mathbb{R}^n . Ist $I(x^0) = \emptyset$, so ist $L(x^0)$ der ganze Raum, ist $I(x^0) = \{i\}$, so ist $L(x^0) = H_i$, ist $I(x^0) = \{i, j\}$, so ist $L(x^0) = H_i \cap H_j$. Die

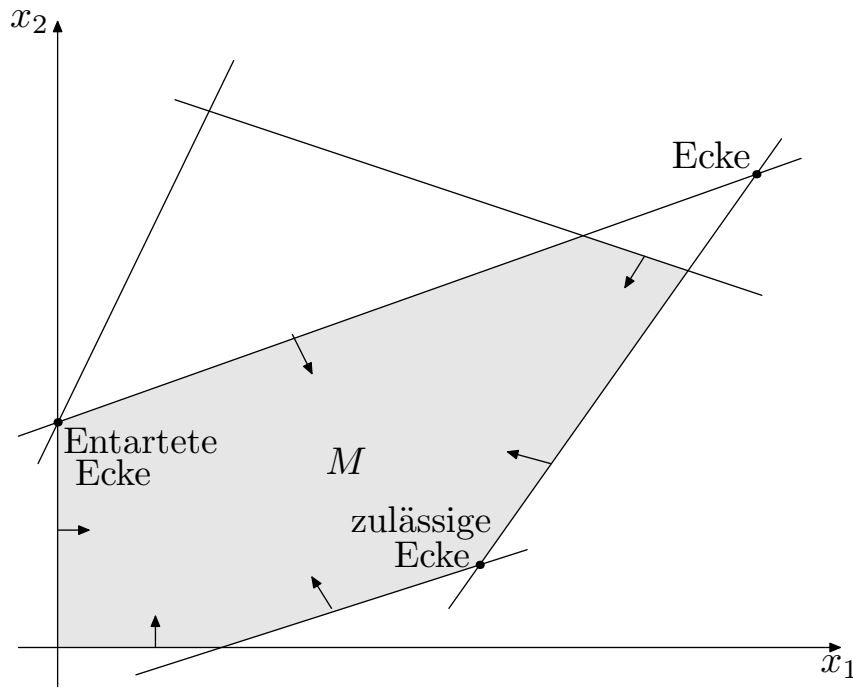


Abbildung 3.1: Zulässige, entartete und (gewöhnliche) Ecken

Dimension $\dim L(x^0)$ von $L(x^0)$ ist die Dimension des zugehörigen linearen Unterraumes. Ist also $\dim L(x^0) = 0$, so enthält $L(x^0)$ nur den Punkt x^0 .

Diese Vorbemerkungen gestatten es nun, den Begriff einer *Ecke* bezüglich eines Restriktionensystems $Ax \leq b, x \geq 0$ einzuführen.

Definition 3.1 Ein Punkt $x^0 \in \mathbb{R}^n$ heißt *Ecke* bezüglich $Ax \leq b, x \geq 0$, wenn $\dim L(x^0) = 0$ gilt. Ist $x^0 \in M$, so heißt x^0 *zulässige Ecke* oder *Ecke* von M . Liegt x^0 auf mehr als n Hyperebenen H_i , so heißt x^0 *entartete Ecke*.

Ein Punkt x^0 ist also *Ecke* bezüglich $Ax \leq b, x \geq 0$, wenn dieser Punkt mindestens n der vorgegebenen Restriktionen mit Gleichheit erfüllt – die Indizes dieser Restriktionen sind gerade in der Menge $I(x^0)$ enthalten – und sich die Hyperebenen H_i mit $i \in I(x^0)$ in einem Punkt schneiden, siehe Abbildung 3.1). Für lineare Programme mit zwei Variablen spielen entartete Ecken keine Rolle. In solch einer Ecke ist eine der Restriktionen immer überflüssig und kann weggelassen werden, ohne die Menge M der zulässigen Punkte zu ändern. Aber schon im \mathbb{R}^3 gilt dies nicht mehr. So ist etwa die Spitze einer vierseitigen Pyramide eine entartete Ecke, aber keine der beteiligten Ebenen kann weggelassen werden, ohne dass M geändert werden würde. Es gilt nun:

Satz 3.2 Ist $M \neq \emptyset$, so gibt es mindestens eine *zulässige Ecke* von M .

Beweis: Es sei $x^0 \in M$. Falls $\dim L(x^0) = 0$ ist, so ist x^0 eine *Ecke*. Andernfalls lege man in $L(x^0)$ eine Gerade durch x^0 und schneide diese Gerade mit den Hyperebenen $H_i, i \notin I(x^0)$. Wegen $x \geq 0$ gibt es mindestens einen Schnittpunkt. Aufgrund der Konvexität von M ist

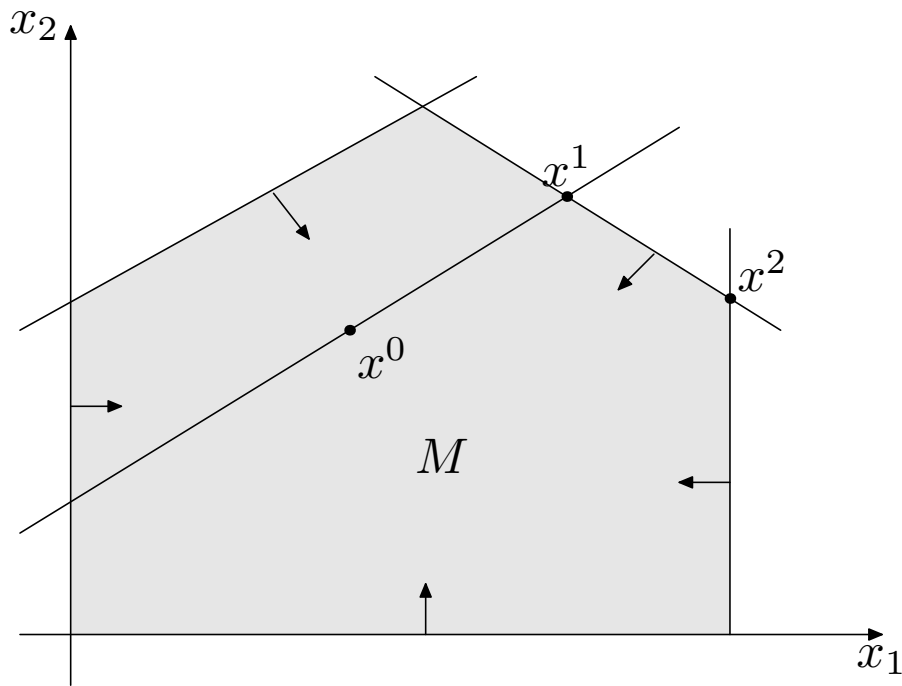


Abbildung 3.2: Zum Beweis von Satz 3.2

der Schnittpunkt ist auch zulässig, da x^0 zulässig war. Dieser zulässige Schnittpunkt sei x^1 . Es gilt nun $|I(x^1)| > |I(x^0)|$ und daher $\dim L(x^1) < \dim L(x^0)$. Ist $\dim L(x^1) = 0$, so ist x^1 zulässige Ecke, andernfalls wiederhole man das obige Verfahren. Auf diese Weise erhält man nach maximal n Schritten eine zulässige Ecke von M . ■

Abbildung 3.2 zeigt die Beweisidee an einem Beispiel in der Ebene.

Wäre M nicht in der Form $M = \{x \mid Ax \leq b, x \geq 0\}$ gegeben, so muß M nicht notwendigerweise eine Ecke enthalten. Ist M etwa eine Halbebene, so hätte die Menge der zulässigen Punkte keine Ecke. Erst durch Einführung der Vorzeichenbedingungen wird die Existenz einer Ecke gewährleistet. Nach Lemma 3.1 kann man aber stets erreichen, dass die Vorzeichenbedingungen gelten.

Satz 3.3 *Ist $x \in M$ keine Ecke von M , dann ist x innerer Punkt einer Strecke $[a, b]$, die ganz in M liegt.*

Beweis: Falls x keine Ecke ist, ist die Dimension von $L(x) \geq 1$. Man lege eine Gerade in $L(x)$ durch x und schneide die Gerade mit den Hyperebenen H_i , $i \notin I(x)$. Nach Definition von $I(x)$ fallen die Schnittpunkte nicht in einem Punkt x zusammen. Auf diese Weise erhält man aufgrund der Konvexität von M eine Strecke in M , die x als inneren Punkt enthält. Falls M unbeschränkt ist, ist es auch möglich, dass es nur einen Schnittpunkt gibt. Der Punkt x liegt dann im Inneren eines Halbstrahls. ■

Satz 3.4

- Eine affin-lineare Funktion $c_0 + c'x$ nimmt das Minimum und das Maximum auf einer Strecke in deren Endpunkten an.
- Auf einem Halbstrahl nimmt eine affin-lineare Funktion entweder ihr Minimum oder ihr Maximum an. Dies liegt dann in dem einen Endpunkt des Halbstrahls.

Beweis: In Parametergestalt lässt sich eine Strecke darstellen durch

$$a^0 + \lambda a^1 \text{ mit } \lambda_0 \leq \lambda \leq \lambda_1; a^0, a^1 \in \mathbb{R}^n.$$

Auf dieser Strecke nimmt $f(x) = c_0 + c'x$ folgende Werte an:

$$c_0 + c'(a^0 + \lambda a^1) = c_0 + c'a^0 + \lambda c'a^1 = k_0 + \lambda k_1 \text{ mit } k_0, k_1 \in \mathbb{R}.$$

Ist $k_1 = 0$, so ist $f(x)$ konstant auf der Strecke. Ist $k_1 > 0$, so wird das Maximum von $f(x)$ für $\lambda = \lambda_1$ angenommen, bei $k_1 < 0$ für $\lambda = \lambda_0$. Auf einem Halbstrahl kann $k_0 + \lambda k_1$, $k_1 > 0$ beliebig groß werden. ■

Satz 3.5 (*Hauptsatz der linearen Optimierung*)

Nimmt eine affin-lineare Funktion $f(x)$ auf M das Maximum oder Minimum an, so auch in einer zulässigen Ecke.

Beweis: Es sei $\bar{x} \in M$ ein Punkt, in dem $f(x)$ maximal ist. Nach Satz 3.3 ist \bar{x} entweder eine Ecke, oder es gibt ein abgeschlossenes Intervall $[a, b]$ in M , das \bar{x} in seinem Inneren enthält. Nach Satz 3.4 folgt o.B.d.A. $f(a) \geq f(\bar{x})$, und da $a \in M$ folgt $f(a) = f(\bar{x})$. Da die Strecke $[a, b]$ in $M \cap L(\bar{x})$ beliebig gewählt werden kann, ist $f(x)$ konstant auf $L(\bar{x})$. Nach Satz 3.2 enthält $L(\bar{x})$ mindestens eine zulässige Ecke von M , die dann ebenfalls optimal ist. ■

Aufgrund von Satz 3.5 braucht man nicht alle $x \in M$ auf Optimalität hin zu überprüfen, sondern man kann sich auf die Ecken von M beschränken. Und davon gibt es nur endlich viele, wie der folgende Satz zeigt:

Satz 3.6 Ist $M \neq \emptyset$, so hat M endlich viele Ecken.

Beweis: Jede Ecke ist Schnitt von mindestens n Hyperebenen. Aus den gegebenen $n + m$ Hyperebenen, die den Restriktionen entsprechen, lassen sich aber nur endlich viele verschiedene n -Tupel von Hyperebenen auswählen. ■

Sind zunächst alle zulässigen Punkte der Menge M Anwärter auf die Optimallösung, so zeigen die Sätze 3.5 und 3.6, dass man sich bei der Suche nach der Optimallösung auf eine *endliche* Kandidatenmenge beschränken kann. Die Struktur der Menge M wird später im Zusammenhang mit Dekompositionsverfahren noch genauer untersucht. Die Überlegungen dieses Abschnitts sind die Grundlage für das Simplexverfahren, das im nächsten Abschnitt dargestellt wird.

Kapitel 4

Das Simplexverfahren

Das von Dantzig entwickelte *Simplexverfahren* zur Lösung linearer Programme lässt sich geometrisch folgenderweise beschreiben.

Gegeben sei ein lineares Programm in der Form $(P) \max\{c'x \mid Ax \leq b, x \geq 0\}$. Man führe folgende Schritte durch:

1. Man beginne in einer zulässigen Ecke.
2. Gibt es, ausgehend von dieser Ecke, eine Kante, entlang der die Zielfunktion wächst? Ist dies nicht der Fall, dann ist die Ecke optimal. Stop.
Andernfalls gehe zu Schritt 3.
3. Man wähle eine Kante, entlang der die Zielfunktion zunimmt.
4. Ist diese Kante ein Halbstrahl in M , so gibt es keine endliche Lösung. Stop.
Andernfalls gehe man zu Schritt 5.
5. Man bestimme die nächstliegende Ecke auf dieser Kante und gehe zu Schritt 2.

Die Bestimmung einer zulässigen Ausgangsecke erfordert i.a. die Lösung eines Hilfsproblems. Dabei zeigt sich eventuell, dass $M = \emptyset$ ist (vgl. Kapitel 5). Durch einen Optimalitätssatz muss sichergestellt werden, dass bei Schritt 2 mit einer Optimallösung abgebrochen wird. Bei der Implementierung von Schritt 3 treten im Falle entarteter Ecken Schwierigkeiten auf. In diesem Falle ist die rechnerische Version des Simplexverfahrens i.a. nicht endlich. Ein Kreisen des Simplexalgorithmus trat bisher aber nur bei konstruierten Beispielen auf. In der Praxis ist das Simplexverfahren schon infolge von Rundungsfehlern endlich. Aber auch theoretisch kann die Endlichkeit durch einfache Zusatzregeln erzwungen werden (vgl. Kapitel 6).

Bevor wir die geometrische Version des Simplexverfahrens in ein Rechenverfahren umsetzen, wollen wir die Ungleichungen in den Restriktionen durch Einführen von *Schlupfvariablen* in Gleichungen überführen und untersuchen, was rechnerisch einer (zulässigen) Ecke entspricht.

Zu jeder Ungleichung $a_i'x \leq b_i$ wird eine *Schlupfvariable* x_{n+i} eingeführt. Dann ist $a_i'x \leq b_i$ äquivalent mit $a_i'x + x_{n+i} = b_i$, $x_{n+i} \geq 0$. Liegt x auf der zur Ungleichung gehörenden Ebene, so ist $x_{n+i} = 0$. Durch die Einführung von Schlupfvariablen geht das Restriktionensystem über

in:

$$\begin{array}{rcccccccc}
 a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & + & x_{n+1} & & = & b_1 \\
 a_{12}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & + & & x_{n+2} & = & b_2 \\
 & & & & \vdots & & & & & & & \\
 a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & + & & x_{n+m} & = & b_m \\
 x_1 \geq 0 & , & x_2 \geq 0 & , & \dots & , & x_n \geq 0 & \dots & & & & x_{n+m} \geq 0.
 \end{array} \tag{4.1}$$

oder kurz in Matrixschreibweise:

$$(A|E)x = b, \quad x \geq 0$$

Dabei ist $(A|E)$ die Matrix A , an die eine $(m \times m)$ Einheitsmatrix E rechts angefügt wird.

Grundlegend für das Weitere ist der Begriff der Basis:

Definition 4.1 Eine Teilmenge B von m Elementen der Indexmenge $\{1, 2, \dots, m+n\}$ heißt Basis von (P) , wenn die zu Indizes $j \in B$ gehörenden Spalten a_j der Matrix $(A|E)$ linear unabhängig sind.

Ist B eine Basis, so setzt man

$$N := \{j \mid j \notin B, 1 \leq j \leq n+m\}.$$

Variablen x_j mit $j \in B$ heißen *Basisvariablen* (BV), während die Variablen x_j mit $j \in N$ *Nichtbasisvariablen* (NBV) genannt werden. Legt man die Reihenfolge der Indizes in B und N fest, so lassen sich die Basisvariablen zum Vektor x_B und die Nichtbasisvariablen zum Vektor x_N zusammenfassen. Analog fasst man auch die Komponenten des Zielfunktionsvektors c und der Matrix A zu c_B, c_N bzw. A_B, A_N zusammen.

Da nach Definition die m zu x_B gehörenden Spalten der Matrix A linear unabhängig sind, lässt sich das Gleichungssystem $A_B x_B = b$ eindeutig auflösen.

Definition 4.2 Es sei B eine Basis von (P) . Ein $x = (x_B, x_N) \in \mathbb{R}^{n+m}$ mit $A_B x_B = b$ und $x_N = 0$ nennt man eine *Basislösung* von (P) . Die *Basislösung* heißt *zulässig*, wenn $x_B \geq 0$ ist.

Den Zusammenhang zwischen Basislösungen und Ecken von M beschreibt der folgende Satz.

Satz 4.1 Jeder Basis entspricht eine Ecke von M . Ist die Ecke nicht entartet, so ist die zugehörige Basis eindeutig bestimmt. Einer zulässigen Basislösung entspricht eine zulässige Ecke.

Beweis: Da für die Basislösung (x_B, x_N) nach Definition $x_N = 0$ gilt, liegt dieser Punkt im Schnitt der Hyperebenen $H_j, j \in N$. Dieser Schnittpunkt ist eindeutig bestimmt, da das lineare Gleichungssystem $A_B x_B = b$ nur eine Lösung besitzt. Daher ist dieser Schnittpunkt eine Ecke von M . Im Falle einer nichtentarteten Ecke schneiden sich genau n Hyperebenen in dieser Ecke, und daher ist die Menge N eindeutig festgelegt. Die Aussage über die Zulässigkeit ergibt sich direkt aus den Definitionen. ■

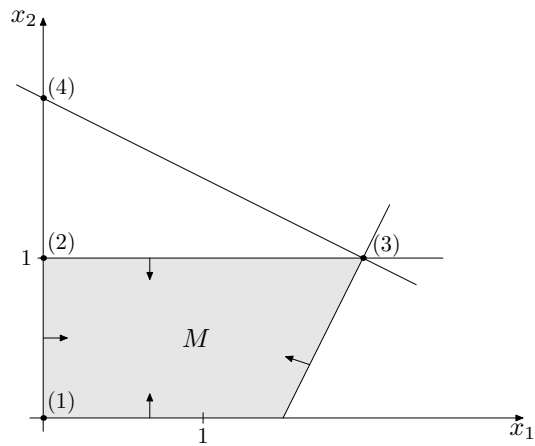


Abbildung 4.1: Graphische Darstellung des Ungleichungssystems des Beispiels 4.1

Beispiel 4.1 Man betrachte das Ungleichungssystem

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ 2x_1 - x_2 &\leq 3 \\ x_2 &\leq 1 \\ x_1 &\geq 0, x_2 \geq 0 \end{aligned}$$

Durch Einführen von Schlupfvariablen erhält man:

$$\begin{aligned} x_1 + 2x_2 + x_3 &= 4 \\ 2x_1 - x_2 + x_4 &= 3 \\ x_2 + x_5 &= 1 \\ x_j &\geq 0 \quad (1 \leq j \leq 5) \end{aligned}$$

Basislösungen sind dann zum Beispiel:

1. Nichtbasisvariable $x_1 = x_2 = 0$
 Basisvariable $x_3 = 4, x_4 = 3, x_5 = 1$
 Diese Lösung entspricht dem Ursprung und ist zulässig.
2. Nichtbasisvariable $x_1 = x_5 = 0$. Dann ist

$$\begin{pmatrix} 2 & 1 & 0 \\ -1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 4 \\ 3 \\ 1 \end{pmatrix}$$

eindeutig lösbar, und ergibt wieder ein $x_B \geq 0$.

3. Die Ecke $x_1 = 2, x_2 = 1$ ist entartete Ecke, der drei verschiedene Basislösungen entsprechen, nämlich:

$$\begin{aligned} B &= \{1, 2, 3\} \quad , \quad N = \{4, 5\} \\ B &= \{1, 2, 4\} \quad , \quad N = \{3, 5\} \\ B &= \{1, 2, 5\} \quad , \quad N = \{3, 4\}. \end{aligned}$$

4. Die Basis $\{2, 4, 5\}$ führt auf eine nichtzulässige Ecke.

Wir fassen im Folgenden die Elemente einer Basis (Indizes der Basisvariablen) zu einem Vektor $B = (B(1), \dots, B(m))$ und die Indizes der Nichtbasisvariablen zum Vektor $N = (N(1), \dots, N(n))$ zusammen. Liegt das lineare Programm in der Form (4.1) mit $b \geq 0$ vor, so ist durch $N = (1, 2, \dots, n)$ und $B = (n + 1, \dots, n + m)$ eine zulässige Basislösung $x_N = 0, x_B = b$ gegeben. (Die Schreibweise $x_B = b$ besagt, dass $x_{B(i)} = b_i$ für $1 \leq i \leq m$). Der Einfachheit halber bezeichnen wir im Folgenden die um eine Einheitsmatrix erweiterte Koeffizientenmatrix ebenfalls mit A .

Im Simplexverfahren löst man zunächst das Gleichungssystem $Ax = b$ nach den Basisvariablen auf. Dazu partitioniert man die Matrix A in $(A_B : A_N)$ und erhält $A_B x_B + A_N x_N = b$. Da A_B regulär ist, gilt:

$$x_B = A_B^{-1}b - A_B^{-1}A_N x_N = \tilde{b} - \tilde{A}_N x_N. \quad (4.2)$$

Der Darstellung (4.2) entnimmt man, dass die Basisvariablen x_B affin-linear von den Nichtbasisvariablen x_N abhängen. x_B bezeichnet man daher auch als *abhängige Variable* und x_N als *unabhängige Variable*. Nun sind die Basisvariablen aus der Zielfunktion zu eliminieren. Man erhält:

$$\begin{aligned} c'x &= c_B'x_B + c_N'x_N = c_B'A_B^{-1}b - c_B'A_B^{-1}A_N x_N + c_N'x_N = \\ &= c_B'A_B^{-1}b + (c_N' - c_B'A_B^{-1}A_N)x_N = \tilde{c}_0 + \tilde{c}_N'x_N. \end{aligned}$$

Die Koeffizienten $\tilde{c}_{N(j)}$ ($1 \leq j \leq n$), heißen *reduzierte Kostenkoeffizienten*. Ist x zulässig, so ist insbesondere $x_N \geq 0$. Sind also alle $\tilde{c}_{N(j)} \leq 0$ ($1 \leq j \leq n$), so kann durch Vergrößerung von $x_{N(j)}$ ($1 \leq j \leq n$) keine Verbesserung des Zielfunktionswertes erreicht werden. Daher ist die augenblickliche Basislösung optimal. Man erhält also das *Optimalitätskriterium*

$$\tilde{c}_N \leq 0.$$

Ist ein $\tilde{c}_{N(j)} > 0$ ($1 \leq j \leq n$), so kann durch Vergrößerung des Wertes von $x_{N(j)}$ möglicherweise der Zielfunktionswert vergrößert werden. Man wähle ein s mit $\tilde{c}_s > 0$ und setze in (4.2)

$$x_{N(j)} = 0, \quad 1 \leq j \leq n; \quad N(j) \neq s$$

und

$$x_B = \tilde{b} - \tilde{a}_s x_s. \quad (4.3)$$

Durch die Bedingung $x_B \geq 0$ und (4.3) wird die Wahl des Wertes von x_s eingeschränkt. Ist $\tilde{a}_s < 0$, so kann x_s beliebig groß gewählt werden. Daher existiert in diesem Fall keine endliche Optimallösung. Ist jedoch ein Koeffizient $\tilde{a}_{is} > 0$, so folgt aus $x_{B(i)} = \tilde{b}_i - \tilde{a}_{is}x_s$ mit $x_s \geq 0$, dass $x_s \leq \tilde{b}_i/\tilde{a}_{is}$ sein muss. Daher bestimmt man

$$\frac{\tilde{b}_r}{\tilde{a}_{rs}} := \min \left\{ \frac{\tilde{b}_i}{\tilde{a}_{is}} \mid \tilde{a}_{is} > 0 \right\}$$

Der Vektor \tilde{b} wird in gleicher Weise wie eine Spalte von \tilde{A} transformiert:

$$\begin{aligned}\bar{b}_r &:= \frac{\tilde{b}_r}{\tilde{a}_{rs}} \\ \bar{b}_i &:= \tilde{b}_i - \frac{\tilde{a}_{is}\tilde{b}_r}{\tilde{a}_{rs}} \quad i = 1, \dots, m, \quad i \neq r.\end{aligned}$$

Die Zielfunktion, ausgedrückt durch die augenblicklichen Nichtbasisvariablen, hat die Form

$$\tilde{c}_0 + \sum_{j=1}^n \tilde{c}_j x_{N(j)}.$$

Dabei ist \tilde{c}_0 der augenblickliche Zielfunktionswert. Schreibt man die Zielfunktion in der Form

$$\sum_{j=1}^n \tilde{c}_j x_{N(j)} + z = -\tilde{c}_0, \quad (4.4)$$

so kann die Zeile (4.4) direkt als 0-te Zeile zum Restriktionensystem $\tilde{A}x = \tilde{b}$ hinzugefügt werden. Somit werden die Zielfunktionskoeffizienten wie eine Zeile der Matrix \tilde{A} transformiert:

$$\begin{aligned}\bar{c}_j &:= \tilde{c}_j - \frac{\tilde{c}_s \tilde{a}_{rj}}{\tilde{a}_{rs}}, \quad 1 \leq j \leq n + m, \\ -\bar{c}_0 &:= -\tilde{c}_0 - \frac{\tilde{c}_s \tilde{b}_r}{\tilde{a}_{rs}}.\end{aligned}$$

Dabei entspricht \bar{c}_0 dem Zielfunktionswert der augenblicklichen Basislösung. Diese neue Matrix enthält dann wieder eine vollständige Einheitsmatrix. Da diese nicht immer mitgespeichert zu werden braucht, führt man das *Simplextableau* ein, bei dem man sich nur merken muss, welche Zeile zu welcher Basisvariablen und welche Spalte zu welcher Nichtbasisvariablen gehört. Das Simplextableau hat die Form

		$N(1)$	$N(2)$	\dots	$N(n)$
	a_{00}	a_{01}	a_{02}	\dots	a_{0n}
$B(1)$	a_{10}	a_{11}	a_{12}	\dots	a_{1n}
$B(2)$	a_{20}	a_{21}	a_{22}	\dots	a_{2n}
\vdots	\vdots				
$B(m)$	a_{m0}	a_{m1}	a_{m2}	\dots	a_{mn}

mit

$$\begin{aligned}a_{00} &:= -c_0, \\ a_{i0} &:= b_i, \quad 1 \leq i \leq m, \\ a_{0j} &:= c_{N(j)}, \quad 1 \leq j \leq n.\end{aligned}$$

Bezeichnen wir mit a_{ij} die Daten eines Tableaus, das zur Basis B gehört, und mit \bar{a}_{ij} die Daten des Tableaus, das zur Basis \bar{B} gehört, bei der die r -te Basisvariable von B gegen die s -te

Nichtbasisvariable von N ausgetauscht wurde, so lautet die durchzuführende Pivotoperation

$$\bar{a}_{rs} := \frac{1}{a_{rs}},$$

$$\bar{a}_{rj} := \frac{a_{rj}}{a_{rs}}, \quad j = 0, 1, \dots, n; \quad j \neq s,$$

$$\bar{a}_{is} := -\frac{a_{is}}{a_{rs}}, \quad i = 0, 1, \dots, m; \quad i \neq r,$$

$$\bar{a}_{ij} := a_{ij} - \frac{a_{is}a_{rj}}{a_{rs}}, \quad i = 0, 1, \dots, m; \quad i \neq r, \quad j = 0, 1, \dots, n; \quad j \neq s.$$

Bevor wir nun den Algorithmus angeben, müssen wir noch festlegen, welche Spalte Pivotspalte wird. Eine häufig gebrauchte Regel lautet hierfür

$$a_{0s} = \max_{1 \leq j \leq n} a_{0j}.$$

Gegeben sei nun eine lineare Optimierungsaufgabe im Ausgangstableau. Dieses ist *zulässig*, wenn $a_{i0} \geq 0$ für $1 \leq i \leq m$. Es ist *optimal*, wenn $a_{0j} \leq 0$ für $1 \leq j \leq n$ gilt. $-a_{00}$ ist der Wert der Zielfunktion der augenblicklichen Basislösung

$$\begin{aligned} x_{N(j)} &:= 0 & 1 \leq j \leq n, \\ x_{B(i)} &:= a_{i0} & 1 \leq i \leq m. \end{aligned}$$

Algorithmus 1 Simplexverfahren zur Bestimmung einer Optimallösung von $\max\{c'x \mid Ax \leq b, x \geq 0\}$ bei zulässiger Ausgangslösung.

Anfangsdaten: Anfangstableau (a_{ij}) für $0 \leq i \leq m$, $0 \leq j \leq n$.
 $B = (n+1, \dots, n+m)$, $N = (1, \dots, n)$.

1. Ist ein $a_{0j} > 0$ ($1 \leq j \leq n$), so gehe man zu 2., andernfalls ist eine Optimallösung erreicht.

Setze

$$\begin{aligned} x_{B(i)} &:= a_{i0}, & 1 \leq i \leq m, \\ x_{N(j)} &:= 0, & 1 \leq j \leq n, \\ z &:= -a_{00}. \end{aligned}$$

Stop.

2. Bestimmung der Austauschspalte: Man wähle Index s so, dass

$$a_{0s} = \max_{1 \leq j \leq n} a_{0j}.$$

3. Sind alle $a_{is} \leq 0$ ($1 \leq i \leq m$), so existiert keine endliche Lösung.

Terminiere.

Gibt es ein $a_{is} > 0$ ($1 \leq i \leq m$), so gehe man zu 4.

4. Bestimmung der Austauschzeile: Man wähle Index r so, dass

$$\frac{a_{r0}}{a_{rs}} = \min \left\{ \frac{a_{i0}}{a_{is}} \mid a_{is} > 0, i = 1, \dots, m \right\}.$$

Gehe zu 5.

5. Führe eine Pivotoperation durch: Vertausche die r -te Komponente von B mit der s -ten Komponente von N , und setze

$$\bar{a}_{rs} := \frac{1}{a_{rs}},$$

$$\bar{a}_{rj} := \frac{a_{rj}}{a_{rs}}, \quad j = 0, 1, \dots, n; \quad j \neq s,$$

$$\bar{a}_{is} := -\frac{a_{is}}{a_{rs}}, \quad i = 0, 1, \dots, m; \quad i \neq r,$$

$$\bar{a}_{ij} := a_{ij} - \frac{a_{is}a_{rj}}{a_{rs}}, \quad i = 0, 1, \dots, m; \quad i \neq r, \quad j = 0, 1, \dots, n; \quad j \neq s.$$

Nun ersetze $a_{ij} := \bar{a}_{ij}$ $0 \leq i \leq m$, $0 \leq j \leq n$ und gehe zu 1.

Beispiel 4.2 Maximiere $z = x_1 + x_2$ unter den Nebenbedingungen

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ 2x_1 - x_2 &\leq 3 \\ x_2 &\leq 1 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

(Die Menge der zulässigen Punkte dieser Optimierungsaufgabe ist in Abbildung 4.1 dargestellt.)
Ausgangstableau:

	x_1	x_2	
	0	1	
x_3	4	1	$N = (1, 2)$
x_4	3	2	$B = (3, 4, 5)$
x_5	1	0	

Dieses Tableau ist zulässig, jedoch nicht optimal. Wir wählen $s = 1, r = 2$, und führen eine Pivotoperation durch. Danach erhält man:

$-\frac{3}{2}$	$-\frac{1}{2}$	$\frac{3}{2}$	
$\frac{5}{2}$	$-\frac{1}{2}$	$\frac{5}{2}$	
$\frac{3}{2}$	$\frac{1}{2}$	$-\frac{1}{2}$	$N = (4, 2)$
1	0	1	$B = (3, 1, 5)$

(Diese Lösung entspricht der Ecke $x_1 = 3/2$, $x_2 = 0$). Wir gehen zurück zum Schritt 1. Die Lösung ist noch nicht optimal, daher setzen wir $s = 2$, $r = 3$ und führen damit erneut eine Pivotoperation durch. Man erhält:

-3	$-\frac{1}{2}$	$-\frac{3}{2}$	
0	$-\frac{1}{2}$	$-\frac{5}{2}$	
2	$\frac{1}{2}$	$\frac{1}{2}$	$N = (4, 5)$
1	0	1	$B = (3, 1, 2)$

Damit ist die Optimallösung erreicht. Sie lautet

$$x_1 = 2, x_2 = 1, z = 3$$

Die Optimallösung ist entartet, da eine Basisvariable zu 0 wird.

Sind alle Ecken des zulässigen Bereiches einer linearen Optimierungsaufgabe nichtentartet, so sind in allen Simplextableaus die Werte \tilde{b}_i , $1 \leq i \leq m$, stets positiv. Daher nimmt der Zielfunktionswert in jeder Simplexiteration zu, denn aus

$$-\bar{c}_0 = -\tilde{c}_0 - \frac{\tilde{b}_r \cdot \tilde{c}_s}{\tilde{a}_{rs}}$$

folgt, da die Größen \tilde{b}_r , \tilde{c}_s , \tilde{a}_{rs} positiv sind, dass $\bar{c}_0 > \tilde{c}_0$. Da die Zielfunktion in jeder zulässigen Basislösung einen eindeutigen Wert hat, kann somit keine Basislösung zweimal auftreten. Es gibt nur endlich viele verschiedene Basislösungen. Somit ist das Simplexverfahren im Falle nichtentarteter Basislösungen endlich.

Satz 4.2 *Sind alle Basislösungen einer linearen Optimierungsaufgabe nicht entartet, so wird diese Aufgabe durch das Simplexverfahren in endlich vielen Schritten gelöst.*

Zum vollständigen Beweis dieses Satzes müsste noch gezeigt werden, dass in endlich vielen Schritten eine zulässige Ausgangslösung gefunden werden kann. Dies ergibt sich aber unmittelbar aus den Überlegungen des folgenden Kapitels.

Bei entarteten Basislösungen kann jedoch das Simplexverfahren ins Kreisen geraten. Dies lässt sich vermeiden, wenn man die Austauschzeile durch ein lexikographisches Minimum bestimmt. Auf den Fall des Kreisens im Simplexalgorithmus wird noch im Abschnitt 6 eingegangen werden. Eine weitere Diskussion über das Laufzeitverhalten des Simplexverfahrens findet sich in Kapitel 17.

Kapitel 5

Die Bestimmung einer zulässigen Ausgangslösung

Algorithmus 1 ist nur anwendbar, wenn eine zulässige Ausgangslösung vorliegt, d.h. wenn im System $Ax \leq b$ die Beziehung $b \geq 0$ gilt. Ist ein Koeffizient $b_i < 0$, so muß eine Ausgangslösung durch einen Hilfsalgorithmus erst bestimmt werden. Dazu stehen mehrere Verfahren zur Verfügung. In der *Zweiphasenmethode* von Dantzig wird eine Hilfszielfunktion, die die augenblickliche Unzulässigkeit modelliert, optimiert. In der *M-Methode* wird eine zusätzliche Variable eingeführt. Für das so erweiterte Problem ist dann eine zulässige Lösung bekannt. Der neu eingeführten Variablen wird in der Zielfunktion (die maximiert werden soll) ein kleines Gewicht $-M$ zugewiesen. Ist das Optimierungsproblem zulässig, so wird dadurch nach einigen Schritten erreicht, dass die zusätzlich eingeführte Variable 0 wird. Damit liegt dann eine zulässige Lösung für das Ausgangsproblem vor.

5.1 Zweiphasenmethode von Dantzig

Gegeben sei das lineare Programm

$$\max\{c'x \mid Ax \leq b, x \geq 0\}$$

Durch Schlupfvariable $y \geq 0$ führen wir dieses System in ein Gleichungssystem über, das nun so normiert wird, dass alle rechten Seiten nichtnegativ sind. Da man o.B.d.A. annehmen kann, dass $b_i \leq 0$ für $1 \leq i \leq k$ galt, werden also die ersten k Zeilen dieses Systems mit -1 multipliziert. Ferner werden für die ersten k Zeilen künstliche Variable $\tilde{y}_i \geq 0$ eingeführt, so dass dann gilt

$$\begin{array}{ccccccccccc} -a_{11} & x_1 & - \dots & -a_{1n} & x_n & -y_1 & & +\tilde{y}_1 & & = & -b_1 \\ & & & \vdots & & \ddots & & \ddots & & & \vdots \\ -a_{k1} & x_1 & - \dots & -a_{kn} & x_n & -y_k & & +\tilde{y}_k & & = & -b_k \\ a_{k+1,1}x_1 & + \dots & & +a_{k+1,n}x_n & & & +y_{k+1} & & & = & b_{k+1} \\ & & & \vdots & & & \ddots & & & & \vdots \\ a_{m1} & x_1 & + \dots & +a_{mn} & x_n & & +y_m & & & = & b_m \end{array} \quad (5.1)$$

Eine Lösung (x, y, \tilde{y}) von (5.1) entspricht genau dann einer Lösung des Ausgangsproblems

$$Ax \leq b, x \geq 0,$$

wenn $x \geq 0, y \geq 0$ und $\tilde{y} = 0$ gilt. Dies führt daher auf das folgende Hilfsproblem

$$\text{Minimiere } \sum_{i=1}^k \tilde{y}_i \text{ unter (5.1) und den Vorzeichenbedingungen } x \geq 0, y \geq 0, \tilde{y} \geq 0.$$

Für dieses Hilfsproblem gibt es zwei Möglichkeiten:

1. Das Hilfsproblem besitzt eine Optimallösung mit Zielfunktionswert 0. Dann ist die zugehörige Basislösung für das Ausgangsproblem zulässig.
2. Das Hilfsproblem besitzt eine Optimallösung mit Zielfunktionswert > 0 . Dann besitzt das Ausgangsproblem keine zulässige Lösung.

In Phase I wird die Hilfszielfunktion minimiert. Man kann dazu als Ausgangsbasis die Variablen y_i wählen, die zu $b_i \geq 0$ gehören und die Variablen \tilde{y}_i , die zu $b_i < 0$ gehören. Nun ist die Hilfszielfunktion durch die augenblicklichen Nichtbasisvariablen auszudrücken. Dazu summieren wir die ersten k Zeilen in (5.1) und erhalten

$$-\sum_{i=1}^k a_{i1}x_1 - \dots - \sum_{i=1}^k a_{in}x_n - \sum_{i=1}^k y_i + \sum_{i=1}^k \tilde{y}_i = -\sum_{i=1}^k b_i.$$

Daher ist

$$-\sum_{i=1}^k b_i - \sum_{i=1}^k \tilde{y}_i = -\sum_{j=1}^n \sum_{i=1}^k a_{ij}x_j - \sum_{i=1}^k y_k.$$

Da wir das Simplexverfahren auf ein Maximierungsproblem anwenden wollen, erhält man für dessen Zielfunktionskoeffizienten

$$d_0 := -\sum_{i=1}^k b_i, \quad d_j := -\sum_{i=1}^k a_{ij} \quad (1 \leq j \leq n), \quad d_{n+i} := -1 \quad (1 \leq i \leq k).$$

Man maximiert nun die negative Hilfszielfunktion. Erhält man für dieses Problem einen negativen Optimalwert, so ist das Ausgangsproblem nicht zulässig. Ist aber der Optimalwert gleich 0, so sind alle künstlich eingeführten Variablen gleich 0. Dann kann man aber (eventuell durch zusätzliche Simplexschritte) erreichen, dass alle künstlichen Variablen Nichtbasisvariable sind. Man streicht die zugehörigen Spalten im Tableau und setzt in Phase II mit der Optimierung der ursprünglich gegebenen Zielfunktion fort.

Beispiel 5.1

Maximiere $-x_1 - 2x_2$ unter den Restriktionen

$$\begin{aligned} x_1 + x_2 &\geq 3 \\ x_2 &\geq 2 \\ -x_1 + x_2 &\leq 3 \\ x_1 - x_2 &\leq 3 \\ x_j &\geq 0 \text{ für } j = 1, 2. \end{aligned}$$

Aufgrund der vorangegangenen Überlegungen führen wir neben den Schlupfvariablen y_1, \dots, y_4 noch zwei künstliche Variablen \tilde{y}_1, \tilde{y}_2 ein, und erhalten dann folgendes Simplextableau:

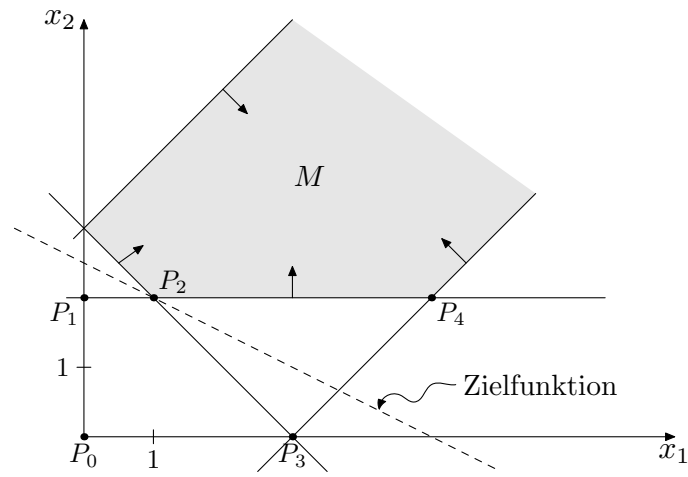


Abbildung 5.1: Graphische Darstellung des Beispiels 5.1

Ausgangstableau

	x_1	x_2	y_1	y_2		
	5	1	2	-1	-1	Hilfszielfunktion
	0	-1	-2	0	0	Zielfunktion
\tilde{y}_1	3	1	1	-1	0	
\tilde{y}_2	2	0	1	0	-1	
y_3	3	-1	1	0	0	
y_4	3	1	-1	0	0	

Start mit Ausgangslösung $x_1 = x_2 = 0$ (entspricht in Abbildung 5.1 dem Punkt P_0). Die Hilfszielfunktion wird nun mit Hilfe von Algorithmus 1 maximiert. Man erhält $s = 2, r = 2$.

Tableau 1

	x_1	\tilde{y}_2	y_1	y_2
1	1	-2	-1	1
4	-1	2	0	-2
\tilde{y}_1	1	1	-1	1
x_2	2	0	1	-1
y_3	1	-1	-1	1
y_4	5	1	1	-1

Diese Lösung entspricht P_1 mit $x_1 = 0, x_2 = 2$. Nun wird $s = r = 1$, und man erhält:

Tableau 2

		\tilde{y}_1	\tilde{y}_1	y_1	y_2
	0	-1	-1	0	0
	5	1	1	-1	-1
x_1	1	1	-1	-1	1
x_2	2	0	1	0	-1
y_3	2	1	-2	-1	2
y_4	4	-1	2	1	-2

Dieses Tableau entspricht P_2 mit $x_1 = 1, x_2 = 2$. Damit ist das Maximum der Hilfszielfunktion erreicht. Da der zugehörige Zielfunktionswert gleich 0 ist, haben wir eine zulässige Lösung des Ausgangsproblems erreicht. Die zu \tilde{y}_1 und \tilde{y}_2 gehörigen Spalten werden im Tableau gestrichen, ebenso die Zeile der Hilfszielfunktion. Das verbleibende Tableau dient nun als Ausgangslösung für Algorithmus 1. In unserem Fall ist es bereits optimal. Daher ist P_2 die gesuchte Optimallösung.

5.2 Die M-Methode

Gegeben sei wieder das Problem

$$\max \{c'x \mid Ax \leq b, x > 0\}.$$

Wir führen das System $Ax \leq b$ durch Schlupfvariable und eine künstliche Variable $\tilde{x} \geq 0$ in folgendes System über:

$$\begin{array}{rcccccc} a_{11}x_1 + a_{12} & x_2 + \cdots + a_{1n} & x_n + x_{n+1} & & & -\tilde{x} = b_1 \\ a_{21}x_1 + a_{22} & x_2 + \cdots + a_{2n} & x_n & +x_{n+2} & & -\tilde{x} = b_2 \\ \vdots & & & & & \vdots \quad \vdots \\ a_{m1}x_1 + a_{m2} & x_2 + \cdots + a_{mn} & x_n & & +x_{n+m} & -\tilde{x} = b_m \end{array}$$

Das Ausgangsproblem besitzt genau dann eine zulässige Lösung, wenn es eine Lösung des erweiterten Systems mit $\tilde{x} = 0$ gibt. Wie kommen wir nun zu einer zulässigen Ausgangslösung für das erweiterte System? O.B.d.A. sei $b_m = \min_{1 \leq i \leq m} b_i$ und $b_m < 0$. Ziehen wir dann die m -te Zeile von den ersten $m - 1$ Zeilen ab, so erhalten wir das System:

$$\begin{array}{rcccccc} (a_{11} & -a_{m1})x_1 + \cdots + (a_{1n} & -a_{mn})x_n + x_{n+1} & & -x_{n+m} = & b_1 & -b_m \\ (a_{21} & -a_{m1})x_1 + \cdots + (a_{2n} & -a_{mn})x_n & +x_{n+2} & -x_{n+m} = & b_2 & -b_m \\ \vdots & & \vdots & & \vdots & & \vdots \\ (a_{m-1,1} - a_{m1})x_1 + \cdots + (a_{m-1,n} & -a_{mn})x_n & & +x_{n+m-1} & -x_{n+m} = & b_{m-1} - b_m \\ -a_{m1} & x_1 + \cdots + & -a_{mn} & x_n & +\tilde{x} - x_{n+m} = & & -b_m \end{array}$$

Die rechte Seite dieses Systems ist nicht negativ. Daher ist durch $(x_{n+1}, \dots, x_{n+m-1}, \tilde{x})$ eine zulässige Basislösung gegeben. Man erhält diese Basislösung, wenn man im Ausgangsproblem die Basisvariable x_{n+m} gegen die Nichtbasisvariable \tilde{x} austauscht. Die künstliche Variable \tilde{x} hat in einer Optimallösung des Ausgangsproblems notwendigerweise den Wert 0. Damit \tilde{x} möglichst

rasch aus der Basis eliminiert wird, modifiziert man die gegebene Zielfunktion $c'x$ zu $c'x - M\tilde{x}$ und erteilt M einen gegenüber den Konstanten c_j großen Wert. Ist für hinreichend großes M in der Optimallösung $\tilde{x} > 0$, so widersprechen die Restriktionen einander, d.h. das Ausgangsproblem ist nicht zulässig.

Um zu vermeiden für die Konstante M einen numerischen Wert festzulegen, kann man eine lexikographische Zielfunktion einführen. Dazu stellt man die Zielfunktionskoeffizienten in der Form $c_j = \alpha_j M + \beta_j$ dar und betrachtet die Vektoren $(\alpha_j, \beta_j)'$ als lexikographisch geordnet. Allgemein ist die *lexikographische Ordnung von Vektoren* folgenderweise definiert:

Definition 5.1 Ein Vektor $v = (v_1, \dots, v_n)$ heißt lexikographisch positiv ($v \succ 0$), wenn das erste von Null verschiedene Element von v positiv ist. Ein Vektor v heißt lexikographisch größer oder gleich als Vektor u , in Zeichen $v \succeq u$, wenn $v = u$ oder $v - u \succ 0$ gilt.

So ist etwa $v = (0, 0, 1, -20)$ lexikographisch positiv und lexikographisch größer als $u = (0, 0, 0, 100)$. Die lexikographische Ordnung \succeq von Vektoren $u \in \mathbb{R}^n$ erfüllt die Ordnungsaxiome, sie ist nämlich reflexiv, transitiv und antisymmetrisch. Außerdem ist sie eine vollständige Ordnung, d.h. es gilt $v \succ u$ oder $v = u$ oder $u \succ v$. Ferner ist die lexikographische Ordnung verträglich mit der Vektoraddition

$$v \succeq u \Rightarrow \forall w \in \mathbb{R}^n : v + w \succeq u + w$$

und verträglich mit der skalaren Multiplikation:

$$c > 0, v \succeq u \Rightarrow c \cdot v \succeq c \cdot u.$$

Zur Durchführung der M-Methode führt man im Tableau eine zusätzliche Zielfunktionszeile ein. Wenn nun ein Koeffizient der Zielfunktion $\alpha M + \beta$ lautet, schreibt man α in die erste Zielfunktionszeile und β in die zweite. Somit steht im Anfangstableau in der ersten Zielfunktionszeile der Koeffizient -1 in der Spalte, die \tilde{x} zugeordnet ist. Alle anderen Koeffizienten der Zeile sind 0. In der zweiten Zielfunktionszeile stehen die gegebenen Koeffizienten c_j und in der Spalte von \tilde{x} der Koeffizient 0. Das erste Pivotelement ist wieder jenes Element der \tilde{x} zugeordneten Spalte, für das die rechte Seite am kleinsten ist. Damit wird \tilde{x} Basisvariable. Man wählt im weiteren die Pivotspalte durch eine *lexikographische Spaltenauswahlregel*:

Als Pivotspalte wird eine Spalte s gewählt, für die der Zielfunktionsvektor $(\alpha_j, \beta_j)'$ lexikographisch positiv ist.

Sobald die erste Zielfunktionszeile den Zielfunktionswert 0 erreicht hat, kann \tilde{x} als Nichtbasisvariable gewählt werden und man kann die zugehörige Spalte sowie die erste Zielfunktionszeile streichen. Damit hat man eine zulässige Ausgangslösung für das gegebene Problem gefunden. Kann die Variable \tilde{x} nicht aus der Basis eliminiert werden, so ist das gegebene Problem nicht zulässig.

Beispiel 5.2 (Beispiel wie in 5.1)

Ausgangstableau

	x_1	x_2	\tilde{x}
	0	0	-1
	0	-1	-2
x_3	-3	-1	-1
x_4	-2	0	-1
x_5	3	-1	1
x_6	3	1	-1

Tableau 1

	x_1	x_2	x_3
	3	1	1
	0	-1	-2
\tilde{x}	3	1	1
x_4	1	1	0
x_5	6	0	2
x_6	6	2	0

Tableau 2

	x_4	x_2	x_3
	2	-1	1
	1	1	-2
\tilde{x}	2	-1	1
x_1	1	1	0
x_5	6	0	2
x_6	6	-2	0

Tableau 3

	x_4	\tilde{x}	x_3
	0	0	-1
	5	-1	2
x_2	2	-1	1
x_1	1	1	0
x_5	2	2	-2
x_6	4	-2	0

Damit ist wieder die Optimallösung $x_1 = 1, x_2 = 2$ erreicht.

Tableau 1

	x_5	x_2	x_3	x_4	
	0	-3	30	$\frac{7}{50}$	-33
x_1	0	4	-240	$-\frac{4}{25}$	36
x_6	0	-2	30	$\frac{3}{50}$	-15
x_7	1	0	0	1	0

Tableau 2

	x_5	x_6	x_3	x_4	
	0	-1	-1	$\frac{2}{25}$	-18
x_1	0	-12	8	$\frac{8}{25}$	-84
x_2	0	$-\frac{1}{15}$	$\frac{1}{30}$	$\frac{1}{500}$	$-\frac{1}{2}$
x_7	1	0	0	1	0

Tableau 3

	x_5	x_6	x_1	x_4	
	0	2	-3	$-\frac{1}{4}$	3
x_3	0	$-\frac{75}{2}$	25	$\frac{25}{8}$	$-\frac{528}{2}$
x_2	0	$\frac{1}{120}$	$-\frac{1}{60}$	$-\frac{1}{160}$	$\frac{1}{40}$
x_7	1	$\frac{75}{2}$	-25	$-\frac{25}{8}$	$\frac{525}{2}$

Tableau 4

	x_5	x_6	x_1	x_2	
	0	1	-1	$\frac{1}{2}$	-120
x_3	0	50	-150	$-\frac{125}{2}$	10500
x_4	0	$\frac{1}{3}$	$-\frac{2}{3}$	$-\frac{1}{4}$	40
x_7	1	-50	150	$\frac{125}{2}$	-10500

Tableau 5

	x_3	x_6	x_1	x_2	
0	$-\frac{1}{50}$	2	$\frac{7}{4}$	-330	
x_5	0	$\frac{1}{50}$	-3	$-\frac{5}{4}$	210
x_4	0	$-\frac{1}{150}$	$\frac{1}{3}$	$\frac{1}{6}$	-30
x_7	1	1	0	0	0

Tableau 6

	x_3	x_4	x_1	x_2	
0	$\frac{1}{50}$	-6	$\frac{3}{4}$	-150	
x_5	0	$-\frac{1}{25}$	9	$\frac{1}{4}$	-60
x_6	0	$-\frac{1}{50}$	3	$\frac{1}{2}$	-90
x_7	1	1	0	0	0

Im Tableau 6 erhält man dieselbe Basislösung wie im Ausgangstableau. Daher gerät der Simplexalgorithmus in der Form von Algorithmus 1 bei diesem Beispiel ins Kreisen.

Zur Behebung des Kreisans kann man folgenderweise vorgehen. Wir führen durch die Schlupfvariablen x_{n+1}, \dots, x_{n+m} das Ungleichungssystem $Ax \leq b$ in ein Gleichungssystem über, und stellen dieses in Tableauform dar, wobei auch die zu den Schlupfvariablen gehörende Einheitsmatrix gespeichert wird. O.B.d.A. können wir nun annehmen, dass die Zeilenvektoren unseres Tableaus lexikographisch positiv sind. (Durch Spaltenvertauschungen ist dies stets erreichbar.) Die Pivotzeile wird nun durch folgende Regel bestimmt:

Lexikographische Zeilenauswahlregel:

Die Pivotzeile r wird durch das lexikographische Minimum der gewichteten Zeilenvektoren

$$\text{lexmin}_{1 \leq i \leq m} \left\{ \frac{1}{a_{is}} \cdot a_i \mid a_{is} > 0 \right\}$$

bestimmt.

Das lexikographische Minimum in der obigen Zeilenauswahlregel ist eindeutig bestimmt. Die Annahme, es wäre nicht eindeutig, führt auf

$$\frac{1}{a_{is}} \cdot a_i = \frac{1}{a_{ks}} \cdot a_k, \quad i \neq k,$$

und daher ist die i -te Zeile ein Vielfaches der k -ten Zeile:

$$a_i = \lambda \cdot a_k, \quad \lambda \neq 0.$$

Nun ist aber der Rang von A gleich m , d.h. die Zeilenvektoren sind linear unabhängig. Dies ist aber ein Widerspruch zu $a_i = \lambda a_k$.

Satz 6.1 Wählt man die Pivotzeile durch die lexikographische Zeilenauswahlregel, so gilt

1. Der Vektor in der Zielfunktionszeile nimmt streng lexikographisch ab.
2. Alle Zeilenvektoren bleiben lexikographisch positiv.

Beweis:

1. Für die neuen Zielfunktionskoeffizienten gilt

$$\bar{a}_{0j} := a_{0j} - a_{rj} \cdot \underbrace{\frac{a_{0s}}{a_{rs}}}_{>0} \quad (1 \leq j \leq n + m).$$

Da das erste nichtverschwindende $a_{rj} > 0$ ist, erhält man

$$\bar{a}_0 \prec a_0.$$

Also nimmt die Zielfunktionszeile lexikographisch ab.

2. Ist $a_{is} > 0$, dann erhält man für die transformierte Zeile \tilde{a}_i

$$\frac{1}{a_{is}} \cdot a_i \succ \frac{1}{a_{rs}} \cdot a_r \Rightarrow \tilde{a}_i = a_i - \frac{a_{is}}{a_{rs}} \cdot a_r \succ 0.$$

Ist aber $a_{is} \leq 0$, so ist wegen $a_{rs} > 0$

$$\tilde{a}_i = a_i - \frac{a_{is}}{a_{rs}} \cdot a_r \succ a_i.$$

Somit bleiben alle Zeilenvektoren lexikographisch positiv. ■

Satz 6.2 Das Simplexverfahren mit lexikographischer Zeilenauswahlregel ist endlich.

Beweis: Da die 0-te Zeile lexikographisch streng abnimmt, kann eine Basislösung nicht zweimal auftreten. Da es nur endlich viele verschiedene Basislösungen gibt, ist daher der Algorithmus endlich. ■

Beispiel 6.2 Wir lösen nun das lineare Programm von Beispiel 6.1 mittels des Simplexverfahrens mit lexikographischer Zeilenauswahlregel. Man erhält

Ausgangstableau

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	$\frac{3}{4}$	-150	$\frac{1}{50}$	-6	0	0	0
0	$\frac{1}{4}$	-60	$-\frac{1}{25}$	9	1	0	0
0	$\frac{1}{2}$	-90	$-\frac{1}{50}$	3	0	1	0
1	0	0	1	0	0	0	1

Tableau 1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	0	30	$\frac{7}{50}$	-33	-3	0	0
0	1	-240	$-\frac{4}{25}$	36	4	0	0
0	0	30	$\frac{3}{50}$	-15	-2	1	0
1	0	0	1	0	0	0	1

Tableau 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	0	0	$\frac{2}{25}$	-18	-1	-1	0
0	1	0	$\frac{8}{25}$	-84	-12	8	0
0	0	1	$\frac{1}{500}$	$-\frac{1}{2}$	$-\frac{1}{15}$	$\frac{1}{30}$	0
1	0	0	1	0	0	0	1

Tableau 3

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	0	-40	0	2	$\frac{5}{3}$	$-\frac{7}{3}$	0
0	1	-160	0	-4	$-\frac{4}{3}$	$\frac{8}{3}$	0
0	0	500	1	-250	$-\frac{100}{3}$	$\frac{50}{3}$	0
1	0	-500	0	250	$\frac{100}{3}$	$-\frac{50}{3}$	1

Tableau 4

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
$-\frac{1}{125}$	0	-36	0	0	$\frac{7}{5}$	$-\frac{11}{5}$	$-\frac{1}{125}$
$\frac{2}{125}$	1	-168	0	0	$-\frac{4}{5}$	$\frac{12}{5}$	$\frac{2}{125}$
1	0	0	1	0	0	0	1
$\frac{1}{250}$	0	-2	0	1	$\frac{2}{15}$	$-\frac{1}{15}$	$\frac{1}{250}$

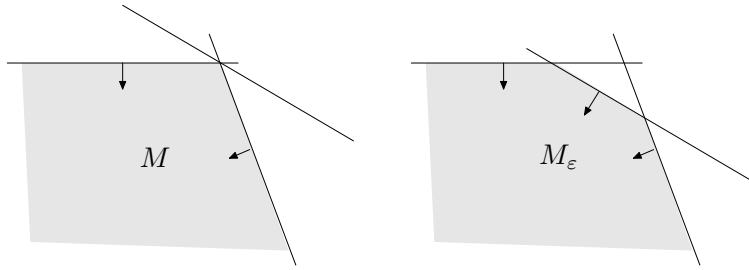
Abbildung 6.1: Entartete Ecke und Behebung der Entartung durch *Störung*

Tableau 5

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
$-\frac{1}{20}$	0	-15	0	$-\frac{21}{2}$	0	$-\frac{3}{2}$	$-\frac{1}{20}$
$\frac{1}{25}$	1	-180	0	6	0	2	$\frac{1}{25}$
1	0	0	1	0	0	0	1
$\frac{3}{100}$	0	-15	0	$\frac{15}{2}$	1	$-\frac{1}{2}$	$\frac{3}{100}$

Wie man an diesem Beispiel erkennt, wird die entartete Ecke bereits im 4. Tableau verlassen. Tableau 5 gibt die zugehörige Optimallösung wieder.

Geometrisch lässt sich die Behebung der Entartung so interpretieren: In einer entarteten Ecke treffen mehr als n Hyperebenen zusammen. Durch eine kleine Störung werden sie auseinandergezogen, und man erhält lauter nichtentartete Basislösungen. Ersetzt man b_i durch

$$\bar{b}_i := a_{i0} + a_{i1}\varepsilon + a_{i2}\varepsilon^2 + \dots + a_{i,n+m}\varepsilon^{n+m}$$

für hinreichend kleines $\varepsilon > 0$, so ist $\bar{b} > 0$, und die lexikographische Zeilenauswahlregel garantiert, dass $\bar{b}_i > 0$ bleibt. Somit ist dieses *gestörte Problem* stets nichtentartet. Lässt man ε gegen Null gehen, erhält man eine Lösung des gegebenen Problems.

Eine andere einfache Möglichkeit das Kreisen beim Simplexverfahren zu verhindern, gab Bland [5] an. Sie beruht auf einfachen kombinatorischen Überlegungen und ist nicht von Zahlenwerten anhängig. Daher spielt sie eine Rolle bei der Verallgemeinerung des Simplexverfahrens in orientierten Matroiden. Die Regel von Bland schreibt nicht nur die Wahl der Pivotzeile vor, sondern auch die Wahl der Pivotspalte.

Satz 6.3 (Bland, 1977) Wird beim Simplexverfahren die Pivotspalte s durch

$$N(s) := \min\{N(j) \mid a_{0j} > 0, 1 \leq j \leq n\}$$

und die Pivotzeile r durch

$$B(r) := \min \left\{ B(t) \mid \frac{a_{t0}}{a_{ts}} = \min_{\substack{1 \leq i \leq m \\ a_{is} > 0}} \frac{a_{i0}}{a_{is}} \right\}$$

bestimmt, so liefert das Verfahren nach endlich vielen Schritten eine Optimallösung, oder es wird nachgewiesen, dass keine endliche Optimallösung existiert.

Beweis: Durch die Regeln von Satz 6.3 ist die Folge der Pivotoperationen eindeutig festgelegt. Wir nehmen an, dass man ausgehend von einer zulässigen Ausgangslösung durch Anwendung der obigen Regeln ins Kreisen gerät. Der Zyklus ist dann eindeutig bestimmt. Es sei $T \subseteq \{1, \dots, m+n\}$ die Indexmenge der Variablen, die während des Zyklus in die Basis eintreten, d.h. $j \notin T$ bedeutet, x_j ist während des ganzen Zyklus Basisvariable oder Nichtbasisvariable. Es sei

$$q := \max\{j \mid j \in T\}$$

und (\bar{A}, \bar{b}) sei das Tableau (mit $m+1$ Zeilen und $n+m+1$ Spalten), in dem die Spalte q Pivotspalte ist. Den Vektor in der 0-ten Zeile von \bar{A} , die die Zielfunktionskoeffizienten enthält, bezeichnen wir mit y , d.h.

$$y_j := \bar{a}_{0j} \quad (1 \leq j \leq m+n).$$

Es gilt

$$y_j \leq 0 \quad (1 \leq j \leq q-1), \quad y_q > 0. \quad (6.1)$$

Der Vektor y liegt im Unterraum von \mathbb{R}^{n+m} , der von den $m+1$ Zeilen der Matrix A erzeugt wird. Da die Pivotoperationen Linksmultiplikationen mit regulären Matrizen entsprechen, bleibt dieser Zeilenraum während des ganzen Verfahrens gleich.

Da x_q im Zyklus Basisvariable wird, muss es auch im Zyklus wieder die Basis verlassen. Es sei also (\tilde{A}, \tilde{b}) das Tableau, in dem die zu x_q gehörende Zeile Pivotzeile wird. Ferner sei t der Index der Pivotspalte in diesem Tableau (\tilde{A}, \tilde{b}) . Bezeichnen wir mit B die zum Tableau (\tilde{A}, \tilde{b}) gehörende Basis, und definieren wir einen Vektor $z \in \mathbb{R}^{m+n}$ durch

$$\begin{aligned} z_t &:= -1 \\ z_{B(i)} &:= \tilde{a}_{it} \\ z_j &:= 0, \quad \text{andernfalls,} \end{aligned}$$

so gilt für alle Zeilenvektoren \tilde{a}_i ($0 \leq i \leq m$) von \tilde{A} :

$$\tilde{a}_i' z = 0.$$

Daher liegt z im orthogonalen Komplement des Zeilenraumes und es gilt insbesondere

$$y' z = 0. \quad (6.2)$$

Nun ist z_q gleich dem Pivotelement im Tableau (\tilde{A}, \tilde{b}) und $y_q > 0$, da y_q der relative Kostenkoeffizient in der Pivotspalte von (\bar{A}, \bar{b}) ist. Daher ist $y_q \cdot z_q > 0$, und folglich gibt es wegen (6.2) ein j , so dass $y_j \cdot z_j < 0$ ist. Die Beziehung $y_j \neq 0$ besagt, dass x_j Nichtbasisvariable im Tableau (\bar{A}, \bar{b}) ist, andererseits besagt $z_j \neq 0$, dass x_j Basisvariable in (\tilde{A}, \tilde{b}) oder $j = t$ ist. Daher ist $j \in T$, und wegen $j \neq q$ gilt $j < q$. Dann ist aber $y_j < 0$, und daher $z_j > 0$. Also ist $j \neq t$. Somit ist x_j eine Basisvariable in (\tilde{A}, \tilde{b}) , etwa $j = B(p)$ und $\tilde{a}_{pt} = z_j > 0$.

Die rechte Seite b bleibt während des ganzen Zyklus unverändert. Daher behält jede Variable während des Zyklus ihren Wert bei. Insbesondere folgt daraus für $x_j, j \in T$: $x_j = 0$ für den ganzen Zyklus. Daraus folgt $b_p = 0$ in allen Tableaus des Zyklus, insbesondere also $\tilde{b}_p = 0$. Also gilt

$$j = B(p) < q, \quad \tilde{a}_{pt} > 0, \quad \tilde{b}_p = 0.$$

Nach der Zeilenauswahl von Satz 6.3 müsste daher eine Variable $x_j, j < q$, die Basis verlassen im Widerspruch dazu, dass in diesem Tableau x_q die Basis verlässt. Dies ist ein Widerspruch. Daher kann kein Kreisen auftreten. ■

Man beachte, dass bei der Regel von Bland sowohl Pivotspalte als auch Pivotzeile vorgeschrieben werden. Wird nur die Zeilenauswahlregel verwendet, so gibt es Beispiele des Kreisens.

Von Dantzig wurde die Frage nach möglichst kleinen linearen Programmen aufgeworfen, bei denen ein Kreisen auftritt. Es gilt

Satz 6.4 (Marshall und Suurballe [38], 1969)

Wird im Simplexverfahren die Pivotspalte durch

$$s := \min\{j^* \mid c_{j^*} = \max c_j\}$$

und die Pivotzeile durch

$$r := \min \left\{ i^* \mid \frac{b_{i^*}}{a_{i^*s}} = \min \left(\frac{b_i}{a_{is}} \mid a_{is} > 0 \right) \right\}$$

gewählt, so muss $m \geq 2, n \geq 3$ und $m + n \geq 6$ gelten, damit ein Kreisen auftritt. Falls ein Kreisen in einer nichtoptimalen Ecke auftritt, muss $m \geq 3, n \geq 3$ und $m + n \geq 7$ gelten. Ferner sind diese Schranken scharf.

Ein Beweis dieses Satzes sowie Beispiele für die angegebenen Werte von m und n finden sich in [38].

Kapitel 7

Spaltenauswahlregeln

Wie man im Abschnitt 6 sehen konnte, beeinflusst die Wahl der Pivotspalte und Pivotzeile wesentlich das Simplexverfahren. Man kann nun danach fragen, welche Auswahlregel für die Pivotspalte möglichst günstig ist, d.h. durch welche Wahl der Pivotspalte führt die Simplexmethode möglichst rasch zur Optimallösung. In numerischen Versuchen (z.B. Kuhn und Quandt [35], 1963) stellten sich vor allem die drei folgenden Typen von Spaltenauswahlregeln als günstig heraus.

- **Methode des steilsten Anstiegs im Raum der Nichtbasisvariablen**

$$\tilde{c}_s := \max_{j \in N} \tilde{c}_j \quad (7.1)$$

\tilde{c} ist der Gradient der Zielfunktion im Raum der (augenblicklichen) Nichtbasisvariablen. Daher kann ein Simplexverfahren mit Regel (7.1) als Nichtbasis-Gradientenverfahren bezeichnet werden.

- **Methode des größten absoluten Zuwachses**

Wählt man eine Spalte j mit $\tilde{c}_j > 0$ als Pivotspalte, so wird die zugehörige Pivotzeile $r(j)$ durch die Zeilenauswahlregel festgelegt. Damit erhält man $\tilde{a}_{r(j),j}$ als Pivotelement. Bei der zugehörigen Pivottransformation würde sich der Zielfunktionswert um

$$\frac{\tilde{c}_j \cdot \tilde{b}_{r(j)}}{\tilde{a}_{r(j),j}} \quad (7.2)$$

ändern. Man kann nun die Pivotspalte s dadurch festlegen, dass der Ausdruck (7.2) möglichst groß wird. Bei dieser Spaltenauswahlregel wird also jene Nichtbasisvariable neu in die Basis aufgenommen, die in dieser Iteration den größten absoluten Zuwachs des Zielfunktionswertes ergibt.

- **Methode des steilsten Anstiegs im Raum aller Variablen**

Wir fassen die Basis- und Nichtbasisvariablen zu einem Vektor $x \in \mathbb{R}^{n+m}$ zusammen, wobei x_i ($1 \leq i \leq m$) den Basisvariablen und x_{m+j} ($1 \leq j \leq n$) den Nichtbasisvariablen entspreche. Einer Basislösung entspricht dann ein Punkt $\begin{pmatrix} \tilde{b} \\ 0 \end{pmatrix} \in \mathbb{R}^{n+m}$. Ändert man die

Nichtbasisvariable $x_{N(j)}$ um eine Einheit, so ändert sich einerseits der Zielfunktionswert um \tilde{c}_j , andererseits ändern sich auch die Werte der Basisvariablen

$$x_B = A_B^{-1}b - A_B^{-1}a_j x_{N(j)} = \tilde{b} - \tilde{a}_j,$$

wobei \tilde{a}_j die zu $x_{N(j)}$ gehörige Spalte der Matrix \tilde{A} ist. Somit ändern sich die Koordinaten des betrachteten Punktes im \mathbb{R}^{n+m} um den Vektor $(-\tilde{a}_{1j}, \dots, -\tilde{a}_{mj}, 0, \dots, 1, 0, \dots, 0)'$. Daher erhält man als $N(j)$ -te Komponente des Gradienten der Zielfunktion im Raum \mathbb{R}^{n+m}

$$\frac{\tilde{c}_j}{\sqrt{1 + \sum_{i=1}^m \tilde{a}_{ij}^2}}$$

und der steilste Anstieg der Zielfunktion erfolgt in der Richtung $x_{N(s)}$, die durch

$$\frac{\tilde{c}_s^2}{1 + \sum_{i=1}^m \tilde{a}_{is}^2} := \max \left\{ \frac{\tilde{c}_j^2}{1 + \sum_{i=1}^m \tilde{a}_{ij}^2} \mid \tilde{c}_j > 0 \right\} \quad (7.3)$$

festgelegt wird.

Ein numerischer Vergleich der Spaltenauswahlregeln (7.1), (7.2) und (7.3) ergab folgende mittlere Iterationsanzahlen bei 100 linearen Programmen mit $m = n = 25$ (vgl. Kuhn und Quandt [36], 1954):

	(7.1)	(7.2)	(7.3)
Mittlere Iterationsanzahl	34.6	22.5	18.6

Zwar führt das Simplexverfahren mit den Regeln (7.2) und (7.3) im Durchschnitt in weniger Iterationen zu einer Optimallösung als mit Regel (7.1), aber andererseits benötigen diese beiden Regeln einen wesentlich höheren Rechenaufwand als (7.1). So berichten Goldfarb und Reid [22], dass bei einem Linearen Programm mit $m = 821$ und $n = 1055$ beim Übergang von Regel (7.1) zu Regel (7.3) der Rechenaufwand für eine Simplexiteration im Mittel um 47% anwuchs, aber an Stelle von 3976 Simplexiterationen mit der Spaltenauswahlregel (7.1) nur mehr 1182 Iterationen notwendig waren, wenn die Spalte nach Regel (7.3) ausgewählt wurde.

Iterationsformeln zur Berechnung der für die Auswahlregel (7.3) benötigten Größen

$$x_j := 1 + \sum_{i=1}^m a_{ij}^2$$

wurden von Goldfarb und Reid [22] hergeleitet. Fügt man die Vorzeichenbedingungen zu den übrigen Restriktionen hinzu, so erhält man eine quadratische Restriktionsmatrix mit $m + n$ Zeilen und Spalten, die man in der Form

$$G = \begin{pmatrix} A_B & A_N \\ 0 & E \end{pmatrix}$$

schreiben kann, wobei E die Einheitsmatrix bezeichnet. Dabei nehmen wir an, dass die ersten m Spalten den Basisvariablen und die restlichen n Spalten den Nichtbasisvariablen entsprechen. Nun ist

$$H := G^{-1} = \begin{pmatrix} A_B^{-1} & -A_B^{-1}A_N \\ 0 & E \end{pmatrix}$$

und die Spalten h_j von H geben die Kantenrichtungen des Polyeders wieder, entlang derer, ausgehend von der Ecke $x_B = A_B^{-1}b$, $x_N = 0$, ein Eckentausch durchgeführt werden kann. Die reduzierten Kostenkoeffizienten sind

$$\bar{c}_j = (c'_B, c'_N)h_j.$$

Nach (7.3) soll die Austauschspalte so gewählt werden, dass

$$\frac{c'h_{m+s}}{\|h_{m+s}\|}, \quad \text{für } 1 \leq s \leq n$$

maximal wird. Zur rekursiven Berechnung der Größen $\|h_j\|^2$ ziehen wir die Sherman-Morrison Formel heran:

Lemma 7.1 *Sei G eine reelle, nichtsinguläre, quadratische Matrix, u und v seien zwei reelle Vektoren und $\alpha \in \mathbb{R}$. Dann gilt*

$$(G - \alpha uv')^{-1} = G^{-1} - \beta G^{-1}uv'G^{-1} \quad \text{mit} \quad \frac{1}{\alpha} + \frac{1}{\beta} = v'G^{-1}u. \quad (7.4)$$

Der Beweis von Lemma 7.1 ist leicht durch Nachrechnen zu erbringen. Nach einem Austausch der r -ten Basisvariablen gegen die s -te Nichtbasisvariable hat die Matrix G folgende neue Gestalt

$$\bar{G} = (G + e_{m+s}(e_r - e_{m+s})')P_{r,m+s} \quad (1 \leq r \leq m, 1 \leq s \leq n),$$

wobei e_r und e_{m+s} Einheitsvektoren sind und $P_{r,m+s}$ eine Permutationsmatrix ist, die die r -te und $(m+s)$ -te Spalte vertauscht. Unter Verwendung von Lemma 7.1 und $e'_{m+s}H = e'_{m+s}$ für $1 \leq s \leq n$ erhält man mit

$$\alpha = -1 \quad \text{und} \quad \beta = \frac{1}{1 + (e_r - e_{m+s})'He_{m+s}}$$

die Beziehung

$$\begin{aligned} P_{r,m+s}\bar{H} &= (\bar{G}P_{r,m+s})^{-1} = (G + e_{m+s}(e_r - e_{m+s})')^{-1} = \\ &= H - \frac{He_{m+s}(e_r - e_{m+s})'H}{1 + (e_r - e_{m+s})'He_{m+s}} = \\ &= H - \frac{He_{m+s}(e'_r H - e'_{m+s})}{e'_r He_{m+s}}. \end{aligned} \quad (7.5)$$

Multipliziert man (7.5) mit e_{m+j} ($1 \leq j \leq n$) von rechts, so erhält man

$$\begin{aligned} P_{r,m+s}\bar{h}_{m+s} &= h_{m+s} - \frac{h_{m+s}(e'_r h_{m+s} - 1)}{e'_r h_{m+s}} = \frac{h_{m+s}}{e'_r h_{m+s}} \quad j = s, \\ P_{r,m+s}\bar{h}_{m+j} &= h_{m+j} - h_{m+s} \frac{e'_r h_{m+j}}{e'_r h_{m+s}} \quad j \neq s. \end{aligned} \quad (7.6)$$

Nun ist $e'_r h_{m+j} = \tilde{a}_{rj}$ ($1 \leq j \leq n$). Daher vereinfacht sich (7.6) zu

$$\begin{aligned} P_{r,m+s} \bar{h}_{m+s} &= \frac{1}{\tilde{a}_{rs}} h_{m+s} \\ P_{r,m+j} \bar{h}_{m+j} &= h_{m+j} - \frac{\tilde{a}_{rj}}{\tilde{a}_{rs}} \cdot h_{m+s} \quad \text{für } j \neq s, 1 \leq j \leq n. \end{aligned}$$

Daraus erhält man mit $\bar{a}_{rj} := \frac{\tilde{a}_{rj}}{\tilde{a}_{rs}}$ die Rekursionsformel

$$\begin{aligned} \bar{\chi}_s &= \bar{h}'_{m+s} \bar{h}_{m+s} = \frac{1}{\tilde{a}_{rs}^2} \chi_s \\ \bar{\chi}_j &= \bar{h}'_{m+j} \bar{h}_{m+j} = \chi_{m+j} - 2\bar{a}_{rj} h'_{m+j} h_{m+s} + \bar{a}_{rj}^2 \chi_s \quad (j \neq s). \end{aligned} \tag{7.7}$$

Man beachte, dass \tilde{a}_{rj} bzw. \bar{a}_{rj} die Koeffizienten der r -ten Zeile im Simplextableau *vor* bzw. *nach* der Pivotoperation sind. Die Formeln (7.7) lassen sich auch durch die Ausgangsdaten ausdrücken:

$$\begin{aligned} \chi_s &= \|1 + A_B^{-1} a_s\|^2 \\ \bar{\chi}_s &= \frac{1}{\tilde{a}_{rs}^2} \chi_s \\ \bar{\chi}_j &= \chi_j - 2\bar{a}_{rj} (a'_j A_B^{-T} A_B^{-1} a_s) + \bar{a}_{rj}^2 \chi_s \quad (j \neq s) \end{aligned} \tag{7.8}$$

Die jeweiligen Werte von χ_j ($1 \leq j \leq n$) können somit im Simplexverfahren leicht berechnet werden. Eine günstige Speicherplatzbelegung und Rechenorganisation wird in [22] angegeben. Etwaige Rundungsfehler bei der Berechnung von $\bar{\chi}_j$ beeinflussen nur die Auswahl der Pivotspalte nicht aber die Genauigkeit des Verfahrens, und sind daher relativ harmlos. Aber es empfiehlt sich trotzdem, von Zeit zur Zeit

$$\chi_j = \|1 + A_B^{-1} a_j\|^2 \quad (j \in N)$$

aus den Ausgangsdaten neu zu berechnen.

Goldfarb und Reid berichten über Testergebnisse an sechs größeren linearen Programmen. So benötigte etwa das Simplexverfahren mit Spaltenauswahlregel (7.3) bei einem aus der Praxis stammenden linearen Programm mit $m = 821, n = 1055$ sowie 11719 von Null verschiedenen Matrixelementen 1182 Iterationen, während 3976 Iterationen nötig waren, um dieses Problem mit der Spaltenauswahlregel (7.1) zu lösen. Die Rechenzeit pro Iteration bei Verfahren mit steilem Anstieg war etwa das 1.47-fache der Rechenzeit pro Iteration des Standardverfahrens.

Der Frage, wieviel Iterationen das Simplexverfahren im Mittel (*average behaviour*) und im schlechtesten Fall (*worst case behaviour*) benötigt, werden wir im Kapitel 17 nachgehen.

Kapitel 8

Die Behandlung von Gleichungen, beschränkten Variablen und nicht vorzeichenbeschränkten Variablen im Simplexverfahren

Treten in der Formulierung eines linearen Programmes Gleichungen oder beschränkte Variablen der Form $0 \leq x \leq d$ auf, so kann man im Prinzip wie in Abschnitt 3 vorgehen, indem man eine Gleichung in zwei Ungleichungen aufspaltet bzw. die Ungleichung $x \leq d$ zu den Restriktionen $Ax \leq b$ hinzufügt. Dadurch wird die Koeffizientenmatrix A vergrößert, was sich negativ auf Rechenzeit und Rechengenauigkeit auswirken kann. Man kann jedoch den Ablauf des Simplexverfahrens leicht so modifizieren, dass Gleichungen und beschränkte Variable ohne Vergrößerung der Koeffizientenmatrix Berücksichtigung finden.

Oft kommen auch Variablen ohne Vorzeichenbeschränkungen in linearen Programmen vor. Auch hier ist es nicht nötig, diese Variablen wie in Abschnitt 3 aufzuspalten, sondern Optimalitätskriterium und Auswahlregel werden im Simplexverfahren leicht modifiziert, um diese Situation mitzuberücksichtigen.

8.1 Die Behandlung von Gleichungen

Liegt eine Restriktion als Gleichung vor, so wird wie bei der Zweiphasenmethode (Abschnitt 5.1) dafür eine künstliche Variable eingeführt, und zunächst eine Hilfszielfunktion, bestehend aus der Summe dieser künstlichen Variablen, minimiert. Ist das Problem zulässig, so sind in der Optimallösung dieses Hilfsproblems alle künstlichen Variablen Nichtbasisvariablen, und die zugehörigen Spalten können gestrichen werden. Das Verfahren wird mit der Maximierung der ursprünglich gegebenen Zielfunktion fortgesetzt.

Beispiel 8.1

$$\begin{array}{rcll}
 \text{Maximiere} & x_1 + 2x_2 - x_3 & & \\
 \text{unter} & x_1 + x_2 & \leq & 4 \\
 & 2x_1 + x_2 + x_3 & = & 5 \\
 & x_2 - x_3 & = & -1 \\
 & x_i & \geq & 0 \quad (i = 1, 2, 3)
 \end{array}$$

Zu diesem LP wird zunächst das folgende Hilfsproblem aufgestellt:

$$\begin{array}{rcll}
 \text{Minimiere} & & & y_1 + y_2 \\
 \text{unter} & x_1 + x_2 & + x_4 & = 4 \\
 & 2x_1 + x_2 + x_3 & + y_1 & = 5 \\
 & -x_2 + x_3 & + y_2 & = 1 \\
 & & & x_i \geq 0 \quad (i = 1, 2, 3, 4) \\
 & & & y_i \geq 0 \quad (i = 1, 2)
 \end{array}$$

Da $y_1 + y_2 = 6 - 2x_1 - 2x_3$ erhält man als

Ausgangstableau

	x_1	x_2	x_3		
	6	2	0	2	Hilfszielfunktion
	0	1	2	-1	Zielfunktion
x_4	4	1	1	0	
y_1	5	2	1	1	
y_2	1	0	-1	1	

Tableau 1

	y_1	x_2	x_3	
	1	-1	-1	1
	$-\frac{5}{2}$	$-\frac{1}{2}$	$\frac{3}{2}$	$-\frac{3}{2}$
x_4	$\frac{3}{2}$	$-\frac{1}{2}$	$\frac{1}{2}$	$-\frac{1}{2}$
x_1	$\frac{5}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
y_2	1	0	-1	1

Tableau 2

	y_1	x_2	y_2	
	0	-1	0	-1
	-1	$-\frac{1}{2}$	0	$-\frac{3}{2}$
x_4	2	$-\frac{1}{2}$	0	$\frac{1}{2}$
x_1	2	$\frac{1}{2}$	1	$-\frac{1}{2}$
x_3	1	0	-1	1

Damit ist die Optimallösung des Hilfsproblems erreicht. Die zu y_1, y_2 gehörenden Spalten werden gestrichen. Das übrigbleibende Tableau ist aber ebenfalls optimal, daher lautet die Optimallösung

$$x_1 = 2, x_2 = 0, x_3 = 1, z = 1.$$

8.2 Beschränkte Variable

Die Bedingung $0 \leq x \leq d$ ist äquivalent mit

$$x + \bar{x} = d, \quad x \geq 0, \quad \bar{x} \geq 0.$$

Durch den Wert von x (bzw. \bar{x}) ist die jeweilige Komplementärvariable \bar{x} (bzw. x) eindeutig bestimmt. Es genügt daher, im Simplextableau eine der beiden Variablen mitzuführen. Man muss jedoch für jede Zeile oder Spalte, die einer beschränkten Variablen entspricht, festhalten, ob sie x oder \bar{x} entspricht.

Im Simplexverfahren ist es nun nötig, die Zeilenauswahlregel zu modifizieren. Eine Nichtbasisvariable, die nach oben beschränkt ist, darf ja nicht beliebig groß gemacht werden. Andererseits dürfen durch eine Pivotoperation die Nichtnegativitätsbedingungen für x und \bar{x} nicht verletzt werden. Daher kann man folgenderweise vorgehen:

Es sei K die Indexmenge der beschränkten Variablen mit den oberen Schranken d_i ($0 < d_i < \infty$, $i \in K$). Ferner sei N der Indexvektor der augenblicklichen Nichtbasisvariablen und B jener der augenblicklichen Basisvariablen. s sei der Index der Austauschspalte. Ist $N(s) \in K$, so darf $x_{N(s)}$, bzw. $\bar{x}_{N(s)}$ nicht größer als $d_{N(s)}$ gemacht werden. Damit die Werte der Basisvariablen nichtnegativ bleiben, darf $x_{N(s)}$ nicht größer als $\frac{b_i}{a_{is}}$ gemacht werden, falls $a_{is} > 0$ ist. Schließlich ist es möglich, dass auch eine Basisvariable $x_{B(i)}$ nach oben durch $d_{B(i)}$ beschränkt ist. Damit der Wert dieser Basisvariablen nach der Pivotoperation nicht größer als $d_{B(i)}$ wird, muss also

$$x_{N(s)} \leq \frac{b_i - d_{B(i)}}{a_{is}} \quad \text{für } a_{is} < 0$$

gelten. Somit ist das Minimum von

$$d_{N(s)}, \quad \left\{ \frac{b_i}{a_{is}} \mid a_{is} > 0 \right\}, \quad \left\{ \frac{b_i - d_{B(i)}}{a_{is}} \mid a_{is} < 0, B(i) \in K \right\}$$

zu bestimmen. Ist die Menge

$$(\{N(s)\} \cap K) \cup \{i \mid a_{is} > 0\} \cup \{i \mid a_{is} < 0, B(i) \in K\}$$

leer, so gibt es keine endliche Optimallösung. Wird das Minimum durch $d_{N(s)}$ angenommen, so muss die zur s -ten Spalte gehörende Variable durch ihre Komplementärvariable ersetzt werden. Ersetzt man etwa in

$$a_{i1}x_1 + \dots + a_{in}x_n = b_i$$

die Variable x_s durch $d_{N(s)} - \bar{x}_s$, so erhält man

$$a_{i1}x_1 + \dots + a_{i,s-1}x_{s-1} - a_{is}\bar{x}_s + \dots + a_{in}x_n = b_i - a_{is}d_{N(s)}$$

Daher wird der Übergang zur Komplementärvariablen im Tableau durch folgende Transformation $T(s)$ vollzogen:

$$\begin{aligned} a_{i0} &:= a_{i0} - a_{is} \cdot d_{N(s)} & (0 \leq i \leq m) \\ a_{is} &:= -a_{is} \end{aligned}$$

Wird das Minimum für b_r/a_{rs} angenommen, so ist eine Pivotoperation mit dem Pivotelement a_{rs} durchzuführen.

Die dritte Möglichkeit besteht darin, daß das Minimum für $(b_r - d_{B(r)})/a_{rs}$ angenommen wird. Dies bedeutet, dass nach einem Variablenaustausch die neue Nichtbasisvariable ihre obere Schranke erreicht. Daher ist zuerst eine Pivotoperation mit dem Pivotelement a_{rs} durchzuführen. Dabei ändern sich die Vektoren N und B . Danach ist die Transformation $T(s)$ auszuführen. Die Rechenvorschrift lässt sich damit folgenderweise zusammenfassen:

Algorithmus zur Lösung des linearen Programms:

$$\max\{c'x \mid Ax \leq b, 0 \leq x_j \leq d_j \text{ für } j \in K, x_j \geq 0 \text{ für } j \notin K\} \text{ mit } b \geq 0.$$

Ausgangstableau a_{ij} ($0 \leq i \leq m$, $0 \leq j \leq n$), Vektoren N und B enthalten mit Vorzeichen versehene Indizes, wobei $B(i) = t$ der Variablen x_t und $B(i) = -t$ der Variablen \bar{x}_t entspricht. Analog für $N(j)$. $k := 0$.

1. Ist $a_{0j} \leq 0$ ($1 \leq j \leq n$), so ist die Basislösung optimal. Setze

$$x_{B(i)} := \begin{cases} b_i & B(i) > 0 \\ d_{B(i)} - b_i & B(i) < 0 \end{cases} \quad (1 \leq i \leq m) \quad (8.1)$$

$$x_{N(j)} := \begin{cases} 0 & N(j) > 0 \\ d_{N(j)} & N(j) < 0 \end{cases} \quad (1 \leq j \leq n). \quad (8.2)$$

Terminiere.

Andernfalls gehe zu 2.

2. Wähle s so, dass

$$\bar{a}_{0s} = \max_{1 \leq j \leq n} a_{0j}.$$

3. Bestimme das Minimum der Größen

$$d_{N(s)} \quad \text{mit } N(s) \in K,$$

$$\frac{a_{i0}}{a_{is}} \quad \text{mit } a_{is} > 0 \quad (1 \leq i \leq m),$$

$$\frac{a_{i0} - d_{B(i)}}{a_{is}} \quad \text{mit } a_{is} < 0, B(i) \in K \quad (1 \leq i \leq m).$$

Ist die Menge

$$\left(\{N(s)\} \cap K \right) \cup \left\{ i \mid a_{is} > 0, 1 \leq i \leq m \right\} \cup \left\{ i \mid a_{is} < 0, B(i) \in K, 1 \leq i \leq m \right\}$$

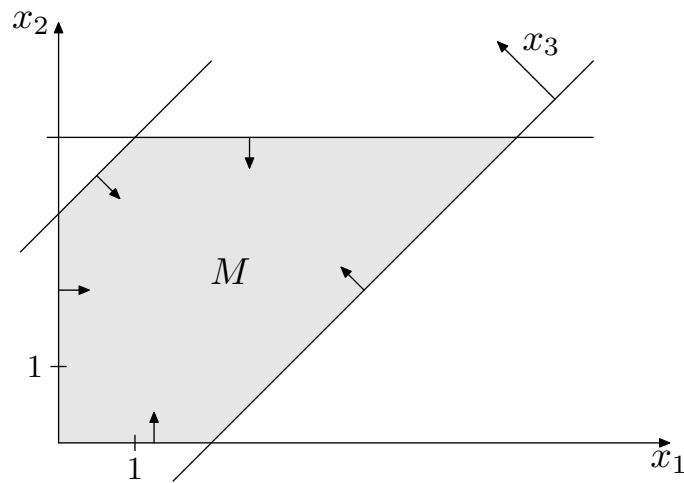


Abbildung 8.1: Zulässige Punkte für das Beispiel 8.2

leer, so existiert keine endliche Optimallösung. Terminiere.

Wird das Minimum für $d_{N(s)}$ angenommen, so gehe zu Schritt 5.

Wird das Minimum für $\frac{a_{r0}}{a_{rs}}$ angenommen, so gehe zu Schritt 4.

Wird das Minimum für $\frac{a_{r0} - d_{B(r)}}{a_{rs}}$ angenommen, so setze $k := 1$ und gehe zu Schritt 4.

4. Führe mit a_{rs} eine Pivotoperation durch. Vertausche die Elemente $B(r)$ und $N(s)$.
Ist $k = 1$, so setze $k := 0$ und gehe zu Schritt 5, sonst gehe zu Schritt 1.
5. Führe die Transformation $T(s)$ durch:

$$\begin{aligned} a_{i0} &:= a_{i0} - a_{is} \cdot d_{N(s)} & (0 \leq i \leq m) \\ a_{is} &:= -a_{is} \end{aligned}$$

Setze $N(s) := -N(s)$ und gehe zu 1.

Beispiel 8.2

$$\begin{aligned} & \text{Maximiere} & -x_1 & + & 4x_2 \\ & \text{unter} & x_1 & - & x_2 & \leq & 2 \\ & & -x_1 & + & x_2 & \leq & 3 \\ & & & & x_2 & \leq & 4 \\ & & & & x_i & \geq & 0 \quad (i = 1, 2) \end{aligned}$$

(vgl. Abbildung 8.1). Durch Einführen einer Schlupfvariablen x_3 erhält man

$$x_1 - x_2 + x_3 = 2, \quad x_3 \geq 0.$$

Die zweite Restriktion lässt sich nun als obere Schranke für x_3 deuten. Sie ist äquivalent mit $x_3 \leq 5$. Daher ist obiges Problem äquivalent zu

$$\begin{array}{ll} \text{Maximiere} & -x_1 + 4x_2 \\ \text{unter} & x_1 - x_2 + x_3 = 2 \\ & x_1 \geq 0, \quad 0 \leq x_2 \leq 4, \quad 0 \leq x_3 \leq 5 \end{array}$$

und man erhält als

Ausgangstableau

	x_1	x_2
0	-1	4
x_3	2	-1

mit $d_2 = 4$, $d_3 = 5$. Die zweite Spalte wird Pivotspalte. Daher hat man das Minimum von

$$d_2 = 4 \quad \text{und} \quad \frac{b_1 - d_3}{a_{12}} = 3$$

zu bilden. Da das Minimum für den zweiten Ausdruck angenommen wird, ist zuerst eine Pivotoperation mit -1 und dann eine Transformation $T(2)$ durchzuführen:

$$\begin{array}{c} \begin{array}{c} x_1 \quad x_3 \\ \begin{array}{|c|c|c|} \hline 8 & 3 & 4 \\ \hline -2 & -1 & -1 \\ \hline \end{array} \\ x_2 \end{array} \quad T(2) : \quad \begin{array}{c} x_1 \quad \bar{x}_3 \\ \begin{array}{|c|c|c|} \hline -12 & 3 & -4 \\ \hline 3 & -1 & 1 \\ \hline \end{array} \\ x_2 \end{array} \end{array}$$

Nun ist mit der ersten Spalte eine Pivotoperation und anschließend $T(1)$ durchzuführen:

$$\begin{array}{c} \begin{array}{c} x_2 \quad \bar{x}_3 \\ \begin{array}{|c|c|c|} \hline -3 & 3 & -1 \\ \hline -3 & -1 & -1 \\ \hline \end{array} \\ x_1 \end{array} \quad T(1) : \quad \begin{array}{c} \bar{x}_2 \quad \bar{x}_3 \\ \begin{array}{|c|c|c|} \hline -15 & -3 & -1 \\ \hline 1 & 1 & -1 \\ \hline \end{array} \\ x_1 \end{array} \end{array}$$

Damit lautet die Optimallösung

$$x_1 = 1, \quad \bar{x}_2 = 0,$$

also

$$x_1 = 1, \quad x_2 = 4, \quad z = 15.$$

Eine weitere mögliche Vorgangsweise zur Behandlung beschränkter Variablen wird in Abschnitt 16 beschrieben.

8.3 Die Behandlung von Variablen ohne Vorzeichenbeschränkung

Gegeben sei das lineare Programm

$$\max\{c'x \mid Ax \leq b, \quad x_j \geq 0 \text{ für } 1 \leq j \leq k, \quad k < n\}.$$

Ist $c_{k+1} \neq 0$, so nimmt die Zielfunktion zu bei Vergrößerung von x_{k+1} , falls $c_{k+1} > 0$ ist, und sie nimmt zu bei Verkleinerung von x_{k+1} , falls $c_{k+1} < 0$ ist. Eine Basislösung ist daher *optimal*, wenn für alle Nichtbasisvariablen ohne Vorzeichenbeschränkung die relativen Kostenkoeffizienten 0 und für alle vorzeichenbeschränkten Nichtbasisvariablen die relativen Kostenkoeffizienten ≤ 0 sind. Als Austauschspalte kann einerseits jede Spalte mit $c_j > 0$ genommen werden. Es kann aber auch eine Spalte mit $c_j < 0$ gewählt werden, sofern die zugehörige Nichtbasisvariable keinen Vorzeichenbeschränkungen unterliegt.

Nehmen wir nun an, die s -te Spalte mit $c_s < 0$ werde als Pivotspalte gewählt. Dann lautet das Restriktionensystem, wenn wir mit x_{n+1}, \dots, x_{n+m} die (augenblicklichen) Basisvariablen bezeichnen, und x_j für $j = 1, 2, \dots, n; j \neq s$ gleich 0 setzen:

$$x_{n+i} = b_i - a_{is}x_s \quad (1 \leq i \leq m)$$

Aus der Vorzeichenbeschränktheit der Basisvariablen x_{n+i} ($1 \leq i \leq m$) ergeben sich nun folgende Bedingungen für den Wert von x_s :

$$x_s \leq \frac{b_i}{a_{is}}, \text{ falls } a_{is} > 0$$

und

$$x_s \geq \frac{b_i}{a_{is}}, \text{ falls } a_{is} < 0.$$

Da x_s negativ gewählt werden soll, ist die erste Bedingung keine Einschränkung für x_s . Wohl aber muss die zweite Bedingung bei der Auswahl der Pivotzeile berücksichtigt werden. Man muss daher die Zeilenauswahlregel folgenderweise modifizieren:

Ist $c_s > 0$, so wähle man als Pivotzeile eine Zeile r , für die gilt

$$\frac{b_r}{a_{rs}} = \min \left\{ \frac{b_i}{a_{is}} \mid a_{is} > 0, x_{B(i)} \text{ vorzeichenbeschränkt} \right\}.$$

Ist $c_s < 0$, so wähle man als Pivotzeile eine Zeile r , für die gilt

$$\frac{b_r}{|a_{rs}|} = \min \left\{ \frac{b_i}{|a_{is}|} \mid a_{is} < 0, x_{B(i)} \text{ vorzeichenbeschränkt} \right\}.$$

Sind die Mengen $\{i \mid a_{is} > 0, x_{B(i)} \text{ vorzeichenbeschränkt}\}$ im Falle $c_s > 0$ bzw. $\{i \mid a_{is} < 0, x_{B(i)} \text{ vorzeichenbeschränkt}\}$ im Falle $c_s < 0$ leer, so besitzt die gestellte Optimierungsaufgabe keine endliche Optimallösung. Man terminiert.

Andernfalls führt man mit a_{rs} eine Pivotoperation durch. Zusammengefasst führen diese Überlegungen auf folgenden Algorithmus:

Algorithmus zur Lösung des linearen Programms

$$\max\{c'x \mid Ax \leq b, x_j \geq 0 \text{ für } j \in K, b \geq 0.\}$$

Ausgangsdaten: Tableau a_{ij} ($0 \leq i \leq m, 0 \leq j \leq n$), Vektoren N und B enthalten Indizes der Nichtbasis- und Basisvariablen.

1. Ist $a_{0j} \leq 0$ für $N(j) \in K$ und $a_{0j} = 0$ für $N(j) \notin K$, so ist die Basislösung optimal. Setze

$$x_{B(i)} := b_i \quad (1 \leq i \leq m) \quad (8.3)$$

$$x_{N(j)} := 0 \quad (1 \leq j \leq n) \quad (8.4)$$

und terminiere.

Andernfalls gehe zu Schritt 2.

2. Wähle s so, dass

$$|a_{0s}| = \max \left(\max\{a_{0j} \mid N(j) \in K\}, \max\{|a_{0j}| \mid N(j) \notin K\} \right).$$

Ist $a_{0s} > 0$, so gehe zu 3, sonst gehe zu 4.

3. Ist $I := \{i \mid a_{is} > 0, B(i) \in K\} = \emptyset$, so gehe zu 5, andernfalls zu 6.

4. Ist $I := \{i \mid a_{is} < 0, B(i) \in K\} = \emptyset$, so gehe zu 5, andernfalls zu 6.

5. Es existiert keine endliche Optimallösung. Terminiere.

6. Zeilenauswahlregel:

Bestimme r so, dass

$$\frac{a_{r0}}{|a_{rs}|} = \min \left\{ \frac{a_{i0}}{|a_{is}|} \mid i \in I \right\}$$

und gehe zu 7.

7. Führe mit a_{rs} eine Pivotoperation aus, und gehe zu 1.

Beispiel 8.3

$$\begin{array}{ll} \text{Maximiere} & x_1 + 2x_2 \\ \text{unter} & x_1 - x_2 \leq 2 \\ & x_1 + x_2 \leq 1 \\ & x_1, x_2 \text{ beliebig} \end{array}$$

Ausgangstableau

		x_1	x_2	
	0	1	2	$K = \{3, 4\}$
x_3	2	1	-1	
x_4	1	1	1	

Tableau 1

		x_1	x_4
	-2	-1	-2
x_3	3	2	1
x_2	1	1	1

Nun wird die erste Spalte Pivotspalte, jedoch ist $I = \emptyset$. Daher gibt es keine endliche Optimallösung.

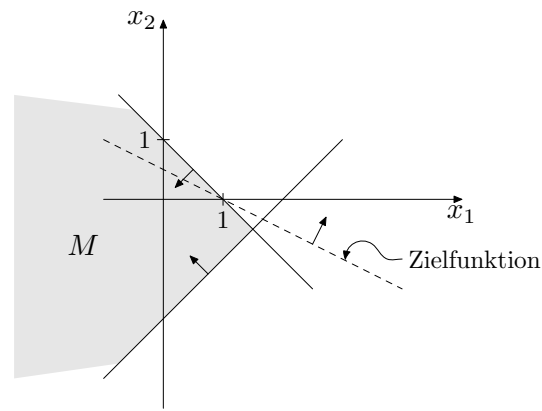


Abbildung 8.2: Menge der zulässigen Punkte für das Beispiel 8.3

8.4 Eine Anwendung: Lineares Ausgleichsproblem

Gegeben seien $n + 1$ linear unabhängige Vektoren $a_0, a_1, \dots, a_n \in \mathbb{R}^m$ ($m > n$). Es werde ein $x \in \mathbb{R}^n$ gesucht, so dass

$$\|a_0 - \sum_{j=1}^n a_j x_j\|_\infty \quad (8.5)$$

minimal wird. Dieses Problem tritt etwa auf, wenn die *wahrscheinlichste* Lösung x eines überbestimmten Gleichungssystems

$$\sum_{j=1}^n a_{ij} x_j = a_{i0} \quad (i = 1, \dots, m)$$

gefunden werden soll. Dabei wird der Ausdruck *wahrscheinlichst* so interpretiert, dass das Maximum der Absolutbeträge

$$|a_{i0} - \sum_{j=1}^n a_{ij} x_j| \quad (i = 1, \dots, m)$$

minimal wird. Somit ist das Problem (8.5) äquivalent zu folgendem linearen Programm:

Minimiere x_0 unter den Restriktionen

$$x_0 + \sum_{j=1}^n a_{ij} x_j \geq a_{i0} \quad (i = 1, \dots, m)$$

$$x_0 - \sum_{j=1}^n a_{ij} x_j \geq -a_{i0} \quad (i = 1, \dots, m) \quad (8.6)$$

$$x_j \in \mathbb{R}$$

Da die Lösung

$$x_0 := \max_{1 \leq i \leq m} |a_{i0}|$$

$$x_j := 0 \quad (1 \leq j \leq n)$$

für obiges LP zulässig ist, besitzt das lineare Programm (8.6) zulässige Punkte. Addiert man für ein festes i zwei der obigen Ungleichungen, so folgt $x_0 \geq 0$. Daher ist die Zielfunktion nach unten beschränkt und folglich besitzt (8.6) eine endliche Optimallösung. Da a_0 von den Vektoren a_1, \dots, a_n linear unabhängig war, ist der Optimalwert von x_0 positiv. Daher kann man eine Koordinatentransformation

$$y_0 := \frac{1}{x_0}, \quad y_j := \frac{-x_j}{x_0} \quad (1 \leq j \leq n)$$

durchführen, die (8.6) in das folgende LP überführt

Maximiere y_0 unter

$$\sum_{j=0}^n a_{ij} y_j \leq 1 \quad (1 \leq i \leq m)$$

$$-\sum_{j=0}^n a_{ij} y_j \leq 1 \quad (1 \leq i \leq m) \tag{8.7}$$

$$y_0 \geq 0, \quad y_j \in \mathbb{R} \quad (1 \leq j \leq n).$$

Führt man nun die Schlupfvariablen z_i ein durch

$$z_i := 1 - \sum_{j=0}^n a_{ij} y_j \quad (1 \leq i \leq m)$$

so wird (8.7) äquivalent zu

Maximiere y_0 unter

$$\sum_{j=0}^n a_{ij} y_j + z_i = 1 \quad (1 \leq i \leq m)$$

$$0 \leq z_i \leq 2 \quad (1 \leq i \leq m) \tag{8.8}$$

$$y_0 \geq 0, \quad y_j \in \mathbb{R} \quad (1 \leq j \leq n)$$

Dieses lineare Programm (8.8) kann nun vorteilhaft mit den Methoden der Abschnitte 8.1 – 8.3 gelöst werden. Man wählt anfangs z_1, \dots, z_m als Basisvariable und kombiniert nun die Schritte des Simplexverfahrens für nicht vorzeichenbeschränkte Variablen mit den Regeln des Simplexverfahrens für Variablen mit oberen Schranken. Dabei kann der Fall eintreten, dass der reduzierte Koeffizient \tilde{c}_s einer nicht vorzeichenbeschränkten Variablen $y_{N(s)}$ negativ ist und in der Spalte s nur Elemente $\tilde{a}_{is} \geq 0$ stehen. In diesem Fall ist das Minimum der Größen

$$\frac{\tilde{a}_{i0} - d_{B(i)}}{\tilde{a}_{is}} \quad \text{für } \tilde{a}_{is} > 0 \tag{8.9}$$

zu bestimmen, wobei die zur Zeile i gehörende Basisvariable durch $d_{B(i)}$ beschränkt ist. Durch das Minimum dieser Ausdrücke (8.9) wird die Pivotzeile r festgelegt. Dann wird mit \tilde{a}_{rs} eine Pivotoperation durchgeführt. Anschließend wird die neue Nichtbasisvariable durch ihre Komplementärvariable ersetzt, d.h. es wird eine Transformation $T(s)$ durchgeführt. Im Falle eines positiven reduzierten Kostenkoeffizienten \tilde{c}_s und Spaltenelementen $\tilde{a}_{is} \leq 0$ wird analog vorgegangen.

Beispiel 8.4 Man bestimme eine möglichst gute Näherungslösung des Gleichungssystems

$$\begin{aligned} x_1 &= 0 \\ x_2 &= 0 \\ x_1 + x_2 &= 4 \end{aligned}$$

Das zugehörige lineare Programm (8.8) lautet:

Maximiere y_0 unter den Restriktionen

$$\begin{aligned} y_1 + z_1 &= 1 \\ y_2 + z_2 &= 1 \\ 4y_0 + y_1 + y_2 + z_3 &= 1 \\ 0 \leq z_i &\leq 2 \quad (i = 1, 2, 3) \\ y_0 &\geq 0 \\ y_1 &\in \mathbb{R} \\ y_2 &\in \mathbb{R} \end{aligned}$$

Ausgehend vom Tableau

	y_0	y_1	y_2
	0	1	0
z_1	1	0	1
z_2	1	0	0
z_3	1	4	1

führt man zunächst mit $a_{31} = 4$ eine Pivotoperation durch, und erhält

	z_3	y_1	y_2
	$-\frac{1}{4}$	$-\frac{1}{4}$	$-\frac{1}{4}$
z_1	1	0	1
z_2	1	0	0
y_0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

Nun wählt man etwa $s = 2$, und erhält aufgrund der obigen Bemerkungen $r = 1$, da $z_1 \leq 2$. Somit führt man zuerst mit a_{12} eine Pivotoperation durch und ersetzt anschließend z_1 durch $\bar{z}_1 = 2 - z_1$:

$$\begin{array}{c}
 \begin{array}{c|ccc}
 & z_3 & z_1 & y_2 \\
 \hline
 0 & -\frac{1}{4} & \frac{1}{4} & -\frac{1}{4} \\
 \hline
 y_1 & 1 & 0 & 1 & 0 \\
 z_2 & 1 & 0 & 0 & 1 \\
 y_0 & 0 & \frac{1}{4} & -\frac{1}{4} & \frac{1}{4}
 \end{array}
 \longrightarrow
 \begin{array}{c|ccc}
 & z_3 & \bar{z}_1 & y_2 \\
 \hline
 -\frac{1}{2} & -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\
 \hline
 y_1 & -1 & 0 & -1 & 0 \\
 z_2 & 1 & 0 & 0 & 1 \\
 y_0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4}
 \end{array}
 \end{array}$$

Wie oben erhält man nun $s = 3, r = 2$, und muss nach der Pivotation wieder z_2 durch die Komplementärvariable \bar{z}_2 ersetzen:

$$\begin{array}{c}
 \begin{array}{c|ccc}
 & z_3 & \bar{z}_1 & z_2 \\
 \hline
 -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} & \frac{1}{4} \\
 \hline
 y_1 & -1 & 0 & -1 & 0 \\
 y_2 & 1 & 0 & 0 & 1 \\
 y_0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4}
 \end{array}
 \longrightarrow
 \begin{array}{c|ccc}
 & z_3 & \bar{z}_1 & \bar{z}_2 \\
 \hline
 -\frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\
 \hline
 y_1 & -1 & 0 & -1 & 0 \\
 y_2 & -1 & 0 & 0 & -1 \\
 y_0 & \frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4}
 \end{array}
 \end{array}$$

Damit ist die Optimallösung erreicht. Sie lautet $y_1 = y_2 = -1$, $y_0 = \frac{3}{4}$. Die Rücktransformation auf die Variablen x_0, x_1, x_2 liefert

$$x_0 = \frac{4}{3}, \quad x_1 = x_2 = \frac{4}{3}.$$

Kapitel 9

Simplexinterpretation des Simplexverfahrens

Dantzig [14] hat anhand eines Mischungsproblems aufgezeigt, woher der Simplexalgorithmus seinen Namen hat. In Anlehnung an seine Ausf[hrungen] werden wir zeigen, dass dem Übergang von einer Basislösung zur nächsten, geometrisch ein Simplex entspricht. Aus dieser geometrischen Interpretation leitet sich der Name des Verfahrens ab.

Gegeben sei das Mischungsproblem: Minimiere z mit

$$\begin{aligned}c_1x_1 + \dots + c_nx_n &= z \\a_1x_1 + \dots + a_nx_n &= b \\x_1 + \dots + x_n &= 1 \\x_i &\geq 0 \quad (i = 1, \dots, n).\end{aligned}$$

Wir fassen die Spalten $(c_j, a_j)'$ ($j = 1, \dots, n$) als Koordinaten von Punkten P_j in der (z, v) -Ebene auf. Dann lässt sich obiges Ungleichungssystem schreiben als

$$\sum_{j=1}^n P_j x_j = \begin{pmatrix} z \\ b \end{pmatrix}, \quad \sum_{j=1}^n x_j = 1, \quad x_j \geq 0 \quad (1 \leq j \leq n),$$

d.h. zulässig sind die Konvexkombinationen der Punkte P_j , die auf der Geraden $v = b$ liegen. Die Optimallösung liefert jener Punkt auf der Geraden $v = b$, der die kleinste Abszisse hat (vgl. Abbildung 9.1).

Wie bestimmt nun das Simplexverfahren diese Optimallösung? Eine zulässige Basislösung entspricht einem Punktepaar P_i, P_j , so dass die Strecke $\overline{P_i P_j}$ die Gerade $v = b$ schneidet. Denn nimmt man x_i und x_j als Basisvariable, und setzt man $x_i = \lambda$, so folgt $x_j = 1 - \lambda$, und wegen $x_N = 0$ weiter $\lambda a_i + (1 - \lambda) a_j = b$. Die Umkehrung folgt analog. In Abbildung 9.1 wäre etwa durch P_6, P_7 eine zulässige Basislösung bestimmt.

Um festzustellen, ob diese Basislösung optimal ist, berechnet man die Gerade durch P_i, P_j und bestimmt sodann den Abstand der Punkte P_k ($k = 1, 2, \dots, n$) von dieser Geraden in z -Richtung. Dieser Abstand entspricht genau den relativen Kostenkoeffizienten: Durch Multiplikation der Zeilen mit k_0 und k_1 und anschließender Subtraktion eliminiere man die Basisvariablen aus der Zielfunktion

$$\left. \begin{array}{l}c_1x_1 + \dots + c_nx_n = z \\a_1x_1 + \dots + a_nx_n = b \\x_1 + \dots + x_n = 1\end{array} \right\} \begin{array}{l} \cdot k_0 \\ \cdot k_1 \end{array} \left. \right\} + \left. \right\} -$$

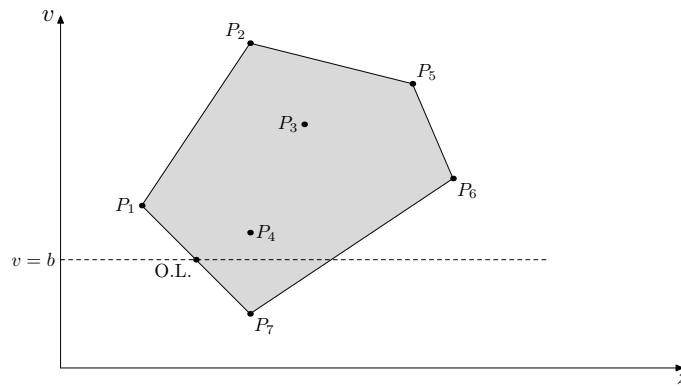


Abbildung 9.1: Beispiel zur geometrischen Interpretation eines Mischungsmodells mit $n = 7$

Daraus ergeben sich für k_0, k_1 die Bestimmungsgleichungen

$$\begin{aligned} c_i &= k_0 a_i + k_1 \\ c_j &= k_0 a_j + k_1 \end{aligned}$$

d.h. P_i, P_j liegen auf der Geraden $z = k_0 v + k_1$. Schneidet man die Gerade $v = a_k$ mit dieser Geraden, so erhält man als horizontalen Abstand des Punktes P_k von der Geraden $z = k_0 v + k_1$ den Wert

$$c_k - (k_0 a_k + k_1) = \tilde{c}_k.$$

Dies ist aber gerade der reduzierte Kostenkoeffizient \tilde{c}_k . Beim betrachteten Minimierungsproblem ist die Basislösung optimal, wenn $\tilde{c}_N \geq 0$ gilt, d.h. alle Punkte liegen *rechts* von der Geraden durch die Punkte P_i und P_j . Anderenfalls wähle man ein P_k mit $\tilde{c}_k < 0$. Meist wird ein Punkt P_k gewählt, für den \tilde{c}_k minimal ist (Spaltenauswahlregel (7.1)). Durch P_i, P_j, P_k wird nun ein Simplex erzeugt. Zwei Kanten dieses Simplexes werden von der Geraden $v = b$ geschnitten, eine Kante entspricht der alten Basislösung, die andere der neuen Basislösung (vgl. Abbildung 9.2).

Die hier gegebene Interpretation des Mischungsproblems ist ein Spezialfall einer zweiten möglichen geometrischen Deutung von linearen Optimierungsproblemen. Gegeben sei das lineare Programm

$$\max\{c'x \mid Ax = b, x \geq 0\}$$

mit einer $m \times n$ -Matrix A ($m < n$). Wir führen einen Vektor $u = (u_0, u_1, \dots, u_m)'$ ein durch

$$\begin{aligned} u_0 &:= \sum_{j=1}^n c_j x_j \\ u_i &:= \sum_{j=1}^n a_{ij} x_j \quad (1 \leq i \leq m) \end{aligned}$$

Dann ist

$$U = \left\{ u \mid u = \begin{pmatrix} c' \\ A \end{pmatrix} x, x \geq 0 \right\}$$

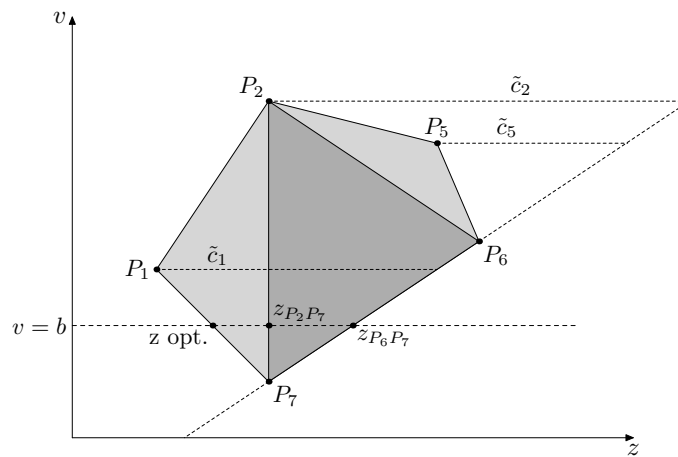


Abbildung 9.2: Basisaustausch ausgehend von Basislösung, bestimmt durch P_6, P_7 . Es ist $\tilde{c}_2 = \min\{\tilde{c}_1, \tilde{c}_2, \tilde{c}_5\}$. Basisaustausch durch Simplex $P_2P_6P_7$. Neue Basislösung: P_2, P_7 . Ein weiterer Simplexschritt (P_1, P_2, P_7) ist notwendig, bis Optimallösung erreicht ist, die durch P_1, P_7 erhalten wird.

das Bild des positiven Orthanten unter der linearen Abbildung, die durch die Matrix

$$\begin{pmatrix} c' \\ A \end{pmatrix}$$

beschrieben wird. Somit ist U ein polyedrischer Kegel im \mathbb{R}^{m+1} . Ferner entspricht der Menge $\{(c'x, b_1, \dots, b_m) \mid x \in \mathbb{R}^n\}$ eine Gerade im \mathbb{R}^{m+1} , parallel zur u_0 -Achse. Schneidet diese Gerade den Kegel nicht, dann gibt es keine zulässige Lösung. Andernfalls ist der Durchschnitt entweder eine Strecke oder ein Halbstrahl, was den beiden anderen Möglichkeiten bei linearen Programmen entspricht. Die Interpretation des Mischungsproblems erhält man, indem man den Kegel mit der Ebene $u_2 = 1$ schneidet.

Kapitel 10

Das revidierte Simplexverfahren

Bei der bisher betrachteten Version des Simplexverfahrens wurden die Spalten der Koeffizientenmatrix transformiert, die zu den Nichtbasisvariablen gehören. Man kann fragen, ob wirklich alle diese Elemente stets betrachtet werden müssen, oder ob man nicht mit weniger Rechenschritten und Speicherbedarf ebenfalls zum Ziel kommt. Dies ist tatsächlich der Fall im sogenannten *revidierten Simplexverfahren*, in dem nicht jeweils die ganze Koeffizientenmatrix transformiert wird, sondern nur die jeweilige Basisinverse berechnet und gespeichert wird. Zusammen mit den gegebenen Anfangsdaten lässt sich dann rasch die ganze Information gewinnen, die zur Durchführung des nächsten Schrittes notwendig ist.

Um die notwendigen Informationen zur Durchführung einer Iteration im Simplexverfahren zu erhalten, ist es notwendig, drei lineare Gleichungssysteme zu lösen: Die augenblickliche Basislösung $x_B = \tilde{b}$ erhält man als Lösung von $A_B x_B = b$. Um die reduzierten Kostenkoeffizienten zu berechnen, ist

$$\check{c}'_N = c'_N - c'_B A_B^{-1} A_N$$

zu bestimmen. Setzt man

$$\pi := A_B^{-T} c_B, \tag{10.1}$$

so erhält man

$$\tilde{c}_N = c_N - A'_N \pi$$

durch Multiplikation der Matrix A'_N mit dem Vektor π . Nach (10.1) ist π' Lösung des linearen Gleichungssystems

$$\pi' A_B = c'_B.$$

Zum Nachweis, dass keine endliche Lösung existiert bzw. zur Bestimmung der Pivotzeile ist die Kenntnis von

$$\tilde{a}_s := A_B^{-1} a_s$$

notwendig, wobei a_s der Vektor der Pivotspalte ist. \tilde{a}_s lässt sich wieder durch das Lösen des linearen Gleichungssystems

$$A_B \tilde{a}_s = a_s$$

Also ergeben sich die neuen Elemente \bar{d}_{ij} von \bar{D}^{-1} aus den Elementen d_{ij} von D^{-1} durch folgende Formeln:

$$\begin{aligned}\bar{d}_{ij} &:= d_{ij} - \frac{\tilde{a}_{is}d_{rj}}{\tilde{a}_{rs}} & 0 \leq i \leq m, i \neq r, 0 \leq j \leq m \\ \bar{d}_{rj} &:= \frac{d_{rj}}{\tilde{a}_{rs}} & 0 \leq j \leq m, i = r\end{aligned}\tag{10.6}$$

Das Lösen der drei Gleichungssysteme erfordert daher im Rahmen des Simplexverfahrens lediglich $O(m^2)$ arithmetische Operationen. Würden die Gleichungssysteme hingegen isoliert vorliegen, so würden $O(m^3)$ arithmetische Operationen zu ihrer Lösung benötigt werden.

Im revidierten Simplexverfahren wird also anstelle der reduzierten Matrix \tilde{A}_N jeweils A_B^{-1} berechnet. Da die Ausgangsbasis bei einem zulässigen Problem ($b \geq 0$) stets $A_B = E$ ist, beginnt man mit $A_B^{-1} = E$, und führt folgenden Algorithmus durch:

Revidiertes Simplexverfahren mit expliziter Inverser zur Lösung von

$$\max\{c'x \mid Ax \leq b, x \geq 0\} \text{ bei } b \geq 0.$$

Ausgangsdaten:

$$(a_{ij})_{\substack{0 \leq i \leq m \\ 1 \leq j \leq n+m}} := \begin{pmatrix} c' & 0 \\ A & E \end{pmatrix}$$

$$c_j := \begin{cases} c_j & 1 \leq j \leq n, \\ 0 & n+1 \leq j \leq n+m \end{cases}$$

$$B := (n+1, \dots, n+m), \quad N := \{1, \dots, n\}$$

$$d_{ij} := \begin{cases} 1 & i = j, \quad i = 0, \dots, m; \quad j = 1, \dots, m \\ 0 & i \neq j, \quad i = 0, \dots, m; \quad j = 1, \dots, m \end{cases}$$

$$d_{i,m+1} := \begin{cases} 0 & i = 0 \\ b_i & 1 \leq i \leq m \end{cases}$$

1. Berechne für $k \in N$:

$$\tilde{c}_k := c_k + \sum_{j=1}^m d_{0j}a_{jk}.$$

2. Ist für alle $k \in N$: $\tilde{c}_k \leq 0$, so ist die augenblickliche Basislösung

$$x_k := 0 \quad (k \in N), \quad x_{B(i)} := d_{i,m+1} \quad (1 \leq i \leq m)$$

optimal und hat den Zielfunktionswert $-d_{0,m+1}$. Stop.

Andernfalls gehe zu 3.

3. Wähle s so, dass $\tilde{c}_s = \max_{k \in N} \tilde{c}_k$.
4. Berechne $\tilde{a}_{is} = \sum_{j=1}^m d_{ij} a_{js}$ ($1 \leq i \leq m$) und setze $\tilde{a}_{0s} = \tilde{c}_s$.
5. Ist für alle i mit $1 \leq i \leq m$ die Zahl $\tilde{a}_{is} \leq 0$, so gibt es keine Optimallösung.
Stop.
Andernfalls gehe zu 6.
6. Wähle r so, dass

$$\frac{d_{r,m+1}}{\tilde{a}_{rs}} = \min \left\{ \frac{d_{i,m+1}}{\tilde{a}_{is}} \mid \tilde{a}_{is} > 0 \right\}.$$

7. Führe die Pivotoperation (10.6) durch:

$$\begin{aligned} \bar{d}_{ij} &:= d_{ij} - \frac{\tilde{a}_{is} d_{rj}}{\tilde{a}_{rs}} & 0 \leq i \leq m, \quad i \neq r, \quad 1 \leq j \leq m+1 \\ \bar{d}_{rj} &:= \frac{d_{rj}}{\tilde{a}_{rs}} & i = r, \quad 0 \leq j \leq m+1. \end{aligned}$$

8. Setze

$$\begin{aligned} B(r) &:= s \\ N &:= \{1, \dots, n+m\} \setminus \{B(i) \mid 1 \leq i \leq m\} \\ d_{ij} &:= \bar{d}_{ij} \quad \text{für alle } i, j \end{aligned}$$

und gehe zu 1.

Das revidierte Simplexverfahren ist besonders dann vorteilhaft, wenn die Optimierungsaufgabe wesentlich mehr Variablen als Restriktionen hat. Im Fall des revidierten Simplexverfahren werden nämlich nicht die $m \times n$ Einträge der Matrix A_N transformiert, sondern nur die $(m \times m)$ -Basisinverse A_B^{-1} .

Beispiel 10.1 Wir betrachten die Mischungsaufgabe

$$\begin{aligned} \text{Minimiere} \quad & x_1 + 3x_2 + 6x_3 + 7x_4 + 5x_5 + 2x_6 \\ \text{unter} \quad & 4x_1 + 6x_2 + 5x_3 + 3x_4 + x_5 + 2x_6 = \frac{7}{2} \\ & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 1 \\ & x_j \geq 0 \quad \text{für } j = 1, 2, \dots, 6. \end{aligned}$$

Die Ausgangsdaten lauten

x_1	x_2	x_3	x_4	x_5	x_6	b
1	3	6	7	5	2	0
4	6	5	3	1	2	$\frac{7}{2}$
1	1	1	1	1	1	1

Wählt man x_3 und x_4 als erste Basisvariable, so erhält man als erste (erweiterte) Basismatrix

$$D = \begin{pmatrix} 1 & 6 & 7 \\ 0 & 5 & 3 \\ 0 & 1 & 1 \end{pmatrix}$$

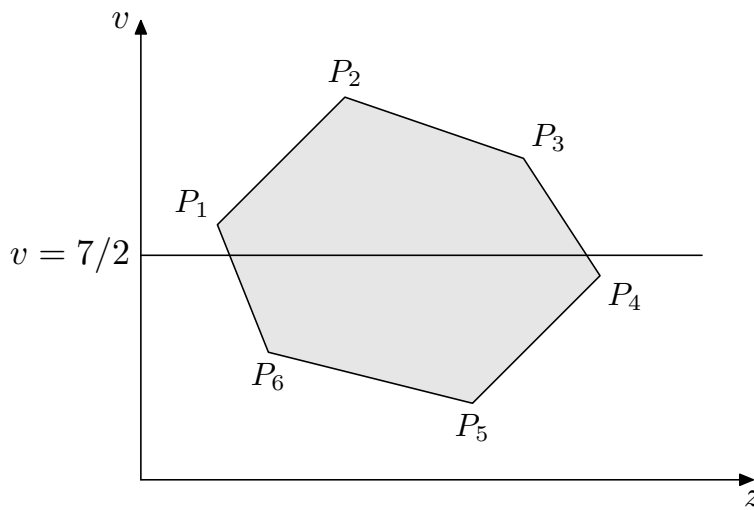


Abbildung 10.1: Graphische Interpretation der Mischungsaufgabe von Beispiel 10.1 im Sinne des Abschnitts 9

Damit erhält man als erste Basisinverse D^{-1} , sowie für $\tilde{b} = D^{-1}b$:

z	x_3	x_4	\tilde{b}	\tilde{a}_1
1	$\frac{1}{2}$	$-\frac{17}{2}$	$-\frac{27}{4}$	$-\frac{11}{2}$
0	$\frac{1}{2}$	$-\frac{3}{2}$	$\frac{1}{4}$	$\frac{1}{2}$
0	$-\frac{1}{2}$	$\frac{5}{2}$	$\frac{3}{4}$	$\frac{1}{2}$

Als reduzierte Kostenkoeffizienten ergeben sich nach Punkt 1 des Algorithmus:

$$\tilde{c}_1 = -\frac{11}{2}, \quad \tilde{c}_2 = -\frac{5}{2}, \quad \tilde{c}_5 = -3, \quad \tilde{c}_6 = -\frac{11}{2}.$$

Da wir die Zielfunktion minimieren, wählen wir die Pivotspalte durch einen minimalen reduzierten Kostenkoeffizienten. Somit können wir die zu x_1 gehörende Spalte als Pivotspalte wählen. Damit werden die kursiv gesetzten zur Spalte \tilde{a}_1 gehörenden Daten zu obiger Tabelle hinzugefügt. Die Zeilenauswahlregel ergibt, dass x_3 gegen x_1 ausgetauscht wird. Die entsprechende Pivotoperation liefert

z	x_1	x_4	\tilde{b}	\tilde{a}_5
1	6	-25	-4	-14
0	1	-3	$\frac{1}{2}$	-2
0	-1	4	$\frac{1}{2}$	3

Als reduzierte Kostenkoeffizienten erhält man

$$\tilde{c}_2 = 14, \tilde{c}_3 = 11, \tilde{c}_5 = -14, \tilde{c}_6 = -11.$$

Somit wird x_5 in die Basis aufgenommen und man fügt die transformierte (kursiv gesetzte) Spalte \tilde{a}_5 zu obiger Tabelle hinzu. Als Pivotelement ergibt sich \tilde{a}_{52} , d.h. es wird x_4 gegen x_5 ausgetauscht. Die entsprechende Pivotoperation liefert

z	x_1	x_5	\tilde{b}	\tilde{a}_6
1	$\frac{4}{3}$	$-\frac{19}{3}$	$-\frac{5}{3}$	$-\frac{5}{3}$
0	$\frac{1}{3}$	$-\frac{1}{3}$	$\frac{5}{6}$	$\frac{1}{3}$
0	$-\frac{1}{3}$	$\frac{4}{3}$	$\frac{1}{6}$	$\frac{2}{3}$

Als reduzierte Kostenkoeffizienten erhält man

$$\tilde{c}_2 = \frac{14}{3}, \tilde{c}_3 = \frac{19}{3}, \tilde{c}_4 = \frac{14}{3}, \tilde{c}_6 = -\frac{5}{3}.$$

Daher wird x_6 in die Basis aufgenommen und man fügt die kursiv gesetzte Spalte zu obigen Tableau hinzu. Das Pivotelement wird \tilde{a}_{26} , also wird x_5 gegen x_6 ausgetauscht und man erhält

z	x_1	x_6	\tilde{b}
1	$\frac{1}{2}$	-3	$-\frac{5}{4}$
0	$\frac{1}{2}$	-1	$\frac{3}{4}$
0	$-\frac{1}{2}$	2	$\frac{1}{4}$

Nun ist $\tilde{c}_2 = 3$, $\tilde{c}_3 = \frac{11}{2}$, $\tilde{c}_4 = \frac{11}{2}$, $\tilde{c}_5 = \frac{r}{2}$. Also ist die Optimallösung

$$x_1 = \frac{3}{4}, x_2 = x_3 = x_4 = x_5 = 0, x_6 = \frac{1}{4}.$$

Die Abbildung 10.2 zeigt den schrittweisen Austausch der Simplexes nach der Interpretation des vorigen Abschnittes.

Die nachfolgende **Rundungsfehleranalyse** des revidierten Simplexverfahrens zeigt, dass die Rundungsfehler beliebig groß werden können. Das revidierte Simplexverfahren ist also nicht numerisch stabil. Dies bestätigen auch die numerischen Erfahrungen, denn bereits bei Beispielen mit ca. $m = 50$ können Rundungsfehler die Lösung sehr stark beeinflussen und verfälschen.

Bezeichnen wir die Gleitkommaoperationen für Addition, Subtraktion, Multiplikation und Division mit $gl(a * b)$, wobei der Operator $*$ für $+$, $-$, \cdot , $:$ steht, so gilt

$$\begin{aligned} gl(a \pm b) &= a(1 + \varepsilon_1) \pm b(1 + \varepsilon_2) \\ gl(a * b) &= (a * b)(1 + \varepsilon_*) \quad \text{für } * \in \{+, \cdot, : \}. \end{aligned}$$

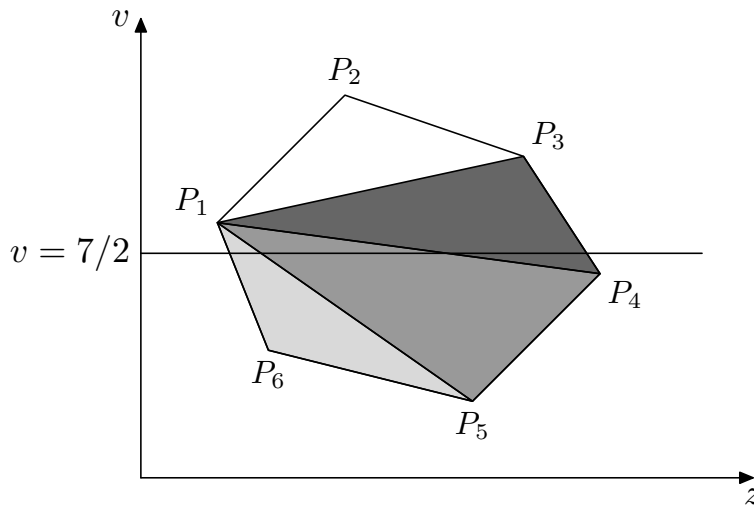


Abbildung 10.2: Ausgehend von der Basislösung P_3, P_4 wird P_3 gegen P_1 ausgetauscht, dann P_4 gegen P_5 und schließlich P_5 gegen P_6

Ferner können wir annehmen, dass die nichtnegativen Größen

$$\varepsilon_1, \varepsilon_2, \varepsilon_3 = \varepsilon., \varepsilon_4 = \varepsilon.$$

durch die Maschinengenauigkeit ε nach oben beschränkt sind. Soll nun der Vektor $u = (u_1, \dots, u_m)$ durch eine Pivotoperation mit dem Vektor $v = (v_1, \dots, v_m)$ und Pivotelement v_s in den Vektor $w = (w_1, \dots, w_m)$ übergeführt werden, so gilt

$$w_s = gl\left(\frac{u_s}{v_s}\right) = \frac{u_s}{v_s}(1 + \eta)$$

$$w_i = gl\left(u_i - \frac{v_i u_s}{v_s}\right) = u_i(1 + \rho_i) - \frac{v_i u_s}{v_s}(1 + \eta)(1 + \sigma_i)(1 + \tau_i) \text{ für } i \neq s.$$

Setzen wir

$$\delta u_i := \begin{cases} u_i \eta & \text{für } i = s \\ u_i \rho_i - \frac{v_i u_s}{v_s}(\sigma_i + \tau_i + \sigma_i \tau_i + \eta \sigma_i + \eta \tau_i + \eta \sigma_i \tau_i) & \text{für } i \neq s \end{cases}$$

so erhalten wir

$$w_i := \begin{cases} \frac{1}{v_s}(u_i + \delta u_i) & \text{für } i = s \\ (u_i + \delta u_i) - \frac{v_i}{v_s}(u_s + \delta u_s) & \text{für } i \neq s. \end{cases}$$

Daher erhält man für die Pivotoperation $T = \left(E - \frac{1}{v_s}(v - e_s)e'_s\right)$ (vgl. Transformation (10.5)):

$$w = gl(Tu) = T(u + \delta u)$$

d.h. die Rundungsfehler wirken sich so aus, dass der im Gleitkomma berechnete Vektor w sich als exakte Transformation eines gestörten Vektors u darstellen lässt. Wenn keine Einschränkungen

für das Pivotelement v_s gemacht werden, können die Komponenten von δu nicht beschränkt werden. Es gilt

$$|\delta u_i| \leq |u_i| \varepsilon + \left| \frac{v_i}{v_s} \right| \cdot |u_s| (2\varepsilon + 3\varepsilon^2 + \varepsilon^3).$$

Ist $|v_s|$ relativ klein gegenüber $|v_i| \cdot |u_s|$, dann kann die rechte Seite der Ungleichung beliebig groß werden.

Diese Überlegungen waren der Ausgangspunkt für eine Stabilisierung des Simplexverfahrens, die im nächsten Abschnitt besprochen werden soll.

Eine andere Form des revidierten Simplexverfahrens erhält man, wenn man an Stelle der expliziten Speicherung der Basisinversen die Transformationen (10.5) speichert. Diese Transformationen lassen sich sehr kompakt speichern, da sie sich nur in einer Spalte von einer Einheitsmatrix unterscheiden. Speichert man diese Spalte sowie den zugehörigen Spaltenindex, so hat man die ganze benötigte Information. Dieses Vorgehen hat den weiteren Vorteil, dass *dünne Matrizen* (d.h. Matrizen mit nur wenig Elementen $\neq 0$) bei diesem Vorgehen erhalten bleiben, während beim Übergang zur Basisinversen sehr viele Nicht-Null Elemente auftreten können. Daher verwenden die meisten kommerziellen Codes ein *revidiertes Simplexverfahren in Produktform* mit speziellen Speichertechniken für dünne Matrizen.

Kapitel 11

Das Simplexverfahren mit LU -Zerlegung

Wie im vorigen Abschnitt gezeigt wurde, ist das Simplexverfahren anfällig gegenüber Rundungsfehlern. Bartels [4] gab eine Version an, die numerisch stabil ist und auf der LU -Zerlegung der Matrix A_B beruht. Ausgangspunkt sind die drei Gleichungssysteme (10.2)

$$A_B x_B = b, \quad \pi' A_B = c'_B, \quad A_B \tilde{a}_s = a_s,$$

wobei A_B eine quadratische nichtsinguläre Matrix ist. Jede quadratische nichtsinguläre Matrix lässt sich aber, gegebenenfalls nach Zeilen- und Spaltenvertauschungen, in eine obere und eine untere Dreiecksmatrix zerlegen. D.h. es gilt

$$PA_B = LU$$

mit einer Permutationsmatrix P , einer unteren Dreiecksmatrix

$$L = \begin{pmatrix} 1 & 0 & \dots & 0 \\ l_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ l_{m1} & \dots & l_{m,m-1} & 1 \end{pmatrix}$$

und einer oberen Dreiecksmatrix

$$U = \begin{pmatrix} u_{11} & \dots & \dots & u_{1m} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & u_{m,m} \end{pmatrix}.$$

Ist die Zerlegung von A_B in L und U bekannt, so kann jedes der Gleichungssysteme (10.2) auf zwei *gestaffelte Gleichungssysteme* zurückgeführt werden, die sich sehr leicht rekursiv lösen lassen. Gilt etwa $A_B x_B = b$ und $PA_B = LU$, so setzt man $Ux_B = v$ und erhält $Lv = Pb$. Die Lösung dieser Systeme lautet, wenn die Permutationsmatrix P der Permutation φ von $\{1, 2, \dots, m\}$ entspricht,

$$v_1 = b_{\varphi(1)}, \quad v_2 = b_{\varphi(2)} - l_{21}v_1, \dots$$

also

$$v_i = b_{\varphi(i)} - \sum_{k=1}^{i-1} l_{ik} v_k \quad (i = 1, 2, \dots, m)$$

und

$$x_i = \frac{1}{u_{ii}} \left(v_i - \sum_{k=i+1}^m u_{ik} x_k \right) \quad (i = m, m-1, \dots, 1).$$

Somit ist die Kenntnis von P, L und U ebensogut, wie die von A_B^{-1} . Numerisch gesehen bringt sie sogar noch Vorteile, so bleiben etwa gewisse Eigenschaften von A_B bei der LU -Zerlegung erhalten (Bandmatrizen, dünne Matrizen), die beim Übergang zur inversen Matrix verloren gehen. Da die Koeffizientenmatrix großer Systeme meist dünn ist, bietet die LU -Zerlegung daher auch erhebliche Speicherplatzvorteile.

Zu klären bleibt, wie sich die LU -Zerlegung der Basismatrix ändert, wenn man ein Element der Basis austauscht. Die Basismatrix vor dem Austausch sei

$$A_B = (a_1, \dots, a_m)$$

mit Spaltenvektoren a_1, a_2, \dots, a_m . In A_B werde nun die r -te Spalte gestrichen und an die letzte Stelle trete die neue Spalte a_s . Die neue Basismatrix hat daher die Form

$$A_{\bar{B}} = (a_1, \dots, a_{r-1}, a_{r+1}, \dots, a_m, a_s).$$

Wegen $LU = PA_B$ gilt nun

$$\begin{aligned} L^{-1}PA_{\bar{B}} &= (L^{-1}Pa_1, \dots, L^{-1}Pa_{r-1}, L^{-1}Pa_{r+1}, \dots, L^{-1}Pa_m, L^{-1}Pa_s) \\ &= (u_1, \dots, u_{r-1}, u_{r+1}, \dots, u_m, L^{-1}Pa_s) =: \bar{H} \end{aligned}$$

\bar{H} hat die Form

$$\begin{pmatrix} h_{11} & \dots & \dots & \dots & \dots & h_{1m} \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & 0 & h_{rr} & & \vdots \\ \vdots & & \vdots & h_{r+1,r} & & \vdots \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & \dots & h_{mm} \end{pmatrix}.$$

\bar{H} ist eine obere Hessenbergmatrix mit Nullelementen in der Subdiagonale der ersten $r-1$ Spalten. \bar{H} ist bekannt, da $L^{-1}Pa_s$ bei der Auswahl der Austauschzeile berechnet wird.

Durch Gauss-Elimination kann \bar{H} auf Dreiecksform gebracht werden. Zunächst führen wir dazu eine Zeilenvertauschung durch, falls

$$|h_{i+1,i}| > |h_{ii}|.$$

Dies entspricht einer Linksmultiplikation mit einer Permutationsmatrix P_i der Form

$$P_i = \begin{pmatrix} 1 & & & & 0 \\ & 1 & & & \\ & & 0 & 1 & \\ & & 1 & 0 & \\ & & & & 1 \\ 0 & & & & & 1 \end{pmatrix}. \quad (11.1)$$

Dabei ist $p_{ii} = p_{i+1,i+1} = 0$, $p_{i+1,i} = p_{i,i+1} = 1$. Durch diese Zeilenvertauschung erreicht man, dass entweder das neue Subdiagonalelement gleich 0 ist, oder dass das Diagonalelement h_{ii} der *neuen* Matrix einen größeren Betrag hat, als das Subdiagonalelement $h_{i+1,i}$. Multipliziert man im zweiten Fall die Matrix von links mit einer Matrix der Form

$$M_i = \begin{pmatrix} 1 & & 0 \\ & 1 & \\ & m_i & 1 \\ 0 & & & 1 \end{pmatrix} \quad (11.2)$$

wobei

$$m_i := -\frac{h_{i+1,i}}{h_{ii}}$$

ist, so verschwindet das Subdiagonalelement. (Man beachte, dass $h_{ii} \neq 0$ gilt!) Multipliziert man daher \bar{H} von links alternierend mit Matrizen P_i und M_i , wobei jedes P_i ($i = r, \dots, m-1$) entweder eine Einheitsmatrix ist oder die Form (11.1) hat, und jedes M_i ($i = r, \dots, m-1$) entweder eine Einheitsmatrix ist oder die Form (11.2) hat, so erhält man wieder eine obere Dreiecksmatrix \bar{U} :

$$\bar{U} = M_{m-1}P_{m-1} \dots M_r P_r \bar{H}. \quad (11.3)$$

Daraus erhält man \bar{H} zu

$$\bar{H} = P_r^{-1}M_r^{-1} \dots P_{m-1}^{-1}M_{m-1}^{-1}\bar{U}.$$

Die dabei auftretenden inversen Matrizen sind sehr leicht berechenbar, denn $P_i^{-1} = P_i$ und M_i^{-1} erhält man aus M_i , indem man das Vorzeichen des Subdiagonalelements m_i umkehrt. Setzt man

$$D := P_r M_r^{-1} \dots P_{m-1} M_{m-1}^{-1},$$

so erhält man folgende Zerlegung der neuen Basismatrix $A_{\bar{B}}$

$$PA_{\bar{B}} = L\bar{H} = LD\bar{U}. \quad (11.4)$$

LD ist nur dann wieder eine untere Dreiecksmatrix, wenn alle Permutationsmatrizen gleich der Einheitsmatrix sind.

Im Simplexverfahren werden nun iterativ Basisaustauschschritte durchgeführt. Ausgehend von einer Basis $B_{(0)}$ geht man zu $B_{(1)}$, dann zu $B_{(2)}$ bis $B_{(k)}$ über. Man erhält

$$PA_{B_{(0)}} = LU_0$$

$$L^{-1}PA_{B_{(1)}} = H_1 = D_1U_1 \Rightarrow PA_{B_{(1)}} = LD_1U_1$$

$$D_1^{-1}L^{-1}PA_{B(2)} = H_2 = D_2U_2 \Rightarrow PA_{B(2)} = LD_1D_2U_2.$$

Daher erhält man folgende Zerlegung für die Basismatrix $A_{B(k)}$:

$$PA_{B(k)} = LD_1D_2 \dots D_kU_k. \quad (11.5)$$

Wie kann man nun (11.5) dazu verwenden, die linearen Gleichungssysteme (10.2) einfach zu lösen? Betrachten wir zunächst ein Gleichungssystem der Form

$$A_{B(k)}x = b. \quad (11.6)$$

Zur Lösung von (11.6) löst man zunächst das gestaffelte Gleichungssystem

$$Lt = Pb$$

und führt dann die Transformation

$$w = (D_1 \dots D_k)^{-1}t \quad (11.7)$$

durch. Anschließend löst man das gestaffelte Gleichungssystem

$$U_kx = w,$$

dessen Lösung auch Lösung von (11.6) ist, denn

$$t = D_1 \dots D_k w$$

$$Lt = LD_1 \dots D_k w = LD_1 \dots D_k U_k x = PA_{B(k)}x = Pb.$$

Auf ähnliche Weise löst man ein Gleichungssystem der Form

$$x'A_{B(k)} = b'. \quad (11.8)$$

Zunächst löst man wieder das gestaffelte Gleichungssystem

$$h'U_k = b'.$$

Dann transformiert man

$$v' = h'(D_1 \dots D_k)^{-1} \quad (11.9)$$

und löst erneut ein gestaffeltes Gleichungssystem

$$(Px)'L = v'.$$

Wenn man

$$x'P^{-1}PA_{B(k)} = (x'P')(LD_1 \dots D_kU_k) = (Px)'LD_1 \dots D_kU_k = b'$$

betrachtet, erkennt man infolge von $v' = (Px)'L$ und $h' = v'D_1 \dots D_k$ die Richtigkeit dieses Vorgehens. Man beachte, dass die Transformationen (11.7) und (11.9) auf einer Rechenanlage sehr effizient durchgeführt werden können.

Ist das Ausgangssystem als Ungleichungssystem mit $b \geq 0$ gegeben, so ist anfangs $P = L = U = E$, d.h. P und L brauchen im Folgenden nicht weiter beachtet werden. Rundungsfehler können durch die Ausdrücke

$$\|b - A_B x_B\| \text{ bzw. } \|c'_B - \pi' A_B\| \quad (11.10)$$

kontrolliert werden. Bei der praktischen Durchführung des Simplexverfahrens empfiehlt es sich eine erneute LU -Zerlegung der Matrix A_B vorzunehmen, wenn die Normen (11.10) eine gewisse vorgesehene Größe infolge unvermeidlicher Rundungsfehler überschreiten.

Es kann nun gezeigt werden, dass die Störungen, die durch Rundungsfehler hervorgerufen werden, beschränkt werden können ohne Festlegung einer speziellen Reihenfolge für Variablen, die in die Basis eintreten oder sie verlassen. Dadurch erhält man ein numerisch stabiles Verfahren. Die Fehlerabschätzungen werden bei Bartels [4] durchgeführt.

Beispiel 11.1

$$\begin{array}{ll} \text{Maximiere} & x_1 + x_2 \\ \text{unter} & x_1 + 2x_2 \leq 4 \\ & 2x_1 - x_2 \leq 3 \\ & x_1 + x_2 \leq 3 \\ & x_i \geq 0 \quad (i = 1, 2) \end{array}$$

Die Ausgangsmatrix, deren Spalten den Variablen x_1, x_2, \dots, x_5 entsprechen, ist

$$A = \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 2 & -1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Wir haben anfangs $N = (1, 2)$ und $B = (3, 4, 5)$ mit $U_0 = L = P = E$. Nun ist nach dem Simplexverfahren $s = 1$, $r = 2$. Somit wird die zweite Spalte in U_0 gestrichen und an die letzte Stelle tritt die zu x_1 gehörende Spalte. Auf diese Weise erhält man die erste Hessenbergmatrix

$$H_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 1 & 1 \end{pmatrix}.$$

Die augenblickliche Basis ist $B_{(1)} = (3, 5, 1)$, $N_{(1)} = (2, 4)$. Trianguliert man die Hessenbergmatrix H_1 so erhält man $H_1 = P_1 \cdot U_1$ mit

$$P_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad U_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix}.$$

Wir lösen nun

$$A_{B_{(1)}} x_{B_{(1)}} = b.$$

Zunächst ist $Lt = b$. Daher ist $t = b$ und

$$w = P_1 t = \begin{pmatrix} 4 \\ 3 \\ 3 \end{pmatrix}.$$

Die Lösung von $U_1 x_{B(1)} = w$ ergibt

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} x_3 \\ x_5 \\ x_1 \end{pmatrix} = \begin{pmatrix} 4 \\ 3 \\ 3 \end{pmatrix} \implies \begin{pmatrix} x_3 \\ x_5 \\ x_1 \end{pmatrix} = \begin{pmatrix} 2.5 \\ 1.5 \\ 1.5 \end{pmatrix}.$$

Nun lösen wir $\pi' A_{B(1)} = c'_{B(1)}$. Dazu lösen wir zunächst $h' U_1 = c'_{B(1)}$:

$$(h_1, h_2, h_3) \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} = (0 \ 0 \ 1) \implies h = \begin{pmatrix} 0 \\ 0 \\ 0.5 \end{pmatrix}.$$

Nun ist $v' = h'(P_1)^{-1} \Rightarrow v' = (0, 0.5, 0)$. Daraus folgt sofort

$$(P\pi)'L = v' \Rightarrow \pi' = v' = (0, 0.5, 0).$$

Somit berechnen sich die reduzierten Kosten zu

$$\begin{aligned} (\tilde{c}_4, \tilde{c}_2) &= (0, 1) - (0, 0.5, 0) \begin{pmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \\ &= (0, 1) - (0.5, -0.5) = (-0.5, 1.5) \end{aligned}$$

Daher wird x_2 in die Basis aufgenommen. Zur Bestimmung der Variablen, die die Basis verlässt, müssen wir

$$A_{B(1)} \tilde{a}_2 = a_2$$

lösen. Analog wie vorhin, erhalten wir

$$w = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix} \text{ und } \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} \tilde{a}_{12} \\ \tilde{a}_{22} \\ \tilde{a}_{32} \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix} \Rightarrow \tilde{a}_2 = \begin{pmatrix} 2.5 \\ 1.5 \\ -0.5 \end{pmatrix}.$$

Daher wird etwa \tilde{a}_{22} das Pivotelement und x_5 verlässt die Basis. Nun gilt $B(2) = (3, 1, 2)$, $N(2) = (4, 5)$ und

$$H_2 = \begin{pmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \\ 0 & 2 & -1 \end{pmatrix}.$$

Man beachte, dass man die letzte Spalte bereits bei der Bestimmung des Vektors w zur Lösung von $A_{B(1)} \tilde{a}_s = a_s$ berechnet hat. Die Triangulierung von H_2 ergibt $H_2 = P_2 M_2^{-1} U_2$ mit

$$P_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{1}{2} & 1 \end{pmatrix}, \quad U_2 = \begin{pmatrix} 1 & 1 & 2 \\ 0 & 2 & -1 \\ 0 & 0 & \frac{3}{2} \end{pmatrix}.$$

Wir lösen nun $A_{B(2)} x_{B(2)} = b$. Nun ist

$$w = M_2 P_2 P_1 b = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 3 \\ 3 \end{pmatrix} = \begin{pmatrix} 4 \\ 3 \\ \frac{3}{2} \end{pmatrix}$$

und

$$\begin{pmatrix} 1 & 1 & 2 \\ 0 & 2 & -1 \\ 0 & 0 & \frac{3}{2} \end{pmatrix} \begin{pmatrix} x_3 \\ x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 3 \\ \frac{3}{2} \end{pmatrix}$$

liefert die Lösung $x_2 = 1$, $x_1 = 2$, $x_3 = 0$. Nun lösen wir $\pi' A_{B(2)} = c'_{B(2)}$:

$$\begin{pmatrix} h_1 & h_2 & h_3 \end{pmatrix} \begin{pmatrix} 1 & 1 & 2 \\ 0 & 2 & -1 \\ 0 & 0 & \frac{3}{2} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix} \Rightarrow h = \begin{pmatrix} 0 \\ \frac{1}{2} \\ 1 \end{pmatrix}.$$

Die Transformation auf v liefert

$$v' = h' M_2 P_2 P_1 = \begin{pmatrix} 0 & \frac{1}{2} & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{1}{2} & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$$

und $\pi' = v'$. Somit berechnen sich die neuen reduzierten Kosten zu

$$(\tilde{c}_4, \tilde{c}_5) = (0, 0) - (0, 0, 1) \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = (0, 0) - (0, 1) = (0, -1) \leq 0.$$

Also ist die zuletzt gefundene Lösung optimal, und es gilt

$$x_1 = 2, \quad x_2 = 1.$$

Die LU -Zerlegung der Matrix A_B ist nicht die einzige Möglichkeit zur Stabilisierung des Simplexverfahrens. So kann zum Beispiel das Simplexverfahren auch durch eine Cholesky-Zerlegung von $A_B A_B'$ stabilisiert werden (vgl. Murty [39]).

$$A_B A_B' = L L'$$

Allerdings ist dann die untere Dreiecksmatrix L im allgemeinen keine dünne Matrix, selbst wenn A_B dünn ist. Für Überlegungen zur Zerlegung und zum Updaten dünner Matrizen beim Simplexverfahren siehe Forrest und Tomlin [18]. Ohne Schwierigkeiten kann beim Simplexverfahren mit LU -Zerlegung die Spaltenauswahl nach der Regel (7.3) vorgenommen werden. Der Leser überlege sich, welche Modifikationen dazu vorgenommen werden müssen.

Kapitel 12

Duale lineare Programme

Jedem linearen Programm kann man ein anderes LP zuordnen und zwischen diesen beiden Problemen bestehen wichtige Beziehungen. Der *primale Aufgabe*

$$\text{Maximiere } c'x \text{ unter den Restriktionen } Ax \leq b, x \geq 0$$

entspricht das *duale Problem*

$$\text{Minimiere } b'y \text{ unter den Restriktionen } A'y \geq c, y \geq 0.$$

Somit gehen die Koeffizienten der Zielfunktion in den Restriktionsvektor des dualen Problems über, ferner die Matrix A in die transponierte Matrix A' und der Vektor b der rechten Seite in den Vektor der Zielfunktionskoeffizienten. An die Stelle eines Maximierungsproblems tritt ein Minimierungsproblem. Offenbar ist das duale Problem der dualen Aufgabe gleich der primalen Aufgabe, daher ist die Festsetzung primal-dual rein willkürlich.

Wie verhalten sich nun Gleichungen beim Übergang zum dualen Problem? Betrachten wir ein lineares Programm mit *einer* Gleichung und den Vorzeichenbedingungen:

$$\max\{c'x \mid a'x = b, x \geq 0\} = \max\{c'x \mid a'x \leq b, -a'x \leq -b, x \geq 0\}.$$

Dual dazu ist

$$\min\{by_1 - by_2 \mid ay_1 - ay_2 \geq c, y_1 \geq 0, y_2 \geq 0\}.$$

Setze $y := y_1 - y_2$. Dann ist y nicht vorzeichenbeschränkt und man erhält:

$$\min\{by \mid ay \geq c, y \in \mathbb{R}\}.$$

Damit erhält man folgende Zuordnungsregeln zwischen zueinander dualen Programmen:

<i>Primales System</i>	<i>Duales System</i>
Zielfunktion ($\rightarrow \max$)	Konstanten
Konstanten	Zielfunktion ($\rightarrow \min$)
Koeffizientenmatrix	Transponierte Koeffizientenmatrix
Relationen:	Variable:
<i>i</i> -te Restriktion ist Ungleichung \leq	$y_i \geq 0$
<i>i</i> -te Restriktion ist Gleichung	y_i nicht vorzeichenbeschränkt.
Variable:	Restriktionen:
$x_j \geq 0$	<i>j</i> -te Ungleichung $\geq c_j$
x_j nicht vorzeichenbeschränkt	<i>j</i> -te Restriktion ist Gleichung

Das primale und das duale Problem lassen sich leicht in folgendem Diagramm ablesen (*“Tucker-Diagramm”*):

Variable	$y_1 \geq 0$	\dots	$y_m \geq 0$	Relation	Konstanten
$x_1 \geq 0$	a_{11}	\dots	a_{m1}	\geq	c_1
\vdots	\vdots		\vdots	\vdots	\vdots
$x_n \geq 0$	a_{1n}	\dots	a_{mn}	\geq	c_n
Relation	\leq		\leq		$\leq \max z$
Konstanten	b_1	\dots	b_m	$\geq \min v$	

→ duales Problem

Beispiel 12.1 *Das duale Problem zu*

$$\begin{aligned}
 & \text{Maximiere } x_1 + 2x_2 && \text{unter den Restriktionen} \\
 & x_1 &\leq & 4 \\
 & 2x_1 + x_2 &\leq & 10 \\
 & -x_1 + x_2 &\leq & 5 \\
 & x_i &\geq & 0 \text{ für } i = 1, 2
 \end{aligned}$$

lautet

$$\begin{aligned}
 & \text{Minimiere } 4y_1 + 10y_2 + 5y_3 && \text{unter den Restriktionen} \\
 & y_1 + 2y_2 - y_3 &\geq & 1 \\
 & y_2 + y_3 &\geq & 2 \\
 & y_i &\geq & 0 \text{ für } i = 1, 2, 3.
 \end{aligned}$$

Auch die duale Aufgabe zu einem Problem besitzt eine (ökonomische) Interpretation, wenn diese auch meist nicht so auf der Hand liegt wie die Deutung des primalen Systems. Dies sei am Transportproblem gezeigt:

Beispiel 12.2 *Duales Transportproblem*

Das Transportproblem (vergleiche Abschnitt 1.2.4) ist das Problem eines Produzenten, einen Plan festzulegen, wie die von ihm hergestellten Waren möglichst günstig an die Abnehmer versandt werden. Nun kommt ein Unternehmer, der meint, den Versand kostengünstiger durchführen zu können, zum Produzenten und bietet ihm an, seine Waren am Erzeugungsort zu kaufen und am Bestimmungsort wieder zu verkaufen. Der Preis der Produkte in dieser Transaktion variiert von Ort zu Ort und wird vom Unternehmer im voraus festgesetzt. Er muss jedoch die Preise so wählen, dass sein Angebot attraktiv für den Hersteller ist. Somit muss der Unternehmer Preise u_1, \dots, u_m für die m Produktionsorte und v_1, \dots, v_n für die n Bestimmungsorte wählen. Um gegenüber den herkömmlichen Transportmöglichkeiten konkurrenzfähig zu sein, müssen seine Preise

$$v_j - u_i \leq c_{ij} \quad (1 \leq i \leq m, 1 \leq j \leq n) \quad (12.1)$$

erfüllen, da $v_j - u_i$ den Betrag pro Einheit darstellt, den der Hersteller zahlen muss, wenn er eine Einheit der Ware am Ort i verkauft und am Ort j zurückkauft. Unter den Bedingungen

(12.1) wird der Unternehmer seinen Gewinn maximieren. Dies führt auf das Problem

$$\text{Maximiere } \sum_{j=1}^n b_j v_j - \sum_{i=1}^m a_i u_i \text{ unter}$$

$$v_j - u_i \leq c_{ij} \quad (1 \leq i \leq m, 1 \leq j \leq n).$$

Dies ist das duale Problem zum Transportproblem.

Wegen der folgenden Sätze spielen duale lineare Programme in Theorie und Praxis eine so große Rolle. Das primale Problem sei

$$(P) \quad \max\{c'x \mid Ax \leq b, x \geq 0\}$$

und das dazu duale Problem bezeichnen wir mit (D):

$$(D) \quad \min\{b'y \mid A'y \geq c, y \geq 0\}.$$

Mit M_P bezeichnen wir die Menge der zulässigen Lösungen des primalen Problems und mit M_D jene des dualen Problems. Dann gilt

Satz 12.1 (Dualitätssatz):

Besitzt eine von zwei zueinander dualen linearen Optimierungsaufgaben eine endliche Optimallösung, so besitzt auch das andere Problem eine endliche Optimallösung und der Optimalwert der beiden Zielfunktionen ist gleich:

$$c'x^* = b'y^*.$$

Zum Dualitätssatz gibt es verschiedene Beweise, einer beruht auf der Anwendung des Simplexverfahrens (Dantzig [14]). In den nächsten Abschnitten wird der Dualitätssatz mit Hilfe eines Trennungssatzes und der Alternativsätze der linearen Algebra bewiesen.

Eine Abschwächung des obigen Resultates ist folgender

Satz 12.2 (Schwacher Dualitätssatz):

Für jede zulässige Lösung x von (P) und für jedes zulässige y von (D) gilt

$$c'x \leq b'y. \tag{12.2}$$

Beweis: Aus $A'y \geq c, x \geq 0$ und $Ax \leq b$ folgt

$$c'x \leq y'Ax \leq y'b.$$

■

Unmittelbar aus (12.2) erhält man

Korollar: Ist $\bar{x} \in M_P$ und $\bar{y} \in M_D$ mit $c'\bar{x} = b'\bar{y}$, so sind \bar{x} und \bar{y} Optimallösungen des primalen bzw. des dualen Programms.

Aus Satz 12.1 und Satz 12.2 folgt dann der nachstehende Existenzsatz

Satz 12.3 (Existenzsatz)

1. Besitzen (P) und (D) zulässige Lösungen, so besitzen beide Probleme endliche Optimallösungen.
2. Besitzt nur eines der beiden Probleme zulässige Lösungen, so besitzt diese Aufgabe keine endliche Optimallösung.
3. Besitzt ein Problem zwar zulässige Lösungen, aber keine endliche Optimallösung, so hat die dazu duale Aufgabe keine zulässigen Punkte.

Beweis:

1. Besitzen beide Probleme zulässige Lösungen x und y , so gilt nach (12.2)

$$c'x \leq b'y,$$

d.h. $c'x$ ist auf M_P nach oben beschränkt und nimmt daher auf M_P sein Maximum an. Nach Satz 12.1 ist dann aber $c'x^* = b'y^*$. Also besitzt auch das duale Problem eine endliche Optimallösung y^* .

2. Ann.: $x \in M_P$, $M_D = \emptyset$.

Hätte (P) eine endliche Optimallösung, so hätte nach Satz 12.1 auch (D) eine endliche Optimallösung im Widerspruch zu $M_D = \emptyset$.

3. $x \in M_P$, $c'x$ unbeschränkt.

Gäbe es ein $\bar{y} \in M_D$, so wäre nach (12.2) $c'x \leq b'\bar{y}$. Also wäre $c'x$ beschränkt im Widerspruch zur Voraussetzung. Daher ist $M_D = \emptyset$.

■

Ein wichtiges Resultat ist auch der nachfolgende Satz vom komplementären Schlupf, der notwendige und hinreichende Bedingungen dafür angibt, dass zulässige Lösungen x von (P) und y von (D) Optimallösungen sind.

Um Zeilen- und Spaltenvektoren einer Matrix zu unterscheiden, führen wir folgende Schreibweise ein: a_i bezeichne den i -ten Zeilenvektor der Matrix A und a_j den j -ten Spaltenvektor von A .

Satz 12.4 (Satz vom komplementären Schlupf, Tucker, 1956)

Besitzen (P) und (D) zulässige Lösungen x und y , so sind die Bedingungen (12.3) und (12.4) notwendig und hinreichend für die Optimalität von x und y .

$$\begin{aligned} x_j > 0 &\Rightarrow a'_{.j}y = c_j \\ a'_{.j}y > c_j &\Rightarrow x_j = 0 \end{aligned} \tag{12.3}$$

$$\begin{aligned} y_i > 0 &\Rightarrow a'_i x = b_i \\ a'_i x < b_i &\Rightarrow y_i = 0. \end{aligned} \tag{12.4}$$

Beweis:

1. Aus $Ax \leq b$, $x \geq 0$ und $A'y \geq c$, $y \geq 0$ sowie $c'x = b'y$ folgt $(Ax)'y \leq b'y = c'x \Rightarrow x'(A'y - c) \leq 0 \Rightarrow x'(A'y - c) = 0$. Dies ergibt (12.3).
2. (12.4) folgt aus $(A'y)'x \geq c'x = y'b$, denn $y'(Ax - b) \geq 0$ und $y \geq 0$, $Ax \leq b$ implizieren $y'(Ax - b) = 0$.
3. Gelten für $x \in M_P$ und $y \in M_D$ die Beziehungen (12.3) und (12.4), so folgt $x'(A'y - c) = y'(Ax - b) = 0$ und daher $c'x = x'A'y = b'y$. Nach dem schwachen Dualitätssatz sind dann x und y Optimallösungen.

■

Kapitel 13

Eine Anwendung der Dualitätstheorie: Der Hauptsatz der Spieltheorie.

Die frühe Entwicklung der linearen Optimierung, insbesondere der Dualitätstheorie, hing eng zusammen mit Untersuchungen auf dem Gebiete der Spieltheorie durch J. von Neumann. Das Interesse, das von Seiten der Spieltheorie vorhanden war, trug ganz wesentlich zur schnellen Entwicklung der linearen Optimierung in den vierziger und fünfziger Jahren des 20. Jahrhunderts bei. In diesem Abschnitt soll gezeigt werden, wie der Minimax-Satz für Zweipersonen-Nullsummenspiele, der auch als Hauptsatz der Spieltheorie bezeichnet wird, aus dem Dualitätssatz der linearen Optimierung gefolgert werden kann.

Endliche Zweipersonen-Nullsummenspiele lassen sich folgenderweise kurz beschreiben. Jedem der Spieler k ($k = 1, 2$) steht eine endliche Menge von Strategien Σ_k zur Verfügung. Ferner ist eine Auszahlungsfunktion $A : \Sigma_1 \times \Sigma_2 \rightarrow \mathbb{R}$ gegeben, die besagt, dass Spieler 2 a_{ij} -Einheiten an Spieler 1 zu zahlen hat, falls Spieler 1 die Strategie $\sigma_i \in \Sigma_1$ und Spieler 2 die Strategie $\tau_j \in \Sigma_2$ gewählt haben. Spiele dieser Art heißen Nullsummenspiele, weil alles, was der eine Spieler verliert, der andere Spieler gewinnt.

Beispiel 13.1 *Beide Spieler können "Kopf" oder "Zahl" raten. Raten beide das Gleiche, so hat der erste Spieler gewonnen, anderenfalls der zweite Spieler. In diesem Fall sind die Strategiemengen $\Sigma_1 = \Sigma_2 = \{ \text{"Kopf"}, \text{"Zahl"} \}$ und die Auszahlungsmatrix hat folgende Form*

	K	Z
K	+1	-1
Z	-1	+1

Beispiel 13.2 *Der erste Spieler habe die Strategien $\sigma_1, \sigma_2, \sigma_3$ zur Verfügung, der zweite Spieler 4 Strategien τ_1, τ_2, τ_3 und τ_4 . Die Auszahlungsmatrix habe folgende Gestalt*

	τ_1	τ_2	τ_3	τ_4
σ_1	-4	2	3	-1
σ_2	5	-1	2	-2
σ_3	1	2	3	0

Der Spieler 1 wird versuchen, den Betrag, den er erhält, zu maximieren. Also wird Spieler 1 eine Strategie σ_i wählen, für die $\min_j a_{ij}$ maximal wird. Dies ist im obigen Beispiel die Strategie σ_3 . Andererseits wird Spieler 2 versuchen, seinen Verlust zu minimieren. Daher wird er eine Strategie τ_j wählen, für die $\max_i a_{ij}$ minimal wird. Dies ist im obigen Beispiel die Strategie τ_4 . Es sei

$$v_1 = \max_i \min_j a_{ij}, \quad v_2 = \min_j \max_i a_{ij}.$$

Dann ist v_1 der Mindestgewinn, den Spieler 1 erreichen kann und v_2 der größte Verlust, den Spieler 2 bei rationalem Verhalten einplanen muss. Gilt $v_1 = v_2$, so sagt man, dass Spiel besitze einen "Gleichgewichtspunkt". Im Beispiel 13.1 gibt es keinen Gleichgewichtspunkt, wohl aber in Beispiel 13.2 für (σ_3, τ_4) . Man kann sich aber leicht überlegen, dass in jedem Falle

$$v_1 \leq v_2$$

gilt. Denn aus $a_{ij} \leq \max_i a_{ij}$ folgt $\min_j a_{ij} \leq \min_j \max_i a_{ij} = v_2$ für alle $1 \leq i \leq m$. Daher ist $v_1 = \max_i \min_j a_{ij} \leq v_2$.

Treffen die Spieler die Wahl ihrer Strategien durch Zufall und nicht aufgrund rationaler Überlegungen, die vom jeweiligen Opponenten einkalkuliert werden könnten, so kann der *Erwartungswert* von v_1 eventuell noch erhöht und der Erwartungswert von v_2 möglicherweise noch erniedrigt werden. Aus diesen Überlegungen führte J. von Neumann *gemischte Strategien* ein: Die Strategiemenge für Spieler k sei $\Sigma_k = \{\sigma_1, \dots, \sigma_{m(k)}\}$. Dabei kann die Anzahl m noch von k abhängen. Eine gemischte Strategie $x = (x_1, \dots, x_m)'$ ist eine Wahrscheinlichkeitsverteilung auf der Menge der *reinen* Strategien Σ_k . Für eine gemischte Strategie gilt daher

$$\sum_{\sigma_i \in \Sigma_k} x_i = 1, \quad x_i \geq 0.$$

Anstelle der Auszahlungsfunktion tritt nun die *erwartete Auszahlung*

$$A(x, y) = \sum_i \sum_j x_i a_{ij} y_j = x' Ay.$$

Nehmen wir nun an, Spieler 1 wählt die Strategie x und Spieler 2 bemerkt dies. Spieler 2 wird nun seine gemischte Strategie y so wählen, dass $A(x, y)$ minimal wird. Das heißt, bei festem x ist der erwartete Gewinn für Spieler 1

$$v(x) = \min \{ x' Ay \mid \sum_j y_j = 1, y_j \geq 0 \}.$$

Nach dem Hauptsatz der linearen Optimierung (Satz 3.4) wird das Minimum in einer Ecke des zulässigen Bereiches $\{y \mid \sum_j y_j = 1, y_j \geq 0\}$ angenommen. Die Ecken dieser Menge entsprechen den Einheitsvektoren des \mathbb{R}^n . Daher entspricht die Optimallösung y^* etwa der *reinen* Strategie τ_j . Somit gilt

$$v(x) = \min_j \sum_i a_{ij} x_i.$$

Spieler 1 muss also seine Strategie so wählen, dass $v(x)$ maximal wird

$$v_1 = \max_x \min_j \sum_i a_{ij} x_i \tag{13.1}$$

(v_1 existiert, da $v(x)$ stetig auf der kompakten Menge $\{x \mid \sum x_i = 1, x_i \geq 0\}$ ist). Die Beziehung (13.1) lässt sich folgenderweise als lineares Programm schreiben

max v unter den Restriktionen

$$\begin{aligned} v &\leq \sum_{i=1}^m a_{ij}x_i \\ \sum_{i=1}^m x_i &= 1 \\ x_i &\geq 0, v \text{ beliebig.} \end{aligned}$$

Ersetzt man v durch $-u$, so erhält man

min u unter den Restriktionen (13.2)

$$\begin{aligned} u + \sum_{i=1}^m a_{ij}x_i &\geq 0 \\ \sum_{i=1}^m x_i &= 1 \\ u \text{ beliebig, } x_i &\geq 0. \end{aligned}$$

Analog hat Spieler 2 bei fester Wahl einer gemischten Strategie y den erwarteten Verlust

$$v(y) = \max_i \sum_j a_{ij}y_j.$$

Spieler 2 muss also y so wählen, dass er

$$v_2 = \min_y \max_i \sum_j a_{ij}y_j \tag{13.3}$$

erhält. Die Beziehung (13.3) lässt sich schreiben als

min w unter den Restriktionen

$$\begin{aligned} w &\geq \sum_{j=1}^n a_{ij}y_j \\ \sum_{j=1}^n y_j &= 1 \\ w \text{ beliebig, } y_j &\geq 0 \end{aligned}$$

Ersetzt man w durch $-z$, so erhält man das äquivalente lineare Programm

max z unter den Restriktionen (13.4)

$$\begin{aligned}
 z + \sum_j a_{ij} y_j &\leq 0 \\
 \sum_j y_j &= 1 \\
 z \text{ beliebig, } y_j &\geq 0.
 \end{aligned}$$

Nun sind aber (13.2) und (13.4) ein Paar zueinander dualer linearer Programme. Beide besitzen zulässige Lösungen. Daher gibt es nach den Sätzen 12.3 und 12.1 gemischte Strategien x^* für Spieler 1 und y^* für Spieler 2, so dass

$$v_1 = x^{*'} A y^* = v_2$$

gilt. Damit haben wir gezeigt

Satz 13.1 (*Minimax-Satz für Zweipersonen-Nullsummenspiele; J. von Neumann*)

Für ein endliches Zweipersonen-Nullsummenspiel gibt es stets Strategien x^* für Spieler 1 und y^* für Spieler 2, so dass der maximale erwartete Gewinn des Spielers 1 gleich dem minimalen erwarteten Verlust des Spielers 2 ist, d.h. für alle Wahrscheinlichkeitsverteilungen x auf Σ_1 und y auf Σ_2 gilt

$$x' A y^* \leq x^{*'} A y^* \leq x^{*'} A y. \quad (13.5)$$

(13.5) besagt, dass durch (x^*, y^*) ein Gleichgewichtspunkt oder Sattelpunkt erreicht wird.

Kapitel 14

Trennungssätze für konvexe Mengen

Der Dualitätssatz der linearen Programmierung ist eng mit Trennungssätzen für konvexe Mengen verbunden. Daher wollen wir zunächst in diesem Abschnitt einige Resultate über die Trennung konvexer Mengen mittels Hyperebenen herleiten.

Zunächst sei daran erinnert, dass eine Teilmenge $C \subset \mathbb{R}^n$ konvex heißt, wenn sie mit zwei Punkten x und y auch die ganze Verbindungsstrecke $\lambda x + (1 - \lambda)y$ ($0 \leq \lambda \leq 1$) enthält.

Eine Hyperebene $H = \{x \mid a'x = a_0\}$ trennt die konvexen Mengen A und B , wenn für alle $x \in A$ und alle $y \in B$ gilt:

$$a'x \leq a_0, \quad a'y > a_0. \quad (14.1)$$

Das Hauptresultat dieses Abschnittes wird sein, dass man zwei disjunkte, nichtleere konvexe Mengen im \mathbb{R}^n stets durch eine Hyperebene trennen kann (vgl. Abbildung 14.1).

Satz 14.1 *Sei C eine nichtleere, konvexe Menge im \mathbb{R}^n und y ein Punkt, der nicht im topologischen Abschluss \overline{C} der Menge C liegt. Dann gibt es eine Hyperebene H durch den Punkt y , so dass C ganz in einem von H erzeugten offenen Halbraum liegt.*

Die Aussage und der Beweis von Satz 14.1 wird durch Abbildung 14.2 verdeutlicht.

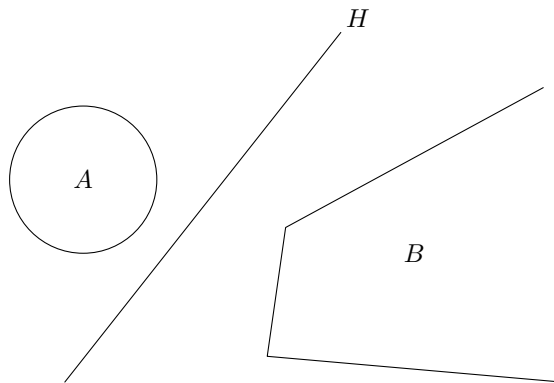


Abbildung 14.1: Trennung der konvexen Mengen A und B durch eine Hyperebene H

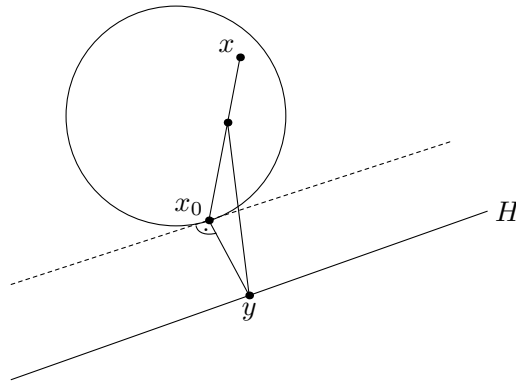


Abbildung 14.2: Zu Satz 14.1 und seinem Beweis

Beweis von Satz 14.1: Sei $\delta = \inf_{x \in C} \|x - y\| > 0$. Da $f(x) = \|x - y\|$ stetig ist und auf der kompakten Menge $\overline{C} \cap \{x \mid \|x - y\| \leq 2\delta\}$ ihr Minimum annimmt, gibt es einen Punkt $x_0 = x_0(y) \in \overline{C}$ mit $\|x_0 - y\| = \delta$.

Sei nun $x \in C$. Infolge der Konvexität von C gilt für alle λ mit $0 \leq \lambda \leq 1$:

$$x_0 + \lambda(x - x_0) \in \overline{C} \text{ und } \|x_0 + \lambda(x - x_0) - y\|^2 \geq \|x_0 - y\|^2.$$

Daraus folgt

$$2\lambda(x_0 - y)'(x - x_0) + \lambda^2\|x - x_0\|^2 \geq 0.$$

Strebt nun λ von rechts gegen 0, so erhält man $(x_0 - y)'(x - x_0) \geq 0$, d.h. $(x_0 - y)'x \geq (x_0 - y)'x_0 = (x_0 - y)'y + (x_0 - y)'(x_0 - y) = (x_0 - y)'y + \delta^2$.

Somit trennt die Hyperebene

$$H = \{x \mid (x_0 - y)'x = (x_0 - y)'y\}$$

den Punkt y von der Menge C . Wegen $\delta^2 > 0$ liegt C in einem der beiden offenen Halbräume, die von H erzeugt werden. ■

Man kann diesen Satz leicht auf den Fall erweitern, dass y Randpunkt von C ist. Dann wird H eine *Stützhyperebene* von C . Eine Stützhyperebene $H = \{x \mid a'x = a_0\}$ einer (nicht notwendig konvexen) Menge C ist eine Hyperebene, für die gilt

$$H \cap \overline{C} \neq \emptyset,$$

$$\forall x \in C : a'x \geq a_0.$$

Satz 14.2 Sei C eine nichtleere, konvexe Menge im \mathbb{R}^n , $C \neq \mathbb{R}^n$, und y ein Randpunkt von C . Dann gibt es eine Hyperebene durch y , die C in einem ihrer Halbräume enthält.

Satz 14.2 besagt also, dass es zu *jedem* Randpunkt y einer konvexen Menge C eine Stützhyperebene von C gibt (vgl. Abbildung 14.3).

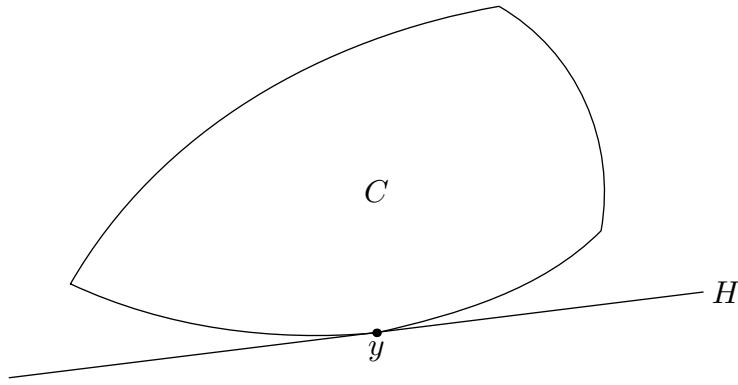


Abbildung 14.3: Stützhyperebene an die Menge C

Beweis von Satz 14.2: Sei $\{y_k\}$ eine gegen y konvergente Folge von Punkten $y_k \notin \overline{C}$. Ferner sei $a_k := x_0(y_k) - y_k$ die nach Satz 14.1 konstruierte Folge, deren Elemente wir auf die Länge 1 normieren. (Die normierten Vektoren bezeichnen wir wieder mit a_k). Dann gilt

$$a'_k y_k \leq \inf_{x \in C} a'_k x.$$

Da $\{a_k\}$ beschränkt ist, hat $\{a_k\}$ eine konvergente Teilfolge $\{a_{k_r}\}$ mit dem Grenzwert a . Für dieses a und alle $x \in C$ gilt:

$$a' y = \lim_{r \rightarrow \infty} a'_{k_r} y_{k_r} \leq \lim_{r \rightarrow \infty} a'_{k_r} x = a' x.$$

■

Nun zeigen wir den Trennungssatz für konvexe Mengen im \mathbb{R}^n :

Satz 14.3 Seien C_1, C_2 disjunkte, nicht-leere konvexe Mengen im \mathbb{R}^n . C_2 sei offen. Dann gibt es eine Hyperebene, die C_1 und C_2 trennt, d.h.

$$a' x \leq \beta < a' y \text{ für alle } x \in C_1, y \in C_2.$$

Beweis: $C := \{y - x \mid x \in C_1, y \in C_2\}$ ist eine offene, konvexe Menge, die 0 nicht enthält, denn $C = \bigcup_{x \in C_1} \{y - x \mid y \in C_2\}$ ist als Vereinigung offener Mengen offen und $C_1 \cap C_2 = \emptyset \Rightarrow 0 \notin C$. Daher gibt es nach Satz 14.1 oder 14.2 eine Hyperebene H , so dass C ganz in einem von H erzeugten Halbraum liegt, d.h. $\exists a : a'(y - x) \geq 0 = a'0$. Setze $\beta := \inf_{y \in C_2} a' y$. Dann gilt

$$a' x \leq \beta < a' y.$$

■

Kapitel 15

Alternativsätze und der Beweis des Dualitätssatzes

Es sei A eine reelle $(m \times n)$ -Matrix mit beliebigem Rang und $b \in \mathbb{R}^m$. Dann gilt

Satz 15.1 *Entweder besitzt*

$$Ax = b \tag{15.1}$$

eine Lösung $x \in \mathbb{R}^n$ oder das Gleichungssystem

$$A'y = 0, \quad b'y = 1 \tag{15.2}$$

besitzt eine Lösung $y \in \mathbb{R}^m$, aber beide Systeme sind nicht gleichzeitig lösbar.

Beweis:

1. Wäre $x \in \mathbb{R}^n$ Lösung von (15.1) und $y \in \mathbb{R}^m$ Lösung von (15.2), so wäre

$$0 = (Ax)'y = x'A'y = b'y = 1. \text{ Widerspruch.}$$

2. Ist (15.1) nicht lösbar, so ist b linear unabhängig von den Spaltenvektoren der Matrix A .

Der Rang von A sei r . Somit hat die $(n+1) \times m$ -Matrix $\begin{pmatrix} A' \\ b' \end{pmatrix}$ den Rang $r+1$. Dann hat aber auch die $(n+1) \times (m+1)$ -Matrix

$$\left(\begin{array}{c|c} A' & 0 \\ \hline b' & 1 \end{array} \right)$$

den Rang $r+1$. Daher hat das inhomogene lineare Gleichungssystem (15.2) den gleichen Rang wie das zugehörige homogene System. Somit ist (15.2) lösbar. ■

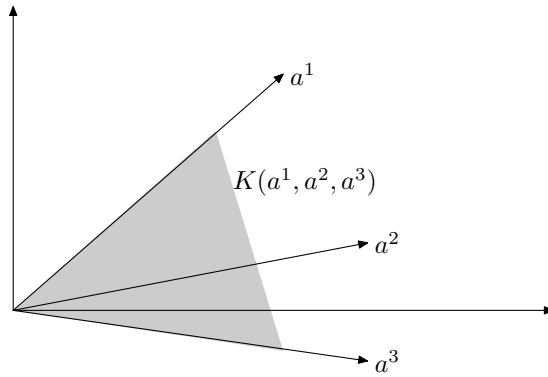
Satz 15.2 *Entweder besitzt*

$$Ax = b, \quad x \geq 0 \tag{15.3}$$

eine Lösung $x \in \mathbb{R}^n$ oder es besitzt

$$A'y \geq 0, \quad b'y < 0 \tag{15.4}$$

eine Lösung $y \in \mathbb{R}^m$, aber beide Systeme sind nicht gleichzeitig lösbar.

Abbildung 15.1: Der von a^1, a^2, a^3 erzeugte Kegel

Bevor wir Satz 15.2 beweisen, wollen wir ihn geometrisch interpretieren. Sind a^1, \dots, a^n Vektoren im \mathbb{R}^m , so bezeichnet man mit $K(a^1, \dots, a^n) := \{\sum_{i=1}^n a^i x_i \mid x_i \geq 0\}$ den von den Spaltenvektoren a^i erzeugten Kegel.

(15.3) besagt nun, dass b im Kegel liegt, der von a^1, \dots, a^n erzeugt wird. Ist dies nicht der Fall, so gibt es eine Hyperebene durch den Nullpunkt, die b vom Kegel trennt. Dies ist aber die Aussage von (15.4).

Satz 15.2* Entweder ist $b \in K(a^1, \dots, a^n)$ oder es gibt eine Hyperebene durch den Nullpunkt, die b vom Kegel trennt.

Beweis:

1. Wäre x eine Lösung von (15.3) und y eine Lösung von (15.4), so wäre

$$0 > b'y = (Ax)'y = x'A'y \geq 0. \text{ Widerspruch.}$$

2. Hat $Ax = b$ keine Lösung, so gibt es nach Satz 15.1 ein \bar{y} mit $A'\bar{y} = 0$, $b'\bar{y} = 1$. Dann ist $y := -\bar{y}$ Lösung von (15.4).
3. Sei nun für jede Lösung x von $Ax = b$ mindestens eine Komponente negativ. Dann ist $b \notin K(a^1, \dots, a^n)$. Der Kegel $K(a^1, \dots, a^n)$ ist eine abgeschlossene, konvexe Menge C_1 im \mathbb{R}^n . Andererseits ist

$$C_2 := \{\alpha \cdot u \mid \|u - b\| < \varepsilon, \alpha > 0\}$$

für ein hinreichend kleines $\varepsilon > 0$ eine offene konvexe Menge im \mathbb{R}^n mit $C_1 \cap C_2 = \emptyset$ (vgl. Abbildung 15.2).

Nach Satz 14.3 gibt es daher eine Hyperebene $H = \{y \mid a'y = a_0\}$, die C_1 und C_2 trennt, d.h. für $u \in C_1$ und für $v \in C_2$:

$$a'u \geq a_0 > a'v.$$

Da $0 \in C_1$, gilt $a_0 \leq 0$. Andererseits ist $\alpha b \in C_2$ für beliebig kleines, aber positives α . Daher ist $a_0 = 0$. Somit gilt für $u = a^j$ ($j = 1, \dots, n$): $a'a^j \geq 0$ und für $v = b$: $a'b < 0$. Daher erfüllt $y := a$ das System (15.4).

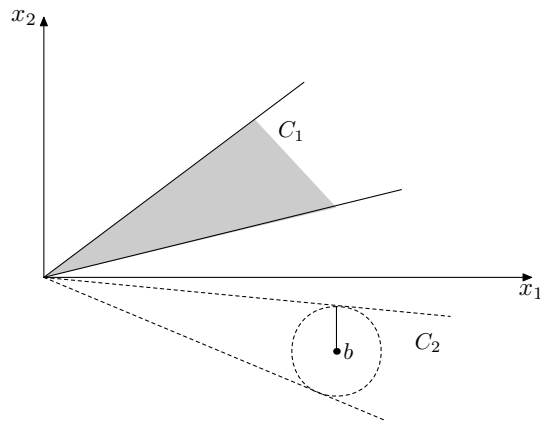


Abbildung 15.2: Zum Beweis von Satz 15.2

Satz 15.2 ist äquivalent zum berühmten Farkas-Lemma, das sich auch so aussprechen lässt: ■

Satz 15.3 (*Lemma von Farkas, 1902*):
 Folgende zwei Aussagen sind äquivalent:

$$\text{Für alle } y \text{ mit } A'y \geq 0 \text{ gilt } b'y \geq 0. \quad (15.5)$$

$$\text{Es gibt einen Vektor } x \geq 0, \text{ so dass } Ax = b. \quad (15.3)$$

Die Äquivalenz der beiden Sätze 15.2 und 15.3 lässt sich auf folgende Weise zeigen:

Beweis:

I. Aus Satz 15.2 folgt Satz 15.3:

- (1) Gilt (15.3), so hat $A'y \geq 0$, $b'y < 0$ keine Lösung. Daher folgt (15.5).
- (2) Gilt (15.3) nicht, so gilt (15.4). Also gibt es ein y mit $A'y \geq 0$, $b'y < 0$ im Widerspruch zu (15.5).

II. Aus Satz 15.3 folgt Satz 15.2:

- (1) Gilt (15.3), so ist wegen (15.5) das System (15.4) nicht lösbar.
- (2) Gilt (15.3) nicht, so gilt (15.5) nicht. Da $A'y \geq 0$ stets mit $y = 0$ lösbar ist, gibt es ein y mit $A'y \geq 0$, $b'y < 0$. Also ist (15.4) erfüllt.

Eine weitere geometrische Deutung des Lemma von Farkas ist Satz 15.3*. ■

Satz 15.3* *Der Halbraum $\{y \mid b'y \geq 0\}$ enthält den polyedrischen Kegel $\{y \mid A'y \geq 0\}$ genau dann, wenn der Vektor b im Kegel $K(a^1, \dots, a^n)$ liegt, der von den Vektoren a^1, \dots, a^n aufgespannt wird.*

Der folgende Satz, der auf Tucker [1956, Theorem 1] zurückgeht, folgt aus dem Lemma von Farkas und spielt eine fundamentale Rolle beim Beweis des Dualitätssatzes der linearen Optimierung:

Satz 15.4 Die Systeme $A'y \geq 0$ und $Ax = 0, x \geq 0$ besitzen Lösungen \bar{x}, \bar{y} mit

$$A'\bar{y} + \bar{x} > 0. \quad (15.6)$$

(Dabei ist das Größerzeichen in (15.6) komponentenweise zu verstehen!)

Beweis: Die Spaltenvektoren von A bezeichnen wir mit a^j ($1 \leq j \leq n$). Für festes k betrachten wir die folgenden beiden Systeme:

$$\sum_{\substack{j=1 \\ j \neq k}}^n a^j x_j = -a^k, \quad x_j \geq 0 \quad (j \neq k) \quad (15.7)$$

und

$$a^{j'} y \geq 0, \quad a^{k'} y > 0 \quad (1 \leq j \leq n, j \neq k). \quad (15.8)$$

Setzt man $b = -a^k$, so ist nach Satz 15.2 das System (15.7) oder das System (15.8) lösbar. Ist (15.7) lösbar, so gibt es ein $\bar{x}(k) \in \mathbb{R}^n$ mit $A\bar{x}(k) = 0$, $\bar{x}(k) \geq 0$ und $\bar{x}_k(k) = 1$. Ist (15.8) lösbar, so gibt es ein $\bar{y}(k) \in \mathbb{R}^m$ mit $A'\bar{y}(k) \geq 0$ und $a^{k'}\bar{y}(k) > 0$. Sei nun $Z_1 = \{k \mid (15.7) \text{ lösbar}\}$, $Z_2 = \{k \mid (15.8) \text{ lösbar}\}$. Es ist $Z_1 \cup Z_2 = \{1, 2, \dots, n\}$. Setzt man

$$\bar{x} := \begin{cases} 0, & \text{falls } Z_1 = \emptyset \\ \sum_{k \in Z_1} \bar{x}(k), & \text{sonst} \end{cases} \quad \bar{y} := \begin{cases} 0, & \text{falls } Z_2 = \emptyset \\ \sum_{k \in Z_2} \bar{y}(k), & \text{sonst} \end{cases}$$

so ist $A'\bar{y} \geq 0$, $A\bar{x} = 0$, $\bar{x} \geq 0$ und $A'\bar{y} + \bar{x} > 0$. ■

Als unmittelbare Folgerungen von Satz 15.4 erhält man zwei klassische Resultate von Stiemke und Gordan:

Satz 15.5 (Stiemke, 1915): Für $A \neq 0$ sind die folgenden zwei Aussagen äquivalent:

$$Ax = 0, \quad x > 0 \text{ hat keine Lösung.} \quad (15.9)$$

$$A'y \geq 0, \quad A'y \neq 0 \text{ hat eine Lösung.} \quad (15.10)$$

Beweis: Nach Satz 15.4 gibt es Vektoren \bar{x}, \bar{y} mit $A'\bar{y} \geq 0$, $A\bar{x} \geq 0$ und $A'\bar{y} + \bar{x} > 0$. Hat (15.9) keine Lösung, so folgt aus (15.6), dass (15.10) eine Lösung besitzt. Hat jedoch (15.9) eine Lösung mit $x > 0$, so ist nach dem Beweis zu Satz 15.4

$$a^{j'} y \geq 0, \quad a^{k'} y > 0 \quad (1 \leq j \leq n, j \neq k)$$

für jedes $k = 1, \dots, n$ nicht lösbar. Daher ist (15.10) nicht lösbar. ■

Auch der folgende Satz ergibt sich aus Satz 15.4:

Satz 15.6 (Gordan, 1873): Für $A \neq 0$ sind die folgenden zwei Aussagen äquivalent:

$$Ax = 0, \quad x \geq 0 \text{ hat eine Lösung } x \neq 0. \quad (15.11)$$

$$A'y > 0 \text{ hat keine Lösung.} \quad (15.12)$$

Beweis: Hat $A'y > 0$ keine Lösung, so folgt wieder aus (15.6) die Lösbarkeit von (15.11). Umgekehrt folgt aus dem Beweis von Satz 15.4, dass (15.11) keine Lösung besitzt, falls $A'y > 0$ eine Lösung besitzt. ■

Eine reelle $(n \times n)$ -Matrix A heißt *schiefsymmetrisch*, wenn $A = -A'$ gilt. Für schiefsymmetrische Matrizen kann man nun zeigen (vgl. Tucker, [1956, Theorem 5]):

Satz 15.7 *Sei A eine reelle, schiefsymmetrische $(n \times n)$ -Matrix. Dann gibt es ein $w \in \mathbb{R}^n$ mit $Aw \geq 0$, $w \geq 0$ und $Aw + w > 0$.*

Beweis: Man betrachte die beiden Systeme

$$\begin{pmatrix} E_n \\ A \end{pmatrix} y \geq 0 \text{ und } \begin{pmatrix} E_n & : & -A \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix} = 0, \begin{pmatrix} x \\ z \end{pmatrix} \geq 0.$$

Dabei ist E_n eine Einheitsmatrix mit n Zeilen und Spalten. Nach Satz 15.4 gibt es Vektoren $\bar{x}, \bar{y}, \bar{z}$ mit $\bar{y} \geq 0$, $A\bar{y} \geq 0$, $\bar{x} - A\bar{z} = 0$, $\bar{x} \geq 0$, $\bar{z} \geq 0$, $\bar{y} + \bar{x} > 0$, $A\bar{y} + \bar{z} > 0$. Daraus folgt $\bar{y} + A\bar{z} > 0$. Setzt man $w := \bar{y} + \bar{z}$, so ist $Aw = A\bar{y} + A\bar{z} \geq 0$, $w \geq 0$ und $Aw + w > 0$. ■

Dieser Satz ist die Grundlage eines Beweises von Goldman und Tucker [1956] zum Dualitätssatz der linearen Optimierung. Gegeben seien zwei zueinander lineare Programme:

$$\begin{aligned} (P) \quad & \max\{c'x \mid Ax \leq b, x \geq 0\} \\ (D) \quad & \min\{b'y \mid A'y \geq c, y \geq 0.\} \end{aligned}$$

Dabei ist A eine $(m \times n)$ -Matrix, x und c sind Vektoren im \mathbb{R}^n , b und y sind Vektoren im \mathbb{R}^m . Den zulässigen Bereich des primalen bzw. dualen Problems bezeichnen wir wieder mit M_P und M_D . Aufgrund des schwachen Dualitätssatzes 12.2 wissen wir, dass $c'x \leq b'y$ gilt. Man bilde nun die schiefsymmetrische Matrix

$$\begin{pmatrix} 0_n & A' & -c \\ -A & 0_m & b \\ c' & -b' & 0 \end{pmatrix},$$

wobei 0_n (und analog 0_m) eine Nullmatrix mit n Zeilen und Spalten ist. Nach Satz 15.7 gibt es ein $\begin{pmatrix} \bar{x} \\ \bar{y} \\ t \end{pmatrix} \in \mathbb{R}^{n+m+1}$ mit folgenden Eigenschaften:

$$\bar{x} \geq 0, \bar{y} \geq 0, t \geq 0, \tag{15.13}$$

$$A'\bar{y} - ct \geq 0, -A\bar{x} + bt \geq 0, \tag{15.14}$$

$$c'\bar{x} - b'\bar{y} \geq 0, \tag{15.15}$$

$$A'\bar{y} - ct + \bar{x} > 0, -A\bar{x} + bt + \bar{y} > 0, \tag{15.16}$$

$$c'\bar{x} - b'\bar{y} + t > 0. \tag{15.17}$$

Wir unterscheiden nun die Fälle $t > 0$ und $t = 0$ (vgl. [Goldmann und Tucker, 1956, Lemma 4 und 5]). Für $t > 0$ erhält man

Satz 15.8 *Ist $t > 0$, so gibt es Optimallösungen x^* von (P) und y^* von (D) mit $c'x^* = b'y^*$ und $A'y^* + x^* > c$, $Ax^* - y^* < b$.*

Beweis: Setze $x^* := \frac{1}{t}\bar{x}$, $y^* := \frac{1}{t}\bar{y}$. Dann sind wegen (15.13) und (15.14) die Vektoren x^* und y^* zulässig. Aus (15.15) folgt zusammen mit Satz 12.2 die Gleichheit $c'x^* = b'y^*$. Daher sind x^* und y^* optimal. Die restliche Aussage ergibt sich aus (15.16). ■

Satz 15.9 *Ist $t = 0$, dann gilt*

1. (P) und (D) besitzen nicht beide zulässige Punkte.
2. Ist $M_P \neq \emptyset$ ($M_D \neq \emptyset$), so ist die primale (duale) Zielfunktion nicht beschränkt.
3. Keines der beiden Probleme besitzt eine endliche Optimallösung.

Beweis:

1. Gibt es ein $x \in M_P$ und $y \in M_D$, so folgt aus (15.17) und (15.14) $b'\bar{y} < c'\bar{x} \leq (A'y)\bar{x} = y'A\bar{x} \leq 0$. Andererseits gilt

$$0 \leq x'A'\bar{y} = (Ax)'\bar{y} \leq b'\bar{y}. \quad (15.18)$$

Somit erhält man einen Widerspruch.

2. Ist $x \in M_P$, so ist $x + \lambda\bar{x} \in M_P$ für alle $\lambda \geq 0$, denn $Ax \leq b$ und $A\bar{x} \leq 0$. Für die Zielfunktion gilt $c'(x + \lambda\bar{x}) = c'x + \lambda c'\bar{x}$. Da nach (15.17) und (15.18) $c'\bar{x} > b'\bar{y} \geq 0$ gilt, ist der Zielfunktionswert nicht nach oben beschränkt. Die dritte Aussage folgt aus 2. ■

Der Dualitätssatz 13.1 folgt nun unmittelbar aus diesen beiden Sätzen, denn besitzt etwa (P) eine endliche Optimallösung, so folgt $t \neq 0$ und aus (15.13) daher $t > 0$ und damit Satz 15.8.

Kapitel 16

Das duale Simplexverfahren

Manchmal ist es vorteilhaft, nicht das gegebene lineare Programm zu lösen, sondern das dazu duale. Dies wird man z. B. immer dann tun, wenn das Ausgangsproblem schon in dual zulässiger Form vorliegt, was bei Unterproblemen in Algorithmen zur konvexen oder ganzzahligen Optimierung oft der Fall ist (Schnittebenenverfahren). Auch im Falle beschränkter Variabler ist ein duales Vorgehen günstig. Der Dualitätssatz der linearen Optimierung garantiert dann, dass der Zielfunktionswert der dual-optimalen Lösung mit dem der gegebenen Aufgabe übereinstimmt. Ferner lässt sich dem Tableau leicht auch die zugehörige primale Optimallösung entnehmen.

Wir gehen aus von einem Paar zueinander dualer linearer Optimierungsaufgaben (P) und (D) :

$$(P) \max\{c'x \mid Ax \leq b, x \geq 0\},$$

$$(D) \min\{b'y \mid A'y \geq c, y \geq 0\}.$$

(D) ist *(dual-)zulässig*, wenn $c \leq 0$ ist, denn dann entspricht die duale Basislösung $y_N = 0, y_B = -c$ einer zulässigen Ecke des dualen Polyeders. Da die Minimierung von $b'y$ der Maximierung von $(-b)'y$ entspricht, liegt eine Optimallösung vor, wenn $-b \leq 0$, d. h. $b \geq 0$ ist.

Speichert man A, b und c wie üblich im Simplextableau, so gilt:

1. Das Tableau ist primal zulässig, wenn $b \geq 0$.
2. Das Tableau ist dual zulässig, wenn $c \leq 0$.
3. Das Tableau gibt eine Optimallösung wieder, wenn $b \geq 0$ und $c \leq 0$ gilt.

Da die Matrix A beim Übergang zum dualen Problem transponiert wird, wechseln im dualen Simplexverfahren Zeilen und Spalten ihre Rolle. Ist eine Lösung nicht dual-optimal, so wird zunächst eine Austauschzeile bestimmt, etwa durch einen Index r mit

$$b_r = \min_{1 \leq i \leq m} b_i.$$

Ist $a_{rj} \geq 0$ für $1 \leq j \leq n$, so gibt es für das duale Problem keine endliche Lösung. Das heißt aber nach Satz 12.3, das primale Problem ist unlösbar. Anderenfalls bestimmt man die Austauschspalte s durch

$$\frac{c_s}{a_{rs}} = \min \left\{ \frac{c_j}{a_{rj}} \mid a_{rj} < 0, 1 \leq j \leq n \right\},$$

denn für das Restriktionensystem $-A'y \leq -c$ musste das folgende Minimum bestimmt werden:

$$\min \left\{ \frac{-c_j}{-a_{rj}} \mid -a_{rj} > 0, 1 \leq j \leq n \right\} = \min \left\{ \frac{c_j}{a_{rj}} \mid a_{rj} < 0, 1 \leq j \leq n \right\}.$$

Entartungen können prinzipiell analog wie beim primalen Simplexverfahren durch eine lexikographische Spaltenauswahlregel behoben werden. Man kann sich ferner aufgrund der gleichen Argumente wie oben leicht überlegen, dass sich an der Pivotoperation nichts ändert. Damit erhält man folgendes duale Simplexverfahren:

Algorithmus 2 *Algorithmus zur Lösung von (P) durch das duale Simplexverfahren.*

(P) sei dual zulässig, d. h. $a_{0j} \leq 0$ ($1 \leq j \leq n$)

Anfangsdaten: (a_{ij}) $0 \leq i \leq m$, $0 \leq j \leq n$

Der Vektor B enthält die Indizes der den Zeilen zugeordneten Variablen, der Vektor N enthält die Indizes der den Spalten zugeordneten Variablen (=dualer Basisvektor).

1. Ist $a_{i0} \geq 0$ für $1 \leq i \leq m$, so ist die Lösung optimal. Setze

$$x_{B(i)} := a_{i0} \quad (1 \leq i \leq m), \quad x_{N(j)} := 0 \quad (0 \leq j \leq n) \quad \text{und} \quad z := -a_{00}.$$

Anderenfalls gehe zu 2.

2. Wähle r so, dass $a_{r0} = \min_{1 \leq i \leq m} a_{i0}$.

3. Ist $a_{rj} \geq 0$ für $1 \leq j \leq n$, so besitzt (P) keine Lösung. Terminiere.
Anderenfalls gehe zu 4.

4. Wähle s so, dass

$$\frac{a_{0s}}{a_{rs}} = \min \left\{ \frac{a_{0j}}{a_{rj}} \mid a_{rj} < 0, 1 \leq j \leq n \right\}.$$

5. Vertausche das s -te Element von N mit dem r -ten Element von B und führe folgende Pivotoperation durch

$$\tilde{a}_{rs} := \frac{1}{a_{rs}}$$

$$\tilde{a}_{rj} := \frac{a_{rj}}{a_{rs}} \quad (0 \leq j \leq n; j \neq s)$$

$$\tilde{a}_{is} := -\frac{a_{is}}{a_{rs}} \quad (0 \leq i \leq m; i \neq r)$$

$$\tilde{a}_{ij} := a_{ij} - \frac{a_{is}a_{rj}}{a_{rs}} \quad (0 \leq i \leq m, 0 \leq j \leq n; i \neq r, j \neq s).$$

6. Setze $a_{ij} := \tilde{a}_{ij}$ ($0 \leq i \leq m, 0 \leq j \leq n$) und gehe zu 1.

Beispiel 16.1 (vgl. Abschnitt 5.1)

$$\begin{aligned}
 & \text{Maximiere } -x_1 - 2x_2 && \text{unter den Restriktionen} \\
 & x_1 + x_2 \geq 3 \\
 & x_2 \geq 2 \\
 & -x_1 + x_2 \leq 3 \\
 & x_1 - x_2 \leq 3 \\
 & x_i \geq 0 \text{ für } i = 1, 2.
 \end{aligned}$$

Ausgangstableau

		x_1	x_2
	0	-1	-2
x_3	-3	-1	-1
x_4	-2	0	-1
x_5	3	-1	1
x_6	3	1	-1

Das Ausgangstableau ist dual zulässig, nicht aber primal zulässig. Wir bestimmen $r = 1, s = 1$ und führen eine Pivotoperation durch:

Tableau 1

		x_3	x_2
	3	-1	-1
x_1	3	-1	1
x_4	-2	0	-1
x_5	6	-1	2
x_6	0	1	-2

Ein weiterer dualer Simplexschritt führt auf

Tableau 2

		x_3	x_4
	5	-1	-1
x_1	1	-1	1
x_2	2	0	-1
x_5	2	-1	2
x_6	4	1	-2

Tableau 2 ist nun primal und dual zulässig, daher dual optimal. Die Lösung des gegebenen Problems lautet

$$x_1 = 1, x_2 = 2, c'x = -5.$$

Wie dieses Beispiel zeigt, kann man durch Anwendung des dualen Simplexverfahrens oft die aufwendige Bestimmung einer zulässigen Ausgangslösung für das primale Problem vermeiden.

Liegt ein Problem vor, das nicht dual zulässig ist und das durch das duale Simplexverfahren gelöst werden soll, so kann man analoge Verfahren wie in Kapitel 5 zur Bestimmung einer dual zulässigen Lösung verwenden.

Das duale Simplexverfahren kann auch mit Gewinn zur Lösung von linearen Optimierungsaufgaben mit beschränkten Variablen herangezogen werden (vgl. Wagner [1958]):

Gegeben sei das lineare Programm

$$\max c'x \text{ unter } Ax \leq b, 0 \leq x_j \leq d_j \quad (1 \leq j \leq n).$$

Für jede Spalte j mit $c_j > 0$ gehen wir durch die Transformation $T(j)$ zur Komplementärvariablen über (vgl. Abschnitt 8.2).

$$T(j): \begin{array}{ll} -c_0 & := -c_0 - d_j \cdot c_j & c_j & := -c_j \\ b_i & := b_i - d_j \cdot a_{ij} & a_{ij} & := -a_{ij} \end{array} \quad (1 \leq i \leq m).$$

Falls alle Variablen beschränkt sind, erhält man auf diese Weise eine dual zulässige Ausgangslösung. Anderenfalls muss man sich eventuell eine dual zulässige Ausgangslösung (etwa durch ein duales M-Verfahren) erzeugen. Nun führt man das duale Simplexverfahren durch. Gilt in dessen Optimallösung für alle Basisvariablen

$$x_{B(i)} \leq d_{B(i)} \quad (1 \leq i \leq m),$$

so ist diese Lösung optimal. Ist jedoch ein $x_{B(i)} > d_{B(i)}$, so geht man zur Komplementärvariablen über. Dadurch wird $b_{B(i)}$ negativ. Das Tableau bleibt jedoch dual zulässig, ist aber nicht mehr optimal. Daher führt man erneut duale Simplexschritte durch bis die Optimallösung erreicht wird.

Man kann auch zunächst das LP ohne obere Schranken lösen und dann von der Optimallösung ausgehend zum dualen Problem übergehen, falls eine Schranke verletzt ist.

Beispiel 16.2

$$\begin{array}{ll} \text{Maximiere} & -x_1 + 4x_2 \\ \text{unter} & x_1 - x_2 \leq 2 \\ & -x_1 + x_2 \leq 3 \\ & x_1 + x_2 \geq 3 \\ & x_2 \leq 4 \\ & x_j \geq 0 \quad (j = 1, 2). \end{array}$$

Das Problem ist weder primal noch dual zulässig. Gehen wir aber zu $\bar{x}_2 = 4 - x_2$ über, so erhalten wir ein dual zulässiges Problem

$$\begin{array}{ll} \text{Maximiere} & 16 - x_1 - 4\bar{x}_2 \\ \text{unter} & x_1 + \bar{x}_2 \leq 6 \\ & -x_1 - \bar{x}_2 \leq -1 \\ & -x_1 + \bar{x}_2 \leq 1 \\ & 0 \leq x_1 \\ & 0 \leq \bar{x}_2 \leq 4 \end{array}$$

Damit liegt ein dual zulässiges Problem vor, zu dem das folgende Tableau gehört

	x_1	\bar{x}_2	
	-16	-1	-4
x_3	6	1	1
x_4	-1	-1	-1
x_5	1	-1	1

Ein dualer Simplexschritt mit $r = 2, s = 1$ liefert nun

	x_4	\bar{x}_2
	-15	-1 -3
x_3	5	1 0
x_1	1	-1 1
x_5	2	-1 2

Damit ist die Optimallösung erreicht. Sie lautet $x_1 = 1, \bar{x}_2 = 0$, also $x_1 = 1, x_2 = 4$ und $c'x = 15$.

Kapitel 17

Komplexität der linearen Optimierung

17.1 Komplexität des Simplexverfahrens

Wir wollen nun der Frage nachgehen, wieviele Simplexiterationen nötig sind, bis die Optimallösung eines linearen Programmes erreicht ist. Aufgrund vieler tausender linearer Programme, die im Laufe der Zeit gelöst wurden, kam G.Dantzig zu der Vermutung, dass etwa $O(m \log n)$ Simplexiterationen zur Lösung eines Problems mit m Restriktionen und n Variablen ausreichen. Groß war daher die Überraschung, als 1972 Klee und Minty [33] nachwiesen, dass es eine Klasse von Problemen - verzerrte Würfel - gibt, bei denen alle 2^n Ecken im Simplexverfahren durchlaufen werden, bevor die Optimallösung erreicht wird. Dabei wurde angenommen, dass im Simplexverfahren die Auswahlspalte durch den größten Zielfunktionskoeffizienten bestimmt wird. Andere Autoren konstruierten in der Folge auch für weitere Spaltenauswahlregeln Beispiellklassen, bei denen im Simplexverfahren alle Ecken durchlaufen werden. Die Frage, ob es eine Spalten- und Zeilenauswahlregel für das Simplexverfahren gibt, dass es nach polynomial vielen Schritten endet, ist bis heute ungelöst.

Dantzigs's empirische Beobachtung wird durch Untersuchungen zum mittleren Verhalten des Simplexalgorithmus untermauert. Borgwardt zeigte 1977 in seiner Dissertation als Erster, dass bei Verwendung spezieller Spalten- und Zeilenauswahlregeln das Simplexverfahren im Mittel nur polynomial viele Iterationen braucht. Eine ausführliche Diskussion des mittleren Verhaltens des Simplexverfahrens findet sich in der Monographie Borgwardt [7]. Später wurde von Adler und Megiddo [1] gezeigt, dass die erwartete Anzahl von Pivotoperationen im Simplexverfahren $O(\min\{m^2, n^2\})$ beträgt.

17.2 Ellipsoidverfahren

Komplexitätsaussagen über das Simplexverfahren sagen noch nichts darüber aus, ob lineare Programme in polynomialer Zeit gelöst werden können, denn es könnte ja ein ganz anderes Verfahren geben, mit dem LP's gelöst werden. Lange Jahre blieb die Frage, ob es für LP's ein polynomiales Lösungsverfahren gibt, ungelöst, bis 1979 Khachyan [32] in einer aufsehenerregenden Arbeit nachwies, dass lineare Programme tatsächlich der Komplexitätsklasse \mathcal{P} angehören. Das heisst,

kodiert man die Eingabedaten eines linearen Programms binär und hat dieser Eingabestring die Länge L , dann gibt es ein Polynom p in L , so dass jedes lineare Programm in höchstens $p(L)$ Schritten gelöst werden kann. Die Hoffnungen für die Praxis, die man mit Khachyan's Ellipsoidverfahren verband, haben sich jedoch nicht erfüllt: nach wie vor werden Simplexalgorithmen zur Lösung linearer Programme herangezogen und sind numerisch den Ellipsoidverfahren in der Praxis weit überlegen. Eine wirklich praktisch brauchbare Alternative zu Simplexverfahren wurde jedoch durch die ebenfalls polynomialen inneren Punktmethoden von Karmarkar [30] gefunden. Wir werden innere Punkteverfahren im nächsten Kapitel besprechen.

Im folgenden werden wir die Idee von Ellipsoidverfahren skizzieren und halten uns dabei an die Ausführungen von Gács und Lovász [19] und Schrader [41]. Zunächst halten wir fest

Satz 17.1 *Die Optimierung einer linearen Zielfunktion über einer polyedrischen Menge kann auf die Aufgabe, für eine polyedrische Menge einen zulässigen Punkt zu bestimmen, zurückgeführt werden.*

Beweis: Besitzt sowohl das primale als auch das duale Problem zulässige Punkte, so erfüllen die Optimallösungen das System

$$Ax \leq b, x \geq 0, A'y \geq c, y \geq 0, b'y \geq c'x.$$

Daher gilt für die primalen und dualen Optimallösungen $Dz \leq d$ mit $z' = (x', y')$,

$$D' = \begin{pmatrix} A' & 0 & -I & 0 & -c \\ 0 & -A & 0 & -I & b \end{pmatrix}$$

und

$$d' = (b', -c', 0, 0, 0).$$

Wenn eines der beiden zueinander dualen Probleme keine endliche Lösung besitzt, so hat das dazu duale Problem zulässige Lösungen. Daher kann auch der Fall, dass das gegebene LP keine endliche Lösung besitzt, auf die Untersuchung zurückgeführt werden, ob eine polyedrische Menge zulässige Punkte besitzt.

Im folgenden werden wir also nur die Aufgabe betrachten:

Hat ein gegebene polyedrische Menge $P(A, b) := \{x | Ax \leq b\}$ zulässige Punkte?

Dabei nehmen wir an, dass A eine $(m \times n)$ -Matrix mit vollem Zeilenrang ist. Khachyan's Idee war es nun, anstelle dieser Aufgabe das Problem

$$\text{Finde ein } x \in P^<(A, b) := \{x | Ax < b\} \tag{17.1}$$

zu betrachten. Nehmen wir an, wir haben einen polynomialen Algorithmus zur Lösung dieser Aufgabe. Dann können wir auf folgende Weise auch prüfen, ob es ein $x \in P(A, b)$ gibt: Zunächst prüft man, ob das System aus den ersten i strengen Ungleichungen konsistent ist. Ist dies der Fall für $i = m$, so hat man wegen $P^<(A, b) \subseteq P(A, b)$ auch einen zulässigen Punkt für das Ausgangsproblem gefunden. Zeigt sich jedoch, dass durch Hinzufügen der i -ten Restriktion $a_i'x < b_i$ das System inkonsistent wird, so muss diese Restriktion in $P(A, b)$ mit Gleichheit erfüllt sein, falls $P(A, b)$ nicht leer ist. Daher kann man aus dieser Restriktion eine Variable eliminieren,

setzt diese in alle anderen Restriktionen ein und beginnt von Neuem mit der Prüfung, ob das vorliegende strenge Ungleichungssystem konsistent ist. Nach höchstens $O(m^2)$ Tests vom Typ (17.1) kann somit entschieden werden, ob $P(A, b)$ leer ist oder nicht. Kann $P(A, b) \neq \emptyset$ in polynomialer Zeit getestet werden, dann kann folglich auch $P(A, b) \neq \emptyset$ in polynomialer Zeit getestet werden.

Im folgenden wollen wir annehmen, dass alle Eingangsdaten in A und b ganzzahlig sind. Dies ist für rationale Daten keine Einschränkung, irrationale Daten können sowieso nicht endlich binär kodiert werden. Die Länge L der Eingangsdaten ist beschränkt durch

$$L = \sum_{i=1}^m \sum_{j=1}^n \log(|a_{ij}| + 1) + \sum_{i=1}^m \log(|b_i| + 1) + \log nm + 1.$$

Als erstes wollen wir nun zeigen, dass die Ecken der polyedrischen Menge $P(A, b)$ in einem Würfel mit der Seitenlänge $2^{L+1}/n$ liegen:

Lemma 17.2 *Falls $P(A, b) \neq \emptyset$, so erfüllt jede Ecke v der polyedrischen Menge $P(A, b)$*

$$\max_{1 \leq j \leq n} |v_j| < 2^L/n$$

und hat rationale Koordinaten, deren Nenner höchstens 2^L ist.

Zum Beweis dieses Lemmas verwenden wir die *Hadamard'sche Ungleichung*:

Lemma 17.3 *Hadamard'sche Ungleichung*

Sei $A = (a_{ij})$ eine reelle, quadratische $(m \times m)$ -Matrix. Dann gilt

$$|\det(A)| \leq \prod_{j=1}^m \left(\sum_{i=1}^m a_{ij}^2 \right)^{1/2}. \quad (17.2)$$

Der Beweis der Hadamard'schen Ungleichung ergibt sich aus der Tatsache, dass $|\det(A)|$ das Volumen des Parallelepipeds ist, das von den Spaltenvektoren der Matrix A aufgespannt wird. Der j -te Spaltenvektor hat die Länge

$$l_j := \left(\sum_{i=1}^m a_{ij}^2 \right)^{1/2}.$$

Das Volumen des Parallelepipeds ist aber durch das Volumen eines Quaders mit den Seitenlängen l_j nach oben beschränkt, woraus sich unmittelbar die Hadamard'sche Ungleichung ergibt.

Wir beweisen nun Lemma 17.2:

Beweis: Es seien v_j , $j = 1, 2, \dots, n$, die Koordinaten der Ecke v . Wenn v eine Ecke von $P(A, b)$ ist, gibt es eine Basismatrix A_B , so dass $A_B v = b$ gilt. Nach der Cramer'schen Regel ergibt sich jede Koordinate von v als Quotient zweier Determinanten $\det(A_B^j)$ und $\det A_B$, wobei A_B^j aus A_B hervorgeht, indem man die j -te Spalte durch den Vektor b ersetzt. Aufgrund ganzzahliger Ausgangsdaten sind alle Einträge in den Determinanten $\det A_B$ und $\det(A_B^j)$ ganzzahlig. Daher sind auch $\det A_B$ und $\det(A_B^j)$ ganzzahlig. Insbesondere gilt $|\det A_B| \geq 1$.

Es sei nun $A_B^j = (a_{ij})$. Da für jede Spalte der Matrix (a_{ij}) gilt

$$\sum_{i=1}^m a_{ij}^2 \leq \prod_{i=1}^m |a_{ij} + 1|^2,$$

folgt aus der Hadamard'schen Ungleichung und der Definition von L :

$$|\det(A_B^j)| \leq \prod_{j=1}^m \left(\sum_{i=1}^m a_{ij}^2 \right)^{1/2} < 2^L/mn < 2^L/n.$$

Mit $|\det A_B| \geq 1$ folgt also $\max_{1 \leq j \leq n} |v_j| < 2^L/n$. Da auch für $\det A_B$ die Abschätzung

$$|\det A_B| < 2^L/mn$$

gilt, ist der Nenner jeder Komponente höchstens 2^L . Damit ist Lemma 17.2 bewiesen. ■

Nun zeigen wir eine untere Schranke für das Volumen der Menge der zulässigen Punkte innerhalb eines Würfels der Kantenlänge $2^{L+1}/n$, vorausgesetzt die zulässige Menge ist nicht leer.

Lemma 17.4 *Ist $P^<(A, b) \neq \emptyset$, dann hat die Menge der zulässigen Punkte innerhalb des Würfels $|x_j| \leq 2^L/n$ mindestens das Volumen $2^{-(n+1)L}$.*

Beweis: Die polyedrische Menge $P^<(A, b)$ sei nicht leer. Dann hat die Menge $P(A, b)$ mindestens eine Ecke $v = (v_1, v_2, \dots, v_n)$, da sie keine Gerade enthält (vgl. Satz 3.2). Für die Koordinaten der Ecke v gilt nach Lemma 17.2 $|v_j| < 2^L/n$. Daher hat das Polytop $M := P(A, b) \cap \{x \mid |x_j| \leq 2^L/n\}$ einen inneren Punkt und damit auch $n+1$ Ecken v^0, \dots, v^n , die nicht auf einer Hyperebene liegen. Damit ist das Volumen von M mindestens so groß wie das Volumen des Simplex, das durch v^0, \dots, v^n aufgespannt wird, also mindestens

$$\frac{1}{n!} \left| \det \begin{pmatrix} 1 & 1 & \dots & 1 \\ v^0 & v^1 & \dots & v^n \end{pmatrix} \right|.$$

Nach Lemma 17.2 gilt mittels $v^j = u^j/D_j$, ganzzahligen Werten D_j und ganzzahligen Vektoren u^j

$$\left| \det \begin{pmatrix} 1 & \dots & 1 \\ v^0 & \dots & v^n \end{pmatrix} \right| = \frac{1}{|D_0| \dots |D_n|} \left| \det \begin{pmatrix} D_0 & \dots & D_n \\ u^0 & \dots & u^n \end{pmatrix} \right| \geq \frac{1}{|D_0| \dots |D_n|} \geq 2^{-(n+1)L} n^{n+1}.$$

Damit ist aber das Volumen der Menge der zulässigen Punkte mindestens

$$\text{vol}(M) \geq \frac{2^{-(n+1)L} n^{n+1}}{n!} > 2^{-(n+1)L}.$$

■

Die Idee des Ellipsoidverfahrens ist nun die folgende. Wenn $P^<(A, b)$ nicht leer ist, dann enthält nach Lemma 17.4 die Kugel E_0 mit dem Mittelpunkt 0 und Radius $2^L/\sqrt{n}$ eine zulässige Menge M von $P^<(A, b)$ mit einem Volumen, das größer oder gleich $2^{-(n+1)L}$ ist. Zunächst testet man, ob der Mittelpunkt x_0 dieser Kugel zulässig für $P^<(A, b)$ ist. Ist dies der Fall, so ist

man fertig. Andernfalls gibt es eine Restriktion $a'_i x < b_i$, die von x_0 verletzt wird. Man betrachtet nun die Halbkugel $E_0 \cap \{x \mid a'_i x \leq a'_i x_0\}$ und legt um diese Halbkugel ein neues Ellipsoid E_1 mit minimalem Volumen und neuen Mittelpunkt x_1 . Allgemein hat man ein Ellipsoid E_r mit Mittelpunkt x_r gegeben, das M enthält. Man prüft, ob x_r zulässig für $P^<(A, b)$ ist und konstruiert im Falle der Unzulässigkeit dieses Punktes ein neues Ellipsoid E_{r+1} mit kleinerem Volumen. Sobald das Volumen von E_r unter die im Lemma 17.4 angegebene Schranke gefallen ist, weiss man, dass E_r keinen zulässigen Punkt von $P^<(A, b)$ enthält, also $P^<(A, b)$ leer ist. Wenn man nun zeigen kann, dass nach polynomial vielen Schritten entweder ein zulässiger Punkt für $P^<(A, b)$ gefunden wird oder das Volumen des zuletzt betrachteten Ellipsoids unter die Schranke $2^{-(n+1)L}$ gefallen ist, dann hat man ein polynomiales Verfahren zur Lösung von linearen Optimierungsaufgaben erreicht.

Die im Verfahren auftretenden Ellipsoide mit dem Mittelpunkt x_0 werden durch die Beziehung

$$E := \{x \in \mathbb{R}^n \mid (x - x_0)' C^{-1} (x - x_0) \leq 1\}$$

mit einer reellen, symmetrischen, positiv definiten $(n \times n)$ -Matrix C beschrieben. Wird nun dieses Ellipsoid mit dem Halbraum $\{x \in \mathbb{R}^n \mid a'x \leq a'x_0\}$ ($a \neq 0$) geschnitten, so erhält man das Halbellipsoid

$$\frac{1}{2}E_a := E \cap \{x \in \mathbb{R}^n \mid a'(x - x_0) \leq 0\}.$$

Nun definiert man ein neues Ellipsoid E' mit Mittelpunkt \bar{x}_0 durch

$$\bar{x}_0 := x_0 - \frac{1}{n+1} C \frac{a}{\sqrt{a'Ca}}, \quad (17.3)$$

$$\bar{C} := \frac{n^2}{n^2-1} \left(C - \frac{2}{n+1} \frac{(Ca) \cdot (Ca)'}{a'Ca} \right). \quad (17.4)$$

Offenbar ist \bar{C} wieder symmetrisch. Aber \bar{C} ist auch wieder positiv definit, denn es gilt nach der Cauchy-Schwarz'schen Ungleichung angewandt auf das Skalarprodukt $\langle x, y \rangle := x'Cy$ mit der positiv definiten Matrix C :

$$(x'Ca)(a'Cx) = \langle x, a \rangle^2 \leq \|a\|^2 \cdot \|x\|^2.$$

Daher ist für beliebiges $x \neq 0$, $n \geq 2$,

$$\begin{aligned} x'\bar{C}x &= \frac{n^2}{n^2-1} \left(x'Cx - \frac{2}{n+1} \frac{(x'Ca) \cdot (a'Cx)}{a'Ca} \frac{x'Cx}{x'Cx} \right) \\ &= \frac{n^2}{n^2-1} (x'Cx) \left(1 - \frac{2}{n+1} \frac{(x'Ca) \cdot (a'Cx)}{(a'Ca) \cdot (x'Cx)} \right) > 0. \end{aligned}$$

Somit wird durch \bar{C} und \bar{x}_0 wieder ein Ellipsoid beschrieben. Es gilt:

Lemma 17.5 *Das Halbellipsoid $\frac{1}{2}E_a$ ist im Ellipsoid E' enthalten.*

Beweis: Da die Aussage des Lemmas invariant gegenüber affinen Transformationen ist, können wir o.B.d.A. annehmen, dass $x_0 = 0$ und $a = (-1, 0, \dots, 0)'$ gilt und das durch C beschriebene Ellipsoid die Einheitskugel ist, d.h. C ist die Einheitsmatrix. Dann erhalten wir nach (17.3)

$$\bar{x}_0 = \left(\frac{1}{n+1}, 0, \dots, 0 \right)$$

und

$$\bar{C} = \text{diag} \left(\frac{n^2}{(n+1)^2}, \frac{n^2}{n^2-1}, \dots, \frac{n^2}{n^2-1} \right). \quad (17.5)$$

Für $x \in \frac{1}{2}E_a$ gilt nun $\|x\| \leq 1$ und $1 \geq x_1 = -a'x \geq 0$. (Die letzte Ungleichung ergibt sich aus der schneidenden Hyperebene). Daher ist

$$\begin{aligned} (x - \bar{x}_0)' \bar{C}^{-1} (x - \bar{x}_0) &= x' \bar{C}^{-1} x - 2x' \bar{C}^{-1} \bar{x}_0 + \bar{x}_0' \bar{C}^{-1} \bar{x}_0 \\ &= \frac{n^2-1}{n^2} \|x\|^2 + \frac{2(n+1)}{n^2} x_1^2 - 2 \frac{n+1}{n^2} x_1 + \frac{1}{n^2} \\ &= \frac{n^2-1}{n^2} (\|x\|^2 - 1) + \frac{2n+2}{n^2} x_1 (x_1 - 1) + 1 \leq 1. \end{aligned}$$

Daher liegt dieser Punkt im neuen Ellipsoid E' . ■

Die obigen Überlegungen zusammengefasst führen auf folgenden Algorithmus

Algorithmus 3

Start $x_0 := 0$, $C_0 := \frac{2^{2L}}{n} I$, $r := 0$.

Schritt 1 Falls $r > 6n(n+1)L$, stop: das Polyeder $P^<(A, b)$ ist leer.

Schritt 2 Falls $\max_{1 \leq i \leq m} (a'_i x_r - b_i) < 0$, dann ist x_r ein zulässiger Punkt. Stop.

Schritt 3 Wähle ein i mit $a'_i x_r \geq b_i$.

Schritt 4 Setze

$$x_{r+1} := x_r - \frac{1}{n+1} C_r \frac{a_i}{\sqrt{a'_i C_r a_i}}, \quad (17.6)$$

$$C_{r+1} := \frac{n^2}{n^2-1} \left(C_r - \frac{2}{n+1} \frac{(C_r a_i) \cdot (C_r a_i)'}{a'_i C_r a_i} \right), \quad (17.7)$$

$$r := r + 1, \quad (17.8)$$

und gehe zu Schritt 1.

Um die Korrektheit dieses Verfahrens zu beweisen, zeigen wir zunächst noch das folgende Lemma, das eine Aussage darüber macht, wie schnell das Volumen der im Algorithmus erzeugten Ellipsoide schrumpft.

Lemma 17.6 *Es gilt*

$$\text{vol}(E') = \left(\frac{n^2}{n^2-1} \right)^{(n-1)/2} \frac{n}{n+1} \text{vol}(E). \quad (17.9)$$

Dabei ist der Schrumpfungsfaktor kleiner als

$$e^{-(1/2(n+1))}. \quad (17.10)$$

Beweis: Da affine Transformationen das Verhältnis zweier Volumina nicht ändern, können wir annehmen, dass E die Einheitskugel mit Mittelpunkt 0 ist. Ferner können wir $a = (-1, 0, \dots, 0)'$ annehmen. Dann ist nach (17.5)

$$\bar{C} = \text{diag} \left(\frac{n^2}{(n+1)^2}, \frac{n^2}{n^2-1}, \dots, \frac{n^2}{n^2-1} \right).$$

Bekanntlich ist das Volumen eines Ellipsoids $\{x \mid x' C^{-1} x \leq 1\}$ gegeben durch $c(n) \cdot \sqrt{\det(C)}$ mit einer von der Dimension abhängigen Konstanten $c(n)$. Daher gilt

$$\begin{aligned} \text{vol}(E') &= \frac{\sqrt{\det(\bar{C})}}{\sqrt{\det C}} \cdot \text{vol}(E) = \sqrt{\det(\bar{C})} \cdot \text{vol}(E) \\ &= \frac{n}{n+1} \left(\frac{n^2}{n^2-1} \right)^{(n-1)/2} \text{vol}(E). \end{aligned}$$

Aus

$$\frac{n^2}{n^2-1} = 1 + \frac{1}{n^2-1} < e^{1/(n^2-1)}$$

und

$$\frac{n}{n+1} = 1 - \frac{1}{n+1} < e^{-1/(n+1)}$$

folgt die zweite Aussage des Lemmas. ■

Nun kann man leicht den folgenden fundamentalen Satz beweisen

Satz 17.7 *Wenn der Algorithmus nicht in $6n(n+1)L$ Schritten stoppt, dann besitzt das Polyeder $P^<(A, b)$ keine zulässigen Punkte.*

Beweis: Nehmen wir an, der Algorithmus stoppt nicht nach $k := 6n(n+1)L$ Schritten und $P^<(A, b)$ besitzt innere Punkte. Die Menge M der zulässigen Punkte geschnitten mit dem Würfel $\{x \mid |x_j| \leq 2^L/n\}$ hat nach Lemma 17.4 mindestens das Volumen $2^{-(n+1)L}$. Der Würfel mit der halben Kantenlänge $2^L/n$ ist aber in der Kugel E_0 mit Mittelpunkt 0 und Radius $2^L/\sqrt{n}$ enthalten. Nach Lemma 17.5 gilt $M \subset E_k$. Die Kugel E_0 ist ihrerseits in einem Würfel mit der Kantenlänge $2^{L+1}/\sqrt{n}$ enthalten. Daher ist $\text{vol}(E_0) < \frac{2^{n(L+1)}}{(\sqrt{n})^n}$. Somit gilt wegen $L \geq 2$ und wegen Lemma 17.6

$$\text{vol}(E_k) < e^{-(k/2(n+1))} \text{vol}(E_0) < e^{-3nL} \frac{2^{n(L+1)}}{(\sqrt{n})^n} < 2^{-2nL+n} < 2^{-(n+1)L},$$

ein Widerspruch. ■

Um zu beweisen, dass durch das obige Verfahren tatsächlich in polynomialer Zeit die Aufgabe gelöst wird, ist noch eine zusätzliche Überlegung notwendig. Im Schritt 4 des Algorithmus werden Quadratwurzeln berechnet, die auf irrationale Zahlen führen können. Khachyan konnte aber zeigen, dass Lemma 17.5 auch gilt, wenn man alle Ellipsoide um den Faktor $2^{1/8n^2}$ aufbläht und die Berechnungen so durchführt, dass nur $23L$ Stellen vor dem Komma und $38nL$ Stellen nach dem Komma berücksichtigt werden.

Eine ausführliche Diskussion des historischen Hintergrunds der Ellipsoidmethode, die u.a. auf grundlegenden Arbeiten von Judin und Nemirovkiĭ [28] aufbaut, sowie Varianten und Weiterentwicklungen dieser Methode finden sich in Schrader [41].

Kapitel 18

Innere Punkte Verfahren: Theorie

18.1 Grundidee zu inneren Punkteverfahren

Gegeben seien die beiden zueinander dualen Programme

$$(P) \quad \max \{c'x \mid Ax \leq b, x \geq 0\}$$

und

$$(D) \quad \min \{b'y \mid A'y \geq c, y \leq 0\}.$$

Um festzustellen, ob es eine endliche Optimallösung gibt und diese anzugeben, kann man nach Satz 15.8 und Satz 15.9 die Restriktionen der beiden Probleme zu

$$D := \begin{pmatrix} 0 & A' & -c \\ -A & 0 & b \\ c' & -b' & 0 \end{pmatrix}$$

zusammenfassen und eine Lösung $w = (x', y', t)$ mit

$$Dw \geq 0, w \geq 0, Dw + w > 0 \tag{18.1}$$

bestimmen. Gilt in dieser Lösung $t > 0$, dann sind $\frac{1}{t}x$ bzw. $\frac{1}{t}y$ endliche Optimallösungen von (P) und (D) . Ist aber $t = 0$, dann hat eines der beiden Probleme keine zulässigen Punkte und damit gibt es auch keine endliche Optimallösung.

Mit Hilfe eines inneren Punkteverfahrens kann nun in polynomial vielen Schritten eine Lösung w von (18.1) gefunden werden. Damit können die gegebenen linearen Programme in polynomialer Zeit gelöst werden. Um (18.1) zu lösen, geht man von einem selbstdualen linearen Programm der Form

$$(S_0) \quad \min\{\bar{q}'\bar{w} \mid \bar{D}\bar{w} \geq -\bar{q}, \bar{w} \geq 0\}$$

aus, wobei

$$\bar{D} := \begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix}, \quad \bar{w} := \begin{pmatrix} w \\ \omega \end{pmatrix}, \quad \bar{q} := \begin{pmatrix} 0 \\ n+1 \end{pmatrix} \in \mathbb{R}^{n+1}$$

und

$$r := e - De, \quad e = (1, 1, \dots, 1)'$$

gilt.

Für dieses lineare Program (S_0) ist $\bar{w} = 0$ zulässig und optimal. Diese Lösung erfüllt jedoch nicht (18.1), da $\bar{D}0 + 0$ nicht komponentenweise größer als 0 ist. In jeder Optimallösung $\bar{w}^* = \begin{pmatrix} w^* \\ \omega^* \end{pmatrix}$ von (S_0) gilt $\omega^* = 0$, da sonst $\bar{q}'\omega^* > 0$ wäre. Wir werden zeigen, dass es eine Optimallösung \bar{w}^* von (S_0) mit den Eigenschaften

$$\bar{w}^* \geq 0, \quad \bar{D}\bar{w}^* + \bar{q} \geq 0 \quad \text{und} \quad \bar{D}\bar{w}^* + \bar{w}^* + \bar{q} > 0$$

gibt, wobei die letzte Beziehung komponentenweise gilt. Da in jeder Optimallösung von (S_0) $\omega^* = 0$ gilt, ist $r\omega^* = 0$ und daher liefert die Einschränkung von \bar{w}^* auf die ersten n Komponenten eine Lösung w^* von (18.1) und damit eine Lösung des gegebenen Problems.

Wie man leicht zeigen kann, ist das Problem (S_0) ein selbstduales lineares Programm, d.h. das zu (S_0) duale Problem fällt mit (S_0) zusammen. Das dazu duale Problem ist

$$\max \{-\bar{q}'y \mid \bar{D}'y \leq \bar{q}, y \geq 0\} = \min \{\bar{q}'y \mid -\bar{D}y \leq \bar{q}, y \geq 0\} = \min \{\bar{q}'y \mid \bar{D}y \geq -\bar{q}, y \geq 0\},$$

fällt also mit (S_0) zusammen.

Eine wichtige Eigenschaft schiefsymmetrischer Matrizen, die wir im Folgenden oft verwenden werden, ist: Für jede schiefsymmetrische $n \times n$ Matrix D und jeden Vektor $x \in \mathbb{R}^n$ gilt

$$x'Dx = 0.$$

Denn ist $x'Dx = \alpha$, so erhält man $\alpha = (x'Dx)' = x'D'x = -(x'Dx) = -\alpha$, also $\alpha = 0$.

Das lineare Programm (S_0) besitzt die *innere Punkte Eigenschaft (IPE)*: es gibt eine zulässige Lösung $\bar{w}^0 > 0$ mit

$$s(\bar{w}^0) := \bar{D}\bar{w}^0 + \bar{q} > 0.$$

Im Falle von (S_0) können wir etwa $\bar{w}^0 = e := (1, 1, \dots, 1)'$ setzen, denn es gilt

$$\begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} De + r \\ -r'e \end{pmatrix} = \begin{pmatrix} e \\ -n \end{pmatrix}$$

nach Definition des Vektors r und wegen

$$r'e = e'(e - De) = e'e - e'De = n.$$

Bildet man nun

$$s(e) := \begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} + \bar{q} = \begin{pmatrix} e \\ -n \end{pmatrix} + \begin{pmatrix} 0 \\ n+1 \end{pmatrix} = \begin{pmatrix} e \\ 1 \end{pmatrix} > 0,$$

so erkennt man, dass $e = (1, \dots, 1)'$ tatsächlich ein zulässiger innerer Punkt von (S_0) ist.

Nach diesen Vorbemerkungen läßt sich nun die grundsätzliche Idee eines inneren Punkteverfahrens folgenderweise beschreiben:

Um (P) und (D) zu lösen, betrachten wir das selbstduale Problem (S_0) und den inneren Punkt $e > 0$, $s(e) := \bar{D}e + \bar{q} > 0$. Wir setzen

$$\mu_0 := e' s(e) > 0.$$

In Satz 18.2 werden wir zeigen, dass das nichtlineare Gleichungssystem

$$\begin{aligned} s &= \bar{D}\bar{w} + \bar{q} \\ \bar{w}'s &= \mu \end{aligned}$$

für jedes $\mu > 0$ eine eindeutige Lösung $\bar{w}(\mu)$ hat. Die Menge $\{\bar{w}(\mu) \mid 0 < \mu \leq \mu_0\}$ bezeichnen wir als zentralen Pfad. Das zweite wichtige Resultat (Satz 18.5) besagt, dass für $\mu \rightarrow 0$ der Vektor $\bar{w}(\mu)$ gegen ein \bar{w}^* strebt, für das

$$\bar{w}^* + s(\bar{w}^*) > 0$$

gilt, d.h. damit ist eine Lösung von (18.1) gefunden. Drittens werden wir zeigen, dass man bereits nach endlich vielen Schritten weiß, welche Komponenten von \bar{w}^* und $s(\bar{w}^*)$ in der Optimallösung positiv sind. Damit kann der Konvergenzprozeß entlang des zentralen Pfades nach endlich vielen Schritten abgebrochen werden. Die Optimallösung \bar{w}^* kann dann durch das Lösen eines linearen Gleichungssystems in $O(n^3)$ Schritten ermittelt werden. Im weiteren Verlauf dieses Kapitels folgen wir im Wesentlichen der schönen Darstellung von Terlaky [48]. Im nächsten Kapitel werden wir uns dann mit der rechnerischen Umsetzung dieses Konzeptes befassen.

Im Folgenden werden wir des öfteren eine problemangepasste Notation verwenden:

1. Sind $x, s \in \mathbb{R}^n$, so ist $x \cdot s = \begin{pmatrix} x_1 s_1 \\ \vdots \\ x_n s_n \end{pmatrix} \in \mathbb{R}^n$ komponentenweise definiert.
2. Analog ist für $x, s \in \mathbb{R}^n$ der Vektor $\max(x, s)$ komponentenweise definiert.
3. Für $x \in \mathbb{R}_+^n$ ist die $(n \times n)$ -Matrix $X := \text{diag}(x)$ gegeben durch

$$X = \text{diag}(x) = \begin{pmatrix} x_1 & & & \\ & x_2 & 0 & \\ & 0 & \ddots & \\ & & & x_n \end{pmatrix}.$$

Damit entspricht $x^{-1} := \left(\frac{1}{x_1}, \dots, \frac{1}{x_n}\right)$ der Matrix X^{-1} .

18.2 Selbstduale lineare Programme

Es sei D eine beliebige schiefsymmetrische $(n \times n)$ -Matrix und $q \in \mathbb{R}_n^+$. Wir betrachten das lineare Programm

$$(S) \quad \min\{q'x \mid Dx \geq -q, x \geq 0\}.$$

Wie schon gezeigt, ist dieses lineare Programm *selbstdual*, das heißt, das duale Problem zu (S) fällt mit (S) zusammen. Wir bezeichnen die Menge der zulässigen Lösungen von (S) mit M_S

und mit M_S^* die Menge der Optimallösungen von (S) . M_S^* ist nicht leer, denn $x = 0$ ist zulässig und optimal. Setzen wir

$$s(x) := Dx + q, \quad (18.2)$$

so erhalten wir

$$x's(x) = x'(Dx + q) = x'Dx + x'q = q'x \geq 0.$$

Also beschreibt $x's(x)$ ebenfalls den Zielfunktionswert der Lösung x . Da $x = 0$ zulässig ist, gilt insbesondere für jede Optimallösung

$$x's = 0 \quad \text{also} \quad x_j s_j = 0 \quad \text{für} \quad 1 \leq j \leq n.$$

Im Folgenden wollen wir immer annehmen, dass (S) die innere Punkte Eigenschaft (IPE) besitzt. Es gibt also ein $x^0 \in M_S$ mit $x^0 > 0$ und $s(x^0) > 0$.

Lemma 18.1 *Erfüllt (S) die innere Punkte Eigenschaft, so ist die Menge der zulässigen Punkte x mit einem Zielfunktionswert $\leq K$ beschränkt, d.h. die Menge $L_K := \{x \in M_S \mid x's(x) \leq K\}$ ist beschränkt.*

Beweis: Für $x^0 > 0$ gelte $s^0 := s(x^0) > 0$. Dann erhält man

$$\begin{aligned} 0 &= (x - x^0)'D(x - x^0) = (x - x^0)'(s - s^0) \\ &= x's + (x^0)'s^0 - x's^0 - (x^0)'s. \end{aligned}$$

Also ist für jedes $j = 1, 2, \dots, n$

$$x_j s_j^0 \leq x's^0 = x's + (x^0)'s^0 - (x^0)'s \leq K + (x^0)'s^0$$

und somit

$$x_j \leq \frac{K + (x^0)'s^0}{s_j^0}.$$

■

Nun zeigen wir

Satz 18.2 *Die folgenden drei Aussagen sind äquivalent*

1. (S) erfüllt die innere Punkte Bedingung.
2. Für jedes $\mu > 0$ gibt es ein $x(\mu) > 0$ und $s(\mu) := s(x(\mu)) > 0$ mit

$$\begin{aligned} s(\mu) &= Dx(\mu) + q, \\ x(\mu) \cdot s(\mu) &= \mu e. \end{aligned} \quad (18.3)$$

3. Für jedes $w > 0$ gibt es ein $x > 0$ mit $s(x) > 0$ so dass

$$\begin{aligned} s &= Dx + q, \\ x \cdot s &= w. \end{aligned} \quad (18.4)$$

Die Lösungen der Systeme (18.3) und (18.4) sind eindeutig.

Beweis: Die zweite Aussage ist ein Spezialfall der dritten Aussage. Daher gilt $3. \Rightarrow 2.$ und, trivialerweise, $2. \Rightarrow 1.$ Somit ist lediglich zu zeigen, dass aus 1. die Behauptung 3. folgt. Wir zeigen zunächst, dass die Lösung des nichtlinearen Gleichungssystems in (18.4) *eindeutig* ist. Nehmen wir an, $x > 0$ und $\bar{x} > 0$ seien verschiedene Lösungen von (18.4). Dann gilt

$$0 = (x - \bar{x})' D(x - \bar{x}) = (x - \bar{x})'(s - \bar{s}) = \sum_{j=1}^n (x_j - \bar{x}_j)(s_j - \bar{s}_j). \quad (18.5)$$

Da aber $x \cdot s = \bar{x} \cdot \bar{s} = w > 0$ gilt, folgt

$$\begin{aligned} x_j < \bar{x}_j &\Rightarrow s_j > \bar{s}_j, \\ x_j > \bar{x}_j &\Rightarrow s_j < \bar{s}_j. \end{aligned}$$

Daher ist $(x_j - \bar{x}_j)(s_j - \bar{s}_j)$ in beiden Fällen negativ und $x \neq \bar{x}$ liefert

$$\sum_{j=1}^n (x_j - \bar{x}_j)(s_j - \bar{s}_j) < 0$$

im Widerspruch zu (18.5).

Nun zeigen wir, dass aus der ersten Aussage die dritte Aussage folgt. Dazu haben wir zu zeigen, dass für ein festes $\hat{w} > 0$ das Gleichungssystem

$$x \cdot (Dx + q) = \hat{w}$$

eine Lösung $x > 0$ besitzt. Nach Voraussetzung gibt es ein $x^0 > 0$ mit $s^0 = Dx^0 + q > 0$. Es sei $w^0 := x^0 \cdot s^0 > 0$.

Wir definieren zwei Vektoren \underline{w} und \bar{w} mit $0 < \underline{w} < \bar{w}$ durch

$$\begin{aligned} \underline{w} &:= \frac{1}{2} \min(w^0, \hat{w}), \\ \bar{w} &:= e + \max(w^0, \hat{w}). \end{aligned}$$

Die Menge C

$$C := \{x \mid x \geq 0, \underline{w} \leq x \cdot s(x) \leq \bar{w}\}$$

enthält x^0 und ist kompakt, da sie abgeschlossen und in der nach Lemma 18.1 beschränkten Menge $\{x \mid x's \leq \bar{w} \cdot e\}$ enthalten ist. Die Funktion

$$d(x) := \|x \cdot (Dx + q) - \hat{w}\|_\infty$$

ist stetig und nimmt daher auf der kompakten Menge C ihr Minimum an, etwa für den Punkt $\tilde{x} > 0$.

Ist $d(\tilde{x}) = 0$, dann ist $\tilde{x} \cdot s(\tilde{x}) = \hat{w}$ und das Gleichungssystem (18.4) wird durch \tilde{x} gelöst. Wir werden nun zeigen, dass $d(\tilde{x}) > 0$ auf einen Widerspruch führt. Nehmen wir dazu an, dass $d(\tilde{x}) > 0$ gilt. Wir setzen $\tilde{s} := s(\tilde{x})$ und $\tilde{w} := \tilde{x} \cdot \tilde{s}$. Nun bestimmen wir den Vektor Δx aus dem linearen Gleichungssystem

$$\tilde{x} \cdot (D\Delta x) + \tilde{s} \cdot \Delta x = \hat{w} - \tilde{w}. \quad (18.6)$$

Multipliziert man dieses Gleichungssystem mit \tilde{X}^{-1} von links, so erhält man

$$(D + \text{diag}(\tilde{x}^{-1} \cdot \tilde{s}))\Delta x = \tilde{X}^{-1}(\hat{w} - \tilde{w}).$$

Da D positiv semidefinit ist und

$$\tilde{x}_i^{-1} \tilde{s}_i = \frac{\tilde{s}_i}{\tilde{x}_i} > 0$$

gilt, ist die Koeffizientenmatrix des Gleichungssystems (18.6) positiv definit, also regulär. Daher besitzt (18.6) eine eindeutige Lösung Δx . Für $\alpha > 0$ definieren wir

$$\begin{aligned} x(\alpha) &:= \tilde{x} + \alpha \Delta x, \\ s(\alpha) &:= s(\tilde{x}) + \alpha D \Delta x. \end{aligned}$$

Mit $\Delta s := D \Delta x$ wird $\tilde{s} + \alpha \Delta s$ zu $s(\tilde{x} + \alpha \Delta x)$. Für hinreichend kleines α ist $x(\alpha) > 0$ und $s(\alpha) > 0$ und damit $w(\alpha) := x(\alpha) \cdot s(\alpha) > 0$. Wir erhalten aus (18.6)

$$\begin{aligned} w(\alpha) - \hat{w} &= (\tilde{x} + \alpha \Delta x) \cdot (\tilde{s} + \alpha \Delta s) - \hat{w} = \tilde{w} - \hat{w} + \alpha(\tilde{x} \cdot \Delta s + \tilde{s} \cdot \Delta x) + \alpha^2 \Delta x \cdot \Delta s \\ &= (\tilde{w} - \hat{w})(1 - \alpha) + \alpha^2 \Delta x \cdot \Delta s. \end{aligned}$$

Ist α so klein gewählt, dass für jedes j mit $\Delta x_j \Delta s_j$ ($\tilde{w}_j - \hat{w}_j$) > 0 die Beziehung

$$0 < \alpha < \frac{\tilde{w}_j - \hat{w}_j}{\Delta x_j \Delta s_j}$$

gilt, dann ist für jede Komponente

$$|\hat{w}_j - w_j(\alpha)| < |\hat{w}_j - \tilde{w}_j|.$$

Daher liegt der Vektor $w(\alpha)$ näher bei \hat{w} als \tilde{w} . Da \hat{w} ein innerer Punkt der Menge $[\underline{w}, \bar{w}]$ war, liegt auch $w(\alpha)$ in dieser Menge und daher ist $x(\alpha) \in C$. Andererseits ist $d(x(\alpha)) = \|w(\alpha) - \hat{w}\|_\infty < d(\tilde{x})$. Dies ist aber ein Widerspruch dazu, dass $d(\tilde{x})$ der kleinste Wert von $d(x)$ auf C war. Damit gilt also $d(\tilde{x}) = 0$ und das Gleichungssystem (18.4) ist lösbar.

Damit ist Satz 18.2 vollständig bewiesen. ■

Ist x^* eine Optimallösung von (S) , dann gilt $x^* \cdot s(x^*) = 0$. Setzt man also

$$\begin{aligned} B &:= \{j \mid x_j^* > 0\}, \\ N &:= \{j \mid s_j(x^*) > 0\}, \end{aligned}$$

so erhält man zwei disjunkte Mengen B und N . Genau dann ist

$$x^* + s(x^*) > 0,$$

wenn B und N eine Partition der Indexmenge $\{1, 2, \dots, n\}$ bilden. Diese Partition (B, N) heißt *optimale Partition*.

Lemma 18.3 *Die optimale Partition ist eindeutig bestimmt.*

Beweis: Sei x eine Optimallösung mit $x + s(x) > 0$, die auf die Partition (B_x, N_x) führt und sei y eine Optimallösung mit $y + s(y) > 0$, die auf die Partition (B_y, N_y) führt.

Aus $(x - y)'D(x - y) = 0$ folgt

$$x' s(x) + y' s(y) = x' s(y) + y' s(x).$$

Infolge der Optimalität ist $x' s(x) = 0$ und $y' s(y) = 0$ also gilt auch, da alle beteiligten Vektoren nichtnegativ sind,

$$x' s(y) = y' s(x) = 0.$$

Aus $x' s(y) = 0$ folgt $B_x \cap N_y = \emptyset$, also $B_x \subseteq B_y$. Analog folgt aus $y' s(x) = 0$ die Beziehung $B_y \subseteq B_x$. Daher ist

$$B_x = B_y \quad \text{und} \quad N_x = N_y.$$

■

Definition 18.1 Sei (B, N) die optimale Partition. Eine Optimallösung x^* mit maximalem Wert

$$\prod_{i \in B} x_i^* \prod_{i \in N} s_i(x^*)$$

heißt das analytische Zentrum von M_S^* .

Lemma 18.4 Das analytische Zentrum ist eindeutig bestimmt.

Beweis: Seien x^* und \bar{x} zwei verschiedene analytische Zentren, $\bar{s} := s(\bar{x})$ and $s^* := s(x^*)$. Wir setzen $\hat{x} := \frac{1}{2}(x^* + \bar{x})$. Die Lösung \hat{x} ist zulässig und es gilt

$$\prod_{j \in B} \hat{x}_j \prod_{j \in N} s_j(\hat{x}) = \prod_{j \in B} \frac{1}{2}(x_j^* + \bar{x}_j) \prod_{j \in N} \frac{1}{2}(s_j^* + \bar{s}_j).$$

Dividiert man die rechte Seite durch $\sqrt{\prod_{j \in B} x_j^* \bar{x}_j \prod_{j \in N} s_j^* \bar{s}_j}$, so erhält man

$$\prod_{j \in B} \hat{x}_j \prod_{j \in N} s_j(\hat{x}) = \prod_{j \in B} \frac{1}{2} \left(\sqrt{\frac{x_j^*}{\bar{x}_j}} + \sqrt{\frac{\bar{x}_j}{x_j^*}} \right) \prod_{j \in N} \frac{1}{2} \left(\sqrt{\frac{s_j^*}{\bar{s}_j}} + \sqrt{\frac{\bar{s}_j}{s_j^*}} \right) \cdot \sqrt{\prod_{j \in B} x_j^* \bar{x}_j \prod_{j \in N} s_j^* \bar{s}_j}.$$

Nun ist für jedes $\alpha \neq 1$, $\alpha > 0$: $\alpha + \frac{1}{\alpha} > 2$. Daher folgt aus

$$\sqrt{\prod_{j \in B} x_j^* \bar{x}_j \prod_{j \in N} s_j^* \bar{s}_j} = \prod_{j \in B} x_j^* \prod_{j \in N} s_j^* :$$

$$\prod_{j \in B} \hat{x}_j \prod_{j \in N} \hat{s}_j > \prod_{j \in B} x_j^* \prod_{j \in N} s_j^*.$$

Dies ist ein Widerspruch dazu, dass x^* analytisches Zentrum ist. ■

Satz 18.5 Erfüllt (S) die innere Punkte Bedingung, so konvergiert $x(\mu)$ für $\mu \rightarrow 0$ gegen das analytische Zentrum x^* von M_S^* und es gilt $x^* + s(x^*) > 0$.

Beweis: Es sei $\{\mu_t\}$ eine monotone Nullfolge. Dann ist $x(\mu_t) \in \{x \mid x' s(x) \leq n\mu_t\} =: L_{n\mu_t}$. Nach Lemma 18.1 ist die Menge $L_{n\mu_t}$ kompakt. Folglich besitzt die Folge $\{x(\mu_t)\}$ einen Häufungspunkt x^* . Wir setzen $s^* := s(x^*)$. Eine Teilfolge von $\{x(\mu_t)\}$ konvergiert gegen x^* , wir bezeichnen sie wieder mit $\{x(\mu_t)\}$.

x^* ist optimal, denn

$$x^* \cdot s^* = \lim_{t \rightarrow \infty} x(\mu_t) \cdot s(\mu_t) = \lim_{t \rightarrow \infty} \mu_t e = 0.$$

Wir setzen

$$\begin{aligned} B &:= \{i \mid x_i^* > 0\}, \\ N &:= \{i \mid s_i^* > 0\}. \end{aligned}$$

Da D schiefssymmetrisch ist, folgt

$$0 = (x^* - x(\mu_t))'(s^* - s(\mu_t)) = x(\mu_t)'s(\mu_t) - x^*'s(\mu_t) - s^*'x(\mu_t).$$

Aus

$$x_j(\mu_t) \cdot s_j(\mu_t) = \mu_t \text{ für jedes } j = 1, 2, \dots, n \quad (18.7)$$

folgt

$$\sum_{j \in B} x_j^* s_j(\mu_t) + \sum_{j \in N} s_j^* x_j(\mu_t) = n\mu_t.$$

Dividiert man diese Gleichung unter Verwendung von (18.7) durch μ_t , so erhält man

$$\sum_{j \in B} \frac{x_j^*}{x_j(\mu_t)} + \sum_{j \in N} \frac{s_j^*}{s_j(\mu_t)} = n.$$

Der Grenzübergang $\lim \mu_t \rightarrow 0$ liefert $|B| + |N| = n$, also die optimale Partition. Daher gilt $x^* + s(x^*) > 0$.

Wir zeigen nun, dass x^* das analytische Zentrum von M_S^* ist. Sei \bar{x} das analytische Zentrum und x^* der Grenzwert der Folge $x(\mu_t)$. Dann gilt

$$(\bar{x} - x(\mu))'(\bar{s} - s(\mu)) = 0 - x(\mu)'\bar{s} - \bar{x}'s(\mu) + x(\mu)'s(\mu).$$

Dies ergibt

$$\bar{x}'s(\mu) + \bar{s}'x(\mu) = n\mu.$$

Dividiert man diese Gleichung unter Verwendung von (18.7) durch μ , so erhält man

$$\sum_{j \in B} \frac{\bar{x}_j}{x_j(\mu)} + \sum_{j \in N} \frac{\bar{s}_j}{s_j(\mu)} = n.$$

Der Grenzübergang $\mu \rightarrow 0$ liefert

$$\sum_{j \in B} \frac{\bar{x}_j}{x_j^*} + \sum_{j \in N} \frac{\bar{s}_j}{s_j^*} = n.$$

Wendet man auf diesen Ausdruck die Ungleichung zwischen arithmetischem und geometrischem Mittel an, so erhält man:

$$\left(\prod_{j \in B} \frac{\bar{x}_j}{x_j^*} \prod_{j \in N} \frac{\bar{s}_j}{s_j^*} \right)^{\frac{1}{n}} \leq \frac{1}{n} \left(\sum_{j \in B} \frac{\bar{x}_j}{x_j^*} + \sum_{j \in N} \frac{\bar{s}_j}{s_j^*} \right) = 1.$$

Daraus folgt

$$\prod_{j \in B} \bar{x}_j \prod_{j \in N} \bar{s}_j \leq \prod_{j \in B} x_j^* \prod_{j \in N} s_j^*,$$

also $\bar{x} = x^*$. Da das analytische Zentrum eindeutig bestimmt ist, gibt es somit nur einen Häufungspunkt und $x(\mu_t)$ konvergiert gegen x^* . ■

18.3 Endlichkeit des Inneren Punkteverfahrens

Als nächstes zeigen wir, dass gar kein Grenzprozess zur Bestimmung der Indexmengen B und N - und damit zur Bestimmung einer Optimallösung von (S) - durchgeführt werden muß, sondern dass diese Indexmengen B und N schon nach endlich vielen Schritten festliegen.

Zunächst definieren wir die *Konditionszahl* σ eines selbstdualen linearen Programms durch

$$\sigma := \max_{x \in M_S^*} \min_{1 \leq j \leq n} (x_j + s_j(x)). \quad (18.8)$$

Ferner definieren wir die *Norm der Spalten* D_1, D_2, \dots, D_n der Matrix D durch

$$\|D_j\|_2 := \sqrt{d_{1j}^2 + d_{2j}^2 + \dots + d_{nj}^2}. \quad (18.9)$$

und setzen

$$\pi := \prod_{j=1}^n \|D_j\|_2. \quad (18.10)$$

Nun gilt

Lemma 18.6 *Hat die schiefsymmetrische, ganzzahlige, $n \times n$ Matrix D keine Nullspalte und ist q ganzzahlig, so gilt*

$$\sigma \geq \frac{1}{\pi}. \quad (18.11)$$

Beweis: Sei $x \in M_S^*$. Die Vektoren x und s sind Lösungen des linearen Gleichungssystems

$$-Dx + s = q.$$

Fasst man x und s zu einem Vektor y zusammen, so gilt mit der Einheitsmatrix E : $(-D, E)y = q$. D_y sei jene Untermatrix von $(-D, E)$ deren Spalten zu den positiven Komponenten von x und s gehören. O.B.d.A. können wir annehmen, dass D_y eine reguläre Matrix ist, denn anderenfalls reduziert man ihre Zeilen und Spalten solange, bis man eine reguläre Matrix erhält. Bezeichnen wir mit D_y^j jene Matrix, die man aus D_y erhält, wenn man deren j -te Spalte durch den Vektor q der rechten Seite des Gleichungssystems ersetzt, so ist die j -te Komponente y_j von y infolge $x \geq 0, s \geq 0$ und der Cramer'schen Regel

$$y_j = \frac{|\det D_y^j|}{|\det D_y|}.$$

Da $|\det D_y| \geq 1$ ist, erhalten wir aus der Hadamard'schen Ungleichung (Lemma 17.3)

$$y_j \geq \frac{1}{|\det D_y|} \geq \frac{1}{\prod_{j=1}^n \|D_j\|}.$$

■

Die Bedingung, dass die Koeffizientenmatrix des selbstdualen Programms keine Nullspalte besitzt, ist für (S_0) erfüllt, denn $r = e - De \neq 0$, da $e'r = e'e - e'De = n \neq 0$ ist. Wäre andererseits eine Spalte D_j eine Nullspalte, dann ist $r_j = 1 - e'D_j = 1$ und somit hat die Matrix

$$\bar{D} := \begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix}$$

keine Nullspalte.

Satz 18.7 Die optimale Partition (B, N) kann aus jeder Lösung $(x(\mu), s(x(\mu)))$ mit $\mu < \frac{\sigma^2}{n^2}$ ermittelt werden.

Da nach Lemma 18.6 $\sigma \geq \frac{1}{\pi}$ gilt, folgt aus Satz 18.7 unmittelbar

Korollar 18.8 Die optimale Partition kann insbesondere aus jeder Lösung $(x(\mu), s(x(\mu)))$ mit

$$\mu < \frac{1}{n^2 \pi^2}$$

ermittelt werden.

Beweis: Ist (x^*, s^*) eine Optimallösung, dann erhält man aus

$$0 = (x(\mu) - x^*)' D (x(\mu) - x^*) = (x(\mu) - x^*)' (s(\mu) - s^*)$$

die Beziehung $x(\mu)' s^* + s(\mu)' x^* = n\mu$. Also gilt

$$x_j(\mu) s_j^* \leq x(\mu)' s^* \leq x(\mu)' s^* + s(\mu)' x^* = n\mu.$$

Aufgrund der Definition (18.8) von σ kann man (x^*, s^*) so wählen, dass für $j \in N$ gilt $s_j^* \geq \sigma$. Daraus ergeben sich

$$x_j(\mu) \leq \frac{n\mu}{s_j^*} \leq \frac{n\mu}{\sigma}$$

und

$$s_j(\mu) = \frac{\mu}{x_j(\mu)} \geq \frac{\mu\sigma}{n\mu} = \frac{\sigma}{n}.$$

Daher gilt für alle $j \in N$:

$$x_j(\mu) \leq \frac{n\mu}{\sigma} \quad \text{und} \quad s_j(\mu) \geq \frac{\sigma}{n}.$$

In analoger Weise erhält man für $j \in B$:

$$x_j(\mu) \geq \frac{\sigma}{n} \quad \text{und} \quad s_j(\mu) \leq \frac{n\mu}{\sigma}.$$

Ist nun $\mu < \frac{\sigma^2}{n^2}$, so gilt für

$$j \in N : \quad x_j(\mu) < \frac{\sigma}{n} \quad \text{und} \quad s_j(\mu) \geq \frac{\sigma}{n}$$

$$j \in B: \quad x_j(\mu) \geq \frac{\sigma}{n} \quad \text{und} \quad s_j(\mu) < \frac{\sigma}{n}.$$

Daher sind die Indexmengen B und N eindeutig festgelegt. \blacksquare

Sind aber einmal die Indexmengen B und N eindeutig festgelegt, so lassen sich x^* und s^* mit $x^* + s^* > 0$ durch Runden aus einer Näherungslösung $(x(\mu), s(\mu))$ bestimmen. Wir fassen in der Matrix D und in den Vektoren x und s die Zeilen und Spalten mit Indizes in B und N zusammen und erhalten

$$D = \begin{pmatrix} D_{BB} & D_{BN} \\ D_{NB} & D_{NN} \end{pmatrix} \quad \text{und} \quad x = \begin{pmatrix} x_B \\ x_N \end{pmatrix}, \quad s = \begin{pmatrix} s_B \\ s_N \end{pmatrix}.$$

Damit wird $Dx + q = s$ zu

$$D_{BB}x_B + D_{BN}x_N + q_B = s_B \quad (18.12)$$

$$D_{NB}x_B + D_{NN}x_N + q_N = s_N. \quad (18.13)$$

Nach Satz 18.5 existiert eine optimale Lösung mit $x_B^* > 0$, $x_N^* = 0$, $s_B^* = 0$ und $s_N^* > 0$. Setzen wir $x_N^* = 0$, $s_B^* = 0$, $x_B^* = x_B(\mu) + \xi$ und $s_N^* = s_N(\mu) + \zeta$, dann erhält man aus $Dx^* + q = s^*$:

$$D_{BB}(x_B(\mu) + \xi) + q_B = 0 \quad (18.14)$$

$$D_{NB}(x_B(\mu) + \xi) + q_N = s_N(\mu) + \zeta. \quad (18.15)$$

Unter Verwendung von (18.12) folgt dann

$$D_{BB}\xi = D_{BN}x_N(\mu) - s_B(\mu) =: -q_B^* \quad (18.16)$$

und

$$\zeta = D_{NB}\xi - D_{NN}x_N(\mu).$$

Mit $q_N^* := -D_{NN}x_N(\mu)$ wird die letzte Gleichung zu

$$\zeta = D_{NB}\xi + q_N^* \quad (18.17)$$

Somit kann ξ durch Lösen eines linearen Gleichungssystems in $O(n^3)$ Schritten gefunden werden und ζ wird durch eine einfache Matrixmultiplikation ermittelt. Zu zeigen bleibt, dass die so gefundene Lösung tatsächlich $x^* + s^* > 0$ erfüllt. Dazu verwenden wir die folgenden Abschätzungen. Setzen wir

$$\beta := \|D\|_\infty := \max_{1 \leq i \leq n} \sum_{j=1}^n |d_{ij}|. \quad (18.18)$$

Dann ist

$$\|q_N^*\|_\infty = \|D_{NN}x_N(\mu)\|_\infty \leq \|D_{NN}\|_\infty \cdot \|x_N(\mu)\|_\infty \leq \beta \frac{n\mu}{\sigma}. \quad (18.19)$$

Ferner ist

$$\begin{aligned} \|D_{BN}x_N(\mu) - s_B(\mu)\| &\leq \sqrt{n} \|D_{BN}x_N(\mu) - s_B(\mu)\|_\infty \\ &\leq \sqrt{n} \|(E, D_{BN})\|_\infty \cdot \max(\|x_N(\mu)\|_\infty, \|s_B(\mu)\|_\infty) \\ &\leq \sqrt{n}(1 + \beta) \frac{n\mu}{\sigma} = \frac{1}{\sigma} \cdot n^{3/2}(1 + \beta)\mu. \end{aligned} \quad (18.20)$$

Nun zeigen wir

Satz 18.9 *Hat die ganzzahlige, $n \times n$ Matrix D keine Nullspalte und ist q ganzzahlig, dann erfüllt die Lösung (x^*, s^*) , die man durch Runden aus $(x(\mu), s(\mu))$ mit*

$$\mu < \frac{\sigma^2}{n^{5/2}(1+\beta)^2\pi} \quad (18.21)$$

erhält, die Beziehung

$$x_B^* > 0 \quad \text{und} \quad s_N^* > 0.$$

Bemerkung. Die Bedingung für μ in Satz 18.9 ist sicher erfüllt, wenn

$$\mu < \frac{1}{n^{5/2}(1+\beta)^2\pi^3}$$

gewählt wird.

Beweis: Betrachten wir zunächst das Gleichungssystem $D_{BB}\xi = D_{BN}x_N(\mu) - s_B = -\bar{q}$. Ist $D_{BB} = 0$, dann kann $\xi := 0$ gesetzt werden. In diesem Fall gilt $x_B^* = x_B(\mu) > 0$. Ferner gilt $\zeta = -D_{NN}x_N(\mu)$. Aus (18.19) folgt dann $\|\zeta\|_\infty \leq \beta \frac{n\mu}{\sigma}$. Daher erhalten wir für s_j^* mit $j \in N$:

$$|s_j^*| = |s_j(\mu) + \zeta_j| \geq |s_j(\mu)| - \|\zeta\|_\infty \geq \frac{\sigma}{n} - \beta \frac{n\mu}{\sigma} > 0$$

auf Grund der Wahl von μ in (18.21).

Ist $D_{BB} \neq 0$, dann reduzieren wir das Gleichungssystem $D_{BB}\xi = -q_B^*$, bis wir eine reguläre Koeffizientenmatrix \bar{D}_{BB} erhalten. Der Vektor $-\bar{q}_B^*$ enthält dann die entsprechenden Komponenten von $-q_B^*$. Durch die Cramer'sche Regel kann man dann ξ wie folgt abschätzen. Dabei ist \bar{D}_{BB}^j jene Matrix, die man aus \bar{D}_{BB} erhält, indem man die j -te Spalte durch die rechte Seite $-\bar{q}_B^*$ ersetzt. Ferner verwenden wir die Tatsache, dass der Absolutbetrag der Determinante einer ganzzahligen Matrix ≥ 1 ist und $-q_B^*$ durch (18.20) abgeschätzt werden kann.

$$\begin{aligned} |\xi_j| &= \frac{|\det \bar{D}_{BB}^j|}{|\det \bar{D}_{BB}|} \leq |\det \bar{D}_{BB}^j| \\ &\leq \|-\bar{q}_B^*\|_\infty \cdot |\det D_{BB}| \\ &\leq \frac{1}{\sigma} n^{3/2}(1+\beta)\mu\pi. \end{aligned} \quad (18.22)$$

Da $x_j(\mu) \geq \frac{\sigma}{n}$ ist, ergibt sich dann aus der Wahl (18.21) von μ , dass $x_j^* = x_j(\mu) + \xi_j > 0$ gilt.

Verwendet man die Abschätzung (18.22) von ξ in der Gleichung (18.17), so erhält man aus (18.19)

$$\begin{aligned} |\zeta_j| &\leq \|q_N^* + D_{NB}\xi\|_\infty \leq (1+\beta) \max(\|q_N^*\|_\infty, \|\xi\|_\infty) \\ &\leq (1+\beta) \max\left(\beta \frac{n\mu}{\sigma}, \frac{1}{\sigma} n^{3/2}(1+\beta)\mu\pi\right) = \frac{1}{\sigma} n^{3/2}(1+\beta)^2\mu\pi. \end{aligned}$$

Da $s_j(\mu) \geq \frac{\sigma}{\mu}$ für $j \in N$ gilt, ist auf Grund der Wahl (18.21) von μ für $j \in N$ auch $s_j^* = s_j(\mu) + \zeta_j > 0$. ■

Kapitel 19

Innere Punkte Verfahren: Algorithmen

19.1 Ein generischer Algorithmus

In diesem Kapitel wollen wir die im letzten Kapitel gewonnenen Ergebnisse algorithmisch umsetzen. Wir gehen von den beiden zueinander dualen linearen Programmen

$$(P) \quad \max \{c'x \mid Ax \leq b, x \geq 0\}$$

und

$$(D) \quad \min \{b'y \mid A'y \geq c, y \leq 0\}$$

aus. Wie wir in Abschnitt 18.1 gesehen haben, betrachtet man das selbstduale lineare Programm

$$(S_0) \quad \min\{(n+1)\omega \mid -\bar{D}\bar{w} + \bar{s} = \bar{q}, \bar{w} \geq 0, \bar{s} \geq 0\}$$

wobei

$$\bar{D} := \begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix}, \bar{w} := \begin{pmatrix} w \\ \omega \end{pmatrix}, \bar{s} := \begin{pmatrix} s \\ u \end{pmatrix}, \bar{q} := \begin{pmatrix} 0 \\ n+1 \end{pmatrix} \in \mathbb{R}^{n+1}$$

und

$$D := \begin{pmatrix} 0 & A' & -c \\ -A & 0 & b \\ c' & -b' & 0 \end{pmatrix} \quad \text{und} \quad r := e - De, \quad e = (1, 1, \dots, 1)'$$

gilt.

Für dieses lineare Programm (S_0) ist $\bar{w}_0 = \bar{s}_0 = e$ ein zulässiger innerer Punkt. Ausgehend von diesem inneren Punkt löst man in den folgenden Schritten für immer kleiner werdendes μ ein nichtlineares Gleichungssystem der Form

$$-\begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix} \begin{pmatrix} w \\ \omega \end{pmatrix} + \begin{pmatrix} s \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ n+1 \end{pmatrix} \tag{19.1}$$

$$\begin{pmatrix} w \\ \omega \end{pmatrix} \cdot \begin{pmatrix} s \\ u \end{pmatrix} = \mu e \tag{19.2}$$

durch das Newtonverfahren. Bezeichnen wir die Lösung mit

$$\begin{pmatrix} w + \Delta w \\ \omega + \Delta\omega \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} s + \Delta s \\ u + \Delta u \end{pmatrix}.$$

Dann erhält man für ein gegebenes μ :

$$-\begin{pmatrix} D & r \\ -r' & 0 \end{pmatrix} \begin{pmatrix} w + \Delta w \\ \omega + \Delta\omega \end{pmatrix} + \begin{pmatrix} s + \Delta s \\ u + \Delta u \end{pmatrix} = \begin{pmatrix} 0 \\ n + 1 \end{pmatrix} \quad (19.3)$$

$$\begin{pmatrix} w + \Delta w \\ \omega + \Delta\omega \end{pmatrix} \cdot \begin{pmatrix} s + \Delta s \\ u + \Delta u \end{pmatrix} = \mu e \quad (19.4)$$

Vernachlässigt man die Glieder zweiter Ordnung, dann erhalten wir aus (19.3)

$$-D(w + \Delta w) - r(\omega + \Delta\omega) + (s + \Delta s) = 0$$

und somit unter Verwendung von (19.1)

$$-D\Delta w - r\Delta\omega + \Delta s = 0. \quad (19.5)$$

Ferner folgt aus (19.4)

$$s \cdot \Delta w + w \cdot \Delta s = \mu e - w \cdot s \quad (19.6)$$

$$u \cdot \Delta\omega + \omega \cdot \Delta u = \mu - u\omega \quad (19.7)$$

Multipliziert man (19.1) mit (w', ω) von links, so erhält man

$$w's + \omega u = \omega(n + 1)$$

Analog erhält man für die Lösung nach dem Newtonschritt

$$(w + \Delta w)'(s + \Delta s) + (\omega + \Delta\omega)(u + \Delta u) = (\omega + \Delta\omega)(n + 1).$$

Durch Ausmultiplikation, Nullsetzen der Terme 2. Ordnung und unter Verwendung von (19.6) und (19.7) erhält man daraus

$$(w's + \omega u) + (n\mu - w's) + (\mu - u\omega) = (n + 1)(\omega + \Delta\omega),$$

also

$$\Delta\omega = \mu - \omega. \quad (19.8)$$

Setzt man dies in (19.7) ein, so bekommt man

$$u\mu - u\omega + \omega\Delta u = \mu - u\omega,$$

also

$$\Delta u = \frac{\mu(1 - u)}{\omega} \quad (19.9)$$

Aus (19.6) erhält man mit $W^{-1} := (\text{diag}(w))^{-1}$

$$\Delta s = \mu W^{-1} e - s - W^{-1} \text{diag}(s) \Delta w. \quad (19.10)$$

Setzen wir Δs und Δw in (19.5) ein, so ergibt sich

$$-D\Delta w - r(\mu - \omega) + \mu W^{-1}e - s - W^{-1}\text{diag}(s)\Delta w = 0.$$

Daher ist

$$(D + W^{-1}\text{diag}(s))\Delta w = \mu W^{-1}e - s - (\mu - \omega)r. \quad (19.11)$$

Dies ist ein lineares Gleichungssystem (mit einer positiv definiten Koeffizientenmatrix) für die Unbekannten Δw , das aufgrund der Form von D außerdem noch eine spezielle Gestalt hat. In der Literatur (siehe etwa Roos, Terlaky und Vial [40]) wurden mehrere spezielle Lösungsverfahren für dieses Gleichungssystem angegeben. Mit (19.10) und (19.11) haben wir die Update-Formeln für w und s gewonnen.

Anstelle von Newtonschritten

$$\begin{aligned} w_{\text{neu}} &:= w + \Delta w \\ s_{\text{neu}} &:= s + \Delta s \end{aligned}$$

verwendet man vielfach *gedämpfte* Newtonschritte mit einem Dämpfungsparameter α , der von Schritt zu Schritt variieren kann:

$$\begin{aligned} w_{\text{neu}} &:= w + \alpha\Delta w \\ s_{\text{neu}} &:= s + \alpha\Delta s. \end{aligned}$$

Da wir das nichtlineare Gleichungssystem (19.2) nur approximativ gelöst haben, liegt der neue Punkt nicht mehr am zentralen Pfad. Aufgrund von Stetigkeitsüberlegungen kann man sich leicht davon überzeugen, dass die Ergebnisse des vorigen Kapitels auch in der Nähe des zentralen Pfades gültig bleiben. In unserem numerischen Verfahren müssen wir sicherstellen, dass wir „in der Nähe“ des zentralen Pfades bleiben. Um den Abstand vom zentralen Pfad zu messen, stehen verschiedene Maße zur Verfügung, die alle davon Gebrauch machen, dass am zentralen Pfad alle Komponenten des Vektors $\bar{w} \cdot \bar{s}$ gleich sind. Man kann also etwa die Größe

$$\delta_c(\bar{w}, \bar{s}) := \frac{\max \bar{w} \cdot \bar{s}}{\min \bar{w} \cdot \bar{s}} \quad (19.12)$$

verwenden. Ein anderes Maß wäre

$$\delta_0(\bar{w}, \bar{s}, \mu) := \frac{1}{2} \left\| \sqrt{\frac{\mu e}{\bar{w} \cdot \bar{s}}} - \sqrt{\frac{\bar{w} \cdot \bar{s}}{\mu e}} \right\| \quad (19.13)$$

Beide Maße führen auf polynomielle Algorithmen (siehe etwa [40]).

Zusammenfassend läßt sich ein Inneres Punkteverfahren somit folgenderweise beschreiben:

Algorithmus 4 *Generisches Innere Punkte Verfahren*

- Wähle einen Genauigkeitsparameter ϵ
- Wähle einen Update-Faktor θ mit $0 < \theta < 1$ für μ
- Wähle einen Nachbarschaftsparameter κ .

- Wähle eine Ausgangslösung \bar{w}_0, \bar{s}_0 und ein $\mu_0 \leq 1$ so dass $\delta(\bar{w}_0, \bar{s}_0, \mu_0) \leq \kappa$.

begin

$\bar{w} := \bar{w}_0;$

$\bar{s} := \bar{s}_0;$

$\mu := \mu_0;$

while $(n+1)\mu \geq \epsilon$ **do**

$\mu := (1-\theta)\mu;$

while $\delta(\bar{w}, \bar{s}, \mu) \geq \kappa$ **do**

$\bar{w} := \bar{w} + \alpha\Delta\bar{w};$

$\bar{s} := \bar{w} + \alpha\Delta\bar{s};$

endwhile

endwhile

end

Eine besonders einfache Version eines inneren Punkte Verfahrens, das zudem noch eine hervorragende Komplexität aufweist, erhält man durch folgende Wahl: Man beginnt mit $\mu_0 := 0$ und wählt als Update Parameter $\theta := \frac{1}{2\sqrt{(n+1)}}$. Man kann dann zeigen, dass bei vollen Newton Schritten die neuen Lösungen in der Nähe des zentralen Pfades bleiben (siehe [40]). Damit erübrigt sich die Abfrage $\delta(\bar{w}, \bar{s}, \mu) \geq \kappa$. Zusammengefaßt kann man den Algorithmus folgenderweise formulieren:

Algorithmus 5 *Primal-duales innere Punkteverfahren mit logarithmischer Barrierfunktion und vollen Newton Schritten*

- Wähle einen Genauigkeitsparameter ϵ

- $\bar{w} := e;$

- $\bar{s} := e;$

- $\mu := 1;$

- $\theta := \frac{1}{2\sqrt{(n+1)}};$

while $(n+1)\mu > \epsilon$ **do**

$\bar{w} := \bar{w} + \Delta\bar{w};$

$\bar{s} := \bar{w} + \Delta\bar{s};$

$\mu := (1-\theta)\mu;$

endwhile

Man kann nun zeigen

Satz 19.1 *Das primal-duale innere Punkteverfahren mit logarithmischer Barrierfunktion und vollen Newton Schritten liefert nach höchstens $\lceil 2\sqrt{n+1} \log \frac{n+1}{\epsilon} \rceil$ Schritten eine zulässige Lösung von (S_0) mit $(n+1)\mu \leq \epsilon$.*

Für einen Beweis dieses Resultates sei wieder auf [40], Theorem II.52 verwiesen.

Kapitel 20

Ganzzahligkeit der Lösungen linearer Programme

Oftmals ist man bei Problemen in der Praxis an Lösungen von linearen Programmen interessiert, die ganzzahlig sind, z.B. wenn es sich um Stückzahlen handelt oder um das Ergebnis einer logischen Entscheidung *ja* oder *nein*. Daher ist das Problem

$$(IP) \quad \max\{c'x \mid Ax \leq b, x \geq 0 \text{ ganzzahlig}\}$$

eine Aufgabe, die beträchtliches Interesse in Praxis und Forschung gefunden hat.

Eine der ersten Fragen, die sich im Zusammenhang mit der Ganzzahligkeit von Lösungen bei linearen Programmen stellt, ist: *Wann sind alle Basislösungen eines linearen Programmes ganzzahlig?* Oder geometrisch ausgedrückt: *Wann hat die Menge der zulässigen Punkte eines linearen Programmes nur ganzzahlige Ecken?* In diesem Fall liefert das Simplexverfahren stets eine ganzzahlige Lösung.

Wir werden dieser Frage in diesem Abschnitt nachgehen und zeigen, dass es eine wichtige Klasse von linearen Optimierungsaufgaben gibt, von denen man zeigen kann, dass sie bei ganzzahligen Ausgangswerten auch ganzzahlige Optimallösungen besitzen. Andererseits besitzt leider die große Masse linearer Programme *nicht* diese schöne Eigenschaft. Daher werden wir uns in späteren Abschnitten überlegen müssen, wie man die Ganzzahligkeit von Lösungen *erzwingen* kann.

Gegeben sei eine ganzzahlige $(n \times n)$ -Matrix A . Die Matrix A heißt *unimodular*, wenn $\det A = \pm 1$. So ist beispielsweise die Matrix

$$A = \begin{pmatrix} 4 & 3 \\ 7 & 5 \end{pmatrix}$$

eine unimodulare Matrix. Ist A eine ganzzahlige, unimodulare Matrix, dann ist auch A^{-1} ganzzahlig, denn

$$A^{-1} = \frac{1}{\det A} A^+,$$

wobei die Elemente von A^+ Polynome in den Elementen von A sind. Daher besitzt auch das Gleichungssystem $Ax = b$ für ganzzahlige rechte Seite b stets eine ganzzahlige Lösung.

Beispiel 20.1 *Das Gleichungssystem*

$$\begin{aligned} 4x_1 + 3x_2 &= b_1 \\ 7x_1 + 5x_2 &= b_2 \end{aligned}$$

besitzt die ganzzahlige Lösung

$$\begin{aligned}x_1 &= -5b_1 + 3b_2, \\x_2 &= 7b_1 - 4b_2.\end{aligned}$$

Eine ganzzahlige $(m \times n)$ -Matrix A heißt *vollständig unimodular*, wenn für jede quadratische Untermatrix B von A gilt

$$\det B \in \{0, +1, -1\}.$$

Offenbar können die Elemente einer vollständigen unimodularen Matrix A nur die Werte 0 und ± 1 annehmen, denn jede (1×1) Untermatrix von A hat ja ebenfalls $\det a_{ij} \in \{0, \pm 1\}$. Nun gilt

Lemma 20.1 *Ist A vollständig unimodular, so hat $A = (a_{ij})$ nur Elemente $a_{ij} \in \{0, \pm 1\}$. Ferner sind auch die Matrizen $A^t, -A, (A|A), (A|E)$ und $(A|-A)$ vollständig unimodular.*

Beweis: Nachdem für jede quadratische Matrix B stets $\det B = \det B^t$ gilt, ist A^t vollständig unimodular, wenn A vollständig unimodular ist. Aus $\det(-A) = \pm \det A$ folgt eine analoge Aussage für die Matrix $-A$.

Nun sei B eine beliebige quadratische Untermatrix von $(A|E)$. Enthält B nur Spalten von A oder nur Spalten von E , so ist nach Voraussetzung $\det B \in \{0, \pm 1\}$. Nun enthalte B sowohl Spalten von A als auch von E . Nach einer Zeilen- und Spaltenpermutation lässt sich B schreiben als

$$\hat{B} = \left(\begin{array}{c|c} A_1 & 0 \\ \hline A_2 & \hat{E} \end{array} \right)$$

wobei \hat{E} wieder eine Einheitsmatrix ist. Nun gilt $\det B = \pm \det(\hat{B})$ und $\det(\hat{B}) = \det A_1 \det \hat{E} = \det A_1 \in \{0, \pm 1\}$. Also ist auch die Matrix $(A|E)$ vollständig unimodular.

Ist nun B eine Untermatrix von $(A|A)$, so hat B entweder zwei gleiche Spalten und ist daher singular, oder alle Spalten sind verschieden. Dann kann aber B auch als Untermatrix von A aufgefaßt werden: In diesem Falle gilt wieder $\det B \in \{0, \pm 1\}$.

Ein analoges Argument zeigt auch, dass $(A|-A)$ vollständig unimodular ist, wenn man berücksichtigt, dass für Spaltenvektoren a^1, a^2, \dots, a^n gilt $\det(\pm a^1, \pm a^2, \dots, \pm a^n) = \pm \det(a^1, a^2, \dots, a^n)$.

■

Achtung! Aus der vollständigen Unimodularität von A und B folgt nicht notwendigerweise die vollständige Unimodularität von $(A|B)$. Denn wäre dies der Fall, so wäre jede Matrix mit Elementen 0 und 1 vollständig unimodular. Aber die Matrix

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \tag{20.1}$$

hat $\det A = 2$ und ist daher nicht vollständig unimodular.

Wir werden nun zeigen, dass die vollständige Unimodularität von A für Probleme der Form $\max\{c'x | Ax \leq b, x \geq 0\}$ notwendig und hinreichend dafür ist, dass *jede* Basislösung dieser linearen Optimierungsaufgabe ganzzahlig ist.

Satz 20.2 *(Hoffman und Kruskal, 1956, [27])*

Sei A eine ganzzahlige Matrix. Dann sind folgende Aussagen äquivalent:

1. A ist vollständig unimodular.
2. Für beliebiges ganzzahliges b sind die Ecken der polyedrischen Mengen $S(b) = \{x | Ax \leq b, x \geq 0\}$ ganzzahlig.
3. Jede quadratische, nichtsinguläre Untermatrix von A hat eine ganzzahlige Inverse.

Beweis: (nach Veinott und Dantzig [50]):

1. \Rightarrow 2. Zunächst gilt nach Lemma 20.1, dass mit A auch die um die Einheitsmatrix erweiterte Koeffizientenmatrix $(A|E)$ vollständig unimodular ist. Somit ist jede Basismatrix A_B unimodular und daher hat das Gleichungssystem $A_B x_B = b$ eine ganzzahlige Lösung. Daher ist jede Ecke von $S(b)$ ganzzahlig.
2. \Rightarrow 3. Sei A_B eine Basismatrix und \tilde{b}_i der i -te Spaltenvektor von A_B^{-1} . Wir zeigen zunächst, dass \tilde{b}_i ganzzahlig ist. Dann ist jede Basisinverse des Systems $Ax + Ey = b$ ganzzahlig.

Wählen wir dazu t beliebig, aber ganzzahlig so, dass $\tilde{b}_i + t \geq 0$ gilt. Setzen wir nun $b(t) = A_B t + e_i$, wobei e_i der i -te Einheitsvektor ist. Nun gilt

$$x_B = A_B^{-1} b(t) = A_B^{-1} A_B t + A_B^{-1} e_i = t + \tilde{b}_i \geq 0.$$

Daher entspricht $(x_B, x_N) = (t + \tilde{b}_i, 0)$ einer zulässigen Ecke von $S(b(t))$, die nach Voraussetzung ganzzahlig ist. Da t ganzzahlig ist, ist damit auch \tilde{b}_i ganzzahlig.

Nun zeigen wir, dass jede nichtsinguläre Untermatrix F von A eine ganzzahlige Inverse besitzt. Ergänzen wir F durch Spalten von E zu einer Basismatrix, so gilt

$$A_B = \left(\begin{array}{c|c} F & 0 \\ \dots & \hat{E} \end{array} \right).$$

Nun ist $A_B^{-1} = \left(\begin{array}{c|c} F^{-1} & 0 \\ \dots & \hat{E} \end{array} \right)$ und da A_B^{-1} ganzzahlig ist, ist auch F^{-1} ganzzahlig.

3. \Rightarrow 1. Sei F eine beliebige quadratische, nichtsinguläre Untermatrix von A . Nach Voraussetzung ist F^{-1} ganzzahlig. Daher sind $\det F$ und $\det F^{-1}$ ganze Zahlen. Aus $\det F \cdot \det F^{-1} = \det(F \cdot F^{-1}) = \det E = 1$ folgt $\det F = \pm 1$, also A vollständig unimodular. ■

Als Korollar erhält man

Korollar 20.3 1. Ist die Koeffizientenmatrix von $\max\{c'x | Ax \leq b, x \geq 0\}$ bei ganzzahligem b und c vollständig unimodular, so haben sowohl das primale als auch das duale Problem polyedrische Mengen mit nur ganzzahligen Ecken.

2. Sind \underline{b} , \bar{b} und c ganzzahlig und ist A vollständig unimodular, so sind die Ecken von $S = \{x | \underline{b} \leq Ax \leq \bar{b}, 0 \leq x \leq c\}$ ganzzahlig.

Beweis: zu 2. S lässt sich in der Form

$$S = \left\{ x \mid \begin{pmatrix} A \\ -A \\ E \end{pmatrix} x \leq \begin{pmatrix} \bar{b} \\ -\bar{b} \\ c \end{pmatrix}, x \geq 0 \right\}$$

schreiben und nach Lemma 20.1 ist die zugehörige Koeffizientenmatrix vollständig unimodular. ■

Die Bedeutung des Satzes von Hoffman und Kruskal liegt darin, dass lineare Programme mit vollständig unimodularer Restriktionsmatrix stets ganzzahlige Basislösungen haben. Löst man daher ein derartiges Problem etwa durch das Simplexverfahren, so erhält man automatisch eine ganzzahlige Lösung.

Wie stellt man nun fest, ob eine gegebene Matrix $A = (a_{ij})$ vollständig unimodular ist? Eine notwendige Bedingung ist offenbar, dass alle Elemente a_{ij} entweder 0 oder ± 1 sind. Wie die Matrix (20.1) zeigt, ist diese Bedingung aber nicht hinreichend. Mittels eines Dekompositionssatzes von Seymour [43] kann man in polynomialer Zeit, nämlich $O((m+n)^4 m)$ Schritten, testen, ob eine gegebene Matrix vollständig unimodular ist. Ein derartiger Test ist ausführlich in Schrijver [42] beschrieben. Im Fall, dass jede Spalte der Matrix A nur zwei Einträge ungleich 0 hat, liefert der folgende Satz eine Bedingung für die vollständige Unimodularität von A :

Satz 20.4 (Heller und Tompkins, 1956 [25])

Eine Matrix A mit Elementen $0, +1, -1$ ist vollständig unimodular, wenn gilt

1. *Jede Spalte von A hat höchstens zwei Elemente $\neq 0$.*
2. *Die Zeilen von A lassen sich in zwei Klassen R_1 und R_2 einteilen, wobei gilt:*
 - (a) *Enthält eine Spalte zwei Elemente $\neq 0$ mit demselben Vorzeichen, so sind die beiden zugehörigen Zeilen nicht in derselben Klasse.*
 - (b) *Enthält eine Spalte zwei Elemente $\neq 0$ mit verschiedenem Vorzeichen, so sind die zugehörigen Zeilen in derselben Klasse.*

Beweis: Wir zeigen durch vollständige Induktion, dass unter den Voraussetzungen des Satzes jede $(k \times k)$ Untermatrix von A die Determinante $0, \pm 1$ hat. Für $k = 1$ ist dies sicher richtig, da $a_{ij} \in \{0, +1, -1\}$. Nehmen wir an, diese Behauptung gelte für alle Untermatrizen mit $k-1$ Zeilen und Spalten. Es sei nun B eine quadratische Untermatrix von A mit k Zeilen. Wir unterscheiden folgende Fälle:

1. Hat B eine Nullspalte, dann ist aber $\det B = 0$.
2. Hat B eine Spalte mit nur einem Element $\neq 0$, so wird $\det B$ nach dieser Spalte entwickelt. Aus der Induktionsannahme folgt $\det B \in \{0, \pm 1\}$.
3. Jede Spalte von B enthält zwei Elemente $\neq 0$. Wegen 2a) und 2b) gilt für jede Spalte j von B :

$$\sum_{i \in R_1} a_{ij} = \sum_{i \in R_2} a_{ij} \quad (j = j_1, \dots, j_k).$$

Also sind die Zeilen von B linear abhängig und daher ist $\det B = 0$. ■

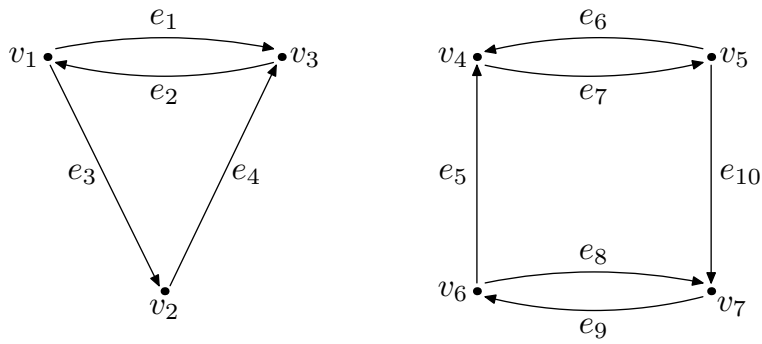


Abbildung 20.1: Ein gerichteter Graph mit 7 Knoten

Die Knoten-Kanten Inzidenzmatrix eines gerichteten Graphen ist definiert durch die Einträge

$$a_{ij} := \begin{cases} 1, & \text{falls die Kante } e_j \text{ vom Knoten } i \text{ ausgeht,} \\ -1, & \text{falls die Kante } e_j \text{ in den Knoten } i \text{ einmündet,} \\ 0, & \text{sonst.} \end{cases}$$

Für obiges Beispiel lautet die Inzidenzmatrix

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}
v_1	1	-1	1							
v_2			-1	1						
v_3	-1	1		-1						
v_4					-1	-1	1			
v_5						1	-1			1
v_6					1			1	-1	
v_7								-1	1	-1

Die Wahl $R_1 = V$ und $R_2 = \emptyset$ im Satz von Heller-Tompkins zeigt, dass eine Knoten-Pfeil Inzidenzmatrix vollständig unimodular ist. Nach Korollar 20.5 sind die Zeilen der Knoten-Kanten Inzidenzmatrix eines gerichteten Graphen linear abhängig.

Kapitel 21

Transportprobleme

Gegeben seien m Fabriken, in denen jeweils die Warenmengen a_i ($1 \leq i \leq m$) erzeugt werden. Die Waren sollen vollständig an n Abnehmer versandt werden, wobei der j -te dieser Abnehmer die Warenmenge b_j ($1 \leq j \leq n$) benötigt. Die Transportkosten für den Versand einer Wareneinheit vom Erzeuger i zum Abnehmer j bezeichnen wir mit c_{ij} . Die Variable x_{ij} gibt an, wieviel Wareneinheiten vom Erzeuger i an den Abnehmer j gesandt werden.

Man kann ohne Beschränkung der Allgemeinheit annehmen, dass

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

gilt. Denn wird mehr erzeugt als verbraucht, so führt man einen zusätzlichen Verbraucher $n + 1$ ein, der die nichtgelieferten Waren übernimmt. Die Größen $c_{i,n+1}$ sind dann Lagerkosten bei den Erzeugern i , $i = 1, 2, \dots, m$. Wird aber mehr gebraucht als erzeugt, so führt man einen zusätzlichen Erzeuger $m + 1$ ein. Die Kosten $c_{m+1,j}$ sind dann Strafkosten für den Fehlbedarf der Abnehmers j , $j = 1, 2, \dots, n$.

Die Minimierung der Gesamtkosten führt auf folgendes Modell, genannt Transportproblem:

$$\text{Minimiere } \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

unter den Nebenbedingungen

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= a_i \quad (1 \leq i \leq m), \\ \sum_{i=1}^m x_{ij} &= b_j \quad (1 \leq j \leq n), \\ x_{ij} &\geq 0 \quad (1 \leq i \leq m, 1 \leq j \leq n). \end{aligned} \tag{21.1}$$

Aufgrund der speziellen Gestalt der Restriktionen kann das Simplexverfahren so modifiziert werden, dass man allein mit Additionen und Subtraktionen eine Optimallösung erhält. Daraus folgt erneut, dass man bei ganzzahligen Ausgangswerten eine ganzzahlige Optimallösung erhält.

Die Koeffizientenmatrix eines Transportproblems (kurz: *Transportmatrix*) hat die Gestalt

$$A = \left(\begin{array}{cccc|cccc|c|cccc} 1 & 1 & \dots & 1 & 1 & 1 & \dots & 1 & \dots & 1 & 1 & \dots & 1 \\ \hline 1 & & & & 1 & & & & \dots & & 1 & 1 & \dots & 1 \\ \hline & 1 & & & & 1 & & & & & & 1 & & \\ & & \ddots & & & & \ddots & & & & & & \ddots & \\ & & & 1 & & & & 1 & & & & & & 1 \end{array} \right). \quad (21.2)$$

Die ersten m Zeilen dieser Matrix entsprechen den Erzeugern, die letzten n Zeilen der Matrix entsprechen den Abnehmern. Jede Spalte $i \cdot n + j$ gibt an, dass j von i aus beliefert werden kann. Da die Summe über die ersten m Zeilen gleich der Summe über die letzten n Zeilen ist, sind die Zeilenvektoren der Matrix A linear abhängig. Streicht man jedoch die letzte Zeile, so erhält man eine Matrix mit dem Rang $m + n - 1$, denn wie man leicht sieht, sind die ersten n Spalten, sowie die $m - 1$ Spalten mit den Indizes $kn + 1$, $k = 1, \dots, m - 1$, linear unabhängig.

Wie wir sehen werden, kann die Transportmatrix kann als Knoten-Kanten Inzidenzmatrix eines bipartiten Graphen $G = (I, J; E)$ aufgefasst werden. Dies ermöglicht eine besonders anschauliche Interpretation des Simplexverfahren adaptiert für Transportprobleme. Aus diesem Grund sehen wir uns zunächst näher den Zusammenhang zwischen Graphen und Transportproblemen an.

Gegeben sei ein ungerichteter Graph $G = (V, E)$ mit endlicher *Knotenmenge* V und *Kantenmenge* E . Jede Kante $e = [i, j]$ verbindet zwei Knoten i und j von V . Zwei Knoten, die durch eine Kante verbunden sind, heißen *benachbart*. Der Grad $d(i)$ eines Knotens $i \in V$ ist die Anzahl der Kanten, die mit i inzidieren. Ist $d(i) = 1$, so bezeichnet man den Knoten i als ein *Blatt*. Ein Graph $\bar{G} = (V, \bar{E})$ heißt *Teilgraph* von $G = (V, E)$, falls $\bar{E} \subseteq E$ gilt. Ein Graph $G' = (V', E')$ mit $V' \subseteq V$ und $E' \subseteq E$ heißt *Untergraph* von G . Gilt für eine Folge von Knoten $P = (i_0, i_1, i_2, \dots, i_n)$, dass $[i_k, i_{k+1}]$ für $k = 0, 1, \dots, n - 1$ eine Kante in E ist und sind die Knoten i_0, i_1, \dots, i_{n-1} paarweise verschieden, so bezeichnet man P als einen *Weg*. Gilt $i_0 = i_n$, so liegt ein *Kreis* vor. Kreise (i_0, i_0) bezeichnet man als *Schlingen*. Ein Graph $G = (V, E)$ heißt *einfach*, wenn er wie oben definiert ist und keine Schlingen besitzt. Die Anzahl der Kanten eines Weges bzw. eines Kreises nennt man die *Länge des Weges (Kreises)*. Ein Weg bzw. Kreis mit einer geraden Anzahl von Kanten heißt ein *gerader Weg* bzw. *gerader Kreis*.

Ein Graph heißt *zusammenhängend*, wenn es zu je zwei Knoten $i, j \in V$ einen Weg gibt mit $i_0 = i$ und $i_n = j$. Ist nun V' die Menge aller Knoten j , die von i aus über einen Weg erreichbar sind, und gilt $E' := \{[i, j] \in E \mid i, j \in V'\}$, so nennt man den Untergraphen $G' = (V', E')$ die *Zusammenhangskomponente* von i . Zwei Zusammenhangskomponenten eines Graphen sind entweder gleich oder knotendisjunkt. Ein *Baum* ist ein zusammenhängender Graph, der keine Kreise besitzt.

Bevor wir beweisen, dass Bäume Basislösungen von Transportproblemen entsprechen, geben wir einige Charakterisierungen von Bäumen an. Mit $|V|$ und $|E|$ bezeichnen wir die Anzahl von Knoten bzw. Kanten in $G = (V, E)$.

Satz 21.1 *Charakterisierung von Bäumen*

Sei $G = (V, E)$ ein ungerichteter Graph mit $1 \leq |V| < \infty$. Dann sind äquivalent:

1. G ist ein Baum.
2. G ist zusammenhängend und $|E| = |V| - 1$.
3. G ist kreisfrei und $|E| = |V| - 1$.
4. Zu je zwei Knoten i, j gibt es genau einen Weg von i nach j .
5. G ist zusammenhängend und jeder Teilgraph mit weniger als $|V| - 1$ Kanten ist nicht zusammenhängend. (G ist ein minimal zusammenhängender Graph).
6. G ist kreisfrei. Fügt man eine neue Kante $[i, j] \in E$ hinzu, so enthält der neue Graph einen Kreis. (G ist ein maximal kreisfreier Graph).

Beweis: **1.** \implies **2.** Sei $n := |V|$. Für $n = 1$ ist die Aussage richtig. Daher können wir annehmen, dass jeder Baum mit n Knoten $n - 1$ Kanten enthält. Sei nun G ein Baum mit n Knoten. G enthält mindestens ein Blatt i_0 , denn anderenfalls kann man in einem beliebigen Knoten starten und zu einem Nachbarn weitergehen, von dort über eine andere Kante zu einem neuen Knoten und so fort. Da es nur endlich viele Knoten gibt, kommt man nach höchstens n Schritten zu einem bereits besuchten Knoten zurück. Dies bedeutet aber, dass G einen Kreis besitzt, was ausgeschlossen ist, da G ein Baum ist. Streicht man nun i_0 und die einzige mit i_0 inzidierende Kante, so erhält man wieder einen zusammenhängenden und kreisfreien Graphen, also einen Baum mit $n - 1$ Knoten, der nach Induktionsvoraussetzung $n - 2$ Kanten besitzt. Daher hat G $n - 1$ Kanten.

2. \implies **3.** Sei G zusammenhängend mit n Knoten und $n - 1$ Kanten. Hätte G Kreise, so könnte man in jedem dieser Kreise eine Kante streichen und man erhielte einen zusammenhängenden, kreisfreien Graphen, also einen Baum mit n Knoten und weniger als $n - 1$ Kanten im Widerspruch zu 2.

3. \implies **4.** Sei G kreisfrei mit n Knoten und $n - 1$ Kanten. In G gibt es von einem Knoten i zu einem Knoten $j \neq i$ nicht zwei oder mehr verschiedene Wege, denn sonst enthielte G einen Kreis. Nehmen wir an, von i nach j gäbe es keinen Weg. Sei nun G_1 die Zusammenhangskomponente von G , die i enthält, und G_2 die Zusammenhangskomponente, die j enthält. Die Knotenmengen von G_1 und G_2 sind disjunkt. Gibt es noch weitere Knoten, die weder in G_1 noch in G_2 liegen, so seien G_3, \dots, G_r ihre entsprechenden Zusammenhangskomponenten. Jede Zusammenhangskomponente ist kreisfrei, also ein Baum mit n_i Knoten und $n_i - 1$ Kanten, $i = 1, 2, \dots, r$. Es gilt

$$\sum_{i=1}^r n_i = n \quad \text{und} \quad \sum_{i=1}^r (n_i - 1) = n - r.$$

Da $r \geq 2$ hat G also weniger als $n - 1$ Kanten im Widerspruch zur Annahme $|E| = |V| - 1$. Also ist G zusammenhängend, d.h. zwischen jedem Knotenpaar gibt es einen Weg.

4. \implies **5.** Sei G ein Graph, in dem es zu jedem Knotenpaar $i, j \neq i$ einen Weg von i nach j gibt. Streicht man in G eine Kante $[i, j]$, dann ist G nicht mehr zusammenhängend, da diese Kante nach Voraussetzung der einzige Weg von i nach j war.

5. \implies **6.** Sei G zusammenhängend. Enthielte G einen Kreis, so könnte man eine Kante in diesem Kreis streichen und erhielte noch immer einen zusammenhängenden Graphen im Widerspruch zur Annahme, dass jeder Teilgraph mit weniger als $|V| - 1$ Kanten nicht zusammenhängend ist.

Fügt man nun zu G eine neue Kante $[i, j]$ hinzu, so bildet diese zusammen mit dem bereits in G vorhandenen Weg von i nach j einen Kreis.

6. \implies 1. Seien i und $j \neq i$ zwei beliebige Knoten in G . Entweder ist $[i, j]$ eine Kante von G oder durch Hinzufügen dieser neuen Kante entsteht ein Kreis, was bedeutet, dass es in G bereits einen Weg von i nach j gegeben hat. Daher ist G zusammenhängend und kreisfrei, also ein Baum. ■

Eine $n \times m$ Matrix $A = (a_{ik})$ heißt *Knoten-Kanten Inzidenzmatrix* eines Graphen $G = (V, E)$ mit $|V| = n$, $|E| = m$, wenn

$$a_{ik} := \begin{cases} 1 & \text{falls Knoten } i \text{ inzidiert mit Kante } k, \\ 0 & \text{falls sonst.} \end{cases} \quad (21.3)$$

Wir ordnen nun einem Transportproblem auf folgende Weise einen Graphen zu: Jedem Index i und jedem Index j entspricht ein Knoten des Graphen. Die Kanten seien durch $[i, j]$ mit $1 \leq i \leq m, 1 \leq j \leq n$ gegeben. Wir nennen diesen Graphen *Transportgraph*. Offenbar ist die Koeffizientenmatrix (21.2) eines Transportproblems die Knoten-Kanten Inzidenzmatrix des Transportgraphen. Der Transportgraph hat noch eine spezielle Eigenschaft, nämlich er ist *bipartit*, d.h. seine Knotenmenge läßt sich in zwei nichtleere, disjunkte Mengen I und J zerlegen, so dass es nur Kanten zwischen einem Knoten $i \in I$ und einem Knoten $j \in J$ gibt. Die Knotenmenge I entspricht den Produktionsstätten, die Knotenmenge J den Abnehmern im Transportmodell.

Lemma 21.2 *Jeder Kreis in einem bipartiten Graphen hat eine gerade Anzahl von Kanten, die abwechselnd blau und rot gefärbt werden können.*

Beweis: Da es in einem Kreis in einem bipartiten Graphen $G = (I, J; E)$ zu jeder Kante von einem Knoten $i \in I$ zu einem Knoten $j \in J$ auch wieder eine Kante geben muß, die von der Menge J in die Menge I zurückführt, ist jeder Kreis in G gerade. Die Kanten, die von I nach J führen, färben wir blau, die Kanten, die von J nach I zurückführen, färben wir rot. Dann sind alle Kanten des Kreises abwechselnd blau und rot gefärbt. ■

Es gilt nun

Satz 21.3 *Jeder Menge linear unabhängiger Spalten der Transportmatrix (21.2) entspricht eindeutig ein Teilgraph des Transportgraphen, der keine Kreise enthält.*

Beweis: Nach Definition entspricht jeder Spalte von A eine Kante in G . Wir zeigen: Sind die Spalten linear abhängig, so enthält der zugehörige Teilgraph einen Kreis und umgekehrt.

1. G enthalte einen Kreis. Da G bipartit ist, enthält G einen geraden Kreis. Seine Kanten $[i, j]$ können abwechselnd blau und rot gefärbt werden. Nun gilt

$$\sum_{[i,j] \text{ blau}} a_{ij} = \sum_{[i,j] \text{ rot}} a_{ij},$$

da jeder Knoten des Kreises mit einer blauen und roten Kante inzidiert. Daher sind die zugehörigen Spaltenvektoren a_{ij} linear abhängig.

2. Sei nun $\{a_{ij} \mid (i, j) \in K\}$ eine Menge linear abhängiger Spalten. Dann gibt es ein $\alpha = (\alpha_{ij}) \neq 0$ mit $\sum_K \alpha_{ij} \cdot a_{ij} = 0$. Sei $K' := \{(i, j) \mid \alpha_{ij} \neq 0\}$. Dann muss jeder Knoten, der mit einer Kante in K' inzidiert, auch mit einer weiteren Kante in K' inzidieren. Das bedeutet, dass im Untergraphen $G' = (V', E')$ mit $E' = \{[i, j] \mid (i, j) \in K'\}$ jeder Knoten i und j einen Grad ≥ 2 hat. Damit enthält G' einen Kreis. ■

Anmerkung. Whitney [52] axiomatisierte 1935 den Begriff der linearen Abhängigkeit und führte dazu das kombinatorische Konzept eines *Matroides* ein. Sei E eine endliche Menge und \mathcal{I} eine nichtleere Familie von Untermengen von E , so dass gilt

1. Mit $F \in \mathcal{I}$ enthält \mathcal{I} auch alle Teilmengen von F .
2. Sind F und G in \mathcal{I} mit $|F| < |G|$, so gibt es ein Element $e \in G \setminus F$, so dass $F \cup \{e\}$ wieder in \mathcal{I} liegt (Steinitz'sche Austauschenschaft).

Das System (E, \mathcal{I}) heißt ein *Matroid*, die Mengen $F \in \mathcal{I}$ werden als *unabhängige Mengen* bezeichnet.

Es ist leicht überprüfbar, dass jedes System linear unabhängiger Spalten einer Matrix A diese Matroideigenschaften erfüllt. Sie bilden ein Matrixmatroid. Ferner kann man auch leicht zeigen, dass alle kreisfreien Untergraphen eines Graphen G die Matroideigenschaften erfüllen. Sie bilden ein graphisches Matroid. Satz 21.3 besagt nun: *Ist A die Inzidenzmatrix eines Graphen G , so sind das Matrixmatroid gegeben durch A und das graphische Matroid gegeben durch G isomorph.*

Teilgraphen von G , die alle Knoten von G enthalten, zusammenhängend sind und keine Kreise enthalten, heißen Gerüste von G . Offenbar ist jedes Gerüst ein Baum. Gerüste sind nach Satz 21.1, Punkt 6 maximal-kreisfreie Graphen. Da der einem Transportproblem zugeordnete Graph $m + n$ Knoten besitzt, hat nach Satz 21.1 ein Gerüst dieses Graphen $m + n - 1$ Kanten. Diese entsprechen nach Satz 21.3 einer maximalen Menge von linear unabhängigen Spalten der Transportmatrix (21.2), also einer Basis B . Und umgekehrt entspricht jede Basis des Transportproblems einem Gerüst im Transportgraphen. Damit haben wir den wichtigen Zusammenhang zwischen Basen und Gerüsten des Transportgraphen hergestellt:

Satz 21.4 *Jeder Basis des Transportproblems entspricht eindeutig ein Gerüst im zugehörigen Transportgraphen G und umgekehrt entspricht jedem Gerüst in G eine Basis des Transportproblems.*

Kehren wir nun zur Lösung des Transportproblems durch das Simplexverfahren zurück. Es sei B eine Basis. Um zu überprüfen, ob die zugehörige Basislösung optimal ist, müssen die reduzierten Kostenkoeffizienten $\tilde{c}'_N = c'_N - c'_B A_B^{-1} A_N$ bestimmt werden. Setzt man

$$u'_{AB} = c'_B, \quad (21.4)$$

so erhält man für die reduzierten Kostenkoeffizienten

$$\tilde{c}'_N = c'_N - u'_{AB} A_B^{-1} A_N = c'_N - u'_{AN}. \quad (21.5)$$

Da A_B regulär ist, ist die Gleichung $u'A_B = c'_B$ eindeutig lösbar. Bezeichnen wir die Lösung mit $u' := (u_1, \dots, u_m, v_1, \dots, v_{n-1})$ und ordnen wir diese Werte den Knoten des Transportgraphen i_1, i_2, \dots, i_m und j_1, j_2, \dots, j_n zu, wobei wir $v_n := 0$ setzen. Die Basis B entspricht einem Gerüst von G . Dessen Kanten $[i, j]$ werden die Größen c_{ij} , $(i, j) \in B$, zugeordnet. Das Gleichungssystem (21.4) besagt, dass für jedes $(i, j) \in B$ die Gleichung $u_i + v_j = c_{ij}$ gelten muß. Da die Basis einem Baum entspricht, kann man also das Gleichungssystem $u'A_B = c'_B$ dadurch rekursiv auflösen, indem man im Knoten j_n beginnt und für die mit diesem Knoten inzidierenden Kanten $[i, n]$ die Werte $u_i := c_{in}$ festsetzt. Für die weiteren mit i inzidierenden Kanten $[i, j]$ muß $u_i + v_j = c_{ij}$ gelten. Daraus ist eindeutig v_j berechenbar. Da ausgehend vom Knoten j_n jeder Knoten des Graphen auf einem eindeutig durch das Gerüst festgelegten Weg erreichbar ist, ist somit das Gleichungssystem $u'A_B = c'_B$ eindeutig rekursiv auflösbar.

Ist u_1, u_2, \dots, u_m und v_1, v_2, \dots, v_{n-1} einmal berechnet, so ergeben sich die reduzierten Kostenkoeffizienten aus $\bar{c}'_N = c'_N - u'A_N$. Setzen wir

$$\bar{c}_N := u'A_N$$

und $v_n := 0$, dann ist

$$\bar{c}_{ij} = u_i + v_j \quad \text{für alle } (i, j) \in N.$$

Aus der Theorie des Simplexverfahrens ist bekannt, dass bei einem Minimierungsproblem eine Basislösung (B, N) optimal ist, wenn die reduzierten Kostenkoeffizienten nichtnegativ sind. Somit ist die Basislösung optimal, wenn für die reduzierten Kostenkoeffizienten $\bar{c}_{ij} \geq 0$, $(i, j) \in N$, gilt. Also muß für alle Indizes $(i, j) \in N$ gelten:

$$c_{ij} \geq \bar{c}_{ij}. \quad (21.6)$$

Ist (21.6) nicht erfüllt, so wird eine Nichtbasisvariable x_{rs} mit $c_{rs} - \bar{c}_{rs} < 0$ in die Basis aufgenommen. Das bedeutet, dass im Transportgraph die Kante $[r, s]$ zum augenblicklichen Gerüst hinzugefügt wird. Dadurch entsteht nach Satz 21.1 genau ein Kreis C mit einer geraden Anzahl von Kanten. Die neue Kante $[r, s]$ wird blau gefärbt. Da der Kreis C gerade ist, können wir seine Kanten abwechselnd blau und rot färben. Eine Kante aus dem Kreis muß gestrichen werden, damit man ein neues Gerüst (eine neue Basislösung) erhält. Dieser Basisaustauschschritt läßt sich nun aber ebenfalls leicht graphisch beschreiben. Dazu ordnen wir zunächst die Werte der Basisvariablen x_{ij} , $(i, j) \in B$, den Kanten des Gerüsts zu. Die Restriktionen

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= 1 & (i = 1, \dots, m), \\ \sum_{i=1}^m x_{ij} &= 1 & (j = 1, \dots, n). \end{aligned}$$

des Transportproblem besagen, dass in jedem Knoten i und j die Summe der Basisvariablen x_{ij} auf Kanten, die mit dem Knoten i bzw. dem Knoten j inzidieren, konstant sein muß. Dies bedeutet: vergrößert man den Wert von x_{ij} um ein δ , dann müssen auf anderen mit i und j inzidierenden Kanten des Gerüsts die Werte der zugehörigen Basisvariablen um δ verringert werden.

Wird also die neue Kante $[r, s]$ zum Gerüst hinzugefügt und soll die neu in die Basis aufzunehmende Variable x_{rs} den Wert δ erhalten, so muss man die Werte der Basisvariablen entlang des Kreises C modifizieren. Da die Kante $[r, s]$ blau ist, kann man δ von den Werten der Basisvariablen auf den roten Kanten abziehen und zu den Werten der Basisvariablen, die blauen Kanten entsprechen, hinzuaddieren. Da C gerade ist, wird durch diese Vorgangsweise garantiert, dass in jedem Knoten die Summe aller Basisvariablen konstant bleibt. Dann bestimmt man

$$\delta_0 := \min\{x_{ij} \mid (i, j) \text{ rot}\}.$$

Dieser Wert werde von der roten Kante $[i_0, j_0]$ angenommen. Setzt man

$$\begin{aligned} x_{ij} &:= x_{ij} + \delta_0, & \text{falls } (i, j) \text{ blau,} \\ x_{ij} &:= x_{ij} - \delta_0 & \text{falls } (i, j) \text{ rot} \end{aligned}$$

und

$$B := B \cup \{(r, s)\} \setminus \{(i_0, j_0)\},$$

so erhält man eine neue Basislösung. Die Variable x_{i_0, j_0} hat nun den Wert 0 und wird eine Nichtbasisvariable. Streicht man die Kante $[i_0, j_0]$ im Kreis C , so erhält man das Gerüst, das der neuen zulässigen Basislösung entspricht.

Eine Ausgangslösung für ein Transportproblem kann auf folgende Weise bestimmt werden: Wir fassen die Variablen x_{ij} in einem Tableau mit Zeilen i , $1 \leq i \leq m$, und Spalten j , $1 \leq j \leq n$, zusammen. Das Tableau wird durch die Zeilen- und Spaltensummen a_i und b_j ergänzt:

	b_1	b_2	\dots	b_n
a_1	x_{11}	x_{12}		x_{1n}
a_2	x_{21}	x_{22}		x_{2n}
\vdots				
a_m	x_{m1}	x_{m2}	\dots	x_{mn}

Nun geht man nach folgendem Schema vor:

Bestimmung einer Ausgangslösung

1. Wähle im Tableau ein Indexpaar (i_0, j_0) und setze

$$x_{i_0 j_0} := \min(a_{i_0}, b_{j_0})$$

2. Tableauänderung

- 2.1 Falls $a_{i_0} < b_{j_0}$, streiche Zeile i_0 und setze $b_{j_0} := b_{j_0} - a_{i_0}$.
- 2.2 Falls $b_{j_0} < a_{i_0}$, streiche Spalte j_0 und setze $a_{i_0} := a_{i_0} - b_{j_0}$.
- 2.3 Gilt $a_{i_0} = b_{j_0}$, und enthält das Tableau mindestens so viele Zeilen als Spalten, so streiche Zeile i_0 und setze $b_{j_0} := 0$. Andernfalls streiche Spalte j_0 und setze $a_{i_0} := 0$.

3. Hat das Tableau keine Zeilen und Spalten mehr, so terminiere. Andernfalls gehe nach 1.

Durch dieses Verfahren wird stets eine Basis für das Transportproblem bestimmt: Denn wählt man ein Indexpaar (i_0, j_0) und streicht nun die Zeile i_0 [Spalte j_0] im Tableau, so sind in der Transportmatrix A (21.2) alle Spalten gestrichen worden, die das Element 1 in der i_0 -ten Zeile [$(m + j_0)$ -ten Zeile] enthalten. Daher ist die zu (i_0, j_0) gehörige Spalte von A von allen noch nicht gestrichenen Spalten von A linear unabhängig. Daher bestimmen die $n + m - 1$ ausgewählten Spalten von A eine Basis.

Zwei einfache Kriterien zur Wahl von (i_0, j_0) sind:

Nordwesteckenregel (linke, obere Eckenregel):

i_0 sei der Index der ersten nichtgestrichenen Zeile, j_0 der Index der ersten nichtgestrichenen Spalte im Tableau.

Regel der geringsten Kosten:

Sei I die Indexmenge der nichtgestrichenen Zeilen und J die Indexmenge der nichtgestrichenen Spalten. Dann wird (i_0, j_0) bestimmt durch

$$c_{i_0 j_0} = \min\{c_{ij} | i \in I, j \in J\}.$$

(Bei Mehrdeutigkeit des minimalen Elementes wähle man ein beliebiges minimales Element aus.) Die Regel der geringsten Kosten hat natürlich einen größeren Rechenaufwand als die Nordwesteckenregel, da das Minimum der Kostenelemente gesucht werden muß. Allerdings ist dafür auch die Ausgangslösung besser und es werden i.a. deutlich weniger Simplexiterationen bis zur Optimallösung benötigt. Eine Verringerung des Aufwandes erreicht man durch die folgende Rollende Zeilen-Minimum Regel, die darauf beruht, dass ja für jede Zeile i in Basiselement x_{ij} bestimmt werden muß.

Rollende Zeilen Minimum Regel:

Wähle beginnend von $i_0 := 1$ den Zeilenindex zyklisch (unter den noch nicht gestrichenen Zeilen) und bestimme j_0 durch

$$c_{i_0 j_0} = \min\{c_{i_0 j} | j \in J\}.$$

Beispiel 21.1 Gegeben seien die Werte $a_1 = 4$, $a_2 = 7$, $a_3 = 4$ und $b_1 = 2$, $b_2 = 4$, $b_3 = 6$, $b_4 = 3$ und c_{ij} , $1 \leq i \leq 3$; $1 \leq j \leq 4$, laut folgendem Tableau

$x :$	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;">2</td> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;">6</td> <td style="width: 20px; height: 20px;">3</td> </tr> <tr> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> <tr> <td style="width: 20px; height: 20px;">7</td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> <tr> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table>		2	4	6	3	4					7					4				
	2	4	6	3																	
4																					
7																					
4																					

$c :$	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px; height: 20px;">2</td> <td style="width: 20px; height: 20px;">3</td> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;">1</td> </tr> <tr> <td style="width: 20px; height: 20px;">5</td> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;">2</td> <td style="width: 20px; height: 20px;">3</td> </tr> <tr> <td style="width: 20px; height: 20px;">4</td> <td style="width: 20px; height: 20px;">2</td> <td style="width: 20px; height: 20px;">8</td> <td style="width: 20px; height: 20px;">6</td> </tr> </table>	2	3	4	1	5	4	2	3	4	2	8	6
2	3	4	1										
5	4	2	3										
4	2	8	6										

Die Nordwesteckenregel liefert folgende Ausgangslösung:

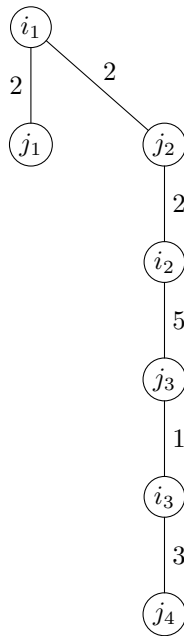


Abbildung 21.1: Basislösung bestimmt durch die Nordwesteckenregel

$$x : \begin{array}{c|cccc} & 2 & 4 & 6 & 3 \\ \hline 4 & 2 & 2 & & \\ 7 & & 2 & 5 & \\ 4 & & & 1 & 3 \end{array} \quad \text{mit } z = \sum_{i,j} c_{ij} \cdot x_{ij} = 54.$$

Die graphische Darstellung als Baum wird in Abbildung 21.1 gezeigt.

Durch die Regel der geringsten Kosten erhält man als Ausgangslösung:

$$x : \begin{array}{c|cccc} & 2 & 4 & 6 & 3 \\ \hline 4 & 1 & & & 3 \\ 7 & 1 & 0 & 6 & \\ 4 & & 4 & & \end{array} \quad \text{mit } z = \sum_{i,j} c_{ij} x_{ij} = 30.$$

Der zu dieser Basislösung gehörige Baum wird in Abbildung 21.2 dargestellt.

Zusammenfassend erhält man also folgenden Algorithmus zur Lösung von Transportproblemen:

Algorithmus 6 Algorithmus zur Lösung von Transportproblemen

Anfangsdaten: a_i ($1 \leq i \leq m$)
 b_j ($1 \leq j \leq n$),
 c_{ij} ($1 \leq i \leq m, 1 \leq j \leq n$),
 $v_n := 0$.

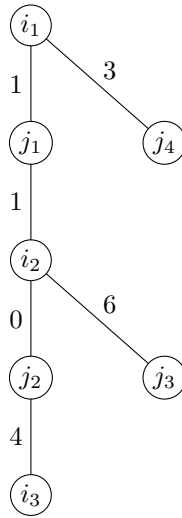


Abbildung 21.2: Basislösung bestimmt durch die Regel der geringsten Kosten

1. Bestimme eine Ausgangslösung. Die Basisvariablen seien x_{ij} mit $(i, j) \in B$.

2. Berechne aus

$$u_i + v_j = c_{ij} \quad (i, j) \in B$$

rekursiv die Größen u_i , $i = 1, 2, \dots, m$, und v_j , $j = 1, 2, \dots, n - 1$.

3. Berechne für $(i, j) \notin B$ die Größen \bar{c}_{ij} als

$$\bar{c}_{ij} := u_i + v_j \quad (i, j) \notin B.$$

4. Ist $c_{ij} \geq \bar{c}_{ij}$ für alle $(i, j) \notin B$, so ist die augenblickliche Basislösung optimal. Terminiere. Anderenfalls wähle (r, s) so, dass $c_{rs} < \bar{c}_{rs}$ ist und gehe zu 5.

5. Füge die Kante $[r, s]$ zum Gerüst, das der Basis B entspricht, hinzu.

6. Bestimme den Kreis C , der durch Hinzufügen der Kante $[r, s]$ entsteht. Färbe die Kanten des Kreises C beginnend mit der blauen Kante $[r, s]$ abwechselnd blau und rot.

7. $\delta := \min\{x_{ij} \mid [i, j] \text{ rot}\}$. Dieser Wert werde auf der Kante $[i_0, j_0]$ angenommen.

8. Setze

$$\begin{aligned} x_{rs} &:= \delta \\ x_{ij} &:= x_{ij} + \delta \quad \text{für } [i, j] \text{ blau} \\ x_{ij} &:= x_{ij} - \delta \quad \text{für } [i, j] \text{ rot} \\ B &:= B \cup \{(r, s)\} \setminus \{(i_0, j_0)\} \end{aligned}$$

und gehe zu 2.

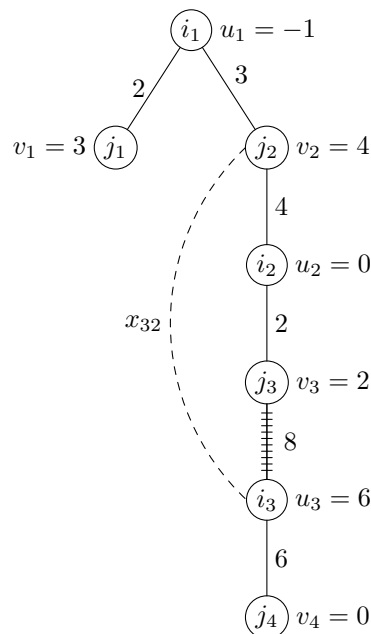


Abbildung 21.3: Ausgangslösung mit gegebene Kostenkoeffizienten. Rekursive Berechnung von u_i and v_j . x_{32} wird als blaue Kante neu in die Basis aufgenommen, die Kante (i_3, j_3) gestrichen.

Beispiel 21.2 (Fortsetzung)

Gegeben sei das folgende Transportproblem mit der Kostenmatrix

$$c_{ij} : \begin{array}{|c|c|c|c|} \hline 2 & 3 & 4 & 1 \\ \hline 5 & 4 & 2 & 3 \\ \hline 4 & 2 & 8 & 6 \\ \hline \end{array}$$

Wir beginnen mit der durch die Nordwesteckenregel bestimmten Ausgangslösung

$$X : \begin{array}{|c|c|c|} \hline 2 & 2 & \\ \hline & 2 & 5 \\ \hline & & 1 & 3 \\ \hline \end{array}$$

Die Ausgangslösung entspricht dem Baum in Abbildung 21.3. Dieser Baum trägt auf den Kanten die entsprechenden Kosten c_{ij} . Nun kann man, beginnend mit $v_4 = 0$ die rekursive Berechnung von u_1, u_2, u_3 und v_1, v_2, v_3 vornehmen. Aus $u_i + v_j = \bar{c}_{ij}$ erhält man die Vergleichsdaten für die reduzierten Kostenkoeffizienten

$$\bar{C} : \begin{array}{c|ccc|c} & 3 & 4 & 2 & 0 & v_j \\ \hline -1 & * & * & 1 & -1 & \\ \hline 0 & 3 & * & * & 0 & \\ \hline 6 & 9 & \mathbf{10} & * & * & \\ \hline u_i & & & & & \end{array}$$

Wir nehmen x_{32} neu in die Basis auf, d.h. zum Baum wird die Kante (i_3, j_2) als blaue Kante hinzugefügt. Dadurch entsteht ein Kreis (vgl. Satz 21.1, Punkt 6). Dieser Kreis ist gerade, seine

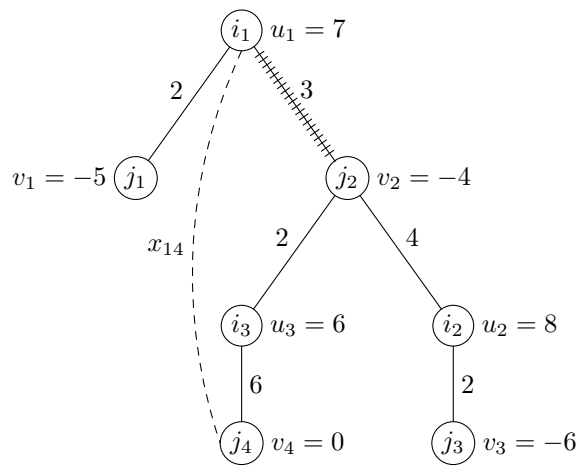


Abbildung 21.4: Zweiter Basisaustausch Schritt. x_{14} wird neu in die Basis aufgenommen. x_{12} verläßt diese Basis.

Kanten können daher blau und rot gefärbt werden. Wie man in Abb. 21.3 erkennt, sind x_{22} und x_{33} rote Kanten des Kreises und x_{32}, x_{23} blaue Kanten des Kreises. Also ist

$$\delta := \min\{x_{22}, x_{33}\} = \min\{1, 2\} = 1.$$

und dieses δ wird auf der Kante $[i_3, j_3]$ angenommen. Damit wird x_{33} Nichtbasisvariable und die neue Basislösung hat als neue Werte

$$\begin{aligned} x_{22} &= 1 \\ x_{32} &= 1 \\ x_{23} &= 6 \end{aligned}$$

Die neue Basislösung

$$X : \begin{array}{|c|cc} \hline 2 & 2 & \\ \hline & 1 & 6 \\ & 1 & 3 \\ \hline \end{array}$$

ist in Abbildung 21.4 dargestellt. Man berechnet wieder u_1, u_2, u_3 und v_1, v_2, v_3 und erhält

$$\bar{C} : \begin{array}{c|ccc} & -5 & -4 & -6 & 0 \\ \hline 7 & * & * & 1 & \mathbf{7} \\ 8 & 3 & * & * & 8 \\ 6 & 1 & * & 0 & * \end{array}$$

Da die Lösung nicht optimal ist, fügt man die Kante (i_1, j_4) als blaue Kante zum Baum hinzu. Die Basisvariablen auf den roten Kanten sind x_{12} und x_{34} . Also ist

$$\delta := \min\{x_{12}, x_{34}\} = \min\{2, 3\} = 2$$

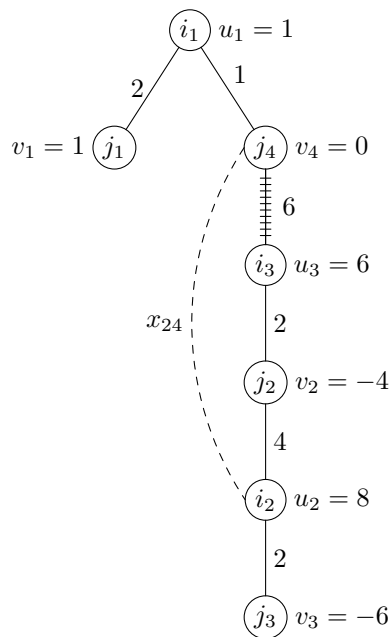


Abbildung 21.5: Dritter Basisaustausch. x_{24} wird neu in die Basis aufgenommen. x_{34} verläßt die Basis.

und δ wird von x_{12} angenommen. Damit verläßt x_{12} die Basis. Man erhält

$$\begin{aligned} x_{34} &= 1 \\ x_{14} &= 2 \\ x_{32} &= 3 \end{aligned}$$

Damit lautet die neue Basislösung (siehe Abbildung 21.5)

$$X : \begin{array}{|c|cc|} \hline 2 & & 2 \\ & 1 & 6 \\ \hline 3 & & 1 \\ \hline \end{array}$$

Wie in den vorhergegangenen Schritten erhält man

$$\bar{C} : \begin{array}{c|ccc|c} & 1 & -4 & -6 & 0 \\ \hline 1 & * & -3 & -5 & * \\ 8 & 9 & * & * & \mathbf{8} \\ 6 & 7 & * & 0 & * \end{array}$$

Damit wird x_{24} neu in die Basis aufgenommen. Die Basisvariablen auf den roten Kanten sind x_{22} und x_{34} . Also ist

$$\delta = \min\{x_{22}, x_{34}\} = 1$$

und x_{34} verläßt die Basis. Die Variable x_{22} bleibt mit dem Wert 0 in der Basis (entartete Basislösung!). Somit erhält man (siehe Abbildung 21.6)

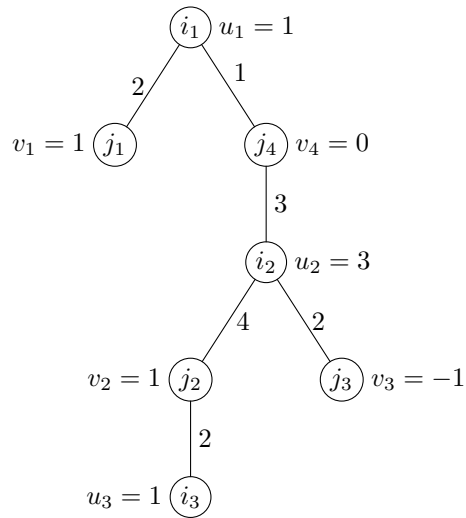


Abbildung 21.6: Optimale Basis

$$X : \begin{array}{|c|c|c|} \hline 2 & & 2 \\ \hline & 0 & 6 & 1 \\ \hline & & 4 & \\ \hline \end{array}$$

Man berechnet die Vergleichsgrößen \bar{c}_{ij} :

$$\bar{C} : \begin{array}{c|cccc} & 1 & 1 & -1 & 0 & v_j \\ \hline 1 & * & 2 & 0 & * & \\ \hline 3 & 4 & * & * & * & \\ \hline 1 & 2 & * & 0 & 1 & \\ \hline u_i & & & & & \end{array}$$

Nun gilt $c_{ij} \geq \bar{c}_{ij}$ für alle $(i, j) \in N$. Also ist die zugehörige Basislösung optimal. Der Zielfunktionswert ist $z = 29$.

Ausgehend vom Tableau, das durch die Regel der geringsten Kosten bestimmt wurde, hätte es lediglich einer Transformation bedurft, um zu einer Optimallösung zu gelangen.

Wie Zadeh [54] durch ein Beispiel zeigte, ist der vorgestellte Transportalgorithmus - wie auch das Simplexverfahren - kein polynomiales Verfahren. Es gibt jedoch streng polynomiale Verfahren für Transportprobleme, deren Rechenaufwand nur polynomial von m und n abhängt, siehe etwa [47]. Bei größeren Problemen der Praxis wird das Transportproblem meist als minimales Kostenflußproblem gelöst. Dazu führt man eine Quelle s und eine Senke t ein. Die Quelle wird mit allen Knoten der Erzeuger durch eine gerichtete Kante mit der Kapazität a_i verbunden. Die Abnehmer j werden alle durch eine gerichtete Kante mit der Mindestkapazität b_j mit der Senke verbunden. Ferner werden alle Kanten von den Erzeugern i zu den Abnehmern j als gerichtete Kanten betrachtet. In diesem Modell lassen sich auch leicht verbotene Transportbeziehungen betrachten: Kann vom Hersteller i nicht an den Abnehmer j geliefert werden, streiche man die Kante von i nach j . Ebenso können leicht Transportbeschränkungen der Form

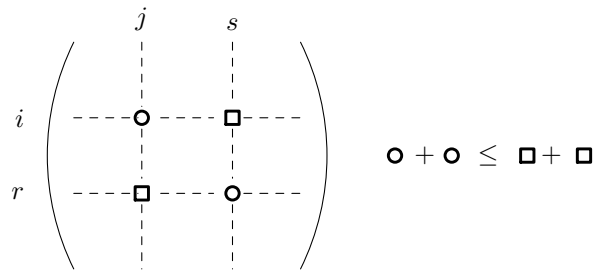


Abbildung 21.7: Monge Eigenschaft

$x_{ij} \leq s_{ij}$ ($1 \leq i \leq m$, $1 \leq j \leq n$) mitberücksichtigt werden. Dazu versieht man die Kante (i, j) mit einer oberen Kapazität s_{ij} . Nun wird im so definierten Netzwerk ein maximaler Fluß mit minimalen Kosten bestimmt. Dieser entspricht dem gesuchten Transportplan mit minimalen Kosten. Eine Reihe von Verfahren zur Bestimmung von maximalen Flüssen mit minimalen Kosten findet man ausführlich in der Monographie von Ahuja, Magnanti und Orlin [2] dargestellt. Die Grundideen zur Bestimmung von maximalen Flüssen werden im nächsten Kapitel besprochen.

Es gibt einen interessanten Spezialfall von Transportproblemen, der optimal durch die Nordwesteckenregel gelöst wird. Dazu definieren wir Monge-Matrizen.

Definition 21.1 Eine $(m \times n)$ Matrix $C = (c_{ij})$ heißt Monge Matrix, wenn für alle $1 \leq i < r \leq m$ und alle $1 \leq j < s \leq n$ gilt:

$$c_{ij} + c_{rs} \leq c_{is} + c_{rj}. \quad (21.7)$$

Diese Monge Eigenschaft ist in Abb. 21.7 graphisch dargestellt. Ist, zum Beispiel $0 \leq p_1 \leq p_2 \leq \dots \leq p_m$ und $q_1 \geq q_2 \geq \dots \geq q_n \geq 0$, und definiert man $c_{ij} := p_i q_j$, so ist die Matrix $C = (c_{ij})$ eine Monge-Matrix. Nun gilt

Satz 21.5 Ist die Kostenmatrix eines Transportproblems eine Monge Matrix, so liefert die Nordwesteckenregel für beliebige nichtnegative Angebot und Nachfrage Vektoren (a_1, \dots, a_m) und (b_1, b_2, \dots, b_n) mit

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

stets eine Optimallösung des Transportproblems.

Beweis: Wir schreiben die Lösung eines Transportproblem als Vektor $x = (x_{11}, x_{12}, \dots, x_{1n}, x_{21}, x_{22}, \dots, x_{2n}, \dots, x_{mn})$. Sei nun x die durch die Nordwesteckenregel erzeugte Lösung und sei y jene Optimallösung des Transportproblem, für die der Vektor $z = (z_{ij}) = (|x_{ij} - y_{ij}|)$ lexikographisch minimal ist. Wir haben zu zeigen, dass $z = 0$ gilt.

Nehmen wir das Gegenteil an und sei z_{ip} die erste Komponente $\neq 0$. Da x durch die Nordwesteckenregel bestimmt wurde, muss $x_{ip} > y_{ip}$ gelten. Aufgrund der Restriktionen (21.1) des Transportproblem gibt es ein $q > p$ mit $y_{iq} > 0$ und ein $j > i$ mit $y_{jp} > 0$. Wir definieren

$\delta := \min(y_{iq}, y_{jp}, x_{ip} - y_{ip}) > 0$ und definieren eine neue Lösung \bar{y} durch

$$\bar{y}_{rs} := \begin{cases} y_{rs} - \delta, & \text{falls } (r, s) = (i, q), (j, p), \\ y_{rs} + \delta, & \text{falls } (r, s) = (i, p), (j, q), \\ y_{rs}, & \text{falls sonst.} \end{cases}$$

Aufgrund der Konstruktion ist \bar{y} wieder eine zulässige Lösung und aufgrund der Monge Bedingung (21.7) gilt $c := c_{iq} + c_{jp} - (c_{ip} + c_{jq}) \geq 0$. Daher ist

$$\sum_{i,j} c_{ij} y_{ij} = c \cdot \delta + \sum_{i,j} c_{ij} \bar{y}_{ij}.$$

Da y optimal war, muß $c = 0$ gelten und damit ist \bar{y} ebenfalls eine Optimallösung. Nun ist aber $(|x_{ij} - \bar{y}_{ij}|)$ lexikographisch kleiner als z . Dies ist ein Widerspruch zur Wahl von y . Daher ist $z = 0$. ■

Die Monge Eigenschaft von Matrizen hängt von der Indizierung der Zeilen und Spalten ab, die natürlich bei Transportproblemen willkürlich ist. Daher stellt sich die Frage: gegeben sei eine beliebige Kostenmatrix C . Kann man deren Zeilen und Spalten so permutieren, dass die permutierte Matrix eine Monge Matrix ist? Eine Matrix mit dieser Eigenschaft heißt *permutierte Monge Matrix*. Sind etwa (p_1, \dots, p_m) und (q_1, \dots, q_n) beliebige nichtnegative Vektoren und definiert man eine Matrix $C = (c_{ij})$ durch $c_{ij} := p_i q_j$, so ist C eine permutierte Monge Matrix. Deĭneko und Filonenko entwickelten 1979 ein Verfahren, siehe [9], p.135, das in $O(mn + m \log m + n \log n)$ Schritten feststellt, ob C eine permutierte Monge Matrix ist und gegebenenfalls die entsprechenden Zeilen- und Spaltenpermutation liefert.

Monge Matrizen spielen eine interessante Rolle in der Optimierung. Optimierungsprobleme, deren Eingabedaten eine Monge Eigenschaft erfüllen, sind oft in sehr einfacher Weise lösbar, siehe Burkard, Klinz und Rudolf [9].

Kapitel 22

Maximale Flüsse in Netzwerken

Flüsse in Netzwerken spielen in der Praxis eine große Rolle, insbesondere das Problem einen Fluss mit maximalem Flusswert zu bestimmen, sowie das Problem minimale Kostenflüsse zu berechnen. Beide Probleme lassen sich als spezielle lineare Programme formulieren. Aufgrund der speziellen Gestalt der Restriktionen ist es jedoch wieder wie bei Transportproblemen möglich, eine dem Problem angepasste Lösungsmethode zu entwickeln.

Ein Netzwerk $\mathcal{N} = (N, A, c)$ besteht aus der Knotenmenge N , der Menge A von gerichteten Kanten (i, j) mit $i, j \in N$ und Kantenkapazitäten $c(i, j) \geq 0$. Oft werden auch Kapazitäten $c(i, j) = \infty$ zugelassen. Diese besagen, dass auf der Kante ein beliebig großer Fluss möglich ist. Zwei der Knoten des Netzwerkes sind ausgezeichnet, die *Quelle* s und die *Senke* t . Ohne Beschränkung der Allgemeinheit können wir annehmen, dass es keine Kanten gibt, die in die Quelle einmünden, und dass es keine Kanten gibt, die die Senke verlassen. Ein *Fluss* von s nach t ordnet jeder Kante eine nichtnegative Zahl, den *Flusswert* $f(i, j)$ auf dieser Kante zu, wobei die folgenden Restriktionen erfüllt sein müssen:

$$\sum_{i \in N; (i, j) \in A} f(i, j) = \sum_{k \in N; (j, k) \in A} f(j, k), \quad \text{für alle } j \in N \setminus \{s, t\} \quad (22.1)$$

$$0 \leq f(i, j) \leq c(i, j), \quad \text{für alle } (i, j) \in A. \quad (22.2)$$

Die Restriktionen (22.1) heißen *Flusserhaltungsgleichungen*. Sie besagen, dass in jedem Knoten $\neq s, t$ die Summe der ankommenden Flüsse gleich der Summe der abgehenden Flüsse ist. Die Restriktionen (22.2) heißen *Kapazitätsrestriktionen*. Sie besagen, dass auf jeder Kante der Flusswert größer oder gleich 0 und kleiner oder gleich der Kantenkapazität sein muss.

Der Wert $z(f)$ eines Flusses ist definiert als

$$z(f) := \sum_{i \in N; (s, i) \in A} f(s, i). \quad (22.3)$$

Aufgrund der Flusserhaltungsgleichungen gilt

$$z(f) = \sum_{i \in N; (i, t) \in A} f(i, t).$$

Beim maximalen Flussproblem wird ein *maximaler Fluss*, d.h. ein Fluss mit möglichst großem Wert $z(f)$ gesucht.

Das maximale Flussproblem lässt sich somit in folgender Weise als lineares Programm formulieren:

$$\text{Maximiere } \sum_{(s,i) \in A} f(s,i)$$

unter den Nebenbedingungen

$$\begin{aligned} \sum_{(i,j) \in A} f(i,j) - \sum_{(j,k) \in A} f(j,k) &= 0 && \text{für alle } j \neq s, t \\ 0 \leq f(i,j) &\leq c(i,j) && \text{für alle } (i,j) \in A. \end{aligned}$$

Daraus erkennt man, dass die Koeffizientenmatrix der ersten Restriktionengruppe die Knoten-Kanten Inzidenzmatrix eines gerichteten Graphen und damit unimodular ist. Da die Koeffizientenmatrix der zweiten Restriktionengruppe die Einheitsmatrix ist, ist also die gesamte Koeffizientenmatrix vollständig unimodular. Daher gibt es bei ganzzahligen Kapazitäten einen maximalen Fluss, dessen Werte auf den Kanten ebenfalls ganzzahlig sind.

Sind auf den Kanten (i,j) zusätzlich Kantenkosten $a(i,j)$ gegeben, so kann man fragen, welcher maximale Fluss hat die geringsten Kosten. Dieses Problem wird als *minimales Kostenflussproblem* bezeichnet. Mit dem minimalen Kostenflussproblem werden wir uns in Abschnitt ?? befassen.

Im folgenden werden wir folgende Notation benützen. Ist $h : A \rightarrow \mathbb{R}$ eine auf den Kanten des Netzwerkes definierte Funktion, dann ist

$$h(X, Y) = \sum_{(i,j) \in A; i \in X, j \in Y} h(i,j).$$

Für einelementige Mengen $X = \{i\}$ schreiben wir $h(i, Y)$ anstelle von $h(\{i\}, Y)$.

Mit Flussproblemen hängen eng Schnitte in Netzwerken zusammen. Ein (s, t) -Schnitt im Netzwerk \mathcal{N} ist eine Partition (X, \bar{X}) der Knotenmenge N mit $s \in X$ und $t \in \bar{X}$. Die Kanten $(i, j) \in A$ mit $i \in X$ und $j \in \bar{X}$ werden als *Kanten im Schnitt* bezeichnet. Eine unmittelbare Folgerung aus der Definition eines Schnittes ist:

Jeder gerichtete Weg von der Quelle s zur Senke t enthält mindestens eine Kante im Schnitt.

Der Wert $c(X, \bar{X})$ des Schnittes (X, \bar{X}) ist die Summe aller Kantenkapazitäten der Kanten im Schnitt:

$$c(X, \bar{X}) := \sum_{(i,j) \in A; i \in X, j \in \bar{X}} c(i,j). \quad (22.4)$$

Dem schwachen Dualitätssatz der linearen Optimierung entspricht das folgende Lemma:

Lemma 22.1 *Der Wert $z(f)$ eines beliebigen Flusses f ist stets durch den Wert $c(X, \bar{X})$ eines beliebigen Schnittes nach oben beschränkt.*

Beweis: Sei f ein beliebiger Fluss und (X, \bar{X}) ein beliebiger Schnitt. Da jeder gerichtete Weg von s nach t eine Kante im Schnitt enthält, ist das Lemma trivial, falls $z(f) = \infty$ gilt. Daher können wir annehmen, dass jeder Flusswert $f(i, j)$, $(i, j) \in A$ endlich ist. Aufgrund der Flusserhaltungsgleichungen gilt für jedes $i \in X$, $i \neq s$

$$f(i, N) = f(N, i)$$

und damit für $X' := \{i \mid i \in X, i \neq s\}$: $f(X', N) = f(N, X') = f(N, X)$, da keine Kante in die Quelle hineinführt. Somit ist

$$f(X, N) = f(s, N) + \sum_{\substack{v \in X \\ v \neq s}} f(v, N) = z(f) + \sum_{\substack{v \in X \\ v \neq s}} f(N, v) = z(f) + f(N, X)$$

Da $f(N, X) = f(X, X) + f(\bar{X}, X)$ und $f(X, N) = f(X, X) + f(X, \bar{X})$ ist, erhält man durch Kürzen von $f(X, X)$:

$$z(f) = f(X, \bar{X}) - f(\bar{X}, X). \quad (22.5)$$

Da $f(i, j) \geq 0$ und $f(X, Y) \leq c(X, Y)$ gilt, erhält man

$$z(f) \leq c(X, \bar{X}). \quad (22.6)$$

■

Grundlegend zur Bestimmung eines maximalen Flusses ist der Satz von Ford und Fulkerson. Er ist eine Spezialisierung des Starken Dualitätssatzes der linearen Optimierung auf die spezielle Fragestellung und besagt, dass der Wert eines maximalen Flusses gleich dem minimalen Wert eines Schnittes ist.

Satz 22.2 *Max Fluss-min Schnitt Satz*

Der Wert eines maximalen Flusses ist gleich dem minimalen Wert eines Schnittes

$$\max_f z(f) = \min_{(X, \bar{X})} c(X, \bar{X}) \quad (22.7)$$

Beweis: Falls $z(f) = \infty$ ist, ist der Satz trivial. Daher nehmen wir an, dass $z(f)$ endlich ist. Nach Lemma 22.1 ist nur zu zeigen, dass es einen Fluss f und einen Schnitt $S = (X, \bar{X})$ gibt mit $z(f) = c(X, \bar{X})$. Es sei nun f ein maximaler Fluss. Wir definieren eine Menge X durch die folgenden drei Bedingungen:

1. $s \in X$
2. Falls $i \in X$ und $f(i, j) < c(i, j)$, dann gilt $j \in X$. (Die Kante (i, j) wird *Vorwärtskante* genannt).
3. Falls $j \in X$ und $f(i, j) > 0$, dann sei $i \in X$. (Die Kante (i, j) wird *Rückwärtskante* genannt).

Zunächst beweisen wir, dass (X, \bar{X}) tatsächlich ein Schnitt ist, d.h. dass $t \in \bar{X}$. Nehmen wir das Gegenteil $t \in X$ an. In diesem Fall gibt es einen Weg P von s nach t , der nur über Knoten in X läuft: $P = (s = i_0, i_1, i_2, \dots, i_k = t)$, wobei (i_k, i_{k+1}) entweder eine Vorwärtskante in A ist oder (i_{k+1}, i_k) eine Rückwärtskante in A ist. Man nennt P einen *augmentierenden Weg*. Sei F die Menge aller Vorwärtskanten in P und B die Menge aller Rückwärtskanten des Weges P . Wir definieren nun

$$\epsilon_1 := \min_{(i,j) \in F} c(i, j) - f(i, j)$$

und

$$\epsilon_2 := \min_{(i,j) \in B} f(i, j).$$

Sei $\epsilon := \min(\epsilon_1, \epsilon_2)$. Aufgrund der obigen Bedingungen 2 und 3 gilt $\epsilon > 0$.

Wir modifizieren den Fluss f , indem wir auf allen Vorwärtskanten (i, j) den Flusswert $f(i, j)$ um ϵ erhöhen und auf allen Rückwärtskanten den Flusswert um ϵ verringern. Man kann leicht nachprüfen, siehe Abb. 22.1, dass der so modifizierte Fluss nach wie vor alle Flusserhaltungsgleichungen und Kapazitätsrestriktionen erfüllt. Er hat aber nun einen Wert $z(f) + \epsilon$. Dies ist ein Widerspruch zur Maximalität von f . Daher kann nicht $t \in X$ gelten und somit ist (X, \bar{X}) tatsächlich ein Schnitt.

Aus der Definition der Menge X folgt:

- Für eine Kante (i, j) mit $i \in X$, $j \in \bar{X}$ gilt $f(i, j) = c(i, j)$.
- Für eine Kante (i, j) mit $i \in \bar{X}$, $j \in X$ gilt $f(i, j) = 0$.

Daher ist $f(X, \bar{X}) = c(X, \bar{X})$ und $f(\bar{X}, X) = 0$. Somit folgt aus Gleichung (22.5) die gewünschte Beziehung $z(f) = c(X, \bar{X})$. ■

Der Beweis zeigt auch einen Weg auf, wie ein maximaler Fluss bestimmt werden kann. Dazu beginnt man mit einem beliebigen zulässigen Fluss f , zum Beispiel mit $f(i, j) = 0$ für alle $(i, j) \in A$. Dann sucht man einen augmentierenden Weg bezüglich f und vergrößert den Fluss entlang des augmentierenden Weges. Ist ein Fluss f gegeben, so enthält das *Inkrementnetzwerk* $\mathcal{N}_f = (N, A_f, c_f)$ alle augmentierenden Wege bezüglich f . Dabei enthält die Menge A_f die Vorwärtskanten $\{(i, j) \in A \mid f(i, j) < c(i, j)\}$ und die Rückwärtskanten $\{(i, j) \mid f(j, i) > 0\}$ bezüglich f . Die Kapazitäten der Vorwärtskanten sind festgelegt durch $c_f(i, j) := c(i, j) - f(i, j)$, die Kapazitäten der Rückwärtskanten sind definiert als $c_f(i, j) := f(j, i)$.

Ford-Fulkerson's Max Fluss-Min Schnitt Satz besagt, dass f ein maximaler Fluss ist, wenn jeder Weg von der Quelle zur Senke eine Kante mit $f(i, j) = c(i, j)$ enthält. Dies bedeutet, übersetzt auf das Inkrementnetzwerk:

Lemma 22.3 *Gibt es im Inkrementnetzwerk \mathcal{N}_f keinen gerichteten Weg von s nach t , dann ist der Fluss f maximal.*

Sei nun f nicht maximal. Einen Fluss von der Quelle s zur Senke t im Inkrementnetzwerk bezeichnen wir als Inkrementfluss Δf . Wir definieren

$$f \oplus \Delta f(u, v) := \begin{cases} f(u, v) + \Delta f(u, v), & \text{falls } (u, v) \text{ eine Vorwärtskante ist,} \\ f(u, v) - \Delta f(v, u), & \text{falls } (v, u) \text{ eine Rückwärtskante ist.} \end{cases} \quad (22.8)$$

Dann gilt

Lemma 22.4 1. *Ist Δf ein Fluss in \mathcal{N}_f , dann ist $f \oplus \Delta f$ ein Fluss in \mathcal{N} .*

2. *Für je zwei beliebige Flüsse f und g in \mathcal{N} mit Flusswerten $z(g) > z(f)$ gibt es einen Inkrementfluss Δf in \mathcal{N}_f so dass $g = f \oplus \Delta f$.*

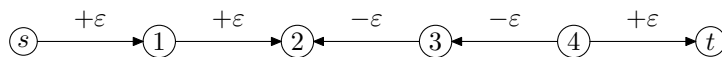


Abbildung 22.1: Flusserhaltungsgleichung entlang eines augmentierenden Weges

Beweis:

1. Aufgrund der Definition der Kapazitäten im Inkrementnetzwerk erfüllt der Fluss $f \oplus \Delta f$ die Kapazitätsrestriktionen im Netzwerk \mathcal{N} . Außerdem erfüllen sowohl f wie auch der Inkrementfluss Δf die Flusserhaltungsgleichungen in allen Knoten $\neq s, t$. Daher erfüllt auch $f \oplus \Delta f$ die Flusserhaltungsgleichungen im Netzwerk \mathcal{N} .
2. Wie man leicht nachprüft, ist die Funktion Δf definiert durch

$$\Delta f(u, v) := \begin{cases} g(u, v) - f(u, v), & \text{falls } g(u, v) \geq f(u, v), \\ f(v, u) - g(v, u), & \text{falls } g(v, u) < f(v, u) \end{cases} \quad (22.9)$$

ein Inkrementfluss und erfüllt $f \oplus \Delta f = g$.

22.1 Ein polynomialer max-Fluß Algorithmus

Kapitel 23

Rucksackprobleme

23.1 Das Rucksackproblem

Zu den einfachsten ganzzahligen linearen Optimierungsproblemen gehört das Rucksackproblem. Dabei handelt es sich um eine Optimierungsaufgabe mit einer linearen Zielfunktion und einer einzigen linearen Restriktion, wobei alle Koeffizienten positiv sind und die Variablen nur ganzzahlige Werte annehmen können:

$$\begin{array}{ll} \max & c_1x_1 + c_2x_2 + \dots + c_nx_n \\ \text{unter} & \\ & a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b \\ & x_1, x_2, \dots, x_n \geq 0, \text{ ganzzahlig.} \end{array}$$

Der Name dieses Problem es rührt von folgender Interpretation her: Ein Wanderer möchte n verschiedene Gegenstände in einem Rucksack mitnehmen. Die Gegenstände haben für ihn den Wert c_j , $j = 1, 2, \dots, n$, und das Gewicht a_j . Die Variable x_j gibt die Anzahl des j -ten Gegenstandes an, die in den Rucksack eingepackt wird. Ziel ist es, möglichst wichtige Gegenstände mitzunehmen, so dass aber das Gesamtgewicht des Rucksacks das Gewicht b nicht überschreitet.

Rucksackprobleme treten in vielen Anwendungen auf, etwa bei der Zusammenstellung eines Wertpapier Portfolios, aber auch bei Verschnittproblemen (Gilmore und Gomory [21]) und in der Optimierung bei sogenannten Spaltengenerierungsverfahren. Es gibt zwei unterschiedliche Problemstellungen. Im 0 – 1-Rucksackproblem treten nur binäre Entscheidungsvariable auf, die besagen, ob der j -te Gegenstand mitgenommen wird oder nicht. Im ganzzahligen Rucksackproblem können die ganzzahligen Variablen x_j auch Werte größer als 1 annehmen und beschreiben damit wie oft der j -te Gegenstand mitgenommen wird. Eine ausführliche Darstellung von Rucksackproblemen findet sich in der Monographie von Kellerer, Pferschy und Pisinger [31].

Im folgenden betrachten wir das binäre Rucksackproblem

$$\begin{array}{ll} \max & c_1x_1 + c_2x_2 + \dots + c_nx_n \\ \text{unter} & \\ & a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b \\ & x_1, x_2, \dots, x_n \in \{0, 1\} \end{array} \tag{23.1}$$

Alle Ergebnisse für das binäre Rucksackproblem lassen sich leicht auf das allgemeine ganzzahlige Rucksackproblem übertragen. Zunächst untersuchen wir die lineare Relaxation von (23.1), d.h.

das lineare Programm

$$\begin{aligned} & \max && c_1x_1 + c_2x_2 + \dots + c_nx_n \\ & \text{unter} && \\ & && a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b \\ & && 0 \leq x_j \leq 1 \quad \text{für } j = 1, 2, \dots, n. \end{aligned} \tag{23.2}$$

Setzt man

$$y_j := a_jx_j \quad \text{für } j = 1, 2, \dots, n$$

dann geht (23.2) über in

$$\begin{aligned} & \max && \frac{c_1}{a_1}y_1 + \frac{c_2}{a_2}y_2 + \dots + \frac{c_n}{a_n}y_n \\ & \text{unter} && \\ & && y_1 + y_2 + \dots + y_n \leq b \\ & && 0 \leq y_j \leq a_j \quad \text{für } j = 1, 2, \dots, n. \end{aligned} \tag{23.3}$$

Ein einfaches Austausch-Argument zeigt, dass man eine Optimallösung des linearen Optimierungsproblems (23.3) erhält, indem man die Quotienten der Kosten zu den Gewichten

$$\frac{c_j}{a_j}$$

absteigend ordnet und dann den Rucksack beginnend mit dem ersten Gegenstand der Reihe nach auffüllt (Greedy-Algorithmus). Gilt

$$\frac{c_1}{a_1} \geq \frac{c_2}{a_2} \geq \dots \geq \frac{c_n}{a_n},$$

so kann man beginnend mit y_1 die y -Werte auf den größtmöglichen Wert setzten. D.h. es gibt einen Index k so dass

$$y_j = a_j \quad \text{für } 1 \leq j \leq k-1 \quad \text{und } 0 < y_k \leq a_k \quad \text{und } y_j = 0 \quad \text{für } j = k+1, \dots, n.$$

Beispiel 23.1 *Betrachten wir die lineare Relaxation des Rucksackproblems*

$$\begin{aligned} & \max && x_1 + 3x_2 + 2x_3 + 5x_4 + 4x_5 \\ & \text{unter} && \\ & && 6x_1 + 7x_2 + 4x_3 + 2x_4 + 9x_5 \leq 19 \\ & && x_1, x_2, \dots, x_n \in \{0, 1\} \end{aligned}$$

Es gilt

$$\frac{5}{2} \geq \frac{2}{4} \geq \frac{4}{9} \geq \frac{3}{7} \geq \frac{1}{6}.$$

Daher wählen wir für die Variablen die Reihenfolge x_4, x_3, x_5, x_2, x_1 . Damit wird $x_4 = x_3 = x_5 = 1$ und $x_2 = 4/7$. Ferner ist $x_1 = 0$. Diese Lösung ergibt den Zielfunktionswert $12 \frac{5}{7}$. Daher ist der Wert der ganzzahligen Lösung sicher ≤ 12 .

Balas und Zemel [3] gaben mit Hilfe von Medianbestimmungen eine Methode an, die das relaxierte Rucksackproblem in linearer Zeit $O(n)$ löst, siehe auch Abschnitt 3.1 in der Monographie von Kellerer, Pferschy und Pisinger [31].

Das Greedyverfahren liefert nicht immer eine gute Näherungslösung für das binäre Rucksackproblem wie das folgende Beispiel mit nur 2 Gegenständen zeigt.

Beispiel 23.2 Wir betrachten ein Rucksackproblem mit 2 Gegenständen und $b = t$. Gegenstand 1 hat einen Wert 3 und das Gewicht 2, Gegenstand 2 hat den Wert $t-1$ und das Gewicht $t-1$. Offenbar gilt:

$$\frac{3}{2} > \frac{t-1}{t-1}.$$

Daher packt der Greedyalgorithmus den 1. Gegenstand ein und liefert damit für den Rucksack den Wert 3. Besser wäre es jedoch, den zweiten Gegenstand einzupacken, denn dann hätte der Rucksack den Wert $t-1$. Somit erhält man für das Verhältnis des optimalen Zielfunktionswertes z_{opt} zum Zielfunktionswert z_{greedy} der Lösung, die durch den Greedyalgorithmus gefunden wurde

$$\frac{z_{\text{opt}}}{z_{\text{greedy}}} = \frac{t-1}{3} \rightarrow \infty. \quad (23.4)$$

Dies zeigt, dass die Greedylösung beliebig schlecht sein kann.

23.2 Ein dynamisches Optimierungsverfahren für das binäre Rucksackproblem

Eine Lösungsmethode für das binäre Rucksackproblem bietet die sogenannten dynamischen Optimierung, die in zwei Phasen abläuft. In der ersten Phase wird rekursiv der optimale Zielfunktionswert berechnet, in der zweiten Phase wird dann durch Backtracking die zugehörige Lösung bestimmt. Das Verfahren beruht auf einer Rekursion für die Rucksackfunktion

$$z(k, d) := \max \left\{ \sum_{j=1}^k c_j x_j \mid \sum_{j=1}^k a_j x_j \leq d, x_j \in \{0, 1\} \text{ für } j = 1, 2, \dots, k \right\}. \quad (23.5)$$

Als Anfangsbedingungen setzen wir fest:

$$\begin{aligned} z(0, d) &= 0, \\ z(k, d) &= -\infty \text{ für } d < 0 \\ z(k, d) &= 0 \text{ für } d = 0. \end{aligned}$$

Dann gilt

Satz 23.1 Die Rucksackfunktion erfüllt die Rekursion

$$z(k, d) = \max(z(k-1, d), c_k + z(k-1, d - a_k)) \text{ für } k = 0, 1, 2, \dots \text{ und alle } d \quad (23.6)$$

Beweis: Für die Variable x_k gibt es nur 2 Möglichkeiten. Entweder ist $x_k = 0$, dann ist $z(k, d) = z(k-1, d)$. Oder es ist $x_k = 1$, dann ist $z(k, d) = c_k + z(k-1, d - a_k)$. Daraus folgt unmittelbar (23.6).

Sind die Werte $z(k-1, d)$ bereits gegeben, dann muss nur eine konstante Anzahl von Rechenoperationen ausgeführt werden um $z(k, d)$ zu berechnen. Dies bedeutet, dass der optimale Zielfunktionswert $z(n, b)$ in $O(nb)$ Schritten berechnet werden kann. Damit ist dieses Verfahren ein *pseudopolynomialer Algorithmus*. (Um polynomial zu sein, dürfte die Konstante b nur als $\log b$ in die Komplexität eingehen, da ja alle Daten binär kodiert werden.)

Wir wissen zwar jetzt, wie der optimale Zielfunktionswert berechnet werden kann, aber wir kennen nicht die zugehörige Lösung. Um die Lösung zu finden, kann man in jedem Schritt abspeichern, ob das Maximum in (23.6) durch den ersten oder den zweiten Term erreicht wird:

$$p(k, d) := \begin{cases} 1 & \text{falls } z(k, d) = z(k-1, d), \\ 2 & \text{anderenfalls.} \end{cases} \quad (23.7)$$

Aus diesen Werten kann man von $p(n, b)$ aus rückwärtsgehend die Optimallösung rekonstruieren. Ist $p(n, b) = 2$, so ist $x_n := 1$ und man betrachtet als nächsten Wert $p(n-1, b-a_n)$. Ist $p(n, b) = 1$, so ist $x_n := 0$ und man betrachtet als nächsten Wert $p(n-1, b)$. Danach bestimmt man $x_{n-1}, x_{n-2}, \dots, x_1$.

Beispiel 23.3 Betrachten wir das binäre Rucksackproblem

$$\begin{aligned} \max \quad & 2x_1 + 3x_2 + x_3 + 4x_4 + 7x_5 \\ \text{unter} \quad & 3x_1 + 2x_2 + x_3 + 4x_4 + 4x_5 \leq 7 \\ & x_1, x_2, \dots, x_n \in \{0, 1\} \end{aligned}$$

Ausgehend von $z(1, 1) := \max(z(0, 1), 2 + z(0, -2)) = 0, \dots, z(1, 3) := \max(z(0, 3), 2 + z(0, 0)) = 2$ erhält man die folgende Tabelle:

k	1	2	3	4	5
$d = 0$	0	0	0	0	0
$d = 1$	0	0	1	1	1
$d = 2$	0	3	3	3	3
$d = 3$	2	3	4	4	4
$d = 4$	2	3	4	4	7
$d = 5$	2	5	5	5	8
$d = 6$	2	5	6	7	10
$d = 7$	2	5	6	8	11

Damit ist der optimale Zielfunktionswert $z(5, 7) = 11$. Nun bestimmen wir die zugehörige Lösung. Ausgehend von $p(5, 7) = 2$ erhält man $x_5 = 1$. Daher betrachtet man als nächsten Wert $p(4, 11 - a_5) = p(4, 4) = 1$. Also ist $x_4 = 0$. Da $p(3, 4) = 2$ ist, ist $x_3 = 1$. Ferner ist $p(2, 4 - a_3) = p(2, 3) = 2$ und daher ist $x_2 = 1$. Dann ist $p(1, 1) = 1$, also $x_1 = 0$. Damit ist die optimale Lösung bestimmt.

Man kann sich das Abspeichern der Werte $p(k, d)$ ersparen, denn es gilt

$$p(k, d) = 2 \text{ genau dann, wenn } z(k, d) - z(k-1, d) > 0 \text{ ist.}$$

Man beachte, dass das beschriebene dynamische Optimierungsverfahren optimale Lösungen für alle Werte $d = 0, 1, \dots, b$ generiert.

23.3 Ein Branch-and-Bound Verfahren für das binäre Rucksackproblem

Eine häufig angewandte Methode zur Lösung \mathcal{NP} -schwerer Optimierungsprobleme ist eine implizite Enumeration aller möglichen Lösungen, bekannt unter dem Namen *Branch-and-Bound Verfahren*. Das Branching bezeichnet das Aufspalten eines Problems in mehrere Teilprobleme (analog zum römischen Leitspruch *divide et impera*), das Bounding ermöglicht das Ausschalten von Teilproblemen aufgrund von Schrankenberechnungen. Ein Branch-and-Bound Verfahren beruht auf den folgenden Überlegungen. Zunächst wird das schwere Optimierungsproblem durch ein relaxiertes Problem ersetzt. Das relaxierte Problem enthält alle zulässigen Lösungen des gegebenen Problems und im allgemeinen noch weitere. Man nennt diesen Schritt *Relaxation*. Die Relaxation des binären Rucksackproblems besteht darin, dass man die binären Variablen x_j , $j = 1, 2, \dots, n$, durch beschränkte, aber stetige Variable ersetzt, für die

$$0 \leq x_j \leq 1 \quad \text{für } j = 1, 2, \dots, n$$

gilt. Dadurch geht das Rucksackproblem in ein einfaches lineares Programm über, das sich durch den Greedyalgorithmus (sogar in linearer Zeit) lösen lässt (siehe Abschnitt 23.1). Ist die Lösung des relaxierten Problems zulässig für das Ausgangsproblem, dann ist man fertig, denn diese Lösung ist dann auch optimal für das Ausgangsproblem. Anderenfalls gibt es in der Lösung des relaxierten Problems genau eine Variable mit einem Wert zwischen 0 und 1. Setzt man diese Variable auf den Wert 0, dann erhält man eine zulässige Lösung für das binäre Rucksackproblem und gleichzeitig eine untere Schranke für den optimalen Zielfunktionswert.

Mittels der nichtganzzahligen Variablen werden zwei neue Teilprobleme generiert: in einem Teilproblem wird die Variable auf den Wert 0 gesetzt, im anderen auf den Wert 1. Damit kommt jede Lösung des binären Rucksackproblems in einem der beiden Teilprobleme vor. Als nächstes wird eines der Teilprobleme aufgrund einer Auswahlregel gewählt und als lineares Programm gelöst. Als Auswahlregel empfiehlt sich die *LIFO (last in first out)-Strategie*: das zuletzt erzeugte, noch nicht weiter untersuchte Teilproblem wird als nächstes gelöst.

In jedem Schritt des Branch-and-Bound Verfahrens haben wir die folgenden Informationen:

- eine obere Schranke \bar{z} für den optimalen Zielfunktionswert des binären Rucksackproblems
- die beste bisher gefundene ganzzahlige Lösung mit dem Zielfunktionswert \underline{z} , der eine untere Schranke für den Wert der binären Optimallösung darstellt.

Gilt $\bar{z} = \underline{z}$, so ist man fertig: die zu \underline{z} gehörende Lösung ist optimal. Gilt für die Optimallösung z_r eines (relaxierten) Teilproblems $z_r \leq \underline{z}$, so kann dieses Teilproblem keine Lösung mit einem besserem Zielfunktionswert enthalten als schon bekannt ist. Daher kann das ganze Teilproblem gestrichen werden und man geht aufgrund der Auswahlregel zum nächsten offenen Teilproblem über. Gilt jedoch für die Optimallösung z_r eines (relaxierten) Teilproblems $z_r > \underline{z}$, so wird man zunächst durch Nullsetzen der nichtganzzahligen Variablen versuchen, eine bessere zulässige Lösung zu erhalten, als bisher bekannt ist. In diesem Fall ersetzt man die gespeicherte Lösung durch die neue Lösung mit höherem Zielfunktionswert. Ferner generiert man zwei neue Teilprobleme, die zur Liste der offenen Teilprobleme hinzugefügt werden. Die Optimallösung des gegebenen Problems ist gefunden, wenn die Liste der offenen Teilprobleme leer ist: in diesem Fall hat man (implizit) alle Lösungen durchsucht.

Beispiel 23.4 Wir lösen nun das binäre Rucksackproblem aus Beispiel 23.1. Es empfiehlt sich anfangs die Variablen nach absteigenden Quotienten

$$\frac{c_j}{a_j}$$

zu ordnen - damit erhöht man die Chance, bereits im Anfang eine gute Lösung zu finden. Wir werden also die Variablen in der Reihenfolge x_4, x_3, x_5, x_2 und x_1 betrachten. Die Lösung des relaxierten Problems ist $x_4 = x_3 = x_5 = 1$, $x_2 = 4/7$ und $x_1 = 0$. Diese Lösung hat den Zielfunktionswert $12 \frac{5}{7}$. Daher ist $\bar{z} = 12$ eine obere Schranke für den optimalen Zielfunktionswert. Setzt man $x_2 = 0$, so erhält man eine erste zulässige Lösung $x_4 = x_3 = x_5 = 1$, $x_2 = x_1 = 0$ mit $\underline{z} = 11$.

Nun generiert man zwei neue Teilprobleme, eines mit $x_2 = 0$ und das andere mit $x_2 = 1$. Nach der LIFO-Regel wird Letzteres als nächstes Teilproblem untersucht. Es lautet

$$\begin{aligned} \max \quad & x_1 + 2x_3 + 5x_4 + 4x_5 + 3 \\ \text{unter} \quad & 6x_1 + 4x_3 + 2x_4 + 9x_5 \leq 12 \\ & 0 \leq x_j \leq 1 \end{aligned}$$

In der Optimallösung dieses Problemes ist $x_4 = x_3 = 1$ und $x_5 = 6/9$ sowie $x_1 = 1$. Setzt man $x_5 = 0$, so erhält man keine bessere zulässige Lösung. Wir generieren zwei neue Teilprobleme, indem wir im ersten $x_5 = 0$ und im zweiten $x_5 = 1$ setzen. Aufgrund der LIFO-Regel wird letzteres zuerst untersucht. Es lautet:

$$\begin{aligned} \max \quad & x_1 + 2x_3 + 5x_4 + 7 \\ \text{unter} \quad & 6x_1 + 4x_3 + 2x_4 \leq 3 \\ & 0 \leq x_j \leq 1. \end{aligned}$$

Als Optimallösung dieses Teilproblems erhalten wir $x_4 = 1$, $x_3 = 1/4$. Setzt man $x_3 = 0$, so erhält man eine neue zulässige Lösung für das gegebene Problem mit $x_2 = x_4 = x_5 = 1$, $x_1 = x_3 = 0$ und $\underline{z} = 12$. Jetzt stimmt aber die untere mit der oberen Schranke überein, das heißt, es kann keine bessere Lösung mehr geben. Damit kann man alle offenen Teilprobleme streichen und hat die Optimallösung des binären Rucksackproblem gefunden.

Der Rechenaufwand eines Branch-and-Bound Verfahrens ist im allgemeinen exponentiell in n , da schlimmstenfalls alle Teilmengen J von $\{1, 2, \dots, n\}$ in Hinblick darauf untersucht werden müssen, ob $x_j = 1$ für $j \in J$ und $x_j = 0$ für $j \notin J$ eine optimale Lösung ist.

Kapitel 24

Konvexe Programme und Kuhn-Tucker Sätze

Unter den nichtlinearen Optimierungsaufgaben spielen konvexe Programme eine wichtige Rolle, da für diese Problemklasse recht effiziente Lösungsverfahren existieren. Wir beginnen mit einer kurzen Diskussion konvexer Funktionen und beweisen im nächsten Abschnitt den Satz von Kuhn und Tucker, der notwendige und hinreichende Bedingungen für die Optimallösungen eines konvexen Programmes angibt.

24.1 Konvexe Funktionen

Es sei C eine konvexe Teilmenge des \mathbb{R}^n . Eine Funktion $f : C \rightarrow \mathbb{R}$ heißt *konvex*, wenn für alle $x, y \in C$ und alle $0 < \lambda < 1$ gilt

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y). \quad (24.1)$$

Gilt in (24.1) die Ungleichung mit " $<$ ", so heißt f *streng konvex*.

Zwischen konvexen Funktionen und konvexen Mengen besteht ein enger Zusammenhang, der durch den Epigraphen der Funktionen vermittelt wird. Der *Epigraph* einer Funktion $f : C \rightarrow \mathbb{R}$ ist definiert durch

$$\text{epi } f := \{(x, z) \mid z \geq f(x), x \in C\}. \quad (24.2)$$

Ist f eine konvexe Funktion, dann ist der Epigraph von f eine konvexe Menge. Man kann auch durch die Konvexität des Epigraphen eine konvexe Funktion definieren. Diese Definition einer konvexen Funktion ist leicht allgemeiner als (24.1).

Konvexe Funktionen besitzen eine Reihe von wichtigen Eigenschaften. So ist jede konvexe Funktion auf ihrem Definitionsbereich stetig. Ferner gilt:

Satz 24.1 (Subgradienten-Ungleichung) Sei $C \subset \mathbb{R}^n$ konvex und $f : C \rightarrow \mathbb{R}$ eine Funktion, deren partielle erste Ableitungen auf C existieren. $f(x)$ ist genau dann konvex, wenn für alle $x, y \in C$ die Subgradientenungleichung

$$f(y) \geq f(x) + (y - x)' \nabla f(x) \quad (24.3)$$

gilt. Im Falle streng konvexer Funktionen gilt in (24.3) für $x \neq y$ " $>$ " und umgekehrt.

Beweis:

1. Es gelte (24.3). Sind $x, y \in C$ mit $z = \lambda x + (1 - \lambda)y$, $0 < \lambda < 1$, so können wir (24.3) auf x und z beziehungsweise auf y und z anwenden und wir erhalten

$$f(x) \geq f(z) + (x - z)' \nabla f(z),$$

$$f(y) \geq f(z) + (y - z)' \nabla f(z).$$

Multipliziert man die erste Ungleichung mit λ und die zweite Ungleichung mit $1 - \lambda$ so erhalten wir

$$\lambda f(x) + (1 - \lambda)f(y) \geq f(z) + [\lambda(x - z) + (1 - \lambda)(y - z)]' \nabla f(z) = f(z).$$

Daher ist f konvex bzw. streng konvex.

2. Sei f eine konvexe Funktion. Wir definieren

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

Ist $x \neq y$, so ist $\phi(\lambda) \geq 0$ für $0 < \lambda < 1$ und $\phi(0) = 0$. Daher gilt $\phi'(0) \geq 0$ und wir erhalten

$$-f(x) + f(y) - (y - x)' \nabla f(x) \geq 0.$$

Daraus folgt unmittelbar die Subgradientenungleichung (24.3).

Ist $F(x)$ streng konvex, so ist $-\phi(\lambda)$ streng konvex als Funktion von λ . Aus $\phi(0) = \phi(1) = 0$ folgt daher $\phi(\frac{1}{2}) > 0$. Setzt man $t = (1 - \lambda)0 + \lambda \cdot \frac{1}{2}$, so erhält man für $0 < t < \frac{1}{2}$

$$\phi(t) > (1 - \lambda)\phi(0) + \lambda \cdot \phi(\frac{1}{2}) = 2\phi(\frac{1}{2})t.$$

Also ist $\phi'(0) > 0$.

■

Besitzt f stetige zweite Ableitungen, so kann man folgenden Satz heranziehen um zu prüfen, ob f konvex ist.

Satz 24.2 $f(x)$ sei auf einer konvexen Menge C des \mathbb{R}^n definiert und besitze dort stetige zweite partielle Ableitungen. Ist

$$H(x) := \left(\frac{\partial^2 f(x)}{\partial x_i \partial x_k} \right) \quad (i, k = 1, 2, \dots, n)$$

für alle $x \in C$ positiv semidefinit (definit), so ist $f(x)$ (streng) konvex auf C .

Der Beweis erfolgt mit Hilfe der Taylorformel: Es sei $x, y \in C$, $x \neq y$, mit $\xi = \lambda x + (1 - \lambda)y$ mit $0 \leq \lambda \leq 1$. Dann gilt

$$F(y) = F(x) + (y - x)' \nabla F(x) + \frac{1}{2}(y - x)' H(\xi)(y - x).$$

Da nach Voraussetzung $H(\xi)$ positiv semidefinit ist, folgt daraus (24.3). \square

Aus Satz 24.2 folgt insbesondere, dass quadratische Funktionen der Form

$$Q(x) = \frac{1}{2}x'Hx + b'x + c$$

mit positiv semidefiniter Matrix H konvexe Funktionen sind.

Konvexe Funktionen besitzen folgende gutartigen Eigenschaften in Bezug auf ihre Minimalpunkte:

Satz 24.3 *Sei f eine konvexe Funktion, definiert auf einer konvexen Teilmenge des \mathbb{R}^n . Dann gilt*

1. *Jedes lokale Minimum von f ist global. Die Menge der Minimalpunkte ist konvex.*
2. *Ist f streng konvex, so gibt es höchstens einen Minimalpunkt.*

Beweis: Ist x ein lokales Minimum von f , aber kein globales Minimum, dann gibt es ein $y \in C$ mit $f(y) < f(x)$. Infolge der Konvexität des Definitionsbereiches C ist die Strecke $[x, y]$ ganz in C . Da f konvex ist, nimmt f auf dieser Strecke ab. Also kann x kein lokales Minimum sein. Die anderen beiden Aussagen folgen unmittelbar aus den Definitionen.

24.2 Kuhn-Tucker Bedingungen

Es seien nun konvexe Funktionen F und f_i ($1 \leq i \leq m$) gegeben. Dann nennt man

$$\min\{F(x) \mid f_i(x) \leq 0 \ (1 \leq i \leq m), \ x \geq 0\} \quad (24.4)$$

eine *konvexe Optimierungsaufgabe*.

Bei einer konvexen Optimierungsaufgabe kann das Minimum in einem beliebigen Punkt der Menge der zulässigen Punkte $M = \{x \mid f_i(x) \leq 0, \ (1 \leq i \leq m), \ x \geq 0\}$ angenommen werden. M ist eine konvexe Menge da

$$f_i(\lambda x + (1 - \lambda)y) \leq \lambda f_i(x) + (1 - \lambda)f_i(y) \leq 0 \quad (1 \leq i \leq m).$$

Wir wollen nun eine Charakterisierung für die Minimalpunkte von konvexen Optimierungsaufgaben finden. Dazu führen wir zunächst die *Lagrangefunktion* $L(x, u)$ zur Optimierungsaufgabe (24.4) ein:

$$L(x, u) = F(x) + \sum_{i=1}^m u_i f_i(x). \quad (24.5)$$

Die Komponenten u_i ($1 \leq i \leq m$) heissen Lagrange'sche Multiplikatoren.

Ein Punkt (x^0, u^0) des \mathbb{R}_+^{n+m} heisst *Sattelpunkt* von $L(x, u)$, wenn gilt

$$L(x^0, u) \leq L(x^0, u^0) \leq L(x, u^0) \text{ für alle } x \geq 0, \ u \geq 0. \quad (24.6)$$

Ohne weitere Voraussetzungen gilt nun

Satz 24.4 Ist (x^0, u^0) ein Sattelpunkt von $L(x, u)$, so ist x^0 eine Minimallösung von

$$\min\{F(x) \mid f_i(x) \leq 0 \ (1 \leq i \leq m), \ x \geq 0\}.$$

Beweis: Fassen wir die Komponenten $f_i(x)$ zu einem Vektor $f(x)$ zusammen. (24.6) besagt, dass für $x \geq 0, u \geq 0$ gilt:

$$F(x^0) + u'f(x^0) \leq F(x^0) + (u^0)'f(x^0) \leq F(x) + (u^0)'f(x).$$

Daher ist $u'f(x^0) \leq (u^0)'f(x^0)$ für alle $u \geq 0$. Das ist nur möglich, wenn $f(x^0) \leq 0$ gilt. Also ist x^0 zulässig. Setzt man nun $u = 0$, so erhält man $(u^0)'f(x^0) \geq 0$. Daraus folgt wegen $u^0 \geq 0$ und $f(x^0) \leq 0$, dass $(u^0)'f(x^0) = 0$. Somit gilt

$$F(x^0) \leq F(x) + (u^0)'f(x) \text{ für alle } x \geq 0.$$

Für einen zulässigen Punkt x gilt $f(x) \leq 0$ und daher $(u^0)'f(x) \leq 0$. Also gilt $F(x^0) \leq F(x)$ und daher ist x^0 minimal. \square

Unter gewissen Regularitätsvoraussetzungen gilt auch die Umkehrung dieses Satzes. Die *Regularitätsvoraussetzung* ((Slater-Bedingung; constraint qualification)) schließt aus, dass M nur aus Randpunkten bzw. nur aus einem zulässigen Punkt besteht.

Beispiel 24.1 $F(x) = -x$, Restriktionen: $x^2 \leq 0, x \geq 0$.

Hier ist $x = 0$ der einzige zulässige Punkt. Die Lagrangefunktion $L(x, u) = -x + ux^2$ hat für $x = 0, u \geq 0$ keinen Sattelpunkt, denn $0 \leq -x + u^0 \cdot x^2$ ist in einer Umgebung von 0 nicht erfüllbar.

Regularitätsvoraussetzung :

$$M \text{ besitze einen inneren Punkt.} \tag{24.7}$$

Die Regularitätsvoraussetzung kann auf verschiedene Weise abgeschwächt werden. Hat das konvexe Programm nur lineare Nebenbedingungen, so kann sie ganz entfallen.

Nun gilt der fundamentale

Satz 24.5 (Satz von Kuhn-Tucker) Für das konvexe Programm

$$\min\{F(x) \mid f_i(x) \leq 0 \ (1 \leq i \leq m), \ x \geq 0\}$$

sei (24.7) erfüllt. $x^0 \geq 0$ ist genau dann Minimallösung des konvexen Programmes, wenn es ein $u^0 \geq 0$ gibt, so dass (x^0, u^0) Sattelpunkt von $L(x, u)$ ist.

Beweis: Dass ein Sattelpunkt zu einer Minimallösung führt, wurde schon in Satz 24.4 gezeigt. Sei nun x^0 eine Minimallösung des konvexen Programms. Wir definieren zwei konvexen Mengen im \mathbb{R}^{m+1} durch

$$C_1 := \{y \in \mathbb{R}^{m+1} \mid \exists x \geq 0 : y_0 \geq F(x), y_i \geq f_i(x) \ (1 \leq i \leq m)\}$$

$$C_2 := \{y \in \mathbb{R}^{m+1} \mid y_0 < F(x^0), y_i < 0 \ (1 \leq i \leq m)\}.$$

C_1 und C_2 sind konvex. Da die konvexe Funktion $F(x)$ auch stetig ist, ist C_2 offen. Ferner sind C_1 und C_2 disjunkt und nicht leer, da x^0 eine Minimallösung ist. Daher liefert der Trennungssatz (Satz 14.3) die Existenz eines Vektors $v \neq 0$ mit $v'y > v'z$ für $y \in C_1$, $z \in C_2$. Da die Komponenten von $z \in C_2$ beliebig negativ werden können, ist $v \geq 0$.

Setzt man $y = (F(x), f_1(x), \dots, f_m(x))'$, $z = (F(x^0), 0, \dots, 0)$, so ist z ein Randpunkt von C_2 und es gilt die Ungleichung mit "≥":

$$v_0 F(x) + \sum_{i=1}^m v_i f_i(x) \geq v_0 F(x^0) \quad \text{für alle } x \geq 0.$$

Daraus folgt $v_0 > 0$. Wäre nämlich $v_0 = 0$, so wäre $\sum_{i=1}^m v_i f_i(x) \geq 0$ für alle $x \geq 0$ und mindestens ein $v_i > 0$. Nach der Regularitätsvoraussetzung (24.7) gibt es einen inneren Punkt \tilde{x} . Für diesen gilt $\sum_{i=1}^m v_i f_i(\tilde{x}) < 0$, also ein Widerspruch. Daher ist $v_0 > 0$.

Man setze $u^0 := \frac{1}{v_0}(v_1, \dots, v_m)'$. Dann ist $u^0 \geq 0$ und $F(x) + (u^0)'f(x) \geq F(x^0)$ für alle $x \geq 0$. Setzt man hier $x = x^0$, so ist $(u^0)'f(x^0) = 0$. Ferner gilt $u'f(x^0) \leq 0$ für $u \geq 0$. Somit erhält man

$$F(x^0) + u'f(x^0) \leq F(x^0) + (u^0)'f(x^0) \leq F(x) + (u^0)'f(x),$$

also ist (x^0, u^0) Sattelpunkt von $L(x, u)$. □

Der Satz von Kuhn-Tucker liefert eine globale Charakterisierung: (x^0, u^0) müssen mit allen $x \geq 0, u \geq 0$ verglichen werden. Im Falle differenzierbarer Funktionen kann nun die globale Sattelpunktsbedingung durch lokale Bedingungen ersetzt werden.

Es seien $F(x)$ und $f_i(x)$ ($1 \leq i \leq m$) konvexe Funktionen, definiert auf dem \mathbb{R}^n , die stetige partielle erste Ableitungen besitzen mögen. Wir setzen

$$L_x := \left(\frac{\partial L}{\partial x_1}, \dots, \frac{\partial L}{\partial x_n} \right)', \quad L_u := \left(\frac{\partial L}{\partial u_1}, \dots, \frac{\partial L}{\partial u_m} \right)'.$$

Dann gilt

Satz 24.6 (Lokale Kuhn-Tucker Bedingungen) *Erfüllt das konvexe Programm eine Regularitätsbedingung und besitzen F bzw. f_i ($1 \leq i \leq m$) stetige erste partielle Ableitungen, so gilt: $x^0 \geq 0$ ist genau dann Minimallösung von*

$$\min\{F(x) \mid f_i(x) \leq 0 \ (1 \leq i \leq m), \ x \geq 0\},$$

wenn es ein $u^0 \geq 0$ gibt mit

$$L_x(x^0, u^0) \geq 0, \quad (x^0)'L_x(x^0, u^0) = 0 \tag{24.8}$$

$$L_u(x^0, u^0) \leq 0, \quad (u^0)'L_u(x^0, u^0) = 0. \tag{24.9}$$

Beweis:

1. Aus der Sattelpunktsbedingung (24.6) folgen (24.8) und (24.9):

Ist eine Komponente von $L_x(x^0, u^0)$ negativ, etwa $\frac{\partial L}{\partial x_k} < 0$, so gibt es ein $x \geq 0$ mit $x_l := x_l^0$ ($l \neq k$) und $x_k > x_k^0$ wobei $L(x, u^0) < L(x^0, u^0)$ gilt im Widerspruch zu (24.6). Also ist $L_x(x^0, u^0) \geq 0$. Wegen $x^0 \geq 0$ ist auch jeder Summand im Skalarprodukt $(x^0)'L_x(x^0, u^0)$ nichtnegativ. Gäbe es einen Index k mit $x_k^0 > 0$, $\frac{\partial L(x^0, u^0)}{\partial x_k} > 0$, so wäre für x , definiert durch $x_l := x_l^0$ ($l \neq k$), $0 < x_k < x_k^0$ die Ungleichung $L(x, u^0) < L(x^0, u^0)$ erfüllt im Widerspruch zu (24.6). Analog zeigt man die Bedingung (24.9).

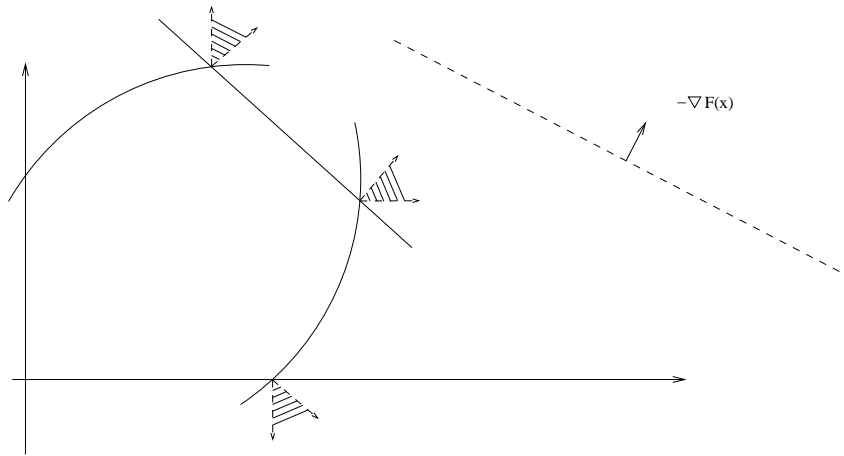


Abbildung 24.1: Geometrische Interpretation der lokalen Kuhn-Tucker Bedingungen (Satz 24.7)

2. Sind (24.8) und (24.9) erfüllt, so gilt (24.6): Für festes $u^0 \geq 0$ ist $L(x, u^0)$ eine konvexe Funktion im \mathbb{R}^n . Deshalb gilt die Subgradientenungleichung

$$L(x, u^0) \geq L(x^0, u^0) + (x - x^0)' L_x(x^0, u^0).$$

Nach (24.8) folgt daraus für $x \geq 0$ die Beziehung $L(x, u^0) \geq L(x^0, u^0)$. Ferner ist $L(x^0, u)$ affin-linear in u , daher gilt

$$L(x^0, u) = L(x^0, u^0) + (u - u^0)' L_u(x^0, u^0).$$

Daraus folgt für $u \geq 0$ mit (24.9) die Beziehung $L(x^0, u) \leq L(x^0, u^0)$.

□

Es sei x^0 nun ein zulässiger Punkt für das konvexe Programm

$$\min\{F(x) \mid f_i(x) \leq 0 \ (1 \leq i \leq m), x \geq 0\}.$$

Eine Restriktion $f_i(x) \leq 0$ bzw. eine Vorzeichenbedingung $x_j \geq 0$ heißt *aktiv* in x^0 , falls $f_i(x^0) = 0$ bzw. $x_j^0 = 0$ gilt. $I_0 = I(x^0)$ sei die Indexmenge der im Punkt x^0 aktiven Restriktionen, $J_0 = J(x^0)$ sei die Indexmenge der im Punkt x^0 aktiven Vorzeichenbedingungen. Dann lässt sich Satz 24.6 folgenderweise geometrisch interpretieren:

Satz 24.7 *Unter denselben Voraussetzungen wie in Satz 24.6 gilt:*

x^0 ist genau dann Optimallösung des konvexen Programms, wenn es $u^0 \geq 0$, $v^0 \geq 0$ gibt mit

$$-\nabla F(x^0) = \sum_{i \in I_0} u_i^0 \nabla f_i(x^0) + \sum_{j \in J_0} v_j^0 (-e_j). \quad (24.10)$$

Der Satz besagt, dass ein Minimalpunkt vorliegt, wenn der negative Gradient der Zielfunktion im Kegel liegt, der von den Gradienten der aktiven Restriktionen und Vorzeichenbedingungen aufgespannt wird (vgl. Abb.24.1).

Beweis:

1. Aus (24.8) und (24.9) folgt (24.10): Sei u^0 ein Vektor, der (24.8) und (24.9) erfüllt. Da $L_u(x^0, u^0) = f(x^0)$, kann nach (24.9) nicht gleichzeitig $f_i(x^0) < 0$ und $u_i^0 > 0$ gelten. Also ist $u_i^0 = 0$ für $i \notin I(x^0)$. Setzt man

$$v^0 = L_x(x^0, u^0) = \nabla F(x^0) + \sum_{i=1}^m u_i^0 \nabla f_i(x^0), \quad (24.11)$$

so ist wegen (24.8) $v^0 \geq 0$ und es kann nicht gleichzeitig $x_j^0 > 0$ und $v_j^0 > 0$ sein. Daher ist $v_j^0 = 0$ für $j \notin J(x^0)$. Aus (24.11) ergibt sich dann unmittelbar die Darstellung (24.10).

2. Wegen $L_u(x^0, u^0) = f(x^0)$ und $u_i^0 = 0$ für $i \notin I(x^0)$ folgt unmittelbar (24.9). Ferner erhält man aus (24.10) die Darstellung (24.11) und wegen $v_j^0 = 0$ für $j \notin J(x^0)$ erhält man (24.8).
□

Man überlege sich, was Satz 24.7 im Falle eines linearen Programms besagt.

Literaturverzeichnis

- [1] I. Adler, N. Megiddo, A simplex algorithm whose average number of steps is bounded between two quadratic functions of the smaller dimension. *J.Assoc. Comput. Mach.* **32**, 1985, 871–895.
- [2] R.K. Ahuja, T.L. Magnanti und J.B. Orlin, *Network Flows*, Prentice Hall: Englewood Cliffs (1993).
- [3] E. Balas und E. Zemel, An algorithm for large zero-one knapsack problems. *Operations Research* **28**, 1980, 1130–1154.
- [4] R. H. Bartels, A stabilization of the simplex method, *Num. Math.* **16**, 1971, 414–434.
- [5] R. G. Bland, New finite pivoting rules for the simplex method, *Math. of Oper. Res.* **2**, 1977, 103–107.
- [6] O. Borůvka, O jistém problému minimálním. *Práce Moravské Přírodovědecké Společnosti* **3**, 1926, 153–154.
- [7] K.H. Borgwardt, *The Simplex Method: A Probabilistic Analysis*. Berlin: Springer, 1977.
- [8] R. E. Burkard, *Methoden der ganzzahligen Optimierung*, Springer: Wien (1972).
- [9] R.E. Burkard, B. Klinz und R. Rudlf, Perspectives of Monge properties in optimization. *Discrete Applied Maths* **70**, 1996, 95–161.
- [10] R. E. Burkard, J. Offermann, Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme, *Zeitschrift für Operation Research* **21 B**, 1977, 121–132.
- [11] A. Charnes, W. W. Cooper, A. Henderson, *An Introduction to Linear Programming*, Wiley & Sons: New York (1953).
- [12] L. Collatz, Some application of nonlinear optimization. In: *P. L. Hammer, G. Zoutendijk (edts.) Math. Progr. in Theory and Practice*, North Holland: Amsterdam (1974), pp. 139–159.
- [13] L. Collatz, W. Wetterling, *Optimierungsaufgaben*, Springer: Berlin–Heidelberg–New York (1971).
- [14] G. B. Dantzig, *Lineare Optimierung und Erweiterungen*, Springer: Berlin (1966).
- [15] G. B. Dantzig, Ph. Wolfe, Decomposition principle for linear programs, *Op. Res.* **8**, 1960, 101–111.
- [16] W. Dinkelbach, *Sensitivitätsanalysen und parametrische Optimierung*, Springer-Verlag: Berlin–Heidelberg–New York (1969).

- [17] J. Farkas, Über die Theorie der einfachen Ungleichungen, *J. für Reine und Angew. Math.* **124**, 1902, 1–24.
- [18] J. J. Forrest, J. A. Tomlin, Updating triangular factors of the basis to maintain sparsity in the product form simplex method, *Math. Progr.* **2**, 1972, 263–278.
- [19] P. Gaács, L. Lovász, Khachyan’s algorithm for linear programming. *Math. Programming Study* **14**, 1981, 61–68.
- [20] S. I. Gass, *Linear programming: methods and application*, Mc Graw-Hill: New York (1964).
- [21] P.C. Gilmore und R.E. Gomory. A linear programming approach to the cutting stock problem. *Operations Research* **9**, 1961, 849–859.
- [22] D. Goldfarb, J. K. Reid, A practicable steepest edge simplex algorithm, *Mathematical Programming* **12**, 1977, 361–371.
- [23] A. J. Goldman, A. W. Tucker, Theory of linear programming. In: *H. W. Kuhn, A. W. Tucker (Hrsg.) Linear inequalities and related Systems*, Princeton: New Jersey (1956), pp. 53–97.
- [24] P. Gordan, Über die Auflösung linearer Gleichungen mit reellen Coeffizienten, *Math. Annalen* **6**, 1873, 23–28.
- [25] I. Heller und C.B. Tompkins, An extension of a theorem of Dantzig’s. In *Linear Inequalities and Related Systems* (H.W. Kuhn and A.W. Tucker, eds.), Princeton University Press, Princeton, N.J., 1956, pp. 247–254.
- [26] F. L. Hitchcock, The distribution of a product from several sources to numerous localities. *J. Math. Phys.* **20**, 1941, 224–230.
- [27] A.J. Hoffman and J.B. Kruskal, Integral boundary points of convex polyhedra. In *Linear Inequalities and Related Systems* (H.W. Kuhn and A.W. Tucker, eds.), Princeton University Press, Princeton, N.J., 1956, pp. 223–246.
- [28] D.B. Judin und A.S. Nemirovskii, Informational complexity and efficient methods for the solution of convex extremal problems (Russisch). *Ekonomika i Matematicheskije Metody* **12**, 1976, 357–369. Englische Übersetzung in *Matekon* **13**, 1977, 25–45.
- [29] L. V. Kantorowicz, Mathematical methods in the organization and planning of production. Englische Übersetzung des Originales von 1939 in: *Management Science* **6**, 1960, 366–422.
- [30] N. Karmarkar, A new polynomial-time algorithm for linear programming. *Combinatorica* **4**, 1984, 373–397.
- [31] H. Kellerer, U. Pferschy und D. Pisinger, *Knapsack Problems*. Springer, Berlin-Heidelberg-New York, 2004.
- [32] L.G.Khachyan, A polynomial algorithm in linear programming (Russisch), *Dokl. Akad. Nauk SSSR* **244**, 1979, 1093–1096. Englische Übersetzung in *Soviet Math. Dokl.* **20**, 1979, 191–194.
- [33] V. Klee, G.L. Minty, How good is the simplex algorithm? In *Inequalities III* (Hrsg. O. Shisha), New York: Academic Press, 1972, pp. 159–175.
- [34] W. Knödel, Lineare Programme und Transportaufgaben. *MTW, Z.moderne Rechentechn. Automat.* **7**, 1960, 63–68.

- [35] H. W. Kuhn, R. E. Quandt, An experimental study of the simplex method, *Proc. Symposia in Appl. Maths.* **15**, 1963, 107–124.
- [36] H. W. Kuhn, R. E. Quandt, On upper bounds for the number of iterations in solving linear programs, *Op. Res.* **12**, 1964, 161–165.
- [37] H. P. Künzi, Die Simplexmethode zur Bestimmung einer Ausgangslösung bei bestimmten linearen Programmen, *Unternehmensforschung* **2**, 1958, 60–69.
- [38] K. T. Marshall, J. W. Suurballe, A note on cycling in the simplex method, *Naval Res. Log. Quat.* **16**, 1969, 121–137.
- [39] K. Murty, *Linear and Combinatorial Programming*, Wiley & Sons: New York–London–Sydney–Toronto (1976).
- [40] C. Roos, T. Terlaky und J.-Ph. Vial, *Theory and Algorithms for Linear Optimization*. Chichester: J. Wiley and Sons, 1997
- [41] R. Schrader, Ellipsoid methods. In *Modern Applied Mathematics* (Hrsg. B. Korte), Amsterdam: North Holland, 1982, pp.265–311.
- [42] A. Schrijver, *Theory of linear and integer programming*. J. Wiley and Sons, Chichester, 1986.
- [43] P.D. Seymour, Decomposition of regular matroids. *J. Combinatorial Theory (B)* **28**, 1980, 305–359.
- [44] E. Stiemke, Über positive Lösungen homogener linearer Gleichungen, *Math. Annalen* **76**, 1915, 340–342.
- [45] G. J. Stigler, The cost of subsistence. *J. Farm. Econ* **27**, 1945, 303–314.
- [46] J. Stoer, C. Witzgall, *Convexity and Optimization in Finite Dimensions I*, Springer: Berlin, 1970.
- [47] É. Tardos, A strongly polynomial minimum cost circulation problem. *Combinatorica* **5**, 1985, 247–255.
- [48] T. Terlaky, An easy way to teach interior.point methods. *European Journal of Operational Research* **130**, 2001, 1–19.
- [49] A. W. Tucker, Dual systems of homogeneous linear relations. In: *Linear inequalities and related systems* (Hrsg. H. W. Kuhn, A. W. Tucker), Princeton: New Jersey, 1956, pp. 3–18.
- [50] A.F. Veinott, Jr. und G.B. Dantzig, Integral extreme points. *SIAM Review* **10**, 1968, 371–372.
- [51] H. M. Wagner, The Dual Simplex Algorithm for Bounded Variables, *Naval Res. Log. Quart.* **5**, 1958, 257–261.
- [52] H. Whitney, On the abstract properties of linear dependence. *Amer.J.Math.* **57**, 1935, 509–533.
- [53] Ph. Wolfe, The simplex method for quadratic programming. *Econometrica* **27**, 1959, 382–398.
- [54] N. Zadeh, A bad network problem for the simplex method and other minimum cost flow algorithms, *Mathematical Programming* **5**, 1973, 255–266.