# Vorlesung Lineare und kombinatorische Optimierung

Manfred Schmelzer
Informatik Stuttgart
manfred.schmelzer@informatik.uni-stuttgart.de

4. Februar 2001

# Inhaltsverzeichnis

# Literatur


[LA] Falko Lorenz: Lineare Algebra 1, Spektrum Verlag 1992 (vormals BI);

[LA1] Rolf Walter: Einführung in die lineare Algebra, Vieweg 1982;

[LA2] G. Fischer: Lineare Algebra, Vieweg Verlag;

[LA3] K. Jänich: Linear Algebra, Springer Verlag;

[LA4] W. Klingenberg: Lineare Algebra und Geometrie, Springer Verlag;

[LA5] M. Koecher: Lineare Algebra und analytische Geometrie, Springer Verlag;

[LA6] B. Brieskorn: Lineare Algebra und analytische Geometrie, Vieweg Verlag;

[LA7] W. Greub: Linear Algebra, Graduate Texts in Mathematics, Springer Verlag;

[V1] Vasek Chvatal: Linear Programming, W. H. Freeman and Company 1983;

[V2] M.S. Bazaraa, J.J Larvis, H.D. Sherali: Linear Programming and Network Flows, Wiley 1990;

[V3] Christos Papadimitriou, Kenneth Steiglitz: Combinatorial Optimization, Dover Publications 1998 (früher Prentice Hall 1982);

[V4] Alexander Schrijver: Theory Of Linear And Integer Programming, Wiley 1986/1999;

[V5] Manfred Padberg: Linear Optimization and Extensions, Springer 1995/1999;

[V6] Dieter Jungnickel: Graphen, Netzwerke und Algorithmen, BI Verlag 1994;

[V7] R. J. Vanderbei: Linear Programming: Foundations and Extensions, Kluwer Academic Publishers 1997;

[E1] Roger Fletcher: Practical Methods of Optimization, Wiley 1991/2000;

[E2] Dieter Jungnickel: Optimierungsmethoden, Springer 1999;

[E3] Tamas Terlaky: Interior Point Methods Of Mathematical Programming, Kluwer 1996;

[E4] J.E. Beasley: Advances in Linear and integer Programming, Clarendon Press 1996;

[R1] M. Benichou, J. M. Gauthier, P. Girodet, G. Hentges, G. Ribiere: The efficient solution of large scale linear programming problems. Some algorithmic techniques and computational results, Math. Prog. 13 (1977) p. 280-322;

[R2] A. L. Brearley, G. Mitra, H. P. Williams: Analysis of mathematical programming problems prior to applying the simplex algorithm, Math. Prog. 8 (1975) p. 54-83;

[R3] R. Bixby: Progress in Linear Programming, ORSA Journal on Computing 6 (1994) p. 15-22;

[R4] H. Crowder, E. L. Johnson, M.W. Padberg: Solving large scale zero-one linear programming problems, Oper. Res. 31 (1983) p. 803-834;

[R5] E. L. Johnson: Modeling and strong linear programs for mixed integer programming, In: Algorithms and Model Formulation in Mathematical Programming, S.W. Wallace (Ed.) (1989) Springer Verlag, p. 1-44;

[R6] N. Karmakar: A Polynomial Time Algorithm for Linear Programming, Combinatoria 4 (1984) p. 373-395;

[R7] Cs. Meszaros: Fast Cholesky factorization for interior point methods of linear programming, Computing and Mathematics with Application, 31 (4/5) (1996) p. 49-54;

[R8] Cs. Mezaros: The Efficient Implementation of Interior Point Methods for Linear Programming and their Applications, Ph.D. Thesis Eötvös Lorand University of Sciences, Ph. D. School of Operations research, Budapest;

[R9] G. Mitra: Investigation of some branch and bound strategies for the solution of mixed integer linear programs, Math. Programming 4 (1973) p. 155-170;

[R10] U. H. Suhl, R. Szymanski: Supernode Processing of Mixed Integer Models, Computational Optimization and Applications 3 (1994) p. 317-331;

[R11] U. H. Suhl, L. M. Suhl: Solving Airline Fleet Scheduling Problems with Mixed Integer Programming, in "Operational Research in Industry" (Eds. T. Ciriani, S. Gliozzi, E.L. Johnson, R. Tadei), Mac. Millan Press, (1999) p. 135-156;

[R12] U. H. Suhl, Cs. Meszaros: Advanced preprocessing for simplex and interior point algorithms, Lehrstuhl für Wirtschaftsinformatik der Freien Universität Berlin (1999);

[R13] L. A. Wosley: Tight formulations for mixed integer programming: A survey, Math. Programming 45 (1989) p. 173-191;

Die mit LA bezeichnete Literatur betrifft die Lineare Algebra, die mit V bezeichneten Verweise betreffen das Vorlesungsthema, die mit E bezeichnete Literatur sind Ergänzungen hierzu. Die mit R bezeichneten Papers sind Forschungsrelevant auf dem Gebiet der linearen Optimierung.

# 1 Notation in der linearen Algebra

Es bezeichnet $\mathbb{R}$ die reellen Zahlen, ferner

$$\mathbb{R}_\infty := \mathbb{R} \cup \infty, \quad \text{sowie} \quad \mathbb{R}_{\geq 0} = \{x \in \mathbb{R} \,|\, x \geq 0\}.$$

Die Anordnung der reellen Zahlen setzen wir fort auf $\mathbb{R}_\infty$ via $r < \infty$ für alle $r \in \mathbb{R}$. Der $\mathbb{R}^n$ bezeichnet den kanonische Vektorraum der Dimension $n$ über den reellen Zahlen. Die Standardbasis ist definiert durch

$$(e_1, \dots, e_n), \quad \text{mit} \quad e_i \in \mathbb{R}^n, \quad (e_i)_j = \begin{cases} 1 & \text{falls } i = j, \\ 0 & \text{sonst.} \end{cases}$$

Unter den Vektoren des $\mathbb{R}^n$ definieren wir eine Teilordnung, für $x, y \in \mathbb{R}^n$ gilt:

$$x \leq y \quad \text{genau dann wenn} \quad x_i \leq y_i \quad \text{für alle} \quad i \in \{1, \dots, n\}.$$

Den euklidischen Abstand $d(x, y) \in \mathbb{R}$ zweier Vektoren $x, y \in \mathbb{R}^n$ ist definiert durch:

$$d(x, y) := \sqrt{\sum_i (x_i - y_i)^2}.$$

Ferner ist das Skalarprodukt zweier Vektoren $x, y \in \mathbb{R}^n$ definiert durch

$$xy := \sum_i x_i y_i.$$

Eine Matrix $A \in \mathbb{R}^{m \times n}$ besteht aus Einträgen $a_{ij} \in \mathbb{R}$ mit den Zeilenindices $I = \{1, \dots, m\}$ und Spaltenindices $J = \{1, \dots, n\}$:

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

Mehrere Spaltenvektoren $A_1, \dots, A_n \in \mathbb{R}^m$ bilden eine Matrix $A = (A_1, \dots, A_n) \in \mathbb{R}^{m \times n}$, es gilt

$$A = (a_{ij}) \quad \text{mit} \quad a_{ij} = (A_j)_i.$$

Die Einheitsmatrix $E_m = (e_1, \dots, e_m) \in \mathbb{R}^{m \times m}$ mit den Vektoren der Standardbasis als Spaltenvektoren hat die Gestalt

$$E_m = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{pmatrix}.$$

Auf der Haupt-Diagonalen steht die 1, alle sonstigen Einträge sind 0. Zu jeder Matrix $A \in \mathbb{R}^{m \times n}$ existiert eine transponierte Matrix $A^T \in \mathbb{R}^{n \times m}$ definiert durch

$$A^T_{ij} := A_{ji}.$$

Die Zeilen von $A$ werden zu den Spalten von $A^T$ gemacht. Sei $B \in \mathbb{R}^{r\times m}$ mit $B^T \in \mathbb{R}^{m\times r}$ so gilt

$$(BA)^T = A^T B^T.$$

Die Reihenfolge der Matrizenmultiplikation dreht sich beim transponieren um. Vektoren werden üblicherweise als Spaltenvektoren interpretiert, d.h. der $\mathbb{R}^n$ wird mit dem $\mathbb{R}^{n\times 1}$ identifiziert. Das Skalarprodukt zweier Vektoren $x, y \in \mathbb{R}^n$ können wir dann als Matrizenprodukt ansehen:

$$xy = x^T y \in \mathbb{R}.$$

Das Produkt $yx^T$ ist eine quadratische Matrix im $\mathbb{R}^{n\times n}$. Im weiteren differenzieren wir Skalarmultiplikation oft nicht von der Matrizenmultiplikation, und verwenden uneinheitlich die Notationen $xy$ sowie $x^T y$ je nach Kontext. (Und Vektoren sowohl als Zeilen- sowie auch als Spaltenvektoren).

Quadratische Matrizen $P \in \mathbb{R}^{n\times n}$, deren Einträge 0 oder 1 sind, $P_{ij} \in \{0,1\}$ heißen Permutationsmatrizen, sofern in jeder Zeile und jeder Spalte genau eine 1 steht. Sei $d = (1,\dots,1)^T \in \mathbb{R}^n$ so gilt

$$Pd = d, \quad \text{und} \quad d^T P = d^T.$$

Dies sind $2n-1$ linear unabhängige Gleichungen. Permutationsmatrizen codieren eine Permutation via

$$\sigma : \{1,\dots,n\} \to \{1,\dots,n\} \quad \text{durch} \quad \sigma(i) = j \iff P_{ij} = 1$$

Basisergänzungssatz: Seien $a_1,\dots,a_r$ linear unabhängige Vektoren eines Vektorraums $V$ der Dimension $n$, d.h. $r \le n$. Dann gibt es $a_{r+1},\dots,a_n \in V$, sodaß $(a_1,\dots,a_n)$ eine Basis von $V$ ist.

Sei $L : U \to V$ eine lineare Abbildung, $\bar{u} = (u_1,\dots,u_n)$ eine Basis von $U$, sowie $\bar{v} = (v_1,\dots,v_m)$ eine solche von $V$, es gilt $\dim U = n$, $\dim V = m$. Die Bilder

$$Lu_i, \quad i = 1,\dots,n,$$

besitzen eine Basisdarstellung in $\bar{v}$:

$$Lu_i = \sum_{j=1}^{m} a_{ji} v_j,$$

mit eindeutigen Koordinaten $a_{ji} \in \mathbb{R}$. In Bezug auf die Basen $\bar{u} = (u_1,\dots,u_n)$, $\bar{v} = (v_1,\dots,v_m)$ ist $L$ eindeutig bestimmt durch die Koordinatenmatrix $A \in \mathbb{R}^{m\times n}$

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

Die Spalten $A_i$ von $A = (A_1,\dots,A_n)$ sind die Koordinatenvektoren der Bilder $Lu_i$. Sei nun $M : V \to W$ eine zweite lineare Abbildung, sowie $\bar{w} = (w_1,\dots,w_r)$

eine Basis von $W$, $\dim W = r$. Die Koordinatenmatrix von $M$ sei $B \in \mathbb{R}^{r\times m}$ mit,

$$B = \begin{pmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{r1} & \dots & b_{rm} \end{pmatrix}.$$

So gilt für die Zusammengesetzte Abbildung $M \circ L : U \to W$

$$\begin{aligned} M \circ Lu_i &= M\left(\sum_{j=1}^{m} a_{ji} v_j\right) = \sum_{j=1}^{m} a_{ji} M(v_j) = \sum_{j=1}^{m} a_{ji} \sum_{k=1}^{r} b_{kj} w_k \\ &= \sum_k \left(\sum_j b_{kj} a_{ji}\right) w_k = \sum_{k=1}^{r} BA_{ki} w_k. \end{aligned}$$

Es ist $BA$ Koordinatenmatrix der linearen Abbildung $M \circ L$ in bezug auf die entsprechenden Basen. Die Basisabhängigkeit der Koordinatenmatrix führt zur Bezeichnung

$$A = L^{\bar{u}}_{\bar{v}}, \quad B = M^{\bar{v}}_{\bar{w}}.$$

Somit folgt

$$M \circ L^{\bar{u}}_{\bar{w}} = M^{\bar{v}}_{\bar{w}} L^{\bar{u}}_{\bar{v}}.$$

Sei $\bar{u}' = (u'_1, \dots, u'_n)$ etwa eine zweite Basis von $U$. Die Koordinatenmatrix

$$E^{\bar{u}'}_{\bar{u}}, \quad \text{mit} \quad u'_i = \sum_{j=1}^{n} E_{ji} u_j.$$

beschreibt den Basiswechsel von $\bar{u}'$ zu $\bar{u}$ für die identischen Abbildung $id_U : x \mapsto x$, für alle $x \in U$. Es folgt z.B.

$$L^{\bar{u}'}_{\bar{v}} = L^{\bar{u}}_{\bar{v}} E^{\bar{u}'}_{\bar{u}}.$$

Soweit wir die Standardbasen $(e_1, \dots, e_n)$ der $\mathbb{R}^n$, $n \in \mathbb{N}$ zugrundelegen, identifizieren wir die Koordinatenmatrizen mit den beschriebenen linearen Abbildungen. Die Hintereinanderausführung von linearen Abbildungen entspricht dann der Matrizenmultiplikation.

Sei $A \in \mathbb{R}^{m\times n}$ eine Matrix mit Spaltenvektoren $(A_1, \dots, A_n)$. Die Abbildung

$$L : x \mapsto Ax, \quad \mathbb{R}^n \to \mathbb{R}^m$$

ist injektiv genau dann wenn die Spaltenvektoren $A_1, \dots, A_n$ linear unabhängig sind. Die maximale Anzahl $r$, $(r \le n)$ linear unabhängiger Spalten einer Matrix ist der Spaltenrang. Entsprechend ist der Zeilenrang definiert. Der Spaltenrang einer Matrix $A$ ist gleich dem Zeilenrang und wird mit $\operatorname{Rang} A$ bezeichnet, es handelt sich um die Dimension das Bildes. Für eine lineare Abbildung

$$L : x \mapsto Ax, \quad \mathbb{R}^n \to \mathbb{R}^m$$

ist das Bild $L(\mathbb{R}^n) \subset \mathbb{R}^m$ ein Untervektorraum, für dessen Dimension gilt

$$\operatorname{Rang} A = \dim L(\mathbb{R}^n).$$

Sei $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ die Lösungsmenge $U = \{x \in \mathbb{R}^n \,|\, Ax = b\} \subset \mathbb{R}^n$ eines linearen Gleichungssystems ist ein affiner Raum. Im Fall $b \neq 0$ ist dies kein Untervektorraum des $\mathbb{R}^n$. Sei $U \subset \mathbb{R}^n$ ein affiner Unterraum. Eine affine Abbildung $\alpha : U \to \mathbb{R}^m$ ist definiert durch

$$\alpha : x \mapsto Ax + b, \quad \text{für} \quad x \in U.$$

Die beschreibende Matrix $A \in \mathbb{R}^{m\times n}$ und Vektor $b \in \mathbb{R}^m$ sind im Fall $U \neq \mathbb{R}^n$ nicht eindeutig bestimmt. Es gibt dann viele weitere Paare $(A', b')$ mit $A' \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ die dieselbe affine Abbildung beschreiben. Sei

$$F \subset U \subsetneqq \mathbb{R}^n$$

Teilmenge eines affinen Raumes $U$, so gilt für alle $x \in F$

$$A'x + b' = Ax + b.$$

Für eine Matrix $A \in \mathbb{R}^{m\times n}$ sei $A = (A_1, \ldots, A_n)$ deren Aufteilung in $n$ Spaltenvektoren. Für eine Teilmenge $B \subset \{1, \ldots, n\}$ der Indices gilt $B = \{i_1, i_2, \ldots, i_r\}$ mit $i_1 < i_2 < \ldots < i_r$ und $r = |B|$. Dann ist

$$A_B := (A_{i_1}, A_{i_2}, \ldots, A_{i_r})$$

diejenige Teilmatrix von $A$, welche aus den Spalten mit Indices in $B$ besteht. Entsprechend ist für $x \in \mathbb{R}^n$ definiert

$$x_B = (x_{i_1}, \ldots, x_{i_r}).$$

Mit $N = \{1, \ldots, n\} \setminus B$ notieren wir $A = (A_B, A_N)$, obwohl die Spalten von $A_B$ <u>nicht</u> die ersten $|B|$- vielen Spalten von $A$ sind. Jedoch gilt mit der entsprechenden Notation $x = (x_B, x_N)$ :

$$Ax = A_B x_B + A_N x_N.$$

Eine solche Aufspaltung von Matrizenmultiplikationen ist im weiteren essentiell.

# 2 Einführung

## 2.1 Definition eines LP (linearen Programms)

Ein lineares Programm wird mit LP bezeichnet. Es handelt sich um ein Optimierungsproblem mit linearer Kostenfunktion und linearen Randbedingungen. Auf einer Menge $F \subset \mathbb{R}^n$ der zulässigen Lösungen ist eine reellwertige Kostenfunktion $z : F \to \mathbb{R}$ definiert, die es zu minimieren gilt (oder zu maximieren).

**Definition 1** *Ein Lineares Programm (LP)*

$$\min_{x \in F} z(x),$$

*heißt zulässig, falls die zulässige Menge $F$ nicht leer ist, $F \neq \emptyset$. Ein zulässiges LP heißt beschränkt, falls die Kostenfunktion nach unten beschränkt ist,*

$$min_{x \in F} z(x) \neq -\infty.$$

Die zulässige Menge $F$ ist der Durchschnitt von (endlich vielen) Halbräumen, $F = \cap_{i \in I} H_i$ mit $H_i = \{x \in \mathbb{R}^n \mid a_i x \leq b_i\}$ (mit $a_i \in \mathbb{R}^n, b_i \in \mathbb{R}$ ). Somit ist die zulässige Menge $F$ eines LP eine konvexe Menge. Ersetzt man die Zielfunktion $z$ durch

$$-z : x \mapsto -z(x),$$

so wird aus einem Minimierungs- ein Maximierungsproblem und umgekehrt.

Wir beschreiben ein LP genauer mit linearen Gleichungen und Ungleichungen.

**Definition 2** *Ein LP in Standardform*

$$\min_{x \in F} z(x),$$

*hat die Gestalt*

$$\begin{array}{ll} \min & cx \\ \textit{unter} & Ax = b, \quad x \geq 0, \end{array}$$

*mit einer Matrix $A \in \mathbb{R}^{m \times n}$, einem Vektor $b \in \mathbb{R}^m$ sowie einem Vektor $c \in \mathbb{R}^n$. Das Tripel $(A, b, c)$ nennen wir eine Darstellung des LP. Mit dieser gilt*

$$F = \{x \in \mathbb{R}^n \mid Ax = b\,,\, x \geq 0\} \quad \textit{und} \quad z(x) = cx.$$

Wir betrachten zwei LP's $\min_{x \in F} z(x)$, $\min_{x \in F'} z'(x)$ als gleich, bei identischer zulässiger Menge $F = F'$, und sofern die Kostenfunktionen sich nur um einen konstanten Wert unterscheiden, für ein $d \in \mathbb{R}$ gilt:

$$z(x) = z'(x) + d, \quad \text{für alle} \quad x \in F.$$

**Definition 3** *Zwei Darstellungen von LP's in Standardform sind äqiuvalent,*

$$(A, b, c) \sim (A', b', c'),$$

*wenn es eine reguläre Matrix* $D \in \mathbb{R}^{m\times m}$ *und ein* $\delta \in \mathbb{R}^m$ *gibt mit*

$$A' = DA \quad b' = Db\,, \quad c' = c + \delta A.$$

Äquivalente Darstellungen beschreiben das gleiche LP. Sei $\min_{x\in F} z(x)$ durch $(A, b, c)$ dargestellt, und $\min_{x\in F'} z'(x)$ entsprechend durch das äquivalente Tupel $(A', b', c')$, dann gilt

$$F = F' \quad \text{und} \quad z(x) = z'(x) + \delta b.$$

Der konstante Term $\delta b$ ändert das Optimierungsproblem nicht.

Hierbei gilt auch die Umkehrung, sofern es einen inneren Punkt $x \in F$ gibt mit $x_i > 0$ für alle Koordinaten $i = 1, \ldots, n$. Dann sind zwei Darstellungen des LP $\min_{x\in F} z(x)$ stets äquivalent.

Vorweggenommen, die Iterationsschritte (Pivotschritte) des Simplexalgorithmus beinhalten den Wechsel solcher Darstellungen, jedes Tableau beschreibt dasselbe LP in einer anderen (Basis-) Darstellung.

**Definition 4** *Ein LP* $\min_{x\in F} z(x)$ *in allgemeiner Form ist gegeben durch ein* $A \in R^{m\times n}$, $x, c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ *sowie Partitionen*

$$e \,\dot{\cup}\, l = \{1, \ldots, m\}, \quad p \,\dot{\cup}\, u = \{1, \ldots, n\},$$

*der Indicesmengen, sodaß mit den entsprechenden Blockzerlegungen*

$$x = \begin{pmatrix} x_p \\ x_u \end{pmatrix}, b = \begin{pmatrix} b_e \\ b_l \end{pmatrix}, \quad \textit{und} \quad A = \begin{pmatrix} A_e \\ A_l \end{pmatrix},$$

*das LP folgende Form hat:*

$$\begin{array}{lll} & \min & cx \\ (PA) & \textit{unter} & A_e x = b_e, \\ & & A_l x \geq b_l, \quad x \geq 0. \end{array}$$

*Hierbei gilt also* $z(x) = cx$, *sowie*

$$F = \{x \in \mathbb{R}^n \mid x_p \geq 0,\, A_e x = b_e,\, A_l x \geq b_l\}.$$

*Ein LP* $\min_{x\in F} z(x)$ *in Ungleichungsform hat die Gestalt*

$$\begin{array}{ll} \min & cx \\ \textit{unter} & Ax \leq b, \quad x \geq 0, \end{array}$$

Durch Schlupfvariablen, sowie zerlegung eines ungebundenen Vektors $y \in \mathbb{R}^n$ in Positiv- und Negativanteil

$$y = y^+ - y^-, \quad \text{mit} \quad y^+, y^- \geq 0,$$

kann aus einem allgemeinen LP ein äquivalentes LP in Standardform (Ungleichungsform) transformiert werden Wir geben ein Beispiel zur Transformation von Ungleichungsform in Standardform:

## 2.2 Transformation eines LP in Standardform

Ein LP in Ungleichungsform sei gegeben durch:

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax \leq b, \quad x \geq 0. \end{array}$$

Es gilt $F = \{x \in \mathbb{R}^n \mid Ax \leq b\,,\, x \geq 0\}$ mit $z(x) = cx$. Dieses ist äquivalent zu folgendem LP $\min_{x \in F'} z'(x)$ in Standardform:

$$F' = \{(x,y) \in \mathbb{R}^{n+m} \mid (A, E_m)\binom{x}{y} = b\,,\, (x,y) \geq 0\},$$

mit $z'(x,y) = cx$. Die Transformations-Abbildung $\tau : F \to F'$ ist gegeben durch:

$$\tau\,:\, x \mapsto (x, b - Ax).$$

Es ist $\tau$ werte- und optimalitäts-erhaltend:

$$\begin{array}{l} z(x) = z'(\tau(x)), \quad \text{für alle} \quad x \in F, \\ \min_{x \in F} z(x) = \min_{y \in F'} z'(y). \end{array}$$

Daher ist $x \in F$ optimal genau dann wenn $(x,y)$ mit $y = b - Ax$ optimal in $F'$ ist. Genauer ist $\tau$ sogar ein affiner Werteisomorphismus, d.h $\tau$ ist bijektiv, werteerhaltend und affin, mit $\sum_i \lambda_i = 1$ gilt

$$\sum_i \lambda_i \tau(x_i) = \tau(\sum_i \lambda_i x_i).$$

(sofern alle $x_i \in F$, sowie $\sum_i \lambda_i x_i \in F$ gilt.)

## 2.3 Beispiele zur linearen Optimierung

Wir betrachten ein Entscheidungsproblem zur Raffinierung von Rohöl. Dieses soll mittels zweier verschiedener Cracktechnologien $C1$, $C2$ in die Endprodukte

- Schweröl ($S$),
- Mittelschweres Öl ($M$),
- Leichtöl ($L$)

verarbeitet werden. Und dies möglichst Kostengünstig. Die Beiden Verfahren $C1$, $C2$ unterscheiden sich in den Kosten, eine Tonne Rohöl nach $C1$ zu cracken kostet 30 Euro, eine Tonne Rohöl mit $C2$ zu verarbeiten kostet 50 Euro. Die beiden Verfahren unterscheiden sich weiter nach der resultierenden Produktmischung an Leicht- Mittelschwerem und Schweröl. Die Verarbeitung von 10 Tonnen Rohöl nach dem Verfahren

$$\begin{array}{l} C1 \quad \text{liefert} \quad 2 \text{ to S}, \quad 2 \text{ to M}, \quad 1 \text{ to L}, \quad \text{und} \\ C2 \quad \text{liefert} \quad 1 \text{ to S}, \quad 2 \text{ to M}, \quad 4 \text{ to L}. \end{array}$$

Die Entscheidungsvariablen $x_1, x_2 \in \mathbb{R}$ bezeichnen die Menge Rohöl die nach Verfahren $C1$ gecrackt wird, bzw. nach $C2$. Beide Variablen sollen so gesetzt werden daß die Lieferverpflichtung (in der betrachteten Planungsperiode) folgende Mengen erfüllt werden:

$$\text{3to S}, \quad \text{5to M}, \quad \text{4to L}.$$

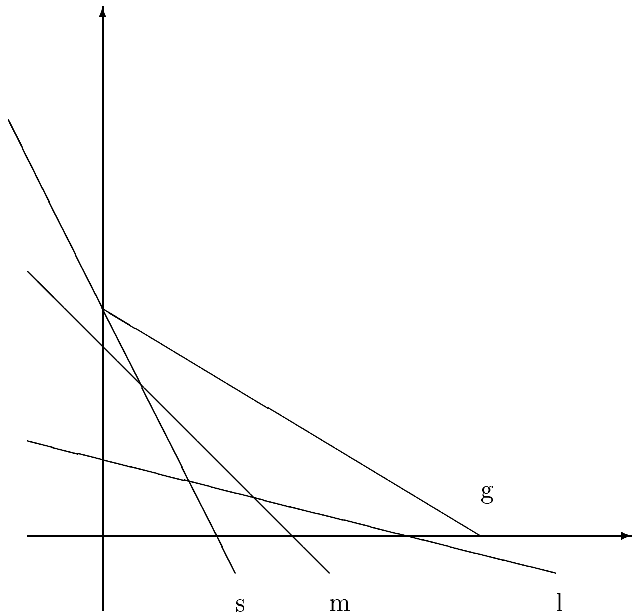
Aufgrund obiger Verteilung im Produktionsmix beider Verfahren führt dies zu folgenden Restriktionen eines Linearen Programms:

$$\begin{array}{rcrcl} 2x_1 & + & x_2 & \geq & 3 \\ 2x_1 & + & 2x_2 & \geq & 5 \\ x_1 & + & 4x_2 & \geq & 4 \end{array}$$

Die erste Zeile adressiert die Lieferverpflichtung für Schweröl ($S$), die zweite diejenige für mittelschweres Öl. Die dritte steht für Leichtöl. Die Zielfunktion

$$30x_1 + 50x_2,$$

steht für die Gesamtkosten des Produktionsprogramms mit $x_1$ Tonnen Rohöl für Verfahren $C1$, und $x_2$ Tonnen für $C2$. Somit haben wir unser LP komplett, es hat die Struktur wie in folgender Graphik:




Die Gerade $g : 30x_1 + 50x_2 = 150$ beschreibt alle Punkte $(x_1, x_2)$ gleicher Kosten zu 150 Euro. Die Paralellen von $g$ beschrieben Punkte zu anderen Gesamtkosten. Wir können $g$ nun so verschieben, daß gerade noch ein Punkt der Geraden im zulässigen Bereich liegt, hier oberhalb aller Lieferverpflichtungsgeraden $s, m, l$.

Die geringsten Produktionskosten liegen (graphisch bestimmt) im Punkt $(x_1, x_2) = (2, 0.5)$ mit den Gesamtkosten

$$30x_1 * 50x_2 = 60 + 25 = 85 \quad \text{Euro.}$$

Diese Lösung liegt in einer "Ecke" des zulässigen Gebietes oberhalb der Geraden $s, m, l$.

Mit $c = (30, 50) \in \mathbb{R}^2$, $x = (x_1, x_2) \in \mathbb{R}^2$, $b = (3, 5, 4) \in \mathbb{R}^3$ sowie

$$A = \begin{pmatrix} 2 & 1 \\ 2 & 2 \\ 1 & 4 \end{pmatrix}.$$

lautet die Algebraische Formulierung wie folgt:

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax \geq b, \quad x \geq 0. \end{array}$$

Die Zeilen des Ungleichungssystems $Ax \geq b$ entsprechen den jeweiligen Lieferverpflichtungen für die entsprechenden Sorten Schwer- Mittel- und Leichtöl.

Als weiteres Beispiel erstellen wir einen Produktionsplan für 4 Zeitperioden, mit einer Grundlast $\alpha \in \mathbb{R}$ und vier Zusatzproduktionsraten $x_1, x_2, x_3, x_4 \in \mathbb{R}$. Der zu deckende Bedarf in den 4 Zeitperioden sei

$$b_1 = 100, \quad b_2 = 125, \quad b_3 = 80, \quad b_4 = 90.$$

Nun ist es möglich über den Bedarf hinaus zu produzieren, und Ware zwischendurch in ein Lager zu stellen. Diese aufkummulierte Lagermenge $y_i$ am Ende einer jeweiligen Planungsperiode lautet dann:

$$y_i = \sum_{j \leq i} (\alpha + x_j - b_j).$$

Die Kosten je Einheit in Normalproduktion sei $\gamma = 80$, in Überproduktion seien die Produktionskosten

$$c_1 = c_2 = c_3 = c_4 = 100.$$

Die Lagerhaltungskosten um eine Mengeneinheit für einen Zeitperiode zu lagern sei

$$l_1 = l_2 = l_3 = 10, \quad l_4 = -50.$$

Die Modellierung von $l_4 = -50$ beinhaltet den Abverkauf der Überproduktion im Abschluss aller Zeitperioden mit einem Resterlös von 60 (etwa in einem Winterschlußverkauf für Textilien). Die Gesamtkosten lauten dann

$$\min ly + cx + 4\gamma\alpha.$$

Die Deckung des Bedarfs in den einzelnen Zeitperioden ist erfüllt durch

$$y \geq 0,$$

Ferner sind positive Produktionsmengen gefordert, $x \geq 0$, $\alpha \geq 0$. Die Konsistenzbedinung zwischen Bedarf, Produktion und Lagermenge ist gegeben durch

$$A \begin{pmatrix} \alpha \\ x \\ y \end{pmatrix} = b.$$

mit der Matrix $A \in \mathbb{R}^{4\times 9}$ wie folgt:

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

Somit haben wir die formulierung dieses Produktionskostenbeispiels als LP in Standardform. Wie sieht eine Lösung aus, und welches ist die Beste ?

## 2.4 Überblick

Zur Entscheidungsunterstützung bei der Lösung komplexer Entscheidungs- und Planungsprobleme spielt die mathematische Optimierung seit Jahrzehnten eine wichtige Rolle. Für die Praxis sind die Modellklassen Lineare Optimierung (LP) und gemischt-ganzzahlige Optimierung (IP) am wichtigsten, da es viele Anwendungen und hocheffiziente Standardsoftware gibt.

Typische Anwendungsfelder sind Chemische Industrie, Lagerhaltungs- und Transportprobleme, Planungsprobleme der Linienfluggesellschaften: Flugpläne, Aircraft Assignement, Crew Scheduling usf.. Kraftwerksoptimierungsprobleme sind ein weiteres Anwendungsgebiet.

Der Trend geht zu "Echtzeitoptimierungen" mit grafischer Visualisierung und hohen Interaktionsmöglichkeiten der Endbenutzer.

Ständige Leistungssteigerungen bei Rechnern und Software verbessern die Voraussetzungen für den praktischen Einsatz in betrieblichen Anwendungen. Vor allem bei der gemischt-ganzzahligen Optimierung wurden in der letzten Dekade enorme Fortschritte bei den Algorithmen erzielt.

Die Entwicklung von LP-Software ist komplex und erfordert spezifisches Wissen über Algorithmen, Datenstrukturen sowie Systementwicklung.

Die Lösung von LP-Modellen basiert auf der Simplexmethode bzw. Innere Punkte Verfahren, die ursprünglich auf die Arbeiten von G.B. Dantzig 1947 bzw. N. Karmakar 1984 zurückgehen. Beide Verfahren weisen unterschiedliche Vor- und Nachteile auf und ergänzen sich:

Im Normalfall werden LP-Modelle aus praktischen Anwendungen mit dem Simplexverfahren gelöst, in $Cm$ vielen Iterationen, wo $m$ die Anzahl Restriktionen, und $C$ eine Konstante ist, üblicherweise $2 \leq C \leq 5$.

Interior Point Codes können manche große LP-Modelle vielfach so schnell lösen, als die besten Simplex Codierungen, da die Anzahl der Iterationen unabhängig von der Modellgröße ist, typisch sind $20-40$ Iterationen. Der Arbeitsaufwand pro Iteration ist im wesentlichen durch das Lösen eines symmetrischen, positiv definiten Gleichungssystems bestimmt. Im Unterschied zum Simplexverfahren ist im Inneren Punkte Verfahren ein Iterationsschritt sehr aufwendig.

Die Simplexmethode hat den Vorteil, daß optimale Basislösungen bestimmt werden, d.h. nur wenige Strukturvariablen einer optimalen Lösung haben Werte ungleich Null.

Bei Interior Point Verfahren liegen die meisten Strukturvariablen zwischen ihren Schranken, was nur selten erwünscht ist. Mindestens für IP-Modelle muß zum Ausnutzen einer strengen LP-Relaxation und zur schnellen LP-Optimierung im Branch and Bound, eine optimale Basislösung bestimmt werden. Dazu stehen Algorithmen bereit, die als "Crossover" oder Basis Identifikation (BI) bezeichnet werden. Die Zeit zum Ausführen einer BI kann größer sein als die Zeit des LP-Problems.

Die Simplexmethode weist im Gegensatz zu Interior Point Methoden ein sehr gutes "Warmstart-Verhalten" auf, wenn ein LP-Modell nur wenig modifiziert wurde und eine nahezu optimale Basis vorhanden ist.

Nachfolgende Tabelle soll die Entwicklung der Software MOPS darstellen in Ihren Versionen 1.0 bis 4.0, anhand eines LP-Problems "Oil" mit ca. 5500 Restriktionen und 6000 Variablen.

| Jahr | Version | Zeit(Sec.) PII400 |
|---|---|---:|
| 1987 | V1.0 Produktform der Inversen | 612.4 |
| 1988 | V1.1 LU-Update | 107.5 |
| 1989 | V1.2 LP-Preprocessing | 35.1 |
| 1991 | V1.3 neue Zeilenauswahl | 31.2 |
| 1992 | V1.4 neue Skalierung | 26.0 |
| 1994 | V2.0 Verfeinerungen, LIFO-IP | 21.9 |
| 1995 | V2.5 Verfeinerungen allgemeine Knotenauswahl | 20.7 |
| 1997 | V3.0 Verfeinerungen, allgemeine duale Algorithmen | 17.6 |
| 1999 | V4.0 Simplex mit neuem LP-Preprocessing | 8.4 |
| 1999 | V4.0 Interior Point Code mit Crossover | 4.9 |

Einen wesentlichen Einfluß auf die Lösungszeit von LP- und IP-Modellen haben auch Techniken, die im LP-Preprocessing eingestzt werden. Sie bestehen aus einer Reihe einzelner Tests und Algorithmen, die zum Ziel haben, Restriktionen als redundant zu identifizieren, bzw. zu fixieren. Einzelne Schranken von Variablen werden relaxiert oder verschärft. Hauptgrund für das Preprocessing ist, daß große Modelle mit Computerprogrammen generiert werden, die keine Programmlöogik zur Redundanzerkennung aufweisen.

Nachfolgende Tabelle gibt einen Überblick zur Reduktion der Problemgröße durch LP-Preprocessing anhand von 6 Testbeispielen.

| Modell | Anzahl Restriktionen | Anzahl Variablen |
|---|---:|---:|
| Oil Orginal | 5593 | 6181 |
| Oil reduziert | 1617 | 2055 |
| Schering Orginal | 10246 | 29246 |
| Schering reduziert | 6062 | 16759 |
| Bayer Orginal | 33690 | 28122 |
| Bayer reduziert | 15345 | 9537 |
| B+M Orginal | 17431 | 81896 |
| B+M reduziert | 7859 | 61854 |
| tough Orginal | 217238 | 172158 |
| tough reduziert | 8633 | 99725 |
| Ken18 Orginal | 105127 | 154699 |
| Ken18 reduziert | 46887 | 96783 |

Man sieht, daß bei einigen Modellen die Problemgröse sich drastisch reduziert. Dies führt zu entsprechend kürzeren Lösungszeiten.

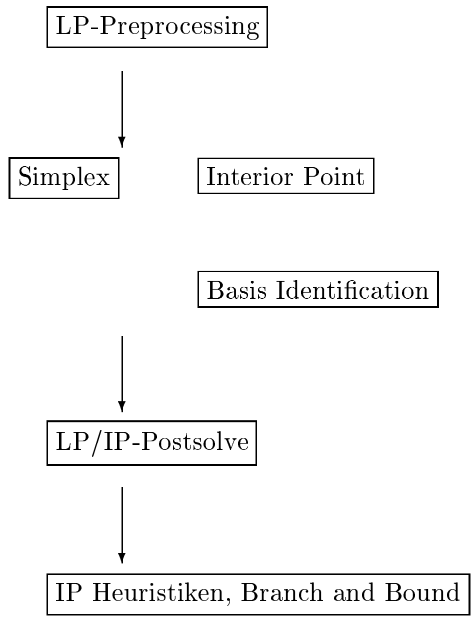
Die Lösungszeiten des Inneren Punkte Verfahrens, einschließlich BI (Basisidentifikation) ist in Nachfolgender Tabelle dem primalen Simplexverfahren gegenübergestellt.

| Modell | Interior Point + BI | | Simplex | |
|---|---|---:|---|---:|
| | Iterationen | Zeit(sec.) | Iterationen | Zeit(sec.) |
| Oil | 25 | 4.50 | 3236 | 8.84 |
| Schering | 42 | 32.24 | 7375 | 30.32 |
| Bayer | 35 | 41.30 | 10214 | 125.51 |
| B+M | 68 | 138.36 | 23070 | 226.29 |
| tough | 27 | 1161.24 | 72652 | 3501.94 |
| Ken18 | 20 | 575.29 | 126652 | 7331.01 |

Bei IP (Integer Programming) Modellen dürfen einige Variablen nur ganzzahlige Werte annehmen. IP Modellierungstechniken, u.a. über $0-1$ Variablen erlauben es nichtlineare deterministische Entscheidungsprobleme in einem IP-Modell abzubilden. gegenüber "KI-Techniken" bietet die Abbildung von Regeln der Aussagenlogik mit $0-1$-Variablen den Vorteil, daß sie direkt in komplexere Modelle mit kontinuierlichen Variablen eingebunden werden können.

Obwohl IP-Modelle formal eine sehr große Ähnlichkeit zu LP-Modellen aufweisen, gehören sie (bis auf spezielle Ausnahmen) mathematisch zur Klasse der NP-harten Probleme, für die keine effizienten Algorithmen bekannt sind. Alle erfolgreichen Softwaresysteme zur Lösung von IP Modellen basieren auf Branch and Bound Algorithmen, bei denen an jedem Knoten des Baumes das für das zugehörige Teilproblem, dessen LP-Relaxation gelöst wird. IP Software, die auf dem Konzept der strengen LP-Relaxation basiert, ist viel leistungsfähiger als klassische Branch and Bound Verfahren.

Die folgende Grafik zeigt den Normalfall einer Optimierung eines IP/LP Modells in einem Softwaresystem (MOPS).




In letzter Zeit sind in den Unternehmen viele interessante Anwendungen entstanden, bzw. in der Entwicklung, die auf LP/IP Technologie beruhen. Auch die APO-Komponente von SAP R3 zeigt, daß Optimierungstechnologie ein fester Bestandteil von ERP (Enterprise Ressource Planning) Software werden wird. Lineare Modelle LP und IP sind bis heute Standard in quantitativen Produktionsplanungs- und Supply Chain Modellen.

Die Erfahrung zeigt, daß mit IP Software schwierige IP Modelle durch verbesserte Algorithmen und schnellere Hardware mindestens um den Faktor 100 schneller gelöst werden können als vor 10 Jahren. Eine wesentliche Abschwächung dieses Trends ist nicht erkennbar.

Gegenstand des Linearen Optimierungsteils der Vorlesung ist der primale

Simplexalgorithmus. Dieser ist zentraler Bestandteil von LP Optimierungssoftware. Die lineare Optimierung ist eines der gebräuchlichsten und am weitesten Entwickleltен Optimierungsverfahren in der Anwendung. Die im zweiten Vorlesungsteil behandelten kombinatorischen Verfahren haben einen starken Bezug zur linearen Optimierung und können zum Teil als Spezialfälle der linearen Optimierung gesehen werden, obwohl diese Sichtweise den Vorgestellten Kombinatorischen Verfahren nicht ganz gerecht wird.

Auf die ganzzahlige Lineare Optimierung, sowie Innere Punkte Verfahren geht die Vorlesung nicht näher ein. Das Thema Komplexität bleibt im weiteren ebenfalls Ausgenommen. Wie erwähnt ist das Simplexverfahren nicht polynomial, im Gegensatz zu den Innere Punkte Verfahren, die polynomiale Laufzeit haben. Die ganzzahlige lineare Optimierung ist bis auf Spezialfälle (auf die wir im kombinatorischen Teil zu Sprechen kommen) nicht polynomial, (außer "P = NP").

## 2.5 Links zur linearen Optimierung

Wird später nachgereicht.

# 3 Dualität

## 3.1 Definition der Dualität

Zu einem LP existiert ein duales LP, mit weitreichenden Eigenschaften für das primale (orginäre) LP, die wir im folgenden untersuchen werden.

**Definition 5** *Gegeben sei folgendes primale LP in Standardform:*

$$(P) \qquad \begin{array}{ll} \min & cx \\ \textit{unter} & Ax = b, \quad x \geq 0. \end{array}$$

*Das hierzu duale LP hat die Darstellung:*

$$(D) \qquad \begin{array}{ll} \max & by \\ \textit{unter} & A^T y + s = c, \quad s \geq 0. \end{array}$$

*Mit $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^{1 \times n}$ gilt $y \in \mathbb{R}^m$ und $s \in \mathbb{R}^n$. Man nennt $y$ die dualen Variablen, und $s$ den dualen Schlupf.*

Für allgemeine LP's $(PA)$ aus Definition (4),

$$(PA) \qquad \begin{array}{ll} \min & cx \\ \text{unter} & A_e x = b_e, \\ & A_l x \geq b_l, \quad x \geq 0. \end{array}$$

mit Partitionen

$$e \,\dot{\cup}\, l = \{1, \ldots, m\}, \quad p \,\dot{\cup}\, u = \{1, \ldots, n\},$$

ist das duale LP mit $B := -A^T$ definiert durch:

$$(DA) \qquad \begin{array}{ll} \max & by \\ \text{unter} & B_u y = -c_u, \\ & B_p y \geq -c_p, \quad y_l \geq 0. \end{array}$$

Die Indices $i \in p$, für welche $x$ im primalen Problem $(PA)$ positiv ist, $x_p \geq 0$, werden zu Ungleichheitsbedinungen $B_p y \geq -c_p$. Die Indices $i \notin p$ werden zu Gleichheitsbedinungen $B_u y = -c_u$. Für die Indices $i \in l$, welche im primalen Problem $(PA)$ den Ungleichheitsbedinungen unterworfen sind $A_l x \geq b_l$ ist $y$ positiv, $y_l \geq 0$.

Die Koinzidenz zum Dualitätsbegriff obiger Definition sei als Übungsaufgabe gestellt. Die Rollen des primalen und dualen LP's sind untereinander nicht ausgezeichnet:

**Lemma 1** *Das primale LP (P) ist selbst dual zum dualen LP (D).*

Wir führen dies aus, in Bezug auf den Dualitätsbegriff aus Definition (5) für LP's in Standardform. Seien $(P)$, $(D)$ wie in (5). Für $y = y^+ - y^-$ mit $y^+, y^- \geq 0$ ist $(D)$ äquivalent zur Standardform:

$$\begin{array}{ll} \min & (-b^T, b^T)\binom{y^+}{y^-} \\ \text{unter} & (A^T, -A^T, E_n)\begin{pmatrix} y^+ \\ y^- \\ s \end{pmatrix} = c, \qquad y^+, y^-, s \geq 0. \end{array}$$

Hierauf die Dualisierungsregeln aus Definition (5) angewandt ergibt:

$$\begin{array}{ll} \max & cx \\ \text{unter} & \begin{pmatrix} A \\ -A \\ E_n \end{pmatrix} x + \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} = \begin{pmatrix} -b \\ b \\ 0 \end{pmatrix}, \qquad s_1, s_2, s_3 \geq 0. \end{array}$$

Aus $Ax \leq -b$ und $-Ax \leq b$ folgt $Ax = -b$. Mit $x + s_3 = 0$ und $s_3 \geq 0$ folgt $x \leq 0$. Ersetzen wir $x$ durch $-x$, so ergibt sich das orginäre primale LP $(P)$:

$$\min c(-x) \quad \text{unter} \quad A(-x) = b, \quad (-x) \geq 0. q.e.d.$$

Eine zulässige Lösung

$$(y, s) \in F_D = \{(y, s) \in \mathbb{R}^{m+n} \mid A^T y + s = c^T,\ s \geq 0\},$$

des dualen LP $(D)$ definiert eine untere Schranke für die Kostenfunktion des primalen LP's:

**Theorem 1 (schwache Dualität)** *Sei $p^* \in \mathbb{R}_\infty$,*

$$p^* = \min_{x \in F_P} cx,$$

*optimaler Kostenfunktionswert des primalen LP, und $d^* \in \mathbb{R}_\infty$,*

$$d^* = \max_{y \in F_D} by,$$

*optimaler Wert des dualen LP, dann gilt:*

$$p^* \geq d^*.$$

Wir können ohne Einschränkung annehmen, daß es einen primal zulässigen Punkt $x \in F_P = \{Ax = b,\ x \geq 0\}$ und einen dual zulässigen Punkt $(y, s) \in F_D$ gibt. Es folgt

$$cx = sx + y^T Ax = sx + yb \geq by,$$

da mit $s \geq 0$ und $x \geq 0$ gilt $sx \geq 0$. Damit folgt aber

$$p^* = \min_{x \in F_P} cx \geq \max_{(y,s) \in F_D} by = d^*. q.e.d.$$

Wir nennen ein $(x, y, s) \in F_P \times F_D$ ein zulässiges Paar, oder eine primal-dual zulässige Lösung. Gilt $sx = cx - by = 0$, dann ist $(x, y, s)$ optimal, d.h $x$ löst das primale LP optimal, während $(y, s)$ optimale Lösung des dualen LP's ist.

Der Wert $sx$ heißt Dualitätslücke. Diese beziffert die Abweichung eines zulässigen Paares $(x, y, s) \in F_P \times F_D$ von den optimalen Kosten.

## 3.2 starker Dualitätssatz

Um den starken Dualitätssatz zu beweisen benötigen wir zwei geometrische Lemmata.

**Lemma 2 (Trennungs Lemma)** *Sei $C \subset \mathbb{R}^n$ eine abgeschlossene konvexe Menge, und $b \notin C$ ein Punkt ausserhalb. Dann gibt es eine strikt trennende Hyperebene, gegeben durch ihren Normalenvektor $y \in \mathbb{R}^n$, sodaß mit einem $\gamma \in \mathbb{R}$ gilt:*

$$by < \gamma \quad \textit{und} \quad xy \geq \gamma \quad \textit{für alle} \quad x \in C.$$

Sei $n \in C$ der zu $b$ nächstgelegene Punkt in $C$ (einen solchen gibt es). Dann erfüllen $y = n - b$ und $\gamma = yn$ die getroffene Aussage. (Beweis als Übungsaufgabe).

**Lemma 3 (Lemma von Farkas)** *Für eine Matrix $B \in \mathbb{R}^{m \times n}$ und $b \in \mathbb{R}^m$ gilt genau einer der beiden folgenden Fälle, aber nicht beide:*

*i) es gibt ein $x \in \mathbb{R}^n$ mit $Bx = b$ und $x \geq 0$, oder*

*ii) es gibt ein $y \in \mathbb{R}^m$ mit $B^T y \geq 0$ und $by < 0$.*

Sei $x \geq 0$ mit $Bx = b$. Für alle $y \in \mathbb{R}^m$ mit $B^T y \geq 0$ gilt dann:

$$by = (Bx)^T y = x^T (B^T y) \geq 0.$$

Somit kann i) und ii) nicht zugleich gelten.

Die Menge $C = \{Bx \,|\, x \geq 0\}$ ist konvex und abgeschlossen. Sofern i) nicht gilt, ist $b \notin C$ ein Punkt ausserhalb, und wir können das Trennungs Lemma anwenden. Für ein $y \in \mathbb{R}^m$ und $\gamma \in \mathbb{R}$ gilt:

$$by < \gamma \quad \text{und} \quad (Bx)^T y = x^T (B^T y) \geq \gamma \quad \text{für alle} \quad x \geq 0.$$

Setzen wir $x = 0$ ein, so folgt $\gamma \leq 0$ und somit $by < 0$. Wäre eine Koordinate $(B^T y)_i$ kleiner Null, so folgt mit dem $\alpha$-fachen des $i$-ten Einheitsvektors $e_i \in \mathbb{R}^n$ ein Widerspruch

$$(\alpha e_i)^T (B^T y) = \alpha (B^T y)_i < \gamma,$$

für $\alpha \in \mathbb{R}^+$ groß genug. Es folgt daher $B^T y \geq 0$ und damit ii), sofern eben i) nicht gilt. q.e.d.

Sofern eine primal- und dual zulässige Lösung existieren $F_P \neq \emptyset$ und $F_D \neq \emptyset$ dann gilt im schwachen Dualitätssatz gleichheit, die Dualitätslücke wird geschlossen. Genauer gilt:

**Theorem 2 (Starker Dualitätssatz)** *Sei $(y^*, s^*) \in F_D$ eine optimale Lösung des dualen LP's. Dann hat auch das primale LP eine optimale Lösung $x^* \in F_P$. Die zugehörigen optimalen Kostenfunktionswerte sind gleich:*

$$p^* = cx^* = by^* = d^*.$$

Angenommen es würde $p^* > d^*$ gelten. Sei $B := \binom{c}{A} \in \mathbb{R}^{m+1\times n}$. Egal ob $F_P = \emptyset$ (und damit $p^* = +\infty$) oder nicht, das System

$$Bx = \binom{cx}{Ax} = \binom{d^*}{b}, \quad x \geq 0,$$

ist unlösbar und es gilt der Fall ii) in Farkas Lemma, es gibt ein $\binom{y_0}{y} \in \mathbb{R}^{m+1}$ mit

$$d^* y_0 + by < 0, \tag{1}$$

$$y_0 c^T + A^T y \geq 0. \tag{2}$$

Wir unterscheiden drei Fälle.

$$\begin{array}{llllll}
i) & y_0 < 0 : & & & \text{sei} & v := 2y^* + \frac{y}{y_0} \ : \\
(1) & \Longrightarrow & b\frac{y}{y_0} > -d^*, & \Longrightarrow & & bv = 2d^* + b\frac{y}{y_0} > d^*, \\
(2) & \Longrightarrow & A^T\frac{y}{y_0} \leq -c^T, & \Longrightarrow & & A^Tv = 2A^Ty^* + A^T\frac{y}{y_0} \leq c^T. \\
ii) & y_0 = 0 : & & & \text{sei} & v := y^* - y \ : \\
(1) & \Longrightarrow & b(-y) > 0, & \Longrightarrow & & bv = d^* + b(-y) > d^*, \\
(2) & \Longrightarrow & A^T(-y) \leq 0, & \Longrightarrow & & A^Tv = A^Ty^* + A^T(-y) \leq c^T. \\
iii) & y_0 > 0 : & & & \text{sei} & v := -\frac{y}{y_0} \ : \\
(1) & \Longrightarrow & d^* + b(-v) < 0, & \Longrightarrow & & bv > d^*, \\
(2) & \Longrightarrow & c^T + A^T(-v) \geq 0, & \Longrightarrow & & A^Tv \leq c^T.
\end{array}$$

Sei $s = c^T - A^T v$, so ist $s \geq 0$ und $(v, s) \in F_D$ dual zulässig. Zudem hat $(v, s)$ einen besseren Kostenfunktionswert als $y^*$, da $bv > d^*$, im Widerspruch zur Optimalität von $y^*$. Somit kann $p^* > d^*$ nicht gelten. Aufgrund des schwachen Dualitätssatzes gilt $p^* = d^*$. q.e.d.

Insbesondere hat das primale LP einen endlichen optimalen Kostenfunktionswert. Es gelten die drei folgenden Aussagen:

**Corollar 1**

*i) Hat entweder das primale LP $(P)$ eine optimale Lösung, oder das duale LP $(D)$, dann ist auch das Andere LP zulässig und beschränkt. Die optimalen Kostenfunktionswerte sind gleich, $p^* = d^*$.*

*ii) Ist $(P)$ zulässig, aber unbeschränkt, dann ist $(D)$ unzulässig und umgekehrt.*

*iii) Ist $(P)$ unzulässig, dann ist $(D)$ entweder ebenfalls unzulässig, oder unbeschränkt. Dasselbe gilt mit vertauschten Rollen.*

Es folgt $i$) aus dem starken Dualitätssatz, $ii$) folgt schon aus dem schwachen Dualitätssatz, und $iii$) ist direkte Folgerung von i). Da ja das primale LP das duale Problem des originär dualen LP's ist, ergeben sich die jeweiligen Aussagen mit vertauschten Rollen von $(P)$ und $(D)$. q.e.d.

Die Kombinationsmöglichkeiten unter primaler und dualer Zulässigkeit und beschränktheit sind im folgenden Diagramm nochmals zusammengestellt. Ein LP mit einer optimalen Lösung, das zulässig und beschränkt ist, wird optimal genannt.

| Primal-Duale Kombinationen | | Duales LP | | |
|---|---|---|---|---|
| | | Optimal | Unzulässig | Unbeschränkt |
| $PrimalesLP$ | Optimal | möglich | unmöglich | unmöglich |
| | Unzulässig | unmöglich | möglich | möglich |
| | Unbeschränkt | unmöglich | möglich | unmöglich |

# 4 Geometrie eines Linearen Programms

## 4.1 Basislösungen und Ecken eines LP

Sei $\min_{x \in F} z(x)$ ein LP in Standardform

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0, \end{array}$$

mit zulässiger Menge $F = \{Ax = b\,,\, x \geq 0\}$ und Kostenfunktion $z(x) = cx$. Für einen zulässigen Punkt $x \in F$ sei $B(x) = \{i \,|\, x_i \neq 0\}$ die Menge der Indices der Koordinaten ungleich 0 von $x$.

**Definition 6** *Eine zulässige Lösung $x \in F$ heißt Basislösung, genau dann wenn $A_i$ , $i \in B(x)$ linear unabhängige Spalten von $A$ sind.*

*Eine zulässige Lösung $x \in F$ heißt Ecke (von $F$), genau dann wenn $x$ nicht innerhalb der Verbindungsgeraden zweier Punkte $u, v \in F$ liegt:*

$$\textit{aus} \quad x = \lambda u + (1-\lambda)v, \quad \lambda \in [0,1] \quad \textit{folgt} \quad x = u \quad \textit{oder} \quad x = v.$$

Die geometrischen Eigenschaften der zulässigen Menge $F$ eines LP begründen wir im folgenden Satz:

**Theorem 3** *Sei $\min_{x \in F} z(x)$ ein LP in Standardform. Dann gelten*

*i) Eine zulässige Lösung $x \in F$ ist Basislösung, genau dann wenn $x$ Ecke von $F$ ist.*

*ii) Ist $x \in F$ keine Ecke, dann existiert eine $u \in F$, sodaß $u$ weniger Koordinaten $\neq 0$ hat als $x$.*

*iii) Ist die Kostenfunktion $z$ nach unten beschränkt, dann können wir zudem $z(u) \leq z(x)$ annehmen.*

*iv) Ist $F$ beschränkt, (und $x$ keine Ecke), dann existiert eine affine Kombination $x = \lambda u + (1-\lambda)v$ mit $u, v \in F$ und $\lambda \in (0,1)$, sodaß $u$ und $v$ weniger Koordinaten $\neq 0$ haben als $x$.*

Sei $F = \{Ax = b, \, x \geq 0\}$, sowie $z(x) = cx$ für $x \in F$.

Sei $x = \lambda u + (1-\lambda)v$ mit $\lambda \in (0,1)$ und $u, v \in F$ keine Ecke. Es gilt dann $B(u), B(v) \subset B(x)$ und folglich $B(u-v) \subset B(x)$. Die Spalten $A_i$ mit $x_i \neq 0$ sind linear abhängig, mit $u - v \neq 0$ gilt:

$$0 = A(u-v) = A_{B(x)}(u-v)_{B(x)}.$$

Sei $x$ keine Basislösung, die Spalten $A_i$, $i \in B(x)$ seien linear abhängig: $Ay = 0$ für ein $y \neq 0$ mit $B(y) \subset B(x)$. Die Punkte $x_t := x + ty$ erfüllen $Ax_t = b$, wegen $B(y) \subset B(x)$ gilt $x_\varepsilon \geq 0$ für $|\varepsilon|$ klein genug. Somit ist $x$ keine Ecke:

$$x = \frac{1}{2}x_{-\varepsilon} + \frac{1}{2}x_\varepsilon \quad \text{mit} \quad x_{-\varepsilon}, x_\varepsilon \in F.$$

Zu ii): Es gilt $x_t \in F$ genau dann wenn $x_i + ty_i \geq 0$ für alle $i = 1, \ldots, n$. Somit folgt:

$$\begin{aligned} y_i > 0 &\implies t \geq -\tfrac{x_i}{y_i}, \\ y_i < 0 &\implies t \leq -\tfrac{x_i}{y_i}. \end{aligned}$$

Somit sind durch

$$t_u := \max_{y_i > 0} -\frac{x_i}{y_i}, \quad t_o := \min_{y_i < 0} -\frac{x_i}{y_i},$$

Schranken $t_u, t_o \in \mathbb{R}_\infty$ definiert, mit $t_u < 0 < t_o$, sodaß gilt:

$$x_t \in F \iff t_u \leq t \leq t_o.$$

Sei $t_u \in \mathbb{R}$, ( wegen $y \neq 0$ dürfen wir $y_i > 0$ annehmen für ein $i \in B(x)$). Für $i \in B(x)$ mit $t_u = -\frac{x_i}{y_i}$ ist dann $(x_{t_u})_i = 0$. Es hat $u := x_{t_u} \in F$ wenigstens eine Koordinate weniger $\neq 0$ als $x$.

Zu iii): Sei $z : F \to \mathbb{R}$ nach unten beschränkt. Im Fall $t_o = \infty$ gilt:

$$lim_{t \to +\infty} z(x_t) = z(x) + tcy \neq -\infty.$$

Und folglich $cy \geq 0$. Im Fall $t_o \in \mathbb{R}$ vertauschen wir eventuell $y$ durch $-y$, somit gilt $cy \geq 0$ in jedem Fall. Für $u = x_{t_u}$ folgt dann

$$z(u) = z(x) + t_u cy \leq z(x).$$

Zu iv): Ist $F$ beschränkt, so gilt $t_u, t_o \in \mathbb{R}$ und $x$ liegt auf der Verbindungsgeraden der beiden Punkte $x_{t_u}, x_{t_o} \in F$, mit weniger Koordinaten $\neq 0$ als $x$. q.e.d.

Somit ist der Begriff Basislösung unabhängig von der Darstellung eines LP, zudem folgt die Existenz von Basislösungen bzw. optimalen Basislösungen.

**Corollar 2 (Hauptsatz der linearen Programmierung)** *Sei* $\min_{x \in F} z(x)$ *ein zulässiges LP, d.h.* $F \neq \emptyset$, *dann enthält* $F$ *eine Basislösung. Hat das LP eine optimale Lösung, (die Kostenfunktion* $z$ *ist nach unten beschränkt), dann hat das LP auch eine optimale Basislösung.*

Dies folgt unmittelbar aus ii),iii), zusammen mit i) im obigen Satz.

## 4.2 Zulässige Menge eines LP als Polyeder

**Definition 7** *Ein Polyeder* $F \subset \mathbb{R}^n$ *ist die konvexe Hülle einer endlichen Menge* $\mathcal{B} \subset \mathbb{R}^n$ *von Punkten:*

$$F = conv(\mathcal{B}).$$

*Mit* $\mathcal{B} = \{b_1, \ldots, b_r\}$ *gilt* $F = \{\sum_i \lambda_i b_i \mid \sum_i \lambda_i = 1\,,\, \lambda_i \geq 0\}$.

Hierbei ist ein $b \in \mathcal{B}$ notwendig, genau dann wenn $b$ eine Ecke von $F$ ist: Sei $b_1 = \lambda u + (1 - \lambda)v$ keine Ecke, $u = \sum_i u_i b_i$, $v = \sum_i v_i b_i$. Dann gilt

$$b_1 = c_1 b_1 + \sum_{i \neq 1} c_i b_i \quad \text{mit} \quad c_1 := (\lambda u_1 + (1 - \lambda)v_1).$$

Hierbei gilt $c_1 < 1$, aus $c_1 = 1$ würde mit $u_1 = 1$ sonst $b_1 = u$ folgen. Sodann ist $b_1$ nicht notwendig:

$$b_1 = \sum_{i \neq 1} \frac{c_i}{1 - c_1} b_i.$$

Sei $b_1$ nicht notwendig:

$$b_1 = \sum_{i=2}^{r} \alpha_i b_i = \alpha_2 b_2 + (1 - \alpha_2) \sum_{i \geq 3} \frac{\alpha_i}{1 - \alpha_2} b_i.$$

Es ist $b_1 \neq b_2$ und daher $\alpha_2 \neq 1$, mit $r$ minimal ist auch $b_1 \neq \sum_{i \geq 3} \frac{\alpha_i}{1-\alpha_2} b_i$ und somit $b_1$ keine Ecke. q.e.d.

Sei $F$ eine zulässige Menge eines LP. Eine Ecke $x \in F$ ist mit $B = B(x)$ eindeutig bestimmt, durch $A_B x_B = b$, da $A_B$ injektiv ist (es sind die Spalten linear unabhängig).

**Theorem 4** *Die zulässige Menge $F$ eines LP hat nur endlich viele Ecken ! Ferner sind äquivalent:*

*i) $F$ ist beschränkt,*

*ii) $F$ ist Polyeder,*

*iii) $F = conv(\mathcal{B})$ mit $\mathcal{B} \subset F$ ist die Menge der Ecken von $F$.*

Die Folgerung $iii) \implies ii) \implies i)$ liegt auf der Hand. Es gilt $\mathrm{conv}(\mathcal{B}) \subset F$, zu $i) \implies iii)$ ist nurmehr die Existenz eines $x \in F \setminus \mathrm{conv}(\mathcal{B})$ auszuschließen:

Sei $x \in F \setminus \mathrm{conv}(\mathcal{B})$ mit minimal vielen Koordinaten ungleich Null. Es ist $x$ keine Ecke (sonst $x \in \mathcal{B}$ !), mit $iv)$ aus Satz (3) folgt $x = \lambda u + (1-\lambda)v$ sodaß $u, v$ weniger Koordinaten $\neq 0$ haben. Daher folgt aber $u, v \in \mathrm{conv}(\mathcal{B})$ und somit der Widerspruch $x \in \mathrm{conv}(\mathcal{B})$. q.e.d.

## 4.3 reduzierte Darstellung und zulässige Basen

Sind in der Darstellung $(A, b, c)$ mit $F = \{Ax = b\,,\, x \geq 0\}$ und $z(x) = cx$ eines LP in Standardform $\min_{x \in F} z(x)$ die Zeilen von $A$ linear unabhängig, dann spricht man von einer reduzierten Darstellung. Eine solche existiert stets.

In dieser gilt für $A \in R^{m \times n}$ dann $m = \mathrm{Rang} A \leq n$. Für linear unabhängige Spalten $A_i$, $i \in I$ gibt es eine Teilmenge $B \subset \{1, \ldots, n\}$ mit $I \subset B$ und $A_B \in \mathbb{R}^{m \times m}$ regulär (Basisergänzungssatz).

**Definition 8** *In dieser Situation heißt $B \subset \{1, \ldots, n\}$ zulässige Basis genau dann wenn $A_B \in R^{m \times m}$ regulär ist und $A_B^{-1} b \geq 0$ gilt.*

Dieser Begriff ist abhängig von einer Darstellung $(A, b, c)$ des LP $\min_{x \in F} z(x)$, jedoch nicht innerhalb einer Äquivalenzklasse $(A, b, c)_\sim$ von Darstellungen.

Die zulässigen Basen einer reduzierten Darstellung stehen zu den Basislösungen in direktem Zusammenhang.

$$\begin{array}{lclcl} \phi & : & B \text{ zulässige Basis} & \mapsto & x = (x_B, x_N) \text{ mit } x_B = A_B^{-1} b \text{ und } x_N = 0. \\ * & : & x \text{ Basislösung} & \mapsto & B \supset B(x) \text{ sodaß } A_B \text{ regulär ist.} \end{array}$$

Es ist möglich, daß $\phi$ zwei oder mehr zulässige Basen auf dieselbe Basislösung abbildet (es ist dann $\phi$ nicht injektiv und $*$ nicht eindeutig definiert), man spricht dann von einem entarteten Darstellung des LP. Wir bezeichnen die Basislösung $\phi(B)$ einer zulässigen Basis mit $x_{(B)} := \phi(B)$. Wir notieren:

$$x_{(B)} = (x_B, x_N) \quad \text{mit} \quad x_N = 0.$$

Der Begriff Basislösung ist Darstellungsunabhängig, er hängt nur von der zulässigen Menge $F$ eines LP ab. Der Begriff zulässige Basis hingegen hängt von einer konkreten Darstellung $(A, b, c)$ eines LP ab, er ist jedoch invariant unter äquivalenten Darstellungen. Da der Pivotschritt des Simplexalgorithmus nur den Wechsel zu äquivalenten Darstellungen beinhaltet können wir in der Diskussion des Simplexalgorithmus im nächsten Abschnitt den Basisbegriff allgemein verwenden.

# 5 Simplex Algorithmus

Wir beschreiben den Simplexalgorithmus für ein LP in Standardform $\min_{x\in F} z(x)$ in einer reduzierten Darstellung $(A, b, c)$:

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0, \end{array}$$

Mit $A \in \mathbb{R}^{n\times m}$ und $\mathrm{Rang}(A) = m$, sowie $b \in \mathbb{R}^m$, $c \in \mathbb{R}^{1\times n}$ gilt also

$$F = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\} \quad \text{und} \quad z(x) = cx.$$

Ob das LP zulässig ist $F \neq \emptyset$, oder unbeschränkt, $\min_{x\in F} z(x) = -\infty$, steht a priori offen. Eine entsprechende Entscheidung liefert der Simplexalgorithmus. Wir erwarten daß er in genau einem der drei folgenden Fälle terminiert:

i) $F = \emptyset$, das LP ist unzulässig,

ii) $F \neq \emptyset$, aber $z(x), x \in F$ ist nach unten unbeschränkt,

iii) Rückgabe eines optimalen $x^* \in F$ mit $z(x^*) = \min_{x\in F} z(x)$.

## 5.1 reduzierte Kosten

Sei $B \subset \{1, \ldots, n\}$ zulässige Basis und $z(x) = cx$ die Kostenfunktion des LP. Ein zulässiger Punkt $x \in F$ in der Zerlegung $x = (x_B, x_N)$ ist wegen $Ax = A_B x_B + A_N x_N = b$ schon durch $x_N$ eindeutig bestimmt:

$$x_B = A_B^{-1}(b - A_N x_N). \tag{3}$$

Die Kosten sind folglich eine Funktion der Nichtbasisvariablen $x_N$ :

$$\begin{aligned} z(x) &= c_B x_B + c_N x_N \\ &= c_B A_B^{-1} b + (c_N - c_B A_B^{-1} A_N) x_N \end{aligned} \tag{4}$$

Mit $y := c_B A_B^{-1} \in \mathbb{R}^{1\times m}$ hat der Zeilenvektor $s := c - yA \in \mathbb{R}^{1\times n}$ die Zerlegung $s = (0, s_N)$:

$$\begin{aligned} s_B &= c_B - (c_B A_B^{-1}) A_B = 0, \\ s_N &= c_N - (c_B A_B^{-1}) A_N. \end{aligned} \tag{5}$$

Somit können wir in (4) einsetzen:

$$\begin{aligned} z(x) &= yb + s_N x_N, \\ &= yb + sx. \end{aligned} \tag{6}$$

Die Kostenfunktion $z'(x) := sx$ ist äquivalent zur originären Kostenfunktion:

$$z'(x) = z(x) - yb.$$

In Bezug auf die zulässige Basis $B$ gilt hierbei $s = (s_B, s_N)$ mit $s_B = 0$.

**Definition 9** *Sei $B$ zulässige Basis, Gilt für die Kostenfunktion*

$$z(x) = cx = c_B x_B + c_N x_N \quad \textit{hierbei} \quad c_B = 0,$$

*so nennen wir $c$ reduzierten Kostenvektor zur Basis $B$.*

Wir können in Bezug auf eine Basis $B$ stets von einer Kostenfunktion mit reduziertem Kostenvektor ausgehen, $z(x) = cx$ mit $c_B = 0$. Die Berechnung der reduzierten Kosten $c = (0, c_N)$ zur Basis $B$ aus beliebigen Kosten $z(x) = c'x$ ist mit (5) gegeben. Nur die Nichtbasisvariablen $x_N$ gehen in die reduzierten Kosten ein:

$$z(x) = c_N x_N.$$

## 5.2 Optimalitätskriterium

Mit den reduzierten Kosten ist folgendes Optimalitätskriterium für eine Basislösung anwendbar:

**Corollar 3 (Optimalitätskriterium)** *Sei $B$ eine zulässige Basis mit Basislösung $x_{(B)}$. Die Kostenfunktion $z(x) = c_N x_N$ sei durch einen reduzierten Kostenvektor gegeben, $c_B = 0$.*

$$\textit{Gilt } c_N \geq 0, \textit{ dann ist } x_{(B)} \textit{ eine optimale Lösung des LP.}$$

*Für jede andere zulässige Lösung $x \in F$ gilt dann*

$$z(x_{(B)}) \leq z(x).$$

Sei $x = (x_B, x_N)$ und $x_{(B)} = (x_B^*, x_N^*)$. Wegen $c_B = 0$, $x_N^* = 0$ sowie $c_N, x_N \geq 0$ gilt:

$$z(x_{(B)}) = c_B x_B^* + c_N x_N^* = 0 \leq c_N x_N = z(x). \quad \textit{q.e.d.}$$

Für $y, s$ aus (5.1) erfüllt das primal-duale Paar $(x_{(B)}, y^T, s^T)$ die Bedinung $A^T y^T + s^T = c^T$. Mit $x_{(B),N} = 0$ und $s_B = 0$ ist die Dualitätslücke geschlossen: $sx_{(B)} = 0$. Die duale Zulässigkeit $s \geq 0$ ist also genau mit dem Optimalitätskriterium erfüllt.

Somit können wir den Simplexalgorithmus wie folgt interpretieren: Die Dualitätslücke $sx = 0$ ist stets geschlossen. Es wird die duale Zulässigkeit der Lösung $(x, y, s)$ angestrebt. Mit dem Optimalitätskriterium ist dann ebenso die duale Zulässigkeit $s = (0, s_N)$ mit $s_N \geq 0$ erfüllt. Mit dem starken Dualitätssatz ist folglich $x_{(B)}$ optimale Lösung des primalen Problems.

Durch den Wechsel auf die jeweils reduzierten Kostenvektoren ändert der Simplexalgotithmus allerdings das duale Problem beständig.

## 5.3 Prinzip des Simplexalgorithmus

Ausgehend von einer zulässigen Basis $B$ mit Basislösung $x_{(B)}$ suchen wir eine neue Basis $B^+$ mit besserer Basislösung:

$$z(x_{(B^+)}) \leq z(x_{(B)}).$$

Hierzu wählen wir eine Suchrichtung $y \in \mathbb{R}^n$ sodaß die Variationslösung

$$x_t := x_{(B)} - ty \qquad \text{für} \quad t \in \mathbb{R}_{\geq 0},$$

die Kosten verbessert. Entweder ist $x_t$ zulässig für alle $t \in \mathbb{R}_{\geq 0}$ und die Kostenfunktion $z : F \to \mathbb{R}$ ist nach unten unbeschränkt, oder es gibt eine optimale Schrittweite $t^+ \in \mathbb{R}_{\geq 0}$, sodaß $x_{t^+}$ gerade noch zulässig ist, mit den besten Kosten $z(x_{t^+})$ unter allen Variationslösungen.

In diesem Fall ist $x_{t^+} = x_{(B^+)}$ wieder Basislösung zu einer neuen Basis $B^+$.

Die Wahl der Suchrichtung $y \in \mathbb{R}^n$ und Schrittweite $t^+$ sodaß $x_{t^+} \in F$ zulässig bleibt, und zudem Basislösung einer neuen Basis $B^+$ wird - sodaß sich außerdem die Kostenfunktion verbessert $z(x_{(B^+)}) \leq z(x_{(B)})$ - beschreiben wir im nächsten Abschnitt.

## 5.4 Der Pivotschritt des Simplexalgorithmus

Sei $B$ zulässige Basis mit Basislösung $x = x_{(B)}$. Der Pivotschritt vollzieht diesen Basiswechsel mittels einer reduzierten Darstellung $(A, b, c)$, deren Kostenfunktion $z(x) = cx$ durch einen reduzierten Kostenvektor $c = (c_B, c_N)$ mit $c_B = 0$ gegeben ist.

Sei nun das Optimalitätskriterium nicht erfüllt, $c_j < 0$ für ein Nichtbasisindex $j \notin B$. Die Suchrichtung

$$y = (y_B, y_N) \quad \text{mit} \quad y_N = -e_j \quad \text{und} \quad A_j = A_B y_B,$$

erfüllt die homogene Gleichung $Ay = 0$.

Die Punkte $x_t := x - ty$ erfüllen somit $Ax_t = b$ und sind zulässig, $x_t \in F$, solange $x_t \geq 0$ gilt:

$$x_i \geq ty_i \quad \text{für alle} \quad i \in \{1, \ldots, n\}.$$

Wir wählen $t^+$ maximal mit $x_t \geq 0$ für alle $t$ im Intervall $[0, t^+]$ . Dies wird erfüllt durch

$$t^+ := \min_{y_i > 0} \frac{x_i}{y_i}.$$

Denn es gilt:

$$\begin{aligned}
&y_i \leq 0 \quad \text{und} \quad t \geq 0 && \Longrightarrow \quad x_i - ty_i \geq 0, \\
&y_i > 0 \quad \text{und} \quad t \leq t^+ && \Longrightarrow \quad x_i - ty_i \geq x_i - t^+ y_i \geq 0 \qquad \text{wegen} \quad t^+ \leq \tfrac{x_i}{y_i}.
\end{aligned}$$

vgl. Beweis zu Satz (3). Ist $y_B \leq 0$, und somit $t^+ = \infty$, so sind die Kosten nach unten unbeschränkt. Mit $c_B = 0$ und $y_N = -e_j$ gilt $-tcy = tc_j$, daher folgt:

$$\lim_{t \to +\infty} z(x_t) = z(x) + tc_j = -\infty,$$

(wegen $c_j < 0$). Andernfalls existiert ein Index $l$ mit

$$t^+ = \frac{x_l}{y_l}.$$

Wegen $y_l > 0$ gilt $l \in B$, zudem ist die Spalte $A_l$ abhängig von $\{A_j, A_k,\, k \in B, k \neq l\}$. Somit ist für $B^+ = B \setminus l \cup j$ die Teilmatrix $A_{B^+}$ regulär.

Wegen $B(x_{t^+}) \subset B^+$ gilt $A_{B^+}^{-1} b = x_{t^+} \geq 0$. Somit ist

$$x^+ := x_{t^+},$$

neue Basislösung mit neuer zulässiger Basis $B^+$. Der Zielfunktionswert hat sich verbessert:

$$z(x^+) = z(x) + t^+ c_j \leq z(x).$$

Er hat sich echt verbessert, sofern $t^+ > 0$ gilt, d.h. wenn aus $y_i > 0$ folgt $x_i > 0$. Dies ist der Fall, bei einem nicht entarteten LP. (warum ?)

**Theorem 5** *Sei $B$ zulässige Basis mit Basislösung $x \in F$, sowie $(A, b, c)$ eine reduzierte Darstellung mit reduziertem Kostenvektor, $c_B = 0$. Im zugehörigen Pivotschritt führt der Simplexalgorithmus den erstmöglichen der folgenden drei Schritte aus:*

1. *überprüfen des Optimalitätskriteriums, im Fall $c_N \geq 0$ stoppt der Algorithmus mit der Optimalen Lösung $x$.*

2. *Feststellen einer nach unten unbeschränkten Kostenfunktion im Fall $y_B \leq 0$, (mit dem von $j \notin B$ mit $c_j < 0$ abhängigen $y_B$ ) Der Algorithmus stoppt mit der Auskunft einer unbeschränkten Kostenfunktion.*

3. *Wechsel zu einer neuen zulässigen Basis $B^+$ mit Basislösung $x^+$ mit besserem (oder zumindest gleichem) Kostenfunktionswert*

$$z(x^+) \leq z(x).$$

*Hiernach wiederholt sich der Pivotschritt.*

*Mindestens einer dieser drei Schritte ist ausführbar.*

Gesetzt den Fall wir verfügen über eine zulässige Basis $B$ und Basislösung $x = (x_B, x_N)$ mit $x_B = A_B^{-1} b$, $x_N = 0$, sowie eine Darstellung $(A, b, c)$ mit reduziertem Kostenvektor, $c_B = 0$. Dann müssen wir die folgenden Aufgaben leisten:

1. Feststellen ob $c_N \geq 0$, andernfalls die Auswahl eines Index $j \notin B$ mit $c_j < 0$, der in die neue Basis eintritt.

2. Bestimmen einer Suchrichtung $y$ mit $y_N = -e_j$ und $A_j = A_B y_B$.

3. Feststellen ob $y_B \leq 0$, andernfalls wahl eines solchen Index $l \in B$ mit
$$t^+ = \frac{x_l}{y_l} = \min\{\frac{x_i}{y_i} \,|\, i \in B \text{ und } y_i > 0\},$$
der die Basis verlässt.

4. Bestimmung der neuen Basis $B^+ = B \setminus l \cup j$ und neuen Basislösung $x^+ = x - t^+ y$.

5. Auffinden einer neuen äquivalenten Darstellung $(A^+, b^+, c^+)$ passend zu $B^+$ mit reduziertem Kostenvektor $c^+_{B^+} = 0$, sodaß weiterhin alle diese Pivotschritte leicht ausgeführt werden können.

Um diese Schritte in übersichtlicher Form ausführen zu können, führen wir das Simplextableau ein.

## 5.5 Simplex Tableau einer zulässigen Basis

Sei $\min_{x \in F} z(x)$ ein LP in Standardform mit reduzierter Darstellung $(A, b, c)$. Mit $A \in \mathbb{R}^{m \times n}$ , $b \in \mathbb{R}^m$ , $c \in \mathbb{R}^{1 \times n}$ lautet das LP:
$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0. \end{array}$$
Mit einem $d \in \mathbb{R}$ ordnen wir dieser Darstellung Tableau $T \in \mathbb{R}^{m+1 \times n+2}$ zu, der Gestalt:
$$T = \begin{pmatrix} 1 & c & -d \\ 0 & A & b \end{pmatrix}.$$
Die erste Zeile und Spalte von $T$ werden als 0-te Zeile, bzw. 0-te Spalte bezeichnet. Der Eintrag $d \in \mathbb{R}$ ist mit $z(x) = cx + d$ durch die Kostenfunktion bestimmt. Gilt $z(x) = cx$, so setzen wir $d = 0$.

Für eine zulässige Basis $B \subset \{1, \ldots, n\}$ suchen wir eine Basisdarstellung $(A, b, c)$ mit zugehörigem Basistableau
$$T_B = \begin{pmatrix} 1 & c & -d \\ 0 & A & b \end{pmatrix},$$
mit folgendem Bezug zur Basis $B$:

i) Es ist $c$ reduzierter Kostenvektor zur Basis $B$, es gilt $c = (c_B, c_N)$ mit $c_B = 0$.

ii) Es ist $A_B$ eine Permutationsmatrix. In Bezug auf eine Bijektion $\pi : B \to \{1, \ldots, m\}$ ist für $i \in B$ der Spaltenvektor $A_i = e_{\pi(i)}$ der $\pi(i)$-te Einheitsvektor. Mit $B = \{i_1, \ldots, i_m\}$ gilt:
$$A_B = \left(e_{\pi(i_1)}, \ldots, e_{\pi(i_m)}\right).$$

**Definition 10** *Sei $(A, b, c)$ eine reduzierte Darstellung eines LP in Standardform. Ein Tablaeu*

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

*nennen wir Basistablau zur Basis $B \subset \{1, \ldots, n\}$, falls $c$ ein reduzierter Kostenvektor zur Basis $B$ ist, und die Teilmatrix $A_B$ eine Permutationsmatrix ist, d.h. wenn i) und ii) wie oben gelten.*

**Corollar 4** *In der Situation obiger Definition existiert zu jeder Basis $B$ ein solches Basistableau $T_B$.*

Sei $(\tilde{A}, \tilde{b}, \tilde{c})$ eine beliebige (reduzierte) Darstellung, und $B$ eine zulässige Basis. Dann ist

$$T_B = \begin{pmatrix} 1 & \tilde{c}_B \\ 0 & \tilde{A}_B \end{pmatrix}^{-1} \begin{pmatrix} 1 & \tilde{c} & -\tilde{d} \\ 0 & \tilde{A} & \tilde{b} \end{pmatrix} = \begin{pmatrix} 1 & \tilde{c} - \tilde{c}_B \tilde{A}_B^{-1} \tilde{A} & -\tilde{d} - \tilde{c}_B \tilde{A}_B^{-1} \tilde{b} \\ 0 & \tilde{A}_B^{-1} \tilde{A} & \tilde{A}_B^{-1} \tilde{b} \end{pmatrix}, \tag{7}$$

ein Tableau zur äquivalenten Darstellung $(A, b, c)$, die i) und ii) erfüllt in Bezug auf die Basis $B$:

$$\begin{aligned} A &= \tilde{A}_B^{-1} \tilde{A}, \\ b &= \tilde{A}_B^{-1} \tilde{b}, \\ c &= \tilde{c} - \tilde{c}_B \tilde{A}_B^{-1} \tilde{A}. \end{aligned}$$

Es ist $c$ reduzierter Kostenvektor, $c_B = 0$, und $A_B = E_m$ ist sogar die Einheitsmatrix des $\mathbb{R}^{m \times m}$. q.e.d.

Wir identifizieren die zugehörige Basislösung $x = x_{(B)}$ im Tableau $T_B$. Für $l \in B$ sei $k = \pi(l)$, d.h. $A_l = e_k$. Für $i \in B, i \neq l$ gilt dann $a_{ki} = 0$ und somit:

$$b_k = (A_B x_B)_k = \sum_{i \in B} a_{ki} x_i = a_{kl} x_l = x_l. \tag{8}$$

Für $c_j < 0$ identifizieren wir die Suchrichtung $y = (y_B, y_N)$ mit $y_N = -e_j$, welche den Index $j$ in die Basis bringt. Mit $A_j = A_B y_B$ folgt für $l \in B$ und $k = \pi(l)$:

$$a_{kj} = (A_B y_B)_k = \sum_{i \in B} a_{ki} y_i = a_{kl} y_l = y_l.$$

Es haben $y_B$ und $A_j$ dieselben Koordinaten. Die Kostenfunktion ist nach unten unbeschränkt, $y_B \leq 0$, genau dann wenn $A_j \leq 0$ gilt. Andernfalls, berechnet sich die optimale Schrittweite $t^+$ aus dem Tableau $T_B$ wie folgt:

$$\begin{aligned} t^+ &= \tfrac{x_l}{y_l} &&= \min\{\tfrac{x_i}{y_i} \mid i \in B \text{ und } y_i > 0\} \\ &= \tfrac{b_{\pi(l)}}{a_{\pi(l)j}} &&= \min\{\tfrac{b_{\pi(i)}}{a_{\pi(i)j}} \mid i \in B \text{ und } a_{\pi(i)j} > 0\} \\ &= \tfrac{b_k}{a_{kj}} &&= \min\{\tfrac{b_{\tilde{i}}}{a_{\tilde{i}j}} \mid \tilde{i} \in \{1, \ldots, m\} \text{ und } a_{ij} > 0\} \end{aligned}$$

**Corollar 5** *Wir nennen das Element $a_{kj}$, mit welchem die obige Minimumbedinung erfüllt ist, das Pivotelement des Tableaus $T_B$.*

*Die Regeln zur Bestimmung des Pivotelements lauten wie folgt: Es ist $j \in \{1,\ldots,n\}$ ein Spaltenindex mit $c_j < 0$, und $k \in \{1,\ldots,m\}$ ein Zeilenindex im Tableau mit*

$$\frac{b_k}{a_{kj}} = min\{\frac{b_i}{a_{ij}} \,|\, i \in \{1,\ldots,m\} \text{ und } a_{ij} > 0\}.$$

Die Spalten $l \in B$ mit $A_l = e_k$ und $j \notin B$ sind durch das Pivotelement $a_{kj}$ ausgezeichnet. Ausführlich lautet das Tableau $T_B$ mit diesen Spalten wie folgt:

$$T_B := \begin{array}{c} \begin{array}{cccccccc} \phantom{1} & \phantom{c_1} & & l & & j & & \end{array} \\ \begin{pmatrix} 1 & c_1 & \cdots & 0 & \cdots & c_j & \cdots & c_n & -d \\ 0 & a_{1\,1} & \cdots & 0 & \cdots & a_{1\,j} & \cdots & a_{1\,n} & b_1 \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & a_{k\,1} & \cdots & 1 & \cdots & a_{k\,j} & \cdots & a_{k\,n} & b_k \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & a_{m\,1} & \cdots & 0 & \cdots & a_{m\,j} & \cdots & a_{m\,n} & b_m \end{pmatrix} \end{array}.$$

Um das Tableau, und die damit verbundene Darstellung, des LP passend zur neuen Basis $B^+ = B \setminus l \cup j$ zu erhalten, mit zu $B^+$ reduziertem Kostenvektor $c^+$ und Permutationsmatrix $A^+_{B^+}$, führen wir folgende Zeilenoperationen auf $T_B$ durch:

$$\begin{array}{l} \text{das } -\frac{c_j}{a_{k\,j}}\text{-fache der } k\text{-ten Zeile wird zur 0-ten Zeile addiert,} \\ \text{das } -\frac{a_{i\,j}}{a_{k\,j}}\text{-fache der } k\text{-ten Zeile wird zur } i\text{-ten Zeile addiert, für } i \neq k, 0, \\ \text{die } k\text{-te Zeile wird durch } a_{k\,j} \text{ dividiert.} \end{array} \tag{9}$$

Dies entspricht einer Linksmultiplikation des Tableau $T_B$ mit

$$\left(\begin{array}{c|c} 1 & \delta \\ \hline 0 & D \end{array}\right) = \left(\begin{array}{c|cccccc} 1 & 0 & \cdots & -\frac{c_j}{a_{k\,j}} & \cdots & 0 \\ \hline \vdots & 1 & \cdots & -\frac{a_{1\,j}}{a_{k\,j}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & \frac{1}{a_{k\,j}} & & \vdots \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{a_{m\,j}}{a_{k\,j}} & \cdots & 1 \end{array}\right).$$

Es ist $\delta = (0, \ldots, -\frac{c_j}{a_{kj}}, \ldots, 0)$ und

$$D = \begin{pmatrix} 1 & \cdots & -\frac{a_{1\,j}}{a_{k\,j}} & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots \\ \vdots & & \frac{1}{a_{k\,j}} & & \vdots \\ \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & -\frac{a_{m\,j}}{a_{k\,j}} & \cdots & 1 \end{pmatrix}.$$

Somit erhalten wir das Tableau

$$T_{B^+} = \begin{pmatrix} 1 & \delta \\ 0 & D \end{pmatrix} T_B.$$

Unter Aufführung der interessanten Spalten $l, j$ erhalten wir im Detail:

$$T_{B^+} := \begin{array}{c} \begin{array}{ccccccccc} \phantom{1} & \phantom{c_1^+} & \phantom{\cdots} & l & \phantom{\cdots} & j & \phantom{\cdots} & \phantom{c_m^+} & \phantom{-d^+} \end{array} \\ \begin{pmatrix} 1 & c_1^+ & \cdots & -\frac{c_j}{a_{kj}} & \cdots & 0 & \cdots & c_m^+ & -d^+ \\ 0 & a_1^+ & \cdots & -\frac{a_{1\,j}}{a_{k\,j}} & \cdots & 0 & \cdots & a_{1\,n}^+ & b_1^+ \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & a_{k\,1}^+ & \cdots & \frac{1}{a_{k\,j}} & \cdots & 1 & \cdots & a_{k\,n}^+ & b_k^+ \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & a_{m\,1}^+ & \cdots & -\frac{a_{m\,j}}{a_{k\,j}} & \cdots & 0 & \cdots & a_{m\,n}^+ & b_m^+ \end{pmatrix} \end{array}.$$

Die Umformungen entsprechen einer Überführung des LP zu einer äquivalenten Darstellung $(A^+, b^+, c^+)$ mittels einer regulären Matrix $D$ (und Zeilenvektor $\delta$):

$$A^+ = DA, \quad b^+ = Db, \quad c^+ = c + \delta A, \quad d^+ = d - \delta b.$$

Für eine in der Basis verbleibenden Index $i \in B$, $i \neq l$ nutzen wir aus $a_{ki} = 0$. Somit bleibt die $i$-te Spalte des Tableaus unter den Operationen mit der $k$-ten Zeile unverändert. Im Tableau

$$T_{B^+} = \begin{pmatrix} 1 & c^+ & -d^+ \\ 0 & A^+ & b^+ \end{pmatrix},$$

gilt:

$$A_i^+ = A_i, \quad c_i^+ = c_i = 0.$$

Die Umformungen haben aus der $j$-ten Spalte den $k$-ten Einheitsvektor gemacht:

$$A_j^+ = e_k, \quad c_j^+ = 0.$$

Damit ist $c^+$ reduzierter Kostenvektor zur neuen Basis $B^+$.Die Teilmatrix $A_{B^+}^+$ hat dieselben Spalten wie $A_B$, und ist daher ebenfalls Permutationsmatrix. Genauer gilt für

$$\pi^+ : B^+ \to \{1, \ldots, m\} \quad \text{mit} \quad \pi^+(i) = \begin{cases} \pi(i) & \text{für} \quad i \neq l, \\ k & \text{für} \quad i = j. \end{cases}$$

mit $B^+ = \{i_1, \ldots, i_m\}$ gilt dann

$$A_{B^+} = \left(e_{\pi^+(i_1)}, \ldots, e_{\pi^+(i_m)}\right).$$

Das neue Tablau erfüllt die Anforderungen i),ii) an die Gestalt in bezug auf die neue Basis $B^+$. Die zugehörige Basislösung $x_{(B^+)}$ ergibt sich aus der letzten Spalte des Tableaus gemäß (8). Wir halten fest:

**Theorem 6** *Auf dem Tableau $T_B$ einer zulässigen Basis $B$ ist die Pivotoperation durch folgende Schritte gegeben:*

1. *gilt $c_N \geq 0$, so ist das Optimalitätskriterium erfüllt.*

2. *Andernfalls wähle ein $j \notin B$ mit $c_j < 0$. Gilt $A_j \leq 0$, dann ist die Zielfunktion $z$ nach unten unbeschränkt auf $F$.*

3. *Sei $c_j < 0$ und nicht $A_j \leq 0$. Dann bestimme $k$ mit*

$$\frac{b_k}{a_{k\,j}} = \min\{\frac{b_i}{a_{i\,j}} \,|\, a_{i\,j} > 0 \quad i = 1, \ldots, m\},$$

*und $l \in B$ mit $A_l = e_k$. Die neue Basis ist dann $B^+ = B \setminus l \cup j$. Führe die obigen Zeilenumformungen (9) durch, sodaß $c_j = 0$ und $A_j = e_k$ gilt. Somit ergibt sich ein Tableau $T_{B^+}$ zur neuen Basis $B^+$ mit reduziertem Kostenvektor in der 0-ten Zeile. Der Pivotschritt wird erneut ausgeführt.*

Wir schliessen diesen Abschnitt mit einer Anmerkung zur Kostenfunktion. Für den Übergang zu einer neuen Basis $B^+$ gilt für die reduzierten Kostenvektoren:

$$cx + d = (c + \delta A)x + d - \delta b = c^+x + d^+.$$

Die Änderungen des Eintrages $d$ im Tableau entspricht also der Konstanten, um den sich die äquivalenten Kostenfunktionen der reduzierten Kostenvektoren unterscheiden.

**Lemma 4** *Sei $\min_{x \in F} z(x)$ ein LP in Standardform mit reduzierter Darstellung $(A, b, c)$,*

$$F = \{Ax = b, x \geq 0\}, \quad z(x) = cx.$$

*Sei $B$ zulässige Basis mit Basislösung $x_{(B)}$. Hat das Basistableau $T_B$ hat die Gestalt*

$$T_B = \begin{pmatrix} 1 & c_B \\ 0 & A_B \end{pmatrix}^{-1} \begin{pmatrix} 1 & c & 0 \\ 0 & A & b \end{pmatrix} = \begin{pmatrix} 1 & c' & -d' \\ 0 & A' & b' \end{pmatrix},$$

*mit Basisdarstellung $(A', b', c')$ und $d' \in \mathbb{R}$, dann gilt für die Kostenfunktion:*

$$z(x) = cx = c'x + d'.$$

*Die Kosten der Basislösung $x_{(B)}$ betragen somit*

$$z(x_{(B)}) = d'.$$

Wir ergänzen daher die Darstellung eines LP $\min_{x\in F} z(x)$ um den Wert $d \in \mathbb{R}$ zu $(A, b, c, d)$, sodaß für die Kostenfunktion gilt

$$z(x) = cx + d.$$

Bislang war die Kostenfunktion durch die Darstellung $(A, b, c)$ nur bis auf einen konstanten Faktor bestimmt. Wir erweitern den Äquivalenzbegriff aus Definition (3) entsprechend. Zwei Darstellungen $(A, b, c, d)$, $(A', b', c', d')$ eines LP in Standardform sind äquivalent, genau dann wenn gilt:

$$A' = DA \quad b' = Db\,, \quad c' = c + \delta A,$$

und zusätzlich

$$d' = d - \delta b.$$

mit einer regulären Matrix $D \in \mathbb{R}^{m\times m}$ und einem Zeilenvektor $\delta \in R^{1\times m}$.

## 5.6 Initialisierung des Simplexalgorithmus

Sei $\min_{x\in F} z(x)$ ein LP in Standardform mit Darstellung $(A, b, c)$ für $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^{1\times n}$. Es gilt

$$F = \{x \in \mathbb{R}^n \mid Ax = b,\, x \geq 0\} \quad \text{und} \quad z(x) = cx.$$

Indem wir die entsprechenden Zeilen von $Ax = b$ mit $-1$ multiplizieren erreichen wir $b \geq 0$. Die Suche nach einer initialen zulässigen Basislösung leistet das folgende Hilfs-LP $\min_{x\in F'} z'(x)$ mit zulässiger Menge

$$F' = \{(x, y) \in \mathbb{R}^{n+m} \mid (A, E_m)\begin{pmatrix} x \\ y \end{pmatrix} = b\,,\, (x, y) \geq 0\}.$$

Mit dem Zeilenvektor $e \in \mathbb{R}^{1\times m}$ mit $e = (1, \ldots, 1)$ sei die Kostenfunktion gegeben durch

$$z'(x, y) = -eAx + eb.$$

Da $E_m$ die Einheitsmatrix ist, hat $(A, E_m)$ vollen Rang. Es ist

$$\big((A, E_m), b, (-eA, 0), eb\big),$$

eine reduzierte Darstellung des Hilfs-LP. Wegen $b \geq 0$ ist $(0, b) \in F'$ zulässig, und sogar Basislösung mit zulässiger Basis $B = \{n + 1, \ldots, n + m\}$.

Zu dieser Basis ist $(-eA, 0)$ ein reduzierter Kostenvektor, $(-eA, 0)_B = 0$, ferner ist $(A, E_m)_B = E_m$ offensichtlich eine Permutationsmatrix. Das Tableau

$$T_B = \begin{pmatrix} 1 & (-eA, 0) & -eb \\ 0 & (A, E_m) & b \end{pmatrix},$$

hat die Gestalt zur Durchführung des Pivotschrittes. Wir können hierauf den Simplexalgorithmus starten.

Sei $(x,y) \in F'$ zulässig, mit $Ax + y = b$ gilt

$$z'(x,y) = -e(Ax - b) = ey = \sum_i y_i \geq 0.$$

Somit ist die Zielfunktion nach unten durch 0 beschränkt. Damit (wie wir im nächsten Abschnitt genauer zeigen) terminiert der Simplexalgorithmus mit einer optimalen Basislösung $(x^*, y^*) \in F'$ mit Basis $B^* \subset \{1, \ldots, n+m\}$. Die Zulässigkeit des orginären LP hängt von $y^*$ ab:

**Lemma 5** *Das orginäre LP* $\min_{x \in F} z(x)$ *ist unzulässig,* $F = \emptyset$*, genau dann wenn* $y^* \neq 0$ *gilt.*

Eine zulässige Lösung $x \in F$ induziert einen Punkt $(x, 0) \in F'$ des Hilfs-LP mit

$$z'(x,0) = 0 \geq z'(x^*, y^*) = \sum_i y_i.$$

Also $F \neq \emptyset$ impliziert $y^* = 0$. Umgekehrt ist mit $y^* = 0$ natürlich $x^* \in F$ zulässig für das orginäre LP. q.e.d.

Im Fall $y^* \neq 0$ resultieren wir mit einem unzulässigen orginären LP. Andernfalls ist $x^* \in F$ eine Basislösung von $\min_{x \in F} z(x)$, da die Spalten $A_i, i \in B(x^*)$ linear unabhängig sind.

Die resultierende Basis $B^*$ des Hilfs-LP ist nicht notwendig in $\{1, \ldots, n\}$ enthalten, und die Darstellung $(A, b, c)$ des orginären LP dann möglicherweise entartet. Wir suchen daher eine Basis $B \supset B(x^*)$, sowie eine reduzierte Darstellung des orginären LP. Mit hilfe des Gaußschen Algorithmus finden wir eine äquivalente Repräsentation des Gleichungssystems $Ax = b$:

$$DAx = Db \quad \text{mit} \quad D \quad \text{regulär},$$

(es ist $DA$ hier obere Dreiecksmatrix) sodaß wir durch Streichen von Nullzeilen eine reduzierte Darstellung erhalten. Wir bezeichnen die so erhaltene reduzierte Darstellung wieder mit $(A, b, c)$:

$$F = \{Ax = b, x \geq 0\},$$

der Kostenvektor $c$ bleibt unverändert. Eventuell hat sich die Zeilenzahl $m$ reduziert. Der Begriff Basislösung ist Darstellungsunabhängig, wir ergänzen die Indices $B(x^*)$ der Koordinaten $\neq 0$ der Basislösung $x^*$ zu einer neuen Basis $B \supset B(x^*)$ der Darstellung $(A, b, c)$.

Im Fall $B^* \subset \{1, \ldots, n\}$ ist die ursprüngliche Darstellung $(A, b, c)$ des LP $\min_{x \in F} z(x)$ reduziert, und $B := B^*$ schon diesbezüglich Basis zur Basislösung $x^*$. Wir erhalten ein Starttableau zur Basis $B$ mit reduziertem Kostenvektor und Permutationsmatrix in Bezug auf $B$ wie in (7):

$$T_B = \begin{pmatrix} 1 & c_B \\ 0 & A_B \end{pmatrix}^{-1} \begin{pmatrix} 1 & c & -d \\ 0 & A & b \end{pmatrix} = \begin{pmatrix} 1 & c - c_B A_B^{-1} A & -d - c_B A_B^{-1} b \\ 0 & A_B^{-1} A & A_B^{-1} b \end{pmatrix}.$$

Wir erhalten $(c - c_B A_B^{-1} A)_B = 0$ und $(A_B^{-1} A)_B = E_m$ ist die Einheitsmatrix. Ab dieser Stelle können wir den Simplexalgorithmuns mit den Pivotschritten zu Ende bringen. Somit hat der gesamte Simplexalgorithmus die folgende Gestalt:

**Theorem 7** *Sei* $\min_{x \in F} z(x)$ *ein LP in Standardform mit*

$$F = \{x \in \mathbb{R}^n \mid Ax = b,\, x \geq 0\} \quad \textit{und} \quad z(x) = cx.$$

*Der Simplexalgorithmus gliedert sich in folgende Schritte:*

1. *Für das Hilfs-LP* $\min_{x \in F'} z'(x)$ *mit zulässiger Menge*

$$F' = \{(x,y) \in \mathbb{R}^{n+m} \mid (A, E_m)\binom{x}{y} = b\,,\, (x,y) \geq 0\},$$

*und Kostenfunktion*

$$z'(x,y) = -eAx + eb,$$

*sind die Voraussetzungen des Pivotschrittes erfüllt, in Bezug auf die Basis* $B = \{n+1, \ldots, n+m\}$ *mit Basislösung* $(0,b)$*, (ObdA gilt* $b \geq 0$*). Die Kostenfunktion* $z' : F' \to \mathbb{R}$ *ist durch einen reduzierten Kostenvektor dargestellt,* $(-eA, 0)_B = 0$ *und* $(A, E_m)_B$ *ist die Permutationsmatrix* $E_m$*. In diesem Hilfs-LP wird eine optimale Basislösung* $(x,y) \in \mathbb{R}^{n+m}$ *bestimmt. Gilt* $z'(x,y) = 0$*, d.h.* $y = 0$*, so ist* $x$ *eine Basislösung des ursprünglichen LP* $\min_{x \in F} z(x)$*. Andernfalls ist dieses unzulässig,* $F = \emptyset$*.*

2. *Zur Basislösung* $x$ *wird mit dem Basisergänzungssatz eine zugehörige Basis* $B \subset \{1, \ldots, n\}$ *gewählt. (Die resultierende Basis des Hilfs-LP ist nicht notwendig in* $\{1, \ldots, n\}$ *enthalten). Durch Streichen von eventuell linear Abhängigen Zeilen wird das orginäre LP in reduzierte Darstellung gebracht Aus der reduzierten Darstellung* $(A, b, c)$ *des LP's mit zulässiger Basis* $B$ *erhält man durch Matrizenmultiplikation ein Starttableau*

$$T_B = \begin{pmatrix} 1 & c_B \\ 0 & A_B \end{pmatrix}^{-1} \begin{pmatrix} 1 & c & 0 \\ 0 & A & b \end{pmatrix},$$

*mit einer Basisdarstellung, welches die Voraussetzungen des Pivotschrittes zur Basis* $B$ *erfüllt.*

3. *Es sind die Pivotschritte auszuführen, welche entweder in einer nach unten unbeschränkten Zielfunktion,* $\min_{x \in F} z(x) = -\infty$*, oder mit einer optimalen Basislösung* $x^* \in F$ *enden, mit*

$$z(x^*) \leq z(x) \quad \textit{für alle} \quad x \in F.$$

Das beschriebene Verfahren ist eine Zwei-Phasen Methode. Die erste Phase beinhaltet das Lösen des Hilfs-LP zum Auffinden einer initialen Basislösung, oder zur Feststellung der Unzulässigkeit des orginärem LP. Die zweite Phase beinhaltet die Lösung des orginären LP, (sofern dieses zulässig ist).

## 5.7 Kreisen des Simplexalgorithmus, Blands Rule

Soweit der Pivotschritt bisher vorgestellt wurde, können in der Abfolge $B^1, B^2, \dots, B^n, B^{n+1}$ der betrachteten zulässigen Basen diese sich wiederholen: $B^{n+1} = B^1$. Man nennt dies ein Kreisen des Simplexalgorithmus, eventuell terminiert der Simplexalgorithmus nicht.

Der Pivotschritt ist bislang nicht deterministisch. Die Auswahl des Index $j \notin B$ mit $c_j < 0$ ist offen, sowie die Wahl desjenigen

$$k \in \{1, \dots, m\} \quad \text{mit} \quad \frac{b_k}{a_{k\,j}} = \min\{\frac{b_i}{a_{i\,j}} \mid a_{i\,j} > 0\}.$$

Es könnte hier mehrere Minima geben. Damit ist auch die Wahl des Index $l \in B$ mit $A_l = e_k$ offen, der die Basis im Übergang zur Nächsten verlässt: $B^+ = B \setminus l \cup j$.

Zur Wahl der Indices die beim Pivotieren in die Basis eintreten und diese verlassen existieren zwei Entscheidungsregeln.

**Theorem 8 (Bland's Rule)** *Zum Basiswechsel im Pivotschritt des Simplexalgorithmus benutzen wir die Auswahlregeln:*

*i) wähle $j \notin B$ minimal mit $c_j < 0$, sodaß $j$ in die Basis eintritt.*

*ii) wähle unter denjenigen Indices $l \in B$ mit $A_l = e_k$ und*

$$\frac{b_k}{a_{k\,j}} = \min\{\frac{b_i}{a_{i\,j}} \mid a_{i\,j} > 0\},$$

*den kleinsten. Somit ist der Index $l$ minimal, der die Basis verlässt.*

*Unter Einhaltung der Regeln i),ii) ist die Abfolge $B^1, \dots, B^n$ der Basen beim Pivotieren ohne Wiederkehr einer Basis. In diesem Fall terminiert der Simplexalgorithmus.*

Sofern ein Kreisen auftritt, in der Abfolge $B^1, B^2, \dots, B^n, B^{n+1}$ der Basen im Simplexalgorithmus, mit $B^1 = B^{n+1}$, ändert sich jedenfalls der Kostenfunktionswert nicht. Sei $x^i$ die zugehörige Basislösung und $(A^i, b^i, c^i, d^i)$ die jeweils zugehörige Darstellung dann gilt für $i, j \in \{1, \dots, n\}$

$$d^i = z(x^i) = z(x^j) = d^j. \tag{10}$$

Unter den Indices, welche in die Basen ein- und austreten sei $q \in \{1, \dots, n\}$ maximal. Ferner sei $B^r$ eine Basis in deren Nachfolger $B^{r+1}$ der Index $q$ eintritt, und $B^s$ eine Basis aus der $q$ austritt:

$$B^{r+1} = B^r \setminus l \cup q \quad \text{für ein} \quad l \in B^r,$$

$$B^{s+1} = B^s \setminus q \cup p \quad \text{für ein} \quad p \in B^{s+1},$$

Sei $x = x_{(B^s)}$ die Basislösung zur Basis $B^s$, und $y = (y_{B^s}, y_{N^s})$ die Suchrichtung im Pivotschritt von $B^s$ zu $B^{s+1}$. Diese erfüllt die homogene Gleichung $A^s y = 0$, sowie $A^s_{B^s} y_{B^s} = A^s_p$. Für die Variationslösung $x_t = x - ty$ im Übergang zu $B^{s+1}$ gilt für alle $t \in \mathbb{R}$

$$A^s x_t = b^s.$$

Die Funktion $z : F \to \mathbb{R}$ lässt sich auf den affinen Unterraum $U = \{x \in \mathbb{R}^n \mid A^s x = b^s\}$ fortsetzen. Sowohl $U$ als auch die Ausdehnung von $z$ auf $U$ hängen nich von der konkreten Darstellung in der Aquivalenzklasse $(A^i, b^i, c^i, d^i)_\sim$ ab. Folglich sind

$$U = \{x \in \mathbb{R}^n \mid A^i x = b^i\} \quad \text{sowie} \quad z(x) = c^i x + d^i \quad \text{für alle} \quad x \in U,$$

unabhängig von $i$. Mit (10) und $x_t \in U$ für alle $t \in \mathbb{R}$ folgt insbesondere

$$c^r x_t = c^s x_t. \tag{11}$$

Wir wollen diese Gleichung zum Widerspruch führen.

Für den Index $q \in B^s$, der aus der Basis $B^s$ austritt gilt:

$$\frac{x_q}{y_q} = \min\{\frac{x_i}{y_i} \mid y_i > 0\,,\, i \in B^s\}.$$

Wir wählen ein $\tau \in \mathbb{R}$ mit

$$\frac{x_q}{y_q} < \tau \leq \min\{\frac{x_i}{y_i} \mid y_i > 0\,,\, i \in B^s \quad \text{und} \quad i < q\}. \tag{12}$$

Ein solches $\tau \in \mathbb{R}$ existiert, wegen der minimalität von $q$ nach Regel ii) gilt

$$\frac{x_q}{y_q} < \min\{\frac{x_i}{y_i} \mid y_i > 0\,,\, i \in B^s \quad \text{und} \quad i < q\}. \tag{13}$$

Sonst hätte ein Index $i < q$ die Basis $B^s$ verlassen. Wegen $\frac{x_q}{y_q} < \tau$ ist die Variationslösung $x_\tau = x^s - \tau y$ ist nicht mehr zulässig, $x_\tau \notin F$ mit

$$x_{\tau,q} < 0. \tag{14}$$

Es liegt aber $x_\tau$ noch im affinen Unterraum $U$. Zudem gilt für alle $i < q$ :

$$x_{\tau,i} \geq 0. \tag{15}$$

Für $i \in B^s \cap \{y_i < 0\}$ folgt dies aus (12). Für alle anderen $i < q$ gilt $y_i \leq 0$ und damit $x_{\tau,i} \geq 0$ sowieso. (Für $i \notin B^s$ folgt dies aus $y_{N^s} = -e_p$).

Wir untersuchen $c^r x_\tau$. Für $i < q$ gilt

$$c^r_i \geq 0.$$

Für $i \in B^r$ sogar $c^r_i = 0$, für $i \notin B^r$ hätte man ansonsten nach Regel i) $i$ anstelle von $q$ in die Basis $B^{r+1}$ eintreten lassen. Mit (15) folgt sodann

$$c^r_i x_{\tau,i} \geq 0. \quad \text{für} \quad i < q.$$

Für $i > q$ gilt $i \in B^r$ genau dann wenn $i \in B^s$, denn keiner dieser Indices ist ein eine Basis des Zyklus $B^1, \ldots, B^n$ ein- noch ausgetreten. Für $i \in B^r$ gilt $c_i^r = 0$, für $i \notin B^s$, $i > q$ gilt aber $x_{\tau,i} = 0$, wegen $B(x_\tau) \subset B^s \cup p$ und $p < q$. Also folgt

$$c_i^r x_{\tau,i} = 0 \quad \text{für} \quad i > q.$$

Da $q$ in die Basis $B^r$ eintritt gilt $c_q^r < 0$. Mit (14) gilt sodann $c_q^r x_{\tau,q} > 0$. Somit folgt:

$$c^r x_\tau = \sum_i c_i^r x_{\tau,i} \geq c_q^r x_{\tau,q} > 0.$$

Nun untersuchen wir $c^s x_\tau$. Wegen $c_{B^s}^s = 0$ und $B(x_\tau) \subset B_s \cup p$ folgt

$$c^s x_\tau = c_p^s x_{\tau,p}.$$

Da $p$ in die Basis $B^s$ eintritt, folgt $c_p^s < 0$, aus (15) und $p < q$ folgt $x_{\tau,p} \geq 0$. Wir erhalten:

$$c^s x_\tau = c_p^s x_{\tau,p} \leq 0.$$

Der Widerspruch in (11) ist hergeleitet, die linke Seite ist größer Null, die Rechte kleiner gleich Null.

Somit kann unter Beachtung der Auswahlregeln von Bland, eine Basis sich nicht widerholen. Somit ist ein Kreisen ausgeschlossen. Da es nur endlich viele Basen gibt, muß der Simplexalgorithmus terminieren.

Die Theorema (5,6,7,8) beschreiben den Simplexalgorithmus und postulieren dessen Korrektheit. Aus dieser folgt die Existenz einer Basisösung $x_{(B)} \in F$ eines zulässigen beschränkten LP's $\min_{x \in F} z(x)$, welche das Optimalitätskriterium (3) erfüllt.

Wir gehen abschließend nochmals auf das Thema Dualität ein. In Bezug auf die initiale Darstellung $(A, b, c)$ hat das LP die Gestalt

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0. \end{array}$$

sowie ein Basistablau zur Basis $B$ der Gestalt, vgl. (7),

$$T_B = \begin{pmatrix} 1 & c - c_B A_B^{-1} A & -c_B A_B^{-1} b \\ 0 & A_B^{-1} A & A_B^{-1} b \end{pmatrix}.$$

Die Optimalitätsbedinung des Simplexalgorithmus lautet

$$c - c_B A_B^{-1} A \geq 0.$$

Für $y^T := c_B A_B^{-1}$ gilt also $c - y^T A \geq 0$. Mit $s^T := c - y^T A$ ist somit $(x_{(B)}, y, s)$ ein primal-dual zulässiges Paar:

$$A^T y + s = c^T, \quad \text{und} \quad s \geq 0.$$

Der optimale Kostenfunktionswert des primalen LP ist gleich den Kosten des dualen Punktes $(y, s)$. Die optimalen Kosten der Basislösung $x_{(B)}$ sind aus der oberen rechten Ecke von $T_B$ abzulesen:

$$z(x_{(B)}) = c_B A_B^{-1} b = y^T b.$$

Die Dualitätslücke ist geschlossen. Mit dem schwache Dualitätssatz ist $(y, s)$ optimale Lösung des dualen LP. Aus der optimalen primalen Lösung $x_{(B)}$ zur Basis $B \subset \{1, \ldots, n\}$ des Simplexalgorithmus folgt die Existenz einer optimalen dualen Lösung $(y, s)$ mit

$$y^T := c_B A_B^{-1} \quad \text{und} \quad s^T := c - y^T A.$$

Somit haben wir einen zweiten Beweis des starken Dualitätssatzes.

## 6 Notation in der Graphentheorie

**Definition 11 (Graph)** *Ein Graph $G = (V, E)$ besteht aus einer Menge $V$ von Ecken ("vertices") sowie einer Menge*

$$E \subset V^{(2)},$$

*von Kanten, ("edges"), die zweielementige Teilmengen von $V$ sind:*

$$V^{(2)} := \{U \subset V \,|\, |U| = 2\}.$$

Somit enthält ein Graph keine Kanten mit identischer Anfangs- und Endecken, sogenannte Schleifen.

**Definition 12 (Digraph)** *Ein Digraph $G = (V, E)$ besteht aus einer Menge $V$ von Ecken (engl. "vertices") sowie einer Menge*

$$E \subset V \times V \setminus \{\,(v, v) \,|\, v \in V\},$$

*von gerichteten Kanten ("arcs"), die geordnete Paare der Ecken sind. Die Wegnahme der Diagonalen $D := \{\,(v, v) \,|\, v \in V\}$ schließt Schleifen aus, wie im Fall ungerichteter Graphen.*

**Definition 13 (Teilgraphen)** *Sei $G = (V, E)$ ein Graph, $T \subset E$ Teilmenge der Kanten und $S \subset V$ Teilmenge der Ecken. Gilt $T \subset S^{(2)}$, so ist $G' := (S, T)$ ein Teilgraph. Wir definieren spezielle Teilgraphen von $G$ wie folgt:*

$$\begin{aligned}
G \cap T &:= (V, T), \\
G \cap S &:= (S, E \cap S^{(2)}), \\
G \setminus T &:= (V, E \setminus T), \\
G \setminus S &:= (V \setminus S, E \cap (V \setminus S)^{(2)}),
\end{aligned}$$

*Für eine Kantenmenge $T \subset V^{(2)}$ sei*

$$G \cup T := (V, E \cup T).$$

*Wir bezeichnen $G \cap T$ im folgenden kurz mit $T$, (und hoffen Verwechslungen durch den Kontext auszuschließen).*

Ein Graph $G = (V, E)$ lässt sich stets durch einen Digraphen beschreiben $D = (V, E_D)$, indem für jede Kante $e = (u, v) \in E$ die gerichteten Kanten $(u, v)$, $(v, u)$ in der Menge $E_D$ enthalten sind.

Digraphen lassen sich mit den folgenden Datenstrukturen beschreiben. Die Wahl einer geeigneten Datenstruktur hat wesentlichen Einfluß auf Laufzeit und Implementierbarkeit eines Graphentheoretischen Algorithmus

**Definition 14 (Adjazenzmatrix,Inzidenzmatrix)** *Sei $D = (V, E)$ ein Digraph mit Anzahl $n = |V|$ an Ecken und $m = |E|$ vielen gerichteten Kanten. Sei $V = \{v_1, \ldots, v_n\}$ Abzählung der Ecken, und $E = \{e_1, \ldots, e_m\}$ eine Solche der Kanten. Die Adjazenzmatrix $A \in \{0,1\}^{n \times n}$ beschreibt die Kantenmenge $E$ durch*

$$A_{i,j} = 1 \iff \begin{cases} 1 & \iff (v_i, v_j) \in E, \\ 0 & sonst. \end{cases}$$

*Es gilt $A_{i,j} = 1$ genau dann wenn eine Kante von $v_i$ zu $v_j$ führt, $v_i, v_j$ heißen dann adjazent. Die Inzidenzmatrix $I \in \mathbb{Z}^{n \times m}$ ist definiert durch*

$$I_{i,j} = \begin{cases} -1 & \iff \exists k \quad mit \quad e_j = (v_i, v_k), \\ 1 & \iff \exists k \quad mit \quad e_j = (v_k, v_i), \\ 0 & sonst. \end{cases}$$

*Es gilt $I_{i,j} = 0$ wenn $v_i$ weder Anfangs- noch Endecke von $e_j$ ist. Es ist $I_{i,j} = -1$ genau dann wenn Kante $e_j$ von Ecke $v_i$ ausgeht, und $I_{i,j} = 1$ wenn $e_j$ in $v_i$ endet. Im Fall $I_{i,j} \neq 0$ sind $v_i, e_j$ inzident, es ist $v_i$ Anfangs- oder Endecke von $e_j$.*

**Definition 15 (total unimodular)** *Eine Matrix $A \in \mathbb{Z}^{n \times m}$ heißt total unimodular, genau dann wenn für alle quadratischen Untermatrizen $B$ gilt*

$$det B \in \{0, 1, -1\}.$$

*Für eine Matrix $A \in \mathbb{R}^{n \times m}$ ist $B \in \mathbb{R}^{r \times r}$ mit $r \leq n, m$ quadratische Untermatrix, ganau dann wenn es Zeilenindices $1 \leq i_1 < i_2 < \ldots < i_r \leq n$ und Spaltenindices $1 \leq j_1 < j_2 < \ldots < j_r \leq m$ gibt mit*

$$B_{k,l} = A_{i_k, j_l} \quad für\ 1 \leq k, l \leq r.$$

Offenbar gilt $A_{i,j} \in \{0, 1, -1\}$ für eine total unimodulare Matrix $A$. Ganzahlige lineare Programme ILP's, mit einer Matrix $A \in \mathbb{Z}^{m \times n}$, einem Vektor $b \in \mathbb{Z}^m$ sowie einem Zeilenvektor $c \in \mathbb{Z}^{1 \times n}$ gegeben durch:

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0, x \in \mathbb{Z}^n, \end{array}$$

mit total unimodularer Matrix $A$ werden schon durch ihre LP-relaxation

$$\begin{array}{ll} \min & cx \\ \text{unter} & Ax = b, \quad x \geq 0, x \in \mathbb{R}^n, \end{array}$$

gelöst, da deren Basislösungen sämtlich ganzzahlig sind $x_B = A_B^{-1} b \in \mathbb{Z}^n$ für alle Basen $B \subset \{1, \ldots, m\}$.

**Theorem 9** *Die Inzidenzmatrix $A$ eines Digraphen $D = (V, E)$ ist total unimodular.*

Beweis: Jede Spalte von $A$ hat genau zwei Einträge $\neq 0$, es ist $A_k = e_j - e_i$ mit Einheitsvektoren $e_i, e_j \in \mathbb{R}^n$, $(n = |V|)$, sofern die Kante $e_k = (v_i, v_j)$ von Anfangsecke $v_i$ zu Endecke $v_j$ zeigt.

Mit dem Determinantenentwicklungssatz hat ein minimaler Verbrecher $B \in \mathbb{Z}^{r\times r}$ als quadratische Untermatrix mit $detB \notin \{0, 1, -1\}$ (und $r$ minimal) genau zwei Einträge in jeder Spalte. Dadurch sind aber seine Zeilen linear abhängig. Mit den Zeilenvektoren $b_i = (B_{i,1}, B_{i,2}, \ldots, B_{i,r})$ gilt genauer:

$$\sum_i b_i = 0,$$

und daher der Widerspruch $detB = 0$. q.e.d.

**Definition 16 (Wege,Kreis)** *Sei $G = (V, E)$ ein Graph, eine Aufzählung von Ecken*

$$\gamma = (v_1, \ldots, v_n),$$

*heißt Weg, sofern die aufeinanderfolgenden Ecken durch Kanten verbunden sind,*

$$e_i := (v_i, v_{i+1}) \in E \quad \textit{für} \quad 1 \leq i < n.$$

*Es ist $\gamma$ ein Kreis, sofern $v_1 = v_n$ gilt, der Weg zu seinem Anfangspunkt zurückkehrt. In der Regel verlangen wir, daß die involvierten Kanten verschieden sind,*

$$e_i \neq e_j \quad \textit{für} \quad i \neq j.$$

**Definition 17 (Nachbarschaft,Eckengrad)** *Für eine Ecke $v \in V$ eines Graphen $G = (V, E)$ sind die Nachbarn $N(v) \subset V$ definiert durch*

$$u \in N(v) \iff (u, v) \in E.$$

*Die Ecke $u$ ist Nachbar von $v$, falls $u, v$ durch eine Kante verbunden sind. Für eine Ecke $v \in V$ eines Graphen $G = (V, E)$ ist der Eckengrad $d(v) \in \mathbb{N}_0$ durch die Anzahl benachbarter (mit einer Kante verbundener) Ecken definiert:*

$$d(v) := |N(v)|.$$

**Definition 18 (Zusammenhang,Baum)** *Ein Graph $G = (V, E)$ heißt zusammenhängend, falls je zwei Ecken $s, t \in V$ durch einen Weg verbunden sind,*

$$\gamma = (v_1, \ldots, v_n) \quad \textit{mit} \quad v_1 = s,\, v_n = t.$$

*Ein zusammenhängender Graph $G$ heißt Baum, falls $G$ keinen Kreis enthält. Dies ist äquivalent zu $|E| = |V| - 1$, (für zusammenhängende Graphen). Ein Blatt $v \in V$ eines Baums ist mit genau einer Kante inzident, der Eckengrad ist eins,*

$$d(v) = 1.$$

*Sei $G = (V, E)$ ein zusammenhängender Graph. Ein aufspannender Baum $T \subset G$ ist ein zusammenhängender Teilgraph mit gleicher Eckenmenge,*

$$T = (V, E_T), \quad \textit{mit} \quad E_T \subset E,$$

*sodaß $T$ ein Baum ist.*

Wir benutzen hier das Symbol $\subset$ der Enthaltenseinsrelation für die Teilgraph beziehung, die in der Tat transitiv ist. Jeder Baum $T = (V, E)$, mit $|V| \geq 2$, hat mindestens zwei Blätter.

**Definition 19 (Hamiltonkreis)** *Sei $G = (V, E)$ ein Graph mit $|V| = n$ Ecken. Deren Aufzählung*

$$H = (v_1, \ldots, v_n),$$

*ist ein Hamiltonkreis, falls je zwei aufeinanderfolgende Ecken durch Kanten verbunden sind:*

$$(v_i, v_{i+1}) \in E \quad \textit{für} \quad 1 \leq i \leq n-1,$$
$$(v_n, v_1) \in E.$$

Wir geben ein hinreichendes Kriterium für die Existenz eines Hamiltonkreises:

**Lemma 6** *Sei $G = (V, E)$ ein Graph mit $|V| \geq 3$, jede Ecke $v \in V$ hat mindestens $|V|/2$ viele Nachbarn,*

$$|N(v)| \geq \frac{|V|}{2}.$$

*Dann besitzt $G$ einen Hamiltonkreis.*

Beweis: Sei $G = (V, E)$ ein maximaler Verbrecher, durch hinzufügen einer Kante $(u, v) \notin E$ erhält $G \cup (u, v)$ einen Hamiltonkreis $H = (u, u_1, \ldots, u_{n-2}, v)$. ObdA hat $G$ vier Ecken, $n = |V| \geq 4$.

In diesem Hamiltonkreis gibt es ein $i \in \{1, \ldots, n-3\}$ mit $u_i$ Nachbar zu $v$, und $u_{i+1}$ Nachbar zu $u$. Die Indexmengen

$$I_1 := \{i \mid u_i \in N(v)\},$$
$$I_2 := \{i \mid u_{i+1} \in N(u)\}.$$

haben nichtleeren Durschnitt $\exists i \in I_1 \cap I_2 \neq \emptyset$, wegen

$$|I_1| \geq n/2, \quad |I_2| \geq n/2 - 1, \quad |I_1 \cup I_2| = n - 2.$$

Somit ist
$$H' = (u, u_1, \dots, u_i, v, u_{n-2}, \dots, u_{i+1}),$$
ein Hamiltonkreis der schon in $G$ enthalten ist, ein Widerspruch. q.e.d.

**Definition 20** *Sei $G = (V, E)$ ein Graph mit Teilmenge $U \subset V$ von Ecken. Der Teilgraph $G \cap U$ ist eine Zusammenhangskomponente von $G$, falls $G \cap U$ zusammenhängend ist, es keine Kanten zwischen $U$ und $V \setminus U$ gibt:*

$$(u, v) \in E \implies u, v \in U \text{ oder } u, v \in V \setminus U.$$

*Ein Graph $G = (V, E)$ ist zusammenhängend, falls $V$ die einzige (nichtleere) Zusammenhangskomponente ist.*

Die Anzahl der Zusammenhangskomponenten eines Graphen $G = (V, E)$ sei mit $\phi(G)$ bezeichnet. Somit können wir nun ein notwendiges Kriterium für die Existenz eines Hamiltonkreises angeben.

**Lemma 7** *Für jede Teilmenge $S \subset V$ eines Graphen $G = (V, E)$ mit Hamiltonkreis $H$ muss gelten:*
$$\phi(G \setminus S) \leq |S|.$$
*Es muß $S$ mindestens so viele Ecken haben wie $G \setminus S$ Zusammenhangskomponenten hat.*

Beweisskizze: Entnimmt man einem Hamiltonkreis $H$ eine Ecke so hat er immer noch eine Zusammenhangskomponente.

Für jede weitere entnommene Ecke erhöht sich die Anzahl Zusammenhangskomponenten höchstens um 1, es gilt

$$\phi(H \setminus S) \leq |S|.$$

Der Teilgraph $H$ ist aus $G$ durch entfernen von Kanten aus $G$ entstanden, und hat daher mehr oder gleich viele Zusammenhangskomponenten wie $G$:

$$|S| \geq \phi(H \setminus S) \geq \phi(G \setminus S). \quad q.e.d.$$

**Definition 21 (Eulertour)** *Sei $G = (V, E)$ ein zusammenhängender Graph mit $|E| = m$ vielen Kanten. Eine Eulertour $ET = (v_1, \dots, v_m)$ ist eine Folge von Ecken, in der jede Kante genau einmal vorkommt:*

$$\{(v_i, v_{i+1}), (v_m, v_1) \,|\, 1 \leq i \leq m-1\} = E.$$

Wir geben ein hinreichendes und notwendiges Kriterium für die Existenz einer Eulertour an:

**Lemma 8** *Ein Graph $G = (V, E)$ besitzt genau dann eine Eulertour wenn der Eckengrad überall gerade ist,*

$$d(v) \equiv 0 \mod 2 \quad \textit{für alle} \quad v \in V.$$

Beweisskizze: Es ist $\Rightarrow$ trivial, zu $\Leftarrow$ : Sei $G$ ein minimaler Verbrecher, es ist $G$ kein Baum, denn Blätter haben den Eckengrad 1. Daher existiert ein Kreis $C = (v_1, \ldots, v_r) \subset G$. Die Zusammenhangskomponenten $G_i, i \in I$ von $G \setminus C$ haben weiter geraden Eckengrad und besitzen daher Eulertouren $ET_i, i \in I$. Jede Komponente besitzt eine Ecke $v_i \in G_i \cap C$ An dieser Stelle kann die Tour $ET_i$ in $C$ eingefügt werden. (Die Ecken $v_i, i \in I$ sind verschieden).

Somit entsteht Eulertour für den gesamten Graphen $G$, ein Widerspruch. q.e.d.

# 7 Minimal aufspannende Bäume

Sei $G = (V, E)$ ein zusammenhängender Graph, mit Gewichtsfunktion

$$\omega : E \to \mathbb{R}.$$

Für eine Teilmenge $T \subset E$ von Kanten, bzw. einen Teilgraphen $T = (V', E')$ mit $V' \subset V$ und $E' \subset V'^{(2)} \cap E$ sei das Gewicht von $G$ gegeben durch:

$$\omega(T) := \sum_{e \in T} \omega(e), \quad .$$

Im Falle eines Teilgraphen $T$ bedeutet $e \in T$ genauer $e \in E'$.

**Definition 22** *Sei $G = (V, E)$ ein zusammenhängender Graph. Ein minimal aufspannender Baum $T$ ist ein Solcher mit mimimalem Gewicht:*

$$\omega(T) \leq \omega(T'),$$

*für alle aufspanneden Bäume $T'$ von $G$.*

Durch hinzufügen einer Kante $f \notin T$ zu einem Baum $T = (V, E)$ enthält $T \cup f$ genau einen Kreis. diesen bezeichnen wir mit $C_T(f)$. Zur Entscheidung ob ein Aufspannender Baum $T$ eines Graphen $G = (V, E)$ ein minimaler Aufspannender Baum, kurz MST ("minimal spanning tree") ist, verfügen wir über das folgende Kriterium.

**Lemma 9** *Sei $G = (V, E)$ ein zusammenhängender Graph, und $T \subset E$ ein aufspannender Baum. Es ist $T$ ein MST genau dann wenn für alle $f \in E \setminus T$ gilt:*

$$\omega(e) \leq \omega(f) \quad \textit{für alle} \quad e \in C_T(f). \tag{16}$$

Beweis: Sei $T$ ein MST für den die Optimalitätsbedingung (16) verletzt ist,

$$\omega(e) > \omega(f) \quad \text{für} \quad e \in T,\ f \notin T$$

Es ist $T' = (T \cup f) \setminus e$ ebenfalls ein Aufspannender Baum, mit kleinerem Gewicht:

$$\omega(T') = \omega(T) - \omega(e) + \omega(f) < \omega(T),$$

ein Widerspruch.

Sei nun $T$ ein aufspannender Baum, der das Optimalitätskriterium (16) erfüllt, und $T'$ ein minimal aufspannender Baum, mit soviel Kanten zu $T$ gemeinsam als möglich ($T \cap T'$ maximal). Da wir $T' \neq T$ annehmen existiert eine Kante $f \in T' \setminus T$. Da $T$ das Optimalitätskriterium erfüllt, gilt

$$\omega(e) \leq \omega(f) \quad \text{für alle} \quad e \in C_T(f).$$

Ein $e' \in C_T(f)$ verbindet die beiden Zusammenhangskomponenten von $T' \setminus f$. Somit folgt $f \in C_{T'}(e')$ und $T'' := T' \cup e' \setminus f$ ist ein aufspannender Baum mit

$$\omega(T'') = \omega(T') + \omega(e') - \omega(f) \leq \omega(T').$$

Es ist somit $T''$ ebenfalls ein MST. Da $e'$ Kante von $T$, aber $f \notin T$ hat $T''$ eine Kante mehr gemeinsam zu $T$ als $T'$, ein Widerspruch. Es muß also $T$ ein MST sein. q.e.d.

Der Beweis zeigt, wie man durch sukzessiven Kantenaustausch zwei verschiedene MST ineinander Überführen kann. Dies ermöglicht entsprechende lokale Suchverfahren nach MST. Hat man mehr als ein Kantengewicht, etwa $\omega_1 : E \to \mathbb{R}$ und $\omega_2 : E \to \mathbb{R}$ und nennt einen Baum $T$ effizient, falls für jeden anderen aufspannenden Baum $T'$ gilt:

$$\begin{aligned} \text{entweder} \quad & \omega_1(T) < \omega_1(T'), \\ \text{oder} \quad & \omega_2(T) < \omega_2(T'), \\ \text{oder} \quad & \omega_1(T) = \omega_1(T') \quad \text{und} \quad \omega_2(T) = \omega_2(T'). \end{aligned}$$

Die effizienten Bäume sind nun nicht mehr durch Kantenaustausch ineinander überführbar (Beispiel ?). In der multikriteriellen Optimierung stellen sich daher die von MST involvierten Optimierungsaufgaben neu.

Sei $G = (V, E)$ ein Graph. Ein Wald $W$ ist ein Teilgraph $W = (V', E')$ mit gleicher Eckenmenge $V = V'$, der keinen Kreis enthält. Wir formulieren den Algorithmus von Kruskal, der einen minimalen aufspanneden Baum generiert.

**Algorithmus 1 (Kruskal)** *Sei $G = (V, E)$ ein zusammenhängender Graph, $W \subset E$ ein Wald.*

- *wähle eine Kante $e \notin W$ von minimalem Kantengewicht, sodaß $W \cup e$ ein Wald bleibt (solange dies geht).*

*Der Algorithmus von Kruskal terminiert mit einem Wald $T$, der ein minimal aufspannender Baum ist.*

Beweis: Offenbar terminiert der Algorithmus mit einem aufspannenden Baum $T$. Dieser erfüllt auch das Optimalitätskriterium (16) aus Lemma (9). Sonst gäbe es ein $f \notin T$ und ein $e \in C_T(f)$ mit

$$\omega(e) > \omega(f).$$

Sei $W \subset T$ der Wald vor Hinzunahme von $e$. Bei der Auswahl von $e$ kann $f$ nicht zur Verfügung gestanden haben, also hat $W \cup f$ ein Kreis. Dann aber auch $T \setminus e \cup f$, ein Widerspruch. q.e.d.

# 8 Flüsse in Netzwerken

**Definition 23** *Sei $G = (V, E)$ ein Digraph, und $c : E \to \mathbb{R}_{\geq 0}$ eine Kapazitätsfunktion auf den Kanten, sowie $s, t \in V$ zwei ausgezeichnete Ecken (Es sei $\mathbb{R}_{\geq 0} = \{x \in \mathbb{R} | x \geq 0\}$). So nennen wir $(G, c, s, t)$ ein Netzwerk. Für ein Funktion $f : E \to \mathbb{R}$, und eine Ecke $v \in V$ sei*

$$f^+(v) := \sum_{(v,u) \in E} f(v, u),$$
$$f^-(v) := \sum_{(u,v) \in E} f(u, v).$$

*Es ist $f^+(v)$ der Fluss, der aus $v$ herausfließt, und $f^-(v)$ derjenige, der in $v$ hineinfließt. Es ist $f$ ein Fluss auf dem Netzwerk $(G, c, s, t)$ sofern gilt:*

$$0 \leq f(e) \leq c(e) \quad \text{für alle Kanten} \quad e \in E,$$

*sowie $f$ die Erhaltungsgleichungen erfüllt:*

$$f^+(v) - f^-(v) = 0 \quad \text{für alle Ecken} \quad v \in V, v \neq s, t.$$

*Der Flusswert von $f$ ist definiert durch*

$$\mathrm{val}(f) := f^+(s) - f^-(s).$$

*Sei $V = S \,\dot{\cup}\, T$ eine Partition der Eckenmenge. Wir nennen $(S, T)$ einen Schnitt sofern weiter gilt:*

$$s \in S, \quad \text{und} \quad t \in T.$$

*Die Kapazität $c(S, T)$ des Schnitts ist definiert durch:*

$$c(S, T) := \sum_{u \in S, v \in T} c(u, v).$$

Der Wert $\mathrm{val}(f)$ eines Flusses $f$ ist stets kleiner als die Kapazität $c(S,T)$ eines Schnittes $(S,T)$:

$$
\begin{aligned}
\mathrm{val}\, f &= f^+(s) - f^-(s) \\
&= \sum_{v \in S} f^+(v) - f^-(v) \\
&= \sum_{\{(v,u)\,\mid\, v \in S\}} f(v,u) - \sum_{\{(u,v)\,\mid\, v \in S\}} f(u,v) \\
&= \sum_{v \in S,\, u \in S} f(v,u) + \sum_{v \in S,\, u \in T} f(v,u) - \sum_{u \in S,\, v \in S} f(u,v) - \sum_{u \in T,\, v \in S} f(u,v) \\
&= \sum_{v \in S,\, u \in T} f(v,u) - \sum_{u \in T,\, v \in S} f(u,v) \\
&\leq \sum_{v \in S,\, u \in T} f(v,u) \\
&\leq \sum_{v \in S,\, u \in T} c(v,u) \\
&= c(S,T).
\end{aligned}
$$

Für einen Fluss $f$ auf einem Netzwerk $(G,c,s,t)$ definieren wir einen $f$-zunehmenden Weg.

**Definition 24** *Sei $\gamma = (v_1, \ldots, v_n)$ ein Folge von Ecken, sodaß für $1 \leq i < n$ gilt entweder*

$$(v_i, v_{i+1}) \in E, \qquad \mathit{mit} \quad f(v_i, v_{i+1}) < c(v_i, v_{i+1}),$$

*es handelt sich um eine Vorwärtskante, oder*

$$(v_{i+1}, v_i) \in E, \qquad \mathit{mit} \quad f(v_{i+1}, v_i) > 0.$$

*es handelt sich um eine Rückwärtskante. Die involvierten Kanten seien alle verschieden, $(v_i, v_{i+1}) \neq (v_j, v_{j+1})$ für $i \neq j$. Dann nennen wir $\gamma$ ein $f$-zunehmender Weg von $v_1$ zu $v_n$ in $(G,c,s,t)$. Für einen solchen definieren wir weiter*

$$\delta(\gamma) := \min\{c(e) - f(e) \,|\, e \ \mathit{Vorwärtskante}\} \cup \{f(e') \,|\, e' \ \mathit{Rückwärtskante}\} > 0.$$

Entlang eines $f$-zunehmenden Weges $\gamma$ von $s$ zu $t$ kann der Flusswert erhöht werden zum Flusswert $\mathrm{val}(f) + \delta(\gamma)$. Hierdurch charakterisieren wir maximale Flüsse $f$ mit maximalem Flusswert $\mathrm{val}(f)$:

**Lemma 10** *Sei $f$ ein Fluss in einem Netzwerk $(G,c,s,t)$. Es ist $f$ ein maximaler Fluss, genau dann wenn es keinen $f$-zunehmenden Weg von $s$ zu $t$ gibt.*

Beweis: Sei $f$ ein maximaler Fluss, und $\gamma$ ein $f$-zunehmender Weg von $s$ zu $t$. Dann können wir $\mathrm{val}(f)$ um den positiven Wert $\delta(\gamma)$ erhöhen, indem wir auf den Vorwärtskanten $e \in E$ den Fluss von $f(e)$ auf $f(e) + \delta(\gamma)$ erhöhen, und auf Rückwärtskanten $e' \in E$ von $f(e')$ auf $f(e') - \delta(\gamma)$ erniedrigen. Hierdurch bleiben die Erhaltungsgleichungen gültig. Ein Widerspruch zu $f$ maximal, es kann kein $f$-zunehmender Weg geben. Sei nun

$$S := \{v \in V \,|\, \text{zu denen es einen } f\text{-zunehmenden Weg von } s \text{ aus gibt}\}$$

Nach Voraussetzung gilt $t \notin S$, mit $T := V \setminus S$ ist $(S,T)$ ein Schnitt mit

$$\mathrm{val}(f) = \sum_{u \in S,\, v \in T} f(u,v) - \sum_{u \in S,\, v \in T} f(v,u) = \sum_{u \in S,\, v \in T} c(u,v) = c(S,T).$$

Denn es gilt

$$\sum_{u \in S,\, v \in T} f(u,v) = \sum_{u \in S,\, v \in T} c(u,v),$$

ansonsten wäre ein $v \in T$ via einer Vorwärtskante $(u,v)$ mit $u \in S$ durch einen $f$-zunehmenden Weg von $s$ aus zu erreichen. Weiter gilt

$$\sum_{u \in S,\, v \in T} f(v,u) = 0,$$

da sonst $v \in T$ via einer Rückwärtskante zu erreichen wäre.

Da ein Schnitt $(S,T)$ existiert mit Kapazität $c(S,T) = \mathrm{val}(f)$, handelt es sich bei $f$ um einen maximalen Fluss. q.e.d.

Der Algorithmus von Ford Fulkerson generiert für ein Netzwerk $(G,c,s,t)$ ein maximalen Fluss $f$. Hierbei verbessert er den Flusswert entlang eines $f$-zunehmenden Weges solange dies geht.

**Algorithmus 2 (Ford Fulkerson)** *Sei $(G,c,s,t)$ ein Netzwerk, der Algorithmus von Ford Fulkerson besteht aus den folgenden Schritten:*

1. *wähle einen initialen Fluss $f$, etwa $f = 0$, oder übernehme einen schon konstruierten Fluss. Initialisiere eine Eckenmenge $S \subset V$ durch $S = \{s\}$, sowie eine Teilmenge $U \subset S$ durch $U = \emptyset$. Ferner sei $d(s) := \infty$.*

2. *Solange $U \subsetneqq S$ und $t \notin S$ gelten, wählen wir ein $v \in S \setminus U$, und führen mit den benachbarten Ecken $w \in N(v)$ folgenden Schritt durch:*

   - *Alle Ecken $w \in N(v) \setminus S$, zu denen es eine Vorwärtskante gibt,*

     $$(v,w) \in E, \quad \textit{mit} \quad f(v,w) < c(v,w),$$

     *nehmen wir in $S$ auf $S^+ := S \cup w$, und definieren*

     $$d(w) := \min\{d(v), c(v,w) - f(v,w)\}, \quad \textit{und} \quad \pi(w) := v,$$

*Sofern es keine Vorwärtskante, aber eine Rückwärtskante gibt,*

$$(w, v) \in E, \quad \text{mit} \quad f(w, v) > 0,$$

*nehmen wir $w$ ebenfalls in $S$ auf ($S^+ = S \cup w$) und definieren in diesem Fall*

$$d(w) := \min\{d(v), f(w, v)\}, \quad \text{und} \quad \pi(w) := v.$$

*Wir ändern $U$ zu $U \cup v$ und widerholen diesen Schritt, sofern möglich.*

3. *Gilt nun $t \in S$, so gilt $\pi^n(t) = s$ für ein $n \in \mathbb{N}$. Es ist*

$$\gamma = (s, \pi^{n-1}(t), \ldots, \pi(t), t),$$

*ein $f$-zunehmender Weg von $s$ zu $t$, entlang dessen wir den Fluss $f$ ändern zu $f^+$ mit höherem Flusswert $\operatorname{val}(f^+) = \operatorname{val}(f) + d(t)$. Wir beginnen mit Schritt (1) von neuem mit dem Fluss $f^+$.*

4. *Andernfalls gilt $S = U$. es gilt*

$$S := \{v \in V \mid \textit{zu denen es einen } f\textit{-zunehmenden Weg von } s \textit{ aus gibt}\}$$

*Der Schnitt $(S, T)$, mit $T = V \setminus S$ hat eine Kapazität gleich dem Flusswert von $f$:*

$$\operatorname{val} f = c(S, T).$$

*Mit diesem Schritt terminiert der Algorithmus mit einem maximalen Fluss.*

Beweisskizze: Es sind $\pi : S \setminus \{s\} \to S$ und $d : S \to \mathbb{R}$ stets wohldefiniert. Für alle $v \in S$ ist mit $s = \pi^n(v)$

$$\gamma = (s, \pi^{n-1}(v), \ldots, \pi(v), v),$$

stets ein $f$-zunehmender Weg von $s$ zu $v$. Für eine Ecke $v \in U$ sind alle Endpunkte einer Vorwärts- oder Rückwärtskante $(v, x)$ mit $f(v, x) < c(v, x)$ bzw. $f(x, v) > 0$ schon in $S$ enthalten. Daher impliziert $U = S$, daß $(S, T)$ ein Schnitt ist, der $f$ als maximalen Fluss ausweist, wie im letzten Schritt des Algorithmus angegeben.

Die beiden Schleifen des Algorithmus brechen jeweils ab, die Wiederholung ab Start mit verbesserten Fluß $f^+$, weil dessen Flusswert echt zunimmt, $\operatorname{val}(f^+) > \operatorname{val}(f)$. Die Widerholung von Schritt (2), weil hierbei $U$ echt zunimmt.

Somit terminiert der Algorithmus von Ford Fulkerson mit einem maximalen Fluss.

# 9 kürzeste Wege

Sei $G = (V, E)$ ein Graph mit positiven Kantengewichten

$$\omega : E \to \mathbb{R}^{>0}, \quad \text{d.h.} \quad \omega(e) > 0, \quad \text{für alle} \quad e \in E.$$

Ein Weg $\gamma = (v_0, \ldots, v_n)$ durch die Kanten $(e_1, \ldots, e_n)$ mit $e_i = (v_{i-1}, v_i)$ hat die Länge

$$\omega(\gamma) = \sum_{i=1}^{n} \omega(e_i).$$

Sei $s \in V$ eine Ecke. Wir suchen nun im Graphen $G$ nach kürzesten Wegen von $s$ aus zu jeder anderen Ecke in $V$. Für $v \in V$ sei die länge $d_k(v)$ des kürzesten Weges von $s$ nach $v$ definiert durch

$$d_k(v) = \min\{\omega(\gamma) \,|\, \gamma = (v_0, \ldots, v_n) \quad \text{mit} \quad v_0 = s, \, v_n = v\}.$$

**Definition 25 (Bellmannungleichungen)** *Sei $G = (V, E)$ Graph mit positiver Kantengewichtung,*

$$\omega : E \to \mathbb{R}^{>0},$$

*sowie $s \in V$ eine Ecke. Eine Eckengewichtung $d : V \to \mathbb{R}$ erfüllt die Bellmannungleichungen genau dann wenn gilt:*

$$d(s) = 0, \quad \textit{und} \quad d(v) = \min_{u \in N(v)} \{d(u) + \omega(u, v)\} \quad \textit{für} \quad v \neq s.$$

Die Ungleichung $d(v) \leq d(u) + \omega(u, v)$ für $u \in N(v)$ gilt schärfer insofern daß $d(v)$ das Minimum auf der rechten Seite annimmt.

In einem zusammenhängenden Graphen erfüllt die kürzeste Wege Funktion $d_k : V \to \mathbb{R}$ offenbar die Bellmannungleichung.

Sei $d_B : V \to \mathbb{R}$ eine weitere Funktion, die den Bellmannungleichungen genügt. Es gilt dann:

$$d_B \geq d_k.$$

Wir definieren eine Vorgängerfunktion $\pi : V \setminus s \to V$, die für $v \in V \setminus s$ einen solchen Nachbarn $\pi(v) \in N(v)$ designiert, in dem die Bellmannungleichungen das Minimum annehmen:

$$d_B(v) = d_B(\pi(v)) + \omega(\pi(v), v).$$

In der Kette $v, \pi(v), \pi^2(v), \ldots$ nimmt $d_B$ echt ab. Diese Kette bricht ab, für ein $m \in \mathbb{N}$ muß gelten $\pi^m(v) = s$. Es ist

$$\gamma = (s, \pi^{m-1}(v), \ldots, \pi(v), v),$$

ein Weg mit

$$d_k(v) \leq \omega(\gamma) = d_B(v).$$

Somit ist $d_k \leq d_B$ gezeigt. Jede den Bellmannungleichungen genügende Funktion ist tatsächlich die kürzeste Wege Funktion: Sei $v \in V$ ein minimaler Verbrecher hierzu,

$$d_B(v) > d_k(v) \quad \text{und} \quad d_k(v) \quad \text{minimal.}$$

Sei $u \in N(v)$ derjenige Nachbar, für den Gleichheit in Bezug auf $d_k$ gilt:

$$d_B(v) > d_k(v) = d_k(u) + \omega(u,v) = d_B(u) + \omega(u,v) \geq d_B(v).$$

Ein Widerspruch, es folgt $d_B = d_k$.

**Lemma 11** *Sei $G = (V, E)$ ein Graph mit positiver Kantengewichtung $\omega : E \to \mathbb{R}^{>0}$, sowie $s \in V$. Die kürzeste Wege Funktion $d_k$ ist die einzige Funktion $V \to \mathbb{R}$, welche den Bellmannungleichungen*

$$d(s) = 0, \quad \textit{und} \quad d(v) = \min_{u \in N(v)} \{d(u) + \omega(u,v)\} \quad \textit{für} \quad v \neq s.$$

*genügt.*

Auf dieser Charakterisierung kürzester Wege beruht der Algorithmus von Dijkstra zur bestimmung kürzester Wege.

**Algorithmus 3 (Dijkstra)** *Sei $G = (V, E)$ ein Graph, $s \in V$ und*

$$\omega : V^{(2)} \to \mathbb{R} \cup \infty \quad \textit{mit} \quad \omega(u,v) = \infty, \quad \textit{falls} \quad (u,v) \notin E.$$

*Der Algorithmus von Dijkstra initialisiert eine Suchmenge $T := V$ sowie eine Eckenmarkierung $d_T : V \to \mathbb{R} \cup \infty$ durch:*

$$d_T(s) = 0, \quad \textit{und} \quad d_T(v) = \infty, \quad \textit{für} \quad v \neq s.$$

*Der Rekursionsschritt des Algorithmus lautet wie folgt:*

- *solange $T \neq \emptyset$ wähle $u \in T$ mit $d_T(u)$ minimal,*
- *$T^+ := T \setminus u$,*
- $$T^+(v) := \begin{cases} d_T(v) & \textit{für} \quad v \notin T^+ \\ min\{d_T(v), d_T(u) + \omega(u,v)\} & \textit{fürv} \in T^+ \end{cases}$$

*Der Dijkstra Algorithmus entfernt aus $T$ sukzessiv eine Ecke, er terminiert mit $T = \emptyset$ und einer Markierung*

$$d_\emptyset : V \to \mathbb{R},$$

*welche den Bellmannungleichungen genügt. Somit ist mit Lemma (11) $d_\emptyset$ die kürzeste Wege Funktion (von $s$ aus).*

Beweis: Nachdem wir eine Ecke $u$ aus $T$ entfernt haben ändern wir deren Wert $d(u)$ nicht mehr. Für $u \notin T$ gilt $d_T(u) = d_\emptyset(u)$. Desweiteren wird der Wert $d(u)$ nur reduziert. Für $T' \subset T$ gilt $d_{T'} \leq d_T$.

Im ersten Schritt folgt $T = V \setminus s$ daher $d_T(s) = d_\emptyset(s) = 0$.

Sei nun $u, v \in V$, $v \neq s$, so ist zu zeigen

$$d_\emptyset(v) \leq d_\emptyset(u) + \omega(u,v). \tag{17}$$

Sei $u$ zuerst aus $T$ entfernt, für $T^+ = T \setminus u$ gilt $v \in T^+$. Es folgt

$$d_\emptyset(v) \leq d_{T^+}(v) \leq d_T(u) + \omega(u,v) = d_\emptyset(u) + \omega(u,v).$$

Sei nun $v$ zuerst aus $T$ entfernt, und somit $d_\emptyset(v) = d_T(v)$. Wir nehmen an daß (17) nicht gilt, folglich $d_\emptyset(v) > d_\emptyset(u)$ mit $uinT$. Daher existiert eine Teilmenge $T' \subset T$ maximal mit $d_{T'}(\bar{u}) < d_T(v)$ für ein $\bar{u} \in T$. Da $d_T(v)$ minimal in $\{d_T(x) \,|\, x \in T\}$ ist, folgt $T' \subsetneq T$ und für ein $u' \in T$ gilt $T' = T'' \setminus u'$ mit

$$d_{T'}(\bar{u}) = d_{T''}(u') + \omega(\bar{u}, u').$$

Der Fall $d_{T'}(\bar{u}) = d_{T''}(\bar{u})$ kann nicht gelten, wegen der maximalität von $T'$. Wir erhalten ein Widerspruch

$$d_T(v) > d_{T'}(\bar{u}) > d_{T''}(u') \geq d_T(v).$$

Die Ungleichung rechts folgt aus der maximalität von $T'$. Genauer ist in (17) zu zeigen, daß das minimum angenommen wird, d.h Wir müssen noch ausschließen den Fall

$$d_\emptyset(v) < \min_{u \in N(v)} \{d_\emptyset(u) + \omega(u,v)\}.$$

Es folgt $d_\emptyset(v) < \infty$ und $d_\emptyset(v) = d_(u) + \omega(v,u)$ für ein $u \in N(v)$, wegen $\omega(u,v) = \infty$ sonst. q.e.d.

# 10 Matchings

**Definition 26** *Sei $G = (V, E)$ ein Graph. Ein Matching ist eine Teilmenge $M \subset E$ von Kanten, sodaß je zwei Kanten aus $M$ keine gemeinsame Ecken haben. Der Eckengrad $d_M$ in Bezug auf den Graph $(V, M)$ ist überall $\leq 1$:*

$$d_M(v) \leq 1 \quad \textit{für alle} \quad v \in V.$$

*Ein Matching $M \subset E$ heißt perfekt, wenn es alle Ecken sättigt. (Eine Kantenmenge $T \subset E$ sättigt eine Eckenmenge $U \subset V$, falls alle $u \in U$ auf einer Kante $e \in T$ liegen, $e = (u, v)$ für ein $v \in V$). Der Eckengrad $d_M$ ist genau 1*

$$d_M(v) = 1, \quad \textit{für alle} \quad v \in V.$$

*Sei $\omega : E \to \mathbb{R}$ eine Kantengewichtung. Ein Matching $M \subset E$ heißt maximal (in bezug auf $\omega$), falls für alle weiteren Matchings $M'$ von $G$ gilt:*

$$\omega(M) \geq \omega(M').$$

Sei $G = (V, E)$ Graph mit Gewichtsfunktion $\omega : E \to \mathbb{R}$. Eine zulässige Eckenmarkierung $l : V \to \mathbb{R}$ erfüllt:

$$l(u) + l(v) \geq \omega(u, v) \quad \text{für alle} \quad (u, v) \in E.$$

Der zugehörige Gleichgewichtsgraph $G_l = (V, E_l)$ ist definiert durch:

$$E_l := \{(u, v) \in E \,|\, l(u) + l(v) = \omega(u, v)\}.$$

Es gilt sodann folgendes Optimalitätskriterium für Matchings:

**Lemma 12** *Sei $G = (V, E)$ Graph, $\omega : E \to \mathbb{R}$ Kantengewichtung, $l : V \to \mathbb{R}$ zulässige Eckenmarkierung mit Gleichgewichtsgraph $E_l$. Ist $M \subset E_l$ ein perfektes Matching, dann ist $M$ ein maximales(optimales) Matching von $G$.*

Beweis: Für ein weiteres Matching $M' \subset E$ gilt

$$\omega(M') = \sum_{e \in M'} \omega(e) \leq \sum_{v \in V} l(v) = \omega(M). \quad \text{q.e.d.}$$

Sei $G = (V, E)$ ein Graph und $M \subset E$ ein Matching.

**Definition 27 ($M$-alternierende, $M$-augmentierende Wege)** *In einem $M$-alternierender Weg $\gamma = (v_0, \ldots, v_n)$ gehören die Kanten $(v_{i-1}, v_i)$ abwechselnd zu $M$:*

$$(v_{i-1}, v_i) \in M \iff (v_i, v_{i+1}) \notin M \quad \textit{für } 1 \leq i < n.$$

*Ist $n$ ungerade und $v_0, v_n$ sind $M$ ungesättigt, dann ist $\gamma$ ein $M$-augmentierender Weg.*

Wir geben nun den Algorithmus von Kuhn und Munkres an, der auf einem bipartiten Graphen ein maximales Matching findet.

**Algorithmus 4 (Ungarischer Algorithmus, Algorithmus von Kuhn Munkres)** *Sei $G = (V, E)$ vollständiger bipartiter Graph $V = X\dot{\cup}Y$, mit $|X| = |Y|$ und Kantenmenge $E = X \times Y$. Ferner existiert eine Kantengewichtung $\omega : E \to \mathbb{R}$. Der Algorithmus gliedert sich in folgende Schritte:*

1. *Initialisiere eine zulässige Eckenmarkierung $l : V \to R$, sowie ein Matching $M \subset E_l$, das im Gleichgewichtsgraphen enthalten ist.*

2. *Ist $M \subset E_l$ ein Matching, das $X$ sättigt, dann stoppt der Algorithmus. Nach obigem Optimalitätskriterium ist $M$ eim maximales Matching.*

3. *Sonst wähle eine Ecke $u \in X$, die nicht von $M$ gesättigt ist, und setze $S := \{u\}$, $S \subset X$ sowie $T = \emptyset$, $T \subset Y$.*

4. *Gilt $N_{E_l}(S) \supsetneqq T$ dann gehe zu (5). Im Fall $N_{E_l}(S) = T$ existiert ein*

$$\alpha := \min_{x \in S, y \notin T} \{l(x) + l(y) - \omega(x, y)\} > 0.$$

*Wir erhalten eine neue zulässige Eckenmarkierung $l^+$ wie folgt:*

$$l^+(v) := \begin{cases} l(v) - \alpha & \textit{für} \quad v \in S, \\ l(v) + \alpha & \textit{für} \quad v \in T, \\ l(v) & \textit{sonst.} \end{cases}$$

*Die Nachbarschaft von $S$ hat sich vergrössert:*

$$N_{E_{l^+}}(S) \supsetneqq N_{E_l}(S).$$

*Nun gehen wir zu (4) zurück.*

5. *Wähle $y \in N_{E_l}(S) \setminus T$,*

   - *Ist $y$ $M$-gesättigt, dann wähle Kante $(x, y) \in M$, und setze*

   $$S^+ := S \cup x, \quad \textit{und} \quad T^+ := T \cup \{y\}.$$

   *Sodann geht der Algorithmus zu Schritt (4).*

- *Ist $y$ $M$-ungesättigt, dann gibt es einen $M$-augmentierenden Weg*

$$\gamma = (v_0, \ldots, v_n),$$

*von $v_0 = u$ zu $v_n = y$ innerhalb des Gleichgewichtsgraphen $E_l$. Wir ersetzen die Kanten von $\gamma \cap M$ durch diejenigen von $\gamma \setminus M$ in $M$:*

$$M^+ = M \setminus \{(v_{i-1}, v_i) \,|\, i \quad \textit{gerade}\} \cup \{(v_{i-1}, v_i) \,|\, i \quad \textit{gerade}\}$$

*Sodann gehe zu (2).*

Beweisskizze: Der Ungarische Algorithmus (von Kuhn Munkres) ist durchführbar, in jedem Schritt sind die implizit gemachten Voraussetzungen erfüllt. Dies sind die folgenden:

1. Es gilt stets $T \subset N_{E_l}(S)$, Die Nachbarschaft der Menge $S \subset X$ umfasst stets die Teilmenge $T$ der Ecken in $Y$.
2. Es gilt stets $S \neq \emptyset$ und $T \subsetneqq Y$ in Schritt (4).
3. Von der Ecken $u \in s$ aus existtiert stets ein $M$-alternierender Weg innerhalb von $E_l \cap (S \cup T)$ zu allen $s \in S$.
4. Es ist $l$ stets eine zulässige Eckenmarkierung.
5. Es gilt $M \subset E_l$.

Sofern diese Punkte nachgewiesen sind, ist der Ungarische Algorithmus durchführbar.

Da in den Übergängen zwischen den einzelnen Schritten entweder $|M|$ echt zunimmt, oder $|M|$ nicht kleiner wird, aber $|N_{E_l}(S)|$ oder $|T|$ echt zunehmen, Terminiert der Algorithmus mit einem maximalen Matching.