Kapitel 4

Approximations- und Optimierungsaufgaben

Ganz allgemein versteht man unter einer *Approximationsaufgabe* das folgende Problem. Gegeben sei ein linearer normierter Raum $(X, \|\cdot\|)$ (also ein linearer Raum X, dessen Skalarkörper grundsätzlich der Körper $\mathbb{R}$ der reellen Zahlen sei und eine Norm $\|\cdot\|$ auf X), eine Menge $M \subset X$ und ein $z \in X$. Gesucht ist ein Element $x^* \in M$, welches unter allen Elementen aus M zu dem vorgegebenen Element z bezüglich der Norm $\|\cdot\|$ den kleinsten Abstand besitzt. Dagegen besteht eine *Optimierungsaufgabe* darin, eine auf einer gewissen Menge M (der sogenannten Menge der zulässigen Lösungen) definierte reellwertige Funktion f (die sogenannte Ziel- oder Kostenfunktion) zu minimieren. Wir wollen versuchen, einen Eindruck über die Vielfalt möglicher Aufgabenstellungen zu geben. Ferner sollen einige interessante und wichtige Resultate wenigstens motiviert und der Einsatz mathematischer Anwendersysteme erprobt werden.

4.1 Approximationsaufgaben

Eine Approximationsaufgabe ist, wie zu Beginn schon angedeutet wurde, durch die folgenden Daten gegeben:

1. Der lineare normierte Raum $(X, \|\cdot\|)$: Der Raum, "in dem sich alles abspielt".
2. Eine Menge $M \subset X$: Die Menge der Elemente, mit denen approximiert wird.
3. Ein Element $z \in X$: Das Objekt, das approximiert bzw. angenähert werden soll.

Die Approximationsaufgabe besteht in

$$\text{(P)} \qquad \text{Minimiere} \quad \|x - z\|, \quad x \in M.$$

Gesucht ist also ein $x^* \in M$ mit $\|x^* - z\| \leq \|x - z\|$ für alle $x \in M$, die *beste Approximierende* an z bezüglich M. Man erkennt, dass es sich hier um eine spezielle Optimierungsaufgabe handelt. Wie bisher stets wollen wir auch diesen Abschnitt mit Beispielen beginnen, bringen aber ausnahmsweise hier schon einen grundlegenden Satz, der nicht recht in die nächsten Unterabschnitte passen würde.

Satz 1.1 *Gegeben sei die Approximationsaufgabe, die durch den linearen normierten Raum $(X, \|\cdot\|)$, die Menge $M \subset X$ und das Element $z \in X$ gegeben ist. Dann gilt:*

1. *Ist $M \subset X$ ein endlichdimensionaler linearer Teilraum von X, so besitzt das Approximationsproblem eine Lösung.*

2. *Die Abgeschlossenheit von M ist eine notwendige Bedingung dafür, dass das Approximationsproblem für jedes $z \in X$ eine Lösung besitzt.*

Beweis: Für den ersten Teil nehmen wir an, $M \subset X$ sei ein n-dimensionaler linearer Teilraum von X. Sei etwa $M = \operatorname{span}\{v_1, \ldots, v_n\}$, also $\{v_1, \ldots, v_n\}$ eine Basis von M. Für $a = (a_j) \in \mathbb{R}^n$ definiere $f(a) := \|\sum_{j=1}^n a_j v_j - z\|$. Die hierdurch auf dem $\mathbb{R}^n$ definierte reellwertige Abbildung f ist stetig. Bei der Suche nach einem Minimum von f auf dem $\mathbb{R}^n$ kann man sich auf die Niveaumenge $L_0 := \{a \in \mathbb{R}^n : f(a) \le f(0)\}$ beschränken. Wegen der Stetigkeit der Funktion f ist L_0 abgeschlossen. L_0 ist aber auch beschränkt. Denn die Abbildung $a \mapsto \|\sum_{j=1}^n a_j v_j\|$ ist eine Norm auf dem $\mathbb{R}^n$. Da je zwei Normen auf dem $\mathbb{R}^n$ äquivalent sind, existiert eine Konstante $c > 0$ mit $\|\sum_{j=1}^n a_j v_j\| \ge c\,\|a\|_\infty$ für alle $a \in \mathbb{R}^n$. Für ein beliebiges $a \in L_0$ ist daher

$$c\,\|a\|_\infty \le \Big\|\sum_{j=1}^n a_j v_j\Big\| \le \Big\|\sum_{j=1}^n a_j v_j - z\Big\| + \|z\| = f(a) + \|z\| \le 2\|z\|.$$

Daher ist L_0 in einer Kugel um den Nullpunkt mit dem Radius $2\|z\|/c$ enthalten und folglich beschränkt. Insgesamt ist L_0 kompakt. Daher nimmt die stetige Funktion auf der kompakten Menge L_0 ihr Minimum in einem $a^* \in L_0$ an. Daher ist $x^* := \sum_{j=1}^n a_j^* v_j$ beste Approximierende an z bezüglich M.

Für jedes $z \in X$ existiere eine beste Approximierende an z bezüglich M. Wir zeigen, dass dann M notwendigerweise abgeschlossen ist. Denn sei $z \in \operatorname{cl}(M)$, also z ein Element aus dem Abschluss von M. Dann ist $\inf_{x\in M} \|x - z\| = 0$, da ja wegen $z \in \operatorname{cl}(M)$ eine Folge $\{x_k\} \subset M$ mit $\|x_k - z\| \to 0$ existiert. Da aber nach Voraussetzung eine beste Approximierende an z bezüglich M existiert, existiert ein $x^* \in M$ mit $\|x^* - z\| = 0$. Folglich ist $z = x^* \in M$. Jedes Element aus dem Abschluss von M gehört also selbst zu M. Das ist die Abgeschlossenheit von M. □

4.1.1 Beispiele

Beispiel: Bei[1] einer Nivellierung sollen die Höhen x_1, x_2, x_3, x_4 von vier Niveaus bestimmt werden. Es wurden einerseits diese Höhen gemessen und andererseits auch die 6 Höhenunterschiede zwischen den 4 Niveaus. Man erhält das folgende überbestimmte

[1] Dieses Beispiel haben wir

E. Stiefel (1965) *Einführung in die Numerische Mathematik*. B. G.Teubner, Stuttgart

entnommen.

lineare Gleichungssystem:

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix}}_{=:A} \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}}_{=:x} = \underbrace{\begin{pmatrix} 3.47 \\ 2.01 \\ 1.58 \\ 0.43 \\ 1.42 \\ 1.92 \\ 3.06 \\ 0.44 \\ 1.53 \\ 1.20 \end{pmatrix}}_{=:b}.$$

Dieses lineare Gleichungssystem $Ax = b$ ist nicht lösbar, wie man sofort feststellt. Um eine "möglichst gute" Lösung zu bestimmen, minimiert man den Defekt $Ax - b$, z. B. bezüglich der euklidischen Norm. Hierdurch kommt man zum *linearen Ausgleichsproblem*, $\|Ax - b\|_2$ zu minimieren. Ist $A \in \mathbb{R}^{m \times n}$ (in unserem Beispiel ist $m = 10$ und $n = 4$), so ordnet sich diese Aufgabe der allgemeinen Problemstellung unter, wenn man $(X, \|\cdot\|) := (\mathbb{R}^m, \|\cdot\|_2)$, $M := R(A)$ (Range oder Wertebereich von A) und $z := b$ setzt[2]. Eine Lösung mit Maple ist einfach:

```
> with(LinearAlgebra):
> A:=Matrix([[1,0,0,0],[0,1,0,0],[0,0,1,0],[0,0,0,1],[1,-1,0,0],[1,0,-1
> ,0],[1,0,0,-1],[0,1,-1,0],[0,1,0,-1],[0,0,1,-1]]):
> b:=<3.47,2.01,1.58,0.43,1.42,1.92,3.06,0.44,1.53,1.20>:
> x:=LeastSquares(A,b);
```

$$x := \begin{bmatrix} 3.47200000000000042 \\ 2.01000000000000068 \\ 1.58200000000000029 \\ .426000000000000101 \end{bmatrix}$$

Man vermutet zu Recht, dass der "Schmutz" in den letzten Dezimalen auf Rundungsfehler zurückzuführen sind. Dass dies wirklich so ist, erkennt man an

```
> b:=(1/100)*<347,201,158,43,142,192,306,44,153,120>:
> x:=LeastSquares(A,b);
```

$$x := \begin{bmatrix} \frac{434}{125} \\ \frac{201}{100} \\ \frac{791}{500} \\ \frac{213}{500} \end{bmatrix}$$

```
> Digits:=20:
> evalf(x);
```

[2] Allerdings ist man weniger an der besten Approximierenden $y^* \in R(A)$, sondern einem Urbild x^* interessiert.

$$\begin{bmatrix} 3.4720000000000000000 \\ 2.0100000000000000000 \\ 1.5820000000000000000 \\ .42600000000000000000 \end{bmatrix}$$

Eine Lösung mit MATLAB ist natürlich auch leicht möglich, wir gehen hierauf nicht mehr ein. □

Allgemein spricht man von einem linearen Ausgleichs- oder Least Squares-Problem, wenn eine Matrix $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$ sowie ein Vektor $b \in \mathbb{R}^m$ gegeben sind und das Problem

$$\text{(P)} \qquad \text{Minimiere} \quad \|Ax - b\|_2, \quad x \in \mathbb{R}^n,$$

zu lösen ist. Wir werden auf dieses Problem, eines der häufigsten auftretenden mathematischen Probleme in der Praxis, im folgenden Unterabschnitt noch etwas genauer eingehen.

Beispiel: Sei $I := [\frac{1}{2}, 1]$. Gesucht sei ein lineares Polynom, welches die Quadratwurzel $\sqrt{t}$ auf dem Intervall I in dem Sinne am besten approximiert, dass der auf I betragsmaximale relative Fehler minimal ist. Gesucht sei also eine Lösung der Aufgabe

$$\text{(P)} \qquad \text{Minimiere} \quad \max_{t\in I} \frac{|p(t) - \sqrt{t}|}{\sqrt{t}}, \quad p \in \mathcal{P}_1,$$

wobei $\mathcal{P}_1$ die Menge der Polynome vom Grad ≤ 1 bezeichnet. Eine Einordnung in das allgemeine Approximationsproblem ist auch hier einfach. Der lineare normierte Raum $(X, \|\cdot\|)$, in dem sich alles abspielt, ist $X := C(I)$, die Menge der auf dem Intervall I stetigen und reellwertigen Funktionen versehen mit der Norm $\|x\| := \max_{t\in I} |x(t)|/\sqrt{t}$. Die Menge, mit der approximiert wird, ist $M := \mathcal{P}_1$ und das zu approximierende Element ist $z(t) := \sqrt{t}$.

Dies ist ein Problem, das Maple behandeln kann. Im `numapprox`-package steht die Funktion `minimax` zur Verfügung. In dieser ist das erste Argument die zu approximierende Funktion, dann kommt das Intervall, auf dem approximiert wird, dann schließlich (es handelt sich um rationale Approximation) Zählergrad und (optional) Nennergrad, schließlich (optional) eine Gewichtsfunktion w (in unserem Fall ist $w(t) = 1/sqrtt$). So erhält man z. B.

```
> with(numapprox):
> Digits:=20:
> z:=proc(t) evalf(sqrt(t)) end proc:
> w:=proc(t) evalf(1/sqrt(t)) end proc:
> p_rel:=minimax(z(t),t=0.5..1,1,w(t));
```

$$p\_rel := .41730759963599880980 + .59016206708659117948\,t$$

```
> p_abs:=minimax(z(t),t=0.5..1,1);
```

$$p\_abs := .42049512883458866117 + .58578643762690495120\,t$$

Mit dem plot-Befehl könnte man diese Funktionen oder die relativen bzw. absoluten Fehler plotten, worauf wir verzichten wollen.

Wir wollen nun die oben mit Hilfe von Maple erhaltenen Ergebnisse auf anderem Wege erhalten. Für eine Lösung machen wir jeweils den Ansatz $p(t) = \alpha + \beta t$ und definieren anschließend in Abhängigkeit von den Parametern α, β (ohne das extra kenntlich zu machen) den relativen bzw. absoluten Defekt:

$$d_{\mathrm{rel}}(t) := \frac{\alpha + \beta t - \sqrt{t}}{\sqrt{t}}, \qquad d_{\mathrm{abs}}(t) := \alpha + \beta t - \sqrt{t}.$$

Wir bestimmen in beiden Fällen $\alpha, \beta \in \mathbb{R}$ und $\hat{t} \in (\frac{1}{2}, 1)$ so, dass

$$d(\tfrac{1}{2}) = d(1), \qquad d'(\hat{t}) = 0, \qquad d(\hat{t}) = -d(1).$$

Aus der ersten Bedingung, dass nämlich der Defekt an den Intervallgrenzen den selben Wert besitzt, erhält man $\alpha = \beta/\sqrt{2}$ bzw. $\beta = 2 - \sqrt{2}$. Die zweite Bedingung liefert $\hat{t} = 1/\sqrt{2}$ bzw. $\hat{t} = 1/[4(2-\sqrt{2})^2]$. Die letzte Bedingung ergibt dann schließlich

$$\alpha = \frac{2}{2^{5/4} + 1 + \sqrt{2}} \qquad \text{bzw.} \qquad \alpha = \frac{3}{8}\Big(\frac{3}{2}\sqrt{2} - 1\Big).$$

Die Koeffizienten des linearen Polynoms $p(t) = \alpha + \beta t$ sind also gegeben durch

$$\alpha = \frac{2}{2^{5/4} + 1 + \sqrt{2}}, \qquad \beta = \frac{2^{3/2}}{2^{5/4} + 1 + \sqrt{2}}$$

bzw.

$$\alpha = \frac{3}{8}\Big(\frac{3}{2}\sqrt{2} - 1\Big), \qquad \beta = 2 - \sqrt{2}.$$

Diese Werte stimmen mit den oben numerisch erhaltenen Werten überein. Wir werden später beweisen können, dass hierdurch wirklich Lösungen der entsprechenden Probleme gegeben sind. Diese sind sogar eindeutig, wie man ebenfalls beweisen kann. □

Bemerkung: Allgemein spricht man von einer univariaten (d. h. es gibt nur eine univariate Variable) linearen Tschebyscheffschen Approximationsaufgabe, wenn die Daten des Approximationsproblems folgendermaßen gegeben sind:

1. Es ist $(X, \|\cdot\|) := (C[a,b], \|\cdot\|_\infty)$, wobei $C[a,b]$ den linearen Raum der auf dem Intervall $[a,b]$ stetigen, reellwertigen Funktionen bezeichnet und $\|\cdot\|_\infty$ die Maximum- bzw. Tschebyscheffnorm auf $C[a,b]$ bedeutet. Lässt man noch eine positive Gewichtsfunktion $w \in C[a,b]$ zu, so ist also

   $$\|x\|_\infty := \max_{t\in[a,b]} (w(t)\,|x(t)|).$$

   Im obigen Beispiel ist $[a,b] = [\frac{1}{2}, 1]$ und $w(t) := 1/\sqrt{t}$ bzw. $w(t) := 1$.

2. Es ist $M \subset C[a,b]$ ein endlichdimensionaler linearer Raum. Z. B. ist $M = \mathcal{P}_n$ der $(n+1)$-dimensionale lineare Raum der Polynome vom Grad $\le n$. Im obigen Beispiel ist $n = 1$.

3. Es ist $z \in C[a,b]$ die zu approximierende Funktion. Im obigen Beispiel ist $z(t) := \sqrt{t}$.

Ist die Menge M derjenigen Funktionen, mit denen approximiert wird, die Menge $\mathcal{R}_{m,n}$ der rationalen[3] Funktionen mit Zählergrad m und Nennergrad n, so ist dies natürlich (für $n \geq 1$) kein linearer Raum mehr. Trotzdem kann einiges analog der linearen Theorie entwickelt werden, was aber den Rahmen dieser Vorlesung bei weitem sprengen würde. Die `minimax`-Funktion in Maple kann sogar rationale beste Approximierende berechnen. In der folgenden Abbildung 4.1 haben wir links den absoluten Fehler bei

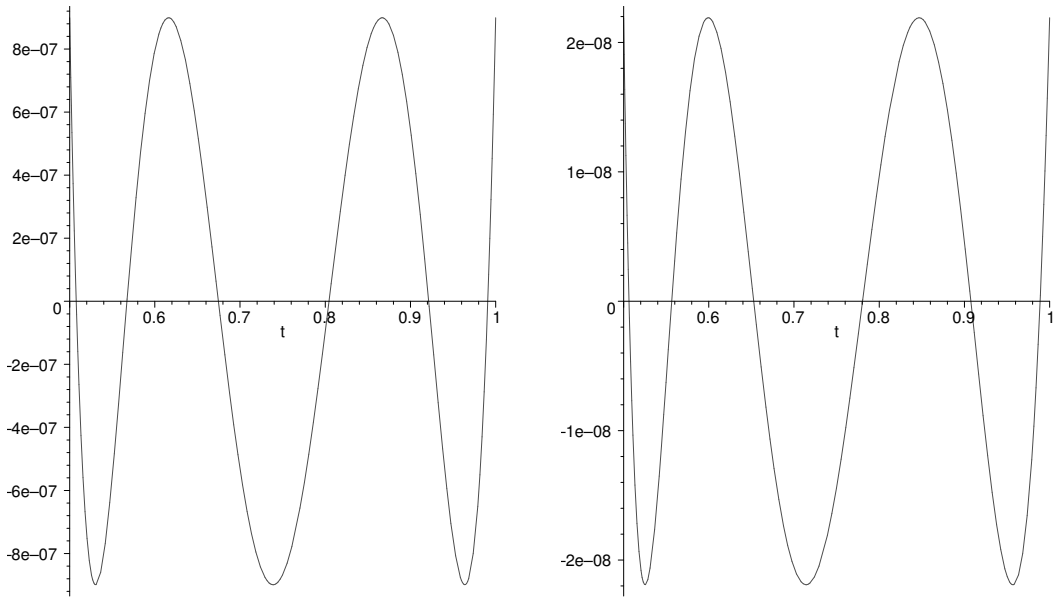

Abbildung 4.1: Beste Approximation von $z(t) := \sqrt{t}$ auf $[\frac{1}{2}, 1]$ bezüglich $\mathcal{P}_5$ und $\mathcal{R}_{3,2}$

der Approximation von $z(t) := \sqrt{t}$ auf $[\frac{1}{2}, 1]$ bezüglich $\mathcal{P}_5$ (Polynom-Approximation) aufgetragen, rechts findet man das entsprechende Ergebnis für die Approximation bezüglich $\mathcal{R}_{3,2}$ (rationale Approximation). Man erkennt sehr deutlich, dass der Defekt jeweils seinen Maximalbetrag mit alternierendem Vorzeichen annimmt. □

Beispiel: Im letzten Beispiel wurde $z(t) := \sqrt{t}$ durch ein Polynom vom Grad ≤ 1 so approximiert, dass der maximale betragsmäßige Fehler auf dem Intervall $[\frac{1}{2}, 1]$ minimal ist. Dagegen handelt es sich bei der *Approximation im Mittel*, angewandt auf das entsprechende Problem, um die Aufgabe

$$\text{Minimiere} \quad \Big(\int_{1/2}^{1} [p(t) - \sqrt{t}]^2 \, dt \Big)^{1/2}, \quad p \in \mathcal{P}_1.$$

Macht man den Ansatz $p(t) = \alpha + \beta t$, so erhält man die äquivalente Aufgabe

$$\text{Minimiere} \quad f(\alpha, \beta) := \frac{1}{2} \int_{1/2}^{1} (\alpha + \beta t - \sqrt{t})^2 \, dt, \quad (\alpha, \beta) \in \mathbb{R} \times \mathbb{R}.$$

[3]Rationale Funktionen sind gerade die Funktionen, die alleine mit Hilfe der vier Grundrechenarten ausgewertet werden können.

Bei der Berechnung von $f(\alpha, \beta)$ machen wir es uns einfach und benutzen Maple. Wir erhalten

$$f(\alpha,\beta) = -\frac{2}{3}\alpha - \frac{2}{5}\beta + \frac{3}{16} + \frac{3}{8}\alpha\beta + \frac{1}{4}\alpha^2 + \frac{7}{48}\beta^2 + \frac{\sqrt{2}}{6}\alpha + \frac{\sqrt{2}}{20}\beta.$$

Notwendig (und auch hinreichend) dafür, dass f in (α, β) ein Minimum besitzt, ist, dass der Gradient bzw. die partiellen Ableitungen verschwunden. Auch bei der Berechnung dieser partiellen Ableitungen (zur Vermeidung von Rechen- oder Flüchtigkeitsfehlern) hilft Maple. Wir erhalten

$$\nabla f(\alpha,\beta) = \begin{pmatrix} \dfrac{\partial f}{\partial \alpha}(\alpha,\beta) \\ \dfrac{\partial f}{\partial \beta}(\alpha,\beta) \end{pmatrix} = \begin{pmatrix} -\dfrac{2}{3} + \dfrac{3}{8}\beta + \dfrac{1}{2}\alpha + \dfrac{\sqrt{2}}{6} \\ -\dfrac{2}{5} + \dfrac{3}{8}\alpha + \dfrac{7}{24}\beta + \dfrac{\sqrt{2}}{20} \end{pmatrix}.$$

Die Bedingung $\nabla f(\alpha, \beta) = 0$ führt also auf das lineare Gleichungssystem

$$\begin{pmatrix} \frac{1}{2} & \frac{3}{8} \\ \frac{3}{8} & \frac{7}{24} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \frac{2}{3} - \frac{\sqrt{2}}{6} \\ \frac{2}{5} - \frac{\sqrt{2}}{20} \end{pmatrix},$$

woraus man

$$\alpha = \frac{2}{15}(64 - 43\sqrt{2}), \qquad \beta = \frac{6}{5}(6\sqrt{2} - 8)$$

erhält. In Abbildung 4.2 haben wir links den Fehler der Approximation im Mittel

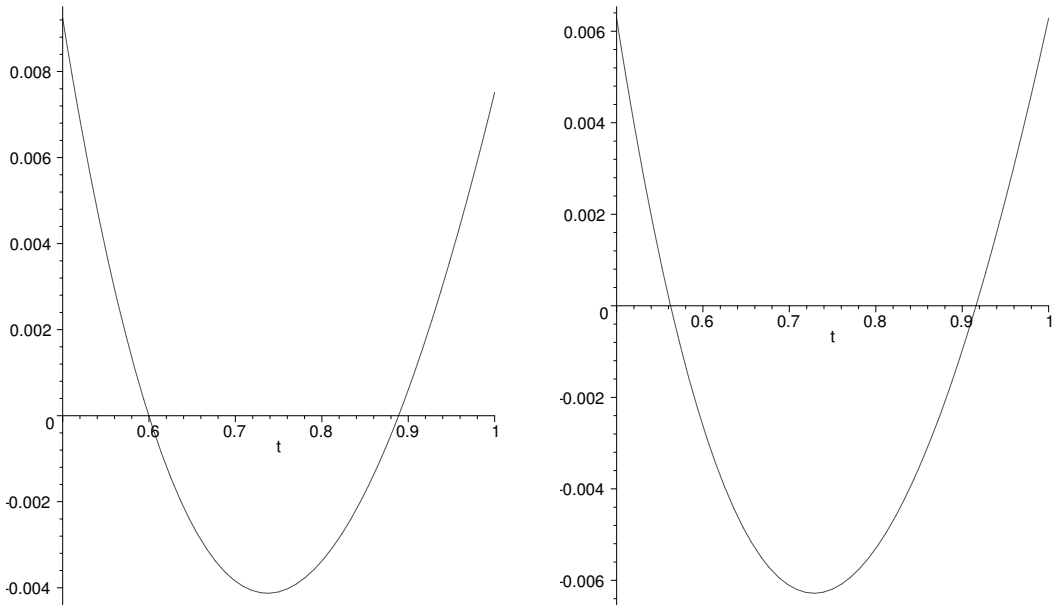

Abbildung 4.2: Fehler bei der Approximation im Mittel bzw. der Tschebyscheff-Approximation

und rechts den entsprechenden Fehler bei der Tschebyscheff-Approximation über dem Intervall $[\frac{1}{2}, 1]$ aufgetragen. □

Bemerkung: Das letzte Beispiel kann wesentlich verallgemeinert werden. Man kann sich nämlich ein Approximationsproblem mit den folgenden Daten vorstellen:

1. Sei $(X, (\cdot,\cdot))$ ein Prä-Hilbertraum, also X eine linearer Raum und $(\cdot,\cdot)$ ein inneres Produkt auf X, die Norm auf X also erzeugt durch $\|x\| := (x,x)^{1/2}$. In unserem obigen Beispiel ist $X = C([\frac{1}{2}, 1])$ und das innere Produkt gegeben durch

$$(x,y) := \int_{1/2}^{1} x(t)y(t)\,dt.$$

2. Es ist $M = \operatorname{span}\{v_1, \ldots, v_n\}$ ein n-dimensionaler linearer Raum.

3. Es ist $z \in X$ das zu approximierende Element.

Das Problem, z im Mittel durch Elemente aus M zu approximieren, ist äquivalent zu:

$$\text{Minimiere} \quad f(a) := \frac{1}{2}\Big\|\sum_{j=1}^{n} a_j v_j - z\Big\|^2, \quad a \in \mathbb{R}^n.$$

Nun ist

$$\begin{aligned} f(a) &= \frac{1}{2}\Big\|\sum_{j=1}^{n} a_j v_j - z\Big\|^2 \\ &= \frac{1}{2}\Big(\sum_{i=1}^{n} a_i v_i - z, \sum_{j=1}^{n} a_j v_j - z\Big) \\ &= \frac{1}{2}\sum_{i,j=1}^{n}(v_i, v_j)a_i a_j - \sum_{i=1}^{n} a_i (v_i, z) + \frac{1}{2}\|z\|^2 \\ &= \frac{1}{2}a^T V a - b^T a + \frac{1}{2}\|z\|^2, \end{aligned}$$

wobei $V := ((v_i, v_j))_{1\le i,j\le n}$ und $b := ((v_i, z))_{1\le i\le n}$. Die Matrix $V \in \mathbb{R}^{n\times n}$ wird eine *Gramsche Matrix* genannt, sie ist symmetrisch und positiv definit (Beweis?), insbesondere also nichtsingulär. Daher ist $x = \sum_{j=1}^{n} a_j v_j$ genau dann eine Lösung der Approximationsaufgabe, $z \in X$ im Mittel durch Elemente aus $M = \operatorname{span}\{v_1, \ldots, v_n\}$ zu approximieren, wenn $Va = b$ bzw. $a = V^{-1}b$. Aus Zeitgründen können wir auf die Approximation in Präe-Hilberträumen (also Räumen, bei denen die Norm durch ein inneres Produkt erzeugt ist) nicht näher eingehen. Siehe aber die Aufgaben □

Bisher haben wir, wenn man einmal von der kurz erwähnten rationalen Tschebyscheff-Approximation absieht, nur *lineare* Approximationsaufgaben betrachtet, also Aufgaben, bei denen die Menge $M \subset X$ derjenigen Elemente, mit denen approximiert wird, ein linearer Teilraum von X ist. Natürlich sind auch nichtlineare Approximationsaufgaben denkbar und sinnvoll. Auf nichtlineare Ausgleichsprobleme (nonlinear least square

fit) gehen wir im Abschnitt über Optimierungsaufgaben ein, da eine solche Aufgabe am besten als eine spezielle unrestringierte Optimierungsaufgabe formuliert wird. So wird z. B. in der univariaten Tschebyscheff-Approximation neben der Approximation mit rationalen Funktionen auch die Approximation mit sogenannten Exponentialsummen betrachtet. Hierauf einzugehen würde den Rahmen der Vorlesung bei weitem sprengen.

4.1.2 Lineare Ausgleichsprobleme

Wie schon wiederholt gesagt, besteht ein lineares Ausgleichs- bzw. Least Squares-Problem darin, bei gegebenen $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$ und $b \in \mathbb{R}^m$, die Aufgabe

$$\text{(P)} \qquad \text{Minimiere} \quad \|Ax - b\|_2, \quad x \in |R^n$$

zu lösen.

Wir formulieren im folgenden Satz die wichtigsten theoretischen Aussagen zum linearen Ausgleichsproblem.

Satz 1.2 *Seien $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$ und $b \in \mathbb{R}^m$ und hiermit das lineare Ausgleichsproblem*

$$\text{(P)} \qquad \textit{Minimiere} \quad f(x) := \frac{1}{2}\|Ax - b\|_2^2, \quad x \in \mathbb{R}^n,$$

gegeben. Dann gilt:

1. *Das lineare Ausgleichsproblem (P) besitzt eine Lösung, d. h. es existiert ein $x^* \in \mathbb{R}^n$ mit $f(x^*) \leq f(x)$ für alle $x \in \mathbb{R}^n$.*
2. *Ein x^* ist genau dann Lösung des linearen Ausgleichsproblems (P), wenn es Lösung des linearen Gleichungssystem $A^TAx = A^Tb$ ist, des Systems der Normalgleichungen.*
3. *Das lineare Ausgleichsproblem (P) ist genau dann eindeutig lösbar, wenn die Spalten von A linear unabhängig sind, also* Rang $(A) = n$ *gilt.*
4. *Unter allen Lösungen von (P) gibt es genau eine mit minimaler euklidischer Norm.*

Beweis: Für einen Beweis des ersten Teiles beachten wir, dass es sich bei einem linearen Ausgleichsproblem (P) darum handelt, auf $(\mathbb{R}^m, \|\cdot\|_2)$ eine beste Approximierende an b bezüglich des Bildraumes $R(A)$ zu bestimmen. Nun ist aber $R(A) \subset \mathbb{R}^m$ ein endlichdimensionaler linearer Teilraum. Wegen Satz 1.1 folgt die Existenz einer Lösung.

Ist x^* eine Lösung von (P), so ist

$$0 = \nabla f(x^*) = A^T(Ax^* - b)$$

bzw. x^* eine Lösung des Systems der Normalgleichngen. Sei umgekehrt $\nabla f(x^*) = 0$ bzw. x^* eine Lösung des Systems der Normalgleichungen. Für ein beliebiges $x \in \mathbb{R}^n$ ist

$$f(x) \;=\; f(x^*) + \underbrace{\nabla f(x^*)^T(x - x^*)}_{=0} + \frac{1}{2}(x - x^*)^T A^T A(x - x^*)$$

$$\begin{aligned} &= f(x^*) + \underbrace{\frac{1}{2}\|A(x-x^*)\|_2^2}_{\geq 0} \\ &\geq f(x^*), \end{aligned}$$

also x^* eine Lösung von (P).

Ist Rang $(A) = n$, so ist $A^TA \in \mathbb{R}^{n\times n}$ symmetrisch und positiv definit, daher die Normalgleichungen und damit auch (P) eindeutig lösbar. Ist umgekehrt Rang $(A) < n$, so ist Kern $(A) \neq \{0\}$ und daher die Normalgleungen und folglich auch (P) nicht eindeutig lösbar.

Sei $\mathcal{L}$ die Menge der Lösungen von (P) bzw. des Systems der Normalgleichungen. Dies ist ein affin linearer Teilraum des $\mathbb{R}^n$, also ein verschobener linearer Raum. In $\mathcal{L}$ gibt es genau ein Element mit minimaler euklidischer Norm, nämlich die orthogonale Projektion des Nullpunktes auf $\mathcal{L}$. Der letzte Teil des Satzes ist bewiesen. □

Auf die numerische Lösung eines linearen Ausgleichsproblems waren wir in Abschnitt 2.1 im Zusammenhang mit der QR-Zerlegung einer Matrix schon eingegangen, wobei allerdings vorausgesetzt werden muss, dass Rang $(A) = n$. Eine kurze Wiederholung: Bekannt sei eine Zerlegung

$$A = Q \begin{pmatrix} \hat{R} \\ 0 \end{pmatrix},$$

wobei $Q \in \mathbb{R}^{m\times m}$ orthogonal und $\hat{R} \in \mathbb{R}^{n\times n}$ eine obere Dreiecksmatrix ist, die wegen (nach Voraussetzung) Rang $(A) = n$ nichtsingulär ist. Einsetzen dieser Zerlegung in die Normalengleichung $A^TAx = A^Tb$ ergibt unter Benutzung von $c := (Q^Tb)(1:n)$ (d. h. der Vektor c besteht aus den ersten n Komponenten von Q^Tb), dass die (eindeutige) Lösung x aus $\hat{R}^T\hat{R}x = \hat{R}^Tc$ bzw. $\hat{R}x = c$ zu bestimmen ist. Bei sogenannten *rangdefizienten* Problemen (bei diesen ist Rang $(A) < n$ bzw. die Spalten von A sind linear abhängig) oder solchen, bei denen das "fast" der Fall ist kann die QR Zerlegung in der früher angegebenen Form nicht angewandt werden. Ein geeignetes Hilfsmittel ist dann die sogenannte *Singulärwertzerlegung*. Wir definieren:

Definition 1.3 Sei $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$ gegeben. Eine Darstellung

$$A = U\Sigma V^T \qquad \text{mit} \quad \Sigma = \begin{pmatrix} \hat{\Sigma} \\ 0 \end{pmatrix},$$

bei der $U \in \mathbb{R}^{m\times m}$ und $V \in \mathbb{R}^{n\times n}$ orthogonal sind und $\hat{\Sigma} = \text{diag}\,(\sigma_1, \ldots, \sigma_n)$ eine $n \times n$-Diagonalmatrix mit

$$\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0$$

ist, heißt eine *Singulärwertzerlegung* von A, die Zahlen $\sigma_1, \ldots, \sigma_n$ heißen die *singulären Werte* von A. Eine Darstellung $A = \hat{U}\hat{\Sigma}V^T$, bei der die Spalten von $\hat{U} \in \mathbb{R}^{m\times n}$ ein Orthonormalsystem bilden, also $\hat{U}^T\hat{U} = I$ gilt, und $\hat{\Sigma}$ und V wie eben sind, heißt eine *reduzierte Singulärwertzerlegung* von A.

Natürlich erhält man aus einer (vollen) Singulärwertzerlegung von A eine reduzierte, indem man $\hat{U}$ durch Weglassen der letzten $m-n$ Spalten aus U gewinnt. Umgekehrt erhält man aus einer reduzierten Singulärwertzerlegung $A = \hat{U}\hat{\Sigma}V^T$ eine volle Singulärwertzerlegung, indem man die Spalten von $\hat{U}$ durch weitere $m-n$ Vektoren zu einer Orthonormalbasis des $\mathbb{R}^m$ ergänzt.

Bevor wir auf die Existenz einer Singulärwertzerlegung eingehen, die Eindeutigkeit der singulären Werte beweisen und die Anwendung beim rangdefizienten Least Squares Problem erläutern, wollen wir uns über die Möglichkeiten von Maple bei der Berechnung einer Singulärwertzerlegung informieren.

Im package `LinearAlgebra` von Maple, das durch `with(LinearAlgebra):` (kein Output) geladen werden kann, gibt es die Funktion `SingularValues`. Am besten machen wir uns die Wirkungsweise dieser Funktion durch Beispiele klar. Wichtig ist hierbei, dass wenigstens einer der Einträge von A eine Gleitkommazahl sein muss, wenn man auch die orthogonalen Matrizen U oder V berechnen will. Daher hat im folgenden Beispiel der erste Eintrag von A noch einen Dezimalpunkt erhalten, andernfalls erscheint eine Fehlermeldung.

```
> with(LinearAlgebra):
> A:=Matrix([[22.,10,2],[14,7,10],[-1,13,-1],[-3,-2,13],[9,8,1]]):
> U,S,Vt:=SingularValues(A,output=['U','S','Vt']):
> Sigma:=DiagonalMatrix(S[1..3],5,3):
> Norm(U.Sigma.Vt-A);
```

$$.621724893790087663\,10^{-14}$$

```
> Norm(Transpose(U).U-IdentityMatrix(5));
```

$$.860856524953490521\,10^{-15}$$

```
> Norm(Vt.Transpose(Vt)-IdentityMatrix(3));
```

$$.714706072102444523\,10^{-15}$$

```
> S;
```

$$\begin{bmatrix} 32.2770990745160375 \\ 15.8010822982450314 \\ 11.8538885408107380 \\ 0. \\ 0. \end{bmatrix}$$

Wir haben hier überprüft, ob mit den angegebenen Werten die Darstellung richtig ist und ob U und V orthogonal sind. Ferner haben wir am Schluss die singulären Werte ausgegeben. Natürlich hätten wir den Output auch anders als mit U, S und Vt benennen können. Mit Hilfe der Funktion `DiagonalMatrix` wird aus einem Vektor eine Matrix konstruiert. Oben wäre es auch möglich gewesen, nur die singulären Werte oder nur die orthogonalen Matrizen U und V auszugeben. Die Funktion `Norm` ohne ein optionales Argument berechnet, angewandt auf eine Matrix, die Maximumnorm dieser Matrix.

Satz 1.4 *Sei $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$ gegeben. Dann gilt:*

1. *Es existiert eine Singulärwertzerlegung $A = U\Sigma V^T$ (mit den in der Definition angegebenen Eigenschaften).*

2. *Die singulären Werte sind die Quadratwurzeln der Eigenwerte von A^TA und daher eindeutig bestimmt.*

3. *Die Anzahl r positiver Singulärwerte ist der Rang von A.*

Beweis: Seien

$$\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_n = 0$$

die Eigenwerte von A^TA und $\{v_1, \ldots, v_n\}$ ein zugehöriges Orthonormalystem von Eigenvektoren. Man definiere $\sigma_i := \lambda_i^{1/2}$, $i = 1, \ldots, n$, und hiermit

$$\hat{\Sigma} := \text{diag}\,(\sigma_1, \ldots, \sigma_n), \qquad V := (\ v_1 \ \cdots \ v_n\ ).$$

Dann ist $V \in \mathbb{R}^{n\times n}$ natürlich eine orthogonale Matrix. Weiter definiere man

$$u_i := \frac{1}{\sigma_i} A v_i, \qquad i = 1, \ldots, r.$$

Dann ist $\{u_1, \ldots, u_r\}$ ein Orthonormalsystem von Eigenvektoren zu $AA^T \in \mathbb{R}^{m\times m}$ mit zugehörigen positiven Eigenwerten $\lambda_1, \ldots, \lambda_r$. Man ergänze $\{u_1, \ldots, u_r\}$ durch $u_{r+1}, \ldots, u_m$ zu einem vollständigen System von Eigenvektoren der Matrix $AA^T \in \mathbb{R}^{m\times m}$, wobei $u_{r+1}, \ldots, u_m$ notwendigerweise Eigenvektoren zum Eigenwert 0 sind. Setzt man nun

$$U := (\ u_1 \ \cdots \ u_m\ ),$$

so ist U orthogonal, ferner

$$(U^TAV)_{ij} = u_i^T A v_j = \sigma_j u_i^T u_j = \sigma_i \delta_{ij}, \qquad 1 \leq i, j \leq r.$$

Wegen

$$AA^T u_i = 0 \quad (i = r+1, \ldots, m), \qquad A^TAv_j = 0 \quad (j = r+1, \ldots, n)$$

sowie $\text{Kern}\,(A) = \text{Kern}\,(A^TA)$ und $\text{Kern}\,(A^T) = \text{Kern}\,(AA^T)$, ist

$$(U^TAV)_{ij} = 0 \qquad \text{falls } i \in \{r+1, \ldots, m\} \text{ oder } j \in \{r+1, \ldots, n\}.$$

Folglich ist durch

$$A = U \begin{pmatrix} \hat{\Sigma} \\ 0 \end{pmatrix} V^T$$

die gesuchte Singulärwertzerlegung gefunden.

Ist $A = \hat{U}\hat{\Sigma}V^T$ eine reduzierte Singulärwertzerlegung von A, so ist $A^TA = V\hat{\Sigma}^2V^T$, insbesondere haben A^TA und $\hat{\Sigma}^2$ die selben Eigenwerte. Die singulären Werte sind also die (nichtnegativen) Quadratwurzeln aus den Eigenwerten von A^TA.

Offenbar ist

$$\text{Rang}\,(A) = n - \dim \text{Kern}\,(A) = n - \dim \text{Kern}\,(A^TA) = \text{Rang}\,(A^TA) = r,$$

wobei[4] r die Anzahl der positiven Singulärwerte bezeichnet. □

[4]Hierbei haben wir benutzt, daß $\text{Kern}\,(A) = \text{Kern}\,(A^TA)$. Beweis?

Kennt man eine Singulärwertzerlegung von A, so kann die Lösungsmenge zum linearen Ausgleichsproblem mit den Daten (A, b) angegeben werden. Denn sei $A = U\Sigma V^T$ eine Singulärwertzerlegung von A, $r := \text{Rang}\,(A)$ und $U = (\ u_1\ \cdots\ u_m\ )$, $V = (\ v_1\ \cdots\ v_n\ )$. Für ein beliebiges $x \in \mathbb{R}^n$ ist dann

$$\begin{aligned}\|Ax - b\|_2^2 &= \|U^T(Ax - b)\|_2^2 \\ &= \|A^T AV(V^T x) - U^T b\|_2^2 \\ &= \|\Sigma(V^T x) - U^T b\|_2^2 \\ &= \sum_{i=1}^{r}[\sigma_i(V^T x)_i - u_i^T b]^2 + \sum_{i=r+1}^{m}(u_i^T b)^2.\end{aligned}$$

Hieraus folgt, dass $x \in \mathbb{R}^n$ genau dann eine Lösung des zu den Daten (A, b) gehörenden linearen Ausgleichsproblem ist, wenn $(V^T x)_i = u_i^T b/\sigma_i$, $i = 1, \ldots, r$. Hieraus erhalten wir sofort:

Satz 1.5 *Sei $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$, $r := \text{Rang}\,(A)$ und $A = U\Sigma V^T$ eine Singulärwertzerlegung von A. Mit u_i bzw. v_i seien die i-te Spalte von U bzw. V bezeichnet, weiter seien $\sigma_1 \geq \cdots \geq \sigma_n$ die singulären Werte von A. Dann gilt:*

1. *Für jedes $b \in \mathbb{R}^m$ ist die Menge $\mathcal{L}$ der Lösungen des zu (A, b) gehörenden linearen Ausgleichsproblems gegeben durch*

$$\mathcal{L} = \Big\{\sum_{i=1}^{r}\frac{u_i^T b}{\sigma_i}v_i + \sum_{i=r+1}^{n}\alpha_i v_i : \alpha_i \in \mathbb{R},\ i = r+1, \ldots, n\Big\}.$$

2. *Es ist*

$$x^* := \sum_{i=1}^{r}\frac{u_i^T b}{\sigma_i}v_i$$

die eindeutige Lösung minimaler euklidischer Norm des zu (A, b) gehörenden linearen Ausgleichsproblems.

Bemerkung: Mit Hilfe der Singulärwertzerlegung kann die *Pseudoinverse* $A^+ \in \mathbb{R}^{n\times m}$ einer Matrix $A \in \mathbb{R}^{m\times n}$ definiert werden, und zwar so, dass für eine quadratische, nichtsinguläre Matrix die Begriffe "Pseudoinverse" und "Inverse" von A zusammenfallen. Hierauf gehen wir in Aufgabe 5 ein. □

Die numerische Berechnung der Singulärwertzerlegung einer Matrix ist nicht ganz einfach. Auf diese gehen wir daher nicht mehr ein.

4.1.3 Lineare Tschebyscheff-Approximation

Wir betrachten in diesem kurzen Unterabschnitt lineare Tscgebyscheff-Approximation, wobei wir uns sogar auf die Approximation mit Polynomen spezialisieren[5]. Wir betrachten also ein Approximationsproblem mit den folgenden Daten:

[5]Ebenso hätten wir auch ohne Mehrarbeit sogenannte Haarsche Teilräume betrachten können, wir wollen aber nicht zu viele Vokabeln einführen.

1. Es ist $(X, \|\cdot\|) = (C[a,b], \|\cdot\|_\infty)$, wobei wir bei der Norm auf die Einführung einer Gewichtsfunktion verzichten, so dass

$$\|x\|_\infty := \max_{t\in[a,b]} |x(t)|.$$

2. Es ist $M = \mathcal{P}_n \subset C[a,b]$, der $(n+1)$-dimensionale Teilraum der Polynome vom Grad $\le n$, die Menge der Funktionen, mit denen approximiert wird.

3. Es ist $z \in C[a,b]$ die zu approximierende Funktion.

Da $\mathcal{P}_n$ ein endlich dimensionaler linearer Teilraum ist, besitzt das Approximationsproblem mindestens eine Lösung (siehe den ersten Teil von Satz 1.1). Mit $d(z,\mathcal{P}_n)$ bezeichnen wir den Abstand von z zu $\mathcal{P}_n$, d. h. es ist $d(z,\mathcal{P}_n) := \min_{x\in\mathcal{P}_n} \|x-z\|_\infty$. Der *Weierstraßsche Approximationssatz* sagt aus, dass $\lim_{n\to\infty} d(z,\mathcal{P}_n) = 0$, dass also jede auf dem Intervall $[a,b]$ stetige Funktion z beliebig genau bezüglich der Norm $\|\cdot\|_\infty$ durch ein Polynom approximiert werden kann.

Wir wollen die wichtigsten Aussagen zur Charakterisierung und der Eindeutigkeit einer Lösung bringen, wobei aber nicht alles bewiesen werden soll. Zunächst beweisen wir den *Satz von de la Vallée-Poussin*.

Satz 1.6 *Gegeben sei das obige lineare Approximationsproblem. Zu $x \in \mathcal{P}_n$ mögen $n+2$ Punkte $t_0,\dots,t_{n+1}$ mit $a \le t_0 < \cdots < t_{n+1} \le b$ und*

$$[x(t_i) - z(t_i)]\,[x(t_{i+1}) - z(t_{i+1})] < 0, \qquad i = 0,\dots,n,$$

existieren. D. h. das Vorzeichen des Defektes $x(\cdot) - z(\cdot)$ alterniere in den Punkten $t_0,\dots,t_{n+1}$. Dann ist

$$\min_{i=0,\dots,n+1} |x(t_i) - z(t_i)| \le d(z,\mathcal{P}_n) \le \|x-z\|_\infty.$$

Beweis: Zu zeigen ist hier natürlich nur die linke Ungleichung. Angenommen, es gibt ein $\hat{x} \in \mathcal{P}_n$ mit $\|\hat{x}-z\|_\infty < \min_{i=0,\dots,n+1} |x(t_i)-z(t_i)|$. Insbesondere ist dann

$$|\hat{x}(t_i) - z(t_i)| < |x(t_i) - z(t_i)|, \qquad i = 0,\dots,n+1.$$

Wir werden uns überlegen, dass $\hat{x} - x \in \mathcal{P}_n \setminus \{0\}$ in den t_i alternierendes Vorzeichen, also mindestens $n+1$ Nullstellen in $[a,b]$ besitzt, was der gewünschte Widerspruch ist. Wir setzen zur Abkürzung $\sigma_i := \operatorname{sign}(x(t_i) - z(t_i))$. Dann ist

$$\begin{aligned}
[\hat{x}(t_i) - x(t_i)][x(t_i) - z(t_i)] &= [(\hat{x}(t_i) - z(t_i)) - (x(t_i) - z(t_i))][x(t_i) - z(t_i)] \\
&= \underbrace{[\sigma_i(\hat{x}(t_i) - z(t_i)) - |x(t_i) - z(t_i)|]}_{<0}\,|x(t_i) - z(t_i)| \\
&< 0.
\end{aligned}$$

Da $x - z$ nach Voraussetzung in den t_i dem Vorzeichen nach alterniert, trifft dies auch auf $\hat{x} - x$ zu, womit der Satz schließlich bewiesen ist. □

Bemerkung: Satz 1.6 gibt einige nützliche Informationen. Zum einen kann mit ihm eine untere (und eine triviale obere) Schranke für den Minimalwert $d(z, \mathcal{P}_n)$ gefunden werden. Zum anderen liefert er sofort eine *hinreichende* Bedingung dafür, dass ein $x^* \in M$ beste Approximierende an z in $\mathcal{P}_n$ ist. Alterniert nämlich $x^* - z$ in $n+2$ aufeinander folgenden Punkten $t_0 < \cdots < t_{n+1}$ aus $[a, b]$ dem Vorzeichen nach und ist $|x^*(t_i) - z(t_i)| = \|x^* - z\|_\infty$, $i = 0, \ldots, n+1$, so ist $d(z, \mathcal{P}_n) = \|x^* - z\|_\infty$ und daher x^* beste Approximierende an z in $\mathcal{P}_n$. □

Beispiel: Sei speziell $[a, b] := [\frac{1}{2}, 1]$, $n := 1$ und $z(t) := \sqrt{t}$. Oben hatten wir schon motiviert und mit der `minimax`-Funktion von Maple nachgeprüft, dass

$$x^*(t) := (2 - \sqrt{2})t + \frac{3}{8}\left(\frac{3}{2}\sqrt{2} - 1\right)$$

eine Lösung der zugehörigen linearen Tschebyscheffschen Approximationsaufgabe ist. Um dies mit Hilfe des Satzes von de la Vallée-Poussin bzw. der anschließenden Bemerkung nachzuprüfen, setzen wir $t_0 := \frac{1}{2}$, $t_2 := 1$ und bestimmen $t_1 \in (t_0, t_1)$ so, dass $x^* - z$ in t_1 extremal wird. Man berechnet

$$t_1 := \frac{3 + 2\sqrt{2}}{8}, \qquad \|x^* - z\|_\infty = -[x^*(t_1) - z(t_1)] = \frac{10 - 7\sqrt{2}}{16}$$

und

$$[x^*(t_0) - z(t_0)] = -[x^*(t_1) - z(t_1)] = [x^*(t_2) - z(t_2)].$$

Folglich ist x^* beste Approximierende an z in $\mathcal{P}_1$. □

Im Anschluss an den Satz von de la Vallée Possin hatten wir in einer Bemerkung schon eine hinreichende Bedingung dafür angegeben, dass ein $x^* \in \mathcal{P}_n$ beste Tschebyscheff-Approximierende an z bezüglich $\mathcal{P}_n$ ist. Der folgende Satz heißt *Alternantensatz* und sagt aus, dass diese hinreichende Optimalitätsbedingung auch notwendig ist. Diese wesentlich schwierigere Richtung wollen wir aber nicht beweisen[6].

Satz 1.7 *Es ist $x^* \in \mathcal{P}_n$ genau dann beste Tschebyscheff-Approximierende an $z \in C[a, b]$ bezüglich $\mathcal{P}_n$, wenn es $n+2$ Punkte $t_0, \ldots, t_{n+1}$ in $[a, b]$ mit*

(a) $a \le t_0 < \cdots < t_{n+1} \le b$,

(b) $|x^*(t_i) - z(t_i)| = \|x^* - z\|_\infty$, $i = 0, \ldots, n+1$,

(c) $[x^*(t_i) - z(t_i)] = -[x^*(t_{i+1} - z(t_{i+1})]$, $i = 0, \ldots, n$

gibt.

Als Folgerung aus dem Alternantensatz erhält man die folgende Eindeutigkeitsaussage.

Satz 1.8 *Die beste Tschebyscheff-Approximierende an $z \in C[a, b]$ bezüglich $\mathcal{P}_n$ ist eindeutig bestimmt.*

[6] Einen "konstruktiven" Beweis findet man bei R. Schaback, H. Werner (1992) *Numerische Mathematik*. Springer, Berlin-Heidelberg-New York.

Beweis: Seien $x_1, x_2 \in \mathcal{P}_n$ jeweils beste Approximierende an $z \in C[a,b]$ bezüglich $\mathcal{P}_n$. Dann ist auch $\frac{1}{2}(x_1 + x_2)$ beste Tschebyscheff-Approximierende. Nach dem Alternantensatz existieren $n+2$ Punkte t_i, $i = 0, \ldots, n+1$, mit $a \le t_0 < \cdots < t_{n+1} \le b$ und $\sigma \in \{-1, 1\}$ mit

$$\frac{1}{2}[x_1(t_i) + x_2(t_i)] - z(t_i) = \sigma(-1)^i d(z, \mathcal{P}_n), \qquad i = 0, \ldots, n+1.$$

Also ist

$$\frac{1}{2}[x_1(t_i) - z(t_i)] + \frac{1}{2}[x_2(t_i) - z(t_i)] = \sigma(-1)^i d(z, \mathcal{P}_n), \qquad i = 0, \ldots, n+1.$$

Nun ist

$$|x_1(t_i) - z(t_i)| \le \|x_1 - z\|_\infty = d(z, \mathcal{P}_n), \quad i = 0, \ldots, n+1,$$

und entsprechend $|x_2(t_i) - z(t_i)| \le d(z, \mathcal{P}_n)$, $i = 0, \ldots, n+1$. Folglich ist

$$\begin{aligned} d(z, \mathcal{P}_n) &= \left|\frac{1}{2}[x_1(t_i) - z(t_i)] + \frac{1}{2}[x_2(t_i) - z(t_i)]\right| \\ &\le \frac{1}{2}|x_1(t_i) - z(t_i)| + \frac{1}{2}|x_2(t_i) - z(t_i)| \\ &\le d(z, \mathcal{P}_n). \end{aligned}$$

Insgesamt folgt $x_1(t_i) - z(t_i) = x_2(t_i) - z(t_i)$, $i = 0, \ldots, n+1$. Daher hat $x_1 - x_2 \in \mathcal{P}_n$ $n+2$ Nullstellen, so dass $x_1 = x_2$. Die Eindeutigkeit ist bewiesen. □

Der Alternantensatz ist Grundlage für das wichtigste Verfahren zur numerischen Berechnung der besten Tschebyscheff-Approximierenden, des Remes-Verfahrens. Auch hierauf kann nicht mehr eingegangen werden.

4.1.4 Aufgaben

1. Gegeben seien $t, b \in \mathbb{R}^m$ (m Beobachtungen b_i zu Zeiten t_i, $i = 1, \ldots, m$). Zur Bestimmung der *Regressionsgeraden* hat man die Aufgabe

   (P) Minimiere $f(x_1, x_2) := \frac{1}{2}\sum_{i=1}^{m}(x_1 + x_2 t_i - b_i)^2, \quad (x_1, x_2) \in \mathbb{R} \times \mathbb{R},$

   zu lösen. Man gebe eine explizite Darstellung für die Lösung.

2. In der folgenden Tabelle gibt t die Länge eines Säuglings bei der Geburt und b die Schwangerschaftsdauer an:

t [cm]	48	49	50	51	52
b [Tage]	277.1	279.3	281.4	283.2	284.8

   Hierbei kann man sich vorstellen, dass die Daten in den fünf Gruppen schon Mittelwerte aus zahlreichen weiteren Messungen sind. Es wird ein linearer Zusammenhang zwischen der Länge bei der Geburt und der Schwangerschaftsdauer vermutet. Man bestimme mit der Methode der kleinsten Quadrate die beiden Parameter bzw. löse das entsprechende lineare Ausgleichsproblem. Hierbei kann Maple (oder ein anderes mathematisches Anwendersystem) oder Aufgabe 1 benutzt werden.

3. Gegeben sei ein Approximationsproblem mit den Daten $(X, \|\cdot\|)$, $M \subset X$ und $z \in X$. Man zeige:

   (a) Ist $M \subset X$ konvex, so ist die Menge der besten Approximierenden an z bezüglich M ebenfalls eine konvexe Menge.

   (b) Ist $M \subset X$ konvex und die Norm $\|\cdot\|$ *strikt*, d. h. gilt die Implikation

   $$\|x + y\| = \|x\| + \|y\| \Longrightarrow \text{ } x \text{ und } y \text{ sind linear abhängig,}$$

   so existiert höchstens eine beste Approximierende an z bezüglich M.

   (c) Ist $\|\cdot\|$ durch ein inneres Produkt $(\cdot, \cdot)$ erzeugt, so ist die Norm $\|\cdot\|$ strikt. Ferner gilt die sogenannte *Parallelogrammgleichung*, d. h. für alle $x, y \in X$ ist

   $$\|x + y\|^2 + \|x - y\|^2 = 2(\|x\|^2 + \|y\|^2).$$

4. Man bestimme alle Lösungen des linearen Ausgleichsproblems zu den Daten

   $$A := \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}, \qquad b := \begin{pmatrix} 2 \\ 0 \\ 0 \\ -1 \end{pmatrix}.$$

   Hinweis: Man wende den ersten Teil von Satz 1.5 an.

5. Sei $A \in \mathbb{R}^{m \times n}$ mit $m \geq n$ gegeben, es sei $r := \text{Rang } (A)$. Sei $A = U\Sigma V^T$ eine Singulärwertzerlegung von A (also $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ orthogonal, $\Sigma \in \mathbb{R}^{m \times n}$ eine Diagonalmatrix mit den singulären Werten $\sigma_1 \geq \cdots \geq \sigma_n$ in der Diagonalen). Man definiere $A^+ := V\Sigma^+ U^T$, wobei $\Sigma_+ \in \mathbb{R}^{n \times m}$ gegeben ist durch $\Sigma^+ := (\ \hat{\Sigma}^+ \quad 0\ )$ mit

   $$\hat{\Sigma}^+ := \text{diag } (1/\sigma_1, \ldots, 1/\sigma_r, \underbrace{0, \ldots, 0}_{n-r}).$$

   Man zeige:

   (a) Für jedes $b \in \mathbb{R}^m$ ist $A^+ b$ die eindeutige Lösung minimaler euklidischer Norm des zu den Daten (A, b) gehörenden linearen Ausgleichsproblems. Insbesondere zeigt dies die Wohldefiniertheit der Pseudoinversen.

   (b) Ist $m \geq n$ und Rang $(A) = n$, so ist $A^+ = (A^T A)^{-1} A^T$.

   (c) Ist $A \in \mathbb{R}^{n \times n}$ nichtsingulär, so ist $A^+ = A$.

6. Sei $z \in C^2[a, b]$ eine Funktion, deren zweite Ableitung auf dem Intervall $[a, b]$ nichtnegativ oder nichtpositiv ist, die also konvex oder konkav ist. Wegen des Mittelwertsatzes existiert ein $t_1 \in (a, b)$ mit $z(b) - z(a) = z'(t_1)(b - a)$. Man zeige, dass

   $$x^*(t) := \frac{z(b) - z(a)}{b - a}\left(t - \frac{a + t_1}{2}\right) + \frac{1}{2}[z(a) + z(t_1)]$$

   die beste Tschebyscheff-Approximierende an z bezüglich $\mathcal{P}_1$ ist.

7. Bei gegebener nichtnegativer ganzer Zahl n heißt die durch $T_n(t) := \cos(n \arccos t)$ definierte Funktion $T_n\colon [-1,1] \longrightarrow \mathbb{R}$ *Tschebyscheff-Polynom (erster Art) vom Grad* n. Man beweise die folgenden Aussagen, von denen durch die erste überhaupt erst nachgewiesen wird, dass es sich um Polynome handelt.

   (a) Es gilt die Rekursionsformel

   $$T_0(t) = 1, \quad T_1(t) = 1, \quad T_{n+1}(t) = 2tT_n(t) - T_{n-1}(t) \quad (n = 2, 3, \ldots).$$

   (b) Es ist $T_n \in \mathcal{P}_n$ und $T_n(t) = 2^{n-1}t^n + p(t)$ mit $p \in \mathcal{P}_{n-2}$.

   (c) Die Nullstellen von T_n sind

   $$s_j := \cos\left(\frac{2(n-j)+1}{2n}\,\pi\right), \qquad j = 1, \ldots, n.$$

   (d) Es ist $\|T_n\|_\infty = 1$ und $T_n(t_i) = (-1)^{n-i}$ mit

   $$t_i := \cos\left(\frac{n-i}{n}\,\pi\right), \qquad i = 0, \ldots, n.$$

   (e) Es ist $T_n(-t) = (-1)^n T_n(t)$.

   (f) Es gilt

   $$\frac{2}{\pi}\int_{-1}^{1} \frac{T_m(t)T_n(t)}{\sqrt{1-t^2}}\,dt = \begin{cases} 2 & \text{für} \quad m = n = 0, \\ 1 & \text{für} \quad m = n \neq 0, \\ 0 & \text{für} \quad m \neq n. \end{cases}$$

   Hinweis: Mache die Variablentransformation $t = \cos\phi$.

   (g) Es ist

   $$T_n(t) = \frac{1}{2}\left[(t + \sqrt{t^2-1})^n + (t - \sqrt{t^2-1})^n\right].$$

   (h) Sei $n \in \mathbb{N}$ und $p^* \in \mathcal{P}_{n-1}$ die beste Tschebyscheff-Approximierende an $z(t) := t^n$ auf dem Intervall $[a,b] := [-1,1]$. Dann ist $t^n - p^*(t) = 2^{-n+1}T_n(t)$ und daher $d(z, \mathcal{P}_{n-1}) = 2^{-n+1}$ der Abstand von z zu $\mathcal{P}_{n-1}$.

8. Sei $(X, (\cdot,\cdot))$ ein Prä-Hilbertraum, also $(\cdot,\cdot)$ ein inneres Produkt und $\|\cdot\|$ die durch $\|x\| := (x,x)^{1/2}$ induzierte Norm. Sei ferner $M \subset X$ nichtleer, konvex und $z \in X$. Dann gilt:

   (a) Es ist $x^* \in M$ genau dann beste Approximierende an z bezüglich M, wenn $(x - x^*, z - x^*) \leq 0$ für alle $x \in M$.

   (b) Ist $(X, (\cdot,\cdot))$ sogar ein *Hilbertraum*, ist also jede Cauchy-Folge in $(X, \|\cdot\|)$ konvergent, und ist M zusätlich abgeschlossen, so existiert (genau eine: das folgt schon aus den Aussagen von Aufgabe 3) eine eindeutige beste Approximierende an z bezüglich M.

4.2 Optimierungsaufgaben

Wie ganz am Anfang dieses Kapitels schon angedeutet wurde, besteht eine Optimierungsaufgabe darin, eine auf einer gewissen Menge M, der Menge der *zulässigen Lösungen*, gegebene reellwertige Funktion f, die sogenannte Zielfunktion, zu minimieren. Im nächsten Unterabschnitt wollen wir durch einige Beispiele einen Eindruck über die Vielfalt der hierdurch erfassten Probleme geben. Danach gehen wir auf lineare Optimierungsaufgaben ein, wobei wir auf die Beschreibung des Simplexverfahrens aus Zeitgründen verzichten. Bei nichtlineare Optimierungsaufgaben konzentrieren wir uns auf die Darstellung und Anwendung des Satzes von Kuhn-Tucker.

4.2.1 Beispiele

Beispiel: Die Bevölkerungszahlen von Lausanne in den Jahren 1950–1959 sind in der folgenden Tabelle angegeben:

Jahr t_i	Bevölkerungszahl p_i	Malthus-Modell
1950	107 680	107 629
1951	108 997	109 230
1952	111 106	110 855
1953	112 849	112 504
1954	114 338	114 178
1955	115 476	115 876
1956	117 323	117 600
1957	118 968	119 349
1958	121 210	121 125
1959	123 328	122 926

Mit dem Malthus-Wachstumsmodell (wir kommen hierauf im Zusammenhang mit gewöhnlichen Differentialgleichungen zurück) machen wir für die Population zur Zeit t einen Ansatz

$$p(a, q_0; t) = q_0 \exp(a(t - 1950)).$$

Die noch unbekannten Parameter a, q_0 werden als Lösungen des *nichtlinearen Ausgleichsproblems*

$$\text{Minimiere} \quad f(a, q_0) := \frac{1}{2} \sum_{i=1}^{10} [p(a, q_0; t_i) - p_i]^2, \quad (a, q_0) \in \mathbb{R}^2,$$

gewonnen. Man hat hier also eine *unrestringierte* Optimierungsaufgabe, nämlich auf $M := \mathbb{R}^2$ die oben definierte Funktion f zu minimieren. Zu ihrer Lösung gibt es spezielle Algorithmen. Es würde zu weit führen, hierauf näher einzugehen[7]. Die Möglichkeiten von Maple scheinen ziemlich beschränkt zu sein. Besser sieht es bei MATLAB aber auch vor allem bei Mathematica (siehe z. B. der Befehl `NonlinearFit`, der nach Laden

[7] Siehe z. B. Kapitel 7 von J. Werner *Numerische Mathematik 2*. Vieweg, Braunschweig-Wiesbaden.

des entsprechenden Zusatzpakets durch `«Statistics‘NonlinearFit‘` zur Verfügung steht, es kann aber auch der Befehl `FindMinimum` nützlich sein) aus. Man erhält als näherungsweise Lösung $a = 0.0147662$, $q_0 = 107629$. Die hiermit erhaltenen Werte sind in der dritten Spalte in obiger Tabelle eingetragen. □

Auch die folgende Aufgabe ist eine *unrestringierte Optimierungsaufgabe*.

Beispiel: Das folgende Problem scheint 1629 zum ersten Mal von Fermat formuliert worden zu sein:

- Gegeben seien drei Punkte in der Ebene. Man finde einen Punkt in der Ebene derart, dass die Summe der Abstände dieses Punktes zu den drei vorgegebenen Punkten minimal ist.

Die Verallgemeinerung auf m Punkte im $\mathbb{R}^n$ heißt das Fermat-Weber-Problem:

- Gegeben seien $m \geq 3$ paarweise verschiedene Punkte $a_1, \ldots, a_m \in \mathbb{R}^n$ und positive reelle Zahlen $w_1, \ldots, w_m$. Man bestimme eine Lösung $x^* \in \mathbb{R}^n$ von

$$\text{(P)} \qquad \text{Minimiere} \quad f(x) := \sum_{i=1}^{m} w_i \, \|x - a_i\|_2 \quad \text{auf} \quad M := \mathbb{R}^n,$$

  wobei $\| \cdot \|_2$ die euklidische Norm auf dem $\mathbb{R}^n$ bedeutet.

Die ökonomische Interpretation (man spricht in den Wirtschaftswissenschaften auch von dem “Standortproblem”) könnte die folgende sein: Eine Warenhauskette mit Filialen in $a_1, \ldots, a_k$ und Zulieferern in $a_{k+1}, \ldots, a_m$ will den Standort eines zusätzlichen Lagers bestimmen. Dieser soll so gewählt werden, dass eine gewichtete Summe der Abstände vom Lager zu den Filialen und von den Zulieferern zum Lager minimal wird.

Beim Fermat-Weber-Problem ist der Abstand zwischen zwei Punkten durch den euklidischen Abstand gegeben. Es liegt nicht nur an der bekannten Verallgemeinerungswut der Mathematiker, dass auch andere Abstandsbegriffe bzw. Normen in der Literatur betrachtet wurden. Hierzu gehören insbesondere die 1-Norm, die ∞-Norm (Maximumnorm) und positive Linearkombinationen dieser beiden Normen als Spezialfälle sogenannter polyedrischer Normen (hier ist die Einheitskugel ein Polyeder).

Wir wollen hier auf das Fermat-Weber-Problem gar nicht weiter eingehen, sondern nur einen hübschen geometrischen Beweis dafür angeben, dass beim eingangs genannten Fermat-Problem der gesuchte Punkt (auch Fermat- oder Torricelli-Punkt genannt) derjenige ist, von dem die drei Seiten des (spitzwinkligen) Dreiecks unter einem Winkel von 120° gesehen werden.

Gegeben sei ein spitzwinkliges Dreieck in der Ebene mit den Ecken A, B und C. In diesem Dreieck wähle man sich einen beliebigen Punkt P und verbinde ihn mit den Ecken. Das innere Dreieck $\triangle APB$ drehe man um 60° um B und erhalte das Dreieck $\triangle C'P'B$. In Abbildung 4.3 ist die Konstruktion angegeben. Dann sind $\triangle ABC'$ und $\triangle PBP'$ gleichseitig, die Winkel in diesen Dreiecken also jeweils 60°. Daher ist

$$|AP| + |BP| + |CP| = |C'P'| + |P'P| + |PC|,$$

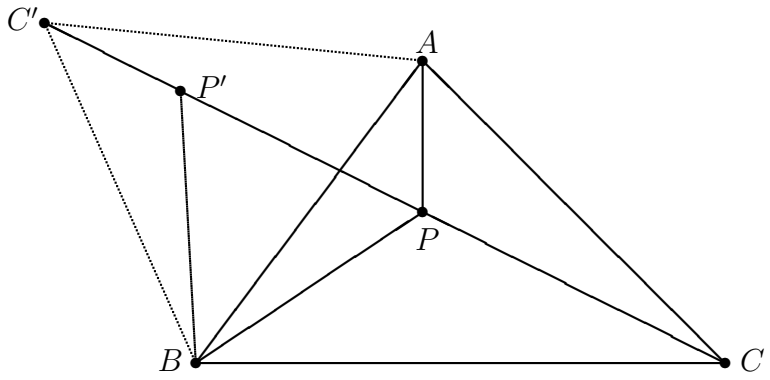

Abbildung 4.3: Konstruktion zum Fermat-Problem

und die rechtsstehende Summe ist die Länge eines i. Allg. gebrochenen Streckenzuges. Dieser ist minimal, wenn er ein Geradensegment ist. In diesem Falle ist

$$\sphericalangle BPC = 180^\circ - \sphericalangle BPP' = 120^\circ$$

und

$$\sphericalangle APB = \sphericalangle C'P'B = 180^\circ - \sphericalangle PP'B = 120^\circ.$$

Der gesuchte Punkt P, für den $|AP|+|BP|+|CP|$ minimal ist, ist also derjenige Punkt P, für den

$$\sphericalangle APB = \sphericalangle BPC = \sphericalangle CPA = 120^\circ.$$

Diese Lösung des Fermat-Problems kann man bei H. S. M. Coxeter (1969, S. 21)[8] nachlesen.

Zur Lösung unrestringierter Optimierungsaufgaben stellt *Mathematica* den Befehl `FindMinimum` bereit. In Abbildung 4.4 geben wir ein Dreieck und den zugehörigen

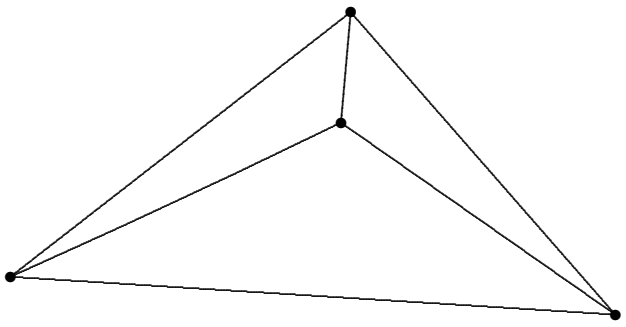

Abbildung 4.4: Der Fermat (Torricelli) Punkt

Fermat- bzw. Torricelli-Punkt an. Diesen haben wir im konkreten Fall mit Hilfe von

[8] H. S. M. Coxeter (1969) Introduction to Geometry. John Wiley & Sons, New York.

Mathematica (weil es in Maple offenbar, ich mag mich täuschen, keine der Mathematica-Funktion `FindMinimum` vergleichbare Funktion gibt) durch

```
xkoord={50,95,130};ykoord={5,40,0};
f[x_,y_]:=Sum[Sqrt[(xkoord[[i]]-x)^2+(ykoord[[i]]-y)^2],{i,3}];
FindMinimum[f[x,y],{x,90},{y,20}]
```

gefunden, als Output erhalten wir nämlich zum einen den minimalen Zielfunktionswert (hier: 107.189) und die Koordinaten des Fermat-Punktes (hier: $(93.7239, 25.3947)$). Hierbei gingen wir von drei Punkten mit den Koordinaten $(50, 5)$, $(95, 40)$ und $(130, 0)$ aus. □

Unter einer linearen Optimierungsaufgabe versteht man die Aufgabe, eine reellwertige lineare Funktion unter (affin) linearen Gleichungen und/oder Ungleichungen zu minimieren. Hier ist also die Menge M der zulässigen Lösungen ein *Polyeder* im $\mathbb{R}^n$ und die Zielfunktion f linear, so dass sie sich mit einem Vektor $c \in \mathbb{R}^n$, dem sogenannten Kostenvektor, in der Form $f(x) = c^T x$ darstellen lässt. Viele Probleme in den Anwendungen führen auf lineare Optimierungsaufgaben, in der Einführung gingen wir z. B. schon auf das klassische Diätproblem ein. Wir wollen hier als ein spezielles lineares Optimierungsproblem das *Transportproblem* kennenlernen. Dieses gehört zu den ersten näher untersuchten Optimierungsaufgaben. Verbunden hiermit werden Namen wie L. V. Kantorowich, T. C. Koopmans (beide erhielten 1975 den Wirtschafts-Nobelpreis) und F. L. Hitchcock. Zunächst geben wir ein spezielles Beispiel für das Transportproblem an, danach formulieren wir das (allgemeine) klassische Transportproblem.

Beispiel: In[9] zwei Rangierbahnhöfen A und B stehen 18 bzw. 12 leere Güterwagen. In den drei Bahnhöfen R, S und T werden 11, 10 bzw. 9 Güterwagen zum Verladen von Waren benötigt. Die Distanzen in km von den Rangierbahnhöfen zu den Bahnhöfen sind durch

	R	S	T
A	5	4	9
B	7	8	10

gegeben. Die Güterwagen sind so zu leiten, dass die totale Anzahl der durchfahrenen Leerkilometer minimal ist. Um dieses Problem zu lösen, führen wir als Variable $x_{AR}, x_{AS}, x_{A,T}, x_{BR}, x_{BS}, x_{BT}$ ein. Hierbei bedeutet x_{AR} z. B. die Anzahl der Güterwagen, die vom Rangierbahnhof A zum Bahnhof R gebracht werden. Die Gesamtzahl der gefahrenen Leerkilometer ist dann

$$z := 5x_{AR} + 4x_{AS} + 9x_{AT} + 7x_{BR} + 8x_{BS} + 10x_{BT},$$

diese gilt es zu minimieren. Als Nebenbedingungen hat man (wir beachten, dass die Gesamtzahl der in den Rangierbahnhöfen A und B zur Verfügung stehenden Güterwagen gleich der Gesamtzahl der in den Bahnhöfen R, S und T benötigten Güterwagen ist)

$$x_{AR} + x_{AS} + x_{AT} = 18, \qquad x_{BR} + x_{BS} + x_{BT} = 12$$

[9]Diese Aufgabe ist ein Beispiel in dem Lehrbuch über Numerische Mathematik von H. R. Schwarz.

(d. h. 18 Güterwagen gehen von A aus auf die Reise, 12 von B) und

$$x_{AR} + x_{BR} = 11, \qquad x_{AS} + x_{BS} = 10, \qquad x_{AT} + x_{BT} = 9$$

(der Bedarf in den Bahnhöfen R, S und T wird befriedigt), hinzu tritt außerdem die Bedingung, dass $x_{AS}, \ldots, x_{BT}$ nichtnegativ und ganzzahlig sind. Letzere Bedingung lassen wir weg (man kann zeigen: Sind die Daten ganzzahlig, so existiert auch eine ganzzahlige Lösung) und erhalten die lineare Optimierungsaufgabe

$$\text{Minimiere} \quad \begin{pmatrix} 5 \\ 4 \\ 9 \\ 7 \\ 8 \\ 10 \end{pmatrix}^T \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} \quad \text{unter den Nebenbedingungen}$$

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} = \begin{pmatrix} 18 \\ 12 \\ 11 \\ 10 \\ 9 \end{pmatrix}, \quad \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Wir wollen diese Aufgabe mit Maple lösen. Das kann z. B. folgendermaßen geschehen.

```
> with(simplex):
```

```
Warning, the protected names maximize and minimize have been redefined
and unprotected
```

```
> Rangier:=[18,12]:
> Bahnhof:=[11,10,9]:
> Distanzen:=array([[5,4,9],[7,8,10]]):
> x:=array(1..2,1..3):
> restr:={seq(sum(x[i,'j'],'j'=1..3)=Rangier[i],i=1..2)}union
> {seq(sum(x['i',j],'i'=1..2)=Bahnhof[j],j=1..3)};
```

$$restr := \{x_{1,1} + x_{1,2} + x_{1,3} = 18,\ x_{2,1} + x_{2,2} + x_{2,3} = 12,\ x_{1,1} + x_{2,1} = 11,\ x_{1,2} + x_{2,2} = 10,\ x_{1,3} + x_{2,3} = 9\}$$

```
> ziel:=add(add(x[i,j]*Distanzen[i,j],i=1..2),j=1..3);
```

$$ziel := 5\,x_{1,1} + 7\,x_{2,1} + 4\,x_{1,2} + 8\,x_{2,2} + 9\,x_{1,3} + 10\,x_{2,3}$$

```
> minimize(ziel,restr,NONNEGATIVE);
```

$$\{x_{1,3} = 0,\ x_{2,2} = 0,\ x_{1,2} = 10,\ x_{2,1} = 3,\ x_{1,1} = 8,\ x_{2,3} = 9\}$$

```
> assign(%);eval(x);
```

$$\begin{bmatrix} 8 & 10 & 0 \\ 3 & 0 & 9 \end{bmatrix}$$

```
> ziel;
```

Hieraus liest man ab: Die Gesamtzahl der Leerkilometer wird minimiert, wenn vom Rangierbahnhof A zu den Bahnhöfen R und S gerade 8 bzw. 10 Güterwagen gebracht werden und vom Rangierbahhof B nach R und T genau 3 bzw. 9 Güterwagen geletet werden. Die minimale Anzahl der Leerkilometer ist 191. □

Nun wollen wir das letzte Beispiel verallgemeinern und kommen dadurch zum klassischen Transportproblem.

Beispiel: Ein Gut, das in m Lagern vorhanden ist, soll zu n Kunden transportiert werden. Es ist zu entscheiden, welche Menge x_{ij} dieses Gutes vom Lager i zum Kunden j zu transportieren ist. Hierbei sei folgendes zu beachten:

- Die Transportkosten einer Mengeneinheit des Gutes vom Lager i zum Kunden j seien c_{ij} Geldeinheiten. Ferner wird angenommen, dass die Transportkosten vom Lager i zum Kunden j proportional zur Menge ist. Es wird also kein Mengenrabatt gewährt, ferner werden auch keine Fixkosten in Rechnung gestellt.
- Die Summe der gesamten Transportkosten $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$ ist zu minimieren.
- Die in Lager i vorhandene Menge $l_i \geq 0$ des Gutes sowie der Bedarf $k_j \geq 0$ des Kunden sind bekannt.
- Der Bedarf jedes Kunden muss befriedigt werden.
- Negative Transportmengen (Rücktransporte) sind ausgeschlossen.

Ein Transportplan $x = (x_{ij})$ ist *zulässig*, wenn

$$\sum_{j=1}^{n} x_{ij} \leq l_i, \qquad i = 1, \ldots, m,$$

die vom Lager i abtransportierte Menge also nicht größer ist als der Bestand l_i, $i = 1, \ldots, m$, und außerdem

$$\sum_{i=1}^{m} x_{ij} \geq k_j, \qquad j = 1, \ldots, n,$$

der Bedarf aller n Kunden also gedeckt wird. Hinzu kommt die Nichtnegativitätsforderung

$$x_{ij} \geq 0 \qquad (i = 1, \ldots, m,\ j = 1, \ldots, n).$$

Offenbar existiert genau dann ein zulässiger Transportplan, wenn

$$\sum_{i=1}^{m} l_i \geq \sum_{j=1}^{n} k_j,$$

der Gesamtbestand also nicht kleiner als der Gesamtbedarf ist. Ist dies erfüllt, so kann o. B. d. A. sogar angenommen werden, dass der Gesamtbestand gleich dem Gesamtbedarf ist. Andernfalls denke man sich einen "fiktiven Kunden" eingeführt, der den

Überschuss ohne Transportkosten aufnimmt, was natürlich in der Praxis bedeutet, dass dieser in den jeweiligen Lagern liegen bleibt. In diesem Falle lautet also das Transportproblem

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

unter den Nebenbedingungen

$$\sum_{j=1}^{n} x_{ij} = l_i \qquad (i=1,\dots,m), \qquad \sum_{i=1}^{m} x_{ij} = k_j \qquad (j=1,\dots,n)$$

sowie

$$x_{ij} \geq 0 \qquad (i=1,\dots,m,\ j=1,\dots,n).$$

Eine Matrix-Vektor-Schreibweise ist leicht möglich, indem man die "Matrix" $x=(x_{ij})$ zeilenweise liest:

$$x = \begin{pmatrix} x^1 \\ \vdots \\ x^m \end{pmatrix} \in \mathbb{R}^{mn} \qquad \text{mit} \quad x^i = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{in} \end{pmatrix}.$$

Bezeichnet ferner I die $n\times n$-Einheitsmatrix und e den Vektor des $\mathbb{R}^n$, dessen Komponenten alle gleich 1 sind, so können die Nebenbedingungen kompakt geschrieben werden als

$$(*) \qquad \begin{pmatrix} e^T & 0^T & \cdots & 0^T \\ 0^T & e^T & & 0^T \\ \vdots & \vdots & \ddots & \vdots \\ 0^T & 0^T & \cdots & e^T \\ I & I & \cdots & I \end{pmatrix} x = \begin{pmatrix} l \\ k \end{pmatrix}, \qquad x \geq 0.$$

Die Koeffizientenmatrix ist eine $(m+n)\times(mn)$-Matrix. Die Summe der ersten m Zeilen ist der Zeilenvektor $(e^T,\dots,e^T)$, was auch die Summe der letzten n Zeilen ist. Daher ist ihr Rang $\leq m+n-1$ und es ist nicht schwierig zu zeigen, dass hier sogar Gleichheit gilt. □

Beispiel: Es sollen 400 m^3 Kies von einem Ort zu einem anderen transportiert werden. Dies geschehe in einer (nach oben!) offenen Box der Länge x, der Breite y und der Höhe z. Der Boden (xy m^2) und die beiden Seiten ($2xz$ m^2) müssen aus einem Material hergestellt sein, das zwar nichts kostet, von dem aber nur 4 m^2 zur Verfügung steht. Das Material für die beiden Enden ($2yz$ m^2) kostet 200 Euro pro m^2. Ein Transport der Box kostet 1 Euro. Wie hat man die Box zu konstruieren?

Die Kosten zum Bau der Box sind $400yz$ Euro. Die Anzahl der Transporte ist $400/(xyz)$, so dass die Gesamtkosten zum Bau der Box und des Transportes der Kiesmenge durch

$$f(x,y,z) := 400\Big(yz + \frac{1}{xyz}\Big)$$

gegeben ist. Wegen der Kapazitätsschranken für das Material des Bodens und der beiden Seiten hat man die Restriktion

$$xy + 2xz \leq 4.$$

Berücksichtigt man noch, dass die Variablen positiv sein sollten, so haben wir insgesamt (wir lassen jetzt den Faktor 400 in der Zielfunktion fort) die Optimierungsaufgabe

$$\begin{cases} \text{Minimiere} \quad f(x,y,z) := 1/(xyz) + yz \quad \text{unter den Nebenbedingungen} \\ \qquad\qquad xy + 2xz \leq 4, \qquad x,y,z > 0. \end{cases}$$

Dies ist eine nichtlineare Optimierungsaufgabe. Die Möglichkeiten von *Mathematica* und Maple bei nichtlinearen Optimierungsaufgaben sind eher dürftig. □

Beispiel: Ein von J. J. Sylvester (1857) gestelltes Problem lautet:

- It is required to find the least circle which shall contain a given system of points in a plane.

Nur leicht verallgemeinert bedeutet dies: Gegeben seien l Punkte $a_1, \ldots, a_l \in \mathbb{R}^n$, gesucht ist die euklidische Kugel $B[x;r] := \{y \in \mathbb{R}^n : \|y - x\|_2 \leq r\}$ mit minimalem Radius r, welche die vorgegebenen Punkte enthält, für die also $\|a_i - x\|_2 \leq r$, $i = 1, \ldots, l$. Mit der Variablentransformation $r = \sqrt{2\delta}$ erhält man die Aufgabe:

$$\begin{cases} \text{Minimiere} \quad f(\delta, x) := \delta \quad \text{auf} \\ M := \{(\delta, x) \in \mathbb{R} \times \mathbb{R}^n : \frac{1}{2}\|x - a_i\|_2^2 \leq \delta,\ i = 1, \ldots, l\}. \end{cases}$$

Dies ist also eine Optimierungsaufgabe mit einer linearen Zielfunktion und (einfachen) quadratischen Ungleichungsnebenbedingungen. □

4.2.2 Lineare Optimierungsaufgaben: Existenz und Dualität

Eine lineare Optimierungsaufgabe ist in *Normalform*, wenn sie in der Form

$$\text{(P)} \qquad \text{Minimiere} \quad c^T x \quad \text{auf} \quad M := \{x \in \mathbb{R}^n : x \geq 0,\ Ax = b\}$$

vorliegt, also alle Variablen vorzeichenbeschränkt sind, die übrigen Nebenbedingungen in Gleichungsform auftreten und es sich um eine Minimierungsaufgabe handelt. Dagegen kann man von einer *allgemeinen linearen Optimierungsaufgabe* sprechen, wenn gewisse Variable vorzeichenbeschränkt sind (o. B. d. A. seien es die ersten, was durch eine Umnumerierung der Variablen erreicht werden kann), wenn ein Teil der Restriktionen in Ungleichungsform (o. B. d. A. seien diese gleichgerichtet, was notfalls durch eine Multiplikation mit -1 erreicht werden kann), ein anderer in Gleichungsform vorliegt, es sich schließlich um eine Minimierungsaufgabe handelt (was ebenfalls durch eine

Multiplikation mit -1 erreicht werden kann). Ein solches Programm ist daher durch

$$\text{Minimiere} \quad \sum_{j=1}^{n} c_j x_j \quad \text{auf}$$

$$M := \left\{ x \in \mathbb{R}^n : x_j \geq 0 \quad (j = 1, \ldots, n_0), \quad \begin{array}{ll} \displaystyle\sum_{j=1}^{n} a_{ij} x_j \geq b_i & (i = 1, \ldots, m_0), \\ \displaystyle\sum_{j=1}^{n} a_{ij} x_j = b_i & (i = m_0 + 1, \ldots, m) \end{array} \right\}$$

gegeben. Hierbei ist $n \in \mathbb{N}$ die Anzahl der Variablen, n_0 mit $0 \leq n_0 \leq n$ die Anzahl der vorzeichenbeschränkten Variablen (die $n - n_0$ übrigen heißen *frei*), $m \in \mathbb{N}$ die Anzahl der Restriktionen und m_0 mit $0 \leq m_0 \leq m$ die Anzahl der Ungleichungsrestriktionen. Es kann praktisch sein, zu einer Vektor-Matrix-Schreibweise überzugehen. Hierzu sei zunächst $A = (a_{ij}) \in \mathbb{R}^{m \times n}$, $b = (b_i) \in \mathbb{R}^m$, $c = (c_j) \in \mathbb{R}^n$. Vektoren des $\mathbb{R}^m$ denke man sich zerlegt in einen Anteil, der aus den ersten m_0 Komponenten besteht, und einem zweiten Anteil, in dem die restlichen $m - m_0$ Komponenten zusammengefasst sind. Für den Vektor b sei etwa

$$b = \begin{pmatrix} b^{(1)} \\ b^{(2)} \end{pmatrix}.$$

Entsprechendes kann auch für Vektoren aus dem $\mathbb{R}^n$ geschehen, so dass der Variablenvektor $x = (x_j) \in \mathbb{R}^n$ und der Kostenvektor $c \in \mathbb{R}^n$ zerlegt werden können:

$$x = \begin{pmatrix} x^{(1)} \\ x^{(2)} \end{pmatrix}, \qquad c = \begin{pmatrix} c^{(1)} \\ c^{(2)} \end{pmatrix}.$$

Zerlegt man auch die Matrix A durch

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix},$$

so lautet obige "allgemeine" lineare Optimierungsaufgabe in Matrix-Vektor-Schreibweise

$$\text{Minimiere} \quad \begin{pmatrix} c^{(1)} \\ c^{(2)} \end{pmatrix}^T \begin{pmatrix} x^{(1)} \\ x^{(2)} \end{pmatrix} \quad \text{auf}$$

$$M := \left\{ \begin{pmatrix} x^{(1)} \\ x^{(2)} \end{pmatrix} : x^{(1)} \geq 0, \quad \begin{array}{l} A_{11} x^{(1)} + A_{12} x^{(2)} \geq b^{(1)}, \\ A_{21} x^{(1)} + A_{22} x^{(2)} = b^{(2)} \end{array} \right\}.$$

Ganz wichtig ist die Bemerkung, dass man dieses "allgemeine" lineare Programm auf äquivalente Normalform zurückführen kann. Hierzu stelle man die freien Variablen x_j, $j = n_0 + 1, \ldots, n$, als Differenz nichtnegativer Variabler x_j^+ und x_j^- dar: $x_j = x_j^+ - x_j^-$, $j = n_0 + 1, \ldots, n$, und führe nichtnegative *Schlupfvariable* y_i, $i = 1, \ldots, m_0$, ein, um die Ungleichungen in äquivalente Gleichungen zu transformieren:

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \iff \sum_{j=1}^{n} a_{ij} x_j - y_i = b_i, \quad y_i \geq 0.$$

In Matrix-Vektor-Schreibweise geht man also von der Darstellung

$$x^{(2)} = x_+^{(2)} - x_-^{(2)}, \qquad x_+^{(2)} \geq 0, \quad x_-^{(2)} \geq 0$$

aus und nutzt ferner aus, daß

$$A_{11}x^{(1)} + A_{12}x^{(2)} \geq b^{(1)} \iff A_{11}x^{(1)} + A_{12}x^{(2)} - y^{(1)} = b^{(1)}, \quad y^{(1)} \geq 0.$$

Insgesamt erhält man

$$\text{Minimiere} \quad \begin{pmatrix} c^{(1)} \\ c^{(2)} \\ -c^{(2)} \\ 0 \end{pmatrix}^T \begin{pmatrix} x^{(1)} \\ x_+^{(2)} \\ x_-^{(2)} \\ y^{(1)} \end{pmatrix} \quad \text{auf}$$

$$M' := \left\{ \begin{pmatrix} x^{(1)} \\ x_+^{(2)} \\ x_-^{(2)} \\ y^{(1)} \end{pmatrix} : \begin{pmatrix} x^{(1)} \\ x_+^{(2)} \\ x_-^{(2)} \\ y^{(1)} \end{pmatrix} \geq 0, \begin{pmatrix} A_{11} & A_{12} & -A_{12} & -I \\ A_{21} & A_{22} & -A_{22} & 0 \end{pmatrix} \begin{pmatrix} x^{(1)} \\ x_+^{(2)} \\ x_-^{(2)} \\ y^{(1)} \end{pmatrix} = \begin{pmatrix} b^{(1)} \\ b^{(2)} \end{pmatrix} \right\}$$

als "äquivalentes" lineares Programm in Normalform. Die Anzahl der Variablen hat sich hierbei erhöht, da Elemente aus M' genau

$$n_0 + (n - n_0) + (n - n_0) + m_0 = 2n - n_0 + m_0$$

Komponenten besitzen. Was bedeutet diese "Äquivalenz" aber genauer? Intuitiv dürfte dies klar sein: Einem Element aus M kann ein Element aus M' (und umgekehrt) zugeordnet werden, wobei die Zielfunktionswerte übereinstimmen. Die Formalitäten wollen wir nicht zu weit treiben, daher begnügen wir uns mit diesem Hinweis. Jedenfalls werden wir im folgenden von einem linearen Programm in Normalform ausgehen, sind uns aber sicher, dass entsprechende Aussagen für allgemeine lineare Optimierungsaufgaben gelten.

Ist die lineare Optimierungsaufgabe (P) in Normalform gegeben, so nennt man

(D) $$\text{Maximiere} \quad b^T y \quad \text{auf} \quad N := \{y \in \mathbb{R}^m : A^T y \leq c\}$$

die zu (P) *duale* lineare Optimierungsaufgabe. Entsprechend kann auch die duale Optimierungsaufgabe zu einer sich nicht in Normalform befindenden linearen Optimierungsaufgabe bestimmt werden: Man bringe diese zunächst in äquivalente Normalform und bilde anschließend die duale Aufgabe, wobei man versucht, eine möglichst einfache Form zu erhalten.

Bemerkung: Die zu (P) duale Aufgabe

(D) $$\text{Maximiere} \quad b^T y \quad \text{auf} \quad N := \{y \in \mathbb{R}^m : A^T y \leq c\}$$

ist äquivalent zu

$$\text{Maximiere} \quad (-b)^T y \quad \text{auf} \quad N := \{y \in \mathbb{R}^m : (-A^T)y \geq -c\}.$$

Eine Überführung in Normalform liefert

$$\text{Minimiere} \quad \begin{pmatrix} -b \\ b \\ 0 \end{pmatrix}^T \begin{pmatrix} y_+ \\ y_- \\ z \end{pmatrix} \quad \text{unter den Nebenbedingungen}$$

$$\begin{pmatrix} -A^T & A^T & -I \end{pmatrix} \begin{pmatrix} y_+ \\ y_- \\ z \end{pmatrix} = -c, \quad \begin{pmatrix} y_+ \\ y_- \\ z \end{pmatrix} \geq 0.$$

Das hierzu duale Problem ist

$$\text{Maximiere} \quad (-c)^T x \quad \text{unter den Nebenbedingungen} \quad \begin{pmatrix} -A \\ A \\ -I \end{pmatrix} x \leq \begin{pmatrix} -b \\ b \\ 0 \end{pmatrix}.$$

Dieses wiederum ist ganz offensichtlich äquivalent zu (P). Grob gesagt: Dualisieren von (D) liefert wieder das Ausgangsproblem (P). □

Beispiel: Das klassische Transportproblem ist (nach eventuellem Einführen eines den Überschuss aufnehmenden fiktiven Lagers) gegeben durch

$$\text{(P)} \qquad \text{Minimiere} \quad c^T x \quad \text{auf} \quad M := \{x = (x_{ij}) \in \mathbb{R}^{mn} : x \geq 0,\ Ax = b\},$$

wobei

$$A := \begin{pmatrix} e^T & 0^T & \cdots & 0^T \\ 0^T & e^T & & 0^T \\ \vdots & \vdots & \ddots & \vdots \\ 0^T & 0^T & \cdots & e^T \\ I & I & \cdots & I \end{pmatrix} \in \mathbb{R}^{(m+n)\times mn}, \quad b = \begin{pmatrix} l \\ k \end{pmatrix} \in \mathbb{R}^{m+n}, \quad c = (c_{ij}) \in \mathbb{R}^{mn}.$$

Hierbei besitzt $e \in \mathbb{R}^n$ nur Einsen als Komponenten und I ist die $n \times n$-Einheitsmatrix. Die duale Variable y besitzt $m + n$ Komponenten. Es liegt nahe, sie durch

$$y = \begin{pmatrix} u \\ v \end{pmatrix}$$

zu partitionieren. Wegen

$$A^T y = \begin{pmatrix} e & 0 & \cdots & 0 & I \\ 0 & e & \cdots & 0 & I \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & e & I \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}, \qquad b^T y = l^T u + k^T v$$

erhält man als duales Problem die Aufgabe

$$\text{(D)} \quad \begin{cases} \text{Maximiere} \quad l^T u + k^T v \quad \text{auf} \\ N := \{(u, v) \in \mathbb{R}^m \times \mathbb{R}^n : u_i + v_j \leq c_{ij},\ (i, j) \in \{1, \ldots, m\} \times \{1, \ldots, n\}\}. \end{cases}$$

□

Sehr einfach ist der folgende *schwache Dualitätssatz*, den wir daher auch ausnahmsweise beweisen wollen.

Satz 2.1 *Gegeben seien die lineare Optimierungsaufgabe*

(P) $$\textit{Minimiere} \quad c^T x \quad \textit{auf} \quad M := \{x \in \mathbb{R}^n : x \geq 0,\ Ax = b\}$$

und die hierzu duale lineare Optimierungsaufgabe

(D) $$\textit{Maximiere} \quad b^T y \quad \textit{auf} \quad N := \{y \in \mathbb{R}^m : A^T y \leq c\}.$$

Dann gilt:

1. *Ist $x \in M$ und $y \in N$, so ist $b^T y \leq c^T x$.*
2. *Ist $x^* \in M$, $y^* \in N$ und $b^T y^* = c^T x^*$, so ist x^* eine Lösung von (P) und y^* eine Lösung von (D).*

Beweis: Seien $x \in M$ und $y \in N$. Dann ist

$$b^T y = (Ax)^T y = x^T A^T y \leq x^T c = c^T x,$$

womit der erste Teil schon bewiesen ist.

Ist $x^* \in M$, $y^* \in N$ und $b^T y^* = c^T x^*$, sind ferner $x \in M$ und $y \in N$ beliebig, so erhält man durch eine Anwendung des ersten Teiles auf die Paare (x^*, y) bzw. (x, y^*), dass

$$b^T y \leq c^T x^* = b^T y^* \leq c^T x,$$

was zu zeigen war. □

Der schwache Dualitätssatz liefert eine *hinreichende Optimalitätsbedingung*: Ist ein $x^* \in M$ gegeben und existiert ein $y^* \in N$ mit $c^T x^* = b^T y^*$, so ist x^* eine Lösung von (P) (und y^* eine Lösung von (D)). Wir geben hierzu ein Beispiel.

Beispiel: Wir betrachten noch einmal das Problem, Güterwagen von den Rangierbahnhöfen A und B zu den Bahnhöfen R, S und T zu leiten (siehe voriger Unterabschnitt). Es lautete

(P)
$$\begin{cases} \text{Minimiere} \begin{pmatrix} 5 \\ 4 \\ 9 \\ 7 \\ 8 \\ 10 \end{pmatrix}^T \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} \quad \text{unter den Nebenbedingungen} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} = \begin{pmatrix} 18 \\ 12 \\ 11 \\ 10 \\ 9 \end{pmatrix}, \quad \begin{pmatrix} x_{AR} \\ x_{AS} \\ x_{AT} \\ x_{BR} \\ x_{BS} \\ x_{BT} \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \end{cases}$$

Das duale Problem hierzu lautet

$$\text{(D)} \qquad \begin{cases} \text{Maximiere} & \begin{pmatrix} 18 \\ 12 \end{pmatrix}^T \begin{pmatrix} u_A \\ u_B \end{pmatrix} + \begin{pmatrix} 11 \\ 10 \\ 9 \end{pmatrix}^T \begin{pmatrix} v_R \\ v_S \\ v_T \end{pmatrix} \\ \\ \text{unter den Nebenbedingungen} & \begin{array}{rcl} u_A + v_R & \leq & 5, \\ u_A + v_S & \leq & 4, \\ u_A + v_T & \leq & 9, \\ u_B + v_R & \leq & 7, \\ u_B + v_S & \leq & 8, \\ u_B + v_T & \leq & 10. \end{array} \end{cases}$$

Mit Hilfe von Maple hatten wir

$$\begin{pmatrix} x^*_{AR} & x^*_{AS} & x^*_{AT} \\ x^*_{BR} & x^*_{BS} & x^*_{BT} \end{pmatrix} = \begin{pmatrix} 8 & 10 & 0 \\ 3 & 0 & 9 \end{pmatrix}$$

als Lösung von (P) erhalten, die zugehörigen Kosten sind 191. "Per Hand" macht man sich leicht klar, dass die von Maple ausgespuckte Lösung in der Tat zulässig für (P) ist. Ihre Optimalität ist aber, da wir misstrauisch sind, streng genommen noch nicht bewiesen. Für einen mathematisch strengen Beweis lösen wir auch das duale Programm (D) mit Hilfe von Maple:

```
> with(simplex):
> ziel:=18*u_A+12*u_B+11*v_R+10*v_S+9*v_T:
> restr:=
> {u_A+v_R<=5,u_A+v_S<=4,u_A+v_T<=9,u_B+v_R<=7,u_B+v_S<=8,u_B+v_T<=10}:
> loesung:=maximize(ziel,restr);
```

$$loesung := \{u\_A = 8,\ u\_B = 10,\ v\_R = -3,\ v\_S = -4,\ v\_T = 0\}$$

```
> opt:=subs(loesung,ziel);
```

$$opt := 191$$

Als Lösung (oder, da wir misstrauisch sind: Lösungskandidat) gibt Maple an:

$$\begin{pmatrix} u^*_A \\ u^*_B \end{pmatrix} = \begin{pmatrix} 8 \\ 10 \end{pmatrix}, \qquad \begin{pmatrix} v^*_R \\ v^*_S \\ v^*_T \end{pmatrix} = \begin{pmatrix} -3 \\ -4 \\ 0 \end{pmatrix},$$

der zugehörige Zielfunktionswert ist 191. Die hinreichende Optimalitätsbedingung im schwachen Dualitätssatz gibt dann einen mathematisch strengen Beweis, dass die von Maple augegebenen Ergebnisse wirklich Lösungen von (P) bzw. (D) sind. □

Nun wollen wir noch auf den Existenzsatz und den starken Dualitätssatz der linearen Optimierung eingehen. Hierzu stellen wir ein berühmtes Resultat, das Farkas-Lemma, an den Anfang. Es wird nicht vollständig bewiesen, sondern in einer anschließenden Bemerkung eine Beweisidee angegeben.

Lemma 2.2 *Das System*

$$\text{(I)} \qquad Ax = b, \qquad x \geq 0$$

besitzt genau dann keine Lösung, wenn das System

$$\text{(II)} \qquad A^T y \leq 0, \qquad b^T y > 0$$

eine Lösung besitzt.

Bemerkung: Ein Teil des Beweises von Lemma 2.2 ist völlig trivial. Angenommen, (I) und (II) besitzen eine Lösung x bzw. y. Dann ist

$$0 < b^T y = (Ax)^T y = x^T A^T y \leq 0,$$

ein Widerspruch. Also können (I) und (II) nicht gleichzeitig lösbar sein. Nun nehmen wir an, (I) sei nicht lösbar. Das bedeutet, dass $b \notin K := \{Ax : x \geq 0\}$. Die Menge $K \subset \mathbb{R}^m$ ist offensichtlich konvex. *Angenommen*, es wäre schon bewiesen, dass K abgeschlossen ist. Dann kann der Projektionssatz für abgeschlossene, konvexe Mengen (siehe Aufgabe 8 in Abschnitt 4.1) angewandt werden. Dieser liefert die Existenz genau einer Lösung der Aufgabe

$$\text{Minimiere} \quad \|u - b\|_2, \quad u \in K,$$

nämlich die sogenannte Projektion von b auf K, diese bezeichnen wir mit $P_K(b)$. Die Projektion $P_K(b) \in K$ ist charakterisiert durch (die ebenfalls geometrisch einsichtige) Bedingung

$$(*) \qquad (b - P_K(b))^T (u - P_K(b)) \leq 0 \qquad \text{für alle } u \in K.$$

Wir setzen $y := b - P_K(b)$ (es ist $y \neq 0$ wegen $b \notin K$) und erhalten aus $(*)$ (setze $u := 0$), dass $0 \leq y^T P_K(b)$. Ferner folgt aus $(*)$, dass

$$(A^T y)^T x \leq y^T P_K(b) \qquad \text{für alle } x \geq 0,$$

und hieraus, dass $A^T y \leq 0$. Ferner ist

$$b^T y = (b - P_k(b))^T y + y^T P_K(b) = \|y\|_2^2 + \underbrace{y^T P_K(b)}_{\geq 0} \geq \|y\|_2^2 > 0.$$

Also ist y eine Lösung von (II). Bis auf den Beweis der Abgeschlossenheit von K ist dies ein vollständiger Beweis des Farkas-Lemmas. □

Es folgen nun der Existenzsatz und der starke Dualitätssatz der linearen Optimierung, die wir fast vollständig beweisen werden. Zunächst der *Existenzsatz*.

Satz 2.3 *Gegeben sei die lineare Optimierungsaufgabe*

$$\text{(P)} \qquad \textit{Minimiere} \quad c^T x \quad \textit{auf} \quad M := \{x \in \mathbb{R}^n : x \geq 0,\ Ax = b\}.$$

Ist $M \neq \emptyset$ *und* $\inf(\mathrm{P}) := \inf_{x \in M} c^T x > -\infty$, *so besitzt (P) eine Lösung, d. h. es existiert ein* $x^* \in M$ *mit* $c^T x^* \leq c^T x$ *für alle* $x \in M$.

Beweis: Wir haben zu zeigen, dass das System

$$\text{(I)} \qquad \begin{pmatrix} A \\ c^T \end{pmatrix} x = \begin{pmatrix} b \\ \inf\,(\mathrm{P}) \end{pmatrix}, \qquad x \geq 0$$

lösbar ist. Angenommen, dies sei nicht der Fall. Dann liefert das Farkas-Lemma die Existenz einer Lösung $(y, \delta) \in \mathbb{R}^m \times \mathbb{R}$ von

$$\text{(II)} \qquad \begin{pmatrix} A^T & c \end{pmatrix} \begin{pmatrix} y \\ \delta \end{pmatrix} \leq 0, \qquad \begin{pmatrix} b \\ \inf\,(\mathrm{P}) \end{pmatrix}^T \begin{pmatrix} y \\ \delta \end{pmatrix} > 0$$

bzw.

$$A^T y + \delta c \leq 0, \qquad b^T y + \delta \inf\,(\mathrm{P}) > 0.$$

Nach Voraussetzung ist $M \neq \emptyset$ bzw. (P) zulässig. Es existiert also ein $x \in \mathbb{R}^n$ mit $Ax = b$ und $x \geq 0$. Multipliziert man die erste Ungleichung in (II) mit diesem x und berücksichtigt man die zweite Ungleichung, so erhält man

$$b^T y + \delta c^T x \leq 0 < b^T y + \delta \inf\,(\mathrm{P}),$$

woraus $\delta < 0$ folgt. Mit $\hat{y} := -y/\delta$ ist dann $A^T \hat{y} \leq c$ bzw. $\hat{y}$ zulässig für das zu (P) duale Programm und $b^T \hat{y} > \inf\,(\mathrm{P})$, was ein Widerspruch zum schwachen Dualitätssatz ist. □

Und nun der *starke Dualitätssatz*.

Satz 2.4 *Gegeben seien die lineare Optimierungsaufgabe*

$$\text{(P)} \qquad \textit{Minimiere} \quad c^T x \quad \textit{auf} \quad M := \{x \in \mathbb{R}^n : x \geq 0,\ Ax = b\}$$

und die hierzu duale lineare Optimierungsaufgabe

$$\text{(D)} \qquad \textit{Maximiere} \quad b^T y \quad \textit{auf} \quad N := \{y \in \mathbb{R}^m : A^T y \leq c\}.$$

Dann gilt:

1. *Ist $M \neq \emptyset$ und $N \neq \emptyset$, so besitzen (P) und (D) jeweils eine Lösung x^* bzw. y^* und es ist $c^T x^* = b^T y^*$.*
2. *Ist $M \neq \emptyset$ und $N = \emptyset$, so ist $\inf_{x \in M} c^T x = -\infty$, die Zielfunktion von (P) ist also auf der Menge M der zulässigen Lösungen von (P) nicht nach unten beschränkt.*
3. *Ist $M = \emptyset$ und $N \neq \emptyset$, so ist $\sup_{y \in N} b^T y = +\infty$, die Zielfunktion von (D) ist also auf der Menge N der zulässigen Lösungen von (D) nicht nach oben beschränkt.*

Beweis: Die Programme (P) und (D) besitzen wegen des Existenzsatzes der linearen Optimierung jeweils eine Lösung x^* bzw. y^*, da es sich hier um zulässige Aufgaben handelt, deren Zielfunktionen auf der Menge der primal bzw. dual zulässigen Lösungen nach unten bzw. oben beschränkt sind. Wir zeigen, dass

$$\text{(I)} \qquad \begin{pmatrix} A \\ c^T \end{pmatrix} x = \begin{pmatrix} b \\ b^T y^* \end{pmatrix}, \qquad x \geq 0,$$

lösbar ist, woraus dann mit Hilfe des schwachen Dualitätssatzes der erste Teil folgt. Angenommen, das ist nicht der Fall. Mit Hilfe des Farkas-Lemmas folgt wie beim Beweis des Existenzsatzes (ersetze nur inf (P) durch $b^T y^*$) die Existenz eines dual zulässigen $\hat{y}$ mit $b^T \hat{y} > b^T y^*$, was natürlich ein Widerspruch dazu ist, dass y^* eine Lösung von (D) ist.

Nun zum Beweis des zweiten Teiles des starken Dualitätssatzes. Wegen $N = \emptyset$ gibt es kein $y \in \mathbb{R}^m$ mit $A^T y \leq c$. Hieraus folgt, dass (führe eine nichtnegative Schlupfvariable z ein und stelle y als Differenz nichtnegativer Vektoren dar) auch das System

$$\text{(I)} \qquad \begin{pmatrix} A^T & -A^T & I \end{pmatrix} \begin{pmatrix} y_+ \\ y_- \\ z \end{pmatrix} = c, \qquad \begin{pmatrix} y_+ \\ y_- \\ z \end{pmatrix} \geq 0,$$

nicht lösbar ist. Aus dem Farkas-Lemma folgt die Existenz eines Vektors $p \in \mathbb{R}^n$ mit

$$\begin{pmatrix} A \\ -A \\ I \end{pmatrix} p \leq 0, \qquad c^T p > 0$$

bzw.

$$Ap = 0, \quad p \leq 0, \quad c^T p > 0.$$

Mit einem beliebigen $z \in M$ (ein solches existiert, da wir $M \neq \emptyset$ vorausgesetzt haben) ist $z - tp \in M$ für alle $t \geq 0$ und $c^T(z - tp) \to -\infty$ mit $t \to \infty$, womit auch der zweite Teil des starken Dualitätssatzes bewiesen ist.

Den Beweis für den dritten Teil des starken Dualitätssatzes zu führen, stellen wir als Aufgabe, siehe Aufgabe 1. □

4.2.3 Lineare Optimierungsaufgaben: Matrixspiele

Wir wollen hier nur auf Zwei-Personen-Nullsummen-Matrixspiele und insbesondere den Hauptsatz[10] der Theorie der Matrixspiele (John von Neumann) eingehen.

Zwei Personen D und P spielen ein Spiel. Jeder von ihnen hat hierbei eine endliche Menge von Handlungsmöglichkeiten, nämlich $S = \{s_1, \ldots, s_m\}$ für D und $T = \{t_1, \ldots, t_n\}$ für P. Vor Beginn des Spiels ist bekannt: Wird s_i von D und t_j von P gewählt, so hat der Spieler P an den Spieler D einen Betrag von a_{ij} Geldeinheiten zu zahlen. Dieser Betrag kann natürlich auch negativ sein, so dass P in diesem Falle von D in Wahrheit etwas erhält. In jedem Fall ist der Gewinn des einen der Verlust des anderen, daher der Name Nullsummen-Spiele. Durch $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ ist die sogenannte *Auszahlungsmatrix* des Spiels gegeben, sie ist beiden Spielern bekannt.

Eines der bekanntesten Beispiele hierzu ist das "Knobel-Spiel" Schere-Stein-Papier: Stein schlägt Schere, Schere schlägt Papier, Papier schlägt Stein. "Klassischerweise"

[10] Bei

A. Schrijver (1986, S. 218) *Theory of Linear and Integer Programming*. J. Wiley & Sons

kann man etwas zur Geschichte dieses Satzes nachlesen.

erhält man die folgende Auszahlungsmatrix:

D \ P	Stein	Schere	Papier
Stein	0	1	−1
Schere	−1	0	1
Papier	1	−1	0

Bei diesem Spiel, das ja i. allg. nicht nur einmal, sondern mehrmals hintereinander gespielt wird, weiß man intuitiv oder aus eigener Erfahrung, dass man jede der sogenannten *reinen Strategien* aus S oder T, hier für beide Spieler "Stein", "Schere" und "Papier", mit derselben Wahrscheinlichkeit $\frac{1}{3}$ spielen, also zu der sogenannten *gemischten Strategie* $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ übergehen sollte.

Allgemeiner werden

$$X := \{x \in \mathbb{R}^n : x \geq 0,\ e^T x = 1\}, \qquad Y := \{y \in \mathbb{R}^m : y \geq 0,\ e^T y = 1\}$$

als Mengen der gemischten Strategien für Spieler P bzw. D bezeichnet. Hier und im folgenden ist e immer der Vektor des $\mathbb{R}^n$ bzw. $\mathbb{R}^m$, dessen Komponenten alle gleich Eins sind. Wählt Spieler P ein $x \in X$, so besagt dies, dass er für $j = 1, \ldots, n$ seine Handlungsmöglichkeit t_j mit der Wahrscheilichkeit x_j spielt.

Wählt Spieler D eine gemischte Strategie $y \in Y$ und Spieler P ein $x \in X$, so hat D von P eine Auszahlung von $y^T Ax$ Geldeinheiten zu erwarten. Der maximale Verlust von P bei Wahl einer gemischten Strategie $x \in X$ ist $\max_{y \in Y} y^T Ax$, diesen wird er versuchen zu minimieren. D. h. Spieler P löst die Aufgabe

$$\text{(P)} \qquad \text{Minimiere} \quad \phi(x) := \max_{y \in Y} y^T Ax, \quad x \in X.$$

Entsprechend ist der Mindestgewinn des Spielers D bei Wahl der gemischten Strategie $y \in Y$ durch $\min_{x \in X} y^T Ax$ gegeben, diesen wird er versuchen zu maximieren. Also hat Spieler D die Aufgabe

$$\text{(D)} \qquad \text{Maximiere} \quad \psi(y) := \min_{x \in X} y^T Ax, \quad y \in Y$$

zu lösen. Der folgende Satz, der sogenannte *Hauptsatz der Theorie der (Zwei-Personen-Nullsummen) Matrixspiele* sagt aus, dass der maximale Mindestgewinn von Spieler D gleich dem minimalen Maximalverlust von Spieler P ist. Dies geschieht dadurch, dass die beiden Optimierungsaufgaben als äquivalent zu linearen Programmen "entlarvt" werden, die zueinander dual sind. Der starke Dualitätssatz der linearen Optimierung wird dann die Behauptung liefern.

Satz 2.5 *Sei* $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ *und*

$$X := \{x \in \mathbb{R}^n : x \geq 0,\ e^T x = 1\}, \qquad Y := \{y \in \mathbb{R}^m : y \geq 0,\ e^T y = 1\},$$

wobei e *der Vektor aus dem* $\mathbb{R}^n$ *bzw.* $\mathbb{R}^m$ *ist, dessen Komponenten alle gleich Eins sind. Dann ist*

$$\max_{y \in Y} \min_{x \in X} y^T Ax = \min_{x \in X} \max_{y \in Y} y^T Ax.$$

Beweis: Wie in obiger Motivation betrachte man die beiden Aufgaben

$$\text{(P)} \qquad \text{Minimiere} \quad \phi(x) := \max_{y \in Y} y^T A x, \quad x \in X$$

und

$$\text{(D)} \qquad \text{Maximiere} \quad \psi(y) := \min_{x \in X} y^T A x, \quad y \in Y.$$

Dann ist

$$\phi(x) = \max_{i=1,\ldots,m} (Ax)_i, \qquad \psi(y) = \min_{j=1,\ldots,n} (A^T y)_j.$$

Denn: Bei vorgegebenem $x \in X$ und für beliebiges $y \in Y$ ist

$$y^T A x = \sum_{i=1}^{m} y_i (Ax)_i \le \max_{i=1,\ldots,m} (Ax)_i$$

und daher $\phi(x) \le \max_{i=1,\ldots,m}(Ax)_i$. Andererseits ist

$$\phi(x) \ge e_i^T A x = (Ax)_i, \qquad i = 1, \ldots, m,$$

und folglich $\phi(x) \ge \max_{i=1,\ldots,m}(Ax)_i$, insgesamt $\phi(x) = \max_{i=1,\ldots,m}(Ax)_i$. Entsprechend zeigt man $\psi(y) = \min_{j=1,\ldots,n}(A^T y)_j$. Daher sind (P) bzw. (D) äquivalent zu

$$\text{(P)} \qquad \left\{ \begin{array}{c} \text{Minimiere} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix}^T \begin{pmatrix} x \\ \alpha \end{pmatrix} \quad \text{auf} \\ M := \left\{ (x,\alpha) \in \mathbb{R}^n \times \mathbb{R} : x \ge 0, \quad \begin{array}{lcl} -Ax + \alpha e & \ge & 0, \\ e^T x & = & 1 \end{array} \right\} \end{array} \right.$$

bzw.

$$\text{(D)} \qquad \left\{ \begin{array}{c} \text{Maximiere} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix}^T \begin{pmatrix} y \\ \beta \end{pmatrix} \quad \text{auf} \\ N := \left\{ (y,\beta) \in \mathbb{R}^m \times \mathbb{R} : y \ge 0, \quad \begin{array}{lcl} -A^T y + \beta e & \le & 0, \\ e^T y & = & 1 \end{array} \right\}. \end{array} \right.$$

Diese beiden linearen Programme sind zulässig und offenbar dual zueinander. Der starke Dualitätssatz zeigt, dass beide lösbar sind (was mit Kompaktheitsargumenten auch leicht direkt gezeigt werden könnte) und $\max(\text{D}) = \min(\text{P})$ ist. Damit ist der Satz bewiesen. □

Bemerkung: Der Beweis des letzten Satzes zeigt, dass beide Spieler zur Berechnung für sie optimaler Strategien zueinander duale (speziell strukturierte) lineare Programme zu lösen haben. Dies wollen wir in dem folgenden Beispiel verdeutlichen. □

Beispiel: Sei die Auszahlungsmatrix eines Zwei-Personen-Nullsummen-Spiels durch

$$A := \begin{pmatrix} 2 & 1 & -1 \\ -1 & -2 & 3 \end{pmatrix}$$

gegeben. Wir wollen auf sehr einfache, nicht sehr elegante Weise das primale und das duale Programm der Spieler P und D mit Maple lösen:

```
> with(simplex):
> pziel:=alpha:
> prestr:={x1>=0,x2>=0,x3>=0,-2*x1-x2+x3+alpha>=0,
> x1+2*x2-3*x3+alpha>=0,x1+x2+x3=1}:
> minimize(pziel,prestr);
```

$$\{x1 = 0,\ x3 = \frac{3}{7},\ x2 = \frac{4}{7},\ \alpha = \frac{1}{7}\}$$

```
> dziel:=beta:
> drestr:={y1>=0,y2>=0,-2*y1+y2+beta<=0,-y1+2*y2+beta<=0,
> y1-3*y2+beta<=0,y1+y2=1}:


> maximize(dziel,drestr);
```

$$\{y1 = \frac{5}{7},\ y2 = \frac{2}{7},\ \beta = \frac{1}{7}\}$$

Spieler P hat also die optimale gemischte Strategie $x^* = (0, \frac{4}{7}, \frac{3}{7})^T$ und den Wert $\alpha^* = \frac{1}{7}$, während Spieler D die optimale gemischte Strategie $y^* = (\frac{5}{7}, \frac{2}{7})^T$ und den Wert $\beta^* = \frac{1}{7} = \alpha^*$ hat. Das Spiel zwischen P und D ist also mit der obigen Auszahlungsmatrix nicht fair in dem Sinne, dass der gemeinsame Wert von Null verschieden ist. In unserem Fall wird der Spieler D letzten Endes immer gewinnen. □

4.2.4 Lineare Optimierungsaufgaben: Netzwerkflussprobleme

Ein Produkt (Öl oder Orangen oder ...) wird in gewissen Orten in einer bestimmten Menge angeboten und an anderen Orten verlangt. Schließlich gibt es Orte, die nichts anbieten und nichts verlangen, in denen das Produkt aber umgeladen werden darf. Gewisse Orte sind miteinander durch Verkehrswege miteinander verbunden. Die Kosten für den Transport einer Mengeneinheit des Gutes längs eines Verkehrsweges sind bekannt, ferner ist die Kapazität eines jeden möglichen Transportweges vorgegeben. Diese gibt Obergrenzen für die zu transportierende Menge auf dem Weg an. Gesucht ist ein kostenminimaler Transportplan. Wir werden gleich ein mathematisches Modell für diese Aufgabenstellung angeben und danach kurz auf wenigstens einen Spezialfall eingehen.

Gegeben sei ein *gerichteter Graph* bzw. *Digraph* $(\mathcal{N}, \mathcal{A})$. Hier steht $\mathcal{N}$ für die (endliche) Menge der *Knoten* (**N**odes) und $\mathcal{A}$ für die Menge der *Pfeile* (**A**rcs), also *geordneten* Paaren von Knoten. Mit jedem Knoten $k \in \mathcal{N}$ ist eine (i. allg. ganzzahlige) Mengenangabe b_k des im Digraphen zu transportierenden Gutes verbunden. Ist $b_k > 0$, so sind b_k Mengeneinheiten dieses Gutes im Knoten k vorhanden und Knoten k wird ein *Angebotsknoten* genannt. Ist dagegen $b_k < 0$, so werden dort $|b_k|$ Mengeneinheiten benötigt, man spricht von einem *Bedarfsknoten*. Im Fall $b_k = 0$ handelt es sich um einen *Umladeknoten*.

Zu jedem Pfeil $(i, j) \in \mathcal{A}$ des Digraphen gehören die Kosten c_{ij} für den Fluss einer Mengeneinheit auf ihm. Mit x_{ij} wird der Fluss auf diesem Pfeil bezeichnet, die *Kapazität* des Pfeils wird durch (i. Allg. ganzzahliges) $u_{ij} > 0$ angegeben. Gesucht wird ein Fluss im Digraphen, der unter Berücksichtigung der Kapazitätsbeschränkungen die

Angebote und "Bedarfe" mengenmäßig ausgleicht und die dafür erforderlichen Kosten minimiert. Dabei ist in jedem Knoten der Fluss zu erhalten. Dies bedeutet für den Knoten $k \in \mathcal{N}$, dass die Summe der Flüsse auf seinen eingehenden Pfeilen plus der in ihm verfügbaren (wenn k ein Angebotsknoten) beziehungsweise minus der von ihm benötigten (wenn k ein Bedarfsknoten) Menge $|b_k|$ gleich der Summe der Flüsse auf seinen ausgehenden Pfeilen ist. Die Flusserhaltungsbedingung für den Knoten k lautet daher

$$\sum_{i:(i,k)\in\mathcal{A}} x_{ik} + b_k = \sum_{j:(k,j)\in\mathcal{A}} x_{kj}.$$

Das kapazitierte lineare Netzwerkflussproblem lässt sich daher wie folgt formulieren:

$$\begin{cases} \text{Minimiere} \quad \displaystyle\sum_{(i,j)\in\mathcal{A}} c_{ij}x_{ij} \\ \text{unter den Nebenbedingungen} \\ \displaystyle\sum_{j:(k,j)\in\mathcal{A}} x_{kj} - \sum_{i:(i,k)\in\mathcal{A}} x_{ik} = b_k \qquad (k \in \mathcal{N}), \qquad 0 \le x_{ij} \le u_{ij} \qquad ((i,j) \in \mathcal{A}). \end{cases}$$

Diese Aufgabe wollen wir nun in Matrix-Vektorschreibweise formulieren. Dies kann folgendermaßen geschehen. Der Fluss $x = (x_{ij})$ hat so viele Komponenten wie es Pfeile gibt, ihre Anzahl sei $n := |\mathcal{A}|$. Es liegt also nahe, $\mathcal{A}$ durchzunummerieren. Es sei etwa $\mathcal{A} = \{l_1, \ldots, l_n\}$. Dann kann $x = (x_{ij})_{(i,j)\in\mathcal{A}}$ als Vektor $x = (x_1, \ldots, x_n)^T$ mit $x_p = x_{l_p}$, $p = 1, \ldots, n$, geschrieben werden, entsprechendes gilt für die Kosten $c = (c_{ij})$ und Kapazitäten $u = (u_{ij})$. Ist ferner $m := |\mathcal{N}|$ die Anzahl der Knoten, so kann man $(b_k)_{k\in\mathcal{N}}$ zu einem Vektor $b = (b_1, \ldots, b_m)^T$ zusammenfassen. Definiert man die *Knoten-Pfeil-Inzidenzmatrix* $A = (a_{kp}) \in \mathbb{R}^{m\times n}$ durch

$$a_{kp} := \begin{cases} +1, & \text{falls: Der Knoten } k \text{ ist Startknoten für den } p\text{-ten Pfeil } l_p, \\ -1, & \text{falls: Der Knoten } k \text{ ist Endknoten für den } p\text{-ten Pfeil } l_p, \\ 0 & \text{sonst,} \end{cases}$$

so erkennt man, dass obiges Netzwerkflussproblem, das sogenannte (kapazitierte) Minimale-Kosten-Fluss-Problem, in der Form

$$\text{(P)} \qquad \text{Minimiere} \quad c^T x \quad \text{auf} \quad M := \{x \in \mathbb{R}^n : 0 \le x \le u,\ Ax = b\}$$

geschrieben werden kann.

Beispiel: Natürlich kann auf ein gegebenes Minimale-Kosten-Fluss-Problem einfach ein zur Verfügung stehender "Solver" für lineare Optimierungsprobleme angewandt werden. Dabei wird allerdings die Struktur des Problems nicht berücksichtigt. Wir wollen ein spezielles Problem mit Hilfe von Maple lösen.

Gegeben sei der in Abbildung 4.5 angegebene Digraph, wobei rechts angegeben ist, welche Bedeutung die angegebenen Zahlen haben. Z. B. sind die Knoten 2 und 3 Umladeknoten, der Knoten 1 ein Angebots- und der Knoten 4 ein Bedarfsknoten. Als

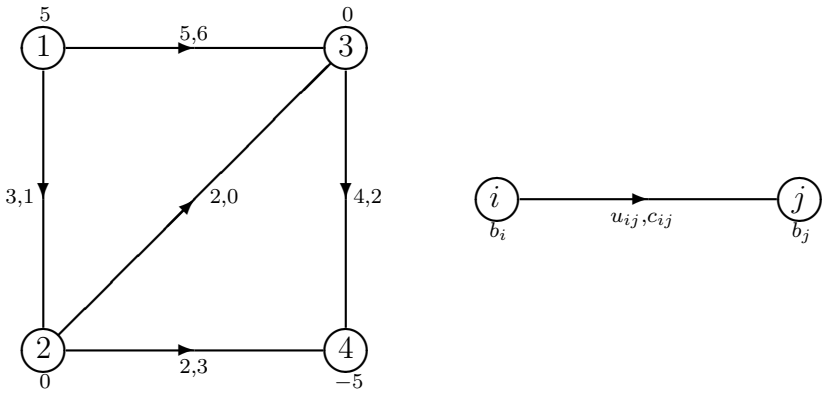

Abbildung 4.5: Ein Umladeproblem

Knoten-Pfeil-Inzidenzmatrix hat man

	$(1,2)$	$(1,3)$	$(2,3)$	$(2,4)$	$(3,4)$
1	1	1	0	0	0
2	−1	0	1	1	0
3	0	−1	−1	0	1
4	0	0	0	−1	−1

Das zugehörige lineare Programm lautet also

$$\text{Minimiere} \quad \begin{pmatrix} 1 \\ 6 \\ 0 \\ 3 \\ 2 \end{pmatrix}^T \begin{pmatrix} x_{12} \\ x_{13} \\ x_{23} \\ x_{24} \\ x_{34} \end{pmatrix} \quad \text{unter den Nebenbedingungen}$$

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \begin{pmatrix} x_{12} \\ x_{13} \\ x_{23} \\ x_{24} \\ x_{34} \end{pmatrix} \leq \begin{pmatrix} 3 \\ 5 \\ 2 \\ 2 \\ 4 \end{pmatrix}, \quad \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{pmatrix} \begin{pmatrix} x_{12} \\ x_{13} \\ x_{23} \\ x_{24} \\ x_{34} \end{pmatrix} = \begin{pmatrix} 5 \\ 0 \\ 0 \\ -5 \end{pmatrix}.$$

Eine Lösung mit Maple könnte folgendermaßen aussehen:

```
> with(simplex):
> ziel:=x12+6*x13+3*x24+2*x34:
> restr:={x12<=3,x13<=5,x23<=2,x24<=2,x34<=4,x12+x13=5,-x12+x23+x24=0,
> -x13-x23+x34=0,-x24-x34=-5}:
> loesung:=minimize(ziel,restr,NONNEGATIVE);
```

$$loesung := \{x24 = 1,\ x13 = 2,\ x12 = 3,\ x23 = 2,\ x34 = 4\}$$

```
> kosten:=subs(loesung,ziel);
```

$$kosten := 26$$

Als kostenminimalen Fluss erhält man also

$$\begin{pmatrix} x_{12}^* \\ x_{13}^* \\ x_{23}^* \\ x_{24}^* \\ x_{34}^* \end{pmatrix} = \begin{pmatrix} 3 \\ 2 \\ 2 \\ 1 \\ 4 \end{pmatrix},$$

die zugehörigen minimalen Kosten sind 26. □

Wir wollen nun noch auf das *Maximaler-Fluss-Problem* (*maximum flow problem*) eingehen. Hierbei ist ein gerichteter Graph $(\mathcal{N}, \mathcal{A})$ gegeben, in dem zwei Knoten s (Quelle, **s**ource, kein Pfeil endet in s) und t (Senke, **t**erminal, kein Pfeil startet in t) ausgezeichnet sind. Längs der Pfeile sind wieder Kapazitäten festgelegt. Es wird angenommen, dass es eine die Quelle s und die Senke t verbindenden (Vorwärts-) Pfad gibt. Es wird nach dem maximalen Fluss von s nach t gefragt, also nach der maximalen Anzahl der Mengeneinheiten, die bei s losgeschickt werden können und in t ankommen, wobei natürlich die Kapazitätsbeschränkungen zu berücksichtigen sind. Es ist möglich, dieses Problem als ein Minimale-Kosten-Fluss-Problem zu formulieren. Hierzu fassen wir alle Knoten als reine Umladeknoten auf, die Kosten längs jeden Pfeils werden auf Null gesetzt, es ist also $c_{ij} := 0$ für alle $(i,j) \in \mathcal{A}$. Ferner führe man einen Pfeil (t,s) von der Senke zur Quelle ein und definiere $c_{ts} := -1$ und $u_{ts} := +\infty$. Dieser Pfeil kann also beliebig viel aufnehmen. Die Aufgabe könnte also formuliert werden als:

$$\text{Minimiere} \quad -x_{ts}$$

unter den Nebenbedingungen

$$\sum_{j:(k,j)\in\mathcal{A}} x_{kj} - \sum_{i:(i,k)\in\mathcal{A}} x_{ik} = 0, \qquad k \in \mathcal{N} \setminus \{s,t\},$$

(alles was in einem Knoten $k \in \mathcal{N} \setminus \{s,t\}$ ankommt, fließt dort auch weg),

$$\sum_{j:(s,j)\in\mathcal{A}} x_{sj} = \sum_{i:(i,t)\in\mathcal{A}} x_{it} = x_{ts},$$

(alles was bei s wegfließt, kommt bei t an und fließt schließlich wieder (über den künstlichen Pfeil) nach t) sowie

$$0 \le x_{ij} \le u_{ij}, \qquad (i,j) \in \mathcal{A}.$$

Offenbar ist dann ein kostenminimaler Fluss ein Maximalfluss.

Etwas natürlicher ist es aber vielleicht, direkt die zum Maximaler-Fluss-Problem gehörende lineare Optimierungsaufgabe aufzustellen. Die Knoten seien so nummeriert, dass die Quelle s der erste und die Senke t der letzte bzw. der m-te Knoten ist. Mit $A = (a_{kp}) \in \mathbb{R}^{m\times n}$ bezeichnen wir wieder die Knoten-Pfeil-Inzidenzmatrix. Es ist also $a_{kp} = +1$, wenn der k-te Knoten Startknoten für den p-ten Pfeil ist, $a_{kp} = -1$, wenn der k-te Knoten Endknoten für den p-ten Pfeil ist, und $a_{kp} = 0$ in allen anderen Fällen. Ist

$x = (x_1, \ldots, x_n)^T$ (hierbei bedeutet x_p den Fluss auf dem p-ten Pfeil), so ist $(Ax)_1 = \sum_{p=1}^{n} a_{1p}x_p$ der an der Quelle austretende Fluss. Diesen gilt es unter Berücksichtigung der Kapazitätsbeschränkungen auf den Pfeilen und der Flussbedingung $(Ax)_k = 0$, $k = 2, \ldots, m-1$, zu minimieren. Weiter ist $(Ax)_m = -(Ax)_1$ (Beweis?). Definiert man daher noch den Vektor $d \in \mathbb{R}^m$ durch

$$d_k := \begin{cases} -1 & \text{für} \quad k = 1, \\ 0 & \text{für} \quad k = 2, \ldots, m-1, \\ +1 & \text{für} \quad k = m, \end{cases}$$

so erkennt man, dass das Maximaler-Fluss-Problem als lineare Optimierungsaufgabe

$$\text{Maximiere} \quad v \quad \text{auf} \quad \{(x, v) \in \mathbb{R}^n \times \mathbb{R} : Ax + dv = 0,\ 0 \le x \le u\}$$

formuliert werden kann. Wieder ist es einleuchtend, dass zur Lösung dieses linearen Programms die spezielle Struktur ausgenutzt werden sollte.

Beispiel: In der folgenden Abbildung geben wir einen Digraphen mit 8 Knoten und 15 Pfeilen an, eingetragen sind ferner die Kapazitäten längs der Pfeile. Was ist der

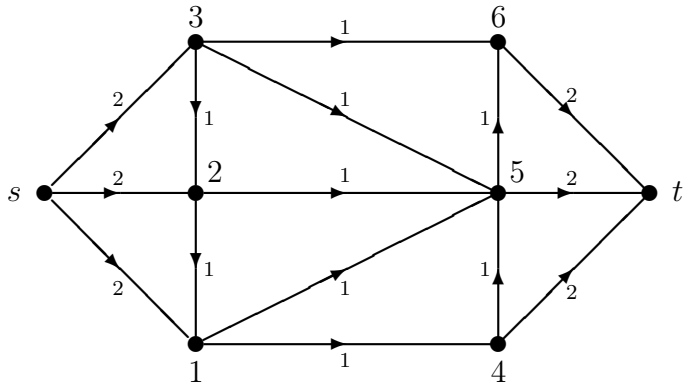

Abbildung 4.6: Ein Digraph mit 8 Knoten und 15 Pfeilen

maximale Fluss? Klar ist, dass dieser nicht größer als 6 sein kann, da die drei Pfeile weg von der Quelle nur eine Gesamtkapazität von 6 besitzen.

In der Abbildung 4.7 geben wir einen Fluss mit dem Wert 5 an. Gibt es auch einen mit dem Wert 6? Das obige lineare Programm ist in diesem Falle eine Aufgabe mit 16 Variablen und 8 Gleichungsrestriktionen. Es gibt hier auch eine nichtganzzahlige Lösung, siehe z. B. die in Abbildung 4.8. □

Wir wollen jetzt noch die Möglichkeiten von Maple zur Lösung des Maximaler-Fluss-Problems untersuchen. Im `networks` package von Maple gibt es die Funktion `flow`, mit welcher der maximale Fluss von einer Quelle s in eine Senke t berechnet werden kann.

Beispiel: Gegeben sei der gewichtete Digraph in Abbildung 4.6. Zunächst wird das `networks`-package geladen, anschließend der Digraph G erzeugt. Danach wird die `flow`-Funktion aufgerufen. Es wird der maximale Fluss ausgegeben, ferner eine Menge saturierter Pfeile (auf diesen ist der Fluss gleich der Kapazität) und eine gewisse Menge von Ecken:

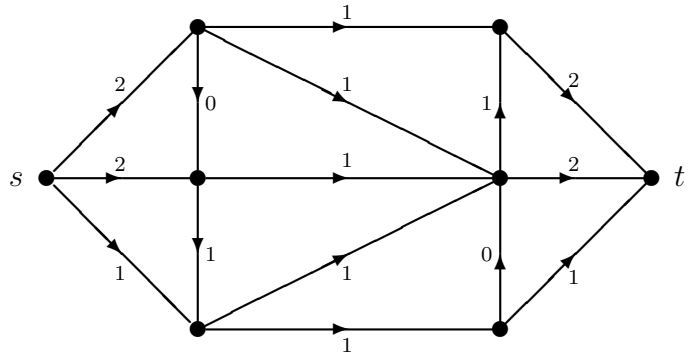

Abbildung 4.7: Ein Fluss mit dem Wert 5

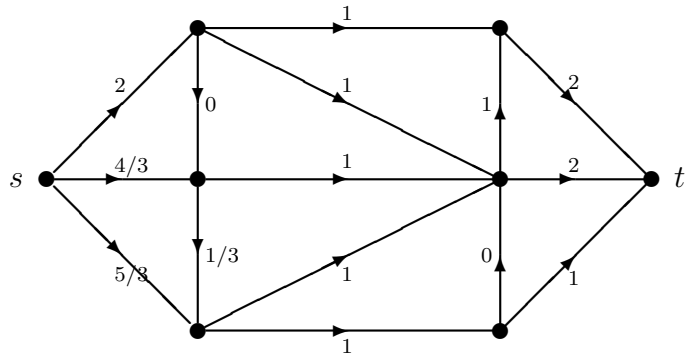

Abbildung 4.8: Ebenfalls ein Fluss mit dem Wert 5

```
> with(networks):
> new(G):
> addvertex({s,t,1,2,3,4,5,6},G):
> GewPfeile:=[[s,1,2],[s,2,2],[s,3,2],[2,1,1],[3,2,1],[1,4,1],
> [1,5,1],[2,5,1],[3,5,1],[3,6,1],[4,5,1],[4,t,2],[5,t,2],[5,6,1],
> [6,t,2]]:
> for i from 1 to nops(GewPfeile) do
> addedge([GewPfeile[i][1],GewPfeile[i][2]],weights=GewPfeile[i][3],G):
> end do:
> flow(G,s,t,'satpfeile','comp');
```

$$5$$

```
> satpfeile;
```

$$\{\{1, s\}, \{3, s\}, \{1, 4\}, \{1, 5\}, \{2, 5\}, \{3, 6\}, \{5, 6\}, \{3, 5\}, \{5, t\}, \{6, t\}\}$$

```
> comp;
```

$$\{1, 2, s\}$$

In Abbildung 4.9 geben wir den hiermit erhaltenen Fluss wieder. Man erkennt, dass er

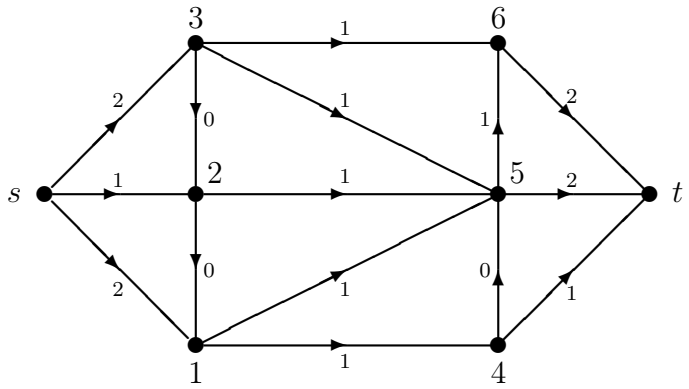

Abbildung 4.9: Ein maximaler Fluss

mit dem in Abbildung 4.7 angegebenen Fluss nicht übereinstimmt. An der Nichteindeutigkeit liegt es, dass es auch nichtganzzahlige Lösungen gibt, da die Konvexkombination von zwei Lösungen natürlich auch eine Lösung ist. □

Bemerkung: Sehr kurz wollen wir das berühmte Max-Flow Min-Cut Theorem von Ford-Fulkerson schildern. Gegeben sei wieder ein Digraph $(\mathcal{N}, \mathcal{A})$, in dem eine Quelle s und eine Senke t ausgezeichnete Knoten sind. Die Pfeile $(i, j) \in \mathcal{A}$ haben jeweils gewisse Kapazitäten u_{ij}. Durch diese Daten ist das Maximaler-Fluss-Problem (Max-Flow Problem) gegeben.

Jetzt schildern wir das Minimaler-Schnitt-Problem (Min-Cut Problem). Ein *Schnitt* ist eine Partition der Knotenmenge $\mathcal{N}$ in zwei (disjunkte) Mengen $\mathcal{N}_1$ und $\mathcal{N}_2$ mit $s \in \mathcal{N}_1$ und $t \in \mathcal{N}_2$. Zu einem Schnitt $(\mathcal{N}_1, \mathcal{N}_2)$ definieren wir die zugehörige *Kapazität* $C(\mathcal{N}_1, \mathcal{N}_2)$ als die Summe aller Kapazitätsschranken über Pfeilen, die in $\mathcal{N}_1$ starten und in $\mathcal{N}_2$ enden, also in der oben eingeführten Notation durch

$$C(\mathcal{N}_1, \mathcal{N}_2) := \sum_{\substack{(i,j)\in\mathcal{A} \\ i\in\mathcal{N}_1,\, j\in\mathcal{N}_2}} u_{ij}.$$

Unter dem *Minimaler-Schnitt-Problem* (min-cut problem) versteht man die Aufgabe, einen Schnitt mit minimaler Kapazität zu bestimmen.

In Abbildung 4.10 geben wir einen Schnitt an. Die zu $\mathcal{N}_1 := \{s, 1, 2\}$ gehörenden Knoten sind durch $\circ$, solche zu $\mathcal{N}_2 := \{3, 4, 5, 6, t\}$ durch $\bullet$ gekennzeichnet. Hier gibt es vier Pfeile, die Knoten aus $\mathcal{N}_1$ mit Knoten aus $\mathcal{N}_2$ verbinden, die zugehörige Kapazität ist 5.

Das Max-Flow Min-Cut Theorem von Ford-Fulkerson sagt unter Benutzung obiger Bezeichnungen aus:

- Ist $x = (x_{ij})_{(i,j)\in\mathcal{A}}$ ein zulässiger Fluss mit dem Wert $v = \sum_{j:(s,j)\in\mathcal{A}} x_{sj}$ und ist $(\mathcal{N}_1, \mathcal{N}_2)$ ein Schnitt mit Kapazität $C(\mathcal{N}_1, \mathcal{N}_2)$, so ist $v \le C(\mathcal{N}_1, \mathcal{N}_2)$.

- Ist x^* eine Lösung des Maximaler-Fluss-Problems mit dem Wert

$$v^* = \sum_{j:(s,j)\in\mathcal{A}} x^*_{sj},$$

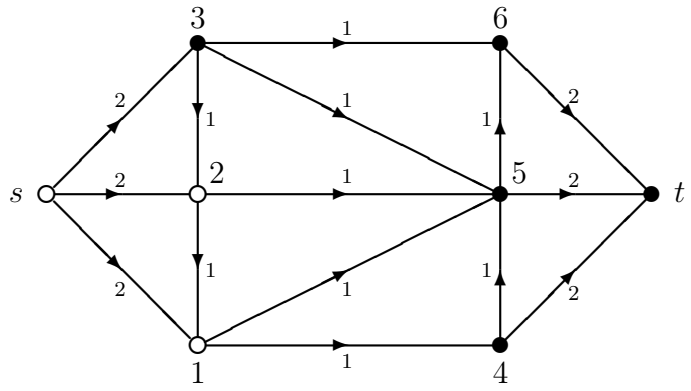

Abbildung 4.10: Ein Schnitt mit der Kapazität 5

so existiert ein Schnitt $(\mathcal{N}_1^*, \mathcal{N}_2^*)$ mit der Kapazität $C(\mathcal{N}_1^*, \mathcal{N}_2^*) = v^*$. Dieser Schnitt ist eine Lösung des Minimaler-Schnitt-Problems.

- Ist $(\mathcal{N}_1^*, \mathcal{N}_2^*)$ eine Lösung des Minimaler-Schnitt-Problems (da es nur endlich viele Schnitte gibt, und mindestens einen, wenn es eine die Quelle und die Senke verbindende Pfeilfolge gibt, hat das Minimaler-Schnitt-Problem trivialerweise eine Lösung), so gibt es einen zulässigen Fluss x^* mit dem Wert $v^* = C(\mathcal{N}_1^*, \mathcal{N}_2^*)$. Dieser Fluss ist eine Lösung des Maximaler-Fluss-Problems.

Dir Situation erinnert an die beim schwachen bzw. starken Dualitätssatz der linearen Optimierung. Man kann zeigen, dass das kein Zufall ist. □

Beispiel: Am kapazitierten Digraphen in Abbildung 4.6 wollen wir die `mincut`-Funktion im `networks`-package von Maple erläutern. Leider ist die Beschreibung ziemlich dürftig. Wir nehmen an, der Graph G sei wie oben erzeugt. Dann erhalten wir:

```
> M:=mincut(G,s,t,cap);
```

$$M := \{e3,\ e6,\ e7,\ e8\}$$

```
> cap;
```

$$5$$

```
> ends(M,G);
```

$$\{[1,\ 4],\ [1,\ 5],\ [s,\ 3],\ [2,\ 5]\}$$

Die minimale Kapazität ist also 5, ferner sind in M die Pfeile aufgeführt, die einen Beitrag zur Kapazität des Schnittes liefern, siehe Abbildung 4.10. □

4.2.5 Der Satz von Kuhn-Tucker

Wir betrachten in diesem Unterabschnitt eine Optimierungsaufgabe der Form

$$\text{(P)} \qquad \text{Minimiere} \quad \text{auf} \quad M := \{x \in \mathbb{R}^n : g(x) \leq 0,\ h(x) = 0\}.$$

Hierbei sind die Zielfunktion $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ und die Restriktionsabbildungen $g\colon \mathbb{R}^n \longrightarrow \mathbb{R}^l$ sowie $h\colon \mathbb{R}^n \longrightarrow \mathbb{R}^m$ gegeben, sie werden als "hinreichend glatt" (das wird im Einzelfall spezifiziert) vorausgesetzt. Wir nennen $x^* \in M$ eine (globale) *Lösung* von (P), wenn

$f(x^*) \leq f(x)$ für alle $x \in M$. Naheliegenderweise nennt man $x^* \in M$ eine *lokale Lösung* von (P), wenn es eine Umgebung U^* von x^* mit $f(x^*) \leq f(x)$ für alle $x \in M \cap U^*$ gibt.

Nun geben wir den wichtigen *Satz von Kuhn-Tucker* an, in dem notwendige Optimalitätbedingungen erster Ordnung formuliert werden. Wenigstens in einem Spezialfall (nämlich dem, dass keine Gleichungen als Restriktionen vorkommen) wollen wir diesen auch beweisen.

Satz 2.6 *Sei x^* eine lokale Lösung von*

$$\text{(P)} \qquad \textit{Minimiere} \quad f(x) \quad \textit{auf} \quad M := \{x \in \mathbb{R}^n : g(x) \leq 0,\ h(x) = 0\}.$$

Hierbei seien die Zielfunktion $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ und die Restriktionsabbildungen $g\colon \mathbb{R}^n \longrightarrow \mathbb{R}^l$ sowie $h\colon \mathbb{R}^n \longrightarrow \mathbb{R}^m$ auf einer Umgebung von x^* stetig differenzierbar. Mit

$$I(x^*) := \{i \in \{1, \ldots, l\} : g_i(x^*) = 0\}$$

wird die Indexmenge der sogenannten aktiven Ungleichungsrestriktionen bezeichnet. Es existiere ein $\hat{p} \in \mathbb{R}^n$ mit $\nabla g_i(x^*)^T \hat{p} < 0$ für alle $i \in I(x^*)$ und $h'(x^*)\hat{p} = 0$, ferner sei Rang $h'(x^*) = m$. *Dann existiert ein Paar $(u^*, v^*) \in \mathbb{R}^l \times \mathbb{R}^m$ mit*

$$u^* \geq 0, \qquad \nabla f(x^*) + g'(x^*)^T u^* + h'(x^*)^T v^* = 0, \qquad g(x^*)^T u^* = 0.$$

Bemerkung: Bevor wir Satz 2.6 wenigstens in einem Spezialfall beweisen, wollen wir uns seine Aussage an hand von Spezialfällen klar machen. Mit

$$g(x) = \begin{pmatrix} g_1(x) \\ \vdots \\ g_l(x) \end{pmatrix}, \qquad h(x) = \begin{pmatrix} h_1(x) \\ \vdots \\ h_m(x) \end{pmatrix}$$

sind die Funktionalmatrizen $g'(x^*)$ bzw. $h'(x^*)$ durch

$$g'(x^*) = \begin{pmatrix} \nabla g_1(x^*)^T \\ \vdots \\ \nabla g_l(x^*)^T \end{pmatrix}, \qquad h'(x^*) = \begin{pmatrix} \nabla h_1(x^*)^T \\ \vdots \\ \nabla h_m(x^*)^T \end{pmatrix}$$

gegeben. Zunächst betrachten wir die Aussage des Kuhn-Tucker Satzes, bei der keine Ungleichungsrestriktionen auftreten.

- *Sei x^* eine lokale Lösung von*

  $$\text{(P)} \qquad \textit{Minimiere} \quad f(x) \quad \textit{auf} \quad M := \{x \in \mathbb{R}^n : h(x) = 0\}.$$

  Hierbei seien $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ und $h\colon \mathbb{R}^n \longrightarrow \mathbb{R}^m$ auf einer Umgebung von x^* stetig differenzierbar. Es sei Rang $h'(x^*) = m$ *bzw. $\{\nabla h_1(x^*), \ldots, \nabla h_m(x^*)\}$ linear unabhängig. Dann existiert $v^* \in \mathbb{R}^m$ mit*

  $$\nabla f(x^*) + \sum_{i=1}^{m} v_i^* \nabla h_i(x^*) = 0.$$

Diese Aussage heißt auch Lagrangesche Multiplikatorenregel. Sie wird häufig schon in einer Vorlesung Analysis II (als Anwendung des Satzes über implizite Funktionen) bewiesen.

Nun betrachten wir den Spezialfall, dass nur Ungleichungen als Restriktionen auftreten.

- *Sei x^* eine lokale Lösung von*

  (P) $\qquad$ *Minimiere* $\quad f(x) \quad$ *auf* $\quad M := \{x \in \mathbb{R}^n : g(x) \leq 0\}.$

  Hierbei seien $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ und $g\colon \mathbb{R}^n \longrightarrow \mathbb{R}^l$ auf einer Umgebung von x^* stetig differenzierbar. Mit

  $$I(x^*) := \{i \in \{1, \ldots, l\} : g_i(x^*) = 0\}$$

  wird die Indexmenge der aktiven Ungleichungsrestriktionen bezeichnet. Es existiere ein $\hat{p} \in \mathbb{R}^n$ mit $\nabla g_i(x^*)^T \hat{p} < 0$ für alle $i \in I(x^*)$. Dann existiert ein $u^* \in \mathbb{R}^l$ mit

  $$u^* \geq 0, \qquad \nabla f(x^*) + \sum_{i=1}^{l} u_i^* \nabla g_i(x^*) = 0, \qquad g(x^*)^T u^* = 0.$$

Diese Aussage werden wir im Anschluss mit Hilfe des Farkas Lemmas beweisen. □

Beweis von Satz 2.6 (wenn keine Gleichungen auftreten): Angenommen, die Aussage sei nicht richtig. Dann existiert keine Lösung u_i^*, $i \in I(x^*)$, von

$$\sum_{i \in I(x^*)} u_i^* \nabla g_i(x^*) = -\nabla f(x^*), \qquad u_i^* \geq 0 \quad (i \in I(x^*))$$

(denn andernfalls setze man $u_i^* := 0, i \notin I(x^*)$, und hat ein gesuchtes $u^* \in \mathbb{R}^l$ gefunden). Das Farkas-Lemma 2.2 liefert die Existenz eines Vektors $q \in \mathbb{R}^n$ mit

$$\nabla g_i(x^*)^T q \leq 0 \quad (i \in I(x^*)), \qquad (-\nabla f(x^*))^T q > 0.$$

Nach Voraussetzung existiert ein $\hat{p} \in \mathbb{R}^n$ mit $\nabla g_i(x^*)^T \hat{p} < 0$, $i \in I(x^*)$. Nun bestimme man ein so kleines $s > 0$, dass $\nabla f(x^*)^T (q + s\hat{p}) < 0$ und setze anschließend $p := q + s\hat{p}$. Dann ist

$$\nabla g_i(x^*)^T p < 0 \quad (i \in I(x^*)), \qquad \nabla f(x^*)^T p < 0$$

und folglich $x^* + tp$ zulässig und $f(x^* + tp) < f(x^*)$ für alle hinreichend kleinen $t > 0$, ein Widerspruch dazu, dass x^* eine lokale Lösung von (P) ist. □

Bemerkung: Ist die Restriktionsabbildung g affin linear (und tritt keine oder nur eine affin lineare Gleichungsrestriktion auf), so zeigt der obige Beweis, dass keine Zusatzbedingung (auch *Constraint Qualification* genannt) nötig ist. Dies gilt aber i. Allg. nicht für nichtlineare Restriktionen, wie man durch Beispiele nachweisen kann. □

4.2.6 Beispiele zum Satz von Kuhn-Tucker

Beispiel: Gegeben sei die Optimierungsaufgabe (siehe Kies-Transport in einem früheren Beispiel)

$$\text{(P)} \qquad \begin{cases} \text{Minimiere} \quad f(x,y,z) := 1/(xyz) + yz \quad \text{unter den Nebenbedingungen} \\ \qquad\qquad g(x,y,z) := xy + 2xz - 4 \le 0, \qquad x,y,z > 0. \end{cases}$$

Sei (x^*, y^*, z^*) eine lokale Lösung. Der Satz von Kuhn-Tucker ist anwendbar (Begründung?) und liefert die Existenz einer nichtnegativen reellen Zahl u^* mit

$$-\frac{1}{x^*y^*z^*}\begin{pmatrix} 1/x^* \\ 1/y^* \\ 1/z^* \end{pmatrix} + \begin{pmatrix} 0 \\ z^* \\ y^* \end{pmatrix} + u^*\begin{pmatrix} y^* + 2z^* \\ x^* \\ 2x^* \end{pmatrix} = 0$$

und

$$u^*[x^*y^* + 2x^*z^* - 4] = 0.$$

Zur Lösung benutzen wir Maple:

```
> solve({-1/(x^2*y*z)+u*(y+2*z)=0,-1/(x*y^2*z)+z+u*x=0,
> -1/(x*y*z^2)+y+2*u*x=0,u*(x*y+2*x*z-4)=0},{x,y,z,u});
```

$$\{y = 1,\ x = 2,\ u = \frac{1}{4},\ z = \frac{1}{2}\},\ \{x = -2 - 2\,\%1,\ u = -\frac{1}{4} - \frac{1}{4}\,\%1,\ y = \%1,\ z = \frac{1}{2}\,\%1\}$$

$$\%1 := \mathrm{RootOf}(\_Z^2 + \_Z + 1,\ label = \_L1)$$

Aso erhalten wir

$$(x^*, y^*, z^*) = (2, 1, \tfrac{1}{2}), \qquad u^* = \tfrac{1}{4}$$

als eine reelle Lösung bzw. genauer als einen reellen Lösungskandidaten und einen zugehörigen Lagrange-Multiplikator. Denn auch wenn durch (x^*, y^*, z^*) und u^* ein sogenanntes Kuhn-Tucker-Paar gefunden ist, ist damit natürlich noch nicht bewiesen, dass in (x^*, y^*, z^*) eine lokale oder gar globale Lösung von (P) gegeben ist. □

Beispiel: Gegeben seien l Punkte $a_1, \ldots, a_l \in \mathbb{R}^n$, gesucht ist die euklidische Kugel $B[x;r] := \{y \in \mathbb{R}^n : \|y - x\|_2 \le r\}$ mit minimalem Radius r, welche die vorgegebenen Punkte enthält, für die also $\|a_i - x\|_2 \le r$, $i = 1, \ldots, l$. Mit der Variablentransformation $r = \sqrt{2\delta}$ erhält man die Aufgabe:

$$\text{(P)} \qquad \begin{cases} \text{Minimiere} \quad f(\delta, x) := \delta \quad \text{auf} \\ M := \{(\delta, x) \in \mathbb{R} \times \mathbb{R}^n : \frac{1}{2}\|x - a_i\|_2^2 \le \delta,\ i = 1, \ldots, l\}. \end{cases}$$

Mit $g_i(\delta, x) := -\delta + \frac{1}{2}\|x - a_i\|_2^2$ ist die Zusatzbedingung im Satz von Kuhn-Tucker erfüllt. Ist also (δ^*, x^*) die Lösung von (P) (es ist nicht schwierig, die Existenz und Eindeutigkeit einer Lösung von (P) nachzuweisen), so erhält man die Existenz eines Vektors $u^* \in \mathbb{R}^m$ mit

$$u^* \ge 0, \qquad \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \sum_{i=1}^{l} u_i^* \begin{pmatrix} -1 \\ x^* - a_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

sowie

$$u_i^*[\tfrac{1}{2}\|x^* - a_i\|_2^2 - \delta^*] = 0, \qquad i = 1, \ldots, l.$$

Aus den ersten Gleichungen folgt $x^* = \sum_{i=1}^{l} u_i^* a_i$ mit nichtnegativen u_i^*, deren Summe 1 ist, man erhält also (was intuitiv klar ist), dass x^* eine Konvexkombination der vorgegebenen a_i, $i = 1, \ldots, l$, ist. Für $l = 3$ und $n = 2$ sowie $a_1 = (50, 5)$, $a_2 = (95, 40)$ und $a_3 = (130, 0)$ berechnen wir aus den notwendigen Optimalitätsbedingungen mit Hilfe von Maple die Lösung. Wir erhalten sieben Lösungskandidaten:

```
> solve({x1=50*u1+95*u2+130*u3,x2=5*u1+40*u2,u1+u2+u3=1,
> u1*((1/2)*((x1-50)^2+(x2-5)^2)-delta)=0,
> u2*((1/2)*((x1-95)^2+(x2-40)^2)-delta)=0,
> u3*((1/2)*((x1-130)^2+x2^2)-delta)=0},{x1,x2,u1,u2,u3,delta});
```

$$\{u3 = 0,\ x2 = 40,\ \delta = 0,\ x1 = 95,\ u2 = 1,\ u1 = 0\},$$
$$\{u3 = 0,\ \delta = 0,\ x2 = 5,\ x1 = 50,\ u2 = 0,\ u1 = 1\},$$
$$\{u3 = 0,\ x2 = \frac{45}{2},\ \delta = \frac{1625}{4},\ x1 = \frac{145}{2},\ u2 = \frac{1}{2},\ u1 = \frac{1}{2}\},$$
$$\{u3 = 1,\ \delta = 0,\ x2 = 0,\ x1 = 130,\ u2 = 0,\ u1 = 0\},$$
$$\{u3 = \frac{1}{2},\ x2 = \frac{5}{2},\ x1 = 90,\ \delta = \frac{6425}{8},\ u2 = 0,\ u1 = \frac{1}{2}\},$$
$$\{u3 = \frac{1}{2},\ x2 = 20,\ x1 = \frac{225}{2},\ \delta = \frac{2825}{8},\ u2 = \frac{1}{2},\ u1 = 0\},$$
$$\{u3 = \frac{7800}{14641},\ u1 = \frac{15481}{29282},\ x2 = \frac{45}{242},\ x1 = \frac{21745}{242},\ \delta = \frac{47191625}{58564},\ u2 = \frac{-1799}{29282}\}$$

Die letzte hiervon ist

$$\delta^* = \tfrac{47191625}{58564}, \quad (x_1^*, x_2^*) = (\tfrac{45}{242}, \tfrac{21745}{242}), \quad (u_1^*, u_2^*, u_3^*) = (\tfrac{15481}{29282}, -\tfrac{1799}{29282}, \tfrac{7800}{14641}).$$

Da der zweite Multiplikator negativ ist, ist diese Lösung für uns irrelevant. Davor werden aber noch die offensichtlich für (P) nicht zulässigen Lösungen

$$\begin{array}{rclcrclcrcl}
\delta^* & = & 0, & & (x_1^*, x_2^*) & = & (50, 5), & & (u_1^*, u_2^*, u_3^*) & = & (1, 0, 0), \\
\delta^* & = & 0, & & (x_1^*, x_2^*) & = & (95, 40), & & (u_1^*, u_2^*, u_3^*) & = & (0, 1, 0), \\
\delta^* & = & 0, & & (x_1^*, x_2^*) & = & (130, 0), & & (u_1^*, u_2^*, u_3^*) & = & (0, 0, 1)
\end{array}$$

sowie

$$\begin{array}{rclcrclcrcl}
\delta^* & = & \frac{1625}{4}, & & (x_1^*, x_2^*) & = & (\frac{145}{2}, \frac{45}{2}), & & (u_1^*, u_2^*, u_3^*) & = & (\frac{1}{2}, \frac{1}{2}, 0), \\
\delta^* & = & \frac{2825}{8}, & & (x_1^*, x_2^*) & = & (\frac{225}{2}, 20), & & (u_1^*, u_2^*, u_3^*) & = & (0, \frac{1}{2}, \frac{1}{2}), \\
\delta^* & = & \frac{6425}{8}, & & (x_1^*, x_2^*) & = & (90, \frac{5}{2}), & & (u_1^*, u_2^*, u_3^*) & = & (\frac{1}{2}, 0, \frac{1}{2})
\end{array}$$

ausgegeben. Um die Zulässigkeit dieser drei Lösungen nachzuweisen, genügt es, die dritte, die erste bzw. die zweite Ungleichung zu betrachten (da der dritte, erste bzw. zweite Multiplikator verschwindet). Hierdurch erhält man, dass

$$\delta^* = \tfrac{6425}{8}, \quad (x_1^*, x_2^*) = (90, \tfrac{5}{2}), \quad (u_1^*, u_2^*, u_3^*) = (\tfrac{1}{2}, 0, \tfrac{1}{2})$$

hiervon die einzige zulässige Lösung ist. In Abbildung 4.11 ist diese eingezeichnet. Der minimale Radius eines die Punkte a_1, a_2, a_3 enthaltenden Kreises ist $r^* = \sqrt{2\delta^*} =$

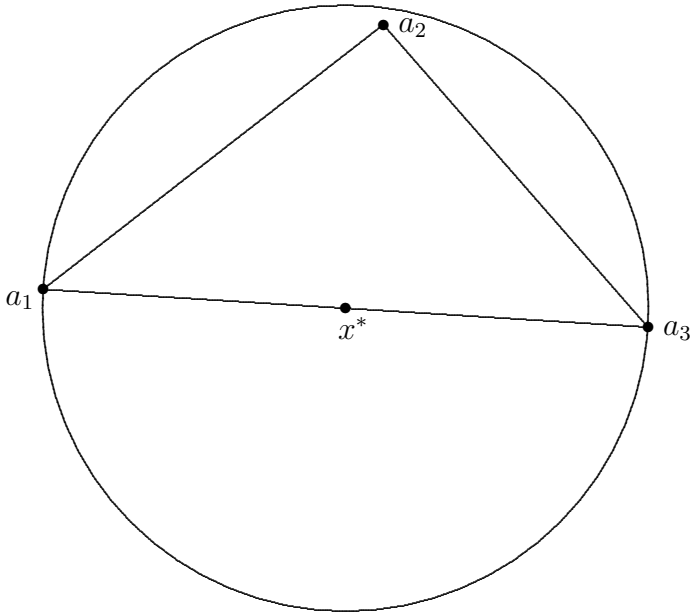

Abbildung 4.11: Lösung des Sylvester-Problems

$5\sqrt{257}/2 \approx 40.078$. Man könnte vermuten, der optimale Kreis sei gerade der, auf dem die Punkte a_1, a_2, a_3 liegen. Der Mittelpunkt und der Radius dieses Kreises ist

$$(x_1, x_2) = (89.8554, 0.18595), \qquad r \approx 40.1451.$$

Der Radius dieses Kreises ist hier also nur unwesentlich größer als der des optimalen Kreises, die Vermutung ist aber trotzdem falsch. □

Beispiel: In einer[11] (x, y)-Ebene gehe ein Lichtstrahl vom Punkt $(0, a_1)$ zum Punkt $(b, -a_2)$ mit $a_1, a_2 > 0$, was in Abbildung 4.12 verdeutlicht werde. In den Halbebenen $y > 0$, $y < 0$ sei jeweils ein konstantes Medium vorhanden, in welchem die Lichtgeschwindigkeit v_1 bzw. v_2 betrage. Der Lichtstrahl beschreibe einen gebrochen geradlinigen Weg, und zwar unter dem Winkel β_1 in der oberen und β_2 in der unteren Halbebene gegenüber der „lotechten“ Richtung (parallel zur y-Achse). Sind die Längen der Lichtwege in den beiden Halbebenen s_1 bzw. s_2, so beträgt die Lichtzeit

$$Q = \frac{s_1}{v_1} + \frac{s_2}{v_2} = \frac{a_1}{v_1 \cos\beta_1} + \frac{a_2}{v_2 \cos\beta_2}.$$

Damit genügen die Variablen β_j bzw. $x_j = \tan\beta_j$ der Nebenbedingung

$$a_1 x_1 + a_2 x_2 = b,$$

[11] Dieses Beispiel haben wir wörtlich

L. Collatz, W. Wetterling (1971) *Optimierungsaufgaben*. Springer-Verlag, Berlin-Heidelberg-New York

entnommen.

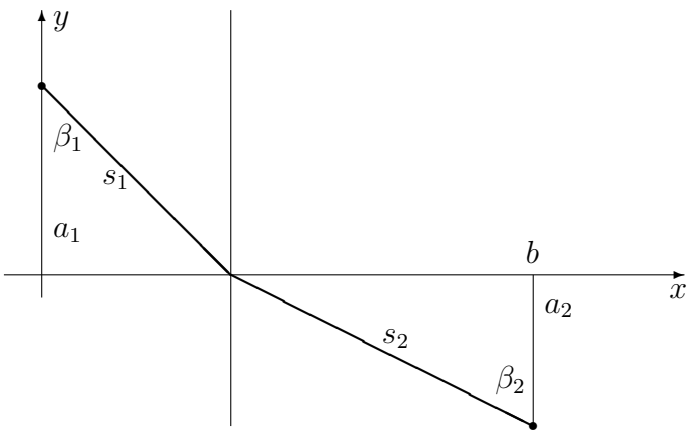

Abbildung 4.12: Kürzeste Lichtzeit

während die Zielfunktion in den Variablen x_1, x_2 die Form

$$Q = \frac{a_1}{v_1}(1 + x_1^2)^{1/2} + \frac{a_2}{v_2}(1 + x_2^2)^{1/2}$$

annimmt. Bei vorgegebenen positiven a_1, a_2, v_1, v_2 und b hat man also die Optimierungsaufgabe

$$\left\{ \begin{array}{l} \text{Minimiere} \quad f(x) := \dfrac{a_1}{v_1}(1 + x_1^2)^{1/2} + \dfrac{a_2}{v_2}(1 + x_2^2)^{1/2} \quad \text{unter der Nebenbedingung} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad a_1 x_1 + a_2 x_2 = b. \end{array} \right.$$

Da die Nebenbedingung (affin) linear ist, kann der Satz von Kuhn-Tucker angewandt werden ohne dass geprüft wird, ob die Zusatzbedingung erfüllt ist. Hiernach existiert zu einer Lösung (x_1^*, x_2^*) eine reelle Zahl v^* mit

$$(*) \qquad \begin{pmatrix} \dfrac{a_1}{v_1} \dfrac{x_1^*}{(1 + (x_1^*)^2)^{1/2}} \\ \dfrac{a_2}{v_2} \dfrac{x_2^*}{(1 + (x_2^*)^2)^{1/2}} \end{pmatrix} + v^* \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = 0.$$

Hieraus folgt

$$\frac{1}{v_1} \frac{x_1^*}{(1 + (x_1^*)^2)^{1/2}} = \frac{1}{v_2} \frac{x_2^*}{(1 + (x_2^*)^2)^{1/2}}.$$

Mit $x_j^* = \tan \beta_j^*$ (Brechungswinkel) erhält man das Brechungsgesetz

$$\frac{\sin \beta_1^*}{\sin \beta_2^*} = \frac{v_1}{v_2}.$$

Z. B. erhält man für $(a_1, a_2) = (1, 3)$, $(v_1, v_2) = (2, 1)$ und $b = 5$ aus den beiden Gleichungen

$$0.5 \frac{x_1}{(1 + x_1^2)^{1/2}} = \frac{x_2}{(1 + x_2^2)^{1/2}}, \qquad x_1 + 3x_2 = 5$$

mittels

```
> fsolve({0.5*x1/sqrt(1+x1^2)=x2/sqrt(1+x2^2),x1+3*x2=5},{x1,x2});
```

$$\{x1 = 3.361881094,\ x2 = .5460396354\}$$

das Ergebnis

$$(x_1^*, x_2^*) = (3.361881094, .5460396354),$$

danach können $\beta_j^* := \arctan(x_j^*)$ und $s_j^* := a_j/\cos(\beta_j^*)$, $j = 1, 2$, berechnet werden. □

4.2.7 Aufgaben

1. Man beweise den dritten Teil des starken Dualitätssatzes 2.4, also: Gegeben seien die lineare Optimierungsaufgabe

   (P) $$\text{Minimiere} \quad c^T x \quad \text{auf} \quad M := \{x \in \mathbb{R}^n : x \geq 0,\ Ax = b\}$$

   und die hierzu duale lineare Optimierungsaufgabe

   (D) $$\text{Maximiere} \quad b^T y \quad \text{auf} \quad N := \{y \in \mathbb{R}^m : A^T y \leq c\}.$$

   Man zeige: Ist $M = \emptyset$ und $N \neq \emptyset$, so ist $\sup_{y \in N} b^T y = +\infty$, die Zielfunktion von (D) ist also auf der Menge N der zulässigen Lösungen von (D) nicht nach oben beschränkt.

2. Die[12] Spieler P und D haben je 3 Karten auf der Hand, und zwar P die Karten Pik As, Karo As und Pik Zwei, D die Karten Pik As, Karo As und Karo Zwei. Beide Spieler legen jeweils zugleich eine ihrer Karten auf den Tisch. D gewinnt, wenn die hingelegten Karten die gleiche Farbe haben, andernfalls P. Ein As hat den Wert 1, eine Zwei den Wert 2. Die Höhe des Gewinnes ist gleich dem Wert derjenigen Karte, die der Gewinner hingelegt hat. Das Spiel hat also die Auszahlungsmatrix

D \ P	♢	♠	♠♠
♢	1	−1	−2
♠	−1	1	1
♢♢	2	−1	−2

   Man hat den Eindruck, das Spiel sei unfair, weil die Auszahlungsmatrix 5 negative Elemente gegenüber 4 positiven enthält. Das gibt Anlass zur Formulierung der

   Zusatzregel: Wenn beide Spieler ihre Zweierkarte hinlegen, so soll keiner an den anderen etwas zahlen, d. h. das Element -2 in der rechten unteren Ecke der Auszahlungsmatrix wird durch 0 ersetzt.

   Man berechne für das Spiel ohne und mit Zusatzregel mit Hilfe von Maple jeweils optimale gemischte Strategien für P und D und entscheide damit, welches der beiden Spiele fair ist.

[12] Siehe

L. Collatz, W. Wetterling (1971) *Optimierungsaufgaben*. Springer-Verlag, Berlin-Heidelberg-New York.

3. Seien α, β und γ Winkel in einem (spitzwinkligen) Dreieck mit $90^\circ \geq \alpha \geq \beta \geq \gamma \geq 0$ und natürlich $\alpha+\beta+\gamma = 180^\circ$. Unter allen solchen Winkeln bestimme man diejenigen, für die

$$g(\alpha,\beta,\gamma) := \min\{\gamma, \beta-\gamma, \alpha-\beta, 90^\circ-\alpha\}$$

maximal ist. Hierzu formuliere man diese Aufgabe als ein lineares Programm und löse es mit Hilfe eines mathematischen Anwendersystems.

Hinweis: Die obige Aufgabenstellung steht im engen Zusammenhang mit einem (sehr witzigen) Aufsatz von B. Tergan (1980)[13], in dem gezeigt wird, dass es (bis auf Ähnlichkeit) genau ein allgemeines, spitzwinkliges Dreieck gibt, dessen Winkel durch $\alpha^* := 75^\circ$, $\beta^* := 60^\circ$ und $\gamma^* := 45^\circ$ gegeben sind.

4. In einer Molkerei[14] werden zwei Sorten Käse hergestellt, etwa Gouda und Edamer. Die Fabrik hat Verträge, bis zu bestimmten Daten eine gewisse Menge (gemessen in einer bestimmten Einheit) von Käse mindestens herzustellen, nämlich

Zeitpunkt	Gouda	Edamer
30. Juni	5 000	3 000
31. Juli	6 000	3 000
31. August	4 000	5 000

Zur Produktion stehen zwei Typen von Maschinen zur Verfügung. Die Anzahl der zur Verfügung stehenden Produktionsstunden für die beiden Maschinen während der Sommermonate sind:

Monat	Maschine A	Maschine B
Juni	700	1 500
Juli	300	400
August	1 000	300

Die Produktionsraten (Stunden pro Mengeneinheit Käse) auf den beiden Typen von Maschinen sind

Typ	Maschine A	Maschine B
Gouda	0.15	0.16
Edamer	0.12	0.14

Unabhängig von den benutzten Typen und dem produzierten Käse kostet eine Arbeitsstunde 100 Euro. Das Material für eine Mengeneinheit Gouda kostet 52.50 Euro, das für Edamer 41.50 Euro. Pro Mengeneinheit Käse kommen noch 4 Euro hinzu. Überschüssiger Käse kann in den nächsten Monat (also von Juni in den Juli und von Juli

[13] Siehe den Anhang 2 bei

F. Wille (1982) *Humor in der Mathematik*. Vandenhoek & Ruprecht, Göttingen.

[14] Die Aufgabe ist im wesentlichen

M. Asghar Bhatti (2000) *Practical Optimization Methods. With Mathematica Applications*. Springer-Verlag, New York-Berlin-Heidelberg

entnommen. Dort handelt es sich allerdings um eine Reifenfabrik (statt einer Molkerei), in der Sommer- und Winterreifen produziert werden. Da die Produktion eines Bruchteils eines Reifens keinen Sinn macht, handelt es sich bei dem dort geschilderten Problem aber um eine *ganzzahlige* lineare Optimierungsaufgabe. Um dies zu vermeiden (es kommen nämlich nicht ganzzahlige Werte heraus) haben wir die Aufgabenstellung ein wenig verändert. Inwiefern diese Aufgabenstellung sinnvoll ist, sei dahin gestellt. Es kommt letzten Endes darauf an, das mathematische Modell aufzustellen.

in den August) übernommen werden, die Lagerkosten sind 1.50 Euro pro Mengeneinheit Käse. Eine Mengeneinheit des produzierten Käses wird für 200 Euro (Gouda) bzw. 150 Euro (Edamer) verkauft. Wie sollte die Produktion organisiert werden, um einerseits den Lieferbedingungen nachzukommen und andererseits den Gewinn der Molkerei zu maximieren?

Hinweis: Als Variable führen man ein:

x_1	Menge des im Juni auf Machine A produzierten Gouda
x_2	Menge des im Juli auf Maschine A produzierten Gouda
x_3	Menge des im August auf Maschine A produzierten Gouda
x_4	Menge des im Juni auf Machine A produzierten Edamer
x_5	Menge des im Juli auf Machine A produzierten Edamer
x_6	Menge des im August auf Machine A produzierten Edamer
x_7	Menge des im Juni auf Machine B produzierten Gouda
x_8	Menge des im Juli auf Maschine B produzierten Gouda
x_9	Menge des im August auf Maschine B produzierten Gouda
x_{10}	Menge des im Juni auf Machine B produzierten Edamer
x_{11}	Menge des im Juli auf Machine B produzierten Edamer
x_{12}	Menge des im August auf Machine B produzierten Edamer

5. Man beweise den ersten Teil des Max-Flow Min-Cut Theorems von Ford-Fulkerson, also: Gegeben sei ein Digraph $(\mathcal{N}, \mathcal{A})$, in dem zwei Knoten s (Quelle) und t (Senke) ausgezeichnet sind. Auf den Pfeilen seien nichtnegative Kapazitäten gegeben. Ist dann $x = (x_{ij})_{(i,j)\in\mathcal{A}}$ ein zulässiger Fluss mit dem Wert $v = \sum_{j:(s,j)\in\mathcal{A}} x_{sj}$ und ist $(\mathcal{N}_1, \mathcal{N}_2)$ ein Schnitt mit Kapazität $C(\mathcal{N}_1, \mathcal{N}_2)$, so ist $v \leq C(\mathcal{N}_1, \mathcal{N}_2)$.

6. Eine Gruppe von 11 Personen trifft sich in San Francisco. Möglichst viele von ihnen sollen nach New York geschickt werden. Es gibt keine Direktflüge, sondern es muss umgestiegen werden, wobei der Anschluss jeweils gesichert ist. In der folgenden Tabelle sind diese Flüge und die jeweils noch vorhandenen freien Sitze aufgelistet.

Von	Nach	Zahl freier Sitze
San Francisco	Denver	5
San Francisco	Houston	6
Denver	Atlanta	4
Denver	Chicago	2
Houston	Atlanta	5
Atlanta	New York	7
Chicago	New York	4

   (a) Man formuliere die Aufgabe als Maximaler-Fluss-Problem in einem geeigneten Digraphen.

   (b) Man rate einen maximalen Fluss und beweise seine Optimalität mit dem Max-Flow Min-Cut Theorem.

7. Für die Aufgabe

$$\text{(P)} \qquad \begin{cases} \text{Minimiere} \quad f(x) := x_1^2 + 4x_2^2 + 16x_3^2 \quad \text{unter der Nebenbedingung} \\ \qquad\qquad\qquad\qquad h(x) := x_1x_2x_3 - 1 = 0 \end{cases}$$

bestimme man mit Hilfe von Maple alle zulässigen Punkte, in denen die notwendigen Optimalitätsbedingungen erster Ordnung erfüllt sind.

8. Bei einer *ganzzahligen linearen Optimierungsaufgabe* handelt es sich um eine lineare Optimierungsaufgabe, bei der die Variablen ganzzahlig sind.

   In der x_1-x_2-Ebene veranschauliche man sich die folgende ganzzahlige lineare Optimierungsaufgabe

$$\left\{ \begin{array}{l} \text{Minimiere} \quad \begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \text{unter den Nebenbedingungen} \\ \begin{pmatrix} 5 & 7 \\ -2 & 1 \\ 1 & -5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 45 \\ 1 \\ 5 \end{pmatrix}, \quad \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad x_1, x_2 \in \mathbb{Z} \end{array} \right.$$

   und gebe die Lösung an. Zum Vergleich bestimme man die Lösung des *relaxierten* Problems, also des Problems, bei dem die Ganzzahligkeitsforderung gestrichen wird.