5. Übung Algorithmentechnik

Lehrstuhl für Algorithmik I
Institut für Theoretische Informatik
Universität Karlsruhe (TH)

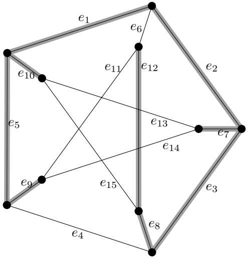
15.01.2008

- ICPC-Praktikum
- Auswertung Befragung
- Fragestunde
  - Letzte Übung am Donnerstag, den 12.02.09
  - Fragen per Mail an Reinhard (rbauer@ira.uka.de)

Aufgabe 1

Aufgabe (Algorithmus von De Pina)

(a) Führen Sie den Algorithmus von De Pina (algebraische Version) auf dem folgenden Graphen aus.



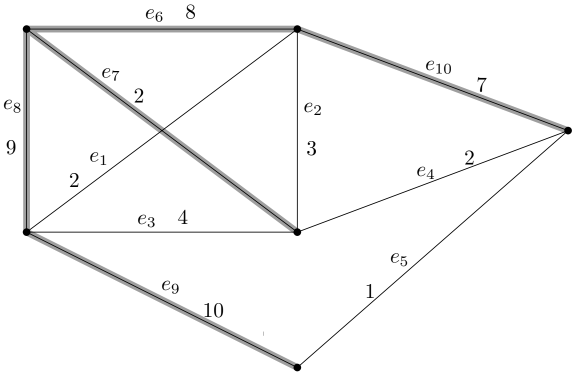
Algorithmus von De Pina

- Algorithmus zur Berechnung einer Kreisbasis minimalen Gewichts
- betrachte Spannbaum (spannenden Wald) T
- seien $E' := \{e_1, \ldots, e_N\}$ Nichtbaumkanten
- sei C_i Fundamentalkreis zu $e_i \in E'$
- jeder Kreis läßt sich als Vektor über $\{0,1\}^N$ darstellen
- Rekonstruktion des vollständigen Kreisvektors aus Vektor $v' \in \{0,1\}^N$:

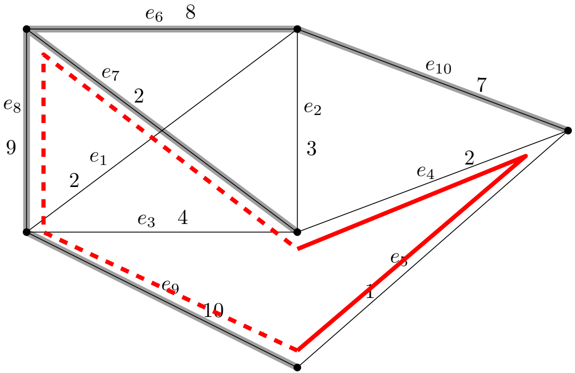
$$v = \bigoplus_{i=1}^{N} v_i C_i$$

- Beispiel: $v' = (1,0,1,1) \rightsquigarrow v = C_1 \oplus C_3 \oplus C_4$

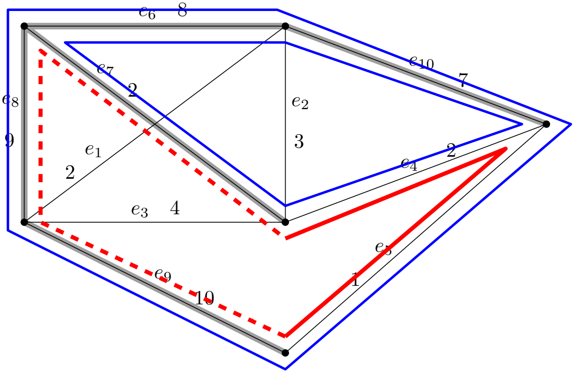
Beispiel



Beispiel



Beispiel



Algorithmus von De Pina

- ≫ seien $X, Y, Z \in \{0,1\}^N$
- ≫ betrachte Bilinearform

$$\langle X, Y \rangle := \sum_{i=1}^{N} X_i \cdot Y_i \quad \text{über } GF(2)$$

- $\Rightarrow$ $\langle X \oplus Y, Z \rangle = \langle X, Z \rangle + \langle Y, Z \rangle$
- ≫ $\langle X, Y \rangle = 0$ orthogonal
- ≫ $\langle X, Y \rangle = 1$ X und Y haben ungerade Anzahl Kanten gemeinsam

Algorithmus von De Pina

iterative Berechnung

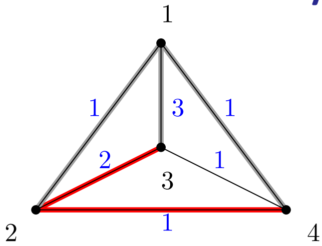
- sei $C_1, \ldots, C_i$ partielle Basis
- berechne S (Zeuge) orthogonal zu $C_1, \ldots, C_i$, d.h. $\langle S, C_j \rangle = 0 \quad j = 1, \ldots i$
- berechne Kreis C minimalen Gewichts mit $\langle S, C \rangle = 1$, d.h. C hat ungerade Anzahl von Kanten aus S

$\Rightarrow$ C linear unabhängig von $C_1, \ldots, C_i$

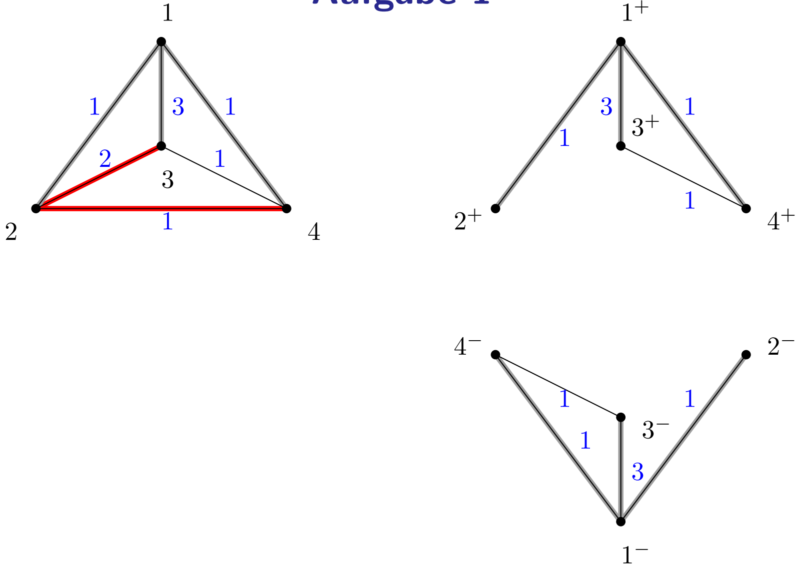
Kürzesten Kreis mit ungerader Anzahl Kanten aus S berechnen

- lässt sich auf Kürzeste-Wege-Suche in geeignetem Graph reduzieren, siehe Beispiel

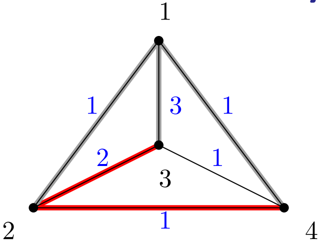
Aufgabe 1

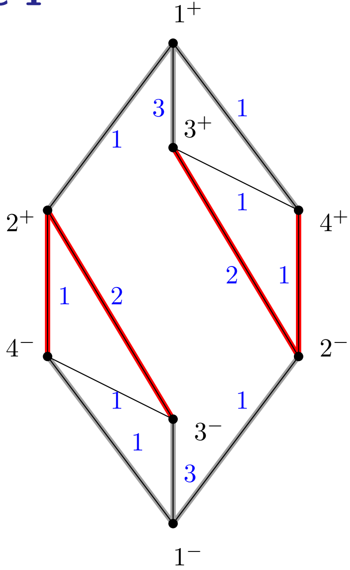


Aufgabe 1

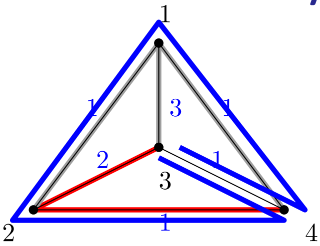


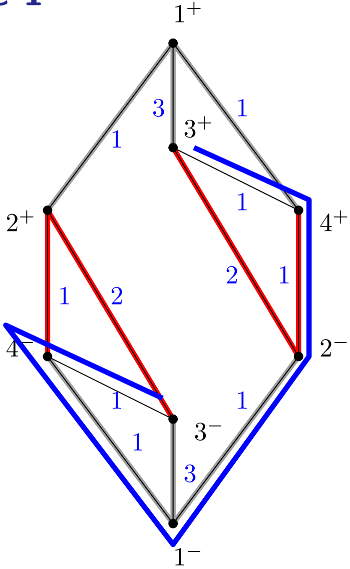
Aufgabe 1



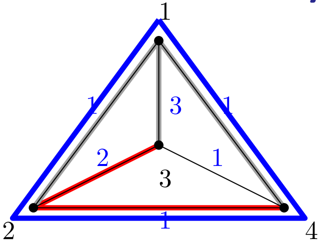


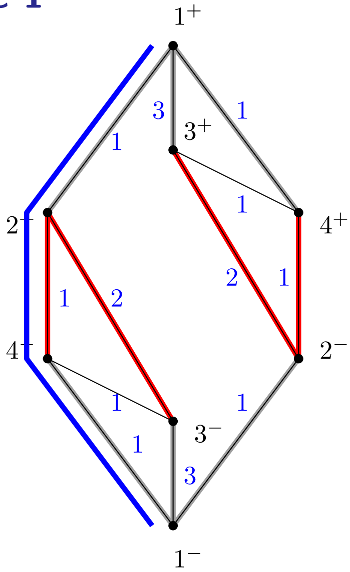
Aufgabe 1





Aufgabe 1



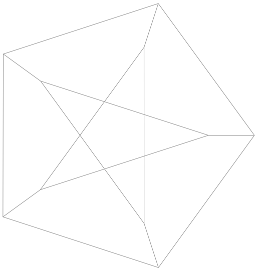


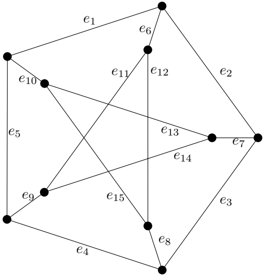
Algorithmus 1 : Algorithmus von De Pina algebraisch

Eingabe : Graph $G = (V, E)$
Ausgabe : MCB von G

für $i = 1$ *bis* N **tue**
  $S_i \leftarrow \{e_i\}$
für $k = 1$ *bis* N **tue**
  Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$
  für $i = k + 1$ *bis* N **tue**
    wenn $\langle C_k, S_i \rangle = 1$ **dann**
      $S_i \leftarrow S_i \oplus S_k$
Ausgabe ist: $\{C_1, \ldots, C_N\}$

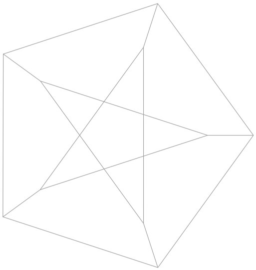
Aufgabe 1

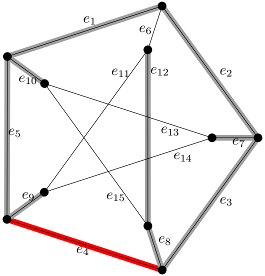




≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

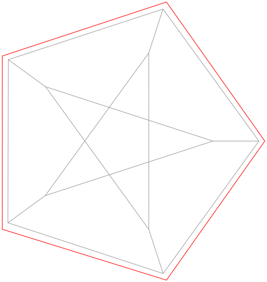


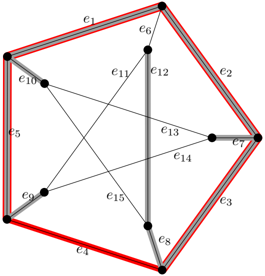


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

- Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

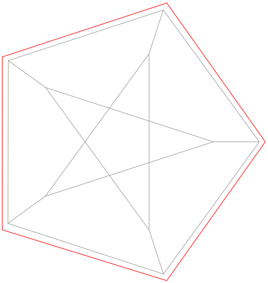


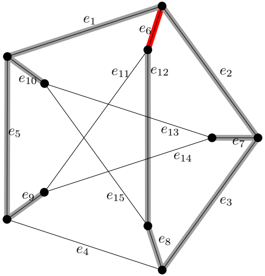


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

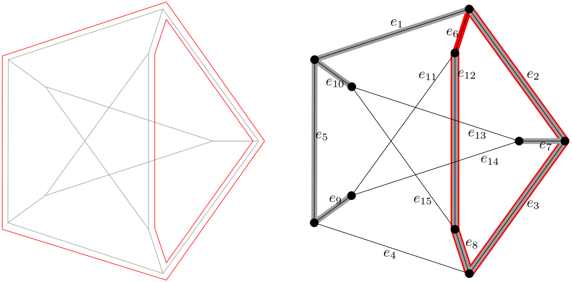




$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

- Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1



$S_1 = \{e_4\}$

$S_2 = \{e_6\}$

$S_3 = \{e_{11}\}$

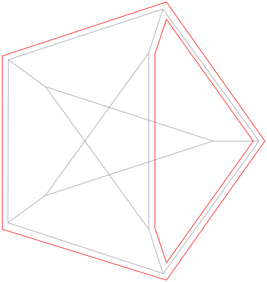
$S_4 = \{e_{13}\}$

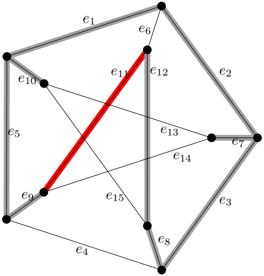
$S_5 = \{e_{14}\}$

$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

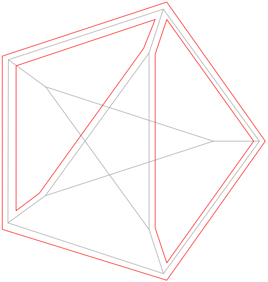


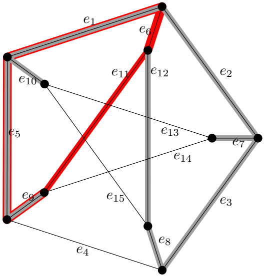


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

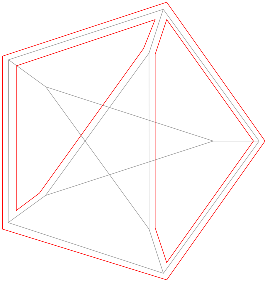


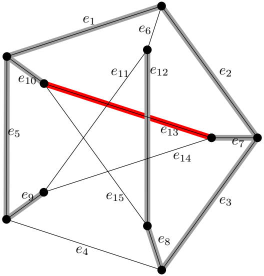


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

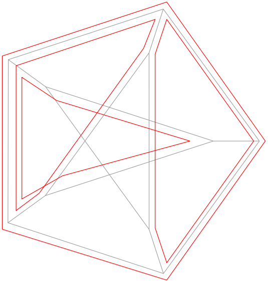


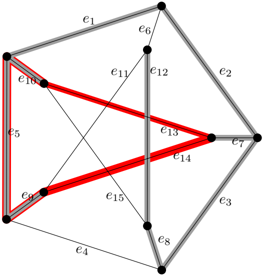


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$
$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1





$S_1 = \{e_4\}$

$S_2 = \{e_6\}$

$S_3 = \{e_{11}\}$

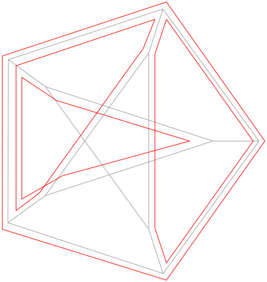
$\rightarrow S_4 = \{e_{13}\}$

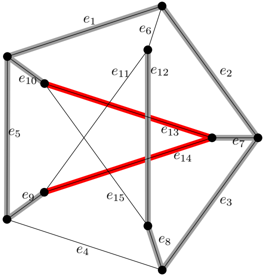
$\boxed{S_5 = \{e_{14}\}} \oplus S_4 = \{e_{13}, e_{14}\}$

$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

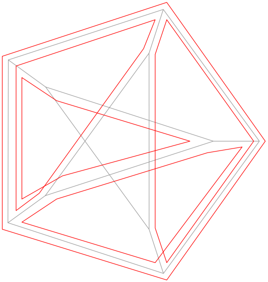


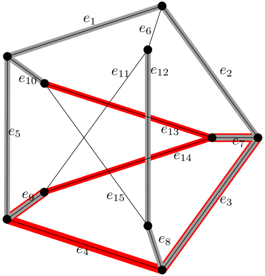


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
→ $S_5 = \{e_{14}\}$ $\oplus S_4 = \{e_{13}, e_{14}\}$
$S_6 = \{e_{15}\}$

- Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

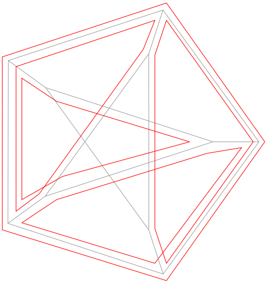


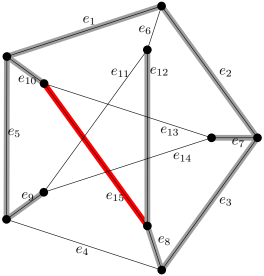


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
→ $S_5 = \{e_{14}\}$ $\oplus S_4 = \{e_{13}, e_{14}\}$
$S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

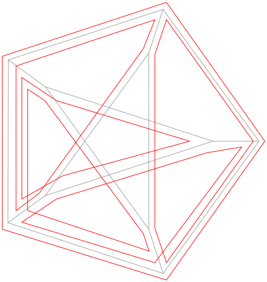


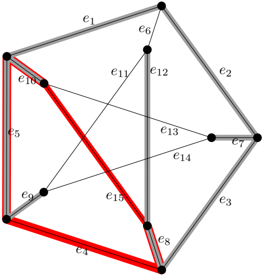


$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$ $\oplus S_4 = \{e_{13}, e_{14}\}$
→ $S_6 = \{e_{15}\}$

≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

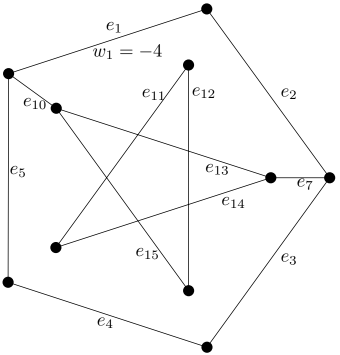




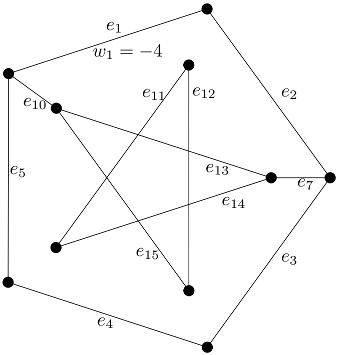
$S_1 = \{e_4\}$
$S_2 = \{e_6\}$
$S_3 = \{e_{11}\}$
$S_4 = \{e_{13}\}$
$S_5 = \{e_{14}\}$ $\oplus S_4 = \{e_{13}, e_{14}\}$
→ $S_6 = \{e_{15}\}$

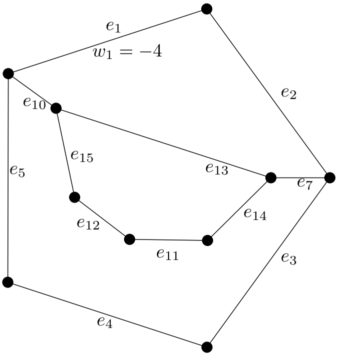
≫ Jeder Kreis hat mindestens 5 Kanten.

Aufgabe 1

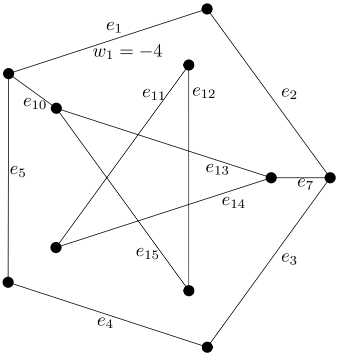


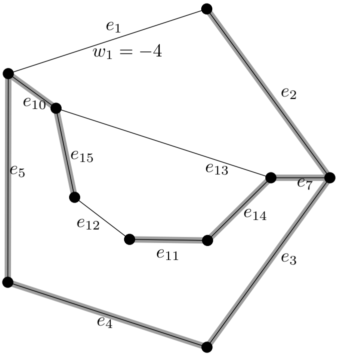
Aufgabe 1


e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}

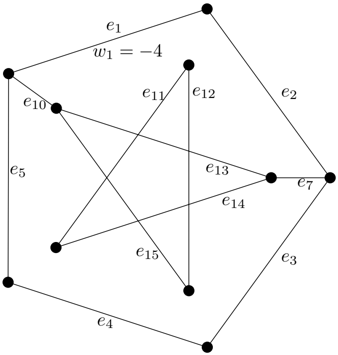

e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}

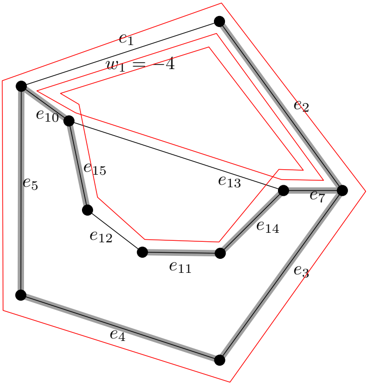
Aufgabe 1


e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}


e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}

Aufgabe 1


e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}


e_1
$w_1 = -4$
e_2
e_3
e_4
e_5
e_7
e_{10}
e_{11}
e_{12}
e_{13}
e_{14}
e_{15}

Funktioniert De Pina auch mit Negativen Gewichten?

Wo wird Nicht-Negativität ausgenutzt?

- Suche nach kürzestem Kreis mit ungerader Anzahl von Kanten aus S_i wird auf Kürzeste-Wege-Suche reduziert
- funktioniert nur, wenn es keine Kreise mit negativem Gewicht gibt

Kurze Wiederholung: Lineare Programmierung

- Standardform für Maximierungsprobleme
- Standardform für Minimierungsprobleme
- Slackform
- Überführung in Standardform
- Dualität
- Komplexität
- Simplex (Very Tiny Nutshell)

Standardform für (I)LPs (Maximierungsproblem)

- $A = (a_{ij}) \in \mathbb{R}^{m \times n},\ x = (x_i), b = (b_i) \in \mathbb{R}^m,\ c = (c_i) \in \mathbb{R}^n$
- Zielfunktion: $c^T x$

$$\text{maximiere } \sum_{i=1}^{n} c_i \cdot x_i = c^T x$$

- Nebenbedingungen: $Ax \leq b$

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad (i = 1, \dots, m)$$

- Nicht-Negativitätsbedingungen: $x \geq 0$

$$x_i \geq 0 \qquad (i = 1, \dots n)$$

Standardform für (I)LPs (Minimierungsproblem)

- $A = (a_{ij}) \in \mathbb{R}^{m \times n},\ x = (x_i), b = (b_i) \in \mathbb{R}^m,\ c = (c_i) \in \mathbb{R}^n$
- Zielfunktion: $c^T x$

$$\text{minimiere } \sum_{i=1}^{n} c_i \cdot x_i = c^T x$$

- Nebenbedingungen: $Ax \geq b$

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \qquad (i = 1, \ldots, m)$$

- Nicht-Negativitätsbedingungen: $x \geq 0$

$$x_i \geq 0 \qquad (i = 1, \ldots n)$$

Slack-Form für (I)LPs (Maximierungsproblem)

- $A = (a_{ij}) \in \mathbb{R}^{m \times n},\ x = (x_i), b = (b_i) \in \mathbb{R}^m,\ c = (c_i) \in \mathbb{R}^n$
- Zielfunktion: $c^T x$

$$\text{maximiere } \sum_{i=1}^{n} c_i \cdot x_i = c^T x$$

- Nebenbedingungen: $Ax = b$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \qquad (i = 1, \ldots, m)$$

- Nicht-Negativitätsbedingungen: $x \geq 0$

$$x_i \geq 0 \qquad (i = 1, \ldots n)$$

Standardformen (Übersicht)

$$\textit{Maximierung}: \max c^T x,\ Ax \leq b,\ x \geq 0$$

$$\textit{Minimierung}: \min c^T x,\ Ax \geq b,\ x \geq 0$$

$$\textit{Slackform}: \max c^T x,\ Ax = b,\ x \geq 0$$

Standardform für (I)LPs

Überführung in Standardform (Maximierungsproblem)

- Minimierungsproblem: miniere $c^T x \rightsquigarrow$ maximiere $-c^T x$
- Variablen ohne Nicht-Negativitätsbedingung $\rightsquigarrow$ ersetze jedes Vorkommen von x durch $x' - x''$ für zwei Variablen $x' \geq 0$ und $x'' \geq 0$
- Nebenbedingung der Form $\sum_{j=1}^{n} a_{ij} x_j \geq b_i \rightsquigarrow -\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$
- Nebenbedingung mit Gleichheit $\sum_{j=1}^{n} a_{ij} x_j = b_i \rightsquigarrow$ neue Ungleichungen $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ und $-\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$

Standardform für (I)LPs

Überführung in Standardform (Maximierungsproblem)

- Minimierungsproblem: miniere $c^T x \rightsquigarrow$ maximiere $-c^T x$
- Variablen ohne Nicht-Negativitätsbedingung $\rightsquigarrow$ ersetze jedes Vorkommen von x durch $x' - x''$ für zwei Variablen $x' \geq 0$ und $x'' \geq 0$
- Nebenbedingung der Form $\sum_{j=1}^{n} a_{ij} x_j \geq b_i \rightsquigarrow -\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$
- Nebenbedingung mit Gleichheit $\sum_{j=1}^{n} a_{ij} x_j = b_i \rightsquigarrow$ neue Ungleichungen $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ und $-\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$

Standardform für (I)LPs

Überführung in Standardform (Maximierungsproblem)

- Minimierungsproblem: miniere $c^T x \rightsquigarrow$ maximiere $-c^T x$
- Variablen ohne Nicht-Negativitätsbedingung $\rightsquigarrow$ ersetze jedes Vorkommen von x durch $x' - x''$ für zwei Variablen $x' \geq 0$ und $x'' \geq 0$
- Nebenbedingung der Form $\sum_{j=1}^{n} a_{ij} x_j \geq b_i \rightsquigarrow -\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$
- Nebenbedingung mit Gleichheit $\sum_{j=1}^{n} a_{ij} x_j = b_i \rightsquigarrow$ neue Ungleichungen $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ und $-\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$

Standardform für (I)LPs

Überführung in Standardform (Maximierungsproblem)

- Minimierungsproblem: miniere $c^T x \rightsquigarrow$ maximiere $-c^T x$
- Variablen ohne Nicht-Negativitätsbedingung $\rightsquigarrow$ ersetze jedes Vorkommen von x durch $x' - x''$ für zwei Variablen $x' \geq 0$ und $x'' \geq 0$
- Nebenbedingung der Form $\sum_{j=1}^{n} a_{ij} x_j \geq b_i \rightsquigarrow -\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$
- Nebenbedingung mit Gleichheit $\sum_{j=1}^{n} a_{ij} x_j = b_i \rightsquigarrow$ neue Ungleichungen $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ und $-\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$

Dualität

Primales LP: minimiere $\sum_{i=1}^{n} c_i x_i = c^T x$

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \qquad (i = 1, \ldots, m)$$

$$x_j \geq 0 \qquad (j = 1, \ldots, n)$$

Duales LP: maximiere $\sum_{i=1}^{n} y_i b_i = y^T b$

$$\sum_{i=1}^{m} y_i a_{ij} \leq c_j \qquad (j = 1, \ldots, n)$$

$$y_i \geq 0 \qquad (i = 1, \ldots, m)$$

Dualität

Schwacher Dualitäts-Satz
Sei P primales LP zu einem Minimierungsproblem mit dualem LP D. Sei x zulässige Lösung für P und y zulässige Lösung für D, d.h. $Ax \geq b$ ($x \geq 0$) und $y^T A \leq c^T$ ($y \geq 0$). Dann gilt

$$y^T b \leq c^T x$$

d.h. y liefert eine untere Schranke für den Optimalwert von P.

Anwendung: Primal-Duales Schema für Approximationsalgorithmen

Komplexität

LP

- Simplex-Algorithmus: kann exponentielle Laufzeit haben, trotzdem meist Algorithmus der Wahl
- Ellipsoid-Methode: polynomiell
- Innere-Punkte-Methode: polynomiell

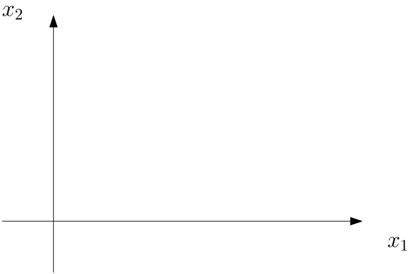
ILP

- $\mathcal{NP}$-schwere Variante

Simplex (Very Tiny Nutshell)

≫ Lineares Programm

$$\begin{aligned} \max\ c^T x \\ Ax \leq b \\ x \geq 0 \end{aligned}$$



≫ Eckentausch

≡ Gauß-Schritt in Matrix

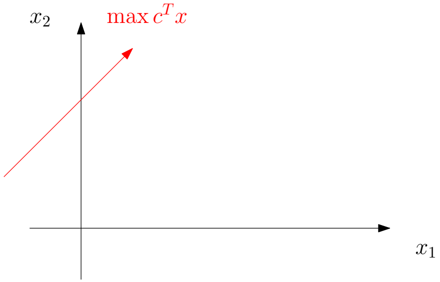
Simplex (Very Tiny Nutshell)

- Lineares Programm

$$\max c^T x$$
$$Ax \leq b$$
$$x \geq 0$$

- Eckentausch

$\equiv$ Gauß-Schritt in Matrix

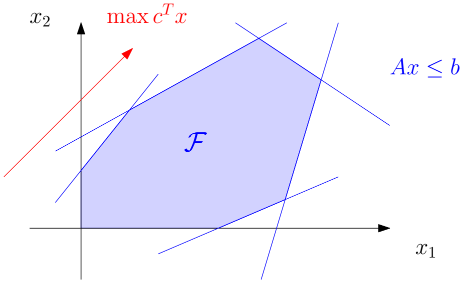


Simplex (Very Tiny Nutshell)

- Lineares Programm

$$\max c^T x$$
$$Ax \leq b$$
$$x \geq 0$$

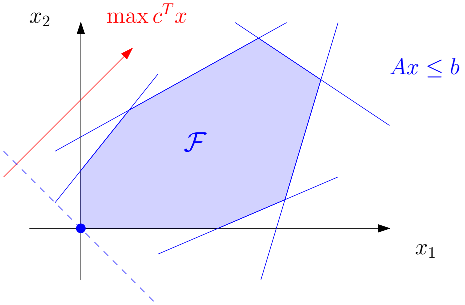
- Eckentausch
- ≡ Gauß-Schritt in Matrix



Simplex (Very Tiny Nutshell)

- Lineares Programm

$$\max c^T x$$
$$Ax \leq b$$
$$x \geq 0$$

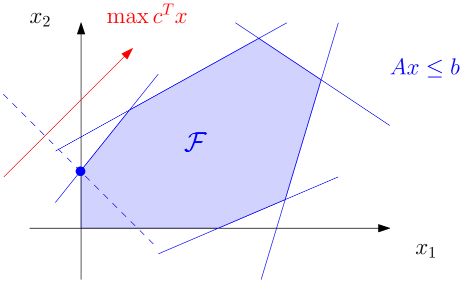


- Eckentausch
- Gauß-Schritt in Matrix

Simplex (Very Tiny Nutshell)

- ≫ Lineares Programm

$$\max c^T x$$
$$Ax \leq b$$
$$x \geq 0$$



- ≫ Eckentausch
- ≡ Gauß-Schritt in Matrix

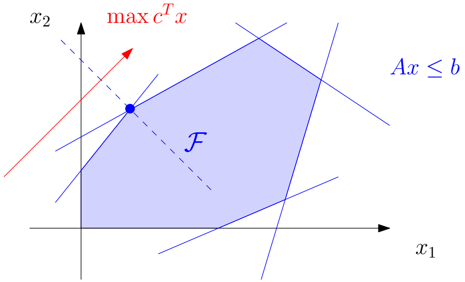
Simplex (Very Tiny Nutshell)

- Lineares Programm

$$\begin{aligned} \max\ c^T x \\ Ax \leq b \\ x \geq 0 \end{aligned}$$

- Eckentausch

$\equiv$ Gauß-Schritt in Matrix

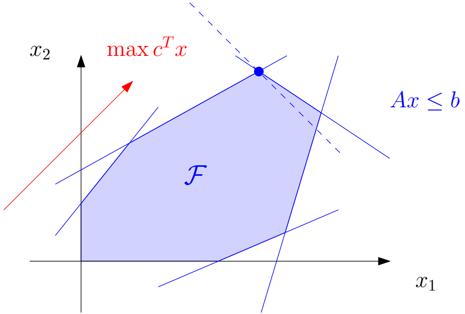


Simplex (Very Tiny Nutshell)

- ≫ Lineares Programm

$$\max c^T x$$
$$Ax \le b$$
$$x \ge 0$$

- ≫ Eckentausch
- ≡ Gauß-Schritt in Matrix



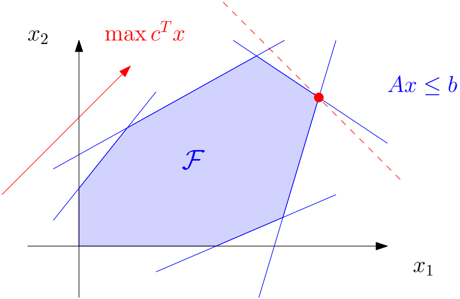
Simplex (Very Tiny Nutshell)

≫ Lineares Programm

$$\begin{aligned} \max\ c^T x \\ Ax \le b \\ x \ge 0 \end{aligned}$$

≫ Eckentausch

≡ Gauß-Schritt in Matrix



Aufgabe 3

Aufgabe (Euklidischer Handlungsreisender)

- Gegeben: n Punkte $P = \{p_1, \ldots, p_n\}$ mit Koordinaten (x_i, y_i) in der Ebene
- Gesucht: kürzeste Rundtour, die jeden Punkt mindestens einmal besucht
- Abstandsmaß ist euklidische Distanz

- Modellieren Sie dieses Probem als (ggf.) ganzzahliges lineares Programm.

Aufgabe 3

binäre Variable

$$x_{ij} \in \{0,1\} \qquad \forall i,j \in \{1,\ldots,n\}$$

Interpretation:

$x_{ij} = 1 \Leftrightarrow p_j$ ist Nachfolger von p_i

Zielfunktion:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij}$$

Aufgabe 3

jeder Punkt hat genau einen Nachfolger:

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \forall i \in \{1, \dots, n\}$$

jeder Punkt hat genau einen Vorgänger

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad \forall j \in \{1, \dots, n\}$$

keine reflexiven Kanten, denn diese sind wegen $d_{ii} = 0$ kostenlos

$$x_{ii} = 0 \qquad \forall i \in \{1, \dots, n\}$$

keine Subzyklen

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq 1 \qquad \forall S \subsetneq \{1, \dots, n\}, S \neq \emptyset$$

Aufgabe 3

Problem: "keine Subzyklen" erfordert in dieser Formulierung $2^n - 2$ Nebenbedingungen

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq 1 \qquad \forall S \subsetneq \{1, \ldots, n\}, S \neq \emptyset$$

Bessere Formulierung: Ganzzahlige Variable (Nummer des Punktes)

$$u_i \in \{1, \ldots, n-1\} \qquad \forall i \in \{1, \ldots, n-1\}$$
$$u_i - u_j + n x_{ij} \leq n - 1 \qquad \forall i, j \in \{1, \ldots, n-1\},\ i \neq j$$

Interpretation: Falls p_j Nachfolger von p_i ist ($x_{ij} = 1$), so ist $u_i < u_j$). Sonst ist Ungleichung trivialerweise erfüllt.

Aufgabe 4

Gegeben

- m Nährstoffe $1, \ldots, m$
- n Produkte $1, \ldots, n$
- b_i Bedarf an Nahrungsmittel i $(i = 1, \ldots, m)$
- c_j Kosten pro Einheit des Produktes j $(j = 1, \ldots, n)$
- a_{ij} Anteil des Nährstoffes i am Produkt j $(i = 1, \ldots, m,\ j = 1, \ldots, n)$

Gesucht

- x_i Menge des Produktes i $(i = 1, \ldots, n)$, so dass Gesamtkosten minimal sind

Primales/Duales Programm

Primales LP: minimiere

$$\sum_{i=1}^{n} c_i x_i = c^T x$$

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \qquad (i = 1, \ldots, m)$$

$$x_i \geq 0 \qquad (i = 1, \ldots, n)$$

Duales LP: maximiere

$$\sum_{i=1}^{m} y_i b_i = y^T b$$

$$\sum_{i=1}^{m} y_i a_{ij} \leq c_j \qquad (j = 1, \ldots, n)$$

$$y_i \geq 0 \qquad (i = 1, \ldots, m)$$

Interpretation Duales Programm

Duales LP: maximiere $\sum_{i=1}^{m} y_i b_i = y^T b$

$$\sum_{i=1}^{m} y_i a_{ij} \leq c_j \qquad (j = 1, \ldots, n)$$

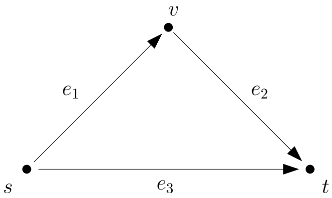
$$y_i \geq 0 \qquad (i = 1, \ldots, m)$$

- Idee: Nährstoffe sind "Zutaten" für Produkte
- a_{ij} beschreibt, wie viele Einheiten der Zutat i man für ein Produkt j benötigt
- c_j ist maximaler Preis des Produktes j
- y_i ist der Preis einer Zutat
- maximiert wird dann der Preis für den Tagesbedarf von Fogg

Aufgabe 2

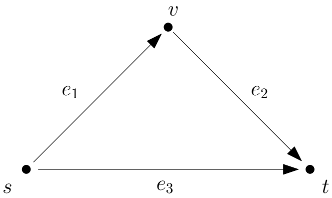
Aufgabe (Flussnetzwerke und Linear Programmierung)

Für das folgende Flussnetzwerk seien die Kantenkapazitäten $c(e_1) := 1$, $c(e_2) := 2$ und $c(e_3) := 3$ gegeben.



(a) Stellen Sie das Maximalflussproblem für dieses Netzwerk als lineares Programm dar und bringen Sie es anschließend in die in der Vorlesung definierte Standardform.

Lineares Programm



- maximiere $e_1 + e_3$
- Nebenbedingungen für Kapazität: $e_1 \leq 1, e_2 \leq 2, e_3 \leq 3$
- Nebenbedingungen für Flusserhaltung: $e_1 = e_2 \rightsquigarrow e_1 - e_2 = 0$ $\rightsquigarrow e_1 - e_2 \leq 0,\ e_1 - e_2 \geq 0 \rightsquigarrow e_1 - e_2 \leq 0,\ e_2 - e_1 \leq 0$
- Nicht-Negativitätsbedingungen: $e_1 \geq 0,\ e_2 \geq 0,\ e_3 \geq 0$

Matrix-Form

≫ maximiere $e_1 + e_3$ mit

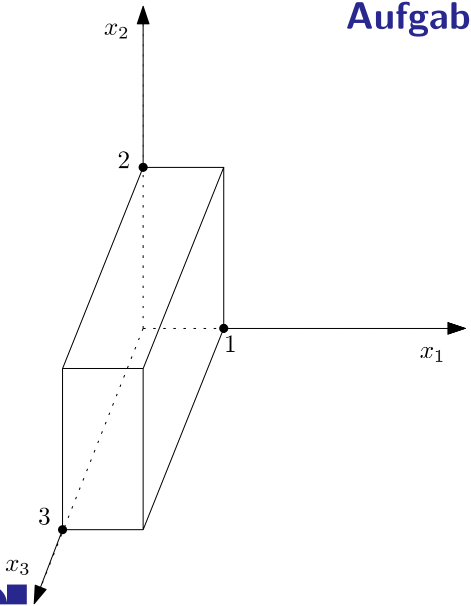
$$\begin{aligned} e_1 &\leq 1 \\ e_2 &\leq 2 \\ e_3 &\leq 3 \\ e_1 - e_2 &\leq 0 \\ -e_1 + e_2 &\leq 0 \end{aligned}$$

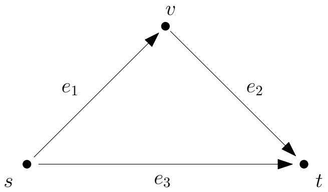
$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & -1 & 0 \\ -1 & 1 & 0 \end{pmatrix}, \quad c = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 0 \\ 0 \end{pmatrix}$$

Aufgabe 2

(b) Stellen sie das durch die Kapazitätsbedingungen gegebene konvexe Polyeder sowie die durch die Flusserhaltungsbedingungen gegebene Hyperebene graphisch dar. Ist das Lösungspolyeder beschränkt?

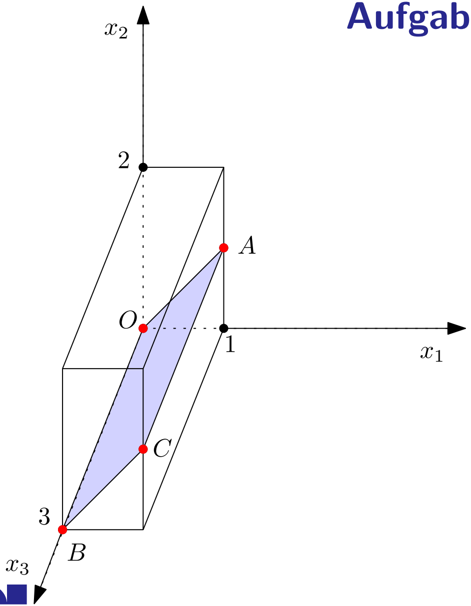
Aufgabe 2

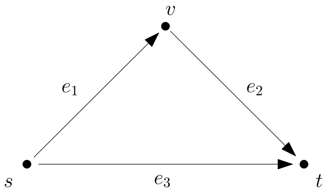




$$
\begin{aligned}
\max e_1 + e_3 \quad & \text{mit} \\
e_1 &\leq 1 \\
e_2 &\leq 2 \\
e_3 &\leq 3 \\
e_1 - e_2 &= 0
\end{aligned}
$$

Aufgabe 2





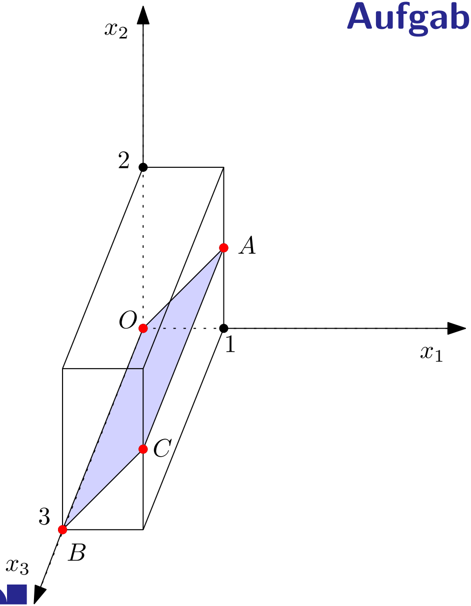
$$\max e_1 + e_3 \quad \text{mit}$$
$$e_1 \leq 1$$
$$e_2 \leq 2$$
$$e_3 \leq 3$$
$$e_1 - e_2 = 0$$

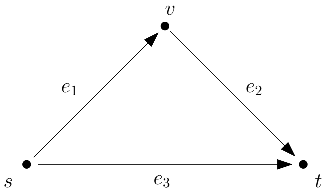
Aufgabe 2

(c) Führen Sie die Simplexmethode auf dem Polyeder durch: Starten Sie dazu im Nullpunkt.

- ≫ Welches sind die *verbessernden Kanten* von dort aus?
- ≫ Welchen Flusserhöhungen entsprechen diese im Ausgangsgraphen?
- ≫ Welches ist der Extrempunkt, der dem maximalen Fluss entspricht?

Aufgabe 2





$$
\begin{aligned}
\max e_1 + e_3 & \quad \text{mit} \\
e_1 &\le 1 \\
e_2 &\le 2 \\
e_3 &\le 3 \\
e_1 - e_2 &= 0
\end{aligned}
$$

Aufgabe 2

Betrachte allgemeines Flussproblem zu $G = (V, E)$.

$$\max \sum_{(s,i)\in E} x_{s,i} - \sum_{(i,s)\in E} x_{i,s} \qquad (\max\ a^T x)$$

unter den Nebenbedingungen

$$\begin{aligned} x_{i,j} &\leq c_{i,j} && (i,j) \in E && (Ix \leq c) \\ x_{i,j} &\geq 0 && (i,j) \in E && (-Ix \leq 0) \\ \sum_{j:(i,j)\in E} x_{i,j} - \sum_{j:(j,i)\in E} x_{j,i} &= 0 && i \in V \setminus \{s,t\} && (Bx = 0) \end{aligned}$$

(e) Welche Dimensionen haben a, I, B?

Aufgabe 2

$$\max \ a^T x$$

$$\begin{bmatrix} I \\ -I \\ B \\ -B \end{bmatrix} x \leq \begin{pmatrix} c \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

- $a \in \mathbb{R}^m$: Kapazität für jede Kante
- $I \in \mathbb{R}^{m \times m}$: Koeffizienten für m Kapazitätsbedingungen
- $B \in \mathbb{R}^{n-2 \times m}$: je eine Flusserhaltungsbedingung für die $n-2$ Knoten in $V \setminus \{s, t\}$

Aufgabe 2

(f) Zeigen oder widerlegen Sie: Falls G zusammenhängend ist, sind die Zeilen der Matrix B linear unabhängig.

Aufgabe 2

Struktur der Matrix zu den Nebenbedingungen

$$\sum_{j:(i,j)\in E} x_{i,j} - \sum_{j:(j,i)\in E} x_{j,i} = 0 \qquad i \in V \setminus \{s,t\} \qquad (Bx = 0)$$

- je eine Zeile für Knoten $i \neq s, t$
- je eine Spalte für eine (gerichtete) Kante
- sei $x = (x_i)$ Spaltenvektor zu (u, v), dann gilt

$$\Rightarrow x = (0, \ldots, 0, -1, 0, \ldots, 0, 1, 0, \ldots, 0)^T$$

mit $x_u = -1$, $x_v = 1$ und $x_i = 0$ für $i \neq u, v$

Aufgabe 2

Annahme: $\sum_{i=1}^{n-2} \alpha_i z_i = 0$ ist Linearkombination der Zeilen, G zusammenhängend

- sei $S \subset V \setminus \{s, t\}$ Teilmenge der Knoten i mit $\alpha_i \neq 0$
- sei $(w, t) \in E$ dann gilt $\alpha_w = 0$: betrachte Spalte zu (w, t) diese enthält genau zwei Einträge ungleich Null, nämlich w und $t \Rightarrow \alpha_t \neq 0$, aber es gibt keine Zeile zu t
- G zusammenhängend $\Rightarrow$ es existiert eine Kante (u, v) mit $u \in S$ und $v \notin S$
- Spalte zu (u, v) entält nur zwei Einträge ungleich Null, nämlich u und v und es muss gelten

$$\alpha_u + \alpha_v = 0$$

$\Rightarrow$ Widerspruch zu $\alpha_u \neq 0$ und $\alpha_v = 0$

Aufgabe 2

(g) Welche Dimension hat das durch die Nebenbedingungen definierte Lösungspolyeder im Allgemeinen?

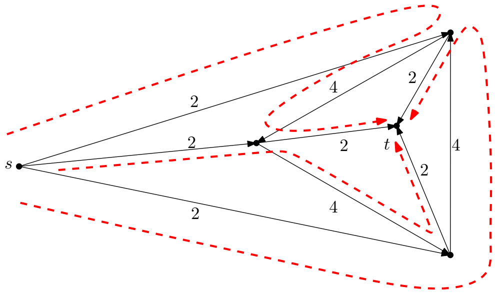
- ≫ Kapazitätsbedingungen definieren Hyperquader (siehe Beispiel)
- ≫ falls alle $c_{ij} > 0$, so hat Hyperquader Dimension m
- ≫ Zeile von B entspricht Hyperebene in $\mathbb{R}^m$ mit nichtleerem Schnitt mit Hyperquader und anderen Hyperebenen (enthält Ursprung)
- ≫ B definiert $n-2$ linear unabhängige Hyperebenen
- ⇒ Dimension des Lösungspolyeders ist i.A. $m-(n-2)$ (jede Hyperebene reduziert die Dimension um genau 1)

Aufgabe 2

(h) Zeigen oder widerlegen Sie: Die Erhöhung des Flusses entlang eines erhöhenden Weges entspricht einem Schritt im Simplexverfahren.

Hinweis: Ein Extrempunkt eines Polygons P kann nicht als Konvexkombination $\lambda \cdot s + (1 - \lambda) \cdot t$ zweier verschiedener Punkte von P dargestellt werden. Ein Punkt auf einer Kante von P kann nur als konvexe Kombination zweier verschiedener Punkte der selben Kante dargestellt werden.

Aufgabe 2



≫ Erhöhung entlang roter Wege führt zu Fluss

$$(2,\ldots,2) = 1/2(2,2,2,4,4,4,2,2,2) + 1/2(2,2,2,0,0,0,2,2,2)$$

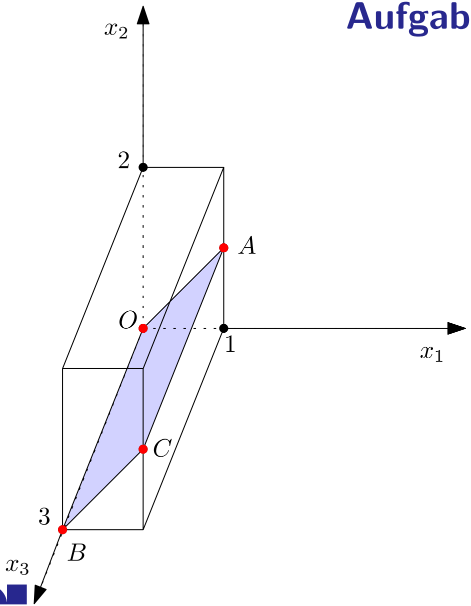
⇒ $(2,2,2,2,2,2,2,2,2)$ kein Extrempunkt

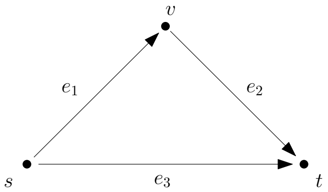
⇒ mind einer der erhöhenden Wege ist kein Simplex-Schritt

Aufgabe 2

(i) Zeigen oder widerlegen Sie die Umkehrung des vorherigen Satzes, d.h. jeder Schritt im Simplex-Verfahren entspricht einer Erhöhung entlang erhöhender Wege.

Aufgabe 2

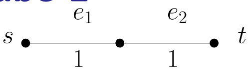




$$\begin{aligned} \max\ e_1 + e_3 &\quad \text{mit} \\ e_1 &\le 1 \\ e_2 &\le 2 \\ e_3 &\le 3 \\ e_1 - e_2 &= 0 \end{aligned}$$

Aufgabe 2

Betrachte einfachen Pfad:



$$
\begin{aligned}
&\max e_1 \\
e_1 &\leq 1 \\
e_2 &\leq 1 \\
e_1 - e_2 &\leq 0 \\
e_2 - e_1 &\leq 0
\end{aligned}
\qquad\qquad
\begin{aligned}
&\max e_1 \\
z_1 &= 1 - e_1 \\
z_2 &= 1 - e_2 \\
z_3 &= e_1 - e_2 \\
z_4 &= e_2 - e_1
\end{aligned}
$$

≫ Basislösung: $(0, 0, 1, 1, 0, 0)$

≫ Einziger möglicher Simplexschritt (Pivot) mit e_1

≫ e_1 kann wegen $z_4 = e_2 - e_1$ und $z_4 \geq 0$ nicht echt erhöht werden

$\Rightarrow$ Simplexschritt führt nicht zu Erhöhung der Zielfunktion