Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Approximationsalgorithmen

4. Vorlesung: Lineare Programmierung und LP-Dualität

– Folien von Joachim Spoerhase –

Alexander Wolff Lehrstuhl für Informatik I WS 2018/19

Einführung Lineare Programmierung

Ein Großteil bekannter Approximationsalgorithmen basiert auf linearer Programmierung.

- Lineare Programmierung (LP) und LP-Dualität
- Min-Max-Beziehungen
- Techniken LP-basierten Algorithmendesigns

Motivation: Obere und untere Schranken

- Betrachte schweres NP-Minimierungsproblem.
- Entscheidungsproblem:
  Ist gegebenes S **obere Schranke** für OPT?
- Leicht verifizierbare „Ja“-Zertifikate (nach Def. von NP)
- **Untere Schranken** / „Nein“-Zertifikate?
  $\rightsquigarrow$ wahrscheinlich nicht! (Vermutung: NP $\neq$ coNP)
- Brauchen untere Schranken $S \geq \mathrm{OPT}/\alpha$ (approximative „Nein“-Zertifikate) für Approximationsalgorithmen!
- Beispiele
  - Vertex Cover: Untere Schranke durch Matchings
  - TSP: Untere Schranke durch MST oder Cycle Cover

Lineare Programmierung

Optimiere (d.h. minimiere oder maximiere) eine lineare Funktion unter Anwesenheit linearer Ungleichungsbedingungen.

$$\begin{array}{lrcrcrcr}
\text{minimize} & 7x_1 & + & x_2 & + & 5x_3 & & \\
\text{subject to} & x_1 & - & x_2 & + & 3x_3 & \geq & 10 \\
 & 5x_1 & + & 2x_2 & - & x_3 & \geq & 6 \\
 & & & & & x_1, x_2, x_3 & \geq & 0
\end{array}$$

- Standardform (ÜA)

Lineare Programmierung – Obere Schranken

Optimiere (d.h. minimiere oder maximiere) eine lineare Funktion unter Anwesenheit linearer Ungleichungsbedingungen.

$$
\begin{array}{lrcrcrcr}
\text{minimize} & 7x_1 & + & x_2 & + & 5x_3 & & \\
\text{subject to} & x_1 & - & x_2 & + & 3x_3 & \geq & 10 \\
 & 5x_1 & + & 2x_2 & - & x_3 & \geq & 6 \\
 & & & & & x_1, x_2, x_3 & \geq & 0
\end{array}
$$

- $\mathbf{x} = (2, 1, 3)$ ist zulässige Lösung
  $\Rightarrow S = 30$ ist obere Schranke für OPT

Lineare Programmierung – Untere Schranken

Optimiere (d.h. minimiere oder maximiere) eine lineare Funktion unter Anwesenheit linearer Ungleichungsbedingungen.

$$
\begin{array}{lrcrcrcr}
\text{minimize} & 7x_1 & + & x_2 & + & 5x_3 & & \\
\text{subject to} & x_1 & - & x_2 & + & 3x_3 & \geq & 10 \\
 & 5x_1 & + & 2x_2 & - & x_3 & \geq & 6 \\
 & & & & & x_1, x_2, x_3 & \geq & 0
\end{array}
$$

- $7x_1 + x_2 + 5x_3 \geq x_1 - x_2 + 3x_3 \geq 10 \;\Rightarrow\; \text{OPT} \geq 10$
- $7x_1 + x_2 + 5x_3 \geq (x_1 - x_2 + 3x_3) + (5x_1 + 2x_2 - x_3) \geq 10 + 6$ $\Rightarrow \text{OPT} \geq 16$

Lineare Programmierung - Untere Schranken

$$
\begin{array}{lrcrcrcrl}
\text{minimize} & 7x_1 & + & x_2 & + & 5x_3 & & & \\
\text{subject to } {\color{red}y_1 (} & x_1 & - & x_2 & + & 3x_3 {\color{red})} & \geq & 10 & {\color{red}y_1} \\
{\color{red}y_2 (} & 5x_1 & + & 2x_2 & - & x_3 {\color{red})} & \geq & 6 & {\color{red}y_2} \\
 & & & & x_1, & x_2, x_3 & \geq & 0 &
\end{array}
$$

$$
\begin{array}{lrcrcl}
{\color{red}\text{maximize}} & {\color{red}10y_1} & {\color{red}+} & {\color{red}6y_2} & & \\
\text{subject to} & y_1 & + & 5y_2 & \leq & 7 \\
 & -y_1 & + & 2y_2 & \leq & 1 \\
 & 3y_1 & - & y_2 & \leq & 5 \\
 & & & y_1, y_2 & \geq & 0
\end{array}
$$

- Jede zulässige Lösung für das **duale** Programm liefert untere Schranke für Optimum des **primalen** Programms.
- $\mathbf{x} = (7/4, 0, 11/4)$ und $\mathbf{y} = (2, 1)$ liefern beide den jeweiligen Zielfunktionswert 26. $\Rightarrow$ Beide Lösungen sind jeweils optimal.

LP – Standardform

Primales Programm:

$$
\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{j=1}^{n} c_j x_j & \\
\text{subject to} & \displaystyle\sum_{j=1}^{n} a_{ij} x_j \geq b_i & i = 1, \ldots, m \\
 & x_j \geq 0 & j = 1, \ldots, n
\end{array}
$$

Duales Programm:

$$
\begin{array}{lll}
\text{maximize} & \displaystyle\sum_{i=1}^{m} b_i y_i & \\
\text{subject to} & \displaystyle\sum_{i=1}^{m} a_{ij} y_i \leq c_j & j = 1, \ldots, n \\
 & y_i \geq 0 & i = 1, \ldots, m
\end{array}
$$

Duale Programme für Max-Probleme

- Durch analoge Überlegungen – Beispiel später.
- Das duales Programm eines dualen Programms ergibt das Ausgangsprogramm.

LP-Dualität

Primal: minimize $\sum_{j=1}^{n} c_j x_j$
subject to $\sum_{j=1}^{n} a_{ij} x_j \geq b_i \quad i = 1, \ldots, m$
$x_j \geq 0 \quad j = 1, \ldots, n$

Dual: maximize $\sum_{i=1}^{m} b_i y_i$
subject to $\sum_{i=1}^{m} a_{ij} y_i \leq c_j \quad j = 1, \ldots, n$
$y_i \geq 0 \quad i = 1, \ldots, m$

Satz. Das primale Programm hat ein endliches Optimum.
$\Leftrightarrow$ Das duale Programm hat ein endliches Optimum.
Sind dann $\mathbf{x}^* = (x_1^*, \ldots, x_n^*)$ und $\mathbf{y}^* = (y_1^*, \ldots, y_m^*)$ optimale Lösungen für das primale bzw. duale Programm, so gilt

$$\sum_{j=1}^{n} c_j x_j^* = \sum_{i=1}^{m} b_i y_i^* \,.$$

Schwache LP-Dualität

Minimize $\sum_{j=1}^{n} c_j x_j$
subject to $\boxed{\sum_{j=1}^{n} a_{ij} x_j \geq b_i}$
$x_j \geq 0$

Maximize $\sum_{i=1}^{m} b_i y_i$
subject to $\sum_{i=1}^{m} a_{ij} y_i \leq c_j$
$\boxed{y_i \geq 0}$

Satz. Sind $\mathbf{x} = (x_1, \ldots, x_n)$ und $\mathbf{y} = (y_1, \ldots, y_m)$ zulässige Lösungen für das primale bzw. duale Programm, so gilt

$$\sum_{j=1}^{n} c_j x_j \geq \sum_{i=1}^{m} b_i y_i \,.$$

Beweis

$$\sum_{j=1}^{n} c_j x_j \geq \sum_{j=1}^{n} \left(\sum_{i=1}^{m} a_{ij} y_i \right) x_j = \sum_{i=1}^{m} \boxed{\left(\sum_{j=1}^{n} a_{ij} x_j \right)} y_i \geq \sum_{i=1}^{m} \boxed{b_i} y_i \,.$$

Komplementärer Schlupf (*complementary slackness*)

Minimize $\sum_{j=1}^{n} c_j x_j$
subject to $\boxed{\sum_{j=1}^{n} a_{ij} x_j \geq b_i}$
$\boxed{x_j \geq 0}$

Maximize $\sum_{i=1}^{m} b_i y_i$
subject to $\boxed{\sum_{i=1}^{m} a_{ij} y_i \leq c_j}$
$\boxed{y_i \geq 0}$

Satz. Seien $\mathbf{x} = (x_1, \ldots, x_n)$ und $\mathbf{y} = (y_1, \ldots, y_m)$ zulässige Lösungen für das primale bzw. duale Programm. Die Lösungen $\mathbf{x}$ und $\mathbf{y}$ sind genau dann optimal, wenn die folgenden Bedingungen erfüllt sind:

Primaler KS:
Für jedes $j = 1, \ldots, n$: Entweder $x_j = 0$ oder $\sum_{i=1}^{m} a_{ij} y_i = c_j$

Dualer KS:
Für jedes $i = 1, \ldots, m$: Entweder $y_i = 0$ oder $\sum_{j=1}^{n} a_{ij} x_j = b_i$

Beweis. Folgt aus LP-Dualität:

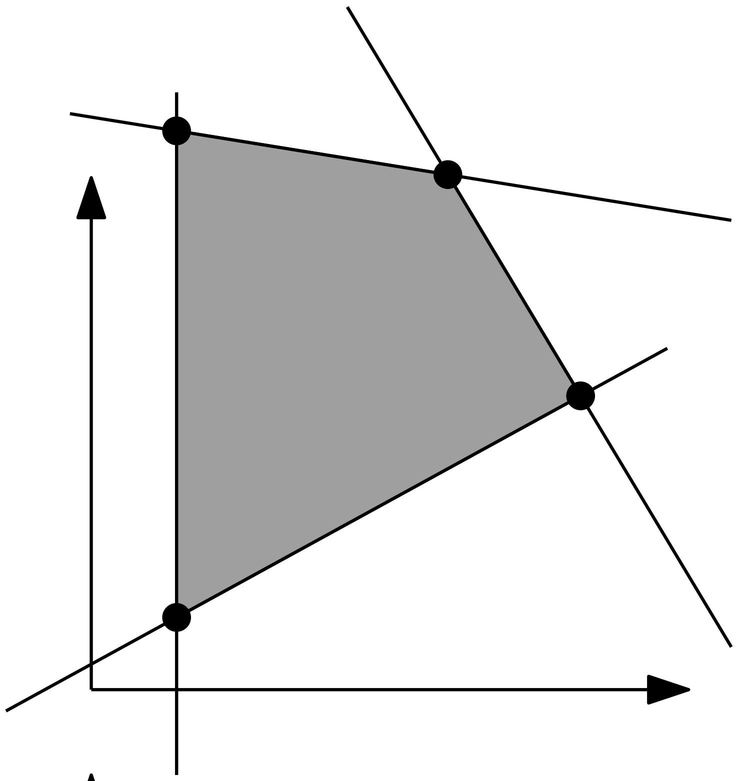
$$\sum_{j=1}^{n} \boxed{c_j x_j} = \sum_{j=1}^{n} \boxed{\left(\sum_{i=1}^{m} a_{ij} y_i \right) x_j} = \sum_{i=1}^{m} \boxed{\left(\sum_{j=1}^{n} a_{ij} x_j \right) y_i} = \sum_{i=1}^{m} \boxed{b_i y_i}.$$

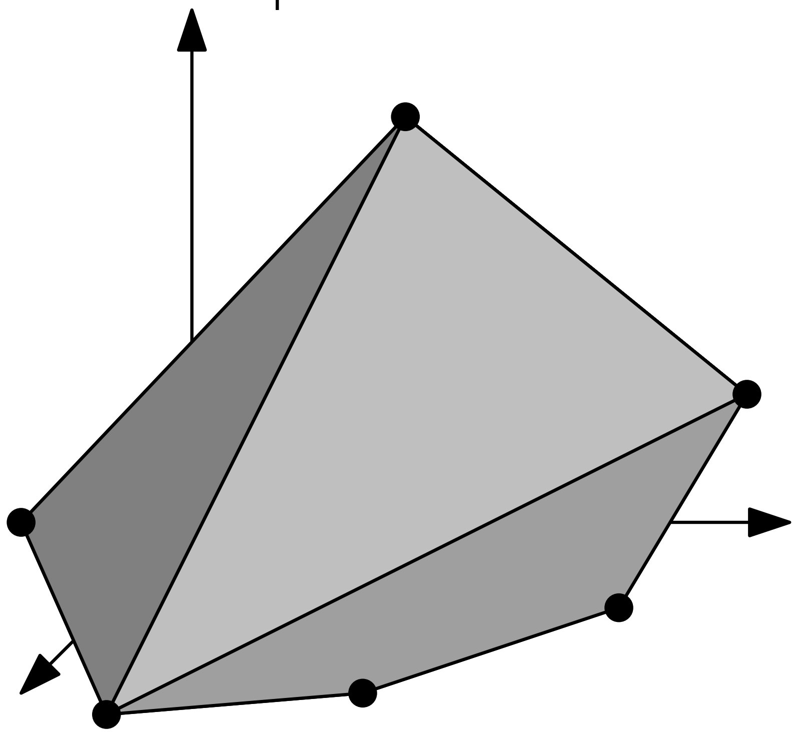
LP und konvexe Polytope

- Menge der zulässigen Lösungen eines LPs mit n Variablen bilden **konvexes Polytop** im $\mathbb{R}^n$ (Schnitt von Halbräumen).

- Ecken des Polytops heißen **Extrempunkt-Lösungen** $\Leftrightarrow$ n linear unabhängige Ungleichungen sind mit Gleichheit erfüllt.

- Wenn optimale Lösung existiert, dann auch eine solche, die Extrempunkt ist.





Ganzzahlige lineare Programme

$$\begin{aligned} \text{Minimize} \quad & \sum_{j=1}^{n} c_j x_j \\ \text{subject to} \quad & \sum_{j=1}^{n} a_{ij} x_j \geq b_i \quad i = 1, \ldots, m \\ & \cancel{x_j \geq 0} \;\; x_j \in \mathbb{N} \quad j = 1, \ldots, n \end{aligned}$$

- Eine Vielzahl von NP-Optimierungsproblemen lässt sich als ILP formulieren; Lösung ist daher NP-schwer.
- LP-Relaxierung liefert untere Schranke: $\text{OPT}_{\text{LP}} \leq \text{OPT}_{\text{ILP}}$

Einführung Lineare Programmierung

Ein Großteil bekannter Approximationsalgorithmen basiert auf linearer Programmierung.

- Lineare Programmierung (LP) und LP-Dualität
- Min-Max-Beziehungen
- Techniken LP-basierten Algorithmendesigns

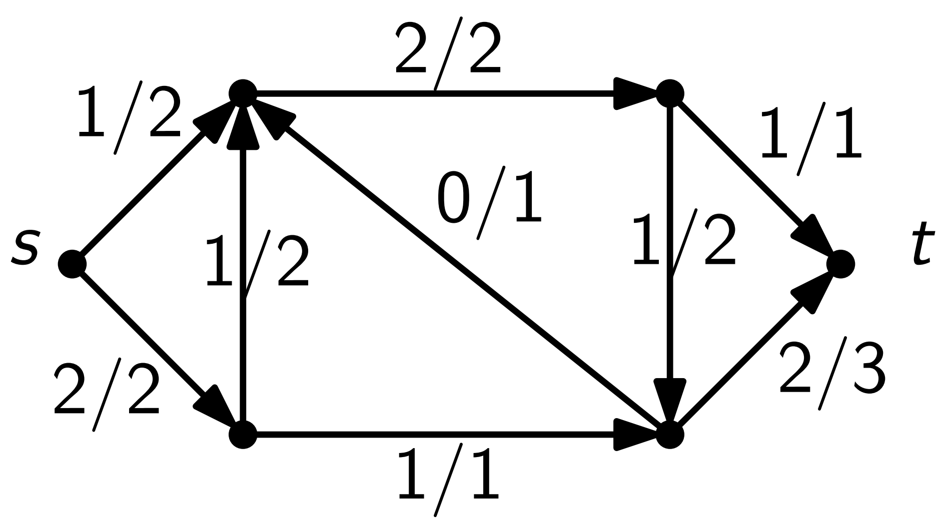
Max-Flow-Problem

Gegeben: Ein gerichteter Graph $G = (V, E)$ mit Kantenkapazitäten $c\colon E \to \mathbb{Q}_+$ und zwei ausgezeichnete Knoten Quelle s und Senke t.

Gesucht: Ein maximaler s–t-Fluss (nicht-negative Kantengewichte f), so dass

- $f(u, v) \leq c(u, v)$ für jede Kante $(u, v) \in E$
- $\sum_{(u,v)\in E} f(u, v) = \sum_{(v,z)\in E} f(v, z)$ für alle $v \in V - \{s, t\}$

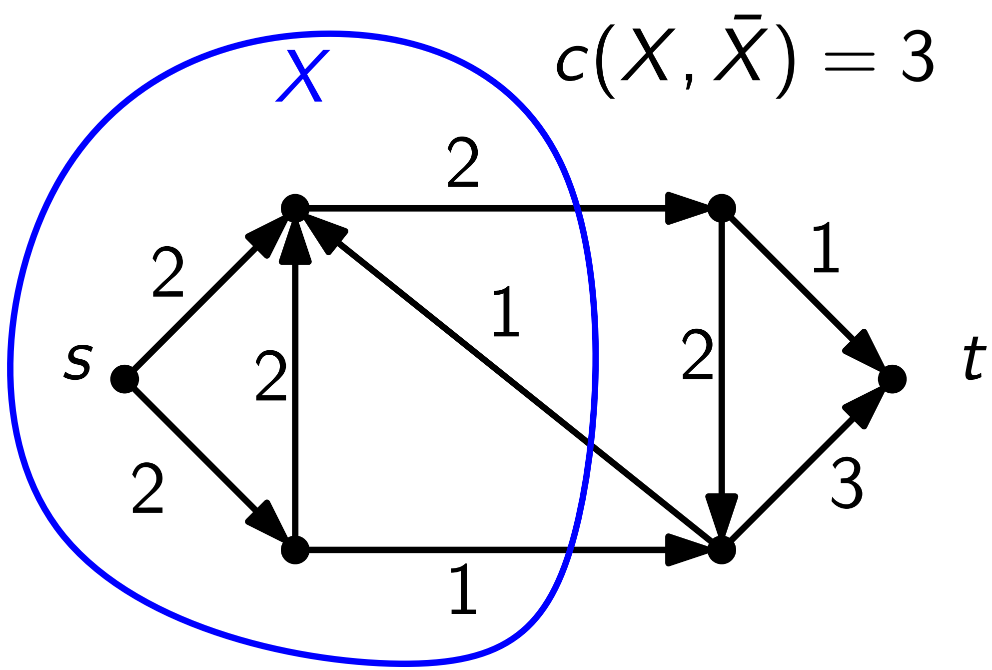
Wert des Flusses hierbei Zufluss zu t minus Abfluss aus t



Min-Cut-Problem

Gegeben: Ein gerichteter Graph $G = (V, E)$ mit Kantenkapazitäten $c \colon E \to \mathbb{Q}_+$ und zwei ausgezeichnete Knoten Quelle s und Senke t.

Gesucht: Ein s–t-Schnitt, d.h. eine Knotenmenge X mit $s \in X$ und $t \in \bar{X}$, so dass das Gesamtgewicht $c(X, \bar{X})$ der Kanten von X nach $\bar{X}$ minimal ist.



Max-Flow-Min-Cut-Theorem

Satz. Der Wert eines maximalen s–t-Flusses und die Kapazität eines minimalen s–t-Schnitts sind gleich.

Beweis. Spezialfall des LP-Dualitätssatzes. . .

Max-Flow als LP

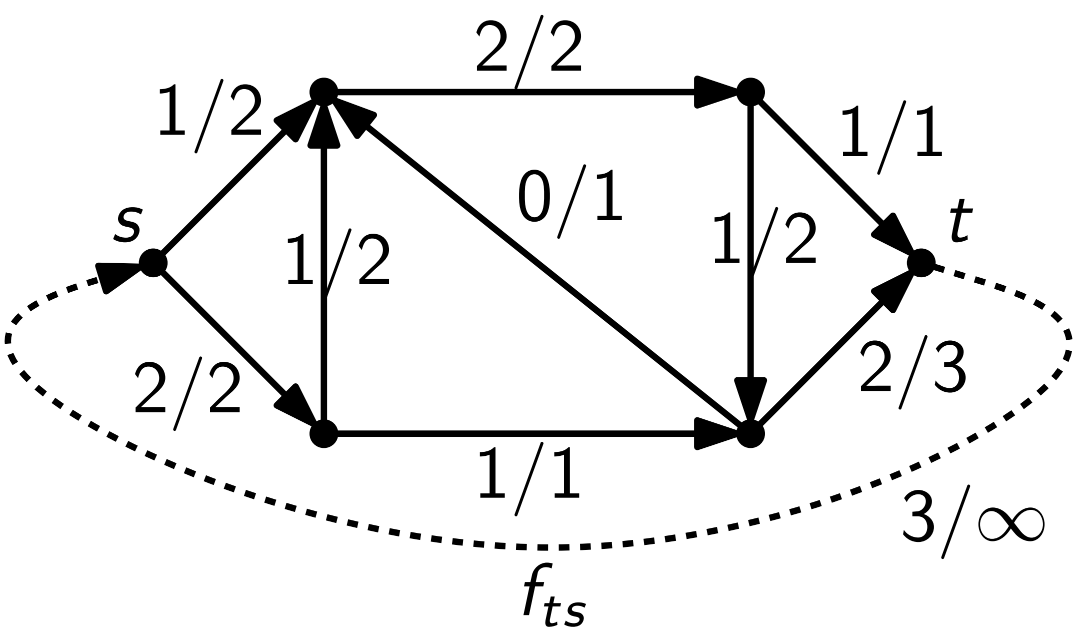
Maximize f_{ts}

subject to

$$f_{uv} \leq c_{uv} \qquad \text{for every } (u,v) \in E$$

$$\sum_{u\colon (u,v) \in E} f_{uv} - \sum_{z\colon (v,z) \in E} f_{vz} \leq 0 \qquad \text{for every } v \in V$$

$$f_{uv} \geq 0 \qquad \text{for every } (u,v) \in E$$



Duales LP

Primales Programm

$$
\begin{array}{llll}
\text{Maximize} & f_{ts} & & \\
\text{subject to} & f_{uv} \leq c_{uv} & \text{for every } (u,v) \in E & d_{uv} \\
& \displaystyle\sum_{u:\,(u,v)\in E} f_{uv} - \sum_{z:\,(v,z)\in E} f_{vz} \leq 0 & \text{for every } v \in V & p_v \\
& f_{uv} \geq 0 & \text{for every } (u,v) \in E &
\end{array}
$$

Duales Programm

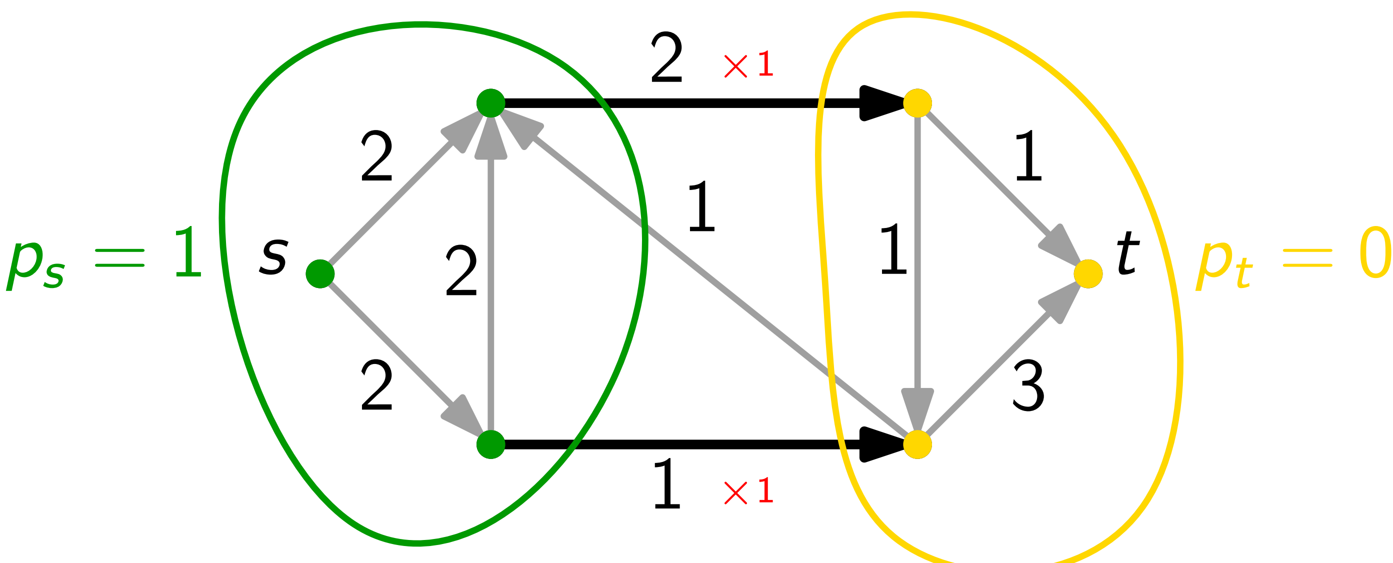
$$
\begin{array}{lll}
\text{Minimize} & \displaystyle\sum_{(u,v)\in E} c_{uv} d_{uv} & \\
\text{subject to} & d_{uv} - p_u + p_v \geq 0 & \text{for every } (u,v) \in E \\
& p_s - p_t \geq 1 & \\
& d_{uv} \geq 0 & \text{for every } (u,v) \in E \\
& p_u \geq 0 & \text{for every } u \in V
\end{array}
$$

Duales LP – Interpretation als ILP

$$\text{Minimize} \quad \sum_{(u,v)\in E} c_{uv} d_{uv}$$

$$\begin{aligned}
\text{subject to} \quad & d_{uv} - p_u + p_v \geq 0 && (u,v) \in E \\
& p_s - p_t \geq 1 \\
& \cancel{d_{uv} \geq 0}\ d_{uv} \in \{0,1\} && (u,v) \in E \\
& \cancel{p_u \geq 0}\ p_u \in \{0,1\} && u \in V
\end{aligned}$$

äquivalent zu Min-Cut!!



Duales LP – Fraktionale Schnitte

$\equiv$ LP-Relaxierung des ILPs

$$\text{Minimize} \quad \sum_{(u,v)\in E} c_{uv} d_{uv}$$

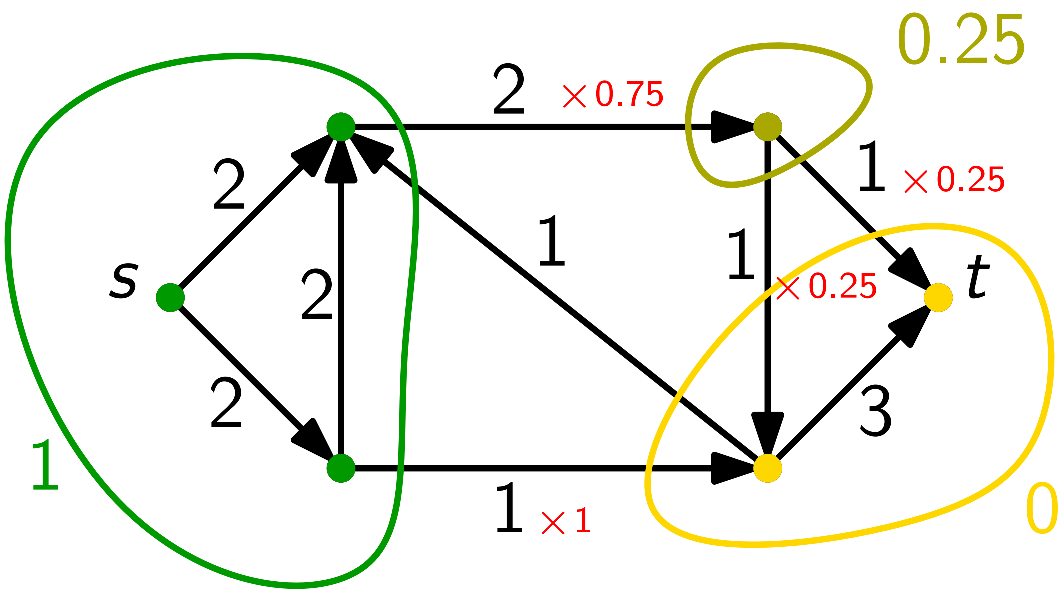
$$\begin{array}{lll} \text{subject to} & d_{uv} - p_u + p_v \geq 0 & (u,v) \in E \\ & p_s - p_t \geq 1 & \\ & d_{uv} \geq 0 & (u,v) \in E \\ & p_u \geq 0 & u \in V \end{array}$$

Jede Extrempunkt-Lösung ist hier **ganzzahlig**! (ÜA)

Jeder s–t-Weg $s = v_0, \ldots, v_k = t$ hat Länge ≥ 1 bezüglich d:

$$\sum_{i=0}^{k-1} d_{i,i+1} \geq \sum_{i=0}^{k-1} (p_i - p_{i+1})$$

$$= p_s - p_t$$



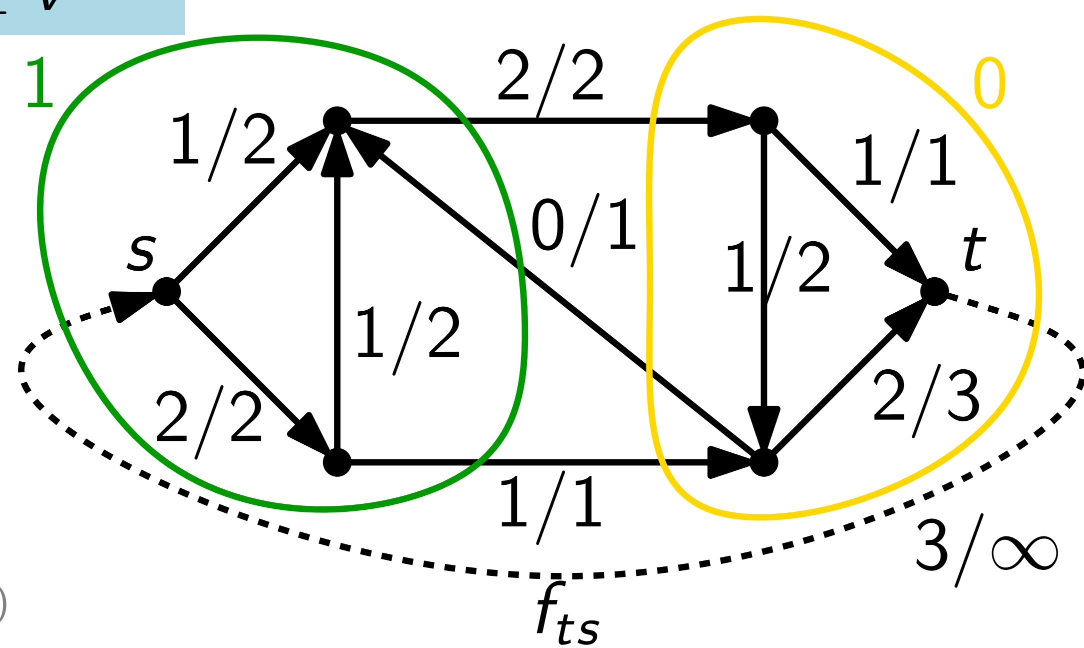
Duales LP – Komplementärer Schlupf

Maximize	f_{ts}	
subject to	$f_{uv} \leq c_{uv}$	$(u,v) \in E$
	$\sum_{u\colon (u,v)\in E} f_{uv} - \sum_{z\colon (v,z)\in E} f_{vz} \leq 0$	$v \in V$
	$f_{uv} \geq 0$	$(u,v) \in E$

Minimize	$\sum_{(u,v)\in E} c_{uv} d_{uv}$	
subject to	$d_{uv} - p_u + p_v \geq 0$	$(u,v) \in E$
	$p_s - p_t \geq 1$	
	$d_{uv} \geq 0$	$(u,v) \in E$
	$p_u \geq 0$	$u \in V$

Für einen maximalen Fluss und minimalen Schnitt gilt:

- Für Vorwärtskanten (u,v) des Schnitts gilt $f_{uv} = c_{uv}$.
  ($d_{uv} = 1$, also nach dualem KS: $f_{uv} = c_{uv}$.)
- Für Rückwärtskanten (u,v) des Schnitts gilt $f_{uv} = 0$.
  (Sonst wäre nach primalem KS: $d_{uv} - 0 + 1 = 0$.)

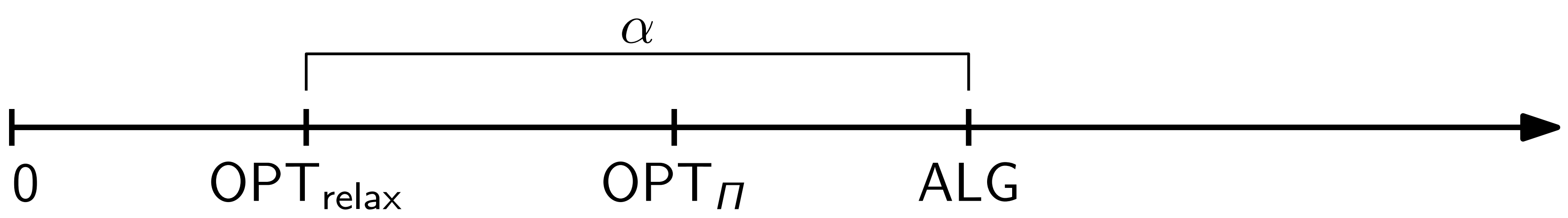


Einführung Lineare Programmierung

Ein Großteil bekannter Approximationsalgorithmen basiert auf linearer Programmierung.

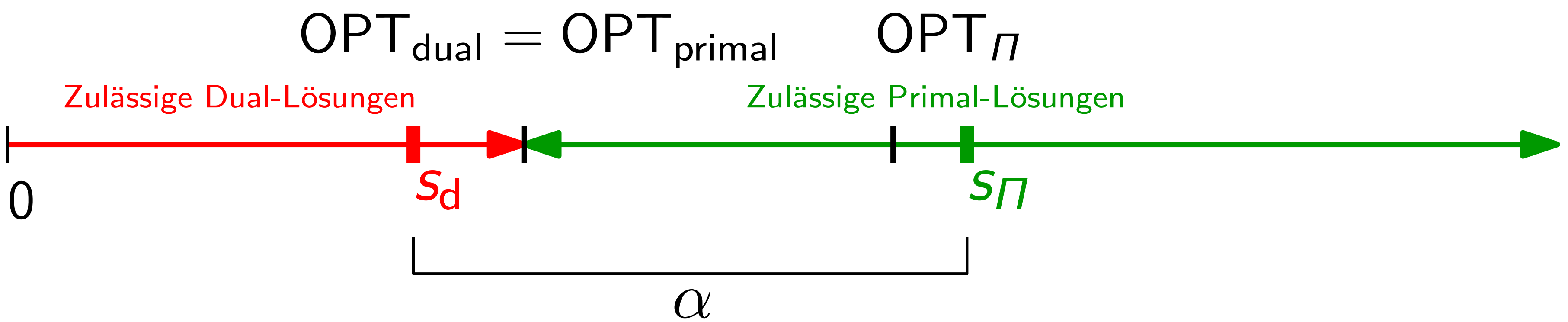
- Lineare Programmierung (LP) und LP-Dualität
- Min-Max-Beziehungen
- Techniken LP-basierten Algorithmendesigns

I) LP-Runden



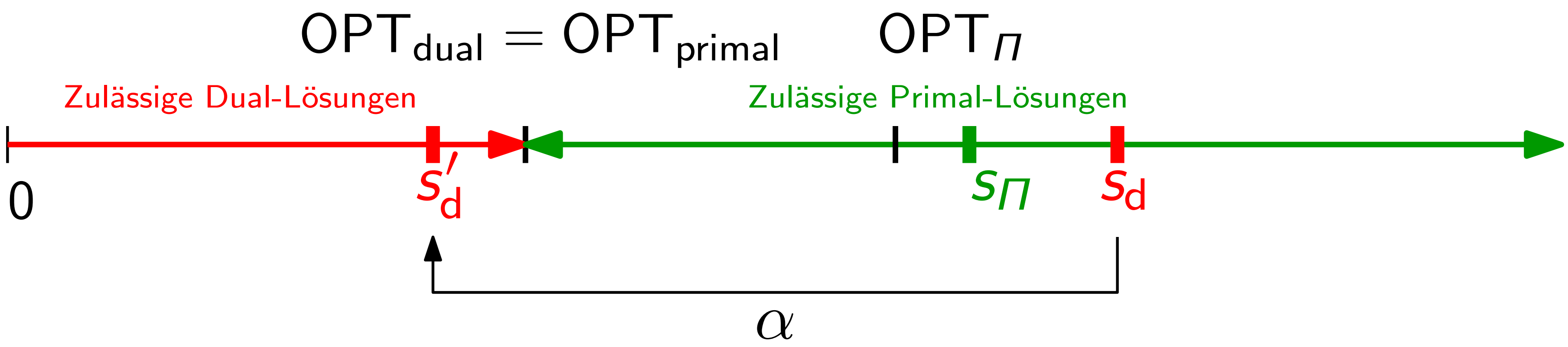
- Betrachte Minimierungsproblem Π in ILP-Form
- Berechne Lösung der LP-Relaxierung
- Runde zu ganzzahliger Lösung für Π
- Schwierigkeit: **Zulässige** Lösung für Π
- Approximationsgüte $\leq \mathrm{ALG}/\,\mathrm{OPT}_{\mathrm{relax}} \leq \alpha$

II) Primal-Dual-Ansatz



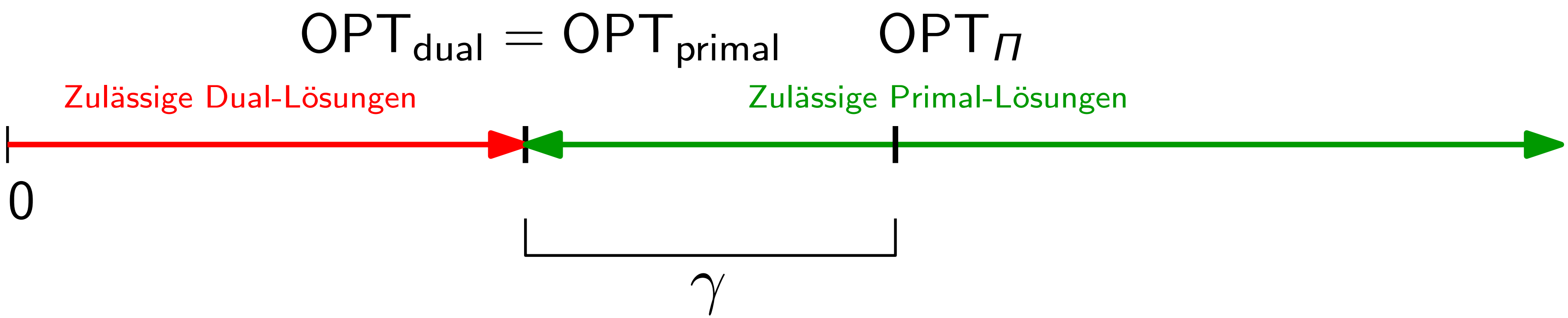
- Betrachte Minimierungsproblem Π in ILP-Form.
- Starte mit (trivialer) zulässiger Duallösung und unzulässiger Primallösung (z.B. alle Variable $= 0$).
- Berechne **duale** Lösung s_{d} und ganzzahlige Lösung s_{Π} für Π iterativ: erhöhe s_{d} entspr. KS und mache s_{Π} „ganzzahliger“.
- Approximationsgüte $\leq \mathrm{obj}(s_{\Pi})/\mathrm{obj}(s_{\mathrm{d}})$
- Vorteil:
  Kommt ohne LP-„Maschinerie“ aus; ggf. schneller, flexibler.

III) Dual Fitting



- Betrachte Minimierungsproblem Π in ILP-Form.
- Kombinatorischer Algorithmus berechnet zul. Lösung s_{Π} und „unzulässige“ Duallösung s_{d}, die s_{Π} „voll auszahlt“, d.h. $\text{obj}(s_{\Pi}) \leq \text{obj}(s_{\text{d}})$.
- Skaliere Dualvariablen, so dass neue Duallösung s'_{d} *zulässig*.
  $\Rightarrow \text{obj}(s_{\Pi})/\alpha = \text{obj}(s'_{\text{d}}) \leq \text{OPT}_{\text{dual}} \leq \text{OPT}_{\Pi}$
  $\Rightarrow$ Skalierungsfaktor α ist Approximationsfaktor.

Wichtig: Integrality Gap



- Betrachte Minimierungsproblem Π in ILP-Form.
- Ohne Ausnutzung weiterer Eigenschaften sind alle diese Methoden höchstens so gut wie das *Integrality Gap* der LP-Relaxierung:

$$\sup_I \frac{\mathrm{OPT}(I)}{\mathrm{OPT}_{\mathrm{primal}}(I)}$$