- 7.1 Einführung 42
- 7.2 Vollständig unimodulare Matrizen 43
- 7.3 Branch and Bound Algorithmen 44
- 7.4 Lagrange Relaxation 45
- 7.5 Schnittebenenverfahren 46
- 7.6 Optimierung und Separierung 47

7.1 Einführung

- Ganzzahlige lineare Programme (Integer Linear Program, ILP, IP) verlangen die Ganzzahligkeit der Variablen für ...
- In diesem Abschnitt:
  - Die Ganzzahligkeit erhöht die Modellierungskraft enorm, viele nichtlineare Effekte können so modelliert werden
  - Daher erhält man i.A. NP-schwere Probleme

- LP Relaxation eines IP
  - Standardform eines IP

    $\min\ c^T x$

    unter $Ax = b$

    $x \geq 0$ und ganzzahlig

    Spezialfall: 0/1 IP oder Binary Integer Program

    $\min\ c^T x$

    unter $Ax = b$

    $x_j \in \{0, 1\}$
  - Die LP Relaxation eines IP ergibt sich durch Weglassen der Ganzzahligkeitsbedingungen, d.h.

    $\min\ c^T x$

unter $Ax = b$

$x \geq 0$

im allgemeinen Fall und

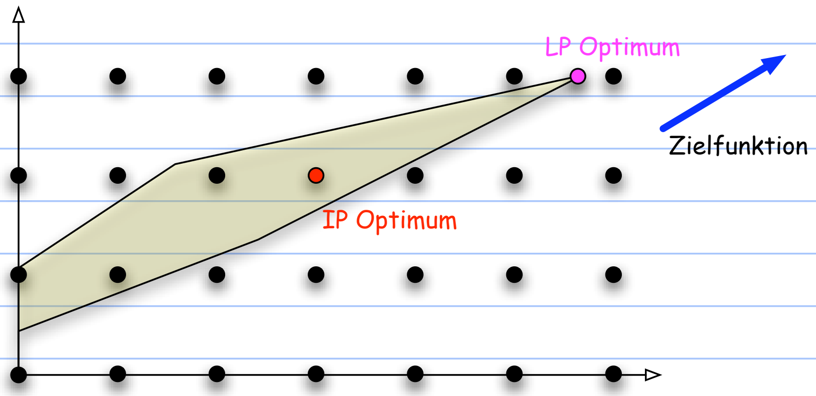
$\min\ c^T x$

unter $Ax = b$

$0 \leq x_j \leq 1$

im 0/1 Fall

- Lösen der LP Relaxation und anschließendes Runden der Variablen
  - erzeugt i.A. keine zulässige Lösung
  - Ohne weitere Überlegungen nur sinnvoll bei großen Werten der Variablen, aber auch dann sind große Fehler möglich



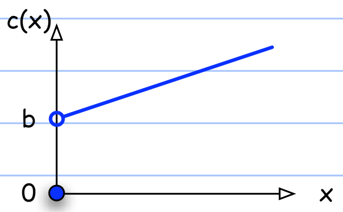
- Sind die $x_j \in \{0, 1\}$ (Entscheidungsvariable)

  z.B. die f_{ij} beim Kürzeste-Wege-Problem

  so ist Runden einer nicht ganzzahligen Lösung zunächst einmal nicht sinnvoll

  (später vorsichtig genutzt für Approximationsalgorithmen)

- Modellierungsmöglichkeiten mit IPs
  - (1) Fixkosten (Fixed Charge Problem)



$c(x) = \begin{cases} ax + b & \text{falls} \quad x > 0 \\ 0 & \text{falls} \quad x = 0 \end{cases}$

$x \in \mathbb{R}^1$, $a, b > 0$

- Führe 0/1-Variable δ und die Restriktion $x \le \delta \cdot U$ ein, wobei U eine obere Schranke für die Werte von x ist
- Claim: Kostenfunktion ist jetzt modellierbar als $c(x,\delta) = ax + b\delta$
  - $x > 0 \Rightarrow \delta = 1 \Rightarrow c(x) = ax + b$
    $x = 0 \Rightarrow \delta = 0$ im Optimum, da $a, b > 0 \Rightarrow c(x) = 0$ ❑

(2) Disjunktive Bedingungen

- $x \ge a$ oder $y \ge b$ mit $a, b \ge 0$ und $x, y \ge 0$
- Führe 0/1-Variable δ ein
- Claim: Die Ungleichungen $x \ge \delta a$ und $y \ge (1-\delta)b$ modellieren die disjunktive Bedingung
  - klar, da δ eine 0/1 Variable ❑

(3) Konditionale Bedingungen

- <u>if</u> $x < a$ <u>then</u> $y \ge b$ <u>else</u> $y \ge 0$ mit $a, b > 0$
- Claim: Die Konditionale Bedingung kann auf Fall (2) reduziert werden
  - die Konditionale Bedingung ist äquivalent zu

    $y \ge 0$

    $x \ge a$ oder $y \ge b$ ❑

(4) Diskrete Variable

- $x \in \{ s_1, \dots, s_m \}$
- Claim: Diskrete Variable $x \in \{ s_1, \dots, s_m \}$ können modelliert werden durch

  $x = s_1\delta_1 + \dots + s_m\delta_m$ mit $\delta_j \in \{ 0, 1 \}$ und $\delta_1 + \dots + \delta_m = 1$
  - klar ❑

7.1 Beispiel (Minimum Weight Perfect Matching Problem als IP)

- Jede Lösung des IP

$\min \sum_{e \in E} c(e)x_e$

$x(\delta(v)) = 1$ für alle $v \in V$

$x_e \in \{0, 1\}$

ist ein perfektes Matching

Komplexität von ILPs

7.2 Satz (Komplexität von ILPs)

(1) SATISFIABILITY (SAT) ist reduzierbar auf ILP

(2) Es ist NP-schwer zu entscheiden, ob ein ILP eine zulässige Lösung hat

(3) Es ist NP-schwer, von einer zulässigen Lösung der LP Relaxation eines ILP auf eine zulässige Lösung des ILP zu runden

Beweis

Sei eine Instanz von SAT gegeben durch m Klauseln $C_1, \ldots, C_m$ in Booleschen Variablen $x_1, \ldots, x_n$

Führe für jede Boolesche Variable x_i eine 0/1-Variable z_i ein mit $z_i = 1$ falls x_i = TRUE

Dann lässt sich die Erfüllung einer Klausel als lineare Ungleichung schreiben und die Existenz einer erfüllenden Belegung ist äquivalent zur Existenz einer zulässigen Lösung für das ILP.

Beispiel:

$\underbrace{x_1 \vee x_2 \vee x_3}_{C_1}, \underbrace{x_1 \vee \bar{x}_2}_{C_2}, \underbrace{x_2 \vee \bar{x}_3}_{C_3}, \underbrace{x_3 \vee \bar{x}_1}_{C_4}, \underbrace{\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3}_{C_5}$

ist äquivalent zu

$z_1 + z_2 + z_3 \geq 1$

$z_1 + (1-z_2) \geq 1$

$z_2 + (1-z_3) \geq 1$

$z_3 + (1-z_1) \geq 1$

$(1-z_1) + (1-z_2) + (1-z_2) \geq 1$

$z_i \in \{0, 1\}$

Hat jede Klausel ≥ 2 Literale (dies ist der nicht-triviale Fall), so ist $z_i = 1/2$ eine zulässige Lösung der LP Relaxation. Das Runden auf eine zulässige Lösung des ILP ist daher genauso schwer wie das Finden einer efüllenden Belegung für die gegebene SAT Instanz. ❑

Beachte: Der Beweis zeigt nicht, dass der Test auf Zulässigkeit NP-vollständig ist. Dazu müssten wir zeigen, dass ein Zertifikat für Zulässigkeit von polynomialer Länge existiert (vgl. ADM I). Das ist zunächst unklar, es kann jedoch gezeigt werden, dass die Komponenten x_i einer ganzzahligen zulässigen Lösung x nicht zu groß werden (eine analoge Aussage zu Lemma 3.4). Daher kann x selbst als Zertifikat genommen werden. NP-schwer kann also überall in Satz 7.2 durch NP-vollständig ersetzt werden.

Fragestellung dieses Abschnittes

- Wann hat ein LP ganzzahlige Basislösungen?
  => Dann lässt sich ein ILP dadurch lösen indem man die LP Relaxation mit dem Simplexalgorithmus löst.
- Hier der Speziallfall:
  Wann hat $Ax = b$ nur ganzzahlige Basislösungen für beliebige Wahl von gannzzahligen rechten Seiten b?
  Dies ist dann eine Eigenschaft der Matrix A

Vollständig unimodulare Matrizen

- Eine quadratische Matrix B mit ganzzahligen Einträgen heißt unimodular
  :<=> $\det B \in \{ -1, 1 \}$
- Eine Matrix A mit ganzzahligen Einträgen heißt vollständig unimodular (totally unimodular, TUM)
  :<=> jede quadratische nicht-singuläre Teilmatrix ist unimodular

- Erste Eigenschaften
  - A TUM => A hat nur Einträge $a_{ij} \in \{ -1, 0, 1 \}$
  - die kleinste nicht unimodulare Matrix ist

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

- Ist B eine Basis von A mit $B = (A_{B(1)}, A_{B(2)}, \ldots, A_{B(m)})$, so folgt aus der Cramerschen Regel

  $x_{B(i)} = \dfrac{\det B^i}{\det B}$ mit $B^i = (A_{B(1)}, \ldots, A_{B(i-1)}, b, A_{B(i+1)}, \ldots, A_{B(m)})$

  => $x_{B(i)}$ ganzzahlig falls A TUM und b ganzzahlig

Polyeder von linearen Optimierungsproblemen mit ganzzahligen Ecken

Sei $R_1(A) := \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \}$ das Polyeder zur Standardform des LP

Sei $R_2(A) := \{ x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0 \}$ das Polyeder zur kanonischen Form des LP

Bemerkung:

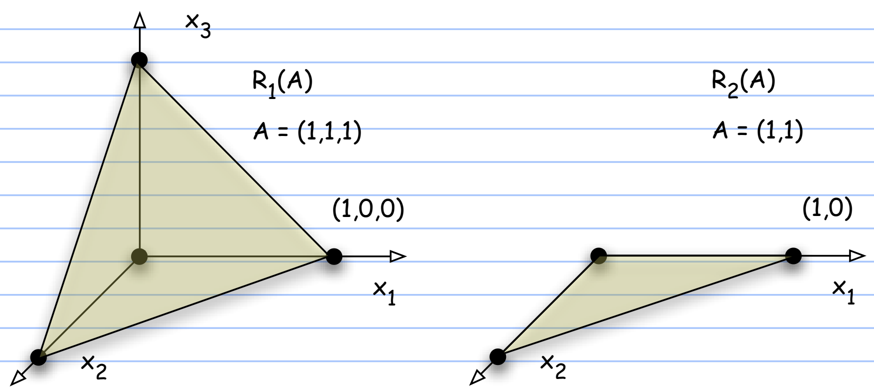
- Beide Polyeder sind hier als Teilmengen des $\mathbb{R}^n$ definiert.
- Die Definition $R_2(A)$ entspricht der in Abschnitt 3.3 betrachtenen Korrespondenz zwischen geometrischer und algebraischer Interpretation von LPs, insbesondere entsprechen die Ecken von $R_2(A)$ den zulässigen Basislösungen des um Schlupfvariablen erweiterten LP $\{ Ax + s = b, x, s \geq 0 \}$

- Die Ecken von $R_1(A)$ entsprechen ebenfalls den zulässigen Basislösungen des LP $\{ Ax = b, x \geq 0 \}$

  denn:

  Ist x zulässige Basislösung zur Basis B, so ist $x_N = 0$, d.h. x liegt im Schnitt der n-m Hyperebenen $x_i = 0$ mit $i \in N$ und der m Hyperebenen $a_i x = b_i$, $i = 1, \ldots, m$.

  Die Umkehrung folgt mit ähnlichen Überlegungen zu denen im Beweis zu Satz 3.6.
- Beispiel:



Hier hat x_3 die Rolle der Schlupfvariablen für $R_2(A)$

7.3 Satz (Ganzzahligkeit von $R_1(A)$)

Ist A vollständig unimodular, so sind alle Ecken von $R_1(A)$ für beliebige ganzzahlige rechte Seite b ganzzahlig.

Insbesondere führt der Simplexalgorithmus in einem LP in Standardform mit vollständig unimodularer Matrix A bei ganzzahliger rechter Seite b immer zu einer ganzzahligen Optimallösung.

Beweis:

folgt aus dem Abschnitt "Erste Eigenschaften" ❑

7.4 Satz (Ganzzahligkeit von $R_2(A)$)

Ist A vollständig unimodular, so sind alle Ecken von $R_2(A)$ für beliebige ganzzahlige rechte Seite b ganzzahlig.

Insbesondere führt der Simplexalgorithmus in einem LP in kanonischer Form mit vollständig unimodularer Matrix A bei ganzzahliger rechter Seite b nach Einführung von Schlupfvariablen immer zu einer ganzzahligen Optimallösung.

Beweis:

Nach Einführung von Schlupfvariablen entsteht die Matrix $(A|I)$.

Sei C eine nicht-singuläre quadratische Teilmatrix von $(A|I)$

=> nach geeigneter Permutation der Zeilen hat C die Form

$$\left(\begin{array}{c|c} B & 0 \\ \hline D & I_k \end{array}\right)$$

mit B = quadratische Teilmatrix von A

I_k = (k,k)-Einkeitsmatrix

=> $|\det(C)| = |\det(B)| = 1$, da A TUM

=> $(A|I)$ ist TUM

=> Behauptung mit Satz 7.3 und Satz 3.10 ❑

Die Sätze 7.3 und 7.4 bedeuten also, dass die Polyeder $R_1(A)$ und $R_2(A)$ ganzzahlige Ecken haben, wenn A vollständig unimodular und die rechte Seite b ganzzahlig ist.

Erkennung von vollständig unimodularen Matrizen

Die Komplexität der Erkennung vollständig unimodularer Matrizen war lange offen und wurde erst durch

Seymour 1980 gelöst, der in einem "Dekompositionssatz" zeigte, dass sich jede vollständig unimodulare Matrix aus "einfachen" vollständig unimodularen Matrizen nach gewissen Konstruktionsprinzipien ergibt. Hieraus lässt sich ein polynomialer Algorithmus zur Erkennung vollständig unimodularer Matrizen herleiten mit einer Laufzeit von $O((m+n)^4 m)$

Details siehe Kapitel 19 und 20 in

A. Schrijver
Theory of Linear and Integer Programming
Wiley 1986

- Hier nur ein hinreichendes Kriterium

- 7.5 Satz (Ein hinreichendes Kriterium für vollständige Unimodularität)
  - Eine Matrix A mit Einträgen $a_{ij} \in \{ -1, 0, 1 \}$ ist vollständig unimodular, falls sie folgende Bedingungen erfüllt
    - (1) A hat pro Spalte maximal 2 Einträge $\neq 0$
    - (2) Die Zeilen von A können in zwei disjunkte Mengen I_1, I_2 aufgeteilt werden mit
      - Für jede Spalte mit 2 Einträgen $\neq 0$ und gleichem Vorzeichen liegen die zugehörigen Zeilen in

verschiedenen I_j
      - Für jede Spalte mit 2 Einträgen $\neq 0$ und verschiedenen Vorzeichen liegen die zugehörigen Zeilen in demselben I_j
  - Beweis durch Induktion nach der Größe k der quadratischen Teilmatrix
    - Induktionsanfang $k = 1$
      - klar, da A nur Einträge $a_{ij} \in \{ -1, 0, 1 \}$ hat
    - Schluss auf k
      - Sei C eine quadratische nichtsinguläre (k,k)-Teilmatrix von A

        => in jeder Spalte von C gibt es mindestens einen Eintrag
      - Fall 1: es gibt in C eine Spalte mit genau einem Eintrag $a_{ij} \neq 0$
        - entwickle $\det(C)$ nach dieser Spalte, wobei C' die Untermatrix von C nach Streichen von Zeile i und Spalte j ist

          => $|\det(C)| = |a_{ij}| \cdot |\det(C')|$

          C nicht singulär => $|\det(C')| \neq 0$

          Induktionsvoraussetzung => $|\det(C')| = 1$

          $a_{ij} \in \{ -1, 1 \}$ => $|\det(C)| = 1$

Fall 2: alle Spalten von C haben mindestens 2 Einträge $\neq 0$

(1) => alle Spalten haben genau 2 Einträge $\neq 0$

Betrachte die Aufteilung der Zeilen in I_1, I_2 gemäß (2)

=> für jede Spalte j ist $\sum_{i \in I_1} a_{ij} = \sum_{i \in I_2} a_{ij}$

=> $\sum_{i \in I_1} a_i - \sum_{i \in I_2} a_i = 0$

d.h. eine Linearkombination der Zeilenvektoren von C ergibt den Nullvektor

=> Widerspruch zu C nicht-singulär

=> dieser Fall kann nicht auftreten ❑

7.6 Korollar (Wichtige vollständig unimodulare Matrizen)

Jedes LP in Standardform oder kanonischer Form, dessen Koeffizientenmatrix gleich der

1. Knoten-Kanten Inzidenzmatrix eines Digraphen
2. Knoten-Kanten Inzidenzmatrix eines bipartiten Graphen

ist, hat nur ganzzahlige optimale Basislösungen (bei ganzzahliger rechter Seite b).

Hierzu gehören u.A. die LP Formulierungen des

Kürzeste-Wege-Problem

Max-Fluss-Problem

Transportproblem

Beweis

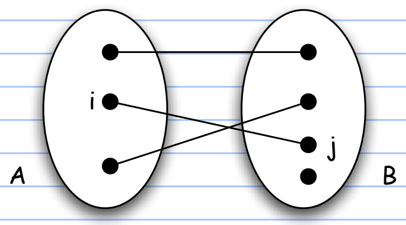
Fall 1

A enthält in diesem Fall pro Spalte genau eine $+1$ und eine -1

=> setze I_1 = Menge aller Zeilen, $I_2 = \emptyset$

Fall 2

Sei G der bipartite Graph mit Bipartition A und B



=> die Spalte zur Kante ij enthält genau 2 Einträge $\neq 0$, und zwar eine +1 für den Knoten i und eine +1 für den Knoten j

=> setze $I_1 = A$, $I_2 = B$ ❑

Der Satz von Birkhoff & von Neumann über doppelt stochastische Matrizen

Wir beweisen mit unseren Mitteln einen bekannten und nützlichen Satz über doppelt stochastische Matrizen

Eine nxn-Matrix mit Einträgen $0 \leq a_{ij} \leq 1$ heißt doppelt stochastisch

:<=> jede Zeilensumme und jede Spaltensumme ergibt 1

Eine nxn-Matrix mit Einträgen $a_{ij} \in \{ 0, 1 \}$ heißt Permutationsmatrix

:<=> in jeder Zeile und in jeder Spalte steht genau eine 1

7.7 Satz (Birkhoff 1946, von Neumann 1953)

Jede doppelt stochastische nxn-Matrix lässt sich als Konvexkombination von nxn-Permutationsmatrizen darstellen

Beweis

Eine doppelt stochastische Matrix M lässt sich auffassen als zulässige Lösung des Zuordnungsproblems

min $\sum_{i,j} c_{ij}f_{ij}$ unter

$\sum_j f_{ij} = 1$ für alle $i = 1, \ldots, n$

$\sum_i f_{ij} = 1$ für alle $j = 1, \ldots, n$

$f_{ij} \geq 0$ für alle i, j

Sei A die zugehörige Koeffizientenmatrix und $R_1(A)$ das zugehörige Polyeder zur Standardform

$R_1(A)$ ist ein Polytop, da der Zulässigkeitsbereich wegen $0 \leq f_{ij} \leq 1$ beschränkt ist

Satz von Minkowski (Satz 3.9) => M ist Konvexkombination der Ecken von $R_1(A)$

A ist dabei Knoten-Kanten Inzidenzmatrix des vollständigen bipartiten Graphen $K_{n,n}$

=> A ist vollständig unimodular nach Korollar 7.6

=> Die Ecken von $R_1(A)$ sind ganzzahlig nach Satz 7.3

$0 \leq f_{ij} \leq 1$ => die Ecken von $R_1(A)$ sind Permutationsmatrizen ❏

Ziel dieses Abschnitts

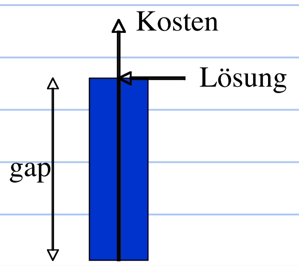
- Vorstellung von Branch and Bound als eine Standard-Technik zur exakten Lösung NP-vollständiger Probleme, speziell von IPs.

  Obwohl einfach, ist Branch and Bound die Grundlage und das Arbeitspferd für alle kommerziellen Codes zur Lösung von IPs, allerdings angereichert mit einer Vielzahl von Verbesserungen und Tricks.

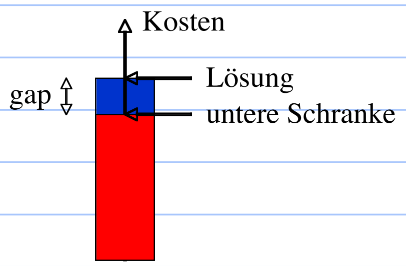
Grundidee von Branch and Bound

- Branch and Bound (B&B) = geschickt (Problem-abhängig) organisierte systematische Durchforstung der Menge der zulässigen Lösungen nach einer Optimallösung oder bis zum Abbruch mit einer guten Lösung (d.h. mit einer Instanz-abhängigen Gütegarantie)

- Der Nutzen unterer Schranken bei der Minimierung

- 

  wissen nicht wie gut eine zulässige Lösung ist, wenn sie heuristisch erzeugt wird (z.B. mit lokaler Suche)

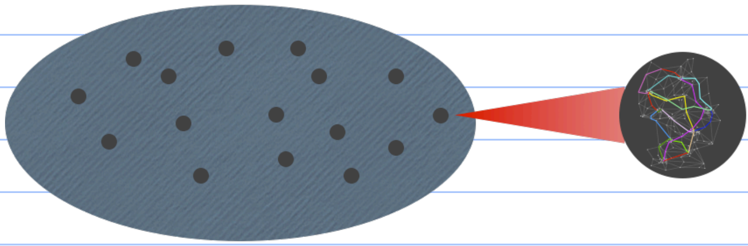
- 

  untere Schranken für den Optimalwert schränken die "Optimalitätslücke" ein

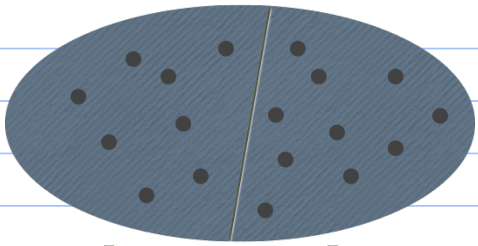
- Also: Ist das Optimum unbekannt, so ergeben untere Schranken Qualitäts-Garantien für die Lösung von schweren Optimierungsproblemen

- Branch & Bound
  - illustriert am Disjunkte-Wege-Problem (vgl. Abschnitt 4.4)
  - Stellen uns Lösungsraum (= Menge der zulässigen Lösungen) als Punktewolke vor

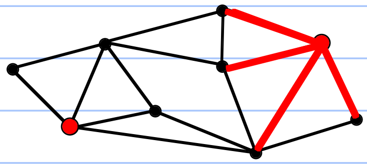
    Jeder Punkt repräsentiert eine zulässige Lösung



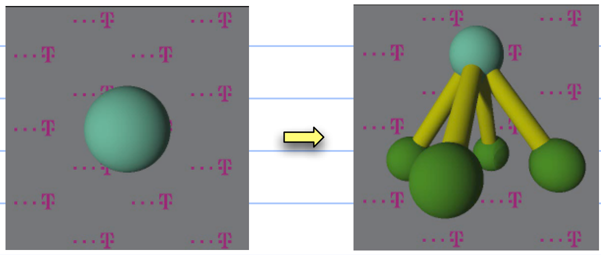
  - Branching = Aufteilung der momentanen Lösungsmenge in ≥ 2 Teilmengen (nicht notwendig disjunkt)



- wird meistens als Baum dargestellt (Branch and Bound Tree)



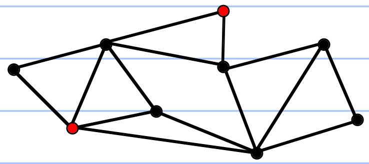
hier: teile Menge der Lösungen in 4 Teilmengen je nachdem welche rote Kante zur Verbindung der roten Terminale verwendet wird



die Teilmengen sind dann im Baum Kinder der geteilten Menge

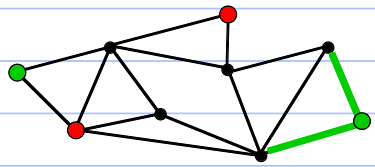
- dieser Prozess des Branching wird iteriert, dadurch entsteht der B&B Baum

  hier: Wahl der oberen roten Kante erzeugt dieses Teilproblem, in dem die obere rote Kante weggelassen wird (also für den verbindenden Weg reserviert wird) und das rote Terminal entsprechend verschoben wird

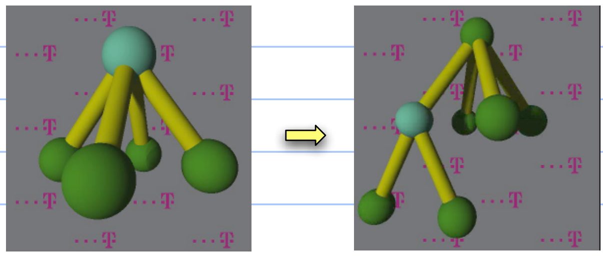


die Menge der zugehörigen zulässigen Lösungen kann dann bzgl einen eines anderen (oder desselben)

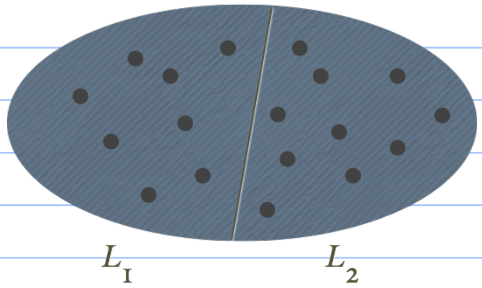
Terminalpaares weiter zerlegt werden



hier: Wahl der grünen Terminale und Aufteilung in 2 Teilmenge in Abhängigkeit der gewählten grünen Kante



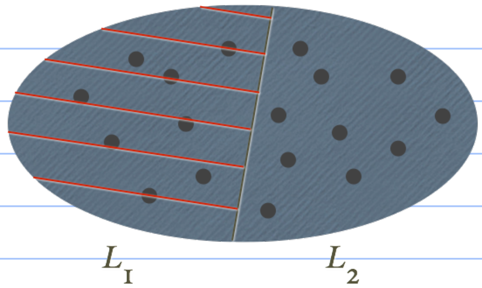
- Bounding = Abschneiden von Ästen des B&B Baums durch Verwendung unterer Schranken



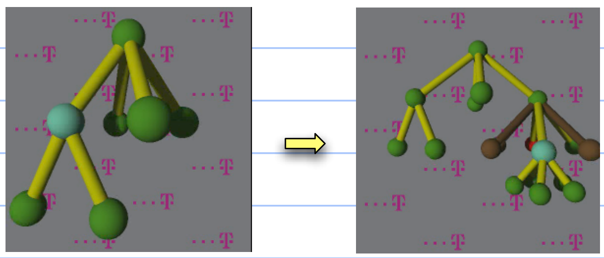
Annahme 1: wir kennen eine zulässige Lösung mit Kosten k

Annahme 2: wir kennen eine untere Schranke s für den Optimalwert in L_1

=> brauchen L_1 nicht zu untersuchen, wenn $s \geq k$

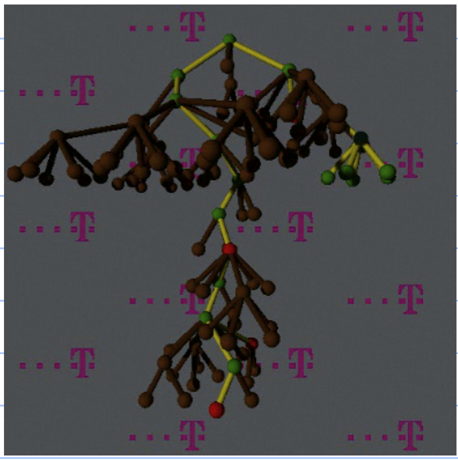


dieses Abschneiden von Ästen des B&B Baums bezeichnet man auch als Pruning, im Baum hier durch braune (verdorrte) Äste dargestellt. Der Knoten wird dann als fathomed bezeichnet.



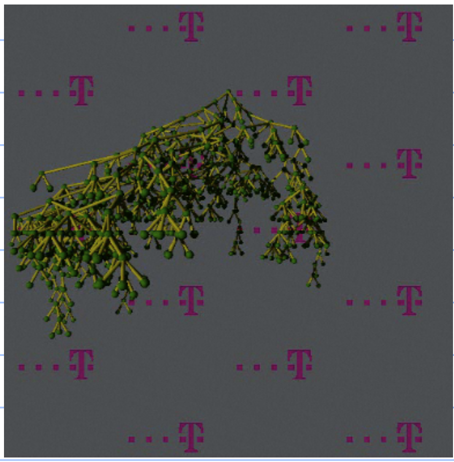
- Branching und Bounding wird verbunden mit
  - guten Auswahlstrategien zur Untersuchung der nächsten Knoten (= Teilmenge der zulässigen Lösungen) im B&B Baum
    - Tiefensuche
    - Breitensuche
    - Best-First-Search (gehe in Richtung bester (= kleinster) unterer Schranke)
    - Kombinationen davon
  - der Baum wird natürlich nur implizit verwaltet und nie explizit erzeugt
  - Techniken zur Erzeugung guter unterer Schranken (nächstes Kapitel)
    - Lagrange Relaxation
    - LP-Relaxation (speziell bei IPs)
  - Techniken zur Erzeugung zulässiger Lösungen (oberer Schranken) in Baumknoten

- Laufzeit ist exponentiell, hängt sehr von der Qualität der unteren Schranken ab



gute untere Schranken

kleiner B&B Baum



schlechte untere Schranken

riesiger B&B Baum

- **Schematische Darstellung von Branch and Bound**
  - Input
    - Instanz I eines Problems

- Output
  - zulässige Lösung $x \in S_I$ mit Gütegarantie in Form des Zielfunktionswertes $c(x)$ und einer unteren Schranke ℓ für den Optimalwert
- Ingredienzen
  - lower bounding strategy
  - branching strategy
  - search strategy
- Methode
  - 1. Arbeit in der Wurzel
    - betrachte eine abgewandelte, leichter zu lösende Instanz I' (Relaxation) zur Bestimmung einer unteren Schranke für I;
      berechne die Optimallösung x' von I' mit Zielfunktionswert z';
      if $x' \in S_I$ then return x' // x' ist optimal
    - setze $\ell := z'$ // anfängliche globale untere Schranke
    - // initialisiere Datenstruktur D zur Verwaltung der noch zu untersuchenden Knoten des B&B Baums
      Füge I mit $\ell(I) := \ell$ in D ein

- Verwende Heuristiken zur Erzeugung zulässiger Lösungen
  setze $x^* :=$ beste gefundene Lösung
  setze $u :=$ bester gefundener Zielfunktionswert // anfängliche obere Schranke
- 2. Main loop
  - <u>while</u> Güte $(u-\ell)/\ell$ nicht klein genug <u>and</u> noch Rechenzeit und Speicher verfügbar <u>do</u>
    - wähle nächsten zu untersuchenden Knoten v des B&B Baumes aus D // search strategy
    - <u>if</u> $\ell(v) \geq u$ <u>then</u> lösche v aus D // pruning
    - <u>else</u>
      - erzeuge die Kinder $v_1, ..., v_k$ von v // branching rule
        // Vereinigung der Zulässigkeitsbereiche der Kinder = Zlässigkeitsbereich von v
      - <u>for</u> jedes Kind v_i <u>do</u>
        - berechne die Optimallösung x' (der Relaxation) des zugehörigen Teilproblems mit Zielfunktionswert z' // bounding rule
        - <u>if</u> $x' \in S_I$ and $z' < u$ <u>then</u>
          - $x^* := x'$ // Aktualisierung bester bekannter zulässiger Lösung
            $u := z'$ // Aktualisierung der oberen Schranke

- else
  - if $z' < u$ then füge v_i mit $\ell(v_i) := z'$ in D ein // neues Teilproblem
- lösche v aus D // v ist abgearbeitet
- $\ell := \min \{ \ell(w) \mid w \text{ in } D \}$ // aktualisiere globale untere Schranke
- return x^* und ℓ

Branch and Bound bei IPs

- Für das Bounding liegt die Verwendung der LP Relaxation nahe
- Für das Branching liegt das Branching bzgl. fraktionaler Variablen in der LP Relaxation nahe

7.8 Beispiel (Das KNAPSACK Problem, vgl. ADM I)

KNAPSACK

Instanz

- n Gegenstände (items) mit Gewicht w_i und Wert (Profit) c_i
- ein Rucksack mit Kapazität W

Aufgabe

Finde eine Teilmenge $S \subseteq \{ 1, ..., n \}$ mit

- maximalem Wert $c(S) := \sum \{ c_j \mid j \in S \}$
- Kapazität des Rucksacks wird nicht überschritten, d.h. $w(S) := \sum \{ w_j \mid j \in S \} \leq W$

Eine IP Formulierung von KNAPSACK

- Führe 0/1-Variable x_j ein mit $x_j = 1$, falls der Gegenstand j mitgenommen wird

$$\min \sum_j -c_j x_j$$
$$\sum_j w_j x_j \leq W$$
$$x_j \in \{ 0, 1 \}$$

7.9 Lemma (Optimallösungen der LP Relaxation von KNAPSACK)

Man erhält eine Optimallösung der LP-Relaxation

$$\min \sum_j -c_j x_j$$
$$\sum_j w_j x_j \leq W$$
$$0 \leq x_j \leq 1$$

der IP Formulierung von KNAPSACK wie folgt

- sortiere und nummeriere die Gegenstände so, dass $c_1/w_1 \geq c_2/w_2 \geq ... c_n/w_n$ (größter Nutzen pro

Gewichtseinheit zuerst)

- berechne in dieser Reihenfolge die kleinste Zahl k, so dass $w_1 + w_2 + \ldots + w_{k+1} > W$
- setze $x_1 = x_2 = \ldots = x_k = 1$

  $x_{k+1} = (W - w_1 - w_2 - \ldots - w_k)/w_{k+1}$

  $x_j = 0$ sonst

- Beweis durch Überprüfung der Bedingungen vom komplementären Schlupf
  - primal duales Paar ist gegeben durch

$$
\begin{array}{c|ccc|c}
 & c_1 & \ldots & c_n & \\
\hline
u & w_1 & \ldots & w_n & W \\
v_1 & 1 & & & 1 \\
 & & \ddots & & \\
v_n & & & 1 & 1 \\
\hline
 & x_1 & \ldots & x_n &
\end{array}
$$

  - die Bedingungen vom komplementären Schlupf ergeben
    - (1) $x_j > 0 \Rightarrow w_j u + v_j = c_j$
    - (2) $u > 0 \Rightarrow \sum_j w_j x_j = W$ (ist von der Lösung x erfüllt)
    - (3) $v_j > 0 \Rightarrow x_j = 1$

- Definiere nun zu x eine dual zulässige Lösung, die mit x die Bedingungen (1) und (3) erfüllt
  - (3) $\Rightarrow v_{k+1} = v_{k+2} = \ldots = v_n := 0$

    $\Rightarrow$ (mit (1) für $j = k+1$) $w_{k+1}u = c_{k+1} \Rightarrow u = c_{k+1}/w_{k+1}$

    $\Rightarrow$ (mit (1) für $j = 1, \ldots k$) $w_j(c_{k+1}/w_{k+1}) + v_j = c_j$

    $\Rightarrow v_j := c_j - w_j(c_{k+1}/w_{k+1})$ für $j = 1, \ldots k$

    $\Rightarrow$ Werte für alle Dualvariablen definiert aus den Bedingungen (1) und (3)
  - zeige noch: dies definiert eine dual zulässige Lösung

    dazu muss nur noch $v_j \geq 0$ gezeigt werden, d.h. $c_j - w_j(c_{k+1}/w_{k+1}) \geq 0$ für $j = 1, \ldots k$.

    Dies folgt aus $c_j/w_j \geq c_{k+1}/w_{k+1}$ für $j = 1, \ldots k$ ❑

- Nutze allgemeines B&B Schema mit folgenden Ingredienzen
  - lower bounding strategy = LP Relaxation gelöst mit Lemma 7.9
  - branching strategy = branche auf fraktionaler Variablen x_{k+1}
  - search strategy = best first

- Daten der Instanz

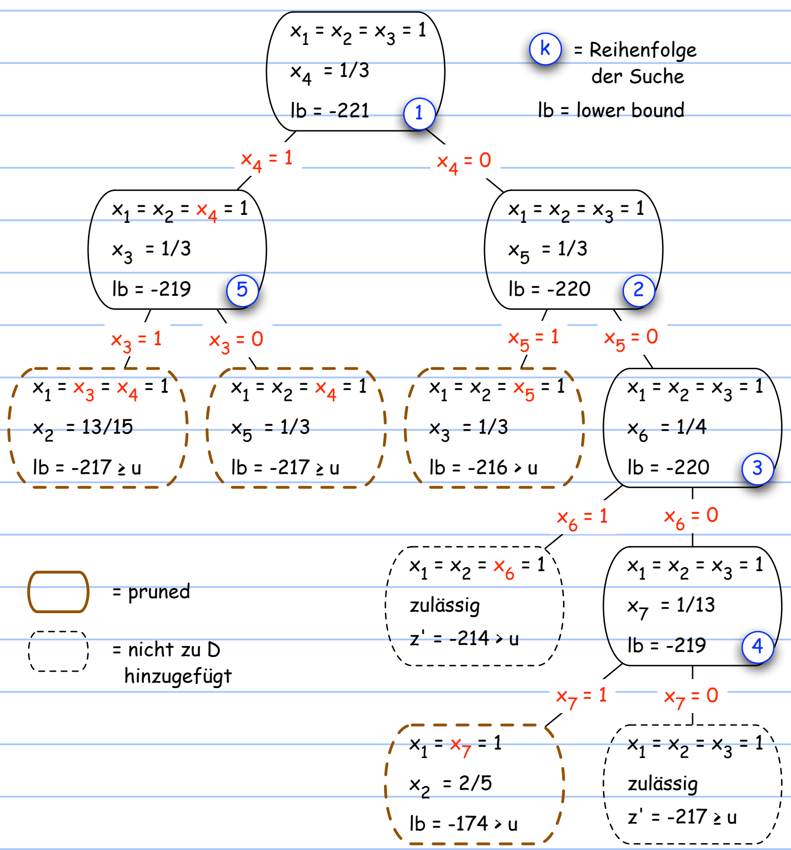
j	w_j	c_j	$\frac{c_j}{w_j}$
1	16	112	7
2	15	90	6
3	3	15	5
4	3	12	4
5	3	9	3
6	4	12	3
7	13	26	2

$W = 35$

heuristische Lösung $x_1 = x_2 = x_3 = 1$, $x_j = 0$ sonst => obere Schranke $u = -217$

LP Relaxation ergibt $x_1 = x_2 = x_3 = 1$, $x_4 = 1/3$, $x_j = 0$ sonst => untere Schranke $\ell = -221$

Branch and Bound Tree



- **Verwendung anderer Relaxationen als der LP-Relaxation**
  - Dies ist möglich, z.B. durch Weglassen von Nebenbedingungen

    => Zulässigkeitsbereich wird größer => Minimum wird kleiner

  - 7.10 Beispiel (TSP in Digraphen)
    - Eine IP Formulierung
      - Führe 0/1-Variable x_{ij} ein mit $x_{ij} = 1 \Leftrightarrow$ Kante (i,j) ist in der Tour

        $\min \sum_{ij} c_{ij} x_{ij}$

        $\sum_j x_{ij} = 1$ für alle $i = 1, \ldots, n$ (7.1)

        $\sum_i x_{ij} = 1$ für alle $j = 1, \ldots, n$ (7.2)

        $\sum_{i,j \in S} x_{ij} \leq |S|-1$ für alle $\emptyset \neq S \subset \{ 1, \ldots, n \}$ (7.3)

        $x_{ij} \in \{ 0, 1 \}$ (7.4)

    - Die Cycle Cover Relaxation des TSP
      - Ergibt sich durch Weglassen der Bedingung (7.3).

      - Die verbleibenden Bedingungen beschreiben ein Zuordnungsproblem, wobei jedoch Kanten (i,i) nicht vorkommen dürfen. Dies kann in der Zielfunktion durch hohe Kosten c_{ii} berücksichtigt werden. Solche Zuordnungsprobleme sind effizient lösbar, z.B. mit der primal-dualen Methode, vgl. Abschnitt 6.5.

    - Verwendung der Cycle Cover Relaxation in Branch and Bound Algorithmen
      - Nehme die Cycle Cover Relaxation als lower bounding strategy
      - Falls die optimale Zuordnung (x_{ij}) Bedingung (7.3) erfüllt, so ist sie eine Tour. Ansonsten branche wie folgt:
        - wähle den kürzesten Zykel bzgl. Kantenzahl und setze die Kanten darauf einzeln zu 0

          => pro Kante im Zykel ein Kind im B&B Baum

Hauptpunkte dieses Abschnittes

- Lagrange Relaxation ist eine wichtige Technik zur Erzeugung von "guten" unteren Schranken für IPs. Sie relaxiert Nebenbedingungen, bestraft aber ihre Verletzung in der Zielfunktion. Durch Variation der Strafkosten kann die untere Schranke verbessert werden.
- Die systematische Verbesserung der Strafkosten führt zur Subgradientenoptimierung, einer Methode zur Maximierung einer nicht-differenzierbaren konkaven Funktion
- Die dadurch erreichbare untere Schranke ist mindestens so gut wie bei der LP Relaxation, unter bestimmten Bedingungen jedoch gleich. Der Vorteil zur LP Relaxation liegt (Problem-abhängig) in der schnelleren (approximativen) Berechnung der Schranke durch kombinatorische Algorithmen.
- Lagrange Relaxation ist eins der Arbeitspferde in Branch and Bound Algorithmen

Grundlagen der Lagrange Relaxation

- Gegeben sei das ganzzahlige Lineare Programm

(P) min $c^T x$

unter $Ax \geq b$ (k "schwere" Nebenbedingungen)

$Bx \geq d$ (m-k "leichte" Nebenbedingungen)

x ganzzahlig

- Relaxiere die "schweren" Nebenbedingungen $Ax \geq b$ und bestrafe ihre Verletzung in der Zielfunktion. Führe dazu Lagrange-Multiplikatoren $\lambda_1, ..., \lambda_k$ für diese Nebenbedingungen ein. Sie bilden eine Art Dualvariable für diese Nebenbedingungen und erfüllen die Bedingungen

$a_i x \geq b_i \Rightarrow \lambda_i \geq 0$ (7.5)

$a_i x = b_i \Rightarrow \lambda_i$ nicht vorzeichenbeschränkt (7.6)

Für festes solches $\lambda = (\lambda_1, ..., \lambda_k)^T$ ist die Lagrange Relaxation (LR_λ) von (P) definiert als

(LR_λ) min $c^T x + \lambda^T(b - Ax) =: L(\lambda, x)$

unter $Bx \geq d$

x ganzzahlig

$L(\lambda,x)$ wird Lagrange Funktion genannt, $\lambda = (\lambda_1, ..., \lambda_k)^T$ heißt auch Lagrange-Vektor und kann als Vektor von Strafkosten interpretiert werden.

Wir bezeichnen die Zulässigkeitsbereiche von (P) und (LR_λ) mit $S(P)$ und $S(LR_\lambda)$ und die zugehörigen Optimalwerte mit $z(P)$ und $z(LR_\lambda)$.

7.11 Lemma (Lagrange Relaxation liefert untere Schranken)

Für jeden Lagrange Vektor λ gilt

(1) $S(LR_\lambda) \supseteq S(P)$

(2) $z(LR_\lambda) \le z(P)$

Beweis

(1) ist trivial, da Nebenbedingungen weggelassen werden

zu (2)

Sei x optimal bzgl. (P)

=> $b_i - a_i x \le 0$ bzw $b_i - a_i x = 0$ bei Gleichheitsrestiktionen

=> $\lambda_i(b_i - a_i x) \le 0$ für alle i => $\lambda^T(b - Ax) \le 0$

=> $z(P) = c^Tx \ge c^Tx + \lambda^T(b - Ax) \ge z(LR_\lambda)$ da $x \in S(P) \subseteq S(LR_\lambda)$ ❑

7.12 Lemma (Optimalitätsbedingungen)

Genügen x und λ den Bedingungen

(1) x ist optimal bzgl. (LR_λ)

(2) $a_i x \ge b_i$ bzw. $a_i x = b_i$ bei Gleichheitsrestriktionen

(3) $\lambda^T(b - Ax) = 0$

so ist x optimal bzgl. (P). Ist (3) nicht erfüllt, so ist x ε-optimal mit $\varepsilon = \lambda^T(b - Ax)$

Beweis

(1), (2) => $x \in S(P)$

=> $z(LR_\lambda) = c^Tx + \lambda^T(b - Ax) = c^Tx \ge z(P)$ wegen (3) und $x \in S(P)$

=> $z(LR_\lambda) = z(P)$ wegen Lemma 7.11.

Ist (3) nicht erfüllt, so ist $\lambda^T(b - Ax)$ der Fehlerterm ❑

Ziel der Lagrange Relaxation

Aufteilung der Restriktionen von (P) so, dass (LR_λ) im Verhältnis zu (P) leicht lösbar ist

$z(P) - z(LR_\lambda)$ möglichst klein machen (Dualitätslücke der Lagrange Relaxation)

d.h. $L(\lambda) := z(LR_\lambda)$ möglichst groß machen durch Variation der Lagrange Multiplikatoren

=> dies führt zu Optimierungsproblem $\max_\lambda L(\lambda)$

Dieses Optimierungsproblem muss bei Verwendung in B&B nicht optimal gelöst werden, es reicht ein guter Wert von $L(\lambda)$, da jeder solche Wert eine untere Schranke für $z(P)$ liefert.

Lagrange Relaxation des symmetrischen TSP mittels 1-Bäumen

IP Formulierung des symmetrischen TSP

Führe 0/1-Variable x_e ein mit $x_e = 1$ <=> Kante e ist in der Tour

(P) $\min \sum_e c_e x_e$

$x(\delta(i)) = 2$ für alle $i = 1, \dots, n$ (7.7)

$x(S) \leq |S|-1$ für alle $\emptyset \neq S \subseteq \{ 2, \dots, n \}$ (7.8)

beachte: $S \subseteq \{ 2, \dots, n \}$ reicht, um Kurzzykel auszuschließen

$x_e \in \{ 0, 1 \}$ (7.9)

Dabei ist $x(S) := \sum_{e = ij,\, i,j \in S} x_e$ und $x(\delta(i)) := \sum_{e \in \delta(i)} x_e$

Variation von (P) führt zu (LR_λ)

spalte (7.7) auf in

$\sum_e x_e = n$ (7.10) redundant in (P)

$x(\delta(i)) = 2$ für $i = 2, \dots, n$ (7.11)

$x(\delta(1)) = 2$ (7.12)

Definiere (LR_λ) durch die Relaxation von (7.11)

(LR_λ) $\min \sum_e c_e x_e + \sum_{i = 2,\dots,n} \lambda_i (2 - x(\delta(i)))$

unter (7.8), (7.9), (7.10), (7.12)

Beachte: (7.10) ist nicht redundant in (LR_λ)

Kombinatorische Struktur der zulässigen Lösungen von (LR_λ)

7.13 Lemma (zulässige Lösungen von (LR_λ) sind 1-Bäume)

x ist zulässige Lösung von (LR_λ) <=> x ist ein 1-Baum, d.h.

x ist ein spannender Baum auf der Knotenmenge $\{ 2, \dots, n \}$

mit 2 zusätzlichen Kanten vom Knoten 1 aus

Beweis

"=>"

x zulässige Lösung von (LR_λ)

(7.9), (7.10), (7.12) => x hat $n-2$ Kanten auf den Knoten $2, \dots, n$

(7.8) => x ist zusammenhängend

ADM I => ein zusammenhängender Graph mit $n-2$ Kanten auf $n-1$ Knoten ist ein spannender Baum

(7.12) => zusätzlich 2 Kanten vom Knoten 1 aus

=> x ist 1-Baum

- "<="
  - jeder 1-Baum erfüllt die Bedingungen (7.8), (7.9), (7.10), (7.12) ❑

- Die Lagrangefunktion $L(\lambda,x)$
  - $L(\lambda,x) = \Sigma_e c_e x_e + \Sigma_{i=2,\dots,n} \lambda_i(2 - x(\delta(i))$, λ_i nicht vorzeichenbeschränkt

    => o.B.d.A. λ_i durch $-\lambda_i$ ersetzen (bessere kombinatorische Interpretation)

    => $L(\lambda,x) = \Sigma_e c_e x_e + \Sigma_{i=2,\dots,n} \lambda_i(x(\delta(i) - 2)$

    mit $x(\delta(i)) - 2$ = Abweichung vom angestrebten Grad 2 des Knoten i
  - Mit $\lambda_1 := 0$ ergibt sich

    $L(\lambda,x) = \Sigma_e c_e x_e + \Sigma_{i=1,\dots,n} \lambda_i(x(\delta(i) - 2)$

    $= \Sigma_e c_e x_e + \Sigma_{i=1,\dots,n} \lambda_i x(\delta(i)) - 2\,\Sigma_{i=1,\dots,n} \lambda_i$

    $= \Sigma_e c_e x_e + \Sigma_{e=ij} (\lambda_i + \lambda_j) x_e - 2\,\Sigma_{i=1,\dots,n} \lambda_i$

    $= \Sigma_{e=ij} (c_e + \lambda_i + \lambda_j)\, x_e - 2\,\Sigma_{i=1,\dots,n} \lambda_i$

    Dies entspricht neuen Kantenkosten $c_e' = c_e + \lambda_i + \lambda_j$ für $e = ij$ abzüglich eines konstanten Terms $2\,\Sigma_{i=1,\dots,n}$

λ_i

- Interpretation der Lagrange Relaxation
  - Relaxiertes Problem

    = Ermittlung eines 1-Baums mit minimalem Gewicht bzgl. der Kantenkosten $c_e + \lambda_i + \lambda_j$ für $e = ij$
  - Variation der Lagrange Multiplikatoren λ_i

    = Variation der Kantenkosten c_e über Knotenbewertungen λ_i
  - Diese Variation der Kantenkosten hat keinen Einfluss auf die Optimalität einer Tour, verändern aber den 1-Baum

    denn:

    $\Sigma_{e=ij} (c_e + \lambda_i + \lambda_j)\, x_e - 2\,\Sigma_{i=1,\dots,n} \lambda_i = \Sigma_e c_e x_e$ wenn x eine Tour ist
  - Ist der minimale 1-Baum eine Tour, so ist diese Tour optimal für (P) nach Lemma 7.12, da $\lambda^T(b - Ax) = 0$ bei einer Tour
  - Einen minimalen 1-Baum findet man in polynomialer Zeit durch

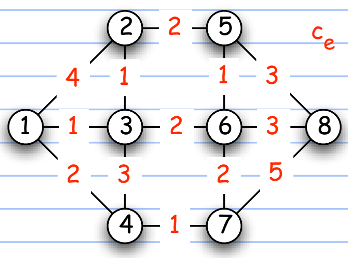
    (1) Ermittlung eines MST auf den Knoten 2, ..., n mit den Algorithmen aus ADM I (Kruskal oder Prim)

    (2) Wahl der beiden billigsten Kanten vom Knoten 1 aus

- Algorithmus zur Verbesserung der unteren Schranke (Variation der λ_i)
  - Input
    - Graph $G = (V, E)$ mit $V = \{ 1, ..., n \}$
      Kantenkosten c_e
  - Output
    - optimale Tour oder 1-Baum mit "guter" unterer Schranke $z(LR_\lambda)$
  - Methode
    - // Initialisierung der λ_i
      setze $\lambda_i := 0$ für jeden Knoten i
    - // Initialisierung einer Schrittweite $w > 0$ für die Variation der λ_i
      setze $w := 1$
    - repeat
      - ermittle minimalen 1-Baum x bzgl der Kantenkosten $c_{ij} + \lambda_i + \lambda_j$
      - if x ist Tour then return x // x ist optimale Tour
      - // Variation der λ_i
        for alle Knoten $i \neq 1$ do

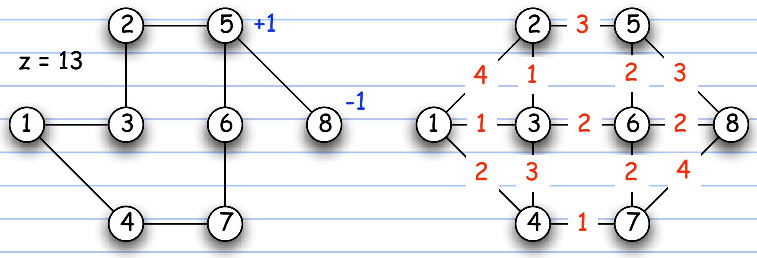
- bestimme den Grad d_i von Knoten i
  if $d_i \neq 2$ then $\lambda_i := \lambda_i + (d_i - 2)w$
- variiere ggf. die Schrittweite w
- until $z(LR_\lambda) = z(x)$ ist "gut" genug
- return bestes bisher gefundenes x und das zugehörige λ

- 7.14 Beispiel (1-Baum Relaxation des symmetrischen TSP)
  - Schrittweite w immer als 1 gewählt
  - Graph mit Kantenkosten

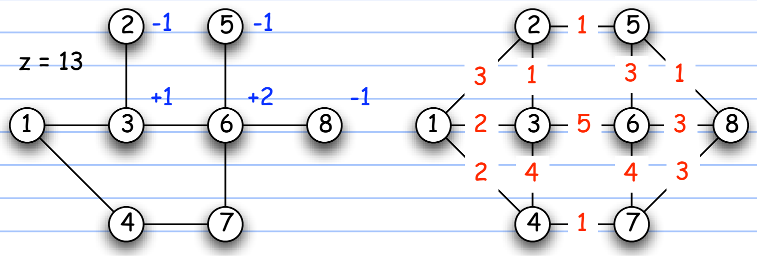


- Iteration 1

- minimaler 1 Baum, Variation der λ_i und neue Kantenkosten

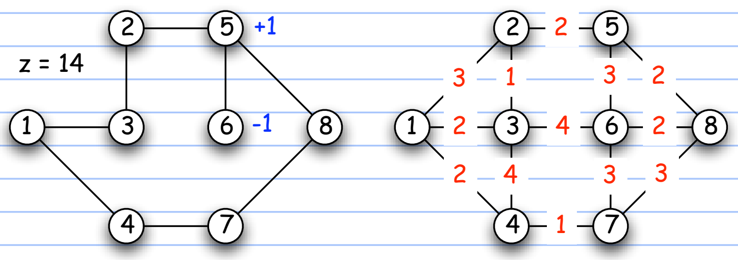


- Iteration 2
  - minimaler 1 Baum, Variation der λ_i und neue Kantenkosten

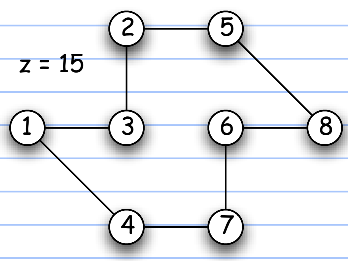


- Iteration 3

- minimaler 1 Baum, Variation der λ_i und neue Kantenkosten

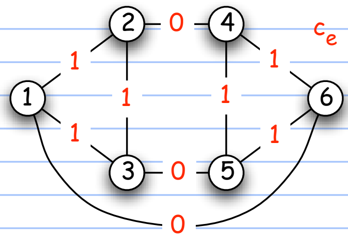


- Iteration 4
  - minimaler 1 Baum ist Tour => optimale Tour konstruiert

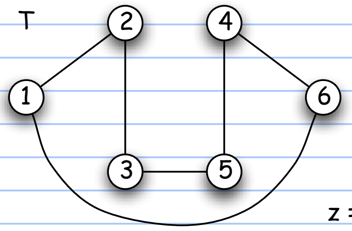


7.15 Beispiel (i.A. gibt es kein λ so dass der optimale 1-Baum eine Tour ist)

Graph mit Kantenkosten und optimale Tour



T

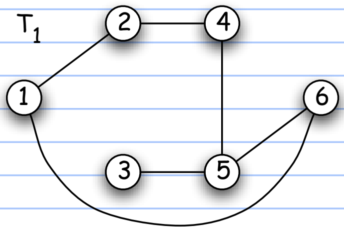


$z = 4$

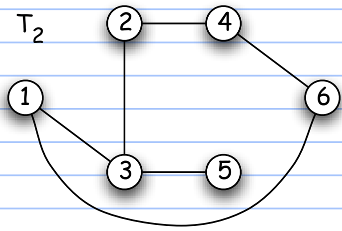
Claim: Für jede Wahl von λ_i (mit $\lambda_1 = 0$) ist keine Tour bzgl. $c_{ij} + \lambda_i + \lambda_j$ ein minimaler 1-Baum

Betrachte die 1-Bäume

T_1



T_2



Die Werte der beiden 1-Bäume bzgl. $c_{i,j} + \lambda_i + \lambda_j$ ergeben sich als

Wert von $T_1 = 3 + 2\lambda_2 + 1\lambda_3 + 2\lambda_4 + 3\lambda_5 + 2\lambda_6 =: z_1$

Wert von $T_2 = 3 + 2\lambda_2 + 3\lambda_3 + 2\lambda_4 + 1\lambda_5 + 2\lambda_6 =: z_2$

Der Wert einer optimalen Tour bzgl. $c_{ij} + \lambda_i + \lambda_j$ ergibt sich als

$4 + 2\lambda_2 + 2\lambda_3 + 2\lambda_4 + 2\lambda_5 + 2\lambda_6 =: z_0$

=> $z_0 - z_1 = 1 + \lambda_3 - \lambda_5$ und $z_0 - z_2 = 1 - \lambda_3 + \lambda_5$

=> entweder $z_0 > z_1$ oder $z_0 > z_2$

denn aus $z_0 < z_1$ und $z_0 < z_2$ folgt $1 + \lambda_3 - \lambda_5 < 0$ und $1 - \lambda_3 + \lambda_5 < 0$

=> $\lambda_3 - \lambda_5 > 1$ und $-\lambda_3 + \lambda_5 > 1$, Widerspruch ❑

Beachte: Der hier beim TSP beobachtete Fall $\max_\lambda L(\lambda) \neq z(P)$ ist der Regelfall. Die Lagrange-Relaxation liefert i.A. nur untere Schranken für $z(P)$, die allerdings oft gut in einen Branch & Bound Algorithmus einbettbar sind.

Zu weiteren Informationen über Lagrange Relaxationen des TSP siehe

E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys, eds.
The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization
John Wiley & Sons, New York, 1985.

Bestimmung von $\max_\lambda L(\lambda)$ mit Subgradientenverfahren

- $\max_\lambda L(\lambda) = \max_\lambda \min_x L(\lambda,x) = \max_\lambda \min\{ L(\lambda,x) \mid x \in S(LR_\lambda) \}$

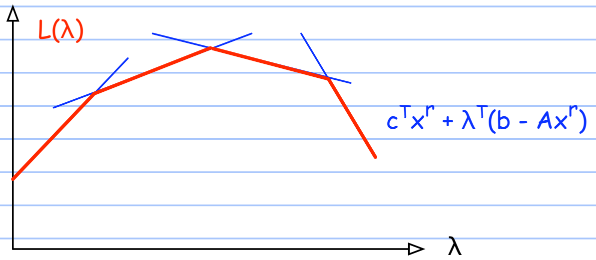
  Das Subgradientenverfahren nutzt die Tatsache, dass $S(LR_\lambda)$ wegen der Ganzzahligkeit von x in der Regel endlich ist. Dies nehmen wir im Folgenden an.

- $S(LR_\lambda)$ endlich => können $S(LR_\lambda)$ schreiben als $S(LR_\lambda) = \{ x^1, x^2, ..., x^R \}$

  => $L(\lambda) = \min\{ c^Tx^r + \lambda^T(b - Ax^r) \mid r = 1, ..., R \}$

  => $L(\lambda)$ ist Minimum von endlich vielen affin linearen Funktionen $c^Tx^r + \lambda^T(b - Ax^r)$ in λ

  => $L(\lambda)$ ist stückweise linear und konkav, aber i.A. nicht differenzierbar



- Subgradientenverfahren

  ~ Gradientenverfahren der Maximierung einer konkaven stetig differenzierbaren Funktion $f : \mathbb{R}^n \to \mathbb{R}^1$

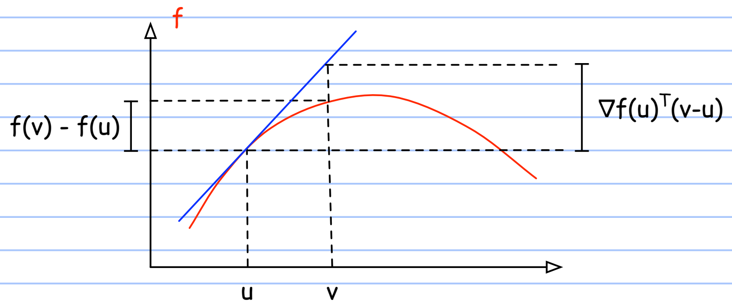
  - Gradient und Subgradient
    - Gradient einer stetig differenzierbaren Funktion in u
      - = Vektor der partiellen Ableitungen in u:

        $$\nabla f(u) = \left(\frac{\partial f}{\partial x_1}(u), \dots, \frac{\partial f}{\partial x_n}(u) \right)$$

        Dann ist aus der Analysis bekannt:

        f ist konkav <=>

        für alle v, u gilt $f(v) - f(u) \le \nabla f(u)^T(v-u)$



Subgradient einer stetigen konkaven Funktion in u

= Vektor d mit $f(v) - f(u) \le d^T(v-u)$ für alle v

Die Menge der Subgradienten in u heißt das Subdifferential von f in u und wird mit $\partial f(u)$ bezeichnet

Dann gilt: f differenzierbar in u => $\partial f(u) = \{ \nabla f(u) \}$

Bedingungen für Maximalpunkte einer konkaven Funktion

Der stetig differenzierbare Fall

Aus der Analysis ist bekannt:

λ^* ist Maximalpunkt von f <=> $\nabla f(\lambda^*) = 0$

Der nicht differenzierbare Fall

7.16 Lemma (Bedingung für Maximalpunkt einer stetigen konkaven Funktion)

Für eine stetige konkave Funktion $f : \mathbb{R}^n \to \mathbb{R}^1$ gilt

λ^* ist Maximalpunkt von f <=> $0 \in \partial f(\lambda^*)$

Beweis

"<="

sei $0 \in \partial f(\lambda^*)$

=> $0 = 0^T(v - \lambda^*) \ge f(v) - f(\lambda^*)$ für alle v => λ^* ist Maximalpunkt von f

"=>"

sei λ^* ist Maximalpunkt von f

=> $0 = 0^T(v - \lambda^*) \ge f(v) - f(\lambda^*)$ für alle v => $0 \in \partial f(\lambda^*)$ ❑

Generisches Subgradientenverfahren

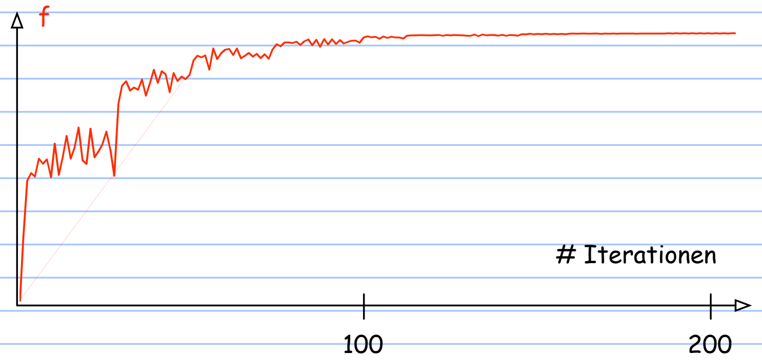
Input

stetige konkave Funktion $f : \mathbb{R}^n \to \mathbb{R}^1$

- Output
  - Maximalpunkt λ^* oder Punkt λ mit "gutem" Wert $f(\lambda)$
- Methode
  - wähle Startwert u_0
  - initialisiere Zähler $i := 0$
  - <u>repeat</u>
    - <u>if</u> $0 \in \partial f(u_i)$ <u>then</u> <u>return</u> u_i // u_i ist Maximalpunkt

      // hierauf kann auch verzichtet werden falls der Test "$0 \in \partial f(u_i)$" aufwändig ist
    - bestimme Subgradient $d_i \in \partial f(u_i)$ und Schrittweite $w_i > 0$
    - setze $u_{i+1} := u_i + w_i \cdot d_i$
    - $i := i+1$
  - <u>until</u> Rechenzeit zuende oder kaum noch Fortschritt
  - <u>return</u> besten Punkt der Folge $u_0, \dots, u_i$

- Beispiel eines typischen Laufes



Lauf zeigt, dass i.A. keine Monotonie gegeben ist

- Hauptingredienzen des Subgradientenverfahren
  - bestimme Schrittweite $w_i > 0$
    - theoretisch gelöst durch Satz von Polyak, aber praktisch dennoch schwierig, erfordert i.A. viele Experimente

7.17 Satz (Satz von Polyak 1967)

Sei $f : \mathbb{R}^n \to \mathbb{R}^1$ konkav und stetig und f nehme ihr Maximum im Punkt λ^* an.

Sei $(w_i)_{i \in N}$ eine Folge von Schrittweiten mit

(1) $w_i \geq 0$ für alle i

(2) $(w_i)_{i \in N}$ ist eine monoton fallende Nullfolge

(3) die Reihe $\sum w_i$ ist divergent

Dann gilt für die im Subgradientenverfahren erzeugte Folge von Punkten u_i

$\lim_{i \to \infty} f(u_i) = f(\lambda^*)$

ohne Beweis ❏

Dieser Satz sichert Konvergenz unter relativ schwachen Bedingungen, die bei konkreten Berechnungen leicht einzuhalten sind. Das Problem ist die Steuerung der Konvergenzgeschwindigkeit. Bei der Verwendung in B&B ist dies aber auch nicht so wesentlich.

bestimme Subgradient $d_i \in \partial f(u_i)$

dies ist einfacher, Subgradienten bekommt man bei der Lagrange Relaxation geschenkt

7.18 Lemma (Subgradienten bei der Lagrange Relaxation)

Sei x^* optimale Lösung von (LR_λ) in $\lambda = u$.

Dann ist $b - Ax^*$ Subgradient von $L(\lambda) = \min_x L(\lambda,x)$ in $\lambda = u$, d.h. $b - Ax^* \in \partial f(u)$.

Beweis durch Überprüfung der Definition

$L(v) - L(u) = \min_x L(v,x) - \min_x L(u,x)$

$= \min_x L(v,x) - L(u,x^*)$ da x^* optimal für (LR_u) ist

$\leq L(v,x^*) - L(u,x^*)$ da x^* zulässig für (LR_v) ist

$= (c^Tx^* + v^T(b - Ax^*)) - (c^Tx^* + u^T(b - Ax^*))$

$= (v^T - u^T)(b - Ax^*) = (b - Ax^*)^T(v - u)$ ❏

Bemerkung: Bei der 1-Baum Relaxation des symmetrischen TSP ergibt sich (nach Übergang von λ_i zu $-\lambda_i$) $x(\delta(i)) - 2$ als Subgradient. Die Veränderung der Multiplikatoren λ_i erweist sich damit als Spezialfall des Subgradientenverfahren.

Lagrange Relaxation vs. LP Relaxation

Es besteht eine Beziehung zwischen dem Optimalwert der Lagrange Relaxation und dem Wert der LP Relaxation eines IP.

Wir betrachten dazu:

- Das Ausgangsproblem

  (P) $\quad \min c^T x$

  unter $\quad Ax \geq b$

  $\qquad\qquad Bx \geq d$

  $\qquad\qquad x$ ganzzahlig

  keine Vorzeichenbedingungen an x, diese sind ggf. in die Nebenbedingungen integriert

- Die Lagrange Relaxation von (P)

  (LR_λ) $\quad \min c^T x + \lambda^T(b - Ax) = \min_x L(\lambda, x) = L(\lambda)$

  unter $\quad Bx \geq d$

  $\qquad\qquad x$ ganzzahlig

- Die LP Relaxation von (P)

  (LP) $\quad \min c^T x$

  unter $\quad Ax \geq b$

  $\qquad\qquad Bx \geq d$

  $\qquad\qquad x$ beliebig

  mit Optimalwert $z(LP)$

7.19 Satz (Beziehung zwischen Lagrange Relaxation und LP Relaxation)

- $\max_\lambda L(\lambda) \geq z(LP)$
- Gleichheit gilt falls das von $Bx \geq d$ erzeugte Polyeder ganzzahlig ist (also die Ganzzahligkeitsbedingung in (LR_λ) weggelassen werden kann).

Beweis

- wird hier geführt für Nebenbedingungen der Form $Ax \geq b$ (=> $\lambda \geq 0$), er folgt für Gleichheitsrestriktionen (λ beliebig) entsprechend.
- $$\max_{\lambda \geq 0} L(\lambda) = \max_{\lambda \geq 0} \min_{\substack{x \\ Bx \geq d \\ x \text{ gzz}}} L(\lambda,x) = \max_{\lambda \geq 0} \min_{\substack{x \\ Bx \geq d}} L(\lambda,x)$$

  $Bx \geq d$ erzeugt ganzzahliges Polyeder, sonst gilt hier $\geq$

$$= \max_{\lambda \geq 0} \min_{\substack{x \\ Bx \geq d}} (c^T x + \lambda^T(b - Ax))$$

$$= \max_{\lambda \geq 0} [\lambda^T b + \min_{\substack{x \\ Bx \geq d}} (c^T - \lambda^T A)x] = \max_{\lambda \geq 0} [\lambda^T b + \max_{\substack{y \geq 0 \\ B^T y = c - A^T\lambda}} d^T y]$$

LP Dualität

$$= \max_{\substack{\lambda \geq 0 \\ y \geq 0 \\ B^T y = c - A^T\lambda}} [b^T\lambda + d^T y] = \min_{\substack{x \text{ beliebig} \\ Ax \geq b \\ Bx \geq d}} c^T x$$

LP Dualität

$= z(LP)$ ❑

7.20 Bemerkung

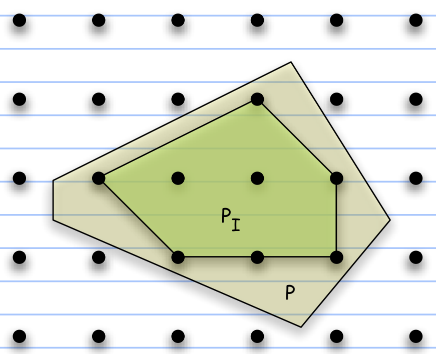
- Die 1-Baum Relaxation entspricht nach Satz 7.19 der LP-Relaxation des TSP-Polytopes.
- Da LPs im Prinzip in polynomialer Zeit lösbar sind (innere Punkte Methoden) scheint die LP-Relaxation vorzuziehen sein, falls $Bx \geq d$ ein ganzzahliges Polyeder definiert. Dennoch ist in der Praxis sehr oft das Subgradientenverfahren vorzuziehen, da es wesentlich schneller ist (oft kann $L(\lambda)$ kombinatorisch berechnet werden) und meist Näherungswerte für $\max_\lambda L(\lambda)$ reichen.

Hauptpunkte dieses Abschnittes

- Schnittebenenverfahren einführen als ein weiteres Verfahren um IPs exakt zu lösen.
- Nachweis, dass dies ein im Prinzip ein endliches Verfahren ist.
- Bei diesem Nachweis lernen wir einiges über ganzzahlige Polytope (Gomory-Chvátal-Schnitte, Chvátal Hülle)

Die ganzzahlige Hülle eines Polyeders

- Die ganzzahlige Hülle (integer hull) P_I eines Polyeders P ist die konvexe Hülle aller ganzzahligen Punkte in P.



- Ein Polyeder P heißt ganzzahlig, wenn alle Ecken ganzzahlig sind.

Dann gilt für Polytope P

(1) P ist ganzzahlig <=> $P = P_I$

(2) Ganzzahlige Optimierung über P <=> lineare Optimierung über P_I

Daher ist man an linearen Beschreibungen (= Beschreibung als Ungleichungssystem) von P_I interessiert

- Leider ist P_I im Allgemeinen kein Polyeder mehr!
  - Ein Beispiel ensteht für

$$P := \{(y,x) \in \mathbb{R}^2 \mid \frac{y}{x} \leq \sqrt{2}\}$$

(Übung)

- Man kann jedoch zeigen, dass P_I für rationale Polyeder P wieder ein Polyeder ist. Wir zeigen dies hier für rationale Polytope P.
  - Ein Polyeder $P = \{ x \in \mathbb{R}^n \mid Ax \leq b \}$ heißt rational, wenn alle Einträge von A und b rationale Zahlen sind. Im Weiteren werden alle Polyeder als rational vorausgesetzt, in den Sätzen wird es der Vollständigkeit halber immer als Voraussetzung aufgeführt.

Kriterien für die Existenz zulässiger Punkte und gültige Ungleichungen

Diese Kriterien sind alternative Formulierungen des Farkas Lemma (Lemma 4.5).

Eine Ungleichung $w^T x \le t$ heißt gültig für das Polyeder P, wenn alle Punkte $x \in P$ die Ungleichung erfüllen.

7.21 Lemma (Farkas Lemma für die Existenz zulässiger Lösungen)

Für ein Polyeder $P = \{ x \in \mathbb{R}^n \mid Ax \le b \}$ gilt:

(1) $P \ne \emptyset \iff y^T b \ge 0$ für alle $y \in \mathbb{R}^m$ mit $y \ge 0$ und $y^T A = 0$

(2) $P = \emptyset \iff$ es gibt $y \in \mathbb{R}^m, y \ge 0$ mit $y^T A = 0$ und $y^T b \le -1$

(3) $P = \emptyset \iff$ die Ungleichung $0^T x \le -1$ ergibt sich als nicht-negative Linearkombination der Ungleichungen in $Ax \le b$

Beweis

zu (1)

"=>"

Betrachte das LP $\max \{ 0^T x \mid Ax \le b \}$

$P \ne \emptyset \Rightarrow$ jedes $x \in P$ ist Optimallösung des LP

Dualitätssatz => das duale LP hat eine optimale Lösung und

$0 = \max \{ 0^T x \mid Ax \le b \} = \min \{ y^T b \mid y^T A = 0, y \ge 0 \}$

$\Rightarrow y^T b \ge 0$ für alle $y \ge 0$ mit $y^T A = 0$

"<="

Betrachte das LP $\min \{ y^T b \mid y^T A = 0, y \ge 0 \}$

0 ist zulässige Lösung dieses LP

Vorraussetzung => die Zielfunktion $y^T b$ ist nach unten durch 0 beschränkt

Dualitätssatz => P hat eine optimale Lösung, also insbesondere eine zulässige Lösung

zu (2)

durch Negation von (1) folgt

$P = \emptyset \iff$ es gibt $y' \in \mathbb{R}^m, y' \ge 0$ mit $(y')^T A = 0$ und $(y')^T b < 0$.

Sei $g := (y')^T b < 0$

Mit $y := y' / |g|$ folgt

$P = \emptyset \iff$ es gibt $y \in \mathbb{R}^m, y \ge 0$ mit $y^T A = 0$ und $y^T b \le -1$

zu (3)

"<="

klar

"=>"

nehme das y aus (2) und multipliziere $Ax \le b$ damit von links =>

$0^T x = y^T A x \le y^T b \le -1$ => $0^T x \le -1$ ❑

7.22 Lemma (Farkas Lemma für gültige Ungleichungen)

Für ein nichtleeres Polyeder $P = \{ x \in \mathbb{R}^n \mid Ax \le b \}$ sind folgende Aussagen äquivalent:

(1) $w^T x \le t$ ist eine gültige Ungleichung für P

(2) Es gibt $y \in \mathbb{R}^m, y \ge 0$ mit $y^T A = w^T$ und $y^T b \le t$

Beweis

(1) => (2)

Betrachte das LP $\max \{ w^T x \mid Ax \le b \}$

$P \neq \emptyset$, $w^T x \le t$ => das LP hat eine optimale Lösung

Dualitätssatz => das duale LP hat eine optimale Lösung y^* und

$t \ge \max \{ w^T x \mid Ax \le b \} = \min \{ y^T b \mid y^T A = w^T, y \ge 0 \} = (y^*)^T b$

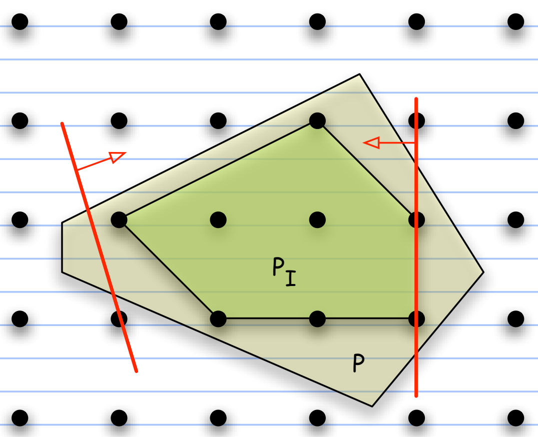
=> y^* erfüllt (2)

(2) => (1)

$Ax \le b,\ y \ge 0$ => $w^T x = (y^T A)x = y^T (Ax) \le y^T b \le t$ ❑

Schnittebenen (cutting planes) und Schnittebenenverfahren

Idee: Schneide durch Hyperebenen Teile eines Polyeders P ab, aber ohne Punkte aus P_I abzuschneiden, d.h. sie sind gültig für P_I (und vielleicht sogar Stützhyperebenen oder Facetten). Eine solche Hyperebene heißt Schnittebene (cutting plane). Eine Schnittebene H, die einen Punkt $x^* \in P - P_I$ abschneidet, heißt x^* separierende Hyperebene.



Schnittebenenverfahren (Idee)

Input

Ganzahliges lineares Programm (IP)

$\min \{ c^T x,\ x \in P_I \}$ mit P = Polyeder der LP Relaxation von (IP)

Output

Optimallösung von (IP)

Methode

<u>repeat</u> <u>forever</u>

löse das LP $\min \{ c^T x,\ x \in P \}$

sei x^* die Optimallösung des LP

<u>if</u> x^* ist ganzzahlig <u>then</u> return x^* // x^* ist Optimallösung von (IP)

berechne Schnittebene, die x^* von P abschneidet und gültig für P_I ist // separating hyperplane

sei H der zugehörige Halbraum, der P_I enthält

setze $P := P \cap H$

Offensichtliche Fragen

(1) wie beweist man, dass eine Ungleichung eine Schnittebene ist?

(2) terminiert das Schnittebenenverfahren überhaupt?

(3) wie berechnet man zu gegebenem $x^* \in P - P_I$ eine x^* separierende Hyperebene?

Hier werden wir vor allem (1) und (2) behandeln und insofern positiv beantworten, als dass man Beweise für (1) angeben kann und es immer eine endliche Menge von Schnittebenen einer speziellen Gestalt gibt, so dass am Ende $P = P_I$ gilt.

(3) hängt sehr vom jeweiligen Problem ab, dazu mehr in Kapitel 8.

Schnittebenenbeweise

Für Polytope $P = \{ x \in \mathbb{R}^n \mid Ax \le b \}$ kann ein Beweis für die Gültigkeit einer Ungleichung $w^T x \le t$ durch das Farkas Lemma (Lemma 7.21) gegeben werden. Für Schnittebenen ist es komplizierter.

7.23 Beispiel (Beispiel eines Schnittebenenbeweises)

Betrachte das Ungleichungssystem

$2x_1 + 3x_2 \le 27$ (1)

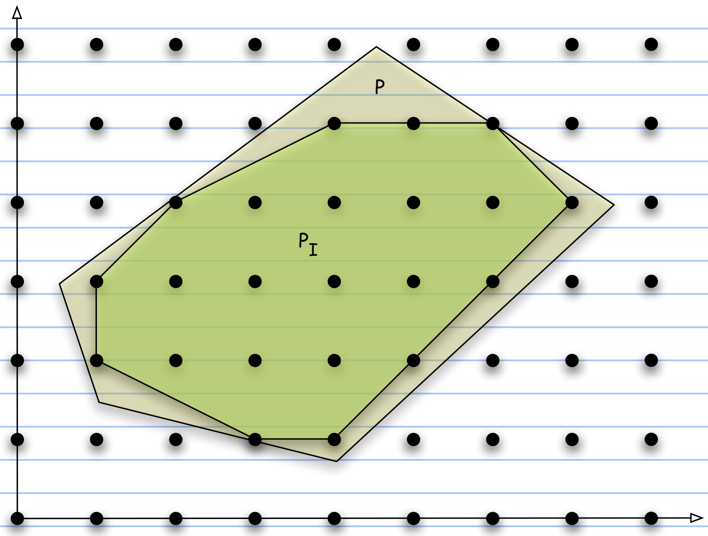
$2x_1 - 2x_2 \le 7$ (2)

$-6x_1 - 2x_2 \leq -9$ (3)

$-2x_1 - 6x_2 \leq -11$ (4)

$-6x_1 + 8x_2 \leq 21$ (5)

- Das zugehörige Polytop P und seine ganzzahlige Hülle



- $x_2 \leq 5$ ist eine gültige Ungleichung für P_I

Wie kann man das aus den Ungleichungen für P ableiten?

- Multipliziere (5) mit 1/2

  => $-3x_1 + 4x_2 \leq 21/2$

  => $-3x_1 + 4x_2 \leq \lfloor 21/2 \rfloor = 10$ ist gültig für P_I (da links nur ganzzahlige Koeffizienten)

  => neue Ungleichung $-3x_1 + 4x_2 \leq 10$ (6) für P_I

- Multpliziere (6) mit 2, (1) mit 3 und addiere die resultierenden Ungleichungen

  => $-6x_1 + 8x_2 \leq 20$

  $6x_1 + 9x_2 \leq 81$

  => $17x_2 \leq 101$

  => gewünschte Ungleichung $x_2 \leq \lfloor 101/17 \rfloor = 5$

- Allgemein haben diese Unleichungen die Form

  $y^TAx \leq \lfloor y^Tb \rfloor$ mit $y \geq 0$ und y^TA ganzzahlig

  wobei $Ax \leq b$ das System der Ungleichungen nach dem "vorigen" Schritt ist.

  Aus dieser Beobachtung leitet sich die allgemeine Definition eines Schnittebenenbeweises ab.

- Allgemeine Definition

Sei $Ax \le b$ ein Ungleichungssystem mit m Ungleichungen.

Ein Schnittebenenbeweis für die Ungleichung $w^T x \le t$ mit ganzzahligem w und t ausgehend von $Ax \le b$ ist eine endliche Folge von Ungleichungen der Form

$$a_{m+k}^T x \le b_{m+k} \quad (k = 1, \ldots, M)$$

zusammen mit nichtnegativen Zahlen

$$y_{k_j} \quad (1 \le k \le M,\ 1 \le j \le m+k-1)$$

so dass für jedes $k = 1, \ldots, M$ die Ungleichung

$$a_{m+k}^T x \le b_{m+k}$$

sich als nichtnegative Linearkombination

$$(y_{k_1}, \ldots, y_{k_{m+k-1}}) \begin{pmatrix} a_{11}x_1 + \ldots + a_{1n}x_n \\ \vdots \\ a_{m+k-1,1}x_1 + \ldots + a_{m+k-1,n}x_n \end{pmatrix} \le \left\lfloor (y_{k_1}, \ldots, y_{k_{m+k-1}}) \begin{pmatrix} b_1 \\ \vdots \\ b_{m+k-1} \end{pmatrix} \right\rfloor$$

der bereits vorhandenen Ungleichungen ergibt, wobei

- die Koeffizienten der Variablen auf der linken Seite ganzzahlig sind
- die rechte Seite nicht ganzzahlig ist und nach unten gerundet wird, und
- die letzte Ungleichung der Folge die Ungleichung $w^T x \le t$ ist

Ungleichungen dieser Form

$$y^T A x \le \lfloor y^T b \rfloor \quad \text{mit } y \ge 0 \text{ und } y^T A \text{ ganzzahlig}$$

heißen Gomory-Chvátal Schnitte.

Gomory hat 1960 gezeigt, dass man mit diesen Schnitten ein endliches Schnittebenenverfahren erhält.

Chvátal hat 1973 das hier behandelte Prinzip der Schnittebenenbeweise eingeführt. Diese Beweise ähneln dem Farkas Lemma in den Varianten Lemma 7.21 (2) und 7.22.

7.24 Satz (Schnittebenenbeweise für rationale Polytope, Chvátal 1973)

Sei $P = \{ x \in \mathbb{R}^n \mid Ax \le b \}$ ein rationales Polytop und sei $w^T x \le t$ eine Ungleichung mit ganzzahligem w und t, die von allen Punkten aus P_I erfüllt wird. Dann gibt es einen Schnittebenenbeweis von $Ax \le b$ für eine Schnittebene $w^T x \le t'$ mit $t' \le t$.

Beweis siehe unten ❑

7.25 Satz (Schnittebenenbeweise für rationale Polytope ohne ganzzahlige Punkte, Chvátal 1973)

Sei $P = \{ x \in \mathbb{R}^n \mid Ax \le b \}$ ein rationales Polytop ohne ganzzahlige Punkte. Dann existiert ein Schnittebenenbeweis von $Ax \le b$ für die Ungleichung $0^T x \le -1$.

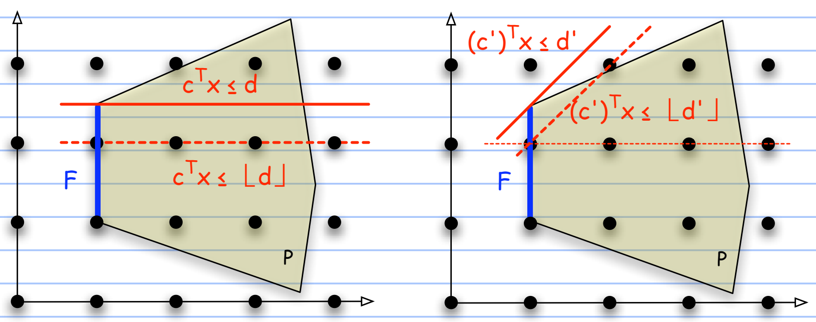
- Beweis siehe unten ❑

- Für die Beweise benötigen wir ein Lemma, das eine Induktion über die Dimension von P möglich macht. Es zeigt, dass Gomory-Chvátal Schnitte von Seitenflächen eines rationalen Polyeders auf das Polyeder selbst "durch Rotation" geliftet werden können.

7.26 Lemma (Rotation von Gomory-Chvátal Schnitten)

- Sei F eine durch ein Gleichungssystem beschriebene Seitenfläche eines rationalen Polytops P und sei $c^Tx \le \lfloor d \rfloor$ ein Gomory-Chvátal Schnitt für F.
- Dann existiert ein Gomory-Chvátal Schnitt $(c')^Tx \le \lfloor d' \rfloor$ für P mit

  $F \cap \{ x \mid c^Tx \le \lfloor d \rfloor \} = F \cap \{ x \mid (c')^Tx \le \lfloor d' \rfloor \}$ (Gleichheit auf F)



Beweis

- Sei o.B.d.A. $P = \{ x \mid A'x \le b',\ A''x \le b'' \}$ mit A'', b'' ganzzahlig.

  Sei $F := \{ x \mid A'x \le b',\ A''x = b'' \}$ (Gleichungen $A''x = b''$ beschreiben F)

  Sei $c^Tx \le \lfloor d \rfloor$ der Schnitt für F laut Voraussetzung

  und o.B.d.A. $d = \max \{ c^Tx \mid x \in F \}$ (schärfster Schnitt mit c^Tx; existiert da P ein Polytop).

  Dualitätzsatz => das duale LP hat eine Optimallösung

  => es gibt $y' \ge 0$ und y'' beliebig mit

  $(y')^TA' + (y'')^TA'' = c^T$ (*)

  $(y')^Tb' + (y'')^Tb'' = d$ (**)

- Konstruktion von c' und d' aus y''

  $(c')^T := c^T - (\lfloor y'' \rfloor)^T A'' = (y')^T A' + (y'' - \lfloor y'' \rfloor)^T A''$ wegen (*)

  ganzzahlig ≥ 0 ≥ 0

  $d' := d - (\lfloor y'' \rfloor)^T b'' = (y')^T b' + (y'' - \lfloor y'' \rfloor)^T b''$ wegen (**)

  c ganzzahlig als Gomory-Chvátal Schnitt, $(\lfloor y'' \rfloor)^T A''$ ganzzahlig => c' ganzzahlig

- $(c')^T x \leq d'$ ist gültige Ungleichung für P

  denn $(c')^T x = (y')^T A' x + (y'' - \lfloor y'' \rfloor)^T A'' x \leq (y')^T b' + (y'' - \lfloor y'' \rfloor)^T b'' = d'$

  ≥ 0 ≥ 0

- Definition von d' => $d = d' + (\lfloor y'' \rfloor)^T b''$

  ganzzahlig, da b'' ganzzahlig

  => $\lfloor d \rfloor = \lfloor d' \rfloor + (\lfloor y'' \rfloor)^T b''$

- Gleichheit auf F

  $F \cap \{ x \mid (c')^T x \leq \lfloor d' \rfloor \}$

  $= F \cap \{ x \mid (c')^T x \leq \lfloor d' \rfloor, (\lfloor y'' \rfloor)^T A'' x = (\lfloor y'' \rfloor)^T b'' \}$

  erfüllt in F wegen $A'' x = b''$

  $= F \cap \{ x \mid (c' + (\lfloor y'' \rfloor)^T A'')^T x \leq (d' + \lfloor y'' \rfloor)^T b'' \}$

  $= F \cap \{ x \mid c^T x \leq \lfloor d \rfloor \}$ ❑

- Beweis von Satz 7.25 (Schnittebenenbeweise für rationale Polytope ohne ganzzahlige Punkte)
  - Induktion nach dim(P)
    - Induktionsanfang
      - $P = \emptyset$
        - => Behauptung mit Farkas Lemma 7.21 (3)
      - $\dim(P) = 0$
        - => $P = \{x^*\}$ und x^* ist nicht ganzzahlig.

          => (Induktion nach n) es gibt gzz. Vektor w mit $w^T x^*$ nicht ganzzahlig

          Sei t so, dass die Hyperebene $H = \{ x \mid w^T x = t \}$ durch x^* geht (durch Verschiebung immer erreichbar).

          $w^T x^*$ nicht ganzzahlig => $t = w^T x^*$ nicht ganzzahlig

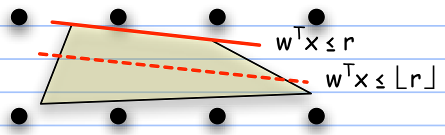
          => $w^T x \leq t$ ist gültig für P, aber $P' := P \cap \{ x \mid w^T x \leq \lfloor t \rfloor \} = \emptyset$

          Farkas Lemma 7.22 => es gibt Schnittebenenbeweis für $w^T x \leq \lfloor t \rfloor$ von $Ax \leq b$

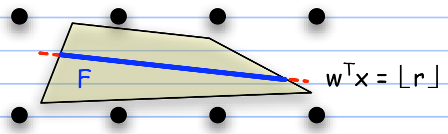
          $P' = \emptyset$ => (Farkas Lemma 7.21 (3)) es gibt Schnittebenenbeweis für $0^T x \leq -1$ von $Ax \leq b$ and $w^T x$

$\leq \lfloor t \rfloor$

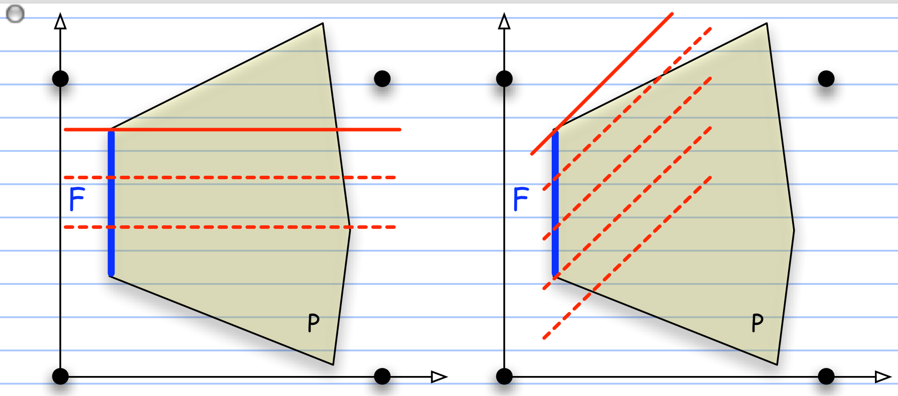
- Induktionsschluss auf $\dim(P) \geq 1$
  - Sei $w^T x \leq r$, w ganzzahlig, eine Ungleichung, die eine echte Seitenfläche von P erzeugt.
    Sei $P' := \{ x \in P \mid w^T x \leq \lfloor r \rfloor \}$



  - Fall 1: $P' = \emptyset$
    - Farkas Lemma 7.22 => können $w^T x \leq r$ aus $Ax \leq b$ herleiten
      Farkas Lemma 7.21 => können $0^T x \leq -1$ aus $Ax \leq b, w^T x \leq \lfloor r \rfloor$ herleiten
  - Fall 2: $P' \neq \emptyset$
    - Sei $F := \{ x \in P' \mid w^T x = \lfloor r \rfloor \}$ => F ist Seitenfläche von P'



- Claim: $\dim(F) < \dim(P)$
  - denn:
    entweder F ist echte Seitenfläche von P (wenn r ganzzahlig)
    oder $w^T x \leq \lfloor r \rfloor$ schneidet von P etwas ab, dann enthält P Punkte, die nicht $w^T x = \lfloor r \rfloor$ erfüllen
    => niedrigere Dimension in beiden Fällen
- F_I leer, Induktionsvoraussetzung
  => es gibt einen Schnittebenenbeweis für $0^T x \leq -1$ von $Ax \leq b, w^T x \leq \lfloor r \rfloor, -w^T x \leq -\lfloor r \rfloor$
- Neue erhaltene Schnittebene $0^T x \leq -1$ bzgl. F mehrfach nutzen und jedesmal Rotationslemma anwenden
  ergibt folgendes Bild



P Polytop => $\min\{ w^T x \mid x \in P \}$ ist endlich =>

Wiederholung des Arguments mit $P'' := \{ x \in P \mid w^T x \le \lfloor r \rfloor - 1 \}$ usw. ergibt nach endlich vielen Schritten einen Schnittebenenbeweis für eine Ungleichung $w^T x \le t$ mit $\{ x \in P \mid w^T x \le t \} = \emptyset$

=> Reduktion auf Fall 1 ❑

Beweis von Satz 7.24 (Schnittebenenbeweise für rationale Polytope)

Fall 1: $P_I = \emptyset$

Satz 7.25 => $0^T x \le -1$ ist beweisbar

P Polytop => $r := \max\{ w^T x \mid x \in P \}$ ist endlich

=> $w^T x \le r$ ist gültige Ungleichung für P

Farkas Lemma 7.22 => $w^T x \le r$ ist beweisbar

w ganzzahlig => $w^T x \le \lfloor r \rfloor$ ist Gomory-Chvátal Schnitt

Addition von $w^T x \le \lfloor r \rfloor$ und $0^T x \le -1$ ergibt $w^T x \le \lfloor r \rfloor -1$

Fortgesetzte Addition von $0^T x \le -1$ ergibt $w^T x \le t' \le t$ in endlich vielen Schritten

Fall 2: $P_I \ne \emptyset$

P Polytop => $r := \max\{ w^T x \mid x \in P \}$ ist endlich

Sei $P' := \{ x \in P \mid w^T x \le \lfloor r \rfloor \}$

Falls $\lfloor r \rfloor \le t$ => fertig

Sei also $\lfloor r \rfloor > t$

Sei $F := \{ x \in P' \mid w^T x = \lfloor r \rfloor \}$ => F ist Seitenfläche von P'

F enthält keine ganzzahligen Punkte, da $w^T x \le t$ gültig für P_I und $t < \lfloor r \rfloor$ ist

Theorem 7.25 => für F gibt es einen Schnittebenenbeweis von $0^T x \le -1$ ausgehend von $Ax \le b$, $w^T x = \lfloor r \rfloor$

Rotations Lemma für F and P' => es gibt einen Schnittebenenbeweis $c^T x \le \lfloor d \rfloor$ für P ausgehend von $Ax \le b$, $w^T x \le \lfloor r \rfloor$ so dass

$F \cap \{ x \mid c^T x \leq \lfloor d \rfloor, w^T x \leq \lfloor r \rfloor \} = F \cap \{ x \mid 0^T x \leq -1, w^T x \leq \lfloor r \rfloor \}$

Hier haben wir die spezielle Ungleichung $0^T x \leq -1$ on F.

Die Konstruktion von c and d im Beweis des Rotationslemma wegibt hier $c = w$ and $d = r - 1$

Also gibt es einen Schnittebenenbeweis von $w^T x = \lfloor r \rfloor - 1$ auf P ausgehend von $Ax \leq b, w^T x \leq \lfloor r \rfloor$

Wiederholung des Arguments ergibt irgendwann $w^T x \leq t' \leq t$ und führt zu Fall 1 ❑

Chvátal Hülle und Chvátal Rank

Schnittebenenbeweise benutzen bereits erzeugte Schnittebenen im Beweis. Wir betrachten jetzt, was geschieht, wenn man nur die ursprünglich gegebenen Schnittebenen $Ax \leq b$ benutzen darf

Sei $P = \{ x \in \mathbb{R}^n \mid Ax \leq b \}$ ein rationales Polytop. Fügt man alle Gomory-Chvátal Schnitte $y^T Ax \leq \lfloor y^T b \rfloor$ mit $y \geq 0$, $y^T A$ ganzzahlig zu P hinzu, so erhält man die Chvatal Hülle P' von P.

7.27 (Eigenschaften der Chvatal Hülle)

Die Chvátal Hülle eines rationalen Polytopes ist wieder ein rationales Polytop. Insbesondere braucht man zu seiner linearen Beschreibung nur $Ax \leq b$ und endlich viele der Gomory-Chvátal Schnitte.

Beweis

Sei $P = \{ x \mid Ax \leq b \}$ mit A und b ganzzahlig

Setze $P' := P \cap \{ x \mid y^T Ax \leq \lfloor y^T b \rfloor$ mit $y \geq 0$, $y^T A$ ganzzahlig $\}$. Es gilt:

(7.13) $P' = P \cap \{ x \mid y^T Ax \leq \lfloor y^T b \rfloor$ mit $y \geq 0$, $y^T A$ ganzzahlig, $0 \leq y < 1 \}$

Beweis (7.13):

Sei $w^T x \leq \lfloor t \rfloor$ ein Gomory-Chvátal Schnitt mit $y \geq 0$, $y^T A = w$, $y^T b = t$

Sei $y' := y - \lfloor y \rfloor$ der fraktionale Teil von y und $0 \leq y' < 1$

Sei $w' := (y')^T A = y^T A - (\lfloor y \rfloor)^T A = w - (\lfloor y \rfloor)^T A$

=> w' ist ganzzahlig, da w und A ganzzahlig sind

Sei $t' := (y')^T b = y^T b - (\lfloor y \rfloor)^T b = t - (\lfloor y \rfloor)^T b$

=> t und t' unterscheiden sich um eine ganze Zahl, nämlich $(\lfloor y \rfloor)^T b$

=> $w^T x \leq \lfloor t \rfloor$ ergibt sich als Summe aus

$(w')^T x \leq \lfloor t' \rfloor$ <--- gemäß (7.13) gebildet

$+ (\lfloor y \rfloor)^T Ax \leq (\lfloor y \rfloor)^T b$ <--- redundant, da nichtnegative Linearkombination der Zeilen von $Ax \leq b$

=> die Ungleichungen gemäß (7.13) reichen zur Bildung der Hülle

Es gibt nur endlich viele Ungleichungen gemäß (7.13)

Seien a_{ij} die Einträge der Matrix A und sei A_j die j-te Spalte von A

=> $y^T A_j \in [-\Sigma_i |a_{ij}|, \Sigma_i |a_{ij}|]$ und ganzzahlig für $0 \le y < 1$

=> es gibt nur endlich viele $y^T A_j$, die dies leisten

Alle Ungleichungen der Form (7.13) haben ganzzahlige Koeffizienten

=> sie sind wieder rational ❑

Die Chvátal Hüllenbildung lässt sich natürlich iterieren und liefert eine Folge

$$P = P^{(0)} \supseteq P^{(1)} \supseteq P^{(2)} \supseteq \ldots \supseteq P_I$$

7.28 Satz (Die Hüllenbildung ist endlich)

Sei P ein rationales Polytop. Dann existiert ein $k \in \mathbb{N}$ mit $P_I = P^{(k)}$.

Insbesondere terminiert das Schnittebenenverfahren bei geigneter Auswahl von Gomory Chvátal Schnitten nach endlich vielen Schritten.

Beweis

P_I ist ein Polytop und damit durch endlich viele Ungleichungen beschreibbar.

Für jede dieser Ungleichungen existiert ein Schnittebenenbeweis einer endlichen Länge r,

also mit Ungleichungen nur aus endlich vielen der $P^{(i)}$

=> das Maximum dieser r erfüllt die Behauptung ❑

Gomory hat 1960 eine solche geeignete Auswahl beschrieben.

Das kleinste k mit $P_I = P^{(k)}$ heißt der Chvátal Rang von P. Dieser bildet eine Art Komplexitätsmaß für die ganzzahlige Hülle von Polytopen. Er kann bereits für Polytope im $\mathbb{R}^2$ beliebig groß werden, ist jedoch für Polytope im 0/1-Würfel im $\mathbb{R}^n$ durch $6n^3 \log n$ beschränkt.

7.29 Beispiel (Ein Polytop mit Chvátal Rang 2)

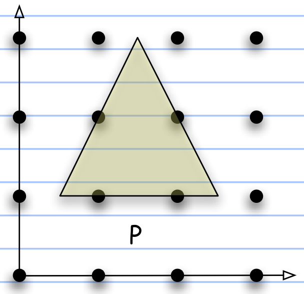
Das Ausgangspolytop P

P sei gegeben durch

$$-2x_1 + x_2 \le 0 \quad (1)$$

$$2x_1 + x_2 \le 6 \quad (2)$$

$$-x_2 \le -1 \quad (3)$$



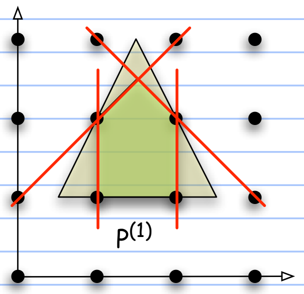
Die erste Chvatal Hülle $P^{(1)}$

$y^T = (0, 1/2, 1/2)$ ergibt $x_1 \leq 5/2 \Rightarrow x_1 \leq 2$

$y^T = (1/2, 0, 1/2)$ ergibt $-x_1 \leq -1/2 \Rightarrow -x_1 \leq -1$

$y^T = (5/6, 1/3, 1/6)$ ergibt $-x_1 + x_2 \leq 11/6 \Rightarrow -x_1 + x_2 \leq 1$

$y^T = (1/3, 5/6, 1/6)$ ergibt $x_1 + x_2 \leq 29/6 \Rightarrow x_1 + x_2 \leq 4$



$x_2 \leq 2$ ist nicht aus $Ax \leq b$ ableitbar

allgemeiner Gomory-Chvátal Schnitt ist

$(-2y_1 + 2y_2)x_1 + (y_1 + y_2 - y_3)x_2 \leq \lfloor 6y_2 - y_3 \rfloor$

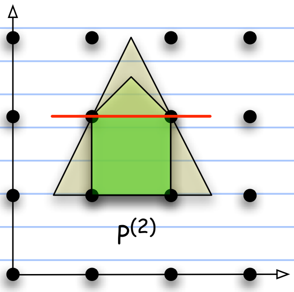
$\Rightarrow -2y_1 + 2y_2 = 0,\ y_1 + y_2 - y_3 = 1,\ \lfloor 6y_2 - y_3 \rfloor = 2$

$\Rightarrow y_1 = y_2,\ y_3 = 2y_2 - 1,\ \lfloor 4y_2 + 1 \rfloor = 2 \Leftrightarrow 1 \leq 4y_2 < 2$

$\Rightarrow y_2 < 1/2 \Rightarrow y_3 < 0 \Rightarrow$ Widerspruch

Die zweite Chvatal Hülle $P^{(2)}$

$x_2 \leq 2$ ist ableitbar aus $-x_1 + x_2 \leq 1,\ x_1 + x_2 \leq 4$ mit $y^T = (1/2, 1/2)$



Abschließende Bemerkungen

- Alle hier getroffenen Ausagen über Schnittebenenbeweise und Chvátal Rang gelten auch für beliebige rationale Polyeder (vgl. Korte & Vygen), aber i.A. nicht mehr für nicht-rationale Polyeder.
- Gomory-Chvátal Schnitte sind ein Standardwerzeug in CPLEX. Eine Auswahl wird bei ganzzahligen Programmen automatisch erzeugt und bereits der LP-Relaxation hinzugefügt. Im Laufe des Branch & Bound Algorithmus werden dann weitere speziell für die Teilprobleme in Knoten des Branch & Bound Baumes erzeugt.
- Für viele kombinatorische Optimierungsprobleme gibt es Aussagen zum Chvátal Rang bestimmter Ungleichungen:
  - Die Odd-Set Ungleichungen des Matching Polytops haben Chvátal Rang 1 bzgl. der LP Formulierung, die nur die Gradbedingungen enthält.

$x(\delta(v)) = 1$ für alle $v \in V$ Gradbedingungen

$\Sigma_{e \subseteq R}\ x_e \leq r$ für alle Mengen $R \subseteq V(G)$ mit $|R| = 2r+1$ Odd-Set Ungleichungen

$x \geq 0$

  - Die Kamm-Ungleichungen für das TSP haben Chvátal Rang 1 bzgl. des 2-Matching Polytops (siehe Abschnitt 8.2)

- Separierung ist das Problem, zu einem gegebenem Punkt x^* und einem Polyeder Q eine Hyperebene H zu ...
- Optimierung (OPT)
  - Input:
    - rationales Polyeder Q,
      $c \in \mathbb{R}^n$ so dass $c^T x$ ist auf P nach unten beschränkt ist
  - Output:
    - $x^* \in Q$ mit $x^* = \min \{ c^T x \mid x \in Q \}$
- Separierung (SEP)
  - Input:
    - rationales Polyeder Q,
      $y \in \mathbb{R}^n$
  - Output:
    - "Ja" falls $y \in Q$
      $d \in \mathbb{R}^n$ mit $d^T x < d^T y$ für alle $x \in Q$ falls $y \notin Q$ (eine separierende Hyperebene)

- 7.30 Satz (Polynomiale Äquivalenz von Separierung und Optimierung; Grötschel, Lovasz, Schrijver 1984)

  - (OPT) polynomial lösbar <=> (SEP) polynomial lösbar
    Dies gilt auch für ε-Approximationen
  - ohne Beweis,
    bei volldimensionalen Polyedern werden folgende Techniken benutzt
    "<=" Ellipsoidmethode und Dualitätssatz
    "=>" Antiblocking von Polyedern
    Details siehe
    M. Grötschel, L. Lovász, and A. Schrijver,
    Geometric Algorithms and Combinatorial Optimization,
    Springer-Verlag, Berlin, 2nd ed., 1993. ❏

- Bemerkungen
  - Ist (OPT) polynomial lösbar, so sind Schnittebenen effizient ermittelbar
  - Ist (OPT) NP-schwer, so wird man (falls P ≠ NP) nicht alle Schnittebenen in polynomialer Zeit finden, aber unter Umständen doch noch viele.
  - Daher verwendet man polynomiale Algorithmen zur Ermittlung von Schnittebenen, bis diese keine mehr finden

und brancht dann nach fraktionalen Variablen. Für die entstehenden Unterprobleme sucht man wieder Schnittebenen bis man brachen muss usw.

Diese Kombination von Branch & Bound mit Schnittebenenverfahren nennt man Branch & Cut. Beispiele siehe Kapitel 8.

Statt der Ellipsoidmethode (die sich als ineffizient für die Praxis erwiesen hat) verwendet man üblicherweise den dualen Simplexalgorithmus, der es einfach gestattet, neue Schnittebenen als zusätzliche Restriktionen einzubauen.