PD Dr. Christoph Buchheim[1]
Dr. Frauke Liers[1]
Dr. Diana Fanghänel[1]

Sommersemester 2008

Vorlesungsskript (2. Teil)

Robuste Optimierung

In der Optimierung bestimmt man im Allgemeinen optimale Lösungen in der Annahme, dass die Eingabedaten exakt bekannt sind. In der Praxis ist dies jedoch häufig nicht der Fall, da Messfehler, Rundungsfehler oder andere Unsicherheiten in den Problemdaten auftreten können. So kann es sein, dass eine Problemlösung bei etwas geänderten Eingabedaten unbrauchbar wird, da sie sehr weit von der realen Lösung entfernt liegt oder gar für das Problem unzulässig ist.

Einen Ausweg aus dieser Situation bietet die robuste Optimierung. Hier werden zusätzlich zu den Eingabedaten Toleranzen angegeben. Gesucht wird eine möglichst gute Lösung, welche für alle innerhalb der Toleranzen liegenden Eingabewerte zulässig ist.

Das vorliegende Vorlesungsskript wurde anhand des 2. Teils der Vorlesung „Robuste Optimierung“ erstellt, welcher Mai bis Juni 2008 an der Universität zu Köln gelesen wurde. In diesem Teil wird die diskrete robuste Optimierung behandelt. Dabei wird kein Anspruch auf Vollständigkeit oder vollständige Übereinstimmung mit der Vorlesung erhoben.

Inhaltsverzeichnis

[1] {buchheim|liers|fanghaenel}@informatik.uni-koeln.de
Hinweise bezüglich der Vorlesung und der Übungen finden Sie unter :
http://www.informatik.uni-koeln.de/ls_juenger/teaching/ss_08/

4 Robuste ganzzahlige Optimierung

Bisher waren alle (nominalen) Probleme lineare Programme. Im folgenden betrachten wir Probleme mit ganzzahligen Variablen:

$$(MIP) \qquad \begin{array}{ll} \min & c^\top x \\ s.d. & Ax \leq b \\ & x_j \in \mathbb{Z},\ j = 1, \ldots, k \\ & x \in \mathbb{R}^n. \end{array}$$

Dies ist ein gemischt ganzzahliges Programm.

4.1 Box-Unsicherheiten

Angenommen, es existieren Unsicherheiten in c und A wieder als Perturbationen in einer Box, d.h.

- Jeder perturbierte Koeffizient c_j ist eine symmetrische beschränkte Zufallsvariable in $[c_j - \Delta c_j; c_j + \Delta c_j]$ mit $\Delta c_j \geq 0$.
- Jeder Eintrag a_{ij} ist eine symmetrische beschränkte Zufallsvariable im Intervall $[a_{ij} - \Delta a_{ij}; a_{ij} + \Delta a_{ij}]$ mit $\Delta a_{ij} \geq 0$.

Wir untersuchen folgenden **flexiblen Ansatz:**
Wir sichern gegen Γ_i Abweichungen in der Zeile i ab mit $i = 1, \ldots, m$ und $\Gamma_i \in \{0, \ldots, n\}$. Dabei bezeichnet der Index $i = 0$ die Zielfunktion.

Es ist also eine Lösung gesucht, die robust ist gegen Veränderung von maximal Γ_i Einträgen in der Zeile i.

Dabei bedeuten:
Große Γ_i: Mehr Sicherheit, aber schlechtere Optima.
Kleine Γ_i: Weniger Sicherheit, aber bessere Optima.

Die robuste Formulierung von (MIP) ist dann also

$$(RMIP) \qquad \begin{array}{ll} \min & c^\top x + \max\limits_{S_0 : |S_0| \leq \Gamma_0} \sum\limits_{j \in S_0} \Delta c_j |x_j| \\ s.d. & \sum\limits_{j=1}^{n} a_{ij} x_j + \max\limits_{S_i : |S_i| \leq \Gamma_i} \sum\limits_{j \in S_i} \Delta a_{ij} |x_j| \leq b_i \quad \forall i = 1, \ldots, m \\ & x_j \in \mathbb{Z},\ j = 1, \ldots, k \\ & x \in \mathbb{R}^n. \end{array}$$

Theorem 4.1. *$(RMIP)$ ist äquivalent zu*

$$
(RMIP^*) \qquad
\begin{array}{ll}
\min & c^\top x + z_0\Gamma_0 + \sum_{j=1}^{n} p_{0j} \\
s.d. & \sum_{j=1}^{n} a_{ij}x_j + z_i\Gamma_i + \sum_{j=1}^{n} p_{ij} \leq b_i \quad \forall i = 1,\dots,m \\
& z_0 + p_{0j} \geq \Delta c_j \cdot y_j \quad \forall j = 1,\dots,n \\
& z_i + p_{ij} \geq \Delta a_{ij} \cdot y_j \quad \forall j = 1,\dots,n, \quad \forall i = 1,\dots,m \\
& p_{ij} \geq 0 \quad \forall j = 1,\dots,n, \quad \forall i = 0,\dots,m \\
& y_j \geq 0 \quad \forall j = 1,\dots,n \\
& z_i \geq 0 \quad \forall i = 0,\dots,m \\
& -y_j \leq x_j \leq y_j \quad \forall j = 1,\dots,n \\
& x_j \in \mathbb{Z} \quad \forall j = 1,\dots,k \\
& x \in \mathbb{R}^n.
\end{array}
$$

Folgerung 4.2. *Die robuste Formulierung eines gemischt ganzzahligen Programms mit Box-Unsicherheiten ist wieder ein gemischt ganzzahliges Programm.*

Beweis von Theorem 4.1. Für festes x gilt

$$
\begin{aligned}
\max_{S_0:|S_0|\leq\Gamma_0} \sum_{j\in S_0} \Delta c_j|x_j| \quad &= \quad \max_{\alpha_{0j}}\{\sum_{j=1}^{n} \Delta c_j|x_j|\alpha_{0j} : \sum_{j=1}^{n} \alpha_{0j} \leq \Gamma_0,\ \alpha_{0j} \in [0;1] \quad \forall j\} \\
&\overset{dual}{=} \quad \min_{z_0,p_{0j}} \sum_{j=1}^{n} p_{0j} + \Gamma_0 z_0 \\
&\qquad s.d.\ z_0 + p_{0j} \geq \Delta c_j|x_j| \quad \forall j, \quad p_{0j} \geq 0 \quad \forall j, \quad z_0 \geq 0.
\end{aligned}
$$

Analog hierzu gilt für alle Indizes $i = 1,\dots,m$

$$
\begin{aligned}
\max_{S_i:|S_i|\leq\Gamma_i} \sum_{j\in S_i} \Delta a_{ij}|x_j| \quad &= \quad \min_{z_i,p_{ij}} \sum_{j=1}^{n} p_{ij} + \Gamma_i z_i \\
&\qquad s.d.\ z_i + p_{ij} \geq \Delta a_{ij}|x_j| \quad \forall j, \quad p_{ij} \geq 0 \quad \forall j, \quad z_i \geq 0.
\end{aligned}
$$

Also ist $(RMIP)$ äquivalent zu

$$\begin{aligned}
\min \quad & c^\top x + \sum_{j=1}^{n} p_{0j} + \Gamma_0 z_0 \\
s.d. \quad & \sum_{j=1}^{n} a_{ij}x_j + \sum_{j=1}^{n} p_{ij} + \Gamma_i z_i \leq b_i \quad \forall i = 1,\dots,m \\
& p_{ij} \geq 0 \quad \forall j = 1,\dots,n, \quad \forall i = 0,\dots,m \\
& z_i \geq 0 \quad \forall i = 0,\dots,m \\
& z_0 + p_{0j} \geq \Delta c_j |x_j| \quad \forall j = 1,\dots,n \\
& z_i + p_{ij} \geq \Delta a_{ij} |x_j| \quad \forall j = 1,\dots,n, \quad \forall i = 1,\dots,m \\
& x_j \in \mathbb{Z} \quad \forall j = 1,\dots,k \\
& x \in \mathbb{R}^n.
\end{aligned}$$

Die Formel $(RMIP^*)$ erhalten wir nun, indem wir für alle Indizes j die Variablen $|x_j|$ durch y_j mit $y_j \geq 0$, $-y_j \leq x_j \leq y_j$ ersetzen. $\square$

4.2 Robuste 0-1 Optimierung

Wir betrachten nun das folgende binäre Problem

$$(BP) \qquad \min_x \left\{ c^\top x : \ x \in X \subseteq \{0,1\}^n \right\}.$$

Dabei gibt die Menge $X \subseteq \{0,1\}^n$ Restriktionen an. Viele kombinatorische Probleme können so modelliert werden. Weiter nehmen wir an, dass Unsicherheiten nur im Zielfunktionsvektor c auftreten, d.h. die c_j sind Zufallsvariablen in $[c_j, c_j + \Delta c_j]$ mit $\Delta c_j \geq 0$.

Die robuste Formulierung von (BP) ist dann

$$(RBP) \qquad \min_x \left\{ c^\top x + \max_{S:|S|\leq\Gamma_0} \sum_{j\in S} \Delta c_j x_j : \ x \in X \subseteq \{0,1\}^n \right\}.$$

Beachte: Da die $x_j \geq 0$ sind, gilt $|x_j| = x_j$ für alle $j = 1,\dots,n$.

Beispiele:

- Kürzeste Wege und Traveling Salesman Problem, wobei die Länge einer Kante unsicher ist (z.B. wegen Stau).
- Netzwerk-Design Probleme, wobei die Kosten für den Bau einer Verbindung unsicher sind (z.B. wegen Bodenbeschaffenheit).

Ein **Typischer Ansatz** besteht darin, eine endliche Zahl möglicher Szenarien $c_1, \dots, c_m$ zu betrachten und deren Maximum zu verwenden. Für zwei Szenarien c_1 und c_2 sieht dies wie folgt aus:

$$(RBP_2) \qquad \min_x \left\{ \max\{c_1^\top x, c_2^\top x\} : \; x \in X \right\}.$$

Jedoch kann (RBP_2) NP-schwer sein, selbst wenn (BP) polynomiell lösbar ist.

Theorem 4.3. *Das Kürzeste-Weg Problem mit zwei Szenarien ist NP-schwer.*

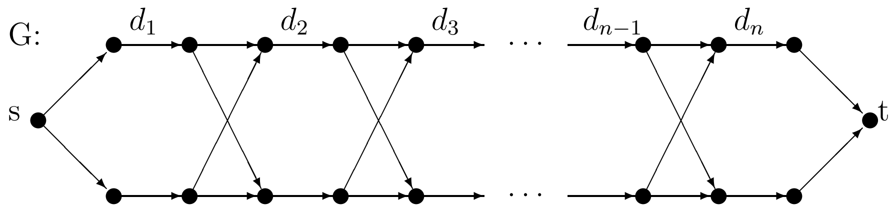
Beweis. Das folgende Problem ist NP-vollständig:

(PARTITION) Gegeben seien $d_1, \dots, d_m \in \mathbb{Z}$. Entscheide, ob ein $S \subseteq \{1, \dots, n\}$ existiert mit $\sum_{j \in S} d_j = \sum_{j \notin S} d_j$.

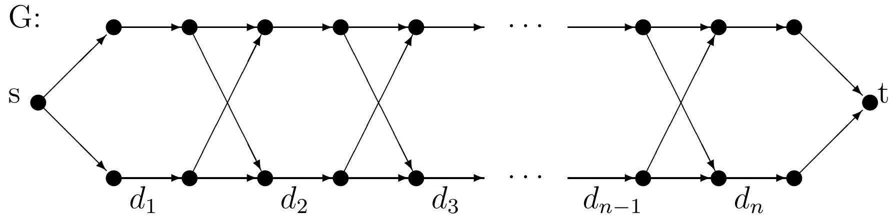
Wir reduzieren (PARTITION) auf

$$(RSP) \qquad \min_x \left\{ \max\{c_1^\top x, c_2^\top x\} : \; x \text{ ist ein Pfad in } G \right\},$$

wobei der Graph G wie folgt gegeben sei



für die Kosten c_1 und



für die Kosten c_2. Alle Kanten ohne explizit gegebenes Kantengewicht haben den Wert Null.

Falls $S \subseteq \{1, \dots, n\}$ existiert mit $\sum_{j \in S} d_j = \sum_{j \notin S} d_j$, so wähle einen Pfad x so, dass für jedes $j \in S$ die obere Kante zu x gehört und sonst die untere Kante. Dann ist also $c_1^\top x = \sum_{j \in S} d_j$ und $c_2^\top x = \sum_{j \notin S} d_j$. Somit ist

$$(*) \qquad \max\{c_1^\top x, c_2^\top x\} = \frac{1}{2} \sum_{j=1}^{n} d_j.$$

Umgekehrt, wenn ein Pfad x mit der Eigenschaft $(*)$ existiert, dann setze

$$S = \{j : \ x \text{ enthält die obere Kante an der Stelle } j\}.$$

Also existiert eine Lösung für (PARTITION) genau dann, wenn eine optimale Lösung von (RSP) $\frac{1}{2}\sum_{j=1}^{n} d_j$ ist. □

Da für jedes Polytop P eine Ecke $c^* \in P$ existiert mit $\max_{c\in P} c^\top x = x^\top c^*$ und P endlich viele Ecken besitzt, folgt mit Theorem 4.3 das folgende Ergebnis:

Folgerung 4.4. *Für ein Polytop P ist das Problem*

$$\min_{x\in X} \max_{c\in P} c^\top x$$

im Allgemeinen NP-schwer, auch wenn (BP) für festes c polynomiell lösbar ist.

Aber Problem (RBP) bleibt (im Gegensatz zu (RBP_2)) polynomiell lösbar, wenn (BP) polynomiell lösbar ist.

Theorem 4.5. *Sei o.B.d.A. $\Delta c_1 \geq \Delta c_2 \geq \ldots \geq \Delta c_n \geq 0$. Dann ist die optimale Lösung von (RBP) gleich*

$$\min_{l=1,\ldots,n+1} \left(\Gamma_0 \Delta c_l + \min_{x\in X} \left(c^\top x + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j \right) \right)$$

wobei $\Delta c_{n+1} := 0$ sei. Man kann also (RBP) lösen, indem man $n+1$ Probleme (BP) löst.

Beweis. Es gilt

$$\begin{aligned}
& \min_{x} \left\{ c^\top x + \max_{S:|S|\leq\Gamma_0} \sum_{j=1}^{n} \Delta c_j x_j : \ x \in X \right\} \\
= \quad & \min_{x} \quad c^\top x + \max_{u} \sum_{j=1}^{n} \Delta c_j x_j u_j \\
& \text{s.d.} \quad 0 \leq u_j \leq 1 \quad \forall j = 1, \ldots, n, \quad \sum_{j=1}^{n} u_j \leq \Gamma_0, \quad x \in X \\
\overset{dual}{=} \quad & \min_{x,\theta,y} \quad c^\top x + \Gamma_0 \theta + \sum_{j=1}^{n} y_j \\
& \text{s.d.} \quad y_j + \theta \geq \Delta c_j x_j, \ y_j \geq 0 \quad \forall j = 1, \ldots, n, \quad \theta \geq 0, \quad x \in X.
\end{aligned}$$

Für jede Optimallösung (x^*, θ^*, y^*) gilt

$$y_j^* = \max\{0; \Delta c_j x_j^* - \theta^*\} \qquad \forall j = 1, \ldots, n,$$

da die y_j positives Gewicht in der Zielfunktion haben. Durch ersetzen der y_j erhalten wir somit ein äquivalentes Problem

$$\min_{x,\theta} \left\{ c^\top x + \Gamma_0 \theta + \sum_{j=1}^{n} \max\{0; \Delta c_j x_j - \theta\} : \ \theta \geq 0, \ x \in X \right\}.$$

Wegen $x \in \{0,1\}^n$ und $\theta \geq 0$ gilt $\max\{0; \Delta c_j x_j - \theta\} = \max\{0; \Delta c_j - \theta\} x_j$ für alle j. Das äquivalente Programm ist somit

$$\min_{x,\theta} \left\{ c^\top x + \Gamma_0 \theta + \sum_{j=1}^{n} \max\{0; \Delta c_j - \theta\} x_j : \ \theta \geq 0, \ x \in X \right\}.$$

Es stört nun noch die Variable $\theta \geq 0$. Wir zerlegen nun $[0, \infty)$ in die Intervalle $[0, \Delta c_n]$, $[\Delta c_n, \Delta c_{n-1}]$, …, $[\Delta c_1, \infty)$. Dann ist

$$\max\{0; \Delta c_j - \theta\} x_j = \begin{cases} \sum_{j=1}^{l-1} (\Delta c_j - \theta) x_j & \text{falls } \theta \in [\Delta c_l, \Delta c_{l-1}] \\ 0 & \text{falls } \theta \in [\Delta c_1, \infty). \end{cases}$$

Das äquivalente Programm ist somit

$$\min_{x,\theta,l>1} \{c^\top x + \Gamma_0 \theta + \sum_{j=1}^{l-1} (\Delta c_j - \theta) x_j : \ \theta \geq 0, \ \theta \in [\Delta c_l, \Delta c_{l-1}], \ x \in X\}.$$

Wenn wir x und l fixieren, so ist die Zielfunktion linear. Das Minimum wird somit für $\theta = \Delta c_{l-1}$ oder $\theta = \Delta c_l$ angenommen. Das äquivalente Programm ist somit

$$\begin{aligned} \min\nolimits_{l>1} \quad & \min\{\Gamma_0 \Delta c_l + \min_{x \in X} \sum_{j=1}^{l-1} (\Delta c_j - \Delta c_l) x_j, \\ & \Gamma_0 \Delta c_{l-1} + \min_{x \in X} \sum_{j=1}^{l-1} (\Delta c_j - \Delta c_{l-1}) x_j\} \\ = \min_{l=1,\ldots,n+1} \quad & \Gamma_0 \Delta c_l + \min_{x \in X} \sum_{j=1}^{l-1} (\Delta c_j - \Delta c_l) x_j. \end{aligned}$$

$\square$

4.3 Beispiele zur robusten 0-1 Optimierung

4.3.1 Das robuste Rucksackproblem

Gegeben seien n Objekte mit Gewichten $a_1, \ldots, a_n$ und Werten $w_1, \ldots, w_n$ und eine Gewichtsschranke b.

Gesucht ist eine Teilmenge von Objekten mit einem maximalen Gesamtwert, so dass die Gewichtsschranke b respektiert ist.

Wir formulieren die Aufgabe als (MIP) mit Variablen $x_i \in \{0,1\}$ für jedes Objekt i. Dabei sei $x_i = 1$ genau dann, wenn i ausgewählt wird.

$$(Knap) \qquad \max_x \{w^\top x : \; a^\top x \leq b, \; x \in \{0,1\}^n\}.$$

Beispiel 4.1. *Es seien 4 Objekte gegeben mit* $b = 14$ *und*

$$\begin{array}{llll} a_1 = 12 & a_2 = 6 & a_3 = 6 & a_4 = 4 \\ w_1 = 200 & w_2 = 100 & w_3 = 75 & w_4 = 50. \end{array}$$

Dann existieren 4 Kandidaten $(1,0,0,0)$, $(0,1,1,0)$, $(0,1,0,1)$, $(0,0,1,1)$ *für die Optimallösung mit Werten 200, 175, 150, 125. Alle anderen zulässigen Lösungen können nicht optimal sein, da bei ihnen noch ein Objekt hinzugefügt werden kann. Es ist somit optimal, nur das erste Objekt auszuwählen.*

Angenommen, die Objekte können beim Transport beschädigt werden und dabei einen Wert Δw_j verlieren. Wir möchten nun die Beschädigung von bis zu Γ_0 Objekten berücksichtigen.

Dies kann wie folgt robust modelliert werden:

$$(RBP) \qquad \min_x \left\{ -w^\top x + \max_{S : |S| \leq \Gamma_0} \sum_{j \in S} \Delta w_j x_j : \; x \in \{0,1\}^n, \; \sum_{j=1}^n a_j x_j \leq b \right\}.$$

Nach Theorem 4.1 ist dies äquivalent zu

$$\begin{array}{ll} \min & (-w^\top x + z\Gamma_0 + \sum_{j=1}^n p_j) \\ s.d. & \sum_{j=1}^n a_j x_j \leq b \\ & z + p_j \geq \Delta w_j x_j \quad \forall j = 1, \ldots, n \\ & p_j \geq 0 \quad \forall j = 1, \ldots, n \\ & z \geq 0 \\ & x \in \{0,1\}^n \end{array}$$

Somit kann das robuste Rucksackproblem für jedes Γ_0 als MIP (mixed integer program) formuliert werden.

Beispiel 4.2 (Fortführung von Beispiel 4.1)**.** *Es sei nun* $\Delta w_1 = 150$*,* $\Delta w_2 = 60$*,* $\Delta w_3 = 50$ *und* $\Delta w_4 = 5$*. Dann erhalten wir folgende Werte für* $\Gamma_0 = 0, 1, 2$*:*

	$\Gamma_0 = 0$	$\Gamma_0 = 1$	$\Gamma_0 = 2$
(1,0,0,0)	*200*	*50*	*50*
(0,1,1,0)	*175*	*115*	*65*
(0,1,0,1)	*150*	*90*	*85*
(0,0,1,1)	*125*	*75*	*70*

Für $\Gamma_0 = 1$ *ist* $(0, 1, 1, 0)$ *die Optimallösung und* $(1, 0, 0, 0)$ *der schlechteste Kandidat. Für* $\Gamma_0 = 2$ *ist* $(0, 1, 0, 1)$ *optimal.*

4.3.2 Robuste minimal aufspannende Bäume

Gegeben sei ein zusammenhängender Graph $G = (V, E)$ mit Kosten c_e für jede Kante $e \in E$.

Gesucht sei ein aufspannender Baum in G mit minimalen Gesamtkosten.

Dies kann mit dem polynomiellen **Algorithmus von Kruskal** berechnet werden:

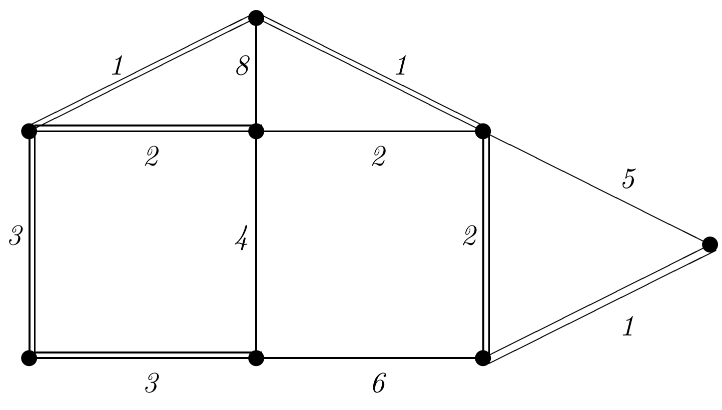
Durchlaufe die Kanten des Graphen so, dass die Kosten steigend sind. Füge dabei eine Kante zum schon berechneten Wald hinzu, wenn sie keinen Kreis erzeugt.

Das Problem minimal aufspannender Bäume

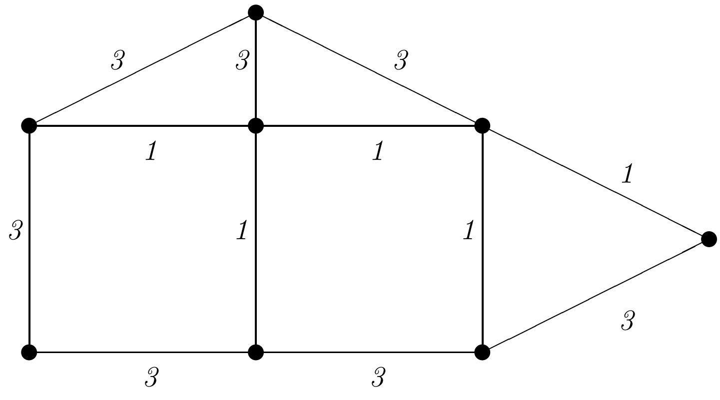
$$\min\{c^\top x : \ x \in \{0,1\}^E, \ x \text{ ist aufspannender Baum in } G\}$$

kann mit diesem Algorithmus in polynomieller Zeit gelöst werden.

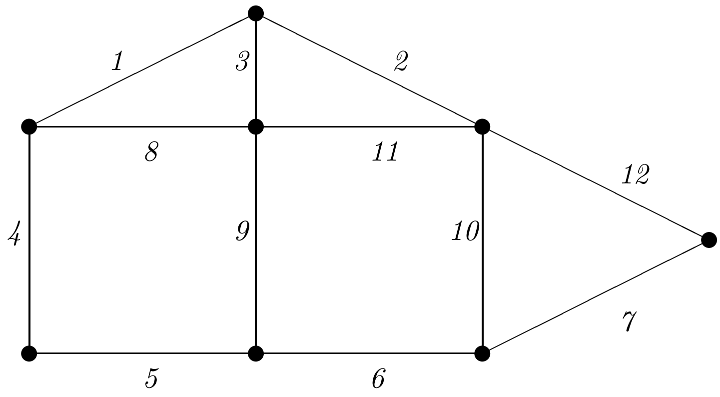
Beispiel 4.3. *Gegeben sei der folgende Graph mit den angegebenen Kantengewichten:*



Der minimal aufspannende Baum in diesem Graphen ist durch die Doppellinien gekennzeichnet. Weiter seien wieder Unsicherheiten Δc_j gegeben, um welche die Kantengewichte vergrößert werden können:



Wir wollen nun Theorem 4.5 anwenden mit $\Gamma_0 = 5$. Hierfür nummerieren wir die Kanten nach absteigender Unsicherheit:



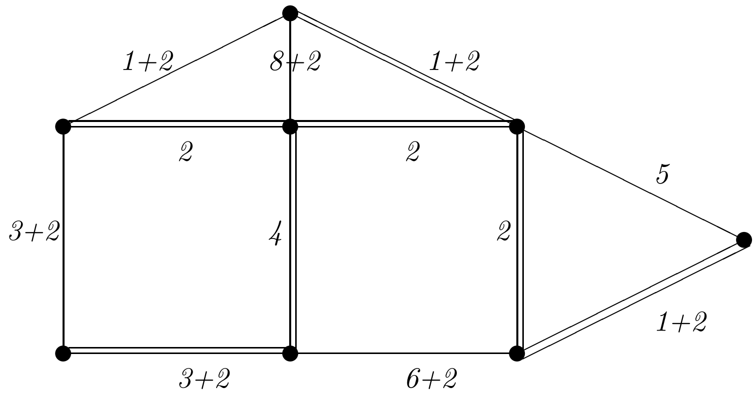
Es ist also $\Delta c_1 = \Delta c_2 = \ldots = \Delta c_7 = 3$ und $\Delta c_8 = \ldots = \Delta c_{12} = 1$. Mit Theorem 4.5 folgt für $l = 1, \ldots, 7$

$$\Gamma_0 \Delta c_l + \min_{x \in X} \left(c^\top x + \sum_{j=1}^{l} (\underbrace{\Delta c_j - \Delta c_l}_{=0}) x_j \right) = \Gamma_0 \Delta c_l + \min_{x \in X} c^\top x = 15 + 13 = 28.$$

Für $l = 8, \ldots, 12$ ist

$$\begin{aligned} & \Gamma_0 \Delta c_l + \min_{x \in X} \left(c^\top x + \sum_{j=1}^{7} (\underbrace{\Delta c_j - \Delta c_l}_{=2}) x_j + \sum_{j=8}^{l} (\underbrace{\Delta c_j - \Delta c_l}_{=0}) x_j \right) \\ = \; & \Gamma_0 \Delta c_l + \min_{x \in X} \left(c^\top x + \sum_{j=1}^{7} 2 x_j \right), \end{aligned}$$

d.h. das Problem minimal aufspannender Bäume wird nochmal gelöst, wobei die Werte c_i, $i = 1, \dots, 7$ um 2 erhöht werden:

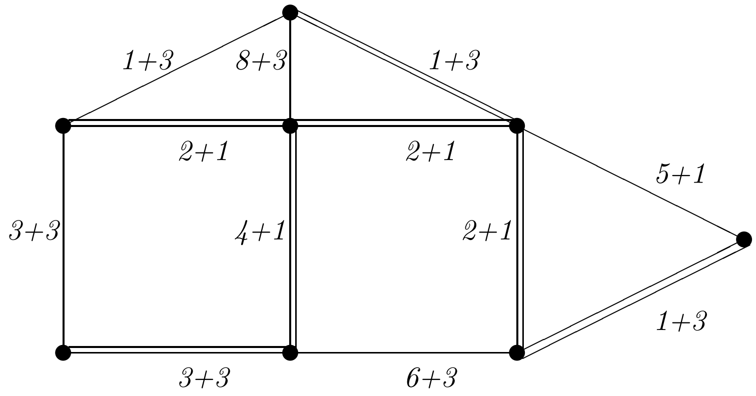


Der minimal aufspannende Baum besitzt den Wert 21*. Somit sind die Gesamtkosten* $5 + 21 = 26$.

Abschließend muss noch der Fall $l = 13$ untersucht werden mit $\Delta c_{13} = 0$:

$$\begin{aligned} &\Gamma_0 \Delta c_l + \min_{x \in X} \left(c^\top x + \sum_{j=1}^{7} (\underbrace{\Delta c_j - \Delta c_l}_{=3}) x_j + \sum_{j=8}^{13} (\underbrace{\Delta c_j - \Delta c_l}_{=1}) x_j \right) \\ = \;& \min_{x \in X} \left(c^\top x + \sum_{j=1}^{7} 3x_j + \sum_{j=8}^{12} x_j \right). \end{aligned}$$

Das Problem minimal aufspannender Bäume wird nun gelöst, wobei die Werte c_i für $i = 1, \dots, 7$ um 3 erhöht werden und für $i = 8, \dots, 12$ um 1.



Die Gesamtkosten sind also 28 für $l = 13$. Wir vergleichen nun die Gesamtkosten für die verschiedenen l. Der niedrigste Wert ist 26. Der zugehörige Spannbaum ist also robust optimal.

4.4 Übertragung von Approximationsergebnissen

Sei $\alpha \geq 1$. Eine Klasse von Problemen der Form (MIP) heißt α-approximierbar, wenn es einen polynomiellen Algorithmus gibt, der für jede Instanz eine zulässige Lösung $x \in \mathbb{R}^n$ berechnet mit $c^\top x \leq \alpha c^\top x^*$, wobei x^* eine optimale Lösung für die Instanz ist.
(Bei max statt min: $c^\top x \geq \frac{1}{\alpha} c^\top x^*$.)

Beispiel 4.4. *Das Rucksackproblem ist NP-schwer, aber der folgende Algorithmus findet immer eine Lösung, die mindestens den halben Wert des Optimums erreicht:*

1. *Sortiere die Indizes so, dass*
$$\frac{w_1}{a_1} \geq \frac{w_2}{a_2} \geq \ldots \geq \frac{w_n}{a_n} \qquad \textit{gilt.}$$
2. *Für $i = 1, \ldots, n$:*
Füge i hinzu, wenn es die Gewichtsgrenze erlaubt.
3. *Wenn diese Lösung schlechter ist, als der größte Wert w_i, ersetze sie durch $\{i\}$.*

Das Rucksackproblem ist also 2-approximierbar.

Beispiel 4.5. *Das Problem des Handlungsreisenden (TSP) ist $\frac{3}{2}$–approximierbar, wenn für die Kosten c die Dreiecksungleichung $c_{ij} + c_{jk} \geq c_{ik}$ für alle i, j, k gilt.*

Bleiben die robusten Versionen α-approximierbar?

Folgerung 4.6. *Falls (BP) α-approximierbar ist für beliebige Kosten c, dann ist auch (RBP) α-approximierbar.*

Beweis. Betrachte die Probleme

$$(BP_l) \qquad \min_x \{c^\top x + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j;\ x \in X\}$$

für $l = 1, \ldots, n+1$. Laut Voraussetzung können wir in polynomieller Zeit zulässige Lösungen $x^l \in X$ für die Aufgaben (BP_l) berechnen. Es seien

$$z_l := c^\top x^l + \max_{S:|S| \leq \Gamma_0} \sum_{j=1}^{l} \Delta c_j x_j^l, \quad l = 1, \ldots, n+1,$$

und $z = \min\{z_1, \ldots, z_{n+1}\}$.

Wir wollen zeigen, dass z höchstens um Faktor α größer ist, als das Optimum von (RBP).

Für alle $l = 1, \ldots, n+1$ gilt:

$$\begin{aligned}
z &\leq c^\top x^l + \max_{S:|S|\leq\Gamma_0} \sum_{j=1}^{l} \Delta c_j x_j^l \\
&= \min_{\theta\geq 0}\left(c^\top x^l + \Gamma_0\theta + \sum_{j=1}^{n} \max\{0; \Delta c_j - \theta\} x_j^l\right) \\
&\quad \text{(wie im Beweis von Theorem 4.5)} \\
&\leq c^\top x^l + \Gamma_0 \Delta c_l + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j^l \qquad \text{wegen } \theta := \Delta c_l \\
&\leq \alpha \cdot \min_{x\in X}\left(c^\top x + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j\right) + \Gamma_0 \Delta c_l \qquad \text{wegen } (*) \\
&\leq \alpha \cdot \left(\min_{x\in X}\left(c^\top x + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j\right) + \Gamma_0 \Delta c_l\right) \qquad \text{wegen } \alpha \geq 1.
\end{aligned}$$

Also ist

$$\begin{aligned}
z &\leq \alpha \cdot \min_{l=1,\ldots,n+1}\left(\min_{x\in X}\left(c^\top x + \sum_{j=1}^{l} (\Delta c_j - \Delta c_l) x_j\right) + \Gamma_0 \Delta c_l\right) \\
&= \alpha \cdot (\text{Optimum von } (RBP)\ ) \qquad \text{wegen Theorem 4.5.}
\end{aligned}$$

□

Beispiel 4.6 (Robuste Approximierung des Rucksackproblems)**.** *Wir führen nun das Beispiel 4.1 bzw. das Beispiel 4.2 fort. Es sei also* $b = 14$ *und*

$$\begin{array}{llll}
a_1 = 12 & a_2 = 6 & a_3 = 6 & a_4 = 4 \\
w_1 = 200 & w_2 = 100 & w_3 = 75 & w_4 = 50.
\end{array}$$

Wir berechnen zunächst die 2-Approximation ohne Unsicherheit($\Gamma_0 = 0$*). Es ist*

$$\frac{w_1}{a_1} = \frac{w_2}{a_2} = \frac{200}{12} > \frac{w_3}{a_3} = \frac{w_3}{a_3} = \frac{150}{12}.$$

Somit ist $\{2, 1, 3, 4\}$ *eine mögliche Sortierung. Es wird also 2 eingepackt, 1 passt nicht, 3 wird eingepackt und 4 passt nicht. Dem entspricht der boolsche Vektor* $x^0 = (0, 1, 1, 0)$ *mit Wert 175. Im 3.Schritt des Algorithmus setzen wir dann* $x^0 = (1, 0, 0, 0)$*, da* $\{1\}$ *einen Wert* $200 > 175$ *besitzt.*

Robuste 2-Approximierung mit $\mathbf{\Gamma_0 = 1}$:

Es sei $\Delta w_1 = 150$, $\Delta w_2 = 60$, $\Delta w_3 = 50$ und $\Delta w_4 = 5$. Wir lösen nun die Probleme (RBP_l) für $l = 1, \ldots, 5$. Dabei ist zu beachten, dass jetzt maximiert statt minimiert wird.

$l = 1$: *Das Problem (RBP_1) entspricht (BP), d.h. die approximierte Lösung ist $x^1 = (1, 0, 0, 0)$ mit dem Wert 200. Wegen $\Delta w_1 = 150$ ist also der Gesamtwert $z_1 = 200 - 150 = 50$.*

$l = 2$: *Das Problem (RBP_2) entspricht*

$$\max_{x \in X}(w^\top x - (\Delta w_1 - \Delta w_2)x_1) = \max_{x \in X}(110x_1 + 100x_2 + 75x_3 + 50x_4).$$

Mit den Quotienten $\frac{110}{12}$, $\frac{200}{12}$, $\frac{150}{12}$, $\frac{150}{12}$ ist die Reihenfolge $\{2, 3, 4, 1\}$ korrekt. Mit dem Approximationsalgorithmus erhält man nun $x^2 = (0, 1, 1, 0)$ mit dem Gesamtwert $z_2 = 175 - \max\{60, 50\} = 115$.

$l = 3$: *Das Problem (RBP_3) entspricht*

$$\begin{aligned} & \max_{x \in X}(w^\top x - (\Delta w_1 - \Delta w_3)x_1 - (\Delta w_2 - \Delta w_3)x_2) \\ = \; & \max_{x \in X}(100x_1 + 90x_2 + 75x_3 + 50x_4). \end{aligned}$$

Mit den Quotienten $\frac{100}{12}$, $\frac{180}{12}$, $\frac{150}{12}$, $\frac{150}{12}$ ist wieder die Reihenfolge $\{2, 3, 4, 1\}$ korrekt. Somit ist $x^3 = (0, 1, 1, 0)$ optimal mit dem Gesamtwert $z_3 = 175 - 60 = 115$.

$l = 4$: *Das Problem (RBP_4) entspricht*

$$\begin{aligned} & \max_{x \in X}(w^\top x - (\Delta w_1 - \Delta w_4)x_1 - (\Delta w_2 - \Delta w_4)x_2) - (\Delta w_3 - \Delta w_4)x_3 \\ = \; & \max_{x \in X}(55x_1 + 45x_2 + 30x_3 + 50x_4). \end{aligned}$$

Mit den Quotienten $\frac{55}{12}$, $\frac{90}{12}$, $\frac{60}{12}$, $\frac{150}{12}$ ist die Reihenfolge $\{4, 2, 3, 1\}$ beim Approximationsalgorithmus zu verwenden. Dieser ergibt die Lösung $x^4 = (0, 1, 0, 1)$ mit $z_4 = 150 - \max\{60, 5\} = 90$.

$l = 5$: *Das Problem (RBP_5) entspricht*

$$\begin{aligned} & \max_{x \in X}(w^\top x - \Delta w_1 x_1 - \Delta w_2 x_2 - \Delta w_3 x_3 - \Delta w_4 x_4) \\ = \; & \max_{x \in X}(50x_1 + 40x_2 + 25x_3 + 45x_4). \end{aligned}$$

Mit den Quotienten $\frac{50}{12}$, $\frac{80}{12}$, $\frac{50}{12}$, $\frac{135}{12}$ ist die Reihenfolge $\{4, 2, 3, 1\}$ beim Approximationsalgorithmus zu verwenden. Man erhält $x^5 = (0, 1, 0, 1)$ mit $z_5 = 150 - \max\{60, 5\} = 90$.

Es ist also $x^2 = (0,1,1,0)$ mit $z = 115$ eine 2-approximierende Lösung des Knapsackproblems mit $\Gamma_0 = 1$. In diesem Beispiel entspricht sie sogar der robust optimalen Lösung (siehe Beispiel 4.2).

4.5 Robuste Netzwerkflüsse

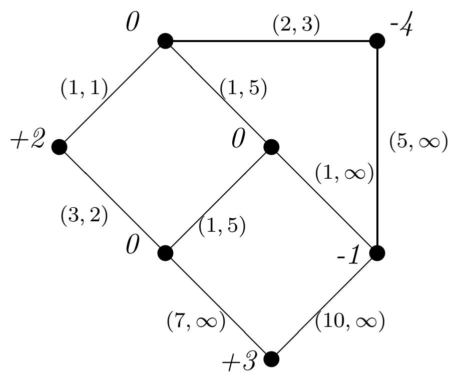
Gegeben sei ein gerichteter Graph $G = (V, E)$ mit Kapazitäten $u_{ij} > 0$ und Kosten $c_{ij} > 0$ für jede Kante $(i,j) \in E$. Außerdem seien für jeden Knoten $i \in V$ Versorgungen $b_i \in \mathbb{R}$ gegeben mit $\sum_{i \in V} b_i = 0$.

Es wird nun das **Minimum-Kosten-Fluss** Problem betrachtet, d.h. wir suchen die billigste Möglichkeit, um einen Fluss von den Knoten i mit $b_i > 0$ (Quellen) zu den Knoten j mit $b_j < 0$ (Senken) zu schicken.

Wir formulieren das Problem wie folgt:

$$\begin{array}{ll} \min & \displaystyle\sum_{(i,j)\in E} c_{ij} x_{ij} \\ s.d. & \displaystyle\sum_{j:(i,j)\in E} x_{ij} - \sum_{j:(j,i)\in E} x_{ji} = b_i \quad \forall i \in V \\ & 0 \le x_{ij} \le u_{ij} \quad \forall (i,j) \in E. \end{array}$$

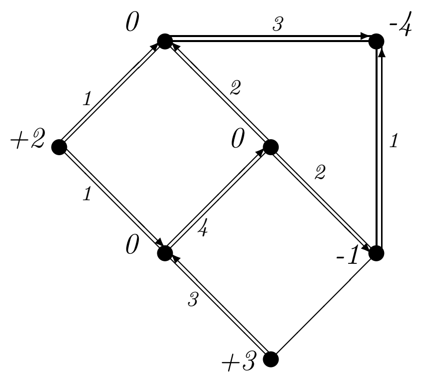
Beispiel 4.7.



(Kosten c_{ij}, Kapazität u_{ij})

Die Werte an den Knoten sind die Versorgungen b_i.

Ein billiger Fluss wird im folgenden Bild angegeben:



Wir wollen die robuste Version von (MCF) durch Lösen von polynomiell vielen Problemen (MCF) berechnen. Dabei kann Theorem 4.5 nicht verwendet werden, da die Variablen in (MCF) nicht binär sind.

Es sei X die Menge der zulässigen Lösungen von (MCF):

$$X = \left\{ x \in \mathbb{R}_+^E \;\middle|\; \begin{array}{cc} \sum_{j:(i,j)\in E} x_{ij} - \sum_{j:(j,i)\in E} x_{ji} = b_i & \forall i \in V \\ 0 \le x_{ij} \le u_{ij} & \forall (i,j) \in E \end{array} \right\}.$$

Die robuste Version von (MCF) ist dann

$$(RMCF) \qquad z^* = \min_{x\in X} \left(c^\top x + \max_{S\subseteq E:|S|\le\Gamma_0} \sum_{(i,j)\in S} \Delta c_{ij} x_{ij} \right).$$

Wir dualisieren nun wie im Beweis von Theorem 4.5. Es ist dann $z^* = \min_{\theta\ge 0} z(\theta)$ mit

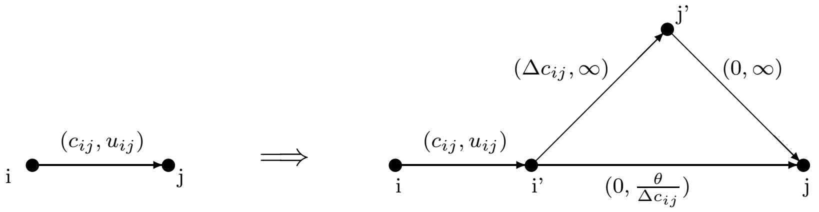
$$\begin{array}{rcll} z(\theta) & = & \Gamma_0\theta + \min\limits_{x,y}(c^\top x + \sum\limits_{(i,j)\in E} y_{ij}) & \\ s.d. & & y_{ij} + \theta \ge \Delta c_{ij} x_{ij} & \forall (i,j) \in E \\ & & y_{ij} \ge 0 & \forall (i,j) \in E \\ & & x \in X. & \end{array}$$

Theorem 4.7. *Für jedes $\theta \ge 0$ kann $z(\theta)$ durch Lösen eines Problems der Form (MCF) berechnet werden.*

Beweis. Zuerst ersetzen wir die Variablen y_{ij} durch $y_{ij} = \max(\Delta c_{ij} x_{ij} - \theta, 0)$. Dann ist

$$\begin{aligned} z(\theta) &= \Gamma_0\theta + \min_{x\in X}(c^\top x + \sum_{(i,j)\in E} \max(\Delta c_{ij} x_{ij} - \theta, 0)) \\ &= \Gamma_0\theta + \min_{x\in X}(c^\top x + \sum_{(i,j)\in E} \Delta c_{ij} \max(x_{ij} - \frac{\theta}{\Delta c_{ij}}, 0)). \end{aligned}$$

Nun ersetzen wir jede Kante $(i,j) \in E$ wie folgt:



Wir behaupten nun: Eine optimale Lösung des konstruierten Problems liefert eine optimale Lösung vom Problem

$$(*) \qquad \min_{x \in X}(c^\top x + \sum_{(i,j)\in E} \Delta c_{ij} \max(x_{ij} - \frac{\theta}{\Delta c_{ij}}, 0)).$$

Sei x optimal für $(*)$. Wir unterscheiden nun 2 Fälle.

1. Sei $x_{ij} - \frac{\theta}{\Delta c_{ij}} \leq 0$. Dann wird im konstruierten Problem der Fluss über (i,i') und (i',j) geschickt, da diese Variante am billigsten ist und die Kapazität ausreicht. Die Kosten hierfür sind $c_{ij}x_{ij}$.

2. Sei $x_{ij} - \frac{\theta}{\Delta c_{ij}} > 0$. Dann wird der Fluss $\frac{\theta}{\Delta c_{ij}}$ über (i,i') und (i',j) geschickt und der Rest $x_{ij} - \frac{\theta}{\Delta c_{ij}}$ über die teuerere Variante (i,i'), (i',j') und (j',j). Die Kosten hierfür sind $c_{ij}x_{ij} + \Delta c_{ij}\frac{\theta}{\Delta c_{ij}}$.

In beiden Fällen stimmen die Kosten mit denen in der Zielfunktion von (*) überein. Also kann $z(\theta)$ durch das Lösen des konstruierten Problems bestimmt werden. □

Theorem 4.8. *Die Funktion $z : \mathbb{R}_+ \to \mathbb{R}$ hat folgende Eigenschaften:*

1. *z ist konvex.*

2. *Für $\theta_1, \theta_2 \geq 0$ gilt $|z(\theta_1) - z(\theta_2)| \leq |E| \cdot |\theta_1 - \theta_2|$.*

Beweis. 1. Seien (x^1, y^1) und (x^2, y^2) optimale Lösungen für $\theta = \theta_1$ und $\theta = \theta_2$. Dann ist $(\lambda x^1 + (1-\lambda)x^2, \lambda y^1 + (1-\lambda)y^2)$ zulässig für $\theta = \lambda\theta_1 + (1-\lambda)\theta_2$. Somit gilt

$$\begin{aligned}
\lambda z(\theta_1) + (1-\lambda)z(\theta_2) &= c^\top(\lambda x^1 + (1-\lambda)x^2) + \sum_{(i,j)\in E}(\lambda y^1_{ij} + (1-\lambda)y^2_{ij}) \\
&\quad + \Gamma_0(\lambda\theta_1 + (1-\lambda)\theta_2) \\
&\geq z(\lambda\theta_1 + (1-\lambda)\theta_2),
\end{aligned}$$

da $z(\lambda\theta_1 + (1-\lambda)\theta_2)$ minimal ist.

2. Da die Minimierungsaufgabe zur Berechnung von $z(\theta)$ linear und lösbar ist, können wir $z(\theta)$ durch die duale Aufgabe wie folgt berechnen:

$$\begin{aligned}
z(\theta) &= \max_{r\geq 0} \min_{x\in X, y\geq 0} \Gamma_0\theta + c^\top x + \sum_{(i,j)\in E} y_{ij} + \sum_{(i,j)\in E} r_{ij}(\Delta c_{ij}x_{ij} - y_{ij} - \theta) \\
&= \max_{r\geq 0} \min_{x\in X, y\geq 0} (\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta + c^\top x + \sum_{(i,j)\in E} y_{ij}(1 - r_{ij}) \\
&\qquad + \sum_{(i,j)\in E} r_{ij}\Delta c_{ij}x_{ij} \\
&= \max_{0\leq r_{ij}\leq 1} \min_{x\in X} \left((\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta + c^\top x + \sum_{(i,j)\in E} r_{ij}\Delta c_{ij}x_{ij} \right).
\end{aligned}$$

Aus $r_{ij} \in [0,1]$ für alle $(i,j) \in E$ und $0 \leq \Gamma_0 \leq |E|$ folgt

$$-|E| \leq \Gamma_0 - \sum_{(i,j)\in E} r_{ij} \leq |E|.$$

Weiter sei o.B.d.A. $\theta_1 > \theta_2 \geq 0$. Dann ist

$$(\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta_1 \leq (\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta_2 + |E|(\theta_1 - \theta_2)$$

und somit

$$\begin{aligned}
z(\theta_1) &= \max_{0\leq r_{ij}\leq 1} \min_{x\in X} \left((\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta_1 + c^\top x + \sum_{(i,j)\in E} r_{ij}\Delta c_{ij}x_{ij} \right) \\
&\leq \max_{0\leq r_{ij}\leq 1} \min_{x\in X} \left((\Gamma_0 - \sum_{(i,j)\in E} r_{ij})\theta_2 + c^\top x + \sum_{(i,j)\in E} r_{ij}\Delta c_{ij}x_{ij} \right) \\
&\quad +|E|(\theta_1 - \theta_2) \\
&= z(\theta_2) + |E|(\theta_1 - \theta_2).
\end{aligned}$$

Analog hierzu ist $z(\theta_1) \geq z(\theta_2) - |E|(\theta_1 - \theta_2)$. Also gilt die Behauptung $|z(\theta_1) - z(\theta_2)| \leq |E| \cdot |\theta_1 - \theta_2|$.

$\square$

Theorem 4.9. *Sei $\epsilon > 0$. Man kann eine Lösung $\bar{x} \in X$ berechnen mit*

$$c^\top \bar{x} + \max_{S:|S|\leq\Gamma_0} \sum_{(i,j)\in S} \Delta c_{ij}\bar{x}_{ij} \leq \epsilon + z^*$$

durch Lösen von

$$3 + 2\lceil \log_2(|E| \cdot \frac{\bar{\theta}}{\epsilon}) \rceil$$

Problemen der Form (MCF), wobei $\bar{\theta} = \max\{u_{ij}\Delta c_{ij} :\ (i,j) \in E\}$ sei.

Beweis. Nach Theorem 4.8 ist die Funktion $z(\theta)$ konvex. Sei $\theta^* \geq 0$ eine Optimallösung von $\min_{\theta \geq 0} z(\theta)$, d.h. $z^* = z(\theta^*)$. Dabei gilt $\theta^* \in [0; \bar{\theta}]$.

Es wird nun binäre Suche verwendet, d.h. wir berechnen zunächst $z(0)$, $z(0.25 \cdot \bar{\theta})$, $z(0.5 \cdot \bar{\theta})$, $z(0.75 \cdot \bar{\theta})$ und $z(\bar{\theta})$.

Im Fall $z(0.25 \cdot \bar{\theta}) \leq z(0.5 \cdot \bar{\theta})$ liegt $\theta^* \in [0, 0.5 \cdot \bar{\theta}]$.
Im Fall $z(0.5 \cdot \bar{\theta}) \geq z(0.75 \cdot \bar{\theta})$ liegt $\theta^* \in [0.5 \cdot \bar{\theta}, \bar{\theta}]$.
Sonst liegt θ^* im Intervall $[0.25 \cdot \bar{\theta}, 0.75 \cdot \bar{\theta}]$.

Das entsprechende Intervall wird nun nochmal halbiert und die Prozedur entsprechend oft wiederholt. In jedem Schritt sind jeweils 2 weitere Berechnungen nötig. Nach k Schritten gilt im entsprechenden Intervall

$$|z(\theta) - z(\theta^*)| \leq |E||\theta - \theta^*| \leq |E| \frac{\bar{\theta}}{2^k}.$$

Nach $\lceil \log_2(|E| \cdot \frac{\bar{\theta}}{\epsilon}) \rceil$ Schritten gilt also $|z(\theta) - z(\theta^*)| \leq \epsilon$. Somit ist die angegebene Anzahl von Problemen der Form (MCF) zu lösen. □

4.6 Robuste binäre Optimierung mit ellipsoidaler Unsicherheit

4.6.1 Grundbegriffe

Wir betrachten wieder das binäre Problem

$$(BP) \qquad \min_x \{c^\top x : \ x \in X\}$$

mit $X \subseteq \{0,1\}^n$. Unsicherheiten seien nur in den Koeffizienten der Zielfunktion vorhanden. Sie sei zunächst durch eine beliebige Menge $\mathcal{C} \subseteq \mathbb{R}^n$ möglicher Zielfunktionsvektoren gegeben. Die robuste Version von (BP) ist dann

$$(RBP) \qquad \min_x \max_{c \in \mathcal{C}} \{c^\top x : \ x \in X\}.$$

Sei nun $c^* \in \mathcal{C}$ und $\mathcal{D} = \{c - c^* : \ c \in \mathcal{C}\}$, dann ist die Aufgabe (RBP) äquivalent zu

$$\min_{x \in X} \left({c^*}^\top x + \xi(x) \right)$$

mit $\xi(x) := \max\{d^\top x : \ d \in \mathcal{D}\}$.

Spezialfälle:

1. Im letzten Kapitel war

$$\mathcal{D} = \left\{ d \in \mathbb{R}^n : \; 0 \le d_j \le \Delta c_j \quad \forall j, \; \sum_{j=1}^{n} \frac{d_j}{\Delta c_j} \le \Gamma_0 \right\},$$

denn damit ist

$$\begin{aligned} \xi(x) &= \max_{d \in \mathcal{D}} d^\top x \\ &= \max_{d} \left\{ d^\top x : \; 0 \le d_j \le \Delta c_j \quad \forall j, \; \sum_{j=1}^{n} \frac{d_j}{\Delta c_j} \le \Gamma_0 \right\} \\ &= \max_{d} \left\{ \sum_{j=1}^{n} \alpha_j \Delta c_j x_j : \; 0 \le \alpha_j \le 1 \quad \forall j, \; \sum_{j=1}^{n} \alpha_j \le \Gamma_0 \right\} \\ &= \max_{S : |S| \le \Gamma_0} \sum_{j=1}^{n} \Delta c_j x_j. \end{aligned}$$

Zum Beispiel für $n = 2$, $\Gamma_0 = 1$, $\Delta c_1 = 1$ und $\Delta c_2 = 2$ ist

$$\begin{aligned} \xi(x) &= \max\{x_1, 2x_2\} \qquad \text{und} \\ \mathcal{D} &= \{d \in \mathbf{R}^2 : \; 0 \le d_1 \le 1, \; 0 \le d_2 \le 2, \; d_1 + 0.5 d_2 \le 1\}. \end{aligned}$$

2. Falls D ein allgemeines Polytop ist, gilt

$$\xi(x) = \max_{d \in \mathcal{D}} d^\top x = \max_{d} \{ d^\top x : \; \text{d ist Ecke von } \mathcal{D} \}.$$

Dies ist also das Maximum über endlich vielen Werten. In Theorem 4.3 haben wir bereits gezeigt, dass solche Probleme i.A. NP-schwer sind.

Im Folgenden betrachten wir als weiteren Spezialfall **ellipsoidale Unsicherheiten** (mit unkorrelierten Kosten)

$$\mathcal{D} = \left\{ d \in \mathbb{R}^n : \; \sqrt{\sum_{j=1}^{n} \left(\frac{d_j}{\sigma_j} \right)^2} \le r \right\} = \left\{ d \in \mathbb{R}^n : \; \left\| \left(\frac{d_1}{\sigma_1}, \ldots, \frac{d_n}{\sigma_n} \right) \right\|_2 \le r \right\}$$

mit $r > 0$, $\sigma = (\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^n$ und $\sigma_j > 0$ für alle j.

4.6.2 Unkorrelierte Kosten

Mit

$$D = \left\{ d \in \mathbb{R}^n : \; \left\| \left(\frac{d_1}{\sigma_1}, \ldots, \frac{d_n}{\sigma_n} \right) \right\|_2 \le r \right\}$$

ist

$$\xi(x) = \max_{d \in \mathcal{D}} d^\top x = r \cdot \sqrt{\sum_{j=1}^{n} \sigma_j^2 x_j^2}.$$

Da wir $x \in \{0,1\}^n$ annehmen, gilt also

$$\xi(x) = r \cdot \sqrt{\sum_{j=1}^{n} \sigma_j^2 x_j^2} = r \cdot \sqrt{\sum_{j=1}^{n} \tilde{\sigma}_j x_j}, \qquad \tilde{\sigma} = (\sigma_1^2, \ldots, \sigma_n^2).$$

Wir wollen nun also die robuste Aufgabe

$$(RBP - E) \qquad z^* = \min_{x \in X} \left(c^\top x + r \cdot \sqrt{\tilde{\sigma}^\top x} \right)$$

lösen. Sei im Folgenden $\mathcal{W} = \{\tilde{\sigma}^\top x : \ x \in \{0,1\}^n\}$, $f(w) = r \cdot \sqrt{w}$ und

$$g(w) = \begin{cases} f'(w) = \frac{1}{2} r \cdot \frac{1}{\sqrt{w}} & \text{für } w \in \mathcal{W} \backslash \{0\} \\ r \cdot \frac{1}{\sqrt{\bar{\sigma}}} & \text{für } w = 0 \end{cases}$$

mit $\bar{\sigma} = \min_j \sigma_j^2$.

Theorem 4.10. *Sei*

$$z(w) = \min_{x \in X} (c + g(w)\tilde{\sigma})^\top x + r\sqrt{w} - wg(w). \qquad (*)$$

Dann gilt $z^* = \min_{w \in \mathcal{W}} z(w)$, d.h. man kann (RBP_E) lösen, indem man $|\mathcal{W}|$ Probleme der Form (BP) löst.

Beweis. Sei x^* eine optimale Lösung von $(RBP - E)$ und sei $w^* = \tilde{\sigma}^\top x^*$. Dann gilt

$$\begin{aligned}
z^* &= c^\top x^* + r \cdot \sqrt{\tilde{\sigma}^\top x^*} \\
&= c^\top x^* + f(w^*) \\
&= (c + g(w^*)\tilde{\sigma})^\top x^* - (g(w^*)\tilde{\sigma})^\top x^* + f(w^*) \\
&= (c + g(w^*)\tilde{\sigma})^\top x^* + f(w^*) - g(w^*)w^* \\
&\geq \min_{x \in X} (c + g(w^*)\tilde{\sigma})^\top x + f(w^*) - g(w^*)w^* \\
&= z(w^*) \\
&\geq \min_{w \in \mathcal{W}} z(w),
\end{aligned}$$

d.h. es ist $z^* \geq \min_{w \in \mathcal{W}} z(w)$.

Für $w \in \mathcal{W}$ sei x_w eine Optimallösung von $(*)$. Dann gilt

$$\begin{aligned} z(w) &= (c + g(w)\tilde{\sigma})^\top x_w + f(w) - wg(w) \\ &= c^\top x_w + f(\tilde{\sigma}^\top x_w) + g(w)(\tilde{\sigma}^\top x_w - w) - (f(\tilde{\sigma}^\top x_w) - f(w)) \\ &\geq c^\top x_w + f(\tilde{\sigma}^\top x_w), \end{aligned}$$

denn für $w > 0$ gilt wegen der Konkavität von f

$$\frac{g(w)(\tilde{\sigma}^\top x_w - w)}{f(\tilde{\sigma}^\top x_w) - f(w)} = \frac{f'(w)}{\frac{f(\tilde{\sigma}^\top x_w) - f(w)}{\tilde{\sigma}^\top x_w - w}} \geq 1$$

und für $w = 0$ gilt

$$\begin{aligned} g(0)(\tilde{\sigma}^\top x_w) &= r \cdot \frac{1}{\sqrt{\bar{\sigma}}} \tilde{\sigma}^\top x_w = r \cdot \sum_{j=1}^n \sqrt{\tilde{\sigma}_j} \cdot \underbrace{\frac{\sqrt{\tilde{\sigma}_j}}{\sqrt{\bar{\sigma}}}}_{\geq 1} (x_w)_j \\ &\geq r \cdot \sum_{j=1}^n \sqrt{\tilde{\sigma}_j} (x_w)_j \overset{x_w \in \{0,1\}^n}{=} r \cdot \sum_{j=1}^n \sqrt{\tilde{\sigma}_j (x_w)_j} \\ &\geq r \cdot \sqrt{\tilde{\sigma}^\top x_w} = f(\tilde{\sigma}^\top x_w) - f(0). \end{aligned}$$

Es gilt somit für alle $w \in \mathcal{W}$

$$z(w) \geq c^\top x_w + f(\tilde{\sigma}^\top x_w) \geq \min_{x \in X} \big(c^\top x + f(\tilde{\sigma}^\top x) \big) = z^*$$

und somit die Behauptung. $\square$

Bemerkung: Dieser Beweis lässt sich auf beliebige konkave Funktionen f verallgemeinern.

Im Allgemeinen liefert Theorem 4.10 keine polynomielle Reduktion, da W exponentielle Größe haben kann. Das Vorgehen ist jedoch polynomiell, wenn alle σ_j gleich sind, d.h. wenn $\sigma_1 = \ldots = \sigma_n = s > 0$ gilt und damit

$$\xi(x) = r \cdot \sqrt{\tilde{\sigma}^\top x} = r \cdot s \cdot \sqrt{\mathbf{1}^\top x} = r \cdot s \cdot \|x\|_2$$

mit $\mathbf{1} = (1, \ldots, 1)^\top$.

Folgerung 4.11. *Sei $\tilde{\sigma} = (s^2, \ldots, s^2)^\top$. Dann ist $z^* = \min_{w=0,1,\ldots,n} z(s^2 w)$.*

Beweis. Es gilt

$$\mathcal{W} = \{\tilde{\sigma}^\top x : \ x \in \{0,1\}^n\} = \{0, s^2, \ldots, ns^2\}$$

wegen $\tilde{\sigma} = (s^2, \ldots, s^2)^\top$. Also folgt Folgerung 4.11 aus Theorem 4.10. $\square$

In diesem Fall kann man $z(s^2 w)$ wie folgt berechnen:

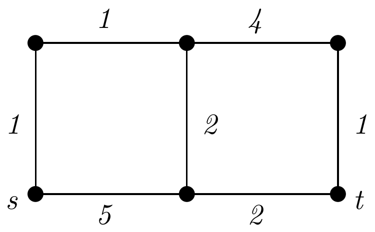
Es gilt $\bar{\sigma} = s^2$. Also ist für $\mathbf{w} = \mathbf{0}$:

$$\begin{aligned} z(0) &= \min_{x \in X} (c + g(0)\tilde{\sigma})^\top x + r\sqrt{0} - 0 \cdot g(0) \\ &= \min_{x \in X} (c + \frac{r}{s} \cdot \mathbf{1} s^2)^\top x \\ &= \min_{x \in X} (c + rs \cdot \mathbf{1})^\top x. \end{aligned}$$

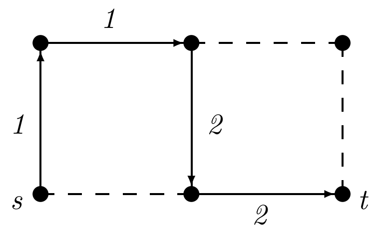
Für $\mathbf{w} > \mathbf{0}$ gilt

$$\begin{aligned} z(s^2 w) &= \min_{x \in X} (c + g(s^2 w)\tilde{\sigma})^\top x + r\sqrt{s^2 w} - s^2 w \cdot g(s^2 w) \\ &= \min_{x \in X} (c + \frac{1}{2} r \frac{1}{s\sqrt{w}} \cdot \mathbf{1} s^2)^\top x + rs\sqrt{w} - s^2 w \cdot \frac{1}{2} r \frac{1}{s\sqrt{w}} \\ &= \min_{x \in X} (c + \frac{rs}{2\sqrt{w}} \cdot \mathbf{1})^\top x + \frac{1}{2} rs\sqrt{w}. \end{aligned}$$

Beispiel 4.8 (Kürzeste Wege)**.** *Es sei folgender Graph G gegeben:*



Dann ist



der (nominal) kürzeste Weg von s nach t in G. Wir betrachten nun eine ellipsoidale Unsicherheit mit $r = 1$, $\sigma_j = \frac{2}{3}$ für alle j. Die Standardabweichung aller Kosten ist also $\frac{2}{3}$. Wir suchen eine robust optimale Lösung. Es gilt somit

$$\begin{aligned} \mathcal{D} &= \left\{ d \in \mathbb{R}^7 : \sqrt{\sum_{j=1}^{7} (\frac{3}{2} d_j)^2} \le 1 \right\} = \left\{ d \in \mathbb{R}^7 : \ \|d\|_2 \le \frac{2}{3} \right\}, \\ \xi(x) &= r \cdot s \cdot \sqrt{\mathbf{1}^\top x}. \end{aligned}$$

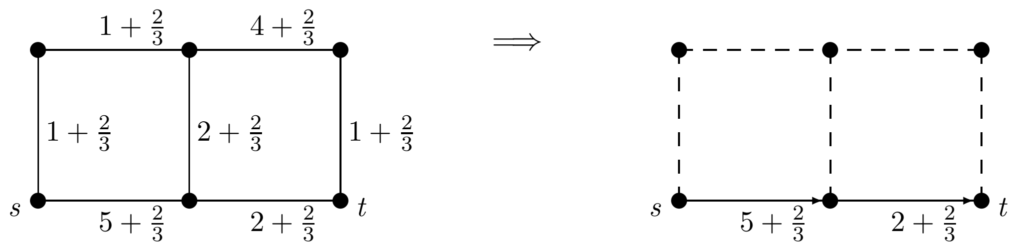
Somit ist das robuste Problem $(RBP - E)$

$$z^* = \min_{x \in X} c^\top x + \frac{2}{3}\sqrt{\mathbf{1}^\top x}$$

mit der Menge X aller Wege von s nach t. Nach Folgerung 4.11 gilt

$$z^* = \min_{w=1,\dots,7} z(\frac{4}{9} w).$$

Es ist $z(0) = \min_{x \in X}(c + \frac{2}{3}\mathbf{1})^\top x$. *Es muss also ein kürzester Weg für neue Kosten berechnet werden:*

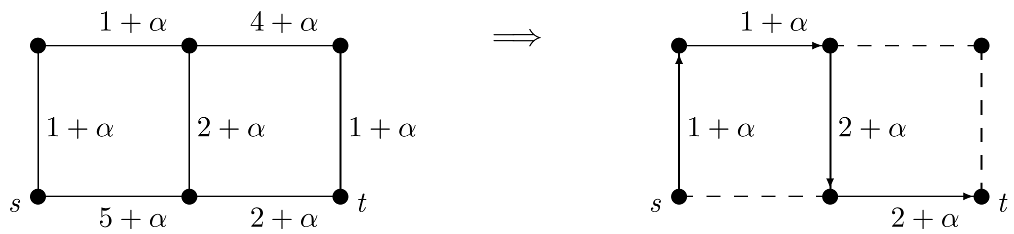


Die Länge des kürzesten Weges ist also $z(0) = 8 + \frac{1}{3}$.

Für $w > 0$ *ist*

$$z(\frac{4}{9}w) = \min_{x \in X}(c + \frac{1}{3\sqrt{w}}\mathbf{1})^\top x + \frac{\sqrt{w}}{3}.$$

Es werden also wieder kürzeste Wege berechnet, wobei die Kosten aller Kanten um einen Wert $\alpha = \frac{1}{3\sqrt{w}}$ *erhöht sind. Wegen* $\alpha < 0.5$ *erhalten wir für alle* $w > 0$ *den folgenden kürzesten Weg:*



Es gilt $z(\frac{4}{9}w) = 6 + \frac{4}{3\sqrt{w}} + \frac{\sqrt{w}}{3}$ *für* $w = 1, \ldots, 7$. *Dies ist eine Funktion von* $w > 0$. *Ihre Ableitung ist*

$$\frac{1}{6\sqrt{w}}\left(\frac{-4}{w} + 1\right),$$

d.h. die Funktion ist monoton fallend für $w \leq 4$ *und monoton steigend für* $w \geq 4$. *Das Minimum wird also bei* $w = 4$ *angenommen. Somit ist*

$$z^* = \min\left\{z(0), z(\frac{4}{9}4)\right\} = \min\left\{8 + \frac{1}{3}, 7 + \frac{1}{3}\right\} = 7 + \frac{1}{3}.$$

Den robust optimalen Weg erhält man also mit $w = 4$.

Beispiel 4.9 (Rucksackproblem)**.** *Wir verwenden wieder das Problem aus Beispiel 4.6. Es sei also* $b = 14$ *und*

$$\begin{array}{llll} a_1 = 12 & a_2 = 6 & a_3 = 6 & a_4 = 4 \\ c_1 = 200 & c_2 = 100 & c_3 = 75 & c_4 = 50. \end{array}$$

(Um die Notationen in den Formeln nicht zu verwechseln, werden hier die Gewichte mit c_i statt w_i bezeichnet.) Wir betrachten dieselben Unsicherheiten wie vorher, aber jetzt ellipsoidal. Es sei also $\sigma_1 = \Delta c_1 = 150$, $\sigma_2 = \Delta c_2 = 60$, $\sigma_3 = \Delta c_3 = 50$ und $\sigma_4 = \Delta c_4 = 5$.

Für $r = 1$ ist das robuste Problem $(RBP - E)$

$$z^* = \max_{x \in X} c^\top x - \sqrt{150^2 x_1 + 60^2 x_2 + 50^2 x_3 + 5^2 x_4}$$

mit $X = \{x \in \{0,1\}^4 : \ a^\top x \le b\}$. Es gilt nun

$$z^* = \max_{w \in \mathcal{W}} \max_{x \in X} \left((c + g(w)\tilde{\sigma})^\top x + r\sqrt{w} - rg(w) \right)$$

nach Theorem 4.10. Leider besitzt die Menge

$$\mathcal{W} = \{150^2 x_1 + 60^2 x_2 + 50^2 x_3 + 5^2 x_4 : \ x \in \{0,1\}^4\}$$

schon 2^4 Elemente. Wir müssen also 16 Rucksackprobleme lösen. Außerdem sind diese NP-schwer.

Bemerkung: Im Allgemeinen ist $(RBP - E)$ ein Optimierungsproblem über einem Lorentzkegel, aber mit zusätzlichen Ganzzahligkeitsbedingungen. Das Problem ist NP-schwer, aber es gibt Software (aktuelles Forschungsgebiet).

4.6.3 Korrelierte Kosten

Wir wollen nun Unsicherheiten untersuchen, wenn die Koeffizienten der Zielfunktion korreliert sind.

Solche Probleme treten z.B. bei der Verkehrsplanung auf, da die Stauwahrscheinlichkeiten einzelner Straßen nicht unabhängig voneinander sind.

Wir betrachten nun die **Kovarianzmatrix** $\Sigma = (\sigma_{ij})$ der Zufallsvariablen Δc_j, definiert durch

$$\sigma_{ij} = E\left((\Delta c_i - E(\Delta c_i)) (\Delta c_j - E(\Delta c_j)) \right).$$

Bedeutung bei $i \neq j$:

$\sigma_{ij} > 0 \ \Rightarrow$ Δc_i und Δc_j bewegen sich tendenziell in dieselbe Richtung.
$\sigma_{ij} < 0 \ \Rightarrow$ Δc_i und Δc_j bewegen sich tendenziell entgegengesetzt.
$\sigma_{ij} = 0 \ \Rightarrow$ Es besteht keine Korrelation.

Wir betrachten jetzt

$$\mathcal{D} = \left\{ d \in \mathbb{R}^n : \ \sqrt{d^\top \Sigma^{-1} d} \le r \right\}.$$

Falls die Variablen unkorreliert sind, ergibt sich $\Sigma = diag(\sigma_i^2)_{i=1}^n$, d.h. Σ ist eine Diagonalmatrix mit den Einträgen σ_i^2 auf der Hauptdiagonalen. Also ist wie in Abschnitt 4.6.1

$$\mathcal{D} = \left\{ d \in \mathbb{R}^n : \sqrt{\sum_{i=1}^{n} \frac{d_i^2}{\sigma_i^2}} \leq r \right\}.$$

Beispiel 4.10. *Sei $n = 2$ und seien Δc_1 und Δc_2 negativ korreliert mit*

$$\Sigma = \begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}.$$

Dann gilt $\Sigma^{-1} = \begin{pmatrix} 4/3 & 2/3 \\ 2/3 & 4/3 \end{pmatrix}$. Also ist $\mathcal{D} = \{d : \sqrt{\frac{4}{3}(d_1^2 + d_1 d_2 + d_2^2)} \leq r\}$. Durch den Term $d_1 d_2$ wird die Ellipse $\mathcal{D}$ im positiven Quadranten kleiner, d.h. die Unsicherheit wird dort geringer.

Für eine beliebige Kovarianzmatrix Σ und für $\mathcal{D} = \left\{ d \in \mathbb{R}^n : \sqrt{d^\top \Sigma^{-1} d} \leq r \right\}$ gilt $\xi(x) = r \cdot \sqrt{x^\top \Sigma x}$. Die robuste Aufgabe lautet somit

$$(RBP - K) \qquad \min_{x \in X} \left(c^\top x + r \cdot \sqrt{x^\top \Sigma x} \right).$$

Theorem 4.12. *Das Problem $(RBP - K)$ kann NP-schwer sein, auch wenn (BP) polynomiell lösbar ist für alle Vektoren c.*

Beweis. Nach Theorem 4.3 ist das Kürzeste-Wege-Problem mit zwei Szenarien NP-schwer:

$$\min_{x \in X} \max\{c_1^\top x, c_2^\top x\}.$$

Wir wollen dieses Problem auf $(RBP - K)$ reduzieren.
Wir setzen $\Sigma = (c_1 - c_2)(c_1 - c_2)^\top$, $c = (c_1 + c_2)/2$ und $r = 0.5$. Dann gilt

$$\begin{aligned}
& \max\{c_1^\top x, c_2^\top x\} \\
= \; & \max\left\{ \frac{c_1^\top x + c_2^\top x}{2} + \frac{c_1^\top x - c_2^\top x}{2}, \frac{c_1^\top x + c_2^\top x}{2} - \frac{c_1^\top x - c_2^\top x}{2} \right\} \\
= \; & \frac{c_1^\top x + c_2^\top x}{2} + \max\left\{ \frac{c_1^\top x - c_2^\top x}{2}, -\frac{c_1^\top x - c_2^\top x}{2} \right\} \\
= \; & \frac{c_1^\top x + c_2^\top x}{2} + \left| \frac{c_1^\top x - c_2^\top x}{2} \right| \\
= \; & c^\top x + \frac{1}{2} \sqrt{x^\top (c_1 - c_2)^\top (c_1 - c_2) x} \\
= \; & c^\top x + r \cdot \sqrt{x^\top \Sigma x}.
\end{aligned}$$

□