(4) Heuristiky a dekompozice

Příklad 4.1. Vyřešte následující binární program odvozením a kombinováním logických nerovností:

$$
\begin{aligned}
7x_1 + 3x_2 - 4x_3 - 2x_4 &\le 1,\\
-2x_1 + 7x_2 + 3x_3 + x_4 &\le 6,\\
- 2x_2 - 3x_3 - 6x_4 &\le -5,\\
3x_1 - 2x_3 &\ge -1,\\
x_1, x_2, x_3, x_4 &\in \{0,1\}.
\end{aligned}
$$

Řešení:
Viz Příklad 2.29 ve skriptech.

Příklad 4.2. Aplikujte subgradientní algoritmus pro Lagrangeovu relaxaci na následující celočíselný program:

$$
\begin{array}{lrl}
\max & 16x_1 + 10x_2 + 4x_4 & \\
\text{za podm.} & 8x_1 + 2x_2 + x_3 + 4x_4 & \le 10,\\
& x_1 + x_2 & \le 1,\\
& x_3 + x_4 & \le 1,\\
& x_1, x_2, x_3, x_4 & \in \{0,1\}.
\end{array}
$$

Řešení:
Subgradientní algoritmus (viz Algoritmus 5.11 ve skriptech) využívá pro nalezení těsného horního odhadu na optimální hodnotu iterativní řešení relaxované úlohy

$$\max\ c^T x + u^T(b - Ax) \text{ za podm. } x \in Q, \tag{P_u}$$

kde $Q = \{x \in \mathbb{R}^n : Dx \le d, x \ge 0, x \in \mathbb{Z}^n\}$ je přípustná množina pro „hezké“ podmínky původní úlohy. Můžeme uvažovat relaxaci první omezující podmínky, pro kterou dostaneme úlohu (P_u) ve tvaru

$$
\begin{array}{ll}
\max & (16 - 8u)x_1 + (10 - 2u)x_2 - ux_3 + (4 - 4u)x_4 + 10u\\
\text{za podm.} & x_1 + x_2 \le 1,\\
& x_3 + x_4 \le 1,\\
& x_1, x_2, x_3, x_4 \in \{0,1\}.
\end{array}
$$

Dále zvolme počáteční hodnotu u, např. $u^1 = 0$. V k-tém kroku algoritmu řešíme úlohu (P_u) pro $u = u^k$, pro kterou najdeme optimální řešení x_k a nastavíme

$$u^{k+1} = \max(u^k - \alpha_k(b - Ax^k), 0)$$

pro zvolenou délku kroku $\alpha_k > 0$.

Zkusme nejdřív aplikovat algoritmus s konstatní délkou kroku $\alpha_k = 1$ pro všechna k. V první iteraci tedy pro zvolené u^1 podmínku $8x_1 + 2x_2 + x_3 + 4x_4 \le 10$ vůbec neuvažujeme. Snadno nahlédneme, že optimálním řešením relaxované celočíselné úlohy je $x^1 = (1, 0, 0, 1)$. V další iteraci řešíme úlohu (P_u) pro hodnotu

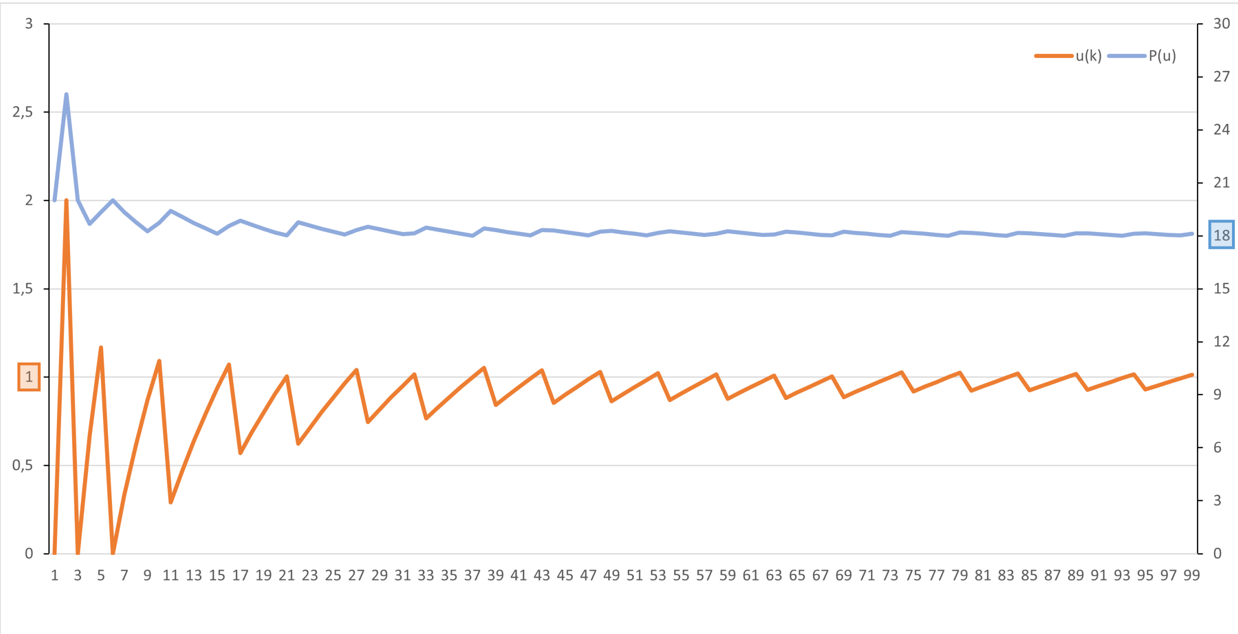
$$u^2 = u^1 - \alpha_1 \cdot (10 - 8x_1^1 - 2x_2^1 - x_3^1 - 4x_4^1) = 0 - 1 \cdot (10 - 8 - 4) = 2,$$

tedy úlohu s účelovou funkcí max $6x_2 - 2x_3 - 4x_4 + 20$. Tato úloha má optimální řešení $x^2 = (0,1,0,0)$. Pro $k = 3$ pak dostaneme $u^3 = \max(2 - 1 \cdot (10-2), 0) = 0$ a řešíme stejnou úlohu (P_u) jako v první iteraci. Algoritmus tedy bude oscilovat mezi $u = 0$ s optimální hodnotou relaxace 20 a $u = 2$ s optimální hodnotou 26.

Aplikujme algoritmus ještě pro délku kroku $\alpha_k = \frac{1}{k}$, pro kterou máme zaručenou konvergenci. Průběh prvních 5 iterací algoritmu je v tomto případě následující:

$$\begin{aligned}
u^1 &= 0, & x^1 &= (1,0,0,1), & f &= 20,\\
u^2 &= 2, & x^2 &= (0,1,0,0), & f &= 26,\\
u^3 &= 0, & x^3 &= (1,0,0,1), & f &= 20,\\
u^4 &= \frac{2}{3}, & x^4 &= (1,0,0,1), & f &= \frac{56}{3} = 18.\overline{6},\\
u^5 &= \frac{7}{6}, & x^5 &= (0,1,0,0), & f &= \frac{58}{3} = 19.\overline{3}.
\end{aligned}$$

Další průběh algoritmu pro prvních 100 iterací znázorňuje následující graf.



Vidíme, že hodnoty u_k postupně konvergují k optimální hodnotě $u = 1$, hodnoty účelové funkce (P_u) konvergují k 18. Můžeme snadno nahlédnout, že optimální hodnota původního programu je 16 pro řešení $x^* = (1,0,0,0)$, případně $x^* = (1,0,1,0)$. Relaxované omezení je pro tato optimální řešení splněno s ostrou nerovností, nemáme tedy zaručený nejtěsnější možný odhad (dle postačující podmínky z Tvrzení 5.8).

Příklad 4.3. Aplikujte metodu column generation pro řešení řezného problému:

- firma Karlovy trubky, s.r.o, skladuje kovové trubky délky 218 cm,
- zákazník zadal objednávku na 44 ks trubek délky 81 cm, 3 ks trubek délky 70 cm a 48 ks trubek délky 68 cm.

Řešení:
Úlohu můžeme modelovat jako celočíselný program ve tvaru

$$
\begin{aligned}
\min \sum_j x_j& \\
\text{za podm. } \sum_j a_{1j} x_j &\geq 44, \\
\sum_j a_{2j} x_j &\geq 3, \\
\sum_j a_{3j} x_j &\geq 48, \\
x_j \geq 0,\ x_j &\in \mathbb{Z},
\end{aligned}
$$

kde x_j reprezentuje počet 218cm trubek nařezaných podle vzoru j a pro konkrétní vzor j značí a_{1j} počet získaných trubek délky 81 cm, a_{2j} počet trubek délky 70 cm a a_{3j} počet trubek délky 68 cm.

Pro metodu column generation uvažujeme v programu pouze podmnožinu proměnných I, pro kterou nalezneme optimální řešení x_I^* menšího programu s účelovou funkcí c_I a maticí omezení A_I. Doplněním řešení $x_i^* = 0$ pro $i \notin I$ dostaneme přípustné řešení původní úlohy. Jako počáteční množinu I můžeme zvolit řezné vzory, kterými z dlouhé trubky vyrobíme 1 ks některé z objednaných trubek (příp. největší možný počet objednaných trubek jednoho typu, nebo i jiné vhodné vzory). Pro tuto množinu vzorů dostaneme zjednodušený program

$$\min x_1 + x_2 + x_3 \text{ za podm. } x_1 \geq 44,\ x_2 \geq 3,\ x_3 \geq 48,\ x_I \geq 0$$

s optimálním řešením $x_I^* = (44, 3, 48)$. Pro duální úlohu

$$\max 44y_1 + 3y_2 + 48y_3 \text{ za podm. } y_1 \leq 1,\ y_2 \leq 1,\ y_3 \leq 1,\ y \geq 0$$

dostaneme optimální řešení $y^* = (1, 1, 1)$. Dále musíme ověřit, zda y^* splňuje všechna omezení původní duální úlohy (tedy, zda je x_I^* optimální).

Omezení původní duální úlohy jsou ve tvaru $a_{1j}y_1 + a_{2j}y_2 + a_{3j}y_3 \leq 1$ pro řezný vzor j. Chceme tedy nalézt řezný vzor popsaný koeficienty (a_{1j}, a_{2j}, a_{3j}), pro který bude hodnota levé strany této nerovnosti s $y^* = (1, 1, 1)$ největší možná (t.j. dostaneme nejvíce porušenou podmínku). Všechny přípustné řezné vzory můžeme popsat podmínkou $81a_{1j} + 70a_{2j} + 68a_{3j} \leq 218$. Pro nalezení vhodného řezného vzoru j, pro který přidáme proměnnou x_j do uvažované množiny I, tedy řešíme podúlohu s proměnnými a_{1j}, a_{2j}, a_{3j} ve tvaru

$$\max 1a_{1j} + 1a_{2j} + 1a_{3j} \text{ za podm. } 81a_{1j} + 70a_{2j} + 68a_{3j} \leq 218,\ a_{ij} \in \mathbb{Z}_+.$$

Řešením podúlohy je vzor $(0, 0, 3)$ (nařežeme 3 trubky délky 68 cm), pro který dostaneme porušenou nerovnost $0y_1^* + 0y_2^* + 3y_3^* \not\leq 1$ (vidíme, že optimální hodnota podúlohy je větší než 1, což je hodnota pravé strany v duálním omezení). První získané řešení $x_I^* = (44, 3, 48)$ tedy není optimální, přidáme do množiny I nový řezný plán typu $(0, 0, 3)$ a vyřešíme další zjednodušený program pro 4 proměnné:

$$\min x_1 + x_2 + x_3 + x_4 \text{ za podm. } x_1 \geq 44,\ x_2 \geq 3,\ x_3 + 3x_4 \geq 48,\ x_I \geq 0.$$

Pro tento program najdeme optimální řešení $x_I^* = (44, 3, 16)$ a optimum duálního zjednodušeného programu $y^* = (1, 1, \frac{1}{3})$. Pro toto y^* ovšem můžeme také najít porušenou podmínku původního duálu, vyřešením podúlohy ve tvaru

$$\max 1a_{1j} + 1a_{2j} + \frac{1}{3}a_{3j} \text{ za podm. } 81a_{1j} + 70a_{2j} + 68a_{3j} \leq 218,\ a_{ij} \in \mathbb{Z}_+.$$

Získáme řezný vzor $(0, 3, 0)$ (nařežeme 3 trubky délky 70 cm), který přidáme do množiny uvažovaných vzorů. Tímto způsobem postupně přidáváme do uvažované množiny další řezné vzory (a do programu odpovídající proměnné x_j), než najdeme optimální řešení x_I^* s optimálním řešením y^* zjednodušeného duálního programu, které bude také optimálním řešením původního duálního programu.

Příklad 4.4. Využijte Bendersovu dekompozici pro řešení smíšené celočíselné úlohy:

$$\begin{array}{lrcrcl}
\max & -5x & + & 3y & & \\
\text{za podm.} & -2x & + & y & \leq & 0, \\
 & -x & + & 3y & \leq & 13, \\
 & & & y & \leq & 10, \\
 & & & x, y & \geq & 0, \\
 & & & y & \in & \mathbb{Z}.
\end{array}$$

Řešení:

Viz Příklad 5.3 ve skriptech.