2. CVIČENÍ Z OPTIMALIZACE

Formulace LPček a IPček

Příklad první S pomocí znalostí z přednášky rozhodněte o platnosti následujících tvrzení:

- Každá úloha LP (polynomiální velikosti) je řešitelná v polynomiálním čase.
- Každá úloha IP (polynomiální velikosti) je NP-těžká.
- Pokud optimální řešení LP existuje, tak existuje jedno ve vrcholu mnohostěnu podmínek.
- Každé optimální řešení LP (pokud existuje) je ve vrcholu mnohostěnu podmínek.

Příklad druhý Pro každý zadaný problém navrhněte lineární nebo celočíselný program, který jej řeší. (Pokud přijdete na lineární program, je to určitě lepší, ale nejde to vždy.)

- (Klasický problém.) Pekárna peče chleby, housky, bagety a koblihy. K upečení jednoho chleba potřebuje půl kila mouky, 10 vajec a 50 g soli. Na jednu housku je zapotřebí 150 g mouky, 2 vejce a 10 g soli. Na bagetu potřebuje 230 g mouky, 7 vajec a 15 g soli. Na jednu koblihu je třeba 100 g mouky a 1 vejce. Pekárna má k dispozici 5 kilo mouky, 125 vajec, a půl kila soli. Za jeden chleba získá pekárna 20 korun, za housku 2 koruny, za bagetu 10 korun a za koblihu 7 korun. Pekárna se snaží vydělat co nejvíce. Jak zjistí, kolik chlebů, housek, baget a koblih má upéct?
- (Dopravní problém jinak.) V Kocourkově je n pekáren a m obchodů. Každý den i-tá pekárna upeče p_i rohlíků a j-tý obchod prodá o_j rohlíků. Převoz jednoho rohlíku z i-té pekárny do j-tého obchodu stojí c_{ij} korun. (Až sem je to klasický dopravní problém, který byl i na přednášce.)

  Jenže! Praxe v Kocourkově ukázala, že když i-tá pekárna zásobuje j-tý obchod, tak musí pro tuto trasu zajistit logistiku, která je stojí l_{ij}. Logistiku l_{ij} je nutné platit pouze tehdy, když i-tá pekárna zásobuje j-tý obchod nenulovým počtem rohlíků, a její cena nezávisí na počtu převážených rohlíků. I nadále je nutné platit přepravné c_{ij}.

  Nalezněte takovou distribuci rohlíků, aby se každá pekárna zbavila všech rohlíků, každý obchod získal (právě) potřebný počet rohlíků a celkové náklady na převoz byly minimální.
- (Oddělování bodů.) V rovině máme zadané červené body $c_1, c_2, \ldots, c_m$ a zelené body $z_1, z_2, \ldots, z_n$. Nalezněte přímku, která odděluje všechny červené body od zelených, případně oznamte její neexistenci.

  Pro jednoduchost máte zaručeno, že žádná svislá ani vodorovná přímka body neodděluje. V tomto příkladu můžete použít několik lineárních programů za sebou (namísto rozboru případů).
- (Souvislost grafu.) Pro zadaný neorientovaný graf G rozhodněme, jestli G je souvislý.

Příklad třetí Mějme zadanou matici A, vektor b a lineární funkci f. Z nich můžeme postavit třeba tento celočíselný program C:

$$\begin{aligned} \max & f(x) \\ Ax &\le b \\ x &\in \{0,1\}^n \end{aligned}$$

Můžeme z nich ale postavit také následující lineární program L:

$$\begin{aligned} \max & f(x) \\ Ax & \leq b \\ x & \in [0,1]^n \end{aligned}$$

Předpokládejme, že oba programy jsou řešitelné. Pojmenujme jedno optimální řešení celočíselného programu x_C^* a jedno optimální řešení lineárního programu x_L^*. Dokažte, že platí následujíci nerovnost:

$$f(x_C^*) \leq f(x_L^*).$$

Příklad čtvrtý Student Josef K. dostal na cvičení z Optimalizace zadaný úkol:

Navrhněte celočíselný program pro problém obchodního cestujíciho, čili pro daný ohodnocený graf $G = (V, E, f)$, kde $f : E \to \mathbb{R}_0^+$, chceme najít Hamiltonovskou kružnici s nejkratší délkou.

Josef K. navrhuje následující řešení:

„Pro každou hranu uv máme proměnnou $x_{uv} \in \{0, 1\}$, cílová funkce je min $\sum_{uv \in E} f(uv) x_{uv}$ a pro každý vrchol u máme podmínku $\sum_{i|ui \in E} x_{ui} = 2$.“

Funguje řešení Josefa K.? Pokud ano, zdůvodněte, pokud ne, zdůvodněte a ještě vymyslete lepší.

Příklad pátý Mějme problém minimální kostry: máme zadaný graf G s hranami ohodnocenými čísly $w_e \geq 0$, a hledáme kostru s nejmenší vahou.

Zkuste najít dva různé celočíselné programy, které oba řeši problém minimální kostry a nejdou na sebe převést jednoduchými úpravami. Programy mohou mít exponenciálně mnoho podmínek.

Nápověda: Minimální kostra je strom, a strom má v grafu mnoho ekvivalentních definic.