4. CVIČENÍ Z APROXIMACÍ

SDP = semidefinitní programování

Příklad první Dokažte ekvivalenci následujících charakterizací positivně semidefinitních (PSD) matic, tedy, že pro reálnou symetrickou matici $A \in \mathbb{R}^{n \times n}$ je následující ekvivalentní:

a) $\forall x \in \mathbb{R}^n$, $x^T A x \geq 0$.
b) Všechna vlastní čísla A jsou nezáporná.
c) Existuje reálná $n \times n$ matice V taková, že $A = V^T V$.

Příklad druhý Nechť podmatice B vznikne z PSD matice A smazáním podmnožiny řádků a *stejné* podmnožiny sloupců. Ukažte, že B je PSD a že $\det B \geq 0$.

Příklad třetí Ukažte, že semidefinitní programy a vektorové programy jsou ekvivalentní: Ke každému semidefinitnímu programu $\mathcal{S}$ najděte vektorový program $\mathcal{V}$ takový, že z každého přípustného řešení $\mathcal{S}$ lze vytvořit přípustné řešení $\mathcal{V}$ se stejnou hodnotou účelové funkce, a naopak.

Příklad čtvrtý Ukažte, že SDP nelze vyřešit přesně. Speciálně najděte semidefinitní program, který má hodnotu účelové funkce omezenou konstantou c a pro každé $c' < c$ existuje přípustné řešení s hodnotou c' (předpokládáme maximalizaci), ale neexistuje přípustné řešení s hodnotou c.

Hint: Zkuste matici

$$X = \begin{pmatrix} x & 1 \\ 1 & y \end{pmatrix} .$$

Napadá vás nějaký jiný důvod, proč nelze vyřešit SDP přesně? Anebo SDP, které nelze v polynomiálním čase ani aproximovat?

Příklad pátý Abychom mohli SDP vyřešit (přibližně) pomocí elipsoidové metody, potřebujeme najít oddělující nadrovinu, pokud nemáme přípustné řešení.

Nechť $\mathcal{S}$ je SDP a $A \in \mathbb{R}^{n \times n}$ matice. Ukažte, že můžeme v polynomiálním čase zjistit, jestli A je přípustné řešení $\mathcal{S}$, a pokud ne, tak najít oddělující nadrovinu. Nadrovina (resp. poloprostor) $\sum_{i,j} c_{ij} x_{ij} \leq b$ odděluje matici A, pokud každé přípustné řešení $\mathcal{S}$ splňuje tuto nerovnost, ale matice A ne.

Hint: Pokud A není přípustné řešení, co může být porušeno?

Příklad šestý Nyní se zaměříme na problém Max 2-SAT, v němž máme množinu klauzulí s nejvýše dvěma literály a cílem je najít ohodnocení proměnných takové, že je splněno co nejvíce klauzulí.

a) Nejprve vytvořte kvadratický program. *Hint:* Může se hodit mít proměnnou $y_0 \in \{-1, 1\}$, která udává, jestli hodnota true je -1, nebo 1.
b) Zrelaxujte kvadratický program na vektorový program, který následně použijte na vytvoření podobného algoritmu jako pro Max Cut na přednášce.
c) Ukažte, že algoritmus má stejný aproximační poměr jako ten pro Max Cut, tedy

$$\alpha = \frac{2}{\pi} \min_{0 \leq \theta \leq \pi} \frac{\theta}{1 - \cos\theta} > 0.87856 .$$

Kromě nerovnosti vyplývající z definice α můžete použít následující nerovnost (umíte ji dokázat?):

$$1 - \frac{\theta}{\pi} \geq \frac{\alpha}{2}(1 + \cos\theta) .$$

Příklad sedmý Vytvořte kvadratické programy pro následující problémy:

Maximální k-řez: Na vstupu máme neorientovaný graf G s váhami na hranách $w\colon E(G) \to \mathbb{R}^+$. Cílem je rozdělit všechny vrcholy grafu do k hromádek $V_1, V_2, \ldots, V_k$ tak, aby součet vah všech mezihromádkových hran byl co největší možný.