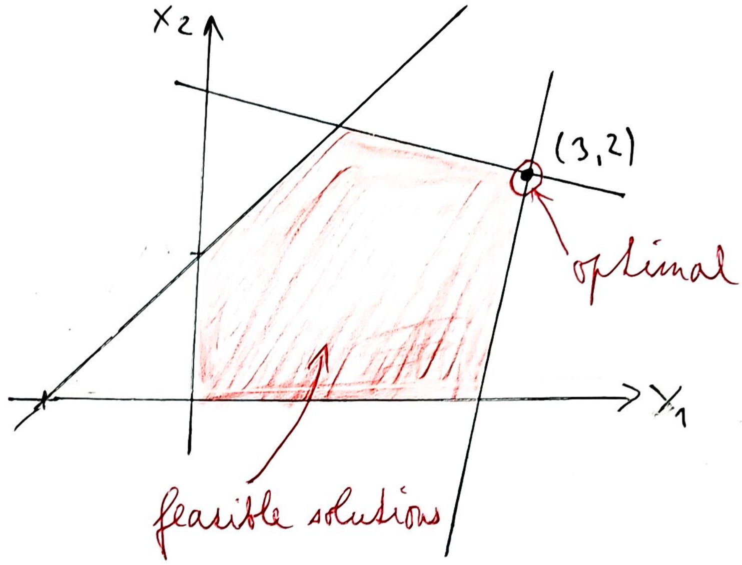
LINEÁRNÍ PROGRAMOVÁNÍ

Příklad: chceme max. x_1+x_2, kde

$$\left.\begin{aligned} x_1 &\ge 0 \\ x_2 &\ge 0 \\ x_2 - x_1 &\le 1 \\ x_1 + 6x_2 &\le 15 \\ 4x_1 - x_2 &\le 10 \end{aligned}\right\} \text{lineární program}$$



→ červená oblast to splňuje

→ bod (3,2) je optimum

Obecný problém:

chceme max. $c^T x$ ← objective function $c, x \in \mathbb{R}^n$

za podmínek

$$\left.\begin{aligned} a_1^T x &\le b_1 \\ a_2^T x &= b_2 \\ a_3^T x &\le b_3 \\ &\vdots \\ a_m^T x &\ge b_m \end{aligned}\right\} \text{constraints}$$

… $Ax \lesseqgtr b$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

Def: $\beta \in \mathbb{R}^n$ is a feasible solution ≡ satisfies all constraints.

Def: $\alpha \in \mathbb{R}^n$ is an optimal solution ≡ $\forall \beta$ feasible sol. β: $c^T\alpha \ge c^T\beta$.

Def: Feasible LP: $\exists$ a feasible solution.

 není f.

Def: Bounded LP: $\exists \gamma \in \mathbb{R}^n$ s.t. $c^T x \le \gamma$ $\forall$ feasible x.

 není b.

→ zajímá nás případ, kdy LP je F & B.

Příklady:

① lineární regrese: body (x_i, y_i)

$$\min \sum_i (\underbrace{ax_i + b}_{\text{predikce}} - \underbrace{y_i}_{\text{actual}})^2$$ ← least squares

vars: a, b

ale tohle není lineární

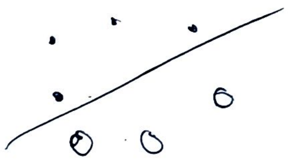
$\Rightarrow \min \sum_i |ax_i + b - y_i|$ ... taky není

$\Rightarrow \sum_i |\ldots| = l_1 + l_2 + \ldots + l_n$

$$\left.\begin{aligned} l_i &\geq ax_i + b - y_i \\ l_i &\geq -(ax_i + b - y_i) \end{aligned}\right\} \; l_i \geq \max(\_\_, -\_\_) = |\_\_| \text{ v optimu}$$

vars: $a, b, l_1, \ldots l_n$

② separace bodů



černé: $p_1, \ldots p_m$
bílé: $q_1 \ldots q_m$

→ rozdělit přímkou

- černé nahoře $y(p_i) > ax(p_i) + b$
- bílé dole $y(q_i) < ax(q_i) + b$

! co ostrými nerovnostmi? → přidat slack variable δ

$$y(p_i) \geq ax(p_i) + b + \delta$$
$$y(q_i) \leq ax(p_i) + b - \delta$$ → max δ

→ nemusí to být přímka

→ parabola: $y \geq ax^2 + bx + c + \delta$

→ obecně:

max δ

vars: $\delta, a_1, \ldots a_k$

$\varphi_1, \ldots \varphi_k$ jsou funkce ← přímky, paraboly, sin, cokoliv

s.t.: $y(p_i) \geq a_1 \varphi_1(p_i) + a_2 \varphi_2(p_i) + \ldots + a_k \varphi_k(p_i) + \delta$
$y(q_i) \leq a_1 \varphi_1(q_i) + a_2 \varphi_2(q_i) + \ldots + a_k \varphi_k(q_i) - \delta$

STANDARD FORM

Def: Kanonický tvar: $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$

$$\max \ c^T x \quad , x \in \mathbb{R}^n$$
$$\text{s.t.} \ Ax \le b$$

EQUATIONAL FORM

Def: Rovnicový tvar:

$$\max \ c^T x \quad , x \ge 0$$
$$\text{s.t.} \ Ax = b$$

Lemma: Každý LP lze převést do rovnicového tvaru. (a i do kanon.)

Dk: příkladem

$$\min: 3x_1 - 2x_2 \rightsquigarrow \max \ -3x_1 + 2x_2$$
$$\text{s.t.:} \ 2x_1 - x_2 \le 4 \rightsquigarrow 2x_1 - x_2 + \delta_1 = 4, \quad \delta_1 \ge 0$$
$$x_1 + 3x_2 \ge 5 \rightsquigarrow -x_1 - 3x_2 + \delta_2 = 5, \quad \delta_2 \ge 0$$
$$x_2 \ge 0$$
$$x_1 \in \mathbb{R} \rightsquigarrow y_1, z_1 \ge 0 \ \& \ \text{substituce} \ x_1 \rightsquigarrow y_1 - z_1$$

$$\Rightarrow \max \ -3(y_1 - z_1) + 2x_2$$
$$\text{s.t.} \ 2(y_1 - z_1) - x_2 + \delta_1 = 4$$
$$-y_1 + z_1 - 3x_2 + \delta_2 = 5$$
$$x_2, y_1, z_1, \delta_1, \delta_2 \ge 0$$

Integer programming

$\max \quad c^T x$

s.t. $Ax \leq b$, $x \in \mathbb{Z}^n$ → NP-úplný problém

↳ lze tak modelovat SAT

→ binární LP: $x \in \{0,1\}^n$

Def. LP-relaxace toho IP je $Ax \leq b$, $x \in \mathbb{R}^n$ nebo (bin) $0 \leq x \leq 1$.

↳ může dát nepoužitelnou aproximaci ale ukáže se, že pro určitou třídu problémů najde optimum

① Maximum-weight matching v BIPARTITNÍM grafu

→ $G=(V,E)$, hrana e má váhu $w_e \geq 0$

→ proměnné x_e ... hrana e je v párování ... $x_e \in \{0,1\}$

$\max \sum_e w_e x_e$

s.t. $\forall v \in V: \sum_{e \ni v} x_e = 1$ ← perfektní párování, ale vážené

$\forall e \in E: x_e \in \{0,1\}$ → LP-relaxace: $0 \leq x_e \leq 1$

Věta: Pokud LP-relaxace má přípustné řešení, tak má alespoň 1 celočíselné optimální řešení, které je optimální i pro ten původní IP.

Dk: Nechť x^* je optimální řešení LP relaxace.

→ označme $w(x^*) := \sum_e w_e x_e^*$

$k(x^*) :=$ # neceločíselných složek vektoru x^*

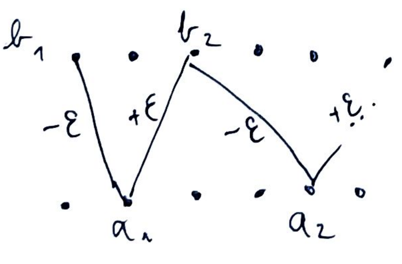
→ pokud $k(x^*) = 0$: vyhráli jsme

→ pro $k(x^*) > 0$ vyrobíme $\tilde{x}$ t.ž. $w(\tilde{x}) = w(x^*)$ & $k(\tilde{x}) < k(x^*)$.

↳ ∃ hrana $e_1 = \{a_1, b_1\}$ t.ž. $0 < x_{e_1} < 1$

→ musí být splněna podmínka $\sum_{e \ni a_1} x_e = 1$

⇒ ∃ hrana $e_2 = \{a_1, b_2\}$, $b_2 \neq b_1$ t.ž. $0 < x_{e_2} < 1$



→ tohle opakuji, nakonec vznikne cyklus (graf je konečný)

↳ bipartitní graf ⇒ cyklus sudé délky

→ označme ten cyklus $e_1, e_2, \dots e_t$

↳ trochu změníme váhy těch hran, čímž vznikne nová celočíselná proměnná

→ lichým hranám odečtu ε

→ sudým hranám přičtu ε ⇝ $\tilde{X}$

→ tím zřejmě zůstane zachováno $\sum_{e \ni v} \tilde{X}_e = 1$

→ pro dostatečně malé ε platí i $0 \leq \tilde{X}_e \leq 1$

→ co objective function?

$$w(\tilde{x}) = \sum_e w_e \tilde{X}_e = w(x^*) + \varepsilon \sum_{i=1}^{t} (-1)^i w_{e_i} = w(x^*) + \varepsilon \Delta$$

↳ pokud $\Delta > 0 \Rightarrow$ volím $\varepsilon > 0 \Rightarrow w(\tilde{x}) > w(x^*)$ ↯

$\Delta < 0 \Rightarrow$ volím $\varepsilon < 0 \Rightarrow$ ——"——

⟹ zřejmě $\Delta = 0 \Rightarrow w(\tilde{x}) = w(x^*) \Rightarrow \tilde{X}$ je optimální

⟹ zvolím největší $\varepsilon > 0$ aby $\tilde{x}$ bylo stále přípustné

⟹ pak musí být nějaká hrana z toho cyklu celočíselná, tedy $\mathcal{E}(\tilde{X}) < \mathcal{E}(x^*)$ ■

(2) Min vertex cover – 2-aproximace

$$\min \sum_v X_v$$

s.t. $X_u + X_v \geq 1$, $\forall uv \in E$

$X_v \in \{0,1\}$, $\forall v \in V$

Věta: Řešení LP relaxace je 2-aproximace (nejhůře 2× horší) tohoto IP.

Dk: Nechť X^* je optimum relaxace a definujme

$$S_{LP} := \{v \in V \mid X_v^* \geq \tfrac{1}{2}\}$$

👁 toto je vertex-cover ∵ $\forall uv \in E: X_u^* + X_v^* \geq 1 \Rightarrow X_u^* \geq \frac{1}{2}$ nebo $X_v^* \geq \frac{1}{2}$

Nechť S_{OPT} je optimální vertex cover, tvrdíme

$$|S_{LP}| \leq 2 \cdot |S_{OPT}|$$

→ tomuto S_{OPT} odpovídá nějaké přípustné celočíselné řešení $\tilde{X}$

1. $\sum_v X_v^* \leq \sum_v \tilde{X}_v = |S_{OPT}|$ ∵ X^* je optimální pro relaxaci

2. $\sum_v X_v^* \geq \frac{1}{2} |S_{LP}|$ ∵ $\forall X_v \in S_{LP}$ přispěje alespoň $\frac{1}{2}$ do té sumy

③ Max independent set

$$\max \sum_v x_v \quad , \; x_v \in \{0,1\}$$

$$\text{s.t. } \forall uv \in E: \; x_u + x_v \leq 1$$

→ vypadá stejně jako min cover, jen opačně verze

→ relaxace $x_v \in [0,1]$

→ $\forall v: x_v := \frac{1}{2}$ je přípustné řešení s obj.f. $= \frac{|V|}{2}$

→ ale úplný graf má největší ind. set. velikosti 1

→ LP relaxace dává max $\geq \frac{|V|}{2} = \frac{n}{2}$

⇒ je to mega špatný

BÁZICKÁ ŘEŠENÍ

→ řeší úlohy LP v equational form

Assumption:

1) systém rovic $Ax = b$ má alespoň 1 řešení ...

2) řádky matice A jsou lineárně nezávislé ⇒ $A \in \mathbb{R}^{m \times n}$ ⇒ $rank(A) = m$

Důvod:

1) easy check (Gaussovka), pokud nemá → konec

2) jinak by nějaká rovnice byla zbytečná

Def: Nechť $A \in \mathbb{R}^{m \times n}$, $B \subseteq \{1, \dots n\}$. $A_B :=$ matice sloupců A z B.

$$A = \begin{pmatrix} 1 & 5 & 3 & 4 & 6 \\ 0 & 1 & 3 & 2 & 7 \end{pmatrix} \quad , \; B = \{2, 4\} \implies A_B = \begin{pmatrix} 5 & 4 \\ 1 & 2 \end{pmatrix}$$

Def: Mějme LP v eq. form: $Ax = b$, $x \geq 0$, $A \in \mathbb{R}^{m \times n}$ (objective func. nás nezajímá) !!!

$x \in \mathbb{R}^n$ je basic feasible solution $\equiv$

$\exists \; B \subseteq \{1, 2, \dots n\}$, $|B| = m$ t.ž.

1) čtvercová matice $A_B \in \mathbb{R}^{m \times m}$ je regulární (sloupce jsou lin. nez.)

2) $j \notin B \implies x_j = 0$

→ $j \in B \implies x_j =$ basic variables → množině B říkáme báze

$j \notin B \implies x_j =$ non-basic

Lemma: Přípustné řešení x LP v rovnicovém tvaru je bázické $\Leftrightarrow$ sloupce matice A_K jsou lin. nez.

$$K = \{ j \in [n] \mid x_j > 0 \}$$

Dk: $\Rightarrow$: definice bázického řešení

$\Leftarrow$: pokud $|K| = m$, tak vezmeme bázi $B = K$

pokud $|K| < m$, tak vyrobíme B tím, že do K přidáme $m - |K|$ dalších indexů, aby sloupce A_B byly l. nez.

→ vždy přidáme nějaký sloupec, který není ve vektorovém prostoru generovaném aktuální bází

→ protože matice má rank m, tak na vygenerování celého sloupcového prostoru potřebuju m vektorů, čili báze B velikosti m existuje

Tvrzení: Každá báze B jednoznačně určuje nejvýše jedno bázické řešení.

→ pro $\forall B \subseteq \{1, 2, \dots n\}$, $|B| = m$, A_B regulární

$\exists$ nejvýše jedno přípustné $x \in \mathbb{R}^n$ t.ž. $j \notin B \Rightarrow x_j = 0$

Více bází ovšem může dávat stejné řešení

Dk: Uvažme soustavu rovnic z LP, bez podmínek $x \geq 0$

$$\underbrace{A}_{n} x = \underbrace{A_B}_{m} x_B + \underbrace{A_N}_{n-m} x_N = b \quad , N := [n] \setminus B$$

→ hledáme bázické x, tedy $x_N = \vec{0} \Rightarrow A_B x_B = b$

→ A_B je čtvercová regulární $\Rightarrow$ rovnice $A_B x_B = b$ má jediné řešení $\tilde{x}_B$.

- pokud jsou všechny složky $\tilde{x}_B$ nezáporné → doplníme nulami a máme bázické řešení
- pokud je tam něco záporného → nesplňuje $x \geq 0 \Rightarrow \nexists$ přípustné bázické ř. ∎

→ v takovém případě říkáme, že <u>báze B je přípustná</u>

Věta: Uvažme LP $\max c^T x$ s.t. $Ax = b$, $x \geq 0$.

1, pokud $\exists$ feasible solution & obj. f. is bounded $\Rightarrow \exists$ optimal sol.

2, pokud $\exists$ optimal solution $\Rightarrow \exists$ basic optimal solution.

Dk: Ekvivalentně dokážeme: Pokud je obj. f. bounded, pak pro každé feasible x_0 $\exists$ basic feasible $\tilde{x}$ s.t. $c^T \tilde{x} \geq c^T x_0$.

→ proč to funguje? basic solutions je jen konečně mnoho ⇒ jedno je největší

→ uvažme libovolné přípustné $x_0 \in \mathbb{R}^n$

$$M := \{x \in \mathbb{R}^n \mid c^T x \geq c^T x_0\}$$

$\tilde{x} \in M$ ... vybereme tak, aby mělo co nejvíc nul

→ ukážeme, že $\tilde{x}$ je basic

$$K := \{j \mid \tilde{x}_j > 0\}$$

a, sloupce A_K jsou lin. nez. ⇒ podle lemmatu je $\tilde{x}$ basic

b, sloupce A_K jsou lin. záv. ⇒ ukážeme sporem že nenastane

$$\Rightarrow \exists \underset{v \neq \vec{0}}{v \in \mathbb{R}^{|K|}} : A_K v = \vec{0} \quad \text{(definice závislosti)}$$

$$\Rightarrow \text{vyrobíme } w \in \mathbb{R}^n : w_i := \begin{cases} v_i, & i \in K \\ 0, & \text{jinak} \end{cases}$$

$$Aw = A_K v + \vec{0} = \vec{0}$$

⇒ definujeme $x(t) := \tilde{x} + t \cdot w$, $t \in \mathbb{R}$

$Ax(t) = A\tilde{x} + t \cdot Aw = A\tilde{x}$ ... pokud $x(t) \geq 0$, tak feasible

↳ $\forall t$ splňuje podmínky

→ ukážeme, že pro nějaké t_0 je $x(t_0)$ přípustné & $c^T x(t_0) \geq c^T x_0$ a navíc má $x(t_0)$ víc nul než $\tilde{x}$ → což bude spor *

BÚNO: ! $w \neq 0$

i) $c^T w \geq 0$ ... jinak vybereme $-w$

ii) $\exists j \in K$ s.t. $w_j < 0$ ... kdyby ne, tak $w \geq 0$

↳ pokud $c^T w = 0$, tak vyberu w / $-w$ aby ii) platilo

↳ jinak $c^T w > 0$ & $w \geq 0$ ⇒ $x(t) \geq 0$ pro $\forall t$

$$\Rightarrow c^T x(t) = c^T \tilde{x} + t \cdot c^T w \longrightarrow \infty \text{, LP is unbounded } \lightning$$

⇒ pokud existuje ⊖ složka nektom w, budeme zvětšovat
$t>0$ aby $x(t)=\hat{x}+t\cdot w$ mělo o jednu nulu víc než $\hat{x}$.

↳ navíc $c^T(\hat{x}+tw) = \underbrace{c^T\hat{x}}_{\geq c^Tx_0} + t\cdot \underbrace{c^Tw}_{\geq 0} \geq c^Tx_0$ → SPOR ↯ ∎

Geometrie LP

Def: Množina $M \subseteq \mathbb{R}^n$ je konvexní $\equiv \forall x,y \in M\ \forall t \in [0,1]:\ tx+(1-t)y \in M$.

Def: Convex polyhedron je průnik konečně mnoho poloprostorů.
Convex polytope je omezený polyhedron.

→ hyperplane: $a^Tx = b$
→ halfspace: $a^Tx \gtreqless b$ ⇒ polyhedron $= \{x \in \mathbb{R}^n \mid Ax \leq b\}$.

Def: Convex hull of $\{v_1, v_2, \dots v_k\}$ is $\{x \in \mathbb{R}^n \mid \overbrace{x = \sum_i \alpha_i v_i : \sum_i \alpha_i = 1,\ \alpha_i \geq 0}^{\text{konvexní kombinace}}\}$

👁 Convex hull M je průnik všech konvexních množin obsahujících M.

Dk: M je konvexní ⇒ obsahuje všechny konvexní kombinace svých bodů

Dk: indukcí podle # prvků M

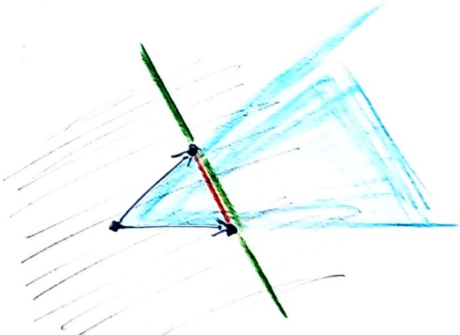
Combinations / hulls: → hull je vždy množina všech kombinací

linear $\text{lin} = \{x \in \mathbb{R}^n \mid x = \sum_i \alpha_i v_i,\ \alpha_i \in \mathbb{R}\}$

conic $\text{cone} = \{x \in \mathbb{R}^n \mid x = \sum_i \alpha_i v_i,\ \alpha_i \geq 0\}$

affine $\text{aff} = \{x \in \mathbb{R}^n \mid x = \sum_i \alpha_i v_i,\ \sum_i \alpha_i = 1\}$

convex $\text{conv} = \{x \mid x = \sum_i \alpha_i v_i,\ \sum_i \alpha_i = 1,\ \alpha_i \geq 0\}$



Věta (Minkowski-Weyl): P je polyhedron $\equiv \exists$ konečné množiny $V, Y \subseteq \mathbb{R}^n$ t.ž.

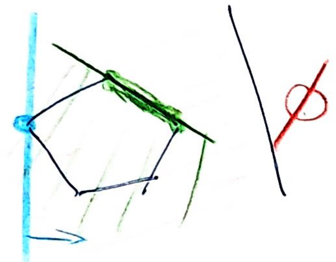
$P = \text{conv}(V) + \text{cone}(Y)$, $A+B := \{a+b \mid a \in A,\ b \in B\}$ ← Minkowski sum

Věta: P je polytop $\equiv \exists$ konečná V t.ž. $P = \text{conv}(V)$.

Def: Uvažme polyhedron P.

- nerovnost $a^T x \le b$ is valid for $P \equiv \forall p \in P: a^T p \le b$
- pokud je $a^T x \le b$ validní pro P, pak množinu

$$P \cap \{x \mid a^T x = b\}$$ nazýváme stěna (face) P



$0^T x \le 1 \to 0 = 1 \to \emptyset$

$0^T x \le 0 \to 0 = 0 \to P$

$\Rightarrow$ Trivial faces: $\emptyset$, P

- $\dim(P) := \dim(\mathrm{aff}(P))$ ... $\dim(\square \text{ v } \mathbb{R}^3) = 2$

  ↳ dim affinního prostoru je jasná

- speciální stěny:

  vertex := face of dim. 0

  hrana := face of dim 1

  faseta := face of dim $n-1$

- bod $v \in P$ je extreme point pro $P \equiv \exists$ obj. f. $c \in \mathbb{R}^n$ s.t.

$$\forall x \in P \setminus \{v\}: c^T x < c^T v$$

v is a vertex of a polytope $P \equiv v$ is an extreme point

Věta: Každý polytop má finitely many vertices.

→ implikuje intuitivní MAX řešení pro polytopy

Dk: $P = \{x \mid Ax \le b\}$, polytop $\qquad Ax \le b$: $\begin{matrix} a_1^T x \le b_1 \\ a_2^T x \le b_2 \\ \vdots \\ a_m^T x \le b_m \end{matrix}$

v is vertex $\Rightarrow \exists i: a_i^T v = b_i$

stěna polytopu je polytop ... k těm nerovnicím jen přidám tu definující rovnost stěny

dim stěny P < dim P

→ $f(m,d)$:= # vrcholů d-dim. polytopu popsaného m nerovnostmi

$$f(m,d) \le m \cdot f(m+1, d-1) \le m \cdot (m+1) f(m+2, d-2) \le m^d \leftarrow \text{finite}$$

↳ rozdělím na m polytopů kde do každého přidám jednu rovnost

LP ~ Polyhedrony

Věta: Basic feasible solutions ~ vertices

objective fcn. nás nezajímá

→ Nechť P je množina všech feasible sols. LP $Ax=b$, $x \geq 0$

tvrdíme $v \in P$ je vertex P $\iff$ v je basic sol.

Dk: $\Rightarrow$: v je vertex, tedy maximalizuje nějakou funkci, řekněme

$$v = P \cap \{x \mid a^T x = d\}$$

→ zvolme pro náš program novou účelovou funkci: $a^T x$

→ pak podle věty (∀∗∗) ∃ bázické optimum

↳ ale je to vrchol, čili optimum je jen jedno $\Rightarrow$ v je to b. s.

$\Leftarrow$: Nechť v je basic feasible s bází B.

→ tedy $j \notin B \Rightarrow v_j = 0$, A_B je čtvercová, regulární

→ definujeme úč. fci $\tilde{c}^T \in \mathbb{R}^n$ jako

$$c^T_j := \begin{cases} 0, & j \in B \\ -1, & j \notin B \end{cases} \Rightarrow \tilde{c}^T v = 0$$

∀ $x \geq 0$ je $c^T x \leq 0$ $\Rightarrow$ v maximalizuje obj. f. $\tilde{c}^T$

→ pokud je v jediné optimum, potom to je vertex

pokud má x nenulovou složku mimo B, pak $c^T x < 0$

→ může být víc vektorů, co mají mimo B samé nuly?

↳ NE ∵ báze B určuje vždy nejvýše jedno bázické řešení

SIMPLEXOVÁ METODA

$$\begin{aligned} \max \quad & x_1 + x_2 \\ \text{s.t.} \quad & -x_1 + x_2 \le 1 \\ & x_1 \le 3 \\ & x_2 \le 2 \\ \hline & x_1, x_2 \ge 0 \end{aligned}$$

1. převod do rovnicového tvaru
→ slack variables

$$\begin{aligned} \max \quad & x_1 + x_2 \\ \text{s.t.} \quad & -x_1 + x_2 + s_1 = 1 \\ & x_1 + s_2 = 3 \\ & x_2 + s_3 = 2 \\ \hline & x_1, x_2, s_1, s_2, s_3 \ge 0 \end{aligned}$$

$$A' = \begin{array}{c} \quad A \qquad\quad I_m \\ \left(\begin{array}{cc|ccc} -1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right) \end{array}$$

2. najdi initial basic feasible solution
→ pokud jsem přidal slack variables — I_m je regulární
⇒ do báze dám slack variables
→ jinak přidám nové auxiliary variables $a_1, \dots a_m$

$x_1 - x_2 = -1$ ⇝ $-x_1 + x_2 = 1$ → možná není hned jasná báze
$x_1 \le 3$ ⇝ $x_1 + s_1 = 3$

↳ $-x_1 + x_2 + a_1 = 1$
$x_1 + s_1 + a_2 = 3$

→ vyřeším tuto úlohu s obj. f. $\max -(a_1 + a_2 + \dots + a_m)$
↳ lehký start s $a_1, \dots a_m$ v bázi
→ pokud optimální hodnota není 0 ⇒ původní úloha nemá řeš.
→ jinak je optimum nějaké $(x_1, x_2, s_1, 0, 0)$
↳ tohle mi dává počáteční řešení pro nastartování původní simplexovky

! simplexovka zlepšuje nějaké aktuální řešení tím, že chodí po vrcholech toho polyhedronu

3. vdělám simplexovou tabulku

$$s_1 = 1 + x_1 - x_2$$
$$s_2 = 3 - x_1$$
$$s_3 = 2 \quad - x_2$$
$$z = \underbrace{x_1 + x_2}_{\text{obj. func.}}$$

→ odpovídá jí basic feasible sol $(0,0,1,3,2)$ s hodnotou 0

→ nalevo jsou bázické proměnné

↳ napravo jejich aktuální hodnota

↳ a jak souvisí s nebázickými (nuly)

4. vyberu proměnnou co má vstoupit do báze

↳ účel. fci zlepším buď přes x_1 nebo x_2 ⇒ vyberu x_1

⇒ pivotovací pravidla: kterou vybrat

- Danzigovo: největší koeficient ... $1 \cdot x_1$, $1 \cdot x_2$
- Largest increase: která nakonec nejvíce zlepší hodnotu z
- Steepest edge: nejvíce mě posune ve směru vektoru c
  ↳ reálně se používá
- Bland's rule: vždy vyber proměnnou s nejmenším indexem
  ↳ zabraňuje cyklům

5. vyber proměnnou co má odejít

$s_1 = 1 + x_1 - x_2$ ... jak ~~moc~~ můžu zvednout x_1 aby $s_1 \geq 0$?

↳ x_1 je tady unbounded

$s_2 = 3 - x_1$ ... $x_1 \leq 3$

$s_3 = 2 - x_2$ ... neomezené

→ vyberu nejvíce omezující podmínku → tím nic neporuším

$$\begin{aligned} s_1 &= 1 + x_1 - x_2 \\ s_2 &= 3 - x_1 \\ s_3 &= 2 - \quad x_2 \\ \hline z &= \quad x_1 + x_2 \end{aligned} \xrightarrow[s_2]{x_1} \begin{aligned} s_1 &= 4 - s_2 - x_2 \\ x_1 &= 3 - s_2 \\ s_3 &= 2 \quad - x_2 \\ \hline z &= 3 - s_2 + x_2 \end{aligned} \xrightarrow[s_3]{x_2} \begin{aligned} s_1 &= 2 - s_2 + s_3 \\ x_1 &= 3 - s_2 \\ x_2 &= 2 - s_3 \\ \hline z &= 5 - s_2 - s_3 \end{aligned}$$

⇒ optimum je $(3,2,2,0,0)$ s hodnotou 5

! nemůže být nějaké $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \tilde{s}_1, \tilde{s}_2, \tilde{s}_3)$ lepší? $\left.\begin{aligned} z &= 5 - \tilde{s}_2 - \tilde{s}_3 \\ \tilde{s}_2, \tilde{s}_3 &\geq 0 \end{aligned}\right\}$ NE

Co se může pokazit?

① Unboundedness

→ pokud je obj. f. neomezená v rámci našeho LP, tak nějakou proměnnou vstupující do báze bude možné zvyšovat neomezeně

② Degeneracy

$$\frac{\begin{array}{l} x_3 = \quad x_1 - x_2 \\ x_4 = 2 - x_1 \end{array}}{z = \quad x_2} \xrightarrow[x_3]{x_2} \frac{\begin{array}{l} x_2 = \quad x_1 - x_3 \\ x_4 = 2 - x_1 \end{array}}{z = \quad x_1 - x_3} \xrightarrow[x_4]{x_1} \frac{\begin{array}{l} x_2 = 2 - x_4 - x_3 \\ x_1 = 2 - x_4 \end{array}}{z = 2 - x_3 - x_4}$$

↓

degenerový krok: úč. fce se nezlepšila

→ dostal jsem se do jiné báze (vrcholu) ale zůstal se stejným solution

→ pokud se nezacyklím, tak to nevadí

③ Infeasibility

→ získání počátečního řešení není obecně snadné

→ pomocí simplexovky vyřeším auxilary LP, jehož optimum nějaké bázické řešení mého programu – viz vedlejší strana

👁 úloha nalezení feasible a optimal sol. jsou podobně těžké

↳ když umím najít feasible v čase $T(n)$, tak optimal v čase $\log(n)\,T(n)$ – půlení intervalu (nadrovinou)

④ Cycling

– může se stát, že se bude točit v degenerovaném kruhu

→ Bland's rule tomu zabraňuje

– v praxi tuto možnost ignoruju – je to rare

⑤ Složitost

– v praxi simplexovka pro m rovnic trvá $2m - 3m$ kroků

→ ale není známé žádné obecné polynomiální pivotovací pravidlo

↳ pro mnoho pravidel jsou známé exponenciální protipříklady

Simplexové tablo obecně

<u>Def</u>: $T(B)$ pro bázi B je systém $m+1$ lineárních rovnic v proměnných $x_1, \dots x_n, z$, které má stejnou množinu řešení jako $Ax = b$
$z = c^T x$

$$\frac{x_B = p + Q x_N}{z = z_0 + r^T x_N}$$

x_B = bázické proměnné
x_N = nebázické ... $N = [n] \setminus B$

↳ $p \in \mathbb{R}^m$, $r \in \mathbb{R}^{n-m}$, $Q \in \mathbb{R}^{m \times (n-m)}$, $z_0 \in \mathbb{R}$

<u>Věta</u>:
- pro každou bázi B existuje právě jedno tablo
- pokud $r \leq 0$, pak je odpovídající bázické řešení optimální
- nebázická proměnná může vstoupit do báze $\iff$ její koeficient v r je kladný
- leaving variable musí omezovat increase entering variable most strictly
- pokud entering x_E a leaving x_L toto splňuje, tak $B' := (B \setminus \{E\}) \cup \{L\}$ je opět feasible báze
- pokud žádná proměnná není kandidát pro leaving (neomezuje vstupní), tak je LP unbounded

<u>Věta</u>: jak získat feasible bases

máme $\max c^T x$ s.t. $Ax = b$, $x \geq 0$
→ nejprve přenásobíme rovnice aby $b \geq 0$
→ zavedeme nové proměnné $s_1, \dots s_m$ — $(x_1, \dots x_n, s_1, \dots s_m)$
⇒ auxiliary LP: $\max -\sum s_i$ s.t. $\bar{A}(x.s) = b$, $x.s \geq 0$
$\bar{A} = (A \mid I_m)$

↳ počáteční báze: $s_1 \dots s_m$

→ vyřeším — optimum není 0 → původní nemá řešení
↳ optimum je 0 ⇒ $(x_1, \dots x_n, 0, 0, \dots 0)$
↳ toto je bázické řešení původního — simplexovka vždy vrátí bázické řešení
→ pokud báze obsahuje nějaké z s_i, tak to musím vyjádřit bez nich

DUALITA

Primární LP

$$\max \quad 2x_1 + 3x_2$$

$$\begin{aligned} \text{s.t.} \quad \text{I.} \quad & 4x_1 + 8x_2 \le 12 \\ \text{II.} \quad & 2x_1 + x_2 \le 3 \\ \text{III.} \quad & 3x_1 + 2x_2 \le 4 \\ & x_1, x_2 \ge 0 \end{aligned}$$

→ chceme odhadnout optimum

$\text{I.}: \; 2x_1 + 3x_2 \le 4x_1 + 8x_2 \le 12$

$\le 2x_1 + 4x_2 \le 6$

$\text{I} \,\&\, \text{II}: \; 2x_1 + 3x_2 = \frac{1}{3}(\text{I} + \text{II}) \le \frac{1}{3} 15 = 5$

→ chceme najít nerovnosti ve tvaru

$d_1 x_1 + d_2 x_2 \le h$, kde $d_1 \ge 2$, $d_2 \ge 3$, h je co nejmenší

conic combination

$\Rightarrow$ pak $2x_1 + 3x_2 \le d_1 x_1 + d_2 x_2 \le h$

$\Rightarrow$ zkombinujeme všechny tři nerovnice – každé přidáme nějaký koef.

$$y_1 \text{I} + y_2 \text{II} + y_3 \text{III} \le y_1 12 + y_2 3 + y_3 4$$

$$\hookrightarrow (4y_1 + 2y_2 + 3y_3) x_1 + (8y_1 + y_2 + 2y_3) x_2 \le 12 y_1 + 3y_2 + 4y_3$$

$\Rightarrow$ Duální LP

$$\min \quad 12y_1 + 3y_2 + 4y_3$$

$$\begin{aligned} \text{s.t.} \quad & 4y_1 + 2y_2 + 3y_3 \ge 2 \\ & 8y_1 + y_2 + 2y_3 \ge 3 \\ & y_1, y_2, y_3 \ge 0 \end{aligned}$$

- $A_B x_B + A_N x_N = b$
  $\Rightarrow x_B = A_B^{-1} b - A_B^{-1} A_N x_N$
- $z = c^T x = c_B^T x_B + c_N^T x_N$
  $= \underbrace{c_B^T A_B^{-1} b}_{z_0} + \underbrace{(c_N^T - c_B^T A_B^{-1} A_N)}_{(*)\; r^T} x_N$

Def: Primár (P) $\rightsquigarrow$ Duál (D)

Primár (P)		Duál (D)
$\max \; c^T x$		$\min \; b^T y$
$\text{s.t.} \; Ax \le b$	$\rightsquigarrow$	$\text{s.t.} \; A^T y \ge c$
$x \ge 0$		$y \ge 0$
m podmínek		m proměnných
n proměnných		n podmínek

Věta (slabá o dualitě) $\forall$ feasible sol. of D is an upper bound of P.

$\hookrightarrow$ x feasible P, y feasible D $\Rightarrow c^T x \le b^T y$

Věta (silná o dualitě): Vždy nastane právě jedna situace:

1. P and D are both infeasible
2. P is unbounded & D is infeasible (spíš $\Rightarrow$)
3. P is infeasible & D is unbounded
4. P and D are both feasible & bounded

a mají optima x^* (P), y^* (D) a platí $c^T x^* = b^T y^*$.

Pozn.: 3 možnosti: infeasible (1), feasible bounded (2), unbounded (3) $\to$ 9 pro P a D

↳ weak věta vylučuje: P3 & D2, P3 & D3, P2 & D3

↳ v důkazu vyloučíme P1 & D2, P2 & D1

Dk: Dokážeme, že pokud je P feasible & bounded, pak je duál feasible (a podle slabé duality taky bounded) a má stejné optimum jako P.

↳ tohle vyloučí P2 & D1, protože duál duálu je primár, tak i P1 & D2

$\to$ uvažme LP $\max c^T x$ s.t. $Ax \leq b$, $x \geq 0$. (P)

$\to$ simplexovka: přidáme slack variables $x_{n+1}, \dots x_{n+m}$

$$\Rightarrow \max \bar{c}^T \bar{x} \text{ s.t. } \bar{A}\bar{x} \leq b, \quad \bar{x} \geq 0$$

↳ $\bar{x} = (x_1, \dots x_{n+m})$, $\bar{c}^T = (c_1, \dots c_n, 0, \dots 0)$, $\bar{A} = (A \mid I_m)$

$\to$ pokud je feasible & bounded, tak simplexovka s Bland's pivot rule najde optimální řešení $\bar{x}^*$ s bází B., kde $x^* := (\bar{x}_1^*, \dots \bar{x}_n^*)$ je optimum původního programu (P)

Lemma: Vektor $y^* = (\bar{c}_B^T \bar{A}_B^{-1})^T$ je přípustné řešení D a $c^T x^* = b^T y^*$.

Důsledek: Ze slabé duality je y^* optimum D, což dokáže větu.

Dk: uvažme tablo, $\because \bar{x}^*$ je optimální, tak $r \leq 0$

$$\bar{x}_B^* = p + Q\bar{x}_N^*$$
$$z = z_0 + r^T \bar{x}_N^*$$

$\to \bar{A}\bar{x}^* = \bar{A}_B \bar{x}_B^* + \bar{A}_N \underbrace{\bar{x}_N^*}_{=\vec{0}} = b \Rightarrow \bar{x}_B^* = \bar{A}_B^{-1} b \; (+\vec{0})$

$$\Rightarrow c^T x^* = \bar{c}^T \bar{x}^* = \bar{c}_B^T \bar{x}_B^* = \bar{c}_B^T (\bar{A}_B^{-1} b) = (\bar{c}_B^T \bar{A}_B^{-1}) b = (y^*)^T b = b^T y^*$$

$\to$ zbývá ukázat, že y^* je feasible, tedy $y^* \geq 0$ & $A^T y^* \geq c$ (D)

↳ tedy $\bar{A}^T y^* \geq \bar{c}$ (*) ... $\because y^* \geq 0 \iff I_m y^* \geq 0$

(+) ↳ $\bar{A}^T y^* = \bar{A}^T (\bar{c}_B^T \bar{A}_B^{-1})^T = (\bar{c}_B^T \bar{A}_B^{-1} \bar{A}^T)^T =: w \in \mathbb{R}^{n+m}$

bázické • $w_B = (\bar{c}_B^T \bar{A}_B^{-1} \bar{A}_B)^T = \bar{c}_B$ ... tedy platí rovnost (*) $\Rightarrow$ je to feasible

nebázické • $w_N = (\bar{c}_B^T \bar{A}_B^{-1} \bar{A}_N^T)^T \overset{(+)}{=} \bar{c}_N - r \geq \bar{c}_N \because r \leq 0$ ... platí nerovnost (*) ■

Komplementarita

Věta: Mějme úlohy

$P:\quad \max c^T x,\quad Ax \le b,\quad x \ge 0$

$D:\quad \min b^T y,\quad A^T y \ge c,\quad y \ge 0$

Přípustná řešení x^*, y^* jsou optimální $\Leftrightarrow$

1) $\forall i \in [n]: \; x_i = 0 \;\vee\; \sum_{j=1}^{m} A_{ji} \cdot y_j = c_i$

2) $\forall j \in [m]: \; y_j = 0 \;\vee\; \sum_{i=1}^{n} A_{ji} \cdot x_i = b_j$

Příklad:

$$(P)\quad \begin{aligned} \max\ & 2x_1 + 4x_2 + 3x_3 + x_4 \\ \text{s.t.}\ & 3x_1 + x_2 + x_3 + 4x_4 \le 12 \\ & x_1 - 3x_2 + 2x_3 + 3x_4 \le 7 \\ & 2x_1 + x_2 + 3x_3 - x_4 \le 10 \end{aligned} \qquad x_i \ge 0$$

$$(D)\quad \begin{aligned} \min\ & 12y_1 + 7y_2 + 10y_3 \\ \text{s.t.}\ & 3y_1 + y_2 + 2y_3 \ge 2 \\ & y_1 - 3y_2 + y_3 \ge 4 \\ & y_1 + 2y_2 + 3y_3 \ge 3 \\ & 4y_1 + 3y_2 - y_3 \ge 1 \end{aligned} \qquad y_j \ge 0$$

$\to x^* = (0, \frac{52}{5}, 0, \frac{2}{5})$ je optimum v P. $\to$ urči optimum v D

$\hookrightarrow x_2 \ne 0 \Rightarrow y_1 - 3y_2 + y_3 = 4$

$\hookrightarrow x_4 \ne 0 \Rightarrow 4y_1 + 3y_2 - y_3 = 1$ ... využil jsem (1)

(2): splňuje x^* nějaké nerovnosti v P neostře?

$\to$ 2. nerovnost v P není splněna těsně $\Rightarrow y_2 = 0$

$\Rightarrow$ celkem $\begin{aligned} y_1 + y_3 &= 4 \\ y_1 - y_3 &= 1 \end{aligned} \Rightarrow y^* = (1, 0, 3)$

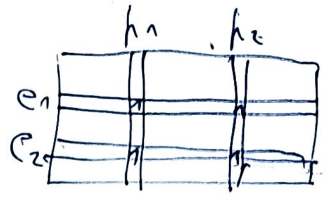
Příklad

$\to$ mám úlohu LP a nějaké přípustné řešení duálu

$\hookrightarrow$ pomocí těchto podmínek můžu zkontrolovat, jestli je optimální

Využití duality

Věta: Max flow = Min cut



$$\Pi := \{p \mid p \text{ je cesta zdroj} \to \text{stok}\}$$

P $\quad \max \sum_{p\in\Pi} f_p \quad$ s.t. $\quad \forall e\in E: \sum_{p\ni e} f_p \le c_e \quad , f_p \ge 0$

D $\quad \min \sum_{e\in E} c_e y_e \quad$ s.t. $\quad \forall p\in\Pi: \sum_{e\in p} y_e \ge \vec{1} \quad \dots \; c^T = (1, \dots, 1)$

👁 P je feasible a bounded ∵ kapacity jsou bounded

👁 D —— " —— ∵ —— " ——

⇒ podle silné věty o dualitě opt P = max flow = opt D

→ stačí ukázat, že D řeší min cut problem

Lemma: C_{min} = opt D

Dk: Pro $S \subseteq V$ definujeme edge-cut jako $E(S,\bar S) := \{uv \in E \mid u\in S, v\in \bar S\}$

$$C(S,\bar S) := \sum_{e\in E(S,\bar S)} c_e \quad \dots \text{ kapacita řezu}$$

i) opt D ≤ C_{min}: Pro každý řez $(S,\bar S)$ existuje nějaké feasible $y \in D$ se stejnou hodnotou, tedy $C(S,\bar S) = \sum_e y_e c_e$.

→ pro řez $(S,\bar S)$ definujeme $y_e := \begin{cases} 1, & e\in E(S,\bar S) \\ 0, & \text{jinak} \end{cases}$

↳ je y feasible? Ano ∵ $(S,\bar S)$ je řez ⇒ ∀ cesta obsahuje hranu z něj.

ii) C_{min} ≤ opt D: Pro ∀ feasible y existuje řez $(S,\bar S)$ t.ž.

$$c(S,\bar S) \le \sum_e y_e c_e$$

Dk: Do grafu navíc každé hraně $e\in E$ přidáme váhu $w_e := y_e$

↳ definujeme $d: V \to \mathbb{R}$ jako vzdálenost vůči těmto vahám (od zdroje)

→ pro $\varrho \in [0,1)$: $S_\varrho := \{v\in V \mid d(v) \le \varrho\}$

zdroj 👁 $d(s) = 0 \Rightarrow s \in S_\varrho$

stok $\quad d(T) \ge 1 \Rightarrow T \notin S_\varrho$ $\quad \Rightarrow (S_\varrho, \bar S_\varrho)$ je řez

↳ z podmínek D

→ nyní uniformně náhodně vybereme $\rho \in (0,1)$

$$\mathbb{E}[c(S_\rho, \overline{S_\rho})] = \sum_{e \in E} c_e \cdot \underbrace{P[e \in c(S_\rho, \overline{S_\rho})]}_{p}$$

→ jaká je pst p, že $e = uv \in$ řez?

$$\underset{0}{S} \vdash \underset{d(u)}{|} \underbrace{\quad}\underset{d(v)}{|} \cdots \underset{d(T)}{|} T$$

$$d(v) - d(u) \leq w_e = y_e$$

→ pokud $\rho < d(u)$ nebo nebo $\rho > d(v)$, tak $e \notin$ řez

$\Rightarrow$ chci aby $\rho \in [d(u), d(v)] \Rightarrow p = P[\rho \in [d(u), d(v)]] \leq y_e$

$$\Rightarrow \mathbb{E}[c(S_\rho, \overline{S_\rho})] \leq \sum_{e \in E} c_e y_e$$

→ $\mathbb{E}$ řezů je $\leq$ než ↑, tedy $\exists$ řez $(S, \overline{S})$ co je $\leq \sum_e c_e y_e$. ∎

Tvrzení: Úloha nalezení přípustného řešení je stejně těžká jako optimalizace.

Dk: P
$$\max \ c^T x \quad \text{s.t.} \ Ax \leq b, \ x \geq 0$$

→ uděláme

$$P' \quad \max \ c^T x \quad \text{s.t.} \ Ax \leq b, \ A^T y \geq c, \ c^T x \geq b^T y, \ x \geq 0, y \geq 0$$

$A^T y \geq c$, $c^T x \geq b^T y$ } vynucuji slabou větu o dualitě

👁 P má optimum $\iff$ P' má feasible solution

∎

TOTÁLNÍ UNIMODULARITA

Def: Čtvercová matice $A \in \mathbb{Z}^{m \times m}$ je unimodulární $\equiv \det(A) \in \{-1, 1\}$.

Def: Matice $A \in \mathbb{R}^{m \times n}$ je totálně unimodulární $\equiv$

$\forall$ čtvercová podmatice má $\det \in \{-1, 0, +1\}$.

TU matice obsahují pouze $0, 1, -1$.

Tvrzení: Pokud je A TUM, pak má polyhedron $\{x \in \mathbb{R}^n,\ Ax \leq b,\ x \geq 0\}$ celočíselné vrcholy pro každé $b \in \mathbb{Z}^m$.

Dk: Každému vrcholu odpovídá nějaké bázické řešení $x \in \mathbb{R}^n$

$x_N = 0$ ... to $\in \mathbb{Z}$

$x_B = ?$ ... $Ax = b \Rightarrow A_B x_B + A_N \overset{0}{x_N} = b \Rightarrow x_B = A_B^{-1} b$

$\rightarrow$ stačí ukázat, že $A_B^{-1} \in \mathbb{Z}^{m \times m}$

$\hookrightarrow$ víme, že A_B obsahuje jen $0, 1, -1$, tedy speciálně $\in \mathbb{Z}^{m \times m}$.

$\rightarrow$ z LA víme, že $\operatorname{adj}(A_B) = \det(A) \cdot A_B^{-1}$ $\quad \boxed{(\operatorname{adj}(A))_{ji} := (-1)^{i+j} \det(A^{ij})}$

$\hookrightarrow$ adjungovaná matice je definovaná pomocí determinantů $A^{ij} \in \{0, 1, -1\}$

$\Rightarrow$ protože $\operatorname{adj}(A_B) \in \mathbb{Z}^{m \times m}$ & $\det(A) \in \{0, 1, -1\}$, tak $A_B^{-1} \in \mathbb{Z}^{m \times m}$

Důsledek: Pokud matice nějakého problému je TUM a vyřešíme relaxaci, tak máme rovnou celočíselné optimum.

Lemma: $A \in \{0, 1, -1\}^{m \times n}$, kde $\forall$ sloupec obsahuje nanejvýš 2 nenulové prvky.

Nechť lze řádky A rozdělit do dvou množin R_1, R_2 tak, že

jsou-li ve stejném sloupci 2 nenulové prvky

i) stejného znaménka $\Rightarrow$ patří do různých množin R_1, R_2

ii) různých znamének $\Rightarrow$ oba leží ve stejné množině

pak A je TUM

Dk: Laplaceovo pravidlo

$\rightarrow$ sloupec j: $\det(A) = \sum_{i=1}^{m} (-1)^{i+j} a_{ij} \det(A^{ij})$

Indukce podle $K :=$ velikost $\square$ podmatice $\rightarrow K = 1$ triviálně platí

$K > 1$: chci $\det(\square) \in \{-1, 0, 1\}$

1) $\exists$ sloupec s 0 $\Rightarrow \det = 0$

2) $\exists$ sloupec s jediným výskytem -1 nebo 1 $\Rightarrow$ pomocí Laplace převedu na $K-1$

3) $\forall$ sloupec má 2 nenuly $\Rightarrow$ det bude 0

$$\begin{bmatrix} & -1 \\ -1 & 1 \\ & \\ & -1 \end{bmatrix}$$

sečtu ř. R_1, sečtu ř. R_2 } odečtu od sebe $\Rightarrow$ nulový ř.

$\Rightarrow \det = 0$

Věta:

a) Matice incidence orientovaného grafu $G=(V,E)$ je $A_G \in \{-1,0,1\}^{|V|\times|E|}$

$$(A_G)_{v,e} := \begin{cases} -1, & e=(v,u) \quad \dots \text{ hrana začíná na } v \\ 1, & e=(u,v) \quad \dots \text{ končí na } v \\ 0, & \text{jinak} \end{cases}$$

b) Matice incidence neorientovaného grafu G je $A_G \in \{0,1\}^{|V|\times|E|}$

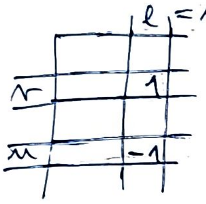
$$(A_G)_{v,e} := \begin{cases} 1, & v \in e \\ 0, & v \notin e \end{cases}$$

Tvrdíme, že

a) pro orientovaný graf je A_G vždy TUM

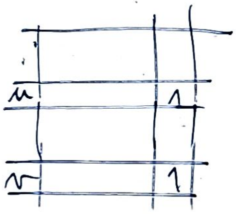
b) pro neorientovaný graf je A_G TUM $\iff$ G je bipartitní

Dk:

a) 

→ prostě vezmu $R_1 = V$ → vždy ve sloupci $1, -1$

$R_2 = \emptyset$

b) 

partity A, B $\iff$ $R_1 = A$, $R_2 = B$ ∎

Důsledek: Pro celočíselné kapacity je max-flow celočíselný.

• Min vertex cover = Max matching

Věta (Kőnigovo lemma): V bipartitním grafu platí $|C_{min}| = |M_{max}|$.

Dk: $\max \sum_{e\in E} x_e$ $\xrightarrow{\text{dual}}$ $\min \sum_{v \in V} y_v$

$\forall v \in V: \sum_{e \ni v} x_e \le 1$ $\qquad$ $\forall e \in E: y_u + y_v \ge 1$, $e = uv$

nepotřebuju ≤ 1 → dělá to tohle $\quad x_e \ge 0$ $\qquad$ $y_v \ge 0$

relaxace max matching $\qquad$ relaxace min cover

→ podle předchozí věty je A TUM

↳ simplexová přidá slack variables a bude řešit s $\bar{A} = (A \mid I_m)$

↳ ale pokud je A TUM, tak $(A \mid I_m)$ je taky TUM

⟹ optimum P je celočíselné

→ A je TUM ⟹ A^T je TUM ⟹ optimum D je celočíselné

$\Big\}$ $C_{min} \overset{\text{silná dualita}}{=} M_{max}$ ∎

ELLIPSOID ALGORITHM

→ algoritmus lineárního programování
  ↳ v praxi pomalejší než simplex
  ↳ ale teoreticky významný ∵ je polynomiální (je to dokázané)

Def: bit-size of

$i \in \mathbb{Z}: \quad \langle i \rangle := \lceil \log_2(|i|+1) \rceil + 1$

$r = \frac{p}{q} \in \mathbb{Q}: \langle r \rangle := \langle p \rangle + \langle q \rangle$

$\vec{v} \in \mathbb{Q}^n: \langle \vec{v} \rangle := \sum_{i=1}^{n} \langle v_i \rangle$

$(A) \in \mathbb{Q}^{m \times n}: \langle A \rangle := \sum_i \sum_j \langle a_{ij} \rangle$

$L: \max \; c^T x \quad \text{s.t.} \quad Ax \leq b$

↳ $\langle L \rangle := \langle A \rangle + \langle b \rangle + \langle c \rangle$

Def: Algoritmus je polynomiální pro lin. prog $\equiv \exists$ polynom $p(x)$ t.ž. pro každý racionální LP L najde řešení v $p(\langle L \rangle)$ krocích.

Def(elipsoidy): Ve 2D je to elipsa plus vnitřek. Obecně

$B_n := \{x \in \mathbb{R}^n \mid x^T x \leq 1\}$ … n-dim jednotková koule

→ n-dimensional ellipsoid je affinní transformace této koule

$E := \{Mx + s \mid x \in B_n\}$

- $M \in \mathbb{R}^{n \times n}$ je regulární
- $s \in \mathbb{R}^n$ určuje střed
- $x \mapsto Mx + s$ … lineární zobrazení + translace = affinní zobrazení

👁 $E = \{y \in \mathbb{R}^n \mid M^{-1}(y-s) \in B_n\}$
$= \{y \in \mathbb{R}^n \mid (y-s)^T (M^{-1})^T M^{-1} (y-s) \leq 1\}$
$= \{y \in \mathbb{R}^n \mid (y-s)^T Q^{-1} (y-s) \leq 1\}, \quad Q = MM^T$

👁 M je reg $\Rightarrow Q = MM^T$ je pozitivně def., tedy $x^T Q x > 0 \iff x \neq 0$

Ekvivalentně: E je elipsoid $\equiv E = \{y \in \mathbb{R}^n \mid (y-s)^T Q^{-1} (y-s) \leq 1\}$

pro nějakou pozitivně definitní Q.

→ s je střed toho elipsoidu

→ pokud je Q diagonální & $s = 0$, tak

axial position

$$E = \left\{ y \in \mathbb{R}^n \;\middle|\; \frac{y_1^2}{q_{11}} + \frac{y_2^2}{q_{22}} + \dots + \frac{y_n^2}{q_{nn}} \leq 1 \right\}$$

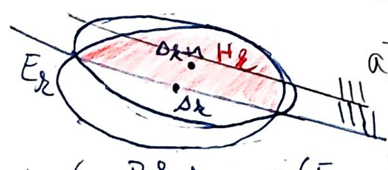
Fakt: Protože je Q p.d., tak ji lze diagonalizovat pomocí ortogonální matice T tak: TQT^{-1}, kde na diagonále jsou vlastní čísla Q

→ geometricky to reprezentuje rotaci systému souřadnic tak, aby se elipsoid dostal do axial position

Elipsoid method

- nehledá optimum, ale najde nějaké přípustné řešení - (což nevadí)
- $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$, předpokládáme, že celý polyhedron je
  1) uzavřený v nějaké kouli $B(\vec{0}, R)$, $R \in \mathbb{Q}$
  2) dost velký na to, aby se do něj vešla koule o poloměru $\varepsilon \in \mathbb{Q}$.
- ellipsoidová metoda vyrábí posloupnost elipsoidů $E_0, E_1, \dots E_t$, kde $P \subseteq E_k$ pro každé k a objemy elipsoidů se zmenšují

1. $E_0 := B(0, R)$
2. aktuální elipsoid $E_k = \{y \in \mathbb{R}^n \mid (y - s_k)^T Q_k^{-1} (y - s_k) \leq 1\}$
   ↳ pokud $A s_k \leq b$, tak je s_k feasible → konec
3. jinak vyber nerovnost, co není splněná
   ↳ nechť to je i-tá nerovnost, tedy $a_i^T s_k > b_i$
   → $P \subseteq E_k$, ale střed $s_k \notin P$ ⇒ chceme P uzavřít nějak líp
   ↳ $E_{k+1} :=$ nejmenší elipsoid obsahující half-elipsoid



$a_i^T x \leq b_i$ $\quad H_k := E_k \cap \{x \in \mathbb{R}^n \mid a_i^T x \leq a_i^T s_k\}$

$a_i^T x \leq a_i^T s_k$ → zjevně $P \in$

4. Pokud $vol(E_{k+1}) < vol(B(0, \varepsilon))$ → NO solution; ELSE: goto 2

Složitost:

bez důkazu: $\dfrac{\text{volume}(E_{k+1})}{\text{volume}(E_k)} \leq e^{\frac{-1}{2n+2}}$

$\Rightarrow \text{volume}(E_k) \leq e^{\frac{-k}{2n+2}} \cdot \text{vol}(B(0,R))$

→ protože objem B_n je $O(R^n)$, tak pokud nějaké k splňuje

$R \cdot e^{\frac{-k}{2n+2}} < \varepsilon$, tak $\text{vol}(E_k) < \text{vol}(B(0,\varepsilon))$

→ takové k dává horní odhad složitosti

$k < n(2n+2)\ln(R/\varepsilon)$

Edmons Algorithm

Def: Mějme bipartitní graf $G=(V,E)$. Definujeme

- matching polytop $P_M(G) := \left\{ x \;\middle|\; \begin{array}{l} \forall v \in V: \sum_{e \ni v} x_e \le 1 \\ \forall e \in E: x_e \ge 0 \end{array} \right\}$

  ↳ nepotřebujeme $1 \ge x_e$, je to implicitně v té Σ

- perfect matching polytop: $P_{PM}(G) := \left\{ x \;\middle|\; \begin{array}{l} \forall v \in V: \sum_{e \ni v} x_e = 1 \\ \forall e \in E: x_e \ge 0 \end{array} \right\}$

Připomenutí: tyto relaxace řeší dané problémy přesně – optimum $\in \mathbb{Z}^n$

dk 1: pomocí sudých cyklů
dk 2: pomocí TUM

→ co pro obecné grafy? Tam to selže na lichých cyklech

↳ pro $S \subseteq V$, $|S| =$ lichá určitě platí

$$0 \le \sum_{e \in E[S]} x_e \le \left\lfloor \frac{|S|}{2} \right\rfloor, \quad E[S] := \text{hrany mezi vrcholy v } S$$

↳ potože  napárujeme na 5 vrcholů max $\left\lfloor \frac{5}{2} \right\rfloor = \frac{5-1}{2}$ hran

Def: Pro obecný graf G definujeme

$$P_M(G) := \left\{ x \;\middle|\; \begin{array}{l} \forall v \in V: \sum_{e \ni v} x_e \le 1 \\ \forall e \in E: x_e \ge 0 \end{array}, \begin{array}{l} \forall U \subseteq V \\ |U| = \text{lichá} \end{array} : \sum_{e \in E[U]} x_e \le \frac{|U|-1}{2} \right\}$$

$$P_{PM}(G) := \left\{ x \;\middle|\; \begin{array}{l} \forall v \in V: \sum_{e \ni v} x_e \le 1 \\ \forall e \in E: x_e \ge 0 \end{array}, \begin{array}{l} \forall U \subseteq V \\ |U| = \text{lichá} \end{array} : \sum_{e \in \delta(U)} x_e \ge 1 \right\}$$

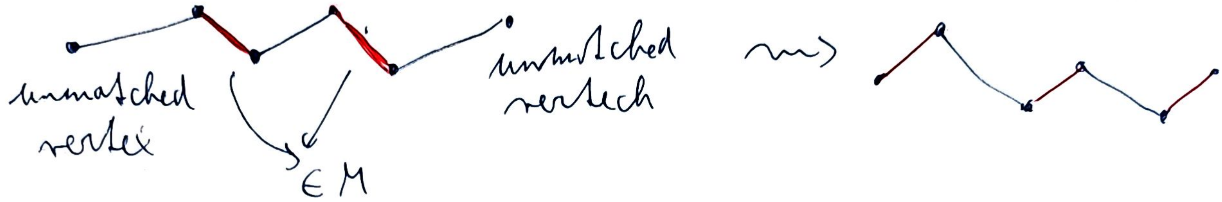
PM ⇒ alespoň 1 vrchol musí být ven

$\delta(S) :=$ hrany mezi U a $V \setminus U$

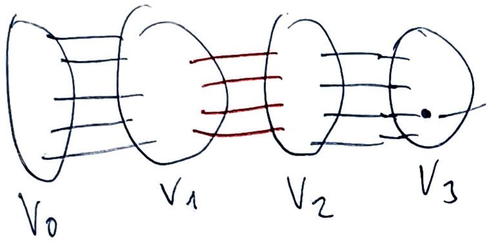
Fakt: Tyto nerovnosti stačí. Optimální řešení těchto relaxací je opět celočíselné a je to optimum celočíselného programu

Problém: je exponenciálně mnoho podmínek.

Lemma (Berge): Párování M v grafu G je maximální $\iff$ $\nexists$ M-augmenting path, což je



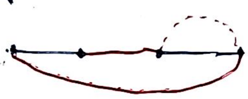
→ jak najít augmenting path?

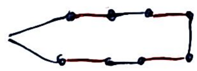


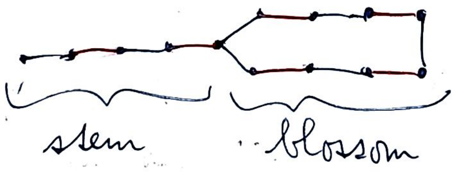
→ když najdu unmatched vertex, tak jsem vyhrál

V_0 = unmatched vertices

→ co když v tom grafu jsou cykly?

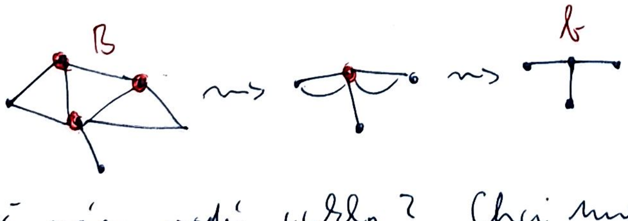
↳ sudé cykly:  — pokud jen sudé, tak je graf bipartitní, takže to je dobré

↳ liché cykly:  — pokud i liché, tak to je horší

M-flower: 

→ shrinking a blossom: představím si, že to je jen 1 vrchol

Def: Pro graf $G=(V,E)$ a $B\subseteq V$ definujeme graf G/B jako

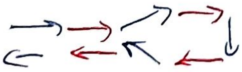


$V' := V \setminus B \cup \{b\}$

$E' := E \setminus \binom{B}{2} \,\dot\cup\, \{ub \mid \exists\, uv \in E: u \notin B,\ v \in B\}$

→ proč nám vadí cykly? Chci umět říct, že v tom grafu $\nexists$ augmenting path nebo ji vždy najít

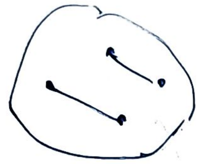
sudé:  ⇒ cyklí se to dopředu

liché:  vrací se dozadu ⇒ nikam nepostupuju, tváří se jako augmenting path tohle

→ pomocí shrinking umíme odhalit augmenting paths co používají blossoms

Věta: Nechť G je graf, M matching a B blossom. Pak $\exists$ M-augmenting path v G $\iff$ $\exists$ M/B-augmenting path v G/B.

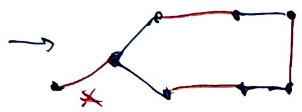
Pozn:



je M/B opět matching?

co by se mohlo pokazit?

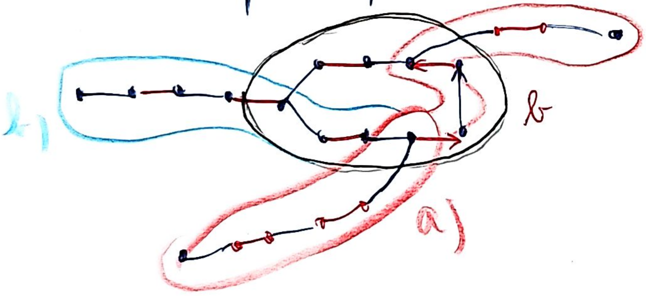
není matching



nevychází se to ∵ B je jen červeně napojený jen tou hranou * a ta je jen jedna — nevrcholové

Dk: $\Rightarrow$: Nechť $\exists$ M-aug. path p in G.

- pokud p se nedotýká B, tak p je M/B-aug. path in G/B.
- pokud p používá B



a) pokud není stonek

⇒ pak b je unmatched vertex v G/B

→ „prefix" je augmenting path

b) pokud je stonek

→ tak ho použiju – končí na matched vertex

$\Leftarrow$: Nechť $\exists$ M/B aug. path in G/B

- p does not touch b ✓
- p uses b

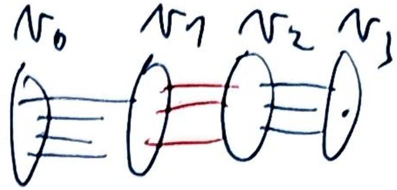
→ stejný argument ale opačně, někde do toho blossom vstupím, někde vylezu ∎

Důsledek: Když shrinknu graf a

- najdu aug. path ⇒ dokážu vyrobit a.p. v tom původním
- nenajdu a.p. ⇒ v původním taky nebyla

Edmond's alg:

→ hledá max-matching v G augmentováním M, $M_0 = \emptyset$



Augment(G, M)

1. Pokus se zjist M-aug. path
2. if fail with blossom B:

$$G' := G/B \qquad M' := M/B \quad \to \text{augment}(G', M')$$

3. if found augmentation → augment using it
4. else failed completely ⇒ max matching

CUTTING PLANES METHOD

→ způsob jak řešit celočíselné programy pomocí nomálních programů

Algoritmus:

Vstup: celočíselný program IP

0. nejprve nechť $LP :=$ relaxace IP

1. Nechť x^* je optimum LP

2. Pokud je x^* celočíselné → vyhrál jsem: je to optimum IP

3. Jinak najdi nerovnost, kterou splňují všechny feasible řešení IP, ale x^* ji porušuje, a přidej ji do LP → goto 1

→ LP je moc laxní, moc optimistické, přestřeluje to

⇒ každá nová nerovnost ho trochu srazí

⇒ postupně dostávám lepší a lepší odhady

→ hard part je step 3: najít tu nerovnost

Věta (Chvátal-Gomory cutting planes): Uvažme IP

$$\max c^T x$$
$$\text{s.t. } Ax \leq b \quad \dots \quad \forall i: a_i^T x \leq b_i \quad \dots \quad \sum_j a_{ij} x \leq b_i$$
$$x \geq 0, \; x \in \mathbb{Z}^n$$

→ zvolme $u \in \mathbb{R}^{+m}$ libovolné. Máme

$$u^T A x \leq u^T b \quad \text{neboli} \quad \sum_i u_i \sum_j a_{ij} x_j \leq u^T b \; \dots$$

→ uděláme trik $\gamma - \gamma = 0$:

$$\sum_j \left\lfloor \sum_i u_i a_{ij} \right\rfloor x_j + \sum_j \underbrace{\left(\sum_i u_i a_{ij} - \left\lfloor \sum_i u_i a_{ij} \right\rfloor \right)}_{\geq 0} \underbrace{x_j}_{\geq 0} \leq u^T b$$

$$\Rightarrow \sum_j \left\lfloor \sum_i u_i a_{ij} \right\rfloor x_j \leq u^T b$$

→ protože každé x co řeší IP je celočíselné, tak

$$\sum_j \left\lfloor \sum_i u_i a_{ij} \right\rfloor x_j \leq \left\lfloor u^T b \right\rfloor \quad \rightarrow \text{ tohle je Ch-G cutting p.}$$

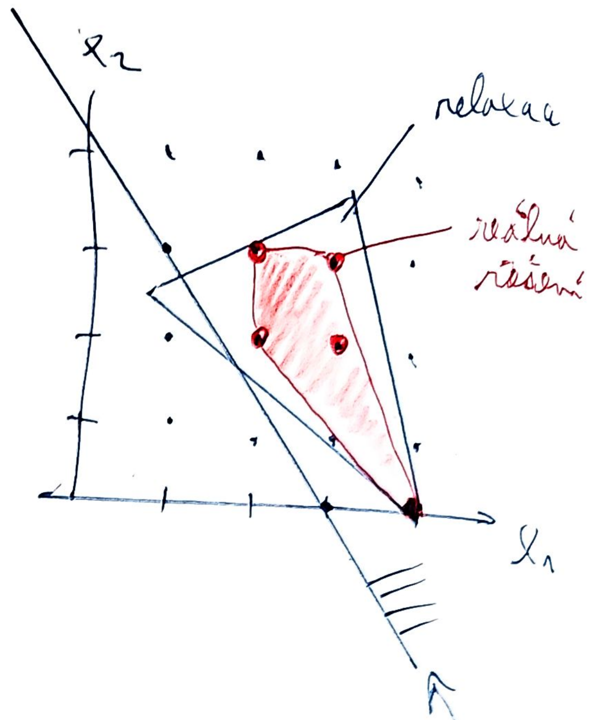
Příklad:

$$\max \quad 2x_1 + x_2$$

$$\text{s. t.} \quad 7x_1 + x_2 \le 28$$

$$-x_1 + 3x_2 \le 7 \qquad x_1, x_2 \ge 0$$

$$-8x_1 - 9x_2 \le -32 \qquad x_1, x_2 \in \mathbb{Z}$$



$\to$ volba $u = \left(0, \frac{1}{3}, \frac{1}{3}\right)$

$\hookrightarrow \sum_j \left\lfloor \sum_i u_i a_{ij} \right\rfloor x_j \le u^T b$

$$\left.\begin{aligned} x_1 &: \left\lfloor 0 \cdot 7 - \tfrac{1}{3} - \tfrac{1}{3} 8 \right\rfloor = \left\lfloor -\tfrac{9}{3} \right\rfloor = -3 \\ x_2 &: \left\lfloor 0 \cdot 1 + 1 - 3 \right\rfloor = \left\lfloor -2 \right\rfloor = -2 \\ u^T b &= \left\lfloor \tfrac{7}{3} - \tfrac{32}{3} \right\rfloor = \left\lfloor \tfrac{-25}{3} \right\rfloor = -9 \end{aligned}\right\} \quad -3x_1 - 2x_2 \le -9$$

Edmondsovy nerovnosti pro matching polytopy jsou instance Chvátal–Gomory cutting planes

Matroids

– popisují věci na které fungují greedy algoritmy

Def: Independence system is (E, I), kde $I \subseteq \mathcal{P}(E)$ splňující

elements ↗ ↖ independent set

i) $\emptyset \in I$

ii) $Y \in I$ & $X \subseteq Y \Rightarrow X \in I$

Příklad: $E = \{1,2,3\}$

$I = \{\emptyset, \{1,2\}, \{1\}, \{2\}\}$

Def: (E, I) is a Matroid $\equiv$ splňuje exchange axiom

$X, Y \in I$ & $|X| < |Y| \Rightarrow \exists e \in Y \setminus X : X \cup \{e\} \in I$.

$\{1,2\}, \{a,b,c\} \in I \Rightarrow \{1,2,a\} \in I \vee \{1,2,b\} \in I \vee \{1,2,c\} \in I$

Příklad: $G = (V, E)$ je graf, $\mathcal{F} := \{F \mid F \text{ je les v } G\}$

→ $(E, \mathcal{F})$ je graphic matroid of G

i) $\emptyset$ je les

ii) odstraněním hrany z lesa dostanu les

ex) do malého lesa můžu přidat nějakou chybějící hranu z velkého lesa

Lemma: Nechť X, Y jsou lesy v G t.ž. $|X| < |Y|$, pak $\exists e \in Y \setminus X$ t.ž. $X \cup \{e\}$ je les.

Dk:  strom: $|Tree| = |V| - 1$

les: $|F| = |V| - \#\text{stromů}$ → $C(F) := \#\text{stromů}$

↳ důsledek: $|X| < |Y| \Rightarrow C(Y) < C(X)$

→ chci najít hranu z Y, co spojí dva stromy z X

↳ kdyby taková hrana $\nexists$, tedy každá hrana z Y leží v jednom stromě X, tak pro každou komponentu X mám aspoň 1 v Y $\Rightarrow C(Y) \geq C(X)$ ↯ ∎

• Greedy algoritmus vs. matroids

Def: Pro matroid $M=(E,I)$ definujeme funkci $r_M: \mathcal{P}(E) \to \mathbb{R}$

$$X \mapsto \max_{\substack{Y \subseteq X \\ Y \in I}} |Y|$$

↳ rank function of M

→ $r_M(E)$ is rank of M

→ největší množiny nazýváme báze

↳ $X \in I$ je base $\equiv$ $|X| = r_M(E)$

Problem: Dostaneme matroid $M=(E,I)$ a weight-func $w: E \to \mathbb{R}$

Najdi bázi s maximální váhou.

(1) Pokud jsou váhy 1: úloha = najdi nějakou bázi

řešení: díky exchange axiom a ii) můžu greedily

→ beru prvky jeden po jednom, přidávám do množiny

→ pokud $e \in E$ je v nějaké $i \in I$, tak ze ii) je $\{e\} \in I$, takže ho pomocí exchange axiom můžu přidat do své báze

(2) váhy nejsou triviální → taky lze greedy

0. $S_0 := \emptyset$ přidávám hladově elementy z E →
   $U := E(M)$ ← univerzum
1. while $(U \neq \emptyset)$
2. vyber $e \in U$ s největší váhou
3. pokud $S \cup \{e\} \in I$:
   $S \leftarrow S \cup \{e\}$
4. $U \leftarrow U \setminus \{e\}$

Věta: Algoritmus je korektní: Pro $\forall k = 1, \dots r_M(E)$ najde nejtěžší množinu.

→ kdyby pro nějaké k našel $S_k = \{s_1, \dots s_k\}$, $w(s_1) \geq \dots \geq w(s_k)$, která není nejtěžší ind. set velikosti k. Nechť k je nejmenší takové. $T = \{t_1, t_2, \dots t_k\}$ je lepší, ale $w(T \setminus \{t_i\}) \leq w(\{s_1, \dots s_{k-1}\})$

⇒ musí být $w(t_i) > w(s_k)$ ⇒ t_i jsem přidal do S_k pro $\forall t_i$ ⇒ $T = S_k$ ↯.

Věta: Nechť (E, I) je independence system. Pokud pro všechny váhové fce $w: E \to \mathbb{R}^+$ vrátí greedy alg. max-weight independence set, pak (E, I) je matroid.

Důsledek: (E, I) je matroid $\iff$ tam funguje greedy alg.

Dk: Nechť $X, Y \in I$, $|X| < |Y|$ s.t. $\forall e \in Y \setminus X : X \cup \{e\} \notin I$.

→ dojdeme ke sporu

→ definujeme $w(e) := \begin{cases} 1+\epsilon & , e \in X \\ 1 & , e \in Y \setminus X \\ 0 & , \text{other} \end{cases}$

→ greedy alg. najde $X' \supseteq X$, protože prvky z X jsou nejtěžší

$X' = X \cup \underbrace{\text{other}}_{\text{nuly}}$ → $X' \cap (Y \setminus X) = \emptyset$

→ zvolím ϵ dost malé aby Y bylo těžší než X' ☒

- **Rank properties**

1) $0 \leq r(X) \leq |X|$, $\forall X \subseteq E$

2) $X \subseteq Y \Rightarrow r(X) \leq r(Y)$

3) submodularita:

$$r(X \cup Y) \leq r(X) + r(Y) - r(X \cap Y)$$

1) n pekáren, m obchodů — pekárna peče p_i, obchod prodá o_j — převod $p_i \to o_j$ stojí c_{ij}

a) **proměnné**: pro $\forall i \in [n], j \in [m]$: $x_{ij} \in \mathbb{N}_0$ ... # rohlíků transport $p_i \to o_j$

úč. fce: $\min \sum_{i\in[n]} \sum_{j\in[m]} x_{ij} \cdot c_{ij}$ $\quad \hookrightarrow \mathbb{Z}$ & $x_{ij} \geq 0$

podmínky:

1) pro $\forall i \in [n]$: $\sum_{j\in[m]} x_{ij} = p_i$ ... pekárna se všeho zbaví

2) pro $\forall j \in [m]$: $\sum_{i\in[n]} x_{ij} = o_j$ ... obchod vše prodá

b) navíc logistika l_{ij}

proměnné: $x_{ij} \in \mathbb{N}_0$, $I_{ij} \in \{0,1\}$ ... indikátor převáží / nepřeváží

úč. fce: $\min \sum_i \sum_j (x_{ij} c_{ij} + I_{ij} \cdot l_{ij})$

podmínky: 1) & 2) stejné

3) $I_{ij} \geq \underbrace{\frac{x_{ij}}{p_i}}_{\in[0,1]} \iff I_{ij} - \frac{1}{p_i} x_{ij} \geq 0$

2) Grafy $G=(V,E)$, $H=(W,F)$ jsou isomorfní $\equiv$
$\exists$ permutace $\pi: V \to W$ t.ž. $uv \in E \iff \pi(u)\pi(v) \in F$.

Tvrzení: G, H jsou isomorfní $\iff$ $\exists$ permutační matice P t.ž. $AP = PB$, kde A, B jsou matice sousednosti grafů G, H.

Dk: Označme π permutaci odpovídající matici P.

$$A_{ij} = \begin{cases} 1 & \dots\ ij \in E \\ 0 & \dots\ \text{jinak} \end{cases}$$

$$(AP)_{i\pi(j)} = \begin{cases} 1 & \dots\ ij \in E \\ 0 & \dots\ \text{jinak} \end{cases}$$

$$(P^{-1}AP)_{\pi(i)\pi(j)} = \begin{cases} 1 & \dots\ ij \in E \\ 0 & \dots\ \text{jinak} \end{cases}$$

$$B_{\pi(i)\pi(j)} = \begin{cases} 1 & \dots\ \pi(i)\pi(j) \in F \\ 0 & \dots\ \text{jinak} \end{cases}$$

$P^{-1}AP = B \iff (ij \in E \iff \pi(i)\pi(j) \in F)$
$\iff G, H$ jsou isomorfní. ∎

Lineární program:

proměnné: $\forall i,j \in [n]$: $p_{ij} \in \{0,1\}$

úč. fce: $\min 0$

podmínky: $\forall i: \sum_j p_{ij} = 1$, $\forall j: \sum_i p_{ij} = 1$ → Aby to byla perm. matice

$AP = PB$: $\forall ij: \sum_{k=1}^{n} A_{ik} p_{kj} = \sum_{k=1}^{n} p_{ik} B_{kj}$

Lin. prg. - úkol 2

① Def: M je konvexní $\equiv \forall x, y \in M: \lambda x + (1-\lambda) y \in M$, $\lambda \in [0,1]$

Def: Konvexní kombinace bodů $x_1, \dots x_n$ je libovolná

$$\lambda_1 x_1 + \dots + \lambda_n x_n, \quad \sum_i \lambda_i = 1, \quad \forall i: \lambda_i \geq 0.$$

Tvrz: M je konvexní $\iff$ M obsahuje všechny konv. komb. svých bodů.

Dk: $\Leftarrow$: z definice ... $\lambda x + (1-\lambda) y$ je konvexní kombinace

$\Rightarrow$: Indukcí. Pro $n=1$ to triviálně platí, $n=2$ je dané konvexitou M. Předpokládejme, že to platí pro nějaké $n \geq 2$.

Nechť $x_1, \dots x_{n+1} \in M$, $\lambda_1, \dots \lambda_{n+1} \geq 0$, $\sum_i \lambda_i = 1$

$$X = \sum_i \lambda_i x_i = \sum_{i=1}^{n} \lambda_i x_i + \lambda_{n+1} x_{n+1} = \alpha \underbrace{\sum_{i=1}^{n} \frac{\lambda_i}{\alpha} x_i}_{Y} + \lambda_{n+1} x_{n+1}$$

$\Rightarrow$ označme $\alpha := \sum_{i=1}^{n} \lambda_i = 1 - \lambda_{n+1}$

$\Rightarrow$ protože $\sum_{i=1}^{n} \frac{1}{\alpha} \lambda_i = 1$, tak $Y \in M$ ... z i.p.

$\Rightarrow X = \alpha Y + \lambda_{n+1} x_{n+1} = (1-\lambda_{n+1}) Y + \lambda_{n+1} x_{n+1} \in M$ ■

② $v = (1,1,1)$

a) $$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leq \begin{pmatrix} 1 \\ 1 \\ 3 \\ 2 \end{pmatrix}$$

Def: Bod $v \in \mathbb{R}^d$ je vrcholem mnohostěnu $P = \{x \in \mathbb{R}^d \mid Ax \leq b\} \equiv \exists c \in \mathbb{R}^d, t \in \mathbb{R}$ t.ž. $c^T v = t$ & $\forall x \in P: c^T x \leq t$.

$\Rightarrow$ průnik lin. nez. facet snižuje dim. o 1

$\Rightarrow$ musím splnit 3 lin. nez. rovnice a tu poslední nerovnost

I, II, IV jsou s rovností, ale nejsou lin. nez.

III je s nerovností $\Rightarrow$ v není vrchol, ale $v \in P$

b) $$\begin{pmatrix} -1 & -6 & 1 \\ -1 & -2 & 7 \\ 0 & 3 & -10 \\ 1 & 6 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leq \begin{pmatrix} -6 \\ 5 \\ -7 \\ 6 \end{pmatrix}$$

I: $-6 = -6$

II: $4 \leq 5$

III: $-7 = -7$

IV: $6 = 6$

I a IV nejsou lin. nez

$\Rightarrow$ není to vrchol

c) $\max\ 3x_1 + 2x_2$

$$2x_1 + x_2 \le 18$$
$$2x_1 + 3x_2 \le 42 \quad \rightsquigarrow$$
$$3x_1 + x_2 \le 24$$
$$x_1, x_2 \ge 0$$

$$2x_1 + x_2 + S_1 = 18$$
$$2x_1 + 3x_2 + S_2 = 42$$
$$3x_1 + x_2 + S_3 = 24$$
$$x_1, x_2, S_i \ge 0$$

$26 - \frac{7}{3} \cdot 6 + 2$

$$\Rightarrow \begin{aligned} S_1 &= 18 - 2x_1 - x_2 \\ S_2 &= 42 - 2x_1 - 3x_2 \\ S_3 &= 24 - 3x_1 - x_2 \\ \hline z &= 3x_1 + 2x_2 \end{aligned} \xrightarrow[S_3]{x_1} \begin{aligned} S_1 &= 2 + \frac{2}{3}S_3 - \frac{1}{3}x_2 \\ S_2 &= 26 + \frac{2}{3}S_3 - \frac{7}{3}x_2 \\ x_1 &= 8 - \frac{1}{3}S_3 - \frac{1}{3}x_2 \\ \hline z &= 24 - S_3 + x_2 \end{aligned} \xrightarrow[S_1]{x_2} \begin{aligned} x_2 &= 6 + 2S_3 - 3S_1 \\ S_2 &= 12 - 4S_3 + 7S_1 \\ x_1 &= 6 - S_3 + S_1 \\ \hline z &= 30 + S_3 - 3S_1 \end{aligned}$$

$$\xrightarrow[S_2]{S_3} \begin{aligned} x_2 &= 12 + \dots \\ S_3 &= 3 - \frac{1}{4}S_2 + \frac{7}{4}S_1 \\ x_1 &= 3 + \dots \\ \hline z &= 33 - \frac{1}{4}S_2 - \frac{5}{4}S_1 \end{aligned}$$

$$x^* = (3, 12, 0, 0, 3)$$

$$c^T x^* = 33$$

Lineární programování a kombinatorická optimalizace – příklady na 4. cvičení*

18. března 2024

1 Základní pojmy z geometrie

1.1 Afinita

Afinní prostor $A \subseteq \mathbb{R}^d$ má tvar $L + \mathbf{v}$ pro nějaký lineární prostor L a vektor $\mathbf{v} \in \mathbb{R}^d$. Afinní prostor jde určit pomocí soustavy rovnic $Mx = \mathbf{b}$. *Dimenze* afinního prostoru A je rovna dimenzi jeho přidruženého lineárního prostoru L. *Afinní kombinací* vektorů $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n \in \mathbb{R}^d$ je vektor $\sum_{i=1}^n \alpha_i \mathbf{a}_i$, kde $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ $\sum_{i=1}^n \alpha_i = 1$. Množina $V \subseteq \mathbb{R}^d$ je *afinně nezávislá*, pokud platí, že žádný vektor $\mathbf{v} \in V$ není afinní kombinací ostatních vektorů z V. *Afinní obal* $\mathrm{af}(V)$ množiny vektorů $V \subseteq \mathbb{R}^d$ je množina všech afinních kombinací jakékoli konečné podmnožiny vektorů z V.

Příklad 1. *Nechť $A \subseteq \mathbb{R}^d$ je afinní prostor. Z definice je pak A tvaru $A = L + \mathbf{v}$ pro nějaký lineární prostor L a nějaký vektor $\mathbf{v}$. Dokažte, že pro dané $\mathbf{v} \in \mathbb{R}^d$ existuje nanejvýš jeden lineární prostor $L \subseteq \mathbb{R}^d$ takový, že $A = L + \mathbf{v}$.*

(*) Charakterizujte všechny vektory $\mathbf{v}$, které posunou lineární prostor L na afinní prostor A.

1.2 Nadroviny

Nadrovina je libovolný afinní prostor v $\mathbb{R}^d$ dimenze $d-1$. V rovině nadroviny odpovídají přímkám a v $\mathbb{R}^3$ zase rovinám. Nadrovinu lze zapsat jako $\{\mathbf{x} \in \mathbb{R}^d \colon \mathbf{c}^\top \mathbf{x} = h\}$ pro $\mathbf{c} \in \mathbb{R}^d$ a $h \in \mathbb{R}$. Nadrovina rozděluje prostor $\mathbb{R}^d$ na dva *poloprostory*. Nadrovinu samotnou počítáme jako součást obou poloprostorů.

nad

Příklad 2. *(a) Mohou se v $\mathbb{R}^4$ dvě roviny protínat v jednom bodě? Jak mohou vypadat průniky dvou rovin v $\mathbb{R}^4$?* ANO

nad

(b) Mohou se v $\mathbb{R}^5$ dva afinní prostory dimenze 3 protínat v jednom bodě? NE

1.3 Konvexita

Množina $K \subseteq \mathbb{R}^d$ je *konvexní*, pokud pro každé $\mathbf{x}, \mathbf{y} \in K$ a $t \in [0,1]$ platí $t\mathbf{x} + (1-t)\mathbf{y} \in K$. Jinak řečeno, každá úsečka se dvěma konci v K je celá v K. Vektor $\mathbf{x}$ je *konvexní kombinací* vektorů $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$, pokud $\mathbf{x} = \sum_{i=1}^n \alpha_i \mathbf{a}_i$, kde $\alpha_1, \ldots, \alpha_n \in [0,1]$ splňují $\sum_{i=1}^n \alpha_i = 1$. Množina $V \subseteq \mathbb{R}^d$ je v *konvexní poloze* (neboli *konvexně nezávislá*), pokud platí, že žádný vektor $v \in V$ není konvexní kombinací ostatních vektorů z V. *Konvexní obal* $\mathrm{conv}(V)$ množiny vektorů $V \subseteq \mathbb{R}^d$ je množina konvexních kombinací jakékoli konečné podmnožiny vektorů z V.

Příklad 3. *Víme, že v $\mathbb{R}^d$ je maximálně d lineárně nezávislých vektorů a maximálně $d+1$ afinně nezávislých vektorů. Kolik nejvýše je v $\mathbb{R}^d$ konvexně nezávislých vektorů?*

1.4 Mnohostěny

leží P celý na jedné straně nadroviny

Konvexní mnohostěn je průnikem konečně mnoha poloprostorů. Alternativně je konvexním mnohostěnem libovolná množina bodů tvaru $\{\mathbf{x} \in \mathbb{R}^d \colon A\mathbf{x} \leq \mathbf{b}\}$ pro nějakou matici $A \in \mathbb{R}^{m \times d}$ a vektor $\mathbf{b} \in \mathbb{R}^m$. Nechť P je konvexní mnohostěn a $\mathbf{c} \in \mathbb{R}^d, t \in \mathbb{R}$. Jestliže pro každé $\mathbf{x} \in P$ platí $\mathbf{c}^\top \mathbf{x} \leq t$ a zároveň existuje $\mathbf{x} \in P : \mathbf{c}^\top \mathbf{x} = t$, pak množina $\{\mathbf{x} \in \mathbb{R}^d \colon \mathbf{c}^\top \mathbf{x} = t\}$ tvoří *tečnou nadrovinu* H mnohostěnu P. Průniky tečných nadrovin s mnohostěnem P pak nazýváme *stěnami* mnohostěnu P. K nim také započítáváme dvě *nevlastní stěny* $\emptyset$ a P. Stěny dimenzí 0, 1 a $d-1$ nazýváme *vrcholy*, *hrany* a *fasety*. Konvexní mnohostěn je *omezený*, pokud je obsažený v kouli s konečně velkým poloměrem.

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/

Příklad 4. *Jaký je počet stěn krychle a osmistěnu v* $\mathbb{R}^3$*?*

Příklad 5. *Mějme mnohostěn* $P = \{x \in \mathbb{R} \colon x \geq 1 \;\&\; x \leq 2\}$*. Převeďte zápis jeho dvou nerovnicových podmínek do rovnicového tvaru a nakreslete mnohostěn z rovnicového tvaru (jeho prostoru vzroste dimenze).*

Příklad 6. *Nechť* P *je konvexní mnohostěn v* $\mathbb{R}^d$ *a nechť* F *a* G *jsou jeho stěny. Dokažte, že* $F \cap G$ *je stěnou* P*.*

Přesněji $F = P \cap \{\mathbf{x} \in \mathbb{R}^d \colon \mathbf{a}^\top \mathbf{x} = \alpha\}$ *a* $G = P \cap \{\mathbf{x} \in \mathbb{R}^d \colon \mathbf{b}^\top \mathbf{x} = \beta\}$*, kde platí* $\mathbf{a}^\top \mathbf{x} \leq \alpha$ *a* $\mathbf{b}^\top \mathbf{x} \leq \beta$ *pro každé* $\mathbf{x} \in P$*. Uveďte formuli, která stěnu* $F \cap G$ *určuje.*

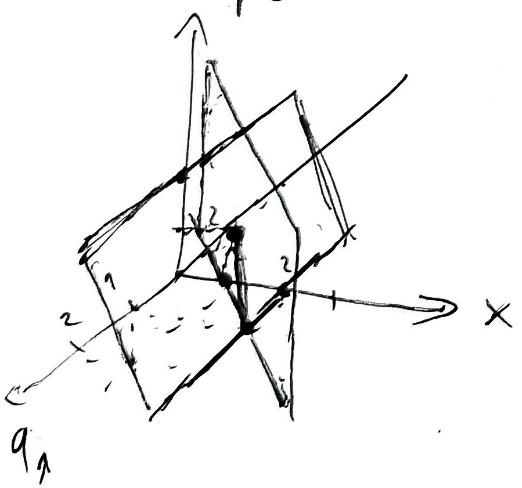
5

P v rovnicovém tvaru

$x \geq 1 \rightsquigarrow x - q_1 = 1$

$x \leq 2 \rightsquigarrow x + q_2 = 2$

$x, q_1, q_2 \geq 0$



$(1,-1,0)$ ↘ roviny

průniky $\underbrace{(1,0,1) \quad , \quad (2,1,0)}$

průnik = conv hull

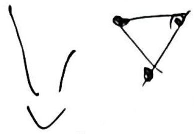
→ pořád to je úsečka a pořád $1 \leq x \leq 2$

LINEÁRNÍ PRG

$v_1 = \alpha v_2 + \beta v_3$

Afinní kombinace:

afinní obal = přímka



afinní obal = rovina

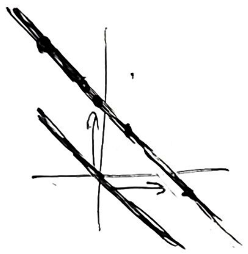
$\Rightarrow$ v $\mathbb{R}^d$ až $d+1$ af. nez. bodů

afinně nezávislé

afinně závislé

(1) $A = L + v$. Tvrzení: pro $\forall v \in \mathbb{R}^d$ $\exists$ nejvýš 1 L t.ž. $A = L + v$

Dk: $A = \{u + v \mid u \in L\} = \{u' + v \mid u \in L'\}$, $L, L' \subseteq \mathbb{R}^d$



$\rightarrow$ chci ukázat $L = L'$, stačí $L \subseteq L'$ + symetrie

$L \subseteq L' \iff \forall u \in L: u \in L'$

Nechť $u \in L$. Potom $w := u + v \in A$, čili $\exists u' \in L': w = u' + v$

Čili $u + v = u' + v \iff u = u'$, tedy $u \in L'$. ■

(*) Charakterizace vektorů v t.ž. $L + v = A$. (nejsou jednoznačné)

Tvrzení: Pokud $A = L + v$, potom pro $\forall w \in A$ je $A = L + w$.

Dk: vezmu a zafixuji nějaký vektor $v \in A$ t.ž. $A = L + v$.

$A = \{u + v \mid u \in L\}$. Chci ukázat, že

$\forall u \in L: A = L + u + v \iff \{u' + v \mid u' \in L\} = \{u' + u + v \mid u' \in L\}$.

Protože $u' \in L$ & $u \in L$, tak $u' + u \in L$. Ještě musím ukázat, že

pro $\forall u'' \in L$ $\exists u' \in L$ t.ž. $u' + u = u''$. Stačí $u' := u'' - u \in L$. ■

Tvrzení: Navíc to jsou právě všechny $v \in A$.

Dk: Pokud $A = L + v$, potom $v \in A$. Ale to je zjevné, protože $0 \in L$. ■

② $\mathbb{R}^d$: Nadrovina je $\{x \in \mathbb{R}^d : c^T x = h\}$ ... afinní podprostor dimenze $d-1$.

a) průnik nadrovin v $\mathbb{R}^4$

$$\{x\in\mathbb{R}^4 \mid c_1^T x = h_1\} \cap \{x\in\mathbb{R}^4 \mid c_2^T x = h_2\} = \{x\in\mathbb{R}^4 \mid c_1^T x = h_1 \ \& \ c_2^T x = h_2\}$$

$c_1^T x = h_1$
$c_2^T x = h_2$

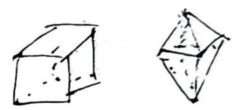
1) nemá řešení
2) kernel má dimenzi 2 ⇒ **rovina** (min.)
jinak dim 3

b) $\mathbb{R}^5$ dva af. p. dim 3

⇒ až 4 podmínky ⇒ 1) nemá řešení
2) kernel má dimenzi 1 ⇒ **přímka** (min.)
dim 1, 2, 3
nic,

c) průnik rovin (dim 2) v $\mathbb{R}^4$

4 rovnice 4 neznámé ⇒ 1 bod, přímka, rovina
jinak



④ # stěn

dim	Krychle	osmistěn	
−1 ($\emptyset$)	1	1	
0	8	6	vrcholy
1	12	12	hrany
2	6	8	facety
3 (full)	1	1	
	28	28	

polární dualita

③ Konečná nezávislost: kolik nejvýš konečně nez. bodů $\mathbb{R}^d$, $d \geq 2$

$$K = \{(x_1, \dots, x_d) \in \mathbb{R}^d \mid x_1^2 + x_2^2 + \dots + x_d^2 = 1\}$$ → sféra → neomezeně

Nechť $n \in \mathbb{N}$. Tvrdím, že K obsahuje n konečně nez. bodů
→ stačí si vzít libovolných n bodů $a_1, \dots, a_n \in K$

Pro spor: Nechť $x \in K$ & $x \in \text{konv}(a_1, \dots, a_n)$ ... x není konečně nez.

BÚNO: $x = (1, 0, 0, \dots, 0)$. Kdyby ne, tak tu sféru zrotuju aby se x zobrazilo na e^1

$x = \sum \alpha_i a_i \Rightarrow x_1 = 1 = \sum \alpha_i a_{i1}$ ... protože $x = e^1$, tak $a_{i1} < 1 \iff a_{i1} + \varepsilon \leq 1$

$\alpha_1, \dots, \alpha_n \in [0,1]$ $\quad \leq (1-\varepsilon) \sum_i \alpha_i = 1 - \varepsilon \Rightarrow 1 \leq 1 - \varepsilon,\ \varepsilon > 0$ ↯

$\sum \alpha_i = 1$

Lineární programování a kombinatorická optimalizace – příklady na 5. cvičení*

25. března 2024

1 Mnohostěny

Množina $K \subseteq \mathbb{R}^d$ je *konvexní*, pokud pro každé $\mathbf{x}, \mathbf{y} \in K$ a $t \in [0,1]$ platí $t\mathbf{x} + (1-t)\mathbf{y} \in K$. Jinak řečeno, každá úsečka se dvěma konci v K je celá v K. *Konvexní mnohostěn* je průnikem konečně mnoha poloprostorů. Alternativně je konvexním mnohostěnem libovolná množina bodů tvaru $\{\mathbf{x} \in \mathbb{R}^d : A\mathbf{x} \le \mathbf{b}\}$ pro nějakou matici $A \in \mathbb{R}^{m \times d}$ a vektor $\mathbf{b} \in \mathbb{R}^m$. Nechť P je konvexní mnohostěn a $\mathbf{c} \in \mathbb{R}^d, t \in \mathbb{R}$. Jestliže pro každé $\mathbf{x} \in P$ platí $\mathbf{c}^\top \mathbf{x} \le t$ a zároveň existuje $\mathbf{x} \in P : \mathbf{c}^\top \mathbf{x} = t$, pak množina $\{\mathbf{x} \in \mathbb{R}^d : \mathbf{c}^\top \mathbf{x} = t\}$ tvoří *tečnou nadrovinu* H mnohostěnu P. Průniky tečných nadrovin s mnohostěnem P pak nazýváme *stěnami* mnohostěnu P. K nim také započítáváme dvě *nevlastní stěny* $\emptyset$ a P. Stěny dimenzí 0, 1 a $d-1$ nazýváme *vrcholy, hrany* a *fasety*.

Příklad 1. *Dokažte, že množina všech optimálních řešení lineárního programu* $\max \mathbf{c}^\top \mathbf{x}$ *za podmínek* $A\mathbf{x} \le \mathbf{b}$, *kde* $\mathbf{x} \ge \mathbf{0}$, *je konvexní množina.* Nebo ukázat, že optima jsou konvexní mnohostěn

$Ax \le b$ & $c^T x =$ optimum

Příklad 2. *Mějme mnohostěn* $P \subseteq \mathbb{R}^3$ *určený množinou vrcholů*

$$\mathbf{a} = (2,1,6), \quad \mathbf{b} = (0,-5,0), \quad \mathbf{c} = (-2,2,-1), \quad \mathbf{d} = (0,-4,0), \quad \mathbf{e} = (0,1,1).$$

Pro každou z následujících rovin určete, zda je vůči P tečná, sečná či mimoběžná. Pro tečné roviny určete dimenzi příslušné stěny.

(a) $\{(x,y,z) \in \mathbb{R}^3 : 5x + 3y - 2z = 1\}$,

(b) $\{(x,y,z) \in \mathbb{R}^3 : x + y - z = 2\}$,

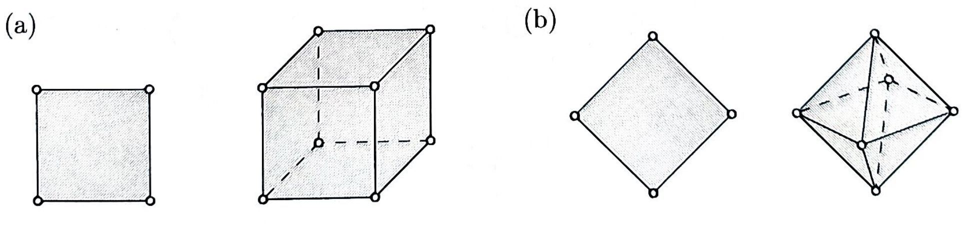
(c) $\{(x,y,z) \in \mathbb{R}^3 : 3x + z = 0\}$.

Příklad 3. *Počty stěn pro známé mnohostěny v* $\mathbb{R}^d$, *kde* $d \in \mathbb{N}$.

(a) Jaký je počet vrcholů a faset d-dimenzionální krychle $[0,1]^d$*?*

(b) Jaký je počet vrcholů a faset d-dimenzionálního křížového mnohostěnu *(neboli zobecněného osmistěnu)*

$$\{(x_1, \dots, x_d) \in \mathbb{R}^d : |x_1| + |x_2| + |x_3| + \dots + |x_d| \le 1\}?$$



Obrázek 1: Příklady (a) d-dimenzionální krychle a (b) d-dimenzionálního křížového mnohostěnu pro $d = 2, 3$.

Příklad 4. *Nalezněte všechny vrcholy mnohostěnu zadaného následujícími nerovnostmi a zdůvodněte, že jiné vrcholy neexistují:*

$$\begin{aligned} x + y + z &\le 3, \\ y + 2z &\le 2, \\ x, y, z &\ge 0. \end{aligned}$$

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/

Příklad 5. *Nechť P je konvexní mnohostěn v $\mathbb{R}^d$ a nechť F a G jsou jeho stěny. Dokažte, že $F \cap G$ je stěnou P.*

Přesněji $F = P \cap \{\mathbf{x} \in \mathbb{R}^d : \mathbf{a}^\top \mathbf{x} = \alpha\}$ a $G = P \cap \{\mathbf{x} \in \mathbb{R}^d : \mathbf{b}^\top \mathbf{x} = \beta\}$, kde platí $\mathbf{a}^\top \mathbf{x} \le \alpha$ a $\mathbf{b}^\top \mathbf{x} \le \beta$ pro každé $\mathbf{x} \in P$. Uveďte formuli, která stěnu $F \cap G$ určuje.

(4) b1

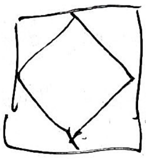
$\{x \in \mathbb{R}^d \mid |x_1| + |x_2| + \dots + |x_d| \le 1\}$

$= \{x \in \mathbb{R}^d \mid \sum_i \varepsilon_i x_i \le 1\} = H(\varepsilon_1, \dots \varepsilon_d)$, $\varepsilon_1, \dots \varepsilon_d \in \{-1, 1\}$

↳ 1 je určující nadrovina

$\Rightarrow P = \bigcap_{\varepsilon \in \{-1,1\}^d} H(\varepsilon) \Rightarrow 2^d$ určujících nadrovin

$\Rightarrow$ mělo by nám vyjít 2^d faset & $2d$ vrcholů



→ vrcholy ⟷ facety
krychle   ortoplex

→ ~~dualita~~ polarita

• $\ge 2^d$ faset

$\Rightarrow$ Mějme $H(\varepsilon_1, \dots \varepsilon_d)$ → musím v ní najít d afinně nez. bodů co to splňují $=$

→ $\forall i \in \{1, \dots d\}$ $\quad \varepsilon_i e_i = (0, \dots \varepsilon_i)$

↳ zvolím vektory kan. báze a $+, -$ podle znaménka ε_i

• Vrcholy $\quad (0, \dots 0 \pm 1, 0 \dots 0)$

musel bych ale ukázat, že každý další vrchol už je komb. těch ostatních vrcholů

Lineární Prg = Mnohostěny

$A(\lambda x_1 + (1-\lambda) x_2) = \lambda \underbrace{A x_1}_{\le b} + (1-\lambda) \underbrace{A x_2}_{\le b}$

→ platí pro b: $\lambda b + (1-\lambda) b = b$ ✓

① LP: $Ax \le b$, max. $c^T x$

→ $\{x \in \mathbb{R}^n \mid x$ je optimum$\}$ je konvexní

→ x_1, x_2 optima $\quad c^T x_1 = c^T x_2 = R$

$c^T(\lambda x_1 + (1-\lambda) x_2) = \lambda c^T x_1 + (1-\lambda) c^T x_2 = \lambda R + (1-\lambda) R = R$

③ a) d-dim krychle je $[0,1]^d$

↳ chci ve tvaru $\{x \in \mathbb{R}^d \mid Ax \le b\}$

$\forall i: x_i \ge 0 \quad x_i \le 1$

↳ $-x_i \le 0$

$d = 2$

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

→ stěna je $P \cap \{x \in \mathbb{R}^d : c^T x = t\}$, kde $\forall x \in P: c^T x \le t$

⇒ $(\forall x)(Ax \le b \Rightarrow c^T x \le t)$

Fasety: Nanejvýš # určujících nadrovin $= 2d$

→ tvrzení: je jich právě $2d$

– ∀ určující nadrovina obsahuje d afinně nezávislých bodů z hyperkrychle

$x_1 = 1$: $(1, 0, \dots, 0)$, $(1, 1, 0 \dots 0)$, $\vdots$, $(1, 1, \dots, 1)$ } d afinně nez.

⇓ afinní obal má dim. $d-1$

→ určuje fasetu

→ Vrcholy: 2^d

$\ge 2^d$: body z $\{0,1\}^d$ → leží v průniku d těch určujících nadrovin

$\le 2^d$: vrchol leží v právě d těch určujících nadrovin, kde pro $\forall i$ leží v právě jedné z těch dvou

→ v d z nich přesně, v d ostatní

⇒ je to stěna

→ potřebujeme, že průnik nadrovin snižuje dimenzi

(3) mnohostěn a vrcholy ... průniky s rovinou

$a=(2,1,6)$ $b=(0,-5,0)$ $c=(-2,2,-1)$ $d=(0,-4,0)$ $e=(0,1,1)$

a) $\{(x,y,z)\in\mathbb{R}^3 \mid 5x+3y-2z=1\}$

$\rightarrow$ dosazení body: a: $1=1$ ... $a\in$ rovina

b: $-15\leq 1$ ... b je pod rovinou

c, d je pod rovinou

e je na rovině

} stejná 2 body $\Rightarrow$ dim 1

b) $x+y-z=2$: všechno pod $\Rightarrow$ je to mimoběžná rovina

c) $3x+z=0$: něco nad, něco pod $\Rightarrow$ sečná rovina

(4) vrcholy mnohostěnu

$x+y+z\leq 3$
$y+2z\leq 2$
$x,y,z\geq 0$

Vrchol: splní aspoň 3 nerovnosti s rovností & splní všechny nerovnosti

$\binom{5}{3}$ kandidátů

$(0,0,0)$
~~$(0,0,2)$~~ není řešení $\rightarrow$ $(0,0,1)$ je
$(0,2,0)$
$(3,0,0)$
$(1,2,0)$, $(2,0,1)$

Lineární programování a kombinatorická optimalizace – příklady na 6. cvičení*

8. dubna 2024

1 Simplexová metoda

Úloha lineárního programování v rovnicovém tvaru je zapsaná jako max $\mathbf{c}^\top \mathbf{x}$ za podmínek $A\mathbf{x} = \mathbf{b}$ a $\mathbf{x} \geq \mathbf{0}$, kde $A \in \mathbb{R}^{m \times n}$, $\mathbf{c}, \mathbf{x} \in \mathbb{R}^n$ a $\mathbf{b} \in \mathbb{R}^m$. Předpokládejme, že $\operatorname{rank}(A) = m$.

Báze je množinou $B \subseteq \{1, \dots n\}$ indexů proměnných takovou, že A_B je regulární, kde A_B značí podmatici A indexovanou sloupci z B. *Bázickým řešením* $\mathbf{x} = (x_1, \dots, x_n)$ odpovídající bázi B je řešení soustavy $A\mathbf{x} = \mathbf{b}$, pro které platí $x_i = 0$ pro každé $i \notin B$. *Přípustná báze* je taková, že jí odpovídající bázické řešení $\mathbf{x}$ je přípustné, tedy $\mathbf{x} \geq \mathbf{0}$.

Vzorový řešený příklad:

$$
\begin{aligned}
\max 2x_1 + x_2 \\
-x_1 + x_2 &\leq 1 \\
x_1 &\leq 3 \\
x_2 &\leq 2 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

Upravíme na rovnicový tvar zavedením nových proměnných $s_1, s_2, s_3 \geq 0$:

$$
\begin{aligned}
\max 2x_1 + x_2 \\
-x_1 + x_2 + s_1 &= 1 \\
x_1 + s_2 &= 3 \\
x_2 + s_3 &= 2 \\
x_1, x_2, s_1, s_2, s_3 &\geq 0
\end{aligned}
$$

Začneme v nějakém přípustném bázickém řešení. Zde lze zvolit původní proměnné $x_1 = x_2 = 0$ a $(s_1, s_2, s_3) = \mathbf{b}^\top = (1, 3, 2)$. Pak přepíšeme soustavu tak, aby bázické proměnné s_1, s_2, s_3 byly na levé straně:

$$
\begin{aligned}
&\max 2x_1 + x_2 \\
s_1 &= 1 + x_1 - x_2 \\
s_2 &= 3 - x_1 \\
s_3 &= 2 - x_2
\end{aligned}
$$

Vstoupíme x_1 do báze, protože má nejvyšší kladný koeficient v účelové funkci, a vystoupíme s_2:

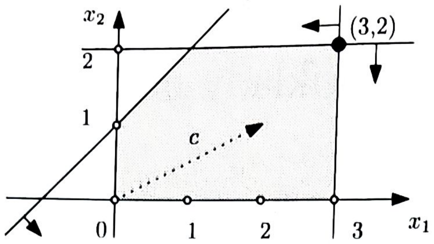
$$
\begin{aligned}
&\max 6 + x_2 - 2s_2 \\
s_1 &= 4 - x_2 - s_2 \\
x_1 &= 3 - s_2 \\
s_3 &= 2 - x_2
\end{aligned}
$$

Vstoupíme x_2 do báze, protože má nejvyšší kladný koeficient v účelové funkci, a vystoupíme s_3:

$$
\begin{aligned}
&\max 8 - 2s_2 - s_3 \\
s_1 &= 2 - s_2 + s_3 \\
x_1 &= 3 - s_2 \\
x_2 &= 2 - s_3
\end{aligned}
$$

Není, co zlepšovat, takže máme optimum pro $x_1 = 3$, $x_2 = 2$, $s_1 = 2$ a $s_2 = s_3 = 0$ s hodnotou účelové funkce 8.

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/



Obrázek 1: Uvedené řešení odpovídá posunu z počátku do vrcholu $(3,0)$ a poté do $(3,2)$.

Pseudokód simplexové metody:

1. *Vstup:* Úloha lineárního programování P v rovnicovém tvaru, max $\mathbf{c}^\top \mathbf{x}$ za podmínek $A\mathbf{x} = \mathbf{b}$ a $\mathbf{x} \geq \mathbf{0}$, kde $A \in \mathbb{R}^{m \times n}$, $\mathbf{c}, \mathbf{x} \in \mathbb{R}^n$ a $\mathbf{b} \in \mathbb{R}^m$. Předpokládáme, že $\operatorname{rank}(A) = m$.
2. *Nalezni počáteční bázické přípustné řešení*: Přenásob soustavu, aby $\mathbf{b} \geq \mathbf{0}$, a vyřeš simplexovou metodou pomocnou úlohu max $-x_{n+1} - \cdots - x_{n+m}$ za $\overline{A}\overline{\mathbf{x}} = \mathbf{b}, \overline{\mathbf{x}} \geq \mathbf{0}$, kde $\overline{A} = (A \mid I_m) \in \mathbb{R}^{m \times (n+m)}$ a $\overline{\mathbf{x}} = (x_1, \ldots, x_{n+m})$. Tato úloha má snadné počáteční řešení $(0, \ldots, 0, b_1, \ldots, b_m)$. Pokud je optimální hodnota záporná, pak **skonči**, protože neexistuje přípustné řešení pro P. Jinak je optimem $(x_1, \ldots, x_n, 0, \ldots, 0)$ a pak je $(x_1, \ldots, x_n)$ počátečním řešením pro P.
3. *Spočítej simplexovou tabulku*: Pro přípustnou bázi $B \subseteq \{1, \ldots, n\}$ přepiš P na max z pro
$$z = z_0 + \mathbf{r}^\top \mathbf{x}_N \text{ za podmínek}$$
$$\mathbf{x}_B = \mathbf{p} + Q\mathbf{x}_N,$$
kde $N = \{1, \ldots, n\} \setminus B$, $\mathbf{p} \in \mathbb{R}^m$, $Q \in \mathbb{R}^{m \times (n-m)}$, $z_0 \in \mathbb{R}$ a $\mathbf{r} \in \mathbb{R}^{n-m}$.
4. *Vrať případné optimum*: Pokud $\mathbf{r} \leq \mathbf{0}$, tak **skonči** a vrať optimum s bázickými proměnnými $\mathbf{x}_B = \mathbf{p}$ a nebázickými proměnnými $\mathbf{x}_N = \mathbf{0}$.
5. *Vyber proměnnou vstupující do báze*: Podle zvoleného pivotovacího pravidla vyber vstupující proměnnou x_t z proměnných x_j s $j \in N$ a $r_j > 0$. Protože není $\mathbf{r} \leq \mathbf{0}$, tak vstupující proměnná x_t vždy existuje. Volbou x_t chceme zvýšit hodnotu účelové funkce.
6. *Vyber proměnnou vystupující z báze*: Uvaž řádky i simplexové tabulky, ve kterých se x_t objevuje, a vyber z nich vystupující proměnnou x_s tak, aby $\frac{-p_s}{Q_{s,t}} = \min_{i \in B: Q_{i,t} < 0} \frac{-p_i}{Q_{i,t}}$. Speciálně tedy musí platit $Q_{s,t} < 0$. Tato volba x_s zajišťuje, že nové bázické řešení je přípustné.

   Pokud vystupující proměnná neexistuje (t-tý sloupec Q je nezáporný), pak **skonči**, protože úloha P je neomezená. Je-li na výběr více vystupujících proměnných, tak vyber podle pivotovacího pravidla, či libovolně, pokud pravidlo ani tak vystupující proměnnou nespecifikuje.
7. *Aktualizuj simplexovou tabulku a iteruj*: Zvol $(B \setminus \{s\}) \cup \{t\}$ jako novou bázi a přepiš simplexovou tabulku tak, aby odpovídala této nové bázi. Pokračuj krokem 4.

V kroce 5 se může stát, že nově vybraná vstupující proměnná nevylepší hodnotu účelové funkce a pak říkáme, že řešení je *degenerované*. To například nastává, pokud je v předešlém kroce na výběr více vystupujících proměnných. U degenerovaných řešení může dojít k *zacyklení* simplexové metody, kdy se nevylepšuje hodnota účelové funkce a algoritmus se nikdy nezastaví. Zacyklení se dá zabránit volbou vhodného pivotovacího pravidla.

Příklady pivotovacích pravidel pro výběr vstupující proměnné x_t a vystupující x_s:

1. *Dantzigovo pravidlo:* Vyber $t \in N$ s maximálním r_t a zvol x_s libovolně z možných proměnných.
2. *Blandovo pravidlo:* Vyber nejmenší možné $t \in N$ a pro něj nejmenší možné $s \in B$. Brání zacyklení, ale je pomalé.

Existuje spousta dalších pivotovacích pravidel (lexikografické, náhodné a další).

Příklad 1. *Převeďte následující soustavu nerovnic do rovnicového tvaru:*

$$
\begin{aligned}
x_1 + x_2 &\leq 3 \\
x_2 + x_3 &\leq 12 \\
x_1 + 3x_2 - x_4 &\geq 7 \\
x_1, x_2, x_3 &\in \mathbb{R} \\
x_4 &\geq 0
\end{aligned}
$$

Převeďte výslednou úlohu v rovnicovém tvaru na lineární program, ze kterého půjde získat bázické přípustné řešení.

Mějme libovolný lineární program s m lineárními nerovnicemi či rovnicemi a n proměnnými. Kolik proměnných nám vždy stačí v rovnicovém tvaru této úlohy a kolik v pomocném LP pro hledání přípustného bázického řešení?

Příklad 2. *Vyřešte pomocí simplexové metody následující úlohu lineárního programování:*

$$
\begin{aligned}
\max 3x_1 + 4x_2 & \\
x_1 + x_2 &\leq 4 \\
2x_1 + x_2 &\leq 5 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

Příklad 3. *Vyřešte pomocí simplexové metody následující úlohu lineárního programování:*

$$
\begin{aligned}
\max 2x_1 - x_2 + 2x_3 & \\
2x_1 + x_2 &\leq 10 \\
x_1 + 2x_2 - 2x_3 &\leq 20 \\
x_2 + 2x_3 &\leq 5 \\
x_1, x_2, x_3 &\geq 0
\end{aligned}
$$

Lineární programování a kombinatorická optimalizace – příklady na 7. cvičení*

15. dubna 2024

1 Simplexová metoda o něco podrobněji

Víme, že simplexová metoda se může zacyklit a pak nikdy neskončí. To nastává pouze v degenerovaném případě a je to jediný způsob, jak metoda může selhat. V praxi toto typicky nenastává a možnost zacyklení se často ignoruje. Jinak se zacyklení dá zabránit volbou vhodného pivotovacího pravidla, poslouží například Blandovo pravidlo.

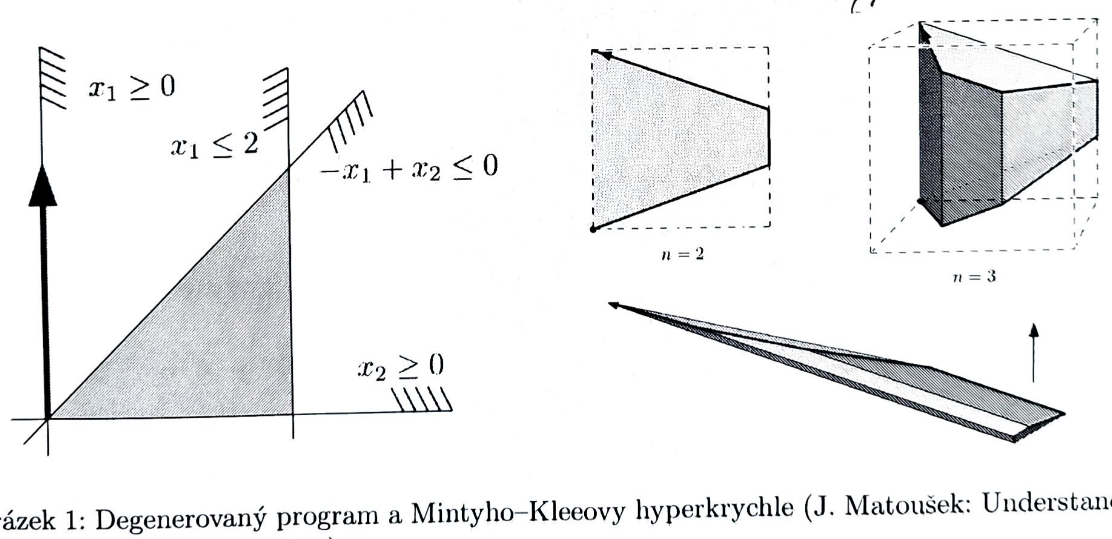
Příklady pivotovacích pravidel pro výběr vstupující proměnné x_t a vystupující x_s:

1. *Dantzigovo pravidlo:* Vyber $t \in N$ s maximálním r_t a zvol x_s libovolně z možných proměnných.
2. *Blandovo pravidlo:* Vyber nejmenší možné $t \in N$ a pro něj nejmenší možné $s \in B$. Brání zacyklení, ale je pomalé.

Existuje spousta dalších pivotovacích pravidel (lexikografické, náhodné a další).

Efektivita simplexové metody: V praxi funguje velmi efektivně, podle počítačových experimentů u úloh v rovnicovém tvaru s m omezeními typicky stačí k nalezení optima $2m$ až $3m$ pivotovacích kroků. Není známé pivotovací pravidlo, pro které by se umělo ukázat, že simplexová metoda skončí v počtu kroků, který je polynomiální vzhledem k počtu omezení m a počtu proměnných n. Naopak pro spoustu pivotovacích pravidel existují příklady v rovnicovém tvaru s $O(n)$ omezeními a $O(n)$ proměnnými, pro které s určitou počáteční bází potřebuje simplexová metoda $2^{\Omega(n)}$ pivotovacích kroků. Tyto příklady jsou ovšem vzácné. Existuje pravděpodobnostní pivotovací pravidlo, pro nějž je známo, že simplexová metoda nad každou vstupní úlohou použije nanejvýš $e^{O(\sqrt{n \ln n})}$ pivotovacích kroků.

? n-dim má 2^n vrchol



Obrázek 1: Degenerovaný program a Mintyho–Kleeovy hyperkrychle (J. Matoušek: Understanding and using linear programming).

Příklad 1. *Vyřešte simplexovou metodou následující úlohu lineárního programování:*

$$\begin{aligned} &\max x_2 \\ -x_1 + x_2 &\leq 0 \\ x_1 &\leq 2 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Jako pivotovací pravidlo použijte Dantzigovo pravidlo.

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/

Příklad 2. *Na následující úlohy lineárního programování aplikujte simplexovou metodu. V nějaké chvíli by již nemělo být možné pokračovat. Zkuste důvodnit, proč se algoritmus zastavil.*

(a) *Optimalizujte funkci* $\max 3x_1 + x_2$ *za podmínek*

$$\begin{aligned} x_1 - x_2 &\leq -1 \\ -x_1 - x_2 &\leq -3 \\ 2x_1 - x_2 &\leq 2 \\ x_1, x_2 &\geq 0. \end{aligned}$$

Po převodu do rovnicového tvaru můžete použít počáteční přípustné bázické řešení s $x_1 = 3$ *a* $x_2 = 4$.

(b) *Optimalizujte funkci* $\max 4x_1 + 5x_2 + 3x_3$ *za podmínek*

$$\begin{aligned} x_1 + x_2 + 2x_3 &\geq 20 \\ 5x_1 + 6x_2 + 5x_3 &\leq 50 \\ x_1 + 3x_2 + 5x_3 &\leq 30 \\ x_1, x_2, x_3 &\geq 0. \end{aligned}$$

Příklad 3. *Vyřešte simplexovou metodou následující úlohu lineárního programování:*

$$\begin{aligned} &\max 5x_1 - 19x_2 - 3x_3 - 4x_4 \\ x_5 &= -0.5x_1 + 2x_2 + 0.5x_3 - 4x_4 \\ x_6 &= -0.5x_1 + 4x_2 + 1x_3 - x_4 \\ x_7 &= 1 - x_1 \\ &x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0. \end{aligned}$$

Jako pivotovací pravidlo použijte Blandovo pravidlo. Změní se výpočet, pokud bychom používali Dantzigovo pravidlo?

Lineární programování a kombinatorická optimalizace – příklady na 9. cvičení*

29. dubna 2024

1 Dualita

Mějme následující úlohu lineárního programování P s n proměnnými a m podmínkami:

$$\max \mathbf{c}^\top \mathbf{x} \text{ za podmínek } A\mathbf{x} \leq \mathbf{b} \text{ a } \mathbf{x} \geq \mathbf{0}. \tag{P}$$

Té budeme říkat *primární lineární program* (neboli *primár*). Jeho *duálním lineárním programem* (neboli *duálem*) nazveme následující lineární program D s m proměnnými a n podmínkami:

$$\min \mathbf{b}^\top \mathbf{y} \text{ za podmínek } A^\top \mathbf{y} \geq \mathbf{c} \text{ a } \mathbf{y} \geq \mathbf{0}. \tag{D}$$

Vysvětlení: při řešení P se snažíme najít lineární kombinaci m podmínek soustavy $A\mathbf{x} \leq \mathbf{b}$ s nějakými koeficienty $y_1, \dots, y_m \geq 0$ takovými, aby výsledná nerovnost měla j-tý koeficient aspoň c_j pro každé $j \in \{1, \dots, n\}$ a pravá strana přitom byla co nejmenší.

Ukazuje se, že program D „hlídá“ program P podle následujícího výsledku, ze kterého například vidíme, že je-li P neomezený, pak D nemá přípustné řešení.

Věta 1 (Slabá věta o dualitě). *Pro každé přípustné řešení $\mathbf{x}$ úlohy P a každé přípustné řešení $\mathbf{y}$ úlohy D platí $\mathbf{c}^\top \mathbf{x} \leq \mathbf{b}^\top \mathbf{y}$.*

Následující zesílení je asi nejdůležitějším teoretickým výsledkem o lineárních programech.

Věta 2 (Silná věta o dualitě). *Pro úlohy P a D nastane právě jedna z následujících čtyř možností:*

(a) Ani P ani D nemá přípustné řešení.

(b) Úloha P je neomezená a D nemá přípustné řešení.

(c) Úloha P nemá přípustné řešení a D je neomezená.

(d) Úlohy P i D mají přípustné řešení. Pak mají i optimální řešení $\mathbf{x}^*$ a $\mathbf{y}^*$ a platí $\mathbf{c}^\top \mathbf{x}^* = \mathbf{b}^\top \mathbf{y}^*$.

Duální lineární programy můžeme uvážit i pro lineární programy v obecném tvaru, stačí postupovat podle následující tabulky. Postup funguje zleva doprava i zprava doleva.

	Primární úloha	Duální úloha
Proměnné	$\mathbf{x} = (x_1, \dots, x_n)$	$\mathbf{y} = (y_1, \dots, y_m)$
Matice	$A \in \mathbb{R}^{m \times n}$	$A^\top \in \mathbb{R}^{n \times m}$
Pravá strana	$\mathbf{b} \in \mathbb{R}^m$	$\mathbf{c} \in \mathbb{R}^n$
Účelová funkce	$\max \mathbf{c}^\top \mathbf{x}$	$\min \mathbf{b}^\top \mathbf{y}$
Podmínky	i-tá podmínka má $\leq$	$y_i \geq 0$
	$\geq$	$y_i \leq 0$
	$=$	$y_i \in \mathbb{R}$
	$x_j \geq 0$	j-tá podmínka má $\geq$
	$x_j \leq 0$	$\leq$
	$x_j \in \mathbb{R}$	$=$

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/

Příklad 1. *Vytvořte duální program D pro následující primární lineární program P:*

$$\begin{aligned} \max 6x_1 + 4x_2 + 2x_3 & \\ 5x_1 + 2x_2 + x_3 &\leq 5 \\ x_1 + x_2 &\leq 2 \\ x_2 + x_3 &\leq 2 \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

Příklad 2. *Vytvořte duální program D pro následující primární lineární program P:*

$$\begin{aligned} \max x_1 - 2x_2 + 3x_4 & \\ x_2 - 6x_3 + x_4 &\leq 4 \\ -x_1 + 3x_2 - 3x_3 &= 0 \\ 6x_1 - 2x_2 + 2x_3 - 4x_4 &\geq 5 \\ x_2 &\leq 0 \\ x_4 &\geq 0 \end{aligned}$$

Příklad 3. *Dokažte nebo vyvraťte tvrzení:*

(a) Pro každý lineární program P platí, že duál duálu P je původní program P. $\to$ ano

(b) Pokud má lineární program optimum s celočíselnými proměnnými, tak má celočíselné optimální řešení i duál.

Příklad 4. *(*) Mějme následující úlohu lineárního programování P:*

$$\max \mathbf{c}^\top \mathbf{x} \text{ za podmínek } A\mathbf{x} \leq \mathbf{b} \text{ a } \mathbf{x} \geq \mathbf{0}.$$

Pomocí duality zkonstruujte soustavu nerovnic, pro kterou ze souřadnic libovolného řešení lze vyčíst optimální řešení pro P.

Tím ukážeme, že asymptotická složitost problému rozhodnout, zda je daný mnohostěn neprázdný, je stejná jako složitost problému nalezení optimálního řešení úlohy lineárního programování. Nebo ještě jinak: nalezení přípustného řešení lineárního programu je asymptoticky stejně obtížné jako nalezení optimálního řešení.

$P \to D \to F \,/\!/\, P$

$\forall x, y: c^T x \leq b^T y \leq c^T x$

Lineární programování a kombinatorická optimalizace – příklady na 10. cvičení*

6. května 2024

1 Dualita její aplikace

Příklad 1. *Sestrojte duální úlohu k lineární relaxaci úlohy* Nalezení minimálního vrcholového pokrytí *ve váženém grafu $G = (V, E, w)$, kde $w\colon V \to \mathbb{R}^+$. Pro připomenutí, tato relaxace vypadá následovně:*

$$\begin{aligned}
\textit{Proměnné:}\quad & x_v \geq 0 \textit{ pro každé } v \in V \\
\textit{Účelová funkce:}\quad & \min \sum_{v \in V} w(v) x_v \\
\textit{Podmínky:}\quad & x_u + x_v \geq 1 \textit{ pro každé } \{u, v\} \in E
\end{aligned}$$

Jaký problém řeší duální úloha pro jednotkové váhy?

Síť je uspořádaná čtveřice (G, z, s, c), kde $G = (V, E)$ je orientovaný graf, neboli $E \subseteq V \times V$, z a s jsou dva různé vrcholy grafu G (zvané *zdroj* a *stok*) a *kapacita* $c\colon E \to \mathbb{R}_0^+$ je funkce ohodnocující hrany. *Tok v síti* je každá funkce $f\colon E \to \mathbb{R}$ splňující $0 \leq f(e) \leq c(e)$ pro každou hranu $e \in E$ a

$$\sum_{v:(u,v) \in E} f(u, v) = \sum_{v:(v,u) \in E} f(v, u)$$

pro každý vrchol $u \in V$ mimo stok a zdroj. *Velikost toku* je

$$w(f) = \sum_{v:(z,v) \in E} f(z, v) - \sum_{v:(v,z) \in E} f(v, z).$$

Řezem v síti je množina R hran vedoucích z množiny vrcholů Z do množiny vrcholů $S = V \setminus S$, kde $z \in Z$ a $s \in S$. *Kapacitou řezu* R je $\sum_{e \in R} c(e)$.

Příklad 2. *Uvažme následující úlohu lineárního programování pro problém* Nalezení maximálního toku *v síti $(G = (V, E), z, s, c)$:*

$$\begin{aligned}
\textit{Proměnné:}\quad & x_e \geq 0 \textit{ pro každé } e \in E \\
\textit{Účelová funkce:}\quad & \max x_{s,z} \\
\textit{Podmínky:}\quad & \sum_{u:(u,v) \in E} x_{u,v} - \sum_{u:(v,u) \in E} x_{v,u} = 0 \textit{ pro každé } v \in V \\
& x_e \leq c(e) \textit{ pro každé } e \in E
\end{aligned}$$

(V tomto programu jsme přidali hranu (s, z) „nekonečně“ velké kapacity, čímž tok cirkuluje a program se tak zjednoduší uvedením podmínek Kirchhoffových zákonů i pro zdroj a stok).

Sestrojte duál této úlohy a (*) nahlédněte, že odpovídá relaxaci úlohy Nalezení řezu minimální kapacity v síti.

Příklad 3. *(a) Uvažte následující lineární program pro orientovaný graf $G = (V, E)$ a jeho vrcholy s a t:*

$$\begin{aligned}
\textit{Proměnné:}\quad & x_v \in \mathbb{R} \textit{ pro každé } v \in V \\
\textit{Účelová funkce:}\quad & \max x_t \\
\textit{Podmínky:}\quad & x_s = 0 \\
& x_v - x_u \leq 1 \textit{ pro každé } (u, v) \in E
\end{aligned}$$

Nahlédněte, že řeší úlohu Nalezení délky nejkratší cesty *mezi vrcholy s a t v G. V účelové funkci je skutečně maximum, i když chceme nalézt nejkratší cestu.*

(b) Zkonstruujte duál k předešlé úloze. Jaký problém duál řeší?

*Informace o cvičení naleznete na http://kam.mff.cuni.cz/~balko/

2 Komplementarita

Věta 1 (Věta o komplementaritě). *Mějme úlohu lineárního programování P a její duál D v následující formě:*

$$\max \mathbf{c}^\top \mathbf{x}, A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}, \tag{P}$$

$$\min \mathbf{b}^\top \mathbf{y}, A^\top \mathbf{y} \geq \mathbf{c}, \mathbf{y} \geq \mathbf{0}. \tag{D}$$

Mějme přípustná řešení $\mathbf{x}^*$ a $\mathbf{y}^*$ pro P a D a označme jako $A_{j,i}$ prvek matice A na pozici (j,i). Pak $\mathbf{x}^*$ a $\mathbf{y}^*$ jsou optimálními řešeními úloh P a D právě tehdy, když platí následující dva vztahy

$$x_i = 0 \textit{ nebo } \sum_{j=1}^{m} A_{j,i} y_j = c_i \textit{ pro každé } i \in \{1, \ldots, n\} \textit{ a}$$

$$y_j = 0 \textit{ nebo } \sum_{i=1}^{n} A_{j,i} x_i = b_j \textit{ pro každé } j \in \{1, \ldots, m\}.$$

První vztah říká, že pro každé i je buď i-tá proměnná primáru nulová nebo je i-tá podmínka duálu *těsná*. Druhý vztah analogicky říká, že pro každé j je buď j-tá proměnná duálu nulová nebo je j-tá podmínka primáru těsná. Tento výsledek nám pomůže ověřovat optimalitu řešení či například určovat optima duálu z optim primáru.

Příklad (Řešený příklad). *Mějme následující primár P a duál D:*

$$\begin{aligned} \max 2x_1 + 4x_2 + 3x_3 + x_4 & \\ 3x_1 + x_2 + x_3 + 4x_4 &\leq 12 \\ x_1 - 3x_2 + 2x_3 + 3x_4 &\leq 7 \\ 2x_1 + x_2 + 3x_3 - x_4 &\leq 10 \\ x_1, x_2, x_3, x_4 &\geq 0 \end{aligned} \tag{P}$$

a

$$\begin{aligned} \min 12y_1 + 7y_2 + 10y_3 & \\ 3y_1 + y_2 + 2y_3 &\geq 2 \\ y_1 - 3y_2 + y_3 &\geq 4 \\ y_1 + 2y_2 + 3y_3 &\geq 3 \\ 4y_1 + 3y_2 - y_3 &\geq 1 \\ y_1, y_2, y_3 &\geq 0 \end{aligned} \tag{D}$$

Buď $\mathbf{x}^* = \left(0, \frac{52}{5}, 0, \frac{2}{5}\right)$ optimem v P. Určete optimum v D.

Řešení. Pode Věty o komplementaritě jsou 2. a 4. nerovnost v D splněny těsně, protože $x_2 \neq 0$ a $x_4 \neq 0$. V primáru P po dosazení hodnot $\mathbf{x}^*$ vidíme, že 2. nerovnost v P není těsná a podle Věty o komplementaritě tedy máme $y_2 = 0$. Dosazením $y_2 = 0$ do 2. a 4. nerovnosti v D zapsaných s rovností dostáváme soustavu

$$\begin{aligned} y_1 + y_3 &= 4 \\ 4y_1 - y_3 &= 1, \end{aligned}$$

jejíž řešení $\mathbf{y}^* = (1, 0, 3)$ je přípustné pro D a tedy je optimem v D. □

Příklad 4. *Optimálním řešením duální úlohy D k následující úloze P je $\mathbf{y}^* = \left(0, 7, \frac{11}{2}, 0\right)$.*

$$\begin{aligned} \max 4x_1 - 2x_2 + 7x_3 & \\ 5x_1 + x_2 - 2x_3 &\leq 12 \\ -x_1 - x_2 + x_3 &\leq -1 \\ 2x_1 + x_2 &\leq 4 \\ x_1 + x_3 &\leq 4 \\ x_1, x_2, x_3 &\geq 0 \end{aligned} \tag{P}$$

(a) *Spočtěte pomocí komplementarity optimální řešení primáru P.*

(b) *Nalezněte vektory $\mathbf{x} \in \mathbb{R}^3$ a $\mathbf{y} \in \mathbb{R}^4$, které splňují oba vztahy z Věty o komplementaritě, ale ani $\mathbf{x}$ a ani $\mathbf{y}$ nejsou optimálními řešeními pro P a D.*

Příklad 5. *Mějme následující zadání duálu D úlohy P:*

$$\begin{aligned} \max 3y_1 + 3y_2 + 6y_3 + 4y_4 & \\ y_1 + 2y_2 + 3y_3 + y_4 &\leq 5 \\ y_1 + y_2 + 2y_3 + 3y_4 &= 3 \\ y_1 + y_2 + y_3 &\geq 1 \\ y_1, y_2, y_3, y_4 &\geq 0 \end{aligned} \tag{D}$$

Přípustným řešením primáru P je $\mathbf{x}' = (4, 0, -1)$. Je toto řešení primáru optimem v P?