FIT ČVUT

MI-LOM

Lineární optimalizace a metody

Dualita

Evropský sociální fond
Praha & EU: Investujeme do vaší budoucnosti

Michal Černý, 2011

Dualita

Evropský sociální fond
Praha & EU: Investujeme do vaší budoucnosti

Z konvergence simplexového algoritmu (Blandova věta) plyne tato věta.

Věta 1. *Nechť polyedr $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}$ je neprázdný, má vrchol a platí $\sup\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\} < \infty$. Pak existuje base B taková, že*

(a) bod $\boldsymbol{x}^*$ definovaný jako řešení systému nezávislých rovnic $\boldsymbol{A}_B\boldsymbol{x} = \boldsymbol{b}_B$ splňuje, že (i) je vrcholem polyedru $\mathcal{F}$ a (ii) platí $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\}$ (to speciálně znamená: optima úlohy $\max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\}$ se nabývá [také] ve vrcholu), a

(b) existuje vektor $\boldsymbol{y}$ takový, že $\boldsymbol{y}_B \geqslant \boldsymbol{0}$, $\boldsymbol{y}_{\overline{B}} = \boldsymbol{0}$ a $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}$.

□

Této věty podstatně využijeme v důkazu věty 4.
Nyní přistupme k formulaci ústředního výsledku teorie lineárního programování.

Definice 2. *Nechť jsou dány $\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$.*

*(a) Polyedr $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}$ se nazývá **primární polyedr** a polyedr $\mathcal{G} := \{\boldsymbol{y} : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c},\ \boldsymbol{y} \geqslant \boldsymbol{0}\}$ se nazývá **duální polyedr**.*

*(b) Úloha $\max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\}$ se nazývá **primární úloha** a úloha $\min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} : \boldsymbol{y} \in \mathcal{G}\}$ se nazývá **duální úloha**.*

*(c) Pro libovolné $\boldsymbol{x} \in \mathcal{F}$ a $\boldsymbol{y} \in \mathcal{G}$ definujeme $\Gamma(\boldsymbol{x}, \boldsymbol{y}) := \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$. Číslo $\Gamma(\boldsymbol{x}, \boldsymbol{y})$ se nazývá **duality gap**.* □

Ze slabé věty o dualitě víme: pro libovolné $\boldsymbol{x} \in \mathcal{F}$ a libovolné $\boldsymbol{y} \in \mathcal{G}$ platí $\Gamma(\boldsymbol{x}, \boldsymbol{y}) \geqslant 0$. Odtud plyne zřejmý důsledek:

Důsledek 3. *Jestliže $\boldsymbol{x} \in \mathcal{F}$ a $\boldsymbol{y} \in \mathcal{G}$ splňují $\Gamma(\boldsymbol{x}, \boldsymbol{y}) = 0$, pak $\boldsymbol{x}$ je optimum primární úlohy a $\boldsymbol{y}$ je optimum duální úlohy.* □

Není ale jasné, zdali antecedent implikace lze vůbec naplnit (jestli to možné není, důsledek neříká nic). Ukážeme, že tomu tak jest.

Věta 4 (von Neumann, silná věta o dualitě)**.** *Nechť primární polyedr je neprázdný. Jestliže existuje optimum $\boldsymbol{x}^*$ primární úlohy, pak také existuje optimum $\boldsymbol{y}^*$ duální úlohy a platí $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}^*$. (Tedy: $\Gamma(\boldsymbol{x}^*, \boldsymbol{y}^*) = 0$.)*

Důkaz. Nechť $\mathcal{F} \neq \emptyset$. Označme $\xi^* := \max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\}$ a $\upsilon^* := \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} : \boldsymbol{x} \in \mathcal{G}\}$.

Předpokládejme, že $\mathcal{F}$ má vrchol; pak na lineární program $\max_{\boldsymbol{x} \in \mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ můžeme spustit simplexový algoritmus. Jestliže ξ^* existuje, algoritmus skončí v kroku [3] s $\boldsymbol{y}$ splňujícím $\boldsymbol{y} \geqslant \boldsymbol{0}$ a $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}$ (viz větu 1). Odtud $\boldsymbol{y} \in \mathcal{G}$, pročež $\mathcal{G} \neq \emptyset$. Podle slabé věty o dualitě víme $\xi^* \leqslant \upsilon^*$; stačí tedy ukázat $\xi^* \geqslant \upsilon^*$. Řekněme (bez újmy na obecnosti), že v poslední iteraci simplexového algoritmu jest $B = \{1, \dots, n\}$; pak $\boldsymbol{y} = \binom{\boldsymbol{y}_B}{\boldsymbol{0}}$ a víme, že $\boldsymbol{A}_B \boldsymbol{x}^* = \boldsymbol{b}_B$, kde $\boldsymbol{x}^*$ je algoritmem nalezený optimální vrchol. Pak

$$\begin{aligned}\xi^* &= \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y})^{\mathrm{T}}\boldsymbol{x}^* = \boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}^* = (\boldsymbol{y}_B^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}) \begin{pmatrix}\boldsymbol{A}_B \\ \boldsymbol{A}_{\overline{B}}\end{pmatrix} \boldsymbol{x}^* = \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{A}_B\boldsymbol{x}^* = \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{b}_B = (\boldsymbol{y}_B^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}) \begin{pmatrix}\boldsymbol{b}_B \\ \boldsymbol{b}_{\overline{B}}\end{pmatrix} \\ &= \boldsymbol{y}^{\mathrm{T}}\boldsymbol{b} = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} \overset{\boldsymbol{y}\in\mathcal{G}}{\geqslant} \min\{\boldsymbol{b}^{\mathrm{T}}\widetilde{\boldsymbol{y}} : \widetilde{\boldsymbol{y}} \in \mathcal{G}\} = \upsilon^*.\end{aligned}$$

Tím je dokončen důkaz případu, kdy $\mathcal{F}$ má vrchol.

Nechť $\mathcal{F}$ je obecný polyedr (který vrchol mít nemusí). Z první fáze simplexového algoritmu víme, že polyedr

$$\mathcal{F}' := \left\{ \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} : \begin{pmatrix} \boldsymbol{A} & -\boldsymbol{A} \\ -\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & -\boldsymbol{I}\end{pmatrix} \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} \leqslant \begin{pmatrix}\boldsymbol{b} \\ \boldsymbol{0} \\ \boldsymbol{0}\end{pmatrix} \right\}$$

má vrchol (zde $\boldsymbol{0}$ značí nulovou matici či vektor příslušného rozměru). Užijme již prokázaný případ na lineární program

$$\max \left\{ (\boldsymbol{c}^{\mathrm{T}}\ \ {-\boldsymbol{c}^{\mathrm{T}}}) \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} : \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} \in \mathcal{F}' \right\}.$$

Obdržíme

$$\max \left\{ (\boldsymbol{c}^{\mathrm{T}}\ \ {-\boldsymbol{c}^{\mathrm{T}}}) \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} : \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} \in \mathcal{F}' \right\} = \min \left\{ (\boldsymbol{b}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}) \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) : \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) \in \mathcal{G}' \right\},$$

kde

$$\mathcal{G}' := \left\{ \begin{pmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{pmatrix} : \begin{pmatrix} \boldsymbol{A}^{\mathrm{T}} & -\boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{A}^{\mathrm{T}} & \boldsymbol{0} & -\boldsymbol{I}\end{pmatrix} \begin{pmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{pmatrix} = \begin{pmatrix}\boldsymbol{c} \\ -\boldsymbol{c}\end{pmatrix},\ \boldsymbol{y}_1 \geqslant \boldsymbol{0},\ \boldsymbol{y}_2 \geqslant \boldsymbol{0},\ \boldsymbol{y}_3 \geqslant \boldsymbol{0} \right\}.$$

Je jasné, že platí

$$\max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\} = \max \left\{ (\boldsymbol{c}^{\mathrm{T}}\ \ {-\boldsymbol{c}^{\mathrm{T}}}) \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} : \begin{pmatrix}\boldsymbol{x}^+ \\ \boldsymbol{x}^-\end{pmatrix} \in \mathcal{F}' \right\};$$

zbývá ukázat, že platí také

$$\min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} : \boldsymbol{y} \in \mathcal{G}\} = \min \left\{ (\boldsymbol{b}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}) \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) : \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) \in \mathcal{G}' \right\}.$$

To je snadné:

$$\begin{aligned}&\min \left\{ (\boldsymbol{b}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}\ \boldsymbol{0}^{\mathrm{T}}) \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) : \left(\begin{smallmatrix}\boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3\end{smallmatrix}\right) \in \mathcal{G}' \right\} \\ &\qquad = \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_1 : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 - \boldsymbol{y}_2 = \boldsymbol{c},\ -\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 - \boldsymbol{y}_3 = -\boldsymbol{c},\ \boldsymbol{y}_1 \geqslant \boldsymbol{0},\ \boldsymbol{y}_2 \geqslant \boldsymbol{0},\ \boldsymbol{y}_3 \geqslant \boldsymbol{0}\} \\ &\qquad = \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_1 : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 \geqslant \boldsymbol{c},\ -\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 \geqslant -\boldsymbol{c},\ \boldsymbol{y}_1 \geqslant \boldsymbol{0}\} \\ &\qquad = \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_1 : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 \geqslant \boldsymbol{c},\ \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 \leqslant \boldsymbol{c},\ \boldsymbol{y}_1 \geqslant \boldsymbol{0}\} \\ &\qquad = \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_1 : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}_1 = \boldsymbol{c},\ \boldsymbol{y}_1 \geqslant \boldsymbol{0}\} \\ &\qquad = \min\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_1 : \boldsymbol{y}_1 \in \mathcal{G}\}. \quad \square\end{aligned}$$

Věta o dualitě má velký význam pro návrh algoritmů pro lineární programování. Připomeňme, že na začátku přednášky jsme zformulovali tři verse úlohy lineárního programování:

(A) je dána racionální matice $\boldsymbol{A}$ a racionální vektor $\boldsymbol{b}$; úkolem je rozhodnout, zdali existuje reálný vektor $\boldsymbol{x}$ splňující $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ (tzv. **rozhodovací verse**),

(B) je dána racionální matice $\boldsymbol{A}$ a racionální vektor $\boldsymbol{b}$; úkolem je rozhodnout, zdali existuje reálný vektor $\boldsymbol{x}$ splňující $\boldsymbol{Ax} \leqslant \boldsymbol{b}$, a je-li tomu tak, je úkolem (nějaké) takové $\boldsymbol{x}$ najít (tzv. **konstruktivní rozhodovací verse**),

(C) je dána racionální matice $\boldsymbol{A}$ a racionální vektory $\boldsymbol{b}$ a $\boldsymbol{c}$. Úkolem je rozhodnout, zdali existuje reálný vektor $\boldsymbol{x}^*$ splňující $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \max\{\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\}$, a je-li tomu tak, je úkolem (nějaké) takové $\boldsymbol{x}^*$ najít; a jestliže $\boldsymbol{x}^*$ neexistuje, úkolem je rozhodnout, zdali je tomu z důvodu $\{\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\} = \emptyset$, anebo z důvodu $\sup\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\} = \infty$ (tzv. **optimalsační verse**).

Věta 5. *Libovolný algoritmus pro (B) dokáže řešit (C).* □

To speciálně znamená:

Důsledek 6. *Jestliže problém (B) je řešitelný v polynomiálním čase, pak také problém (C) je řešitelný v polynomiálním čase.* □

Důkaz věty 5. Dokažme větu v tomto znění: *Systém*

$$\boldsymbol{Ax} \leqslant \boldsymbol{b}, \quad \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}, \quad \boldsymbol{y} \geqslant \boldsymbol{0}, \quad \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} = 0 \tag{1}$$

má řešení, právě když existuje $\max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\}$*; a existuje-li, pak libovolné řešení* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ *systému (1) splňuje*

$$\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\}. \tag{2}$$

Dokažme implikaci *jestliže* $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ *neexistuje, pak systém (1) nemá řešení.* Jestliže číslo $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ neexistuje, může to být ze dvou důvodů: buďto (i) $\mathcal{F} = \emptyset$ anebo (ii) $\mathcal{F} \neq \emptyset$ a funkce $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ roste na $\mathcal{F}$ přes všechny meze.

Jestliže $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ neexistuje z důvodu (i), systém (1) nemá řešení, protože obsahuje nesplnitelnou nerovnost $\boldsymbol{Ax} \leqslant \boldsymbol{b}$.

Jestliže $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ neexistuje z důvodu (ii), pak musí být $\mathcal{G} = \emptyset$. Kdyby totiž bylo $\mathcal{G} \neq \emptyset$, pak existuje $\boldsymbol{y}_0 \in \mathcal{G}$, a pak ze slabé věty o dualitě plyne

$$(\forall \boldsymbol{x} \in \mathcal{F})[\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \leqslant \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_0],$$

a tedy funkce $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ je na $\mathcal{F}$ shora omezená; to je ovšem spor s tím, že $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ roste na $\mathcal{F}$ přes všechny meze. Systém (1) nemůže mít řešení: obsahuje totiž systém $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}$, $\boldsymbol{y} \geqslant \boldsymbol{0}$, a ten je díky $\mathcal{G} = \emptyset$ nesplnitelný.

Opačná implikace. Jestliže $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ existuje, pak z věty o dualitě plyne

$$(\exists \boldsymbol{x}_0 \in \mathcal{F})(\exists \boldsymbol{y}_0 \in \mathcal{G})[\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_0 = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}_0],$$

pročež $(\boldsymbol{x}_0, \boldsymbol{y}_0)$ je řešením systému (1). Tedy, systém (1) je řešitelný.

Zbývá ukázat, že každé řešení $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ systému (1) splňuje (2). Vzhledem k tvaru systému (1) jest $\boldsymbol{x}^* \in \mathcal{F}$ a $\boldsymbol{y}^* \in \mathcal{G}$. Protože $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}^*$, jest $\Gamma(\boldsymbol{x}^*, \boldsymbol{y}^*) = 0$ a tvrzení plyne z důsledku 3.

Algoritmu pro problém (B) stačí předhodit systém (1). Algoritmus pro (B) dokáže rozhodnout, zdali má systém (1) nějaké řešení. Jestliže řešení nemá, pak $\mathcal{F} = \emptyset$ anebo $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ roste na $\mathcal{F}$ přes všechny meze (a snadno dokážeme tyto dva případy rozlišit); jestliže řešení má, najděme jej. Tím získáme řešení problému (C). □

Je zajímavé, že platí i tato věta:

Věta 7. *Libovolný algoritmus pro (A) dokáže řešit (B).* □

To speciálně znamená:

Důsledek 8. *Jestliže problém (A) je řešitelný v polynomiálním čase, pak také problém (B) je řešitelný v polynomiálním čase.* □

Důkaz věty 7 a důsledku 8. Nejprve algoritmem pro (A) otestujeme, zdali $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\} \neq \emptyset$. Je-li $\mathcal{F} \neq \emptyset$, uvážíme obecnější problém. Budiž dán systém

$$\begin{array}{c} \boldsymbol{\alpha}_1^{\mathrm{T}}\boldsymbol{x} \leqslant b_1, \quad \boldsymbol{\alpha}_2^{\mathrm{T}}\boldsymbol{x} \leqslant b_2, \quad \ldots, \quad \boldsymbol{\alpha}_p^{\mathrm{T}}\boldsymbol{x} \leqslant b_p, \\ \boldsymbol{\alpha}_{p+1}^{\mathrm{T}}\boldsymbol{x} = b_{p+1}, \quad \boldsymbol{\alpha}_{p+2}^{\mathrm{T}}\boldsymbol{x} = b_{p+2}, \quad \ldots, \quad \boldsymbol{\alpha}_m^{\mathrm{T}}\boldsymbol{x} = b_m; \end{array} \tag{3}$$

popíšeme algoritmus k nalezení $\boldsymbol{x}$, jež jej splňuje. (Systém $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$ je speciální případ s $p = m$.)

Postupujme rekursivně v počtu nerovností p. Je-li $p = 0$, máme systém rovností, který lze vyřešit Gaussovou eliminací. Víme, že Gaussova eliminace pracuje v polynomiálním čase.

Je-li $p \geqslant 1$, modifikujeme systém (3) tak, že v nerovnici $\boldsymbol{\alpha}_p^{\mathrm{T}}\boldsymbol{x} \leqslant b_p$ nahradíme znak „$\leqslant$“ rovností; dostaneme tak nový systém (X), který má o jednu nerovnici méně. Algoritmem pro (A) otestujme, zdali systém (X) má řešení.

Kruciální pozorování. Jestliže systém (X) *nemá řešení, pak nerovnice* $\boldsymbol{\alpha}_p^{\mathrm{T}}\boldsymbol{x} \leqslant b_p$ *je v systému* (3) *zbytečná.* (To znamená, že množina $\{\boldsymbol{x} : \boldsymbol{x}$ je řešením $(3)\}$ se vypuštěním nerovnice $\boldsymbol{\alpha}_p^{\mathrm{T}}\boldsymbol{x} \leqslant b_p$ nezmění). Pozorování je zřejmé; pro intuici je dobré si jej představit alespoň v rovině.

Jestliže (X) nemá řešení, podle pozorování můžeme volat rekursivně proceduru na nalezení řešení systému

$$\begin{array}{c} \boldsymbol{\alpha}_1^{\mathrm{T}}\boldsymbol{x} \leqslant b_1, \quad \boldsymbol{\alpha}_2^{\mathrm{T}}\boldsymbol{x} \leqslant b_2, \quad \ldots, \quad \boldsymbol{\alpha}_{p-1}^{\mathrm{T}}\boldsymbol{x} \leqslant b_{p-1}, \\ \boldsymbol{\alpha}_{p+1}^{\mathrm{T}}\boldsymbol{x} = b_{p+1}, \quad \boldsymbol{\alpha}_{p+2}^{\mathrm{T}}\boldsymbol{x} = b_{p+2}, \quad \ldots, \quad \boldsymbol{\alpha}_m^{\mathrm{T}}\boldsymbol{x} = b_m, \end{array}$$

který má o jednu nerovnost méně.

Jestliže (X) má řešení, volejme rekursivně proceduru na systém (X); ten má o jednu nerovnost méně než (3) a řešení (X) je i řešením (3).

Je zřejmé, že právě popsaná procedura volá algoritmus pro (A) nanejvýš polynomiálně-krát, a pracuje proto v polynomiálním čase (pracuje-li v polynomiálním čase algoritmus pro (A)). □

Z důsledků 6 a 8 plyne: chceme-li ukázat, že úlohu (C) lze řešit v polynomiálním čase, stačí ukázat, že problém (A) je v $\boldsymbol{P}$. Tak učiníme v další přednášce.