FIT ČVUT

MI-LOM

Lineární optimalizace a metody

Simplexový (Dantzigův) algoritmus

Evropský sociální fond
Praha & EU: Investujeme do vaší budoucnosti

Michal Černý, 2011

Simplexový (Dantzigův) algoritmus

Evropský sociální fond
Praha & EU: Investujeme do vaší budoucnosti

Lemma 1 (slabá věta o dualitě)**.** *Nechť jsou dány* $\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$*. Pro každé* $\boldsymbol{x} \in \mathcal{F} := \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}$ *a* $\boldsymbol{y} \in \mathcal{G} := \{\boldsymbol{y} : \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}, \boldsymbol{y} \geqslant \boldsymbol{0}\}$ *jest*

$$\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \leqslant \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y};$$

speciálně tedy $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \leqslant \min_{\boldsymbol{y}\in\mathcal{G}} \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}$*, jestliže existují.*

Důkaz. Platí $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y})^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} \overset{(\star)}{\leqslant} \boldsymbol{y}^{\mathrm{T}}\boldsymbol{b} = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}$; první rovnost plyne z toho, že $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}$ (z definice $\mathcal{G}$), a nerovnost $(\star)$ plyne z toho, že $\boldsymbol{y} \geqslant \boldsymbol{0}$ (z definice $\mathcal{G}$) a $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$ (z definice $\mathcal{F}$). □

Simplexový (Dantzigův) algoritmus — intuitivní verse

Nechť je dán systém $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$, kde $\boldsymbol{A}$ je plné sloupcové hodnosti. Připomeňme, že symbolem m značíme počet rovnic systému a symbolem n jeho dimensi (=rozměr $\boldsymbol{x}$). Označme dále $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}$ a F budiž systém lineárních *rovností*, které vzniknou nahrazením nerovností $\leqslant$ v systému $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$ rovnostmi. Je-li dána neprázdná množina $B \subseteq \{1, \ldots, m\}$, buď F_B systém lineárních rovností tvořený i_1-, i_2-, $\ldots, i_{|B|}$-tou rovnicí systému F, kde $B = \{i_1, \ldots, i_{|B|}\}$. Má-li k dané B systém F_B jednoznačné řešení *a toto řešení leží v* $\mathcal{F}$, označíme toto řešení $\boldsymbol{r}_B$; jinak není $\boldsymbol{r}_B$ definováno.

Všimněme si, že bod $\boldsymbol{x}_0 \in \mathcal{F}$ je vrcholem, právě když je jednoznačným řešením systému některých n nezávislých rovnic vybraných ze systému F, a tedy je vrcholem, právě když existuje B, $|B| = n$ tak, že $\boldsymbol{x}_0 = \boldsymbol{r}_B$.

Simplexový algoritmus řeší tento problém. *Nechť je dán systém* $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$*, množina* $B \subseteq \{1, \ldots, m\}$, $|B| = n$ *taková, že* $\boldsymbol{r}_B$ *je vrchol* $\mathcal{F}$*, a lineární funkce* $f(\boldsymbol{x}) = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ *taková, že* $f(\mathcal{F}) \geqslant 0$*. Úkolem je rozhodnout, zdali* $\min_{\boldsymbol{x}\in\mathcal{F}} f(\boldsymbol{x}) = 0$.

Algoritmus je založen na následujícím nápadu. Předpokládejme, že systém $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$ je *nedegenerovaný*: to znamená, že každý vrchol $\mathcal{F}$ je jednoznačným řešením některých n rovnic ze systému F a ne více. (Příklad. Obsahuje-li dvoudimensionální systém $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$, jehož vrcholem je $(0, 0)^{\mathrm{T}}$, nerovnice

$$x_1 \geqslant 0, \quad x_2 \geqslant 0, \quad x_1 + x_2 \geqslant 0,$$

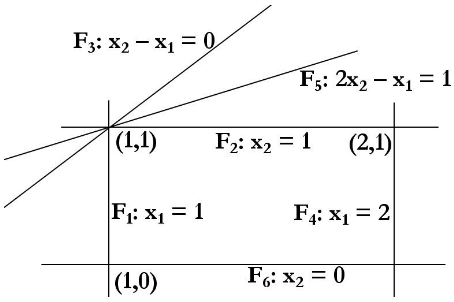
pak tato podmínka není splněna: vrchol $(0, 0)^{\mathrm{T}}$ je řešením systému *tří* rovnic

$$x_1 = 0, \quad x_2 = 0, \quad x_1 + x_2 = 0.$$

Degeneraci ošetříme později, až vyložíme základní myšlenku.) Známe-li vrchol $\boldsymbol{x}_0 := \boldsymbol{r}_B$ polyedru $\mathcal{F}$, uvažme všechny vrcholy, do kterých se z vrcholu $\boldsymbol{x}_0$ dá přejít po jediné hraně polyedru $\mathcal{F}$, a přejděme do takového vrcholu, který způsobí největší pokles hodnoty f. Každý vrchol $\boldsymbol{x}'$, do kterého lze z $\boldsymbol{x}_0$ přejít po jedné hraně polyedru $\mathcal{F}$, lze dostat tak, že se některé $i \in B$ z B odstraní ($B' := B \setminus \{i\}$) a do B se zařadí některé jiné i', které v B nebylo ($B'' := B' \cup \{i'\}$) — $\boldsymbol{x}'$ je pak $\boldsymbol{r}_{B''}$. Systém $F_{B'}$ určuje přímku, jež obsahuje hranu, po níž lze přejít z $\boldsymbol{x}_0$ do $\boldsymbol{x}'$.

To vede k následující myšlence: prozkoumejme všechny sousedy $\boldsymbol{x}_0$ a přejděme do toho souseda, který způsobí největší pokles funkce f; pak celý proces iterujme, dokud bude možné hodnotu f snižovat.

[1] vstup: $\boldsymbol{A}$, $\boldsymbol{b}$, f, B
[2] **for** $i \in B$ **do**
[3]   **for** $j \in \{1, \ldots, m\} \setminus B$ **do**
[4] $B_{i,j} := (B \setminus \{i\}) \cup \{j\}$
[5]   **if** $\boldsymbol{r}_{B_{i,j}}$ není definováno[1] **then** $F_{i,j}$ není definováno
[6]   **else** $F_{i,j} := f(\boldsymbol{r}_{B_{i,j}})$
[7] **end if end do end do**
[8] $(i^*, j^*) = \mathrm{argmin}_{(i,j)}\{F_{i,j} : F_{i,j} \text{ definováno}\}$
[9] výstup: B_{i^*,j^*}.



Obrázek 1: *Degenerace lineárního programu.*

Je-li v [8] *card* argmin$\{\cdots\} \geqslant 1$, vyberme $(i^*, j^*) \in$ argmin$\{\cdots\}$ libovolně. Popsaná procedura představuje jeden krok simplexového algoritmu: druhý krok probíhá tak, že se na vstup za B dosadí výstup B_{i^*,j^*} z prvního kroku etc. Jestliže se krokem algoritmu hodnota funkce f sníží, pokračujeme dalším krokem.

Chtělo by se říci, že jestliže se hodnota funkce f nesníží, můžeme skončit. Ovšem fakt, že se hodnota f nesníží, ještě nemusí znamenat, že ji *není možné* snížit — mohou za to již zmíněné *degenerované lineární programy.* Uvažme například, že dvoudimensionální polyedr $\mathcal{F}$ má vrcholy $(1,0)^{\mathrm{T}}$, $(1,1)^{\mathrm{T}}$, $(2,1)^{\mathrm{T}}$ a $(2,0)^{\mathrm{T}}$. Nechť systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ tvoří nerovnosti

$$x_1 \geqslant 1, \quad x_2 \leqslant 1, \quad x_2 - x_1 \leqslant 0, \quad x_1 \leqslant 2, \quad 2x_2 - x_1 \leqslant 1, \quad x_2 \geqslant 0$$

(viz obrázek 1). V systému F jsou tedy zastoupeny rovnosti

$$F_1 : x_1 = 1, \quad F_2 : x_2 = 1, \quad F_3 : x_2 - x_1 = 0, \quad F_4 : x_1 = 2, \quad F_5 : 2x_2 - x_1 = 1, \quad F_6 : x_2 = 0.$$

Všimněme si, že

$$\boldsymbol{r}_{\{F_1,F_2\}} = \boldsymbol{r}_{\{F_1,F_3\}} = \boldsymbol{r}_{\{F_1,F_5\}} = \boldsymbol{r}_{\{F_2,F_3\}} = \boldsymbol{r}_{\{F_2,F_5\}} = \boldsymbol{r}_{\{F_3,F_5\}} = (1,1)^{\mathrm{T}}.$$

Nechť jest $f(x_1, x_2) = x_1 + \frac{1}{2}x_2$. Začal-li výpočet simplexového algoritmu v bodě $(2,1)^{\mathrm{T}} = \boldsymbol{r}_{\{F_2,F_4\}}$, v první iteraci jsme učinili krok z vrcholu $(2,1)^{\mathrm{T}}$ do vrcholu $(1,1)^{\mathrm{T}}$, který sníží hodnotu f z 2.5 na 1.5. Krok $(2,1)^{\mathrm{T}} \longrightarrow (1,1)^{\mathrm{T}}$ jsme mohli učinit třeba volbou $\{F_2, F_4\} \longrightarrow \{F_2, F_5\} =: B$.

Pokusme se z B učinit další krok. Odstranění rovnosti F_5 z B a její nahrazení libovolnou jinou rovností z F nemůže hodnotu f zlepšit, a pokus o odstranění rovnosti F_2 z B a její nahrazení

[1] Připomeňme, že $\boldsymbol{r}_B$ je definováno, jestliže systém F_B má jednoznačné řešení, které je v $\mathcal{F}$.

libovolnou jinou rovněž nemůže hodnotu f zlepšit, *protože* $F_5 \cap \mathcal{F} = (1,1)^{\mathrm{T}}$. Přitom ale hodnotu f lze snížit krokem $(1,1)^{\mathrm{T}} \longrightarrow (1,0)^{\mathrm{T}}$. Museli bychom ovšem nejprve učinit „krok $(1,1)^{\mathrm{T}} \longrightarrow (1,1)^{\mathrm{T}}$" tak, že $B = \{F_2, F_5\}$ změníme na $B' := \{F_1, F_2\}$ (nebo třeba na $\{F_1, F_5\}$), což sice hodnotu funkce f nesníží, ale otevře nám to cestu pro další krok do vrcholu $(1,0)^{\mathrm{T}} = \boldsymbol{r}_{\{F_1,F_6\}}$.

Příklad ukazuje, že pokud krok simplexové metody nedokázal snížit hodnotu funkce f, ještě nás to neopravňuje prohlásit, že snížení možné není. Je třeba postupovat obezřetněji. Začneme budovat speciální strom. Kořenem je aktuální B (v našem příkladu $B = \{F_2, F_5\}$). Jeho syny budou všechna B', která vzniknou z B tak, že některý prvek F_i v B odstraníme a nahradíme jej některým jiným prvkem F_j, a to tak, aby $\boldsymbol{r}_{B'}$ bylo definováno, $\boldsymbol{r}_{B'} = \boldsymbol{r}_B$ a $B' \neq B$. Označme $\mathcal{B}'$ množinu takto vzniklých synů kořene B. V našem příkladu je

$$\mathcal{B}' = \{B_1' := \{F_3, F_5\},\ B_2' := \{F_2, F_3\},\ B_3' := \{F_1, F_2\},\ B_4' := \{F_1, F_5\}\}.$$

Platí totiž $\boldsymbol{r}_B = \boldsymbol{r}_{B_i'} = (1,1)^{\mathrm{T}}$, a tedy i $f(\boldsymbol{r}_B) = f(\boldsymbol{r}_{B_i'})$, pro $i = 1, \dots, 4$. Tento postup budeme iterovat; ke každému $B' \in \mathcal{B}'$ zkonstruujeme všechny možné syny B'' tak, že z B' některé F_i odstraníme a přidáme jiné F_j, aby bylo splněno, že $\boldsymbol{r}_{B''}$ je definováno, $\boldsymbol{r}_{B''} = \boldsymbol{r}_B$ a přitom B'' *se v dosud zkonstruovaném stromě ještě nevyskytuje.* Kursivou vysázená podmínka je důležitá — jinak by se procedura mohla zacyklit. Konstrukci stromu ukončíme, jestliže (a) se nám podaří snížit hodnotu f, anebo (b) již není možné přidat další syny. V případě (a) pokračujeme přirozeně dalším krokem simplexového algoritmu; v případě (b) už jsme vyčerpali všechny nadějné kandidáty, kteří by mohli otevřít dosud skrytou cestu ke snížení hodnoty f, a tak můžeme skončit a prohlásit, že f *opravdu snížit nelze.*

Dokončeme náš jednoduchý příklad. Začněme konstruovat syny vrcholů z $\mathcal{B}'$; řekněme, že začneme s $B_1' = \{F_3, F_5\}$. Jediným jeho synem bude $\{F_1, F_3\}$ (vrcholy $\{F_3, F_i\}$ s $i = 2, 5$ již ve stromě jsou a $\boldsymbol{r}_{\{F_3,F_i\}}$ s $i = 4, 6$ není definováno; $\{F_j, F_5\}$ s $j = 1, 2, 3$ již ve stromě jsou a pro $j = 4, 6$ není $\boldsymbol{r}_{\{F_j,F_5\}}$ definováno). Pokračujme s konstrukcí pro $\{F_2, F_3\} = B_2' \in \mathcal{B}'$; ten nebude mít syna žádného. Při konstrukci synů pro $\{F_1, F_2\} = B_3' \in \mathcal{B}'$ narazíme i na kandidáta $\{F_1, F_6\}$ — snadno se vidí, že nastal případ (a), ježto $\boldsymbol{r}_{\{F_1,F_6\}} = (1,0)^{\mathrm{T}}$ a tedy $1 = f(\boldsymbol{r}_{\{F_1,F_6\}}) < f(\boldsymbol{r}_B) = 1.5$. Podařilo se nám učinit krok $(1,1)^{\mathrm{T}} \longrightarrow (1,0)^{\mathrm{T}}$, který snížil hodnotu funkce f.

Obecný případ. Simplexový algoritmus řeší jistou speciální úlohu. Obecná úloha lineárního programování je rozhodnout, zdali daný systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ má řešení.[2] Systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ má řešení, právě když systém

$$\boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) + \boldsymbol{y} = \boldsymbol{b}, \quad \boldsymbol{y} \geqslant \boldsymbol{0}, \quad \boldsymbol{x}^+ \geqslant \boldsymbol{0}, \quad \boldsymbol{x}^- \geqslant \boldsymbol{0} \tag{1}$$

má řešení, právě když systém

$$\boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) + \boldsymbol{y}^+ - \boldsymbol{y}^- = \boldsymbol{b}, \quad \boldsymbol{y}^+ \geqslant \boldsymbol{0}, \quad \boldsymbol{y}^- = \boldsymbol{0}, \quad \boldsymbol{x}^+ \geqslant \boldsymbol{0}, \quad \boldsymbol{x}^- \geqslant \boldsymbol{0}$$

má řešení, právě když

$$\min\{\boldsymbol{1}^{\mathrm{T}}\boldsymbol{y}^- : \underbrace{\boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) + \boldsymbol{y}^+ - \boldsymbol{y}^- = \boldsymbol{b},\ \boldsymbol{y}^+ \geqslant \boldsymbol{0},\ \boldsymbol{y}^- \geqslant \boldsymbol{0},\ \boldsymbol{x}^+ \geqslant \boldsymbol{0},\ \boldsymbol{x}^- \geqslant \boldsymbol{0}}_{=:S}\} = 0. \tag{2}$$

Všimněme si, že systém S má vrchol

$$\begin{pmatrix} \boldsymbol{x}_0^+ = \boldsymbol{0} \\ \boldsymbol{x}_0^- = \boldsymbol{0} \\ \boldsymbol{y}_0^+ = \max\{\boldsymbol{b}, \boldsymbol{0}\} \\ \boldsymbol{y}_0^- = \max\{-\boldsymbol{b}, \boldsymbol{0}\} \end{pmatrix}.$$ [3]

[2] Vzhledem k větě o dualitě to stačí i na řešení optimalisační verse úlohy lineárního programování.

Ten je totiž jednoznačným řešením systému rovnic

$$\begin{aligned}
\boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) + \boldsymbol{y}^+ - \boldsymbol{y}^- &= \boldsymbol{b}, \\
\boldsymbol{x}^+ &= \boldsymbol{0}, \\
\boldsymbol{x}^- &= \boldsymbol{0}, \\
(\boldsymbol{y}^+)_i &= 0 \quad \text{pro } i \text{ taková, že } (\boldsymbol{b})_i < 0, \text{ a} \\
(\boldsymbol{y}^-)_i &= 0 \quad \text{pro } i \text{ taková, že } (\boldsymbol{b})_i \geqslant 0
\end{aligned}$$

vybraného ze systému S', kde S' je systém rovnic, které z nerovnic v S vzniknou nahrazením symbolu $\geqslant$ rovností. Jednoznačnost řešení znamená, že je to vrchol. Jistě je

$$f\begin{pmatrix} \boldsymbol{x}^+ \\ \boldsymbol{x}^- \\ \boldsymbol{y}^+ \\ \boldsymbol{y}^- \end{pmatrix} := \mathbf{1}^{\mathrm{T}}\boldsymbol{y}^- = \sum_i (\boldsymbol{y}^-)_i \geqslant 0$$

pro každé $\boldsymbol{x}^+, \boldsymbol{x}^-, \boldsymbol{y}^+, \boldsymbol{y}^-$ splňující S. Můžeme tedy na S a f použít simplexový algoritmus (rovnost v S se dá přepsat jako dvojice nerovností).

Simplexový (Dantzigův) algoritmus — tradiční verse

V této části popíšeme versi simplexového algoritmu, která se uvádí snad v každé učebnici lineárního programování. Využijeme myšlenku popsanou v předchozí části, ovšem poněkud odlišným způsobem budeme volit sousední vrchol, čímž se vyhneme potřebě explicitně konstruovat strom příslušný degenerovanému vrcholu.

Tradiční verse simplexového algoritmu řeší tento problém: *je dán systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$, vektor $\boldsymbol{c}$ a vrchol $\boldsymbol{x}_0$ polyedru $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\}$. Algoritmus buď konstatuje, že funkce $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ roste na $\mathcal{F}$ přes všechny meze, anebo najde vrchol $\boldsymbol{x}^*$ splňující $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}^* = \max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$.*[4]

Předpoklad znamená, že $\mathcal{F}$ je neprázdný a má vrchol. Je-li dán obecný systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$, nevíme, zdali je předpoklad splněn. Můžeme postupovat stejně jako při popisu obecného případu v předchozí části:

1. Voláním (tradiční verse) simplexového algoritmu řešíme úlohu (2) — je-li její optimum $= 0$, získáme zároveň některý vrchol polyedru (1), a je-li její optimum > 0, pak $\mathcal{F} = \emptyset$.

2. Známe-li vrchol polyedru (1), pomocí (tradiční verse) simplexového algoritmu maximalisujeme $\boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}^+ + \boldsymbol{x}^-)$ přes polyedr (1). Obdržíme-li optimum $(\boldsymbol{x}^+, \boldsymbol{x}^-, \boldsymbol{y})$, pak $\boldsymbol{x}^+ - \boldsymbol{x}^-$ je optimum úlohy $\max_{\boldsymbol{x}\in\mathcal{F}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$; jestliže funkce $\boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}^+ + \boldsymbol{x}^-)$ roste na polyedru (1) přes všechny meze, pak také $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ na $\mathcal{F}$ roste přes všechny meze.

Tomuto postupu se říká *dvoufázový algoritmus.*

Poznámka. Jiný (ovšem velmi podobný) způsob je tento. Máme řešit lineární program $F_0 := \{\max \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{x} \in \mathcal{F}\}$, kde $\mathcal{F} := \{\boldsymbol{x} : \boldsymbol{Ax} \leqslant \boldsymbol{b}\}$. Systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ můžeme psát ve tvaru $\boldsymbol{A}_1\boldsymbol{x} \leqslant \boldsymbol{b}_1$ a $\boldsymbol{A}_2\boldsymbol{x} \geqslant \boldsymbol{b}_2$, kde $\boldsymbol{b}_1 \geqslant \boldsymbol{0}$ a $\boldsymbol{b}_2 > \boldsymbol{0}$ (je-li například systém $\boldsymbol{Ax} \leqslant \boldsymbol{b}$ tvaru

$$x_1 + x_2 \leqslant -1, \quad x_1 + 3x_2 \leqslant 0, \quad 2x_1 - x_2 \leqslant 4,$$

pak $\boldsymbol{A}_1\boldsymbol{x} \leqslant \boldsymbol{b}_1$ je systém $x_1 + 3x_2 \leqslant 0,\ 2x_1 - x_2 \leqslant 4$ a $\boldsymbol{A}_2\boldsymbol{x} \geqslant \boldsymbol{b}_2$ je systém s jedinou nerovností $-x_1 - x_2 \geqslant 1$).

Fáze 1. Simplexovým algoritmem řešíme lineární program

$$F_1 := \{\max \mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{y}) : \ \boldsymbol{A}_1(\boldsymbol{x}^+ - \boldsymbol{x}^-) \leqslant \boldsymbol{b}_1, \boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{y} \leqslant \boldsymbol{b}_2,\ \boldsymbol{x}^+ \geqslant \boldsymbol{0}, \boldsymbol{x}^- \geqslant \boldsymbol{0}, \boldsymbol{y} \geqslant \boldsymbol{0}\}.$$

Očividně za počáteční vrchol můžeme vzít $\boldsymbol{x}^+ = \boldsymbol{0}$, $\boldsymbol{x}^- = \boldsymbol{0}$, $\boldsymbol{y} = \boldsymbol{0}$.

---

[3] Maximum z dvojice k-rozměrných vektorů se myslí po složkách, tedy $(\max\{\boldsymbol{a}, \boldsymbol{a}'\})_i = \max\{(\boldsymbol{a})_i, (\boldsymbol{a}')_i\}$ pro $i = 1, \ldots, k$.

[4] Všimněme si, že až prokážeme korektnost algoritmu, získáme také tuto zajímavou větu: *jestliže polyedr $\mathcal{F}$ má vrchol a $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ na $\mathcal{F}$ neroste přes všechny meze, pak svého maxima nabývá (mimo jiné i) v nějakém vrcholu $\mathcal{F}$.*

(i) Jestliže obdržíme optimum $\boldsymbol{x}_1^+, \boldsymbol{x}_1^-, \boldsymbol{y}_1$ úlohy F_1 a platí $\mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1) < \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$, pak $\mathcal{F} = \emptyset$.

(ii) Jestliže obdržíme optimum $\boldsymbol{x}_1^+, \boldsymbol{x}_1^-, \boldsymbol{y}_1$ úlohy F_1 a platí $\mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1) = \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$, pak

$$\text{bod } \boldsymbol{x}_1^+, \boldsymbol{x}_1^- \text{ je vrcholem polyedru } \{\boldsymbol{x} : \boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) \leqslant \boldsymbol{b}, \boldsymbol{x}^+ \geqslant \mathbf{0}, \boldsymbol{x}^- \geqslant \mathbf{0}\}. \tag{3}$$

Fáze 2. Se znalostí vrcholu z (ii) můžeme spustit simplexový algoritmus na úlohu $F_2 := \max\{\boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}^+ - \boldsymbol{x}^-) : \boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) \leqslant \boldsymbol{b}, \boldsymbol{x}^+ \geqslant \mathbf{0}, \boldsymbol{x}^- \geqslant \mathbf{0}\}$. Je-li $\boldsymbol{x}_2^+, \boldsymbol{x}_2^-$ optimum úlohy F_2, pak $\boldsymbol{x}_0 := \boldsymbol{x}_2^+ - \boldsymbol{x}_2^-$ je optimem původní úlohy F_0. Je-li úloha F_2 neomezená (=její účelová funkce roste přes všechny meze), pak také úloha F_0 je neomezená.

Důkaz tvrzení (i). Je-li $\mathcal{F} \neq \emptyset$, existuje $\boldsymbol{x}$ tak, že $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$, a tedy také existují $\boldsymbol{x}^+ \geqslant \mathbf{0}, \boldsymbol{x}^- \geqslant \mathbf{0}$ tak, že $\boldsymbol{A}(\boldsymbol{x}^+ - \boldsymbol{x}^-) \leqslant \boldsymbol{b}$. Odtud také $\boldsymbol{A}_1(\boldsymbol{x}^+ - \boldsymbol{x}^-) \leqslant \boldsymbol{b}_1, \boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) \geqslant \boldsymbol{b}_2$. Proto také existuje $\boldsymbol{y} \geqslant \mathbf{0}$ tak, že $\boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{y} = \boldsymbol{b}_2$ (jest $\boldsymbol{y} = \boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{b}_2 \geqslant \mathbf{0}$). Odtud $\mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{y}) = \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$; tedy maximum úlohy F_1 jest $\geqslant \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$.

Důkaz tvrzení (3). Jestliže $\mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1) = \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$, pak $\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1 = \boldsymbol{b}_2$ (protože $\boldsymbol{b}_2 > \mathbf{0}$ a $\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1 \leqslant \boldsymbol{b}_2$). Jelikož $\boldsymbol{y}_1 \geqslant \mathbf{0}$, jest $\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) \geqslant \boldsymbol{b}_2$. Nerovnost $\boldsymbol{A}_1(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) \leqslant \boldsymbol{b}_1$ je splněna triviálně.

Zbývá ukázat, že nemůže nastat situace $\mathbf{1}^{\mathrm{T}}(\boldsymbol{A}_2(\boldsymbol{x}_1^+ - \boldsymbol{x}_1^-) - \boldsymbol{y}_1) > \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$. To je ale jasné z toho, že $\boldsymbol{A}_2(\boldsymbol{x}^+ - \boldsymbol{x}^-) - \boldsymbol{y} \leqslant \mathbf{1}^{\mathrm{T}}\boldsymbol{b}_2$.

Popis tradiční verse simplexového algoritmu. Je dán systém $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$, vektor $\boldsymbol{c}$ a vrchol $\boldsymbol{x}_0$. Nechť $\boldsymbol{a}_i^T\boldsymbol{x} \leqslant b_i$ je i-tá nerovnost systému $\boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}$ ($i = 1, \ldots, m$). Je-li $B = \{1 \leqslant i_1 < i_2 < \cdots < i_{|B|} \leqslant m\}$, symbolem $\boldsymbol{A}_B$ označíme matici s řádky $\boldsymbol{a}_{i_1}^{\mathrm{T}}, \ldots, \boldsymbol{a}_{i_{|B|}}^{\mathrm{T}}$ a symbolem $\boldsymbol{b}_B$ označíme vektor $(b_{i_1}, \ldots, b_{i_{|B|}})^{\mathrm{T}}$ (a analogicky i s dalšími maticemi a vektory). Symbolem $\overline{B}$ rozumíme množinu $\{1, \ldots, m\} \setminus B$. Je-li například $m = 5$ a $B = \{2, 4\}$, můžeme definovat vektor $\boldsymbol{y} = (0, 3, 1, 7, 5)^{\mathrm{T}}$ předpisem $\boldsymbol{y}_B := (3, 7)^{\mathrm{T}}$, $\boldsymbol{y}_{\overline{B}} := (0, 1, 5)^{\mathrm{T}}$.

[1] Ze systému $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ vybereme n nezávislých rovnic, jejichž řešením je $\boldsymbol{x}_0$. Řekněme, že jsou to rovnice s pořadovými čísly $i_1, \ldots, i_n$. Položme $B_0 := \{i_1, \ldots, i_n\}$. Tedy

$$\boldsymbol{A}_{B_0}\boldsymbol{x}_0 = \boldsymbol{b}_{B_0}. \tag{4}$$

Namísto B_0 nadále píšeme jen B.

[2] Definujme vektor $\boldsymbol{y}$ předpisem

$$\boldsymbol{y}_B := (\boldsymbol{A}_B^{\mathrm{T}})^{-1}\boldsymbol{c}, \quad \boldsymbol{y}_{\overline{B}} := \mathbf{0}. \tag{5}$$

Pozorování A. $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{c}$. *Důkaz.* Bez újmy na obecnosti $B = \{1, \ldots, n\}$. Pak

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} = (\boldsymbol{A}_B^{\mathrm{T}}\ \boldsymbol{A}_{\overline{B}}^{\mathrm{T}}) \begin{pmatrix} \boldsymbol{y}_B \\ \boldsymbol{y}_{\overline{B}} \end{pmatrix} = (\boldsymbol{A}_B^{\mathrm{T}}\ \boldsymbol{A}_{\overline{B}}^{\mathrm{T}}) \begin{pmatrix} \boldsymbol{y}_B \\ \mathbf{0} \end{pmatrix} = \boldsymbol{A}_B^{\mathrm{T}}\boldsymbol{y}_B \stackrel{(5)}{=} \boldsymbol{c}.$$

Tím je pozorování dokončeno.

[3] Jestliže $\boldsymbol{y}_B \geqslant \mathbf{0}$, skončíme — vrchol $\boldsymbol{x}_0$ je optimální.

Důkaz. Bez újmy na obecnosti $B = \{1, \ldots, n\}$. Pak

$$\begin{aligned}
\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_0 &\stackrel{\text{poz. A}}{=} (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y})^{\mathrm{T}}\boldsymbol{x}_0 = \boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}_0 = (\boldsymbol{y}_B^{\mathrm{T}}\ \boldsymbol{y}_{\overline{B}}^{\mathrm{T}}) \begin{pmatrix} (\boldsymbol{A}\boldsymbol{x}_0)_B \\ (\boldsymbol{A}\boldsymbol{x}_0)_{\overline{B}} \end{pmatrix} = (\boldsymbol{y}_B^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}) \begin{pmatrix} (\boldsymbol{A}\boldsymbol{x}_0)_B \\ (\boldsymbol{A}\boldsymbol{x}_0)_{\overline{B}} \end{pmatrix} \\
&= \boldsymbol{y}_B^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}_0)_B = \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{A}_B\boldsymbol{x}_0 \stackrel{(4)}{=} \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{b}_B = (\boldsymbol{y}_B^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}) \begin{pmatrix} \boldsymbol{b}_B \\ \boldsymbol{b}_{\overline{B}} \end{pmatrix} \\
&= \boldsymbol{y}^{\mathrm{T}}\boldsymbol{b} \stackrel{(\star)}{\geqslant} \min\{\boldsymbol{b}^{\mathrm{T}}\overline{\boldsymbol{y}} : \boldsymbol{A}\overline{\boldsymbol{y}} = \boldsymbol{c}, \overline{\boldsymbol{y}} \geqslant \mathbf{0}\} \stackrel{\text{lemma 1}}{\geqslant} \max\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}.
\end{aligned}$$

Nerovnost $(\star)$ plyne z toho, že $\boldsymbol{y} \in \{\overline{\boldsymbol{y}} : \boldsymbol{A}\overline{\boldsymbol{y}} = \boldsymbol{c},\ \overline{\boldsymbol{y}} \geqslant \mathbf{0}\}$ (z pozorování A a z předpokladu $\boldsymbol{y}_B \geqslant \mathbf{0}$). Tím je důkaz dokončen.

[4] Jestliže $\boldsymbol{y}_B \not\geqslant \mathbf{0}$, vyberme

$$i^* := \min\{i : (\boldsymbol{y})_i < 0\}. \tag{6}$$

Zřejmě $i^* \in B$ (protože $\boldsymbol{y}_{\overline{B}} = \mathbf{0}$). Nechť proměnná i^* je ι^*-tá v pořadí v B (je-li např. $B = \{1, 3, 4\}$ a $i^* = 3$, pak $\iota^* = 2$). Položme

$$\boldsymbol{v} := -\boldsymbol{A}_B^{-1}\boldsymbol{e}_{\iota^*}, \tag{7}$$

kde $\boldsymbol{e}_{\iota^*}$ značí ι^*-tý sloupec jednotkové matice.

Pozorování B. $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{v} = -(\boldsymbol{y})_{i^*} > 0$. *Důkaz.* Bez újmy na obecnosti $B = \{1, \dots, n\}$. Pak

$$\boldsymbol{c}^{\mathrm{T}}\boldsymbol{v} \overset{\text{poz. A}}{=} \boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{v} = (\boldsymbol{y}_B^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}) \begin{pmatrix} (\boldsymbol{A}\boldsymbol{v})_B \\ (\boldsymbol{A}\boldsymbol{v})_{\overline{B}} \end{pmatrix} = \boldsymbol{y}_B^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{v})_B = \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{A}_B\boldsymbol{v}$$

$$\overset{(7)}{=} \boldsymbol{y}_B^{\mathrm{T}}\boldsymbol{e}_{\iota^*} = (\boldsymbol{y}_B)_{\iota^*} = (\boldsymbol{y})_{i^*} \overset{(6)}{>} 0.$$

Tím je pozorování dokončeno.

Důsledek pozorování B. (i) *Je-li* $\mu > 0$, *pak* $\boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) > \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_0$.
(ii) $\lim_{\mu\longrightarrow\infty} \boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) = \infty$.
Důkaz je zřejmý.

[5] Jestliže $\boldsymbol{A}\boldsymbol{v} \leqslant \mathbf{0}$, skončíme — funkce $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ roste na $\mathcal{F}$ přes všechny meze.

Důkaz. Je-li $\mu \geqslant 0$, pak

$$\boldsymbol{A}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) = \boldsymbol{A}\boldsymbol{x}_0 + \mu \cdot \boldsymbol{A}\boldsymbol{v} \overset{\boldsymbol{x}_0 \in \mathcal{F}}{\leqslant} \boldsymbol{b} + \mu \cdot \boldsymbol{A}\boldsymbol{v} \overset{\boldsymbol{A}\boldsymbol{v} \leqslant \mathbf{0}}{\leqslant} \boldsymbol{b}.$$

Proto jest $\boldsymbol{x}_0 + \mu\boldsymbol{v} \in \mathcal{F}$ pro libovolné $\mu \geqslant 0$. Ovšem podle důsledku (ii) pozorování B jest $\boldsymbol{c}^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) \longrightarrow \infty$ s $\mu \longrightarrow \infty$. Tím je důkaz dokončen.

[6] Jestliže $\boldsymbol{A}\boldsymbol{v} \not\leqslant \mathbf{0}$, najdeme $\mu^0 := \max\{\mu : \boldsymbol{x}_0 + \mu\boldsymbol{v} \in \mathcal{F}\}$ a položíme

$$\boldsymbol{x}_1 := \boldsymbol{x}_0 + \mu^0\boldsymbol{v}. \tag{8}$$

Pozorování C.

$$\mu^0 = \min\left\{ \frac{b_i - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_0}{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v}} : i \in \{1, \dots, m\}\ \ \&\ \ \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} > 0 \right\}.$$

Důkaz. Máme najít největší μ splňující $\boldsymbol{A}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) \leqslant \boldsymbol{b}$. Nerovnost s pořadovým číslem i má tvar $\boldsymbol{a}_i^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) \leqslant b_i$.

Jestliže $\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} \leqslant 0$, pak i-tá nerovnost je splněna pro libovolné $\mu \geqslant 0$, protože

$$\boldsymbol{a}_i^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) = \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_0 + \mu \cdot \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} \overset{\boldsymbol{x}_0 \in \mathcal{F}}{\leqslant} b_i + \mu \cdot \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} \leqslant b_i;$$

to znamená, že i-tá nerovnost neklade na μ žádné omezení.

Jestliže $\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} > 0$, pak

$$\boldsymbol{a}_i^{\mathrm{T}}(\boldsymbol{x}_0 + \mu\boldsymbol{v}) \leqslant b_i \quad \longleftrightarrow \quad \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_0 + \mu \cdot \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} \leqslant b_i \quad \longleftrightarrow \quad \mu \leqslant \frac{b_i - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_0}{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v}}.$$

Pozorování je dokončeno.

Pozorování D. Jestliže $\boldsymbol{x}_1 \neq \boldsymbol{x}_0$*, pak* conv$(\boldsymbol{x}_0, \boldsymbol{x}_1)$ *je hrana polyedru* $\mathcal{F}$*; speciálně,* $\boldsymbol{x}_1$ *je vrchol.*
Důkaz. Zřejmě $C := \mathrm{conv}(\boldsymbol{x}_0, \boldsymbol{x}_1) = \{\boldsymbol{x}_0 + \mu'\boldsymbol{v} : 0 \leqslant \mu' \leqslant \mu^0\}$. Buď $\mu' \in [0, \mu^0]$ libovolné. Pak

$$\boldsymbol{A}_B(\boldsymbol{x}_0 + \mu'\boldsymbol{v}) \stackrel{(4)}{=} \boldsymbol{b}_B + \mu' \cdot \boldsymbol{A}_B\boldsymbol{v} \stackrel{(7)}{=} \boldsymbol{b}_B + \mu' \cdot \boldsymbol{e}_{\iota^*};$$

to znamená, že libovolný bod z C splňuje systém $n-1$ nezávislých rovnic $\boldsymbol{A}_{B\setminus\{i^*\}}\boldsymbol{x} = \boldsymbol{b}_{B\setminus\{i^*\}}$ (protože $(\boldsymbol{e}_{\iota^*})_k = 0$ pro $k \neq \iota^*$) vybraných ze systému $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. Protože navíc $C \subseteq \mathcal{F}$ a μ^0 jsme volili největší možné, je to hrana. Tím je pozorování dokončeno.[5]

Pozorování E. $\boldsymbol{x}_1 \neq \boldsymbol{x}_0$*, právě když* $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_1 > \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_0$*. Důkaz* plyne z důsledku (i) pozorování B. Tím je pozorování E dokončeno.

[7] Položme

$$J := \{i \in \{1, \ldots, m\} : \underbrace{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v} > 0}_{(\star)} \ \& \ \frac{b_i - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_0}{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v}} = \mu^0\}, \quad j^* := \min J. \tag{9}$$

Nyní víme, že $\boldsymbol{A}_{(B\setminus\{i^*\})\cup\{j^*\}}\boldsymbol{x}_1 = \boldsymbol{b}_{(B\setminus\{i^*\})\cup\{j^*\}}$ (analogie (4)); zřejmě jde o systém nezávislých rovnic definující vrchol $\boldsymbol{x}_1$.

[8] Pokračujme krokem [2] s $\boldsymbol{x}_1$ namísto $\boldsymbol{x}_0$ a s

$$B_1 := (B \setminus \{i^*\}) \cup \{j^*\} \tag{10}$$

namísto B $(= B_0)$.

Poznámka. V (10) říkáme, že index i^* *opustil* B (též *vystoupil z* B) a index j^* *vstoupil do* B.

Poznámka. Všimněme si, že $j^* \in \overline{B}$ (protože z volby j^* jest $\boldsymbol{a}_{j^*}^{\mathrm{T}}\boldsymbol{v} > 0$, zatímco pro $i \in B$ platí $\boldsymbol{a}_i^T\boldsymbol{v} \in \{-1, 0\}$ podle (7)). To speciálně znamená, že $i^* \neq j^*$.

Popis algoritmu je dokončen. Dokázali jsme tuto větu: *jestliže algoritmus skončí, vydá správnou odpověď.*

Věta 2 (Bland)**.** *Simplexový algoritmus skončí.*

Důkaz. Pro spor řekněme, že algoritmus neskončí. Na začátku κ-té iteraci máme vrchol $\boldsymbol{x}_\kappa$ a n-prvkovou množinu indexů B_κ, pro které platí: vrchol $\boldsymbol{x}_\kappa$ je určen jako jednoznačné řešení systému rovnic $\boldsymbol{A}_{B_\kappa}\boldsymbol{x}_\kappa = \boldsymbol{b}_{B_\kappa}$. Hodnoty $i^*, j^*, \boldsymbol{y}, \boldsymbol{v}, \mu^0$ v κ-té iteraci značíme $i^*_\kappa, j^*_\kappa, \boldsymbol{y}_\kappa, \boldsymbol{v}_\kappa, \mu^0_\kappa$.

Možností, jak vybrat n rovnic ze systému $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, je jen konečně mnoho — to znamená, že počítá-li algoritmus věčně, pro nějaká k a k' splňující $k < k'$ jest $B_k = B_{k'}$. Tedy $\boldsymbol{x}_k = \boldsymbol{x}_{k'}$ a $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_k = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}_{k'}$. Odtud podle pozorování E plyne

$$\boldsymbol{x}_k = \boldsymbol{x}_{k+1} = \cdots = \boldsymbol{x}_{k'}; \tag{11}$$

vzhledem k (8) můžeme rovněž psát

$$\mu^0_k = \mu^0_{k+1} = \cdots = \mu^0_{k'-1} = 0. \tag{12}$$

Připomeňme, že v κ-té iteraci je i^*_κ index opouštějící B a j^*_κ je index vstupující do B. Platí tedy

$$i^*_\kappa \in B_\kappa \ \& \ i^*_\kappa \notin B_{\kappa+1}; \quad j^*_\kappa \notin B_\kappa \ \& \ j^*_\kappa \in B_{\kappa+1},$$

[5] Podobně se vidí, že v případě [5] je $\{\boldsymbol{x}_0 + \mu\boldsymbol{v} : \mu \geqslant 0\}$ neomezená hrana $\mathcal{F}$.

viz (10). Označme

$$\mathcal{I}_{out} := \{i_k^*, i_{k+1}^*, \ldots, i_{k'-1}^*\}, \qquad \mathcal{I}_{in} := \{j_k^*, j_{k+1}^*, \ldots, j_{k'-1}^*\}. \tag{13}$$

Tedy, $\mathcal{I}_{out}$ je množina indexů vystupujících z B a $\mathcal{I}_{in}$ je množina indexů vstupujících do B během iterací $k, k+1, \ldots, k'-1$.

Pozorování F. $\mathcal{I}_{out} = \mathcal{I}_{in}$. Jinými slovy: pro daný index j během iterací $k, k+1, \ldots, k'-1$ nastává právě jedna z možností: (i) j nejprve vstoupí do B a poté z B vystoupí; (ii) j nejprve vystoupí z B a poté se do B vrátí; (iii) j je celou dobu v B; (iv) j je celou dobu mimo B. Důkaz. Proberme inklusi $\subseteq$; opačná inkluse se nahlédne analogicky. Jestliže $j \in B_k$ a poté index j vystupuje z B, musí později do B opět vstoupit — jinak by nemohlo být $B_k = B_{k'}$. Jestliže $j \notin B_k$ a index j v některé z iterací vystupuje z B, musel do B nejprve vstoupit. Tím je pozorování dokončeno.

Označme

$$I^* := \max \mathcal{I}_{out}. \tag{14}$$

Podle pozorování F jest $I^* \in \mathcal{I}_{in}$. Víme tedy, že existují $k_{out}, k_{in} \in \{k, k+1, \ldots, k'-1\}$ takové, že

$$i_{k_{out}}^* = I^*, \quad j_{k_{in}}^* = I^*.$$

Můžeme říci: I^* vystupuje z B v iteraci k_{out} a vstupuje do B v iteraci k_{in}.

Pozorování G. Nechť $j > I^*$. Jestliže $(\exists \kappa \in \{k, k+1, \ldots, k'\})[j \in B_\kappa]$, pak $(\forall \kappa \in \{k, k+1, \ldots, k'\})[j \in B_\kappa]$. Důkaz. Podle (14) jest $j \notin \mathcal{I}_{out}$, a tedy také $j \notin \mathcal{I}_{in}$ (podle pozorování F); případy (i) a (ii) z tvrzení pozorování F tudíž nenastávají. Jestliže $(\exists \kappa \in \{k, k+1, \ldots, k'\})[j \in B_\kappa]$, nastává případ (iii). Tím je pozorování dokončeno.

Podle pozorování A v k_{out}-té iteraci platí $\boldsymbol{y}_{k_{out}}^{\mathrm{T}} \boldsymbol{A} = \boldsymbol{c}^{\mathrm{T}}$. Podle pozorování B v k_{in}-té iteraci platí $\boldsymbol{c}^{\mathrm{T}} \boldsymbol{v}_{k_{in}} > 0$. Odtud

$$\sum_{j=1}^{m} \underbrace{(\boldsymbol{y}_{k_{out}})_j \cdot (\boldsymbol{a}_j^{\mathrm{T}} \boldsymbol{v}_{k_{in}})}_{=:\nu_j} = \boldsymbol{y}_{k_{out}}^{\mathrm{T}} \boldsymbol{A} \boldsymbol{v}_{k_{in}} = \boldsymbol{c}^{\mathrm{T}} \boldsymbol{v}_{k_{in}} > 0.$$

To znamená, že alespoň pro jedno j platí $\nu_j > 0$; nechť je to třeba j_0. Jestliže $j_0 \notin B_{k_{out}}$, pak $j_0 \in \overline{B}_{k_{out}}$; podle (5) je ovšem $(\boldsymbol{y}_{k_{out}})_{j_0} = 0$, a tedy $\nu_{j_0} = 0$ — spor. Dokázali jsme

$$j_0 \in B_{k_{out}}. \tag{15}$$

Ukažme, že tvrzení $\nu_{j_0} > 0$ vede ke sporu.

- Nechť $j_0 > I^*$. Protože $j_0 \in B_{k_{out}}$, podle pozorování G jest $j_0 \in B_{k_{in}}$ a $j_0 \in B_{k_{in}+1}$. To znamená, že

  $$j_0 \neq i_{k_{in}}^*$$

  ($i_{k_{in}}^*$ je totiž index vystupující z B v iteraci k_{in}; pro něj tudíž platí $i_{k_{in}}^* \notin B_{k_{in}+1}$). Vztah (7) můžeme v k_{in}-té iteraci číst takto:

  $$(\forall j \in B_{k_{in}}) \ \ \boldsymbol{a}_j^{\mathrm{T}} \boldsymbol{v}_{k_{in}} = \begin{cases} 0 & \text{pro } j \neq i_{k_{in}}^*, \\ -1 & \text{pro } j = i_{k_{in}}^*. \end{cases}$$

  Odtud $\boldsymbol{a}_{j_0}^{\mathrm{T}} \boldsymbol{v}_{k_{in}} = 0$, a tedy $\nu_{j_0} = 0$. Spor.

- Nechť $j_0 = I^*$. Dokážeme (i) $(\boldsymbol{y}_{k_{out}})_{j_0} < 0$ a (ii) $\boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{v}_{k_{in}} > 0$; odtud $\nu_{j_0} < 0$ — spor.

  Důkaz (i). V k_{out}-té iteraci vystupuje index I^* z B. To znamená, že $i^*_{k_{out}} = I^* = j_0$. Podle (6) jest $0 > (\boldsymbol{y}_{k_{out}})_{i^*_{k_{out}}} = (\boldsymbol{y}_{k_{out}})_{j_0}$.

  Důkaz (ii). V k_{in}-té iteraci vstupuje index I^* do B. To znamená, že $j^*_{k_{in}} = I^* = j_0$. Díky podmínce $(\star)$ v (9) jest $0 < \boldsymbol{a}_{j^*_{k_{in}}}^{\mathrm{T}}\boldsymbol{v}_{k_{in}} = \boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{v}_{k_{in}}$.

- Nechť $j_0 < I^*$. Dokážeme (i) $(\boldsymbol{y}_{k_{out}})_{j_0} \geqslant 0$ a (ii) $\boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{v}_{k_{in}} \leqslant 0$; odtud $\nu_{j_0} \leqslant 0$ — spor. (Zde se ukáže význam minim v (6) a v (9).)

  Důkaz (i). V k_{out}-té iteraci vystupuje index I^* z B; jest tedy $i^*_{k_{out}} = I^*$. Podle (6) je $i^*_{k_{out}}$ *nejmenší* index i splňující $(\boldsymbol{y}_{k_{out}})_i < 0$. Protože $j_0 < I^* = i^*_{k_{out}}$, platí $(\boldsymbol{y}_{k_{out}})_{j_0} \geqslant 0$.

  Důkaz (ii). V k_{in}-té iteraci vstupuje index I^* do B; jest tedy $j^*_{k_{in}} = I^*$. Podle (9) je $j^*_{k_{in}}$ *nejmenší* index j splňující $\boldsymbol{a}_j^{\mathrm{T}}\boldsymbol{v}_{k_{in}} > 0$ & $\frac{b_i - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}_{k_{in}}}{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{v}_{k_{in}}} = \mu^0_{k_{in}}$. Protože $j_0 < I^* = j^*_{k_{in}}$, jest

  $$\underbrace{\boldsymbol{a}_j^{\mathrm{T}}\boldsymbol{v}_{k_{in}} \leqslant 0}_{(\alpha)} \quad\vee\quad \underbrace{\frac{b_{j_0} - \boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{x}_{k_{in}}}{\boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{v}_{k_{in}}} \neq \mu^0_{k_{in}}}_{(\beta)}.$$

  Jestliže nastává případ (α), je tvrzení (ii) splněno.

  Ukážeme, že případ (β) nemůže nastat. Nechť (β) platí. Podle (12) jest $\mu^0_{k_{in}} = 0$; výraz (β) tedy lze psát ve tvaru $\boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{x}_{k_{in}} \neq b_{j_0}$. Podle (11) také platí $\boldsymbol{a}_{j_0}^{\mathrm{T}}\boldsymbol{x}_{k_{out}} \neq b_{j_0}$. Pak $j_0 \notin B_{k_{out}}$ (podle (4) totiž $j_0 \in B_{k_{out}}$ implikuje rovnost) — spor s (15). $\square$

Všimněme si, že pro nedegenerované lineární programy je důkaz konečnosti triviální. Na degenerovaný případ se dá nahlížet tak, že volba minim v (6) a v (9) je metodou k (implicitnímu) procházení stromu $\{B\} \cup \mathcal{B}' \cup \mathcal{B}'' \cup \cdots$ příslušného degenerovanému vrcholu, který jsme v intuitivní versi algoritmu konstruovali explicitně.