

Simplexová metoda na řešení lineárního programování

Tomáš Werner

(Netiskněte, obsahuje animace!)

Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázové řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázové řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázové řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$[\mathbf{A} \quad \mathbf{b}] = \begin{array}{cccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 & \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 & \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 & \end{array}$$



Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázové řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázové řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázové řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$[\mathbf{A} \quad \mathbf{b}] = \begin{array}{cccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 \end{array}$$

- ▶ $\{2, 3, 5\}$ není báze (sloupce jsou lineárně závislé)

Báze a bázevé řešení

Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázevé řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázevé řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázevé řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$\begin{aligned} [\mathbf{A} \quad \mathbf{b}] &= \begin{array}{ccccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 \end{array} \\ \mathbf{x} &= \begin{array}{ccccccc} 3 & 0 & 0 & 4 & 1 & 0 \end{array} \end{aligned}$$

- ▶ $\{1, 4, 5\}$ je báze. Bázevé řešení \mathbf{x} je řešením soustavy

$$\begin{bmatrix} -1 & 1 & 0 \\ 1 & 0 & 1 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix}, \quad x_2 = x_3 = x_6 = 0.$$

Je přípustné, není degenerované.

Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázové řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázové řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázové řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$\begin{aligned} [\mathbf{A} \quad \mathbf{b}] &= \begin{array}{cccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 \end{array} \\ \mathbf{x} &= \begin{array}{cccccc} 4 & -1 & 0 & 6 & 0 & 0 \end{array} \end{aligned}$$

- ▶ $\{1, 2, 4\}$ je báze. Bázové řešení je nepřípustné, není degenerované.

Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázeové řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázeové řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázeové řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$\begin{aligned} [\mathbf{A} \quad \mathbf{b}] &= \begin{array}{ccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 \end{array} \\ \mathbf{x} &= \begin{array}{cccccc} 0 & 0 & 1 & -2 & 0 & 0 \end{array} \end{aligned}$$

- ▶ $\{3, 4, 5\}$ je báze. Bázeové řešení je nepřípustné a degenerované.

Vstupní mnohostěn $\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$ kde $\mathbf{A} \in \mathbb{R}^{m \times n}$ a $\text{rank } \mathbf{A} = m$.

- ▶ **Báze** je množina $J \subseteq \{1, \dots, n\}$ taková, že $|J| = m$ a sloupce matice \mathbf{A} s indexy J jsou lineárně nezávislé (tvoří tedy regulární matici $m \times m$).
- ▶ **Bázevé řešení** příslušné bázi J je řešení $\mathbf{x} = (x_1, \dots, x_n)$ soustavy $\mathbf{Ax} = \mathbf{b}$ takové, že $x_j = 0$ pro všechna $j \notin J$. Je právě jedno.
- ▶ Bázevé řešení \mathbf{x} je **přípustné**, pokud $\mathbf{x} \geq \mathbf{0}$.
- ▶ Bázevé řešení \mathbf{x} je **degenerované**, pokud má méně než m nenulových složek.

$$\begin{array}{r} [\mathbf{A} \quad \mathbf{b}] = \\ \mathbf{x} = \end{array} \begin{array}{ccccccc} -1 & 1 & 3 & 1 & 0 & 2 & 1 \\ 1 & 0 & 4 & 0 & 1 & 4 & 4 \\ -1 & 0 & 4 & 1 & 1 & 4 & 2 \\ 0 & 0 & 1 & -2 & 0 & 0 & \end{array}$$

- ▶ Stejné bázevé řešení odpovídá bázi $\{3, 4, 6\}$.
Degenerované bázevé řešení odpovídá více než jedné bázi!

Simplexová metoda používá jen **standardní báze**. Výhody:

- ▶ Ihned vidíme bázové řešení \mathbf{x} : jeho nenulové složky jsou složky \mathbf{b} .
- ▶ Tedy: bázové řešení \mathbf{x} je přípustné $\Leftrightarrow \mathbf{b} \geq \mathbf{0}$

$$\begin{array}{r} [\mathbf{A} \quad \mathbf{b}] = \\ \mathbf{x} = \end{array} \begin{array}{r} 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\ 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\ 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\ 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0 \end{array}$$

Bázové řešení příslušné (standardní) bázi $\{1, 4, 5\}$ je řešením soustavy

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 1 \end{bmatrix}, \quad x_2 = x_3 = x_6 = 0$$

Ekvivalentní řádkové úpravy:

- ▶ Řádek matice $[\mathbf{A} \ \mathbf{b}]$ vynásob nenulovým číslem.
- ▶ K řádku matice $[\mathbf{A} \ \mathbf{b}]$ přičti lineární kombinaci ostatních řádků.

Nemění množinu řešení soustavy $\mathbf{Ax} = \mathbf{b}$.

Chceme bázový sloupec $j' \in J$ nahradit nebázovým sloupcem $j \notin J$:

- ▶ Necht' i je řádek, kde $a_{ij'} = 1$. Prvek (i, j) se nazývá **pivot**.
- ▶ Proveď **ekvivalentní úpravu kolem pivotu** (i, j) , tj. nastav $a_{ij} = 1$ a $a_{i'j} = 0$ pro každé $i' \neq i$:

Přechod k sousední standardní bázi

Ekvivalentní řádkové úpravy:

- ▶ Řádek matice $[\mathbf{A} \ \mathbf{b}]$ vynásob nenulovým číslem.
- ▶ K řádku matice $[\mathbf{A} \ \mathbf{b}]$ přičti lineární kombinaci ostatních řádků.

Nemění množinu řešení soustavy $\mathbf{Ax} = \mathbf{b}$.

Chceme bázový sloupec $j' \in J$ nahradit nebázovým sloupcem $j \notin J$:

- ▶ Necht' i je řádek, kde $a_{ij'} = 1$. Prvek (i, j) se nazývá **pivot**.
- ▶ Proveď **ekvivalentní úpravu kolem pivotu** (i, j) , tj. nastav $a_{ij} = 1$ a $a_{i'j} = 0$ pro každé $i' \neq i$:

$$\begin{array}{cccccccc} & & 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ & & 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ i & & 0 & -1 & 1 & 0 & 1 & 2 & 1 \\ & & & j & & & j' & & \end{array}$$

Přechod k sousední standardní bázi

Ekvivalentní řádkové úpravy:

- ▶ Řádek matice $[\mathbf{A} \ \mathbf{b}]$ vynásob nenulovým číslem.
- ▶ K řádku matice $[\mathbf{A} \ \mathbf{b}]$ přičti lineární kombinaci ostatních řádků.

Nemění množinu řešení soustavy $\mathbf{Ax} = \mathbf{b}$.

Chceme bázový sloupec $j' \in J$ nahradit nebázovým sloupcem $j \notin J$:

- ▶ Necht' i je řádek, kde $a_{ij'} = 1$. Prvek (i, j) se nazývá **pivot**.
- ▶ Proveď **ekvivalentní úpravu kolem pivotu** (i, j) , tj. nastav $a_{ij} = 1$ a $a_{i'j} = 0$ pro každé $i' \neq i$:
 - 1 Vyděl řádek i pivotem a_{ij} .

$$\begin{array}{cccccccc} & & 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ & & 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ i & & 0 & \mathbf{1} & -1 & 0 & -1 & -2 & -1 \\ & & & j & & & j' & & \end{array}$$

Přechod k sousední standardní bázi

Ekvivalentní řádkové úpravy:

- ▶ Řádek matice $[\mathbf{A} \ \mathbf{b}]$ vynásob nenulovým číslem.
- ▶ K řádku matice $[\mathbf{A} \ \mathbf{b}]$ přičti lineární kombinaci ostatních řádků.

Nemění množinu řešení soustavy $\mathbf{Ax} = \mathbf{b}$.

Chceme bázový sloupec $j' \in J$ nahradit nebázovým sloupcem $j \notin J$:

- ▶ Necht' i je řádek, kde $a_{ij'} = 1$. Prvek (i, j) se nazývá **pivot**.
- ▶ Proveď **ekvivalentní úpravu kolem pivotu** (i, j) , tj. nastav $a_{ij} = 1$ a $a_{i'j} = 0$ pro každé $i' \neq i$:
 - 1 Vyděl řádek i pivotem a_{ij} .
 - 2 Pro každé $i' \neq i$ odečti $a_{i'j}$ -násobek řádku i od řádku i' .

$$\begin{array}{cccccccc} & & 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ & & 1 & 0 & 4 & 0 & 1 & 0 & 4 \\ i & & 0 & 1 & -1 & 0 & -1 & -2 & -1 \\ & & & j & & & j' & & \end{array}$$

Přechod k sousední standardní bázi

Ekvivalentní řádkové úpravy:

- ▶ Řádek matice $[\mathbf{A} \ \mathbf{b}]$ vynásob nenulovým číslem.
- ▶ K řádku matice $[\mathbf{A} \ \mathbf{b}]$ přičti lineární kombinaci ostatních řádků.

Nemění množinu řešení soustavy $\mathbf{Ax} = \mathbf{b}$.

Chceme bázový sloupec $j' \in J$ nahradit nebázovým sloupcem $j \notin J$:

- ▶ Necht' i je řádek, kde $a_{ij'} = 1$. Prvek (i, j) se nazývá **pivot**.
- ▶ Proveď **ekvivalentní úpravu kolem pivotu** (i, j) , tj. nastav $a_{ij} = 1$ a $a_{i'j} = 0$ pro každé $i' \neq i$:
 - 1 Vyděl řádek i pivotem a_{ij} .
 - 2 Pro každé $i' \neq i$ odečti $a_{i'j}$ -násobek řádku i od řádku i' .

$$\begin{array}{cccccccc} & 0 & 0 & 8 & 1 & 2 & 8 & 6 \\ & 1 & 0 & 4 & 0 & 1 & 0 & 4 \\ i & 0 & 1 & -1 & 0 & -1 & -2 & -1 \\ & & j & & & j' & & \end{array}$$

Bude nové bázevé řešení přípustné?

Nechť $\mathbf{b} \geq \mathbf{0}$. Jak poznáme, zda po ekv. úpravě kolem pivotu (i, j) zůstane $\mathbf{b} \geq \mathbf{0}$?

Jak se změní \mathbf{b} po ekv. úpravě kolem pivotu (i, j) ?

- ▶ b_i se změní na b_i/a_{ij} .
- ▶ Pro každé $i' \neq i$ se $b_{i'}$ změní na $b_{i'} - a_{i'j}b_i/a_{ij}$.

Tato čísla musejí zůstat nezáporná, což nastane právě když (rozmyslete!)

- ▶ $a_{ij} > 0$
- ▶ Pro každé $i' \neq i$ platí $(a_{i'j} \leq 0)$ nebo $(b_i/a_{ij} \leq b_{i'}/a_{i'j})$.

Bude nové bázevé řešení přípustné?

Nechť $\mathbf{b} \geq \mathbf{0}$. Jak poznáme, zda po ekv. úpravě kolem pivotu (i, j) zůstane $\mathbf{b} \geq \mathbf{0}$?

Jak se změní \mathbf{b} po ekv. úpravě kolem pivotu (i, j) ?

▶ b_i se změní na b_i/a_{ij} .

▶ Pro každé $i' \neq i$ se $b_{i'}$ změní na $b_{i'} - a_{i'j}b_i/a_{ij}$.

Tato čísla musejí zůstat nezáporná, což nastane právě když (rozmyslete!)

▶ $a_{ij} > 0$

▶ Pro každé $i' \neq i$ platí $(a_{i'j} \leq 0)$ nebo $(b_i/a_{ij} \leq b_{i'}/a_{i'j})$.

0	2	6	1	0	4	4
1	1	3	0	0	2	3
0	-1	1	0	1	2	1

▶ Po úpravě kolem pivotu $(3, 2)$ **nebude** $\mathbf{b} \geq \mathbf{0}$, neboť $-1 \not\geq 0$.

Bude nové bázevé řešení přípustné?

Nechť $\mathbf{b} \geq \mathbf{0}$. Jak poznáme, zda po ekv. úpravě kolem pivotu (i, j) zůstane $\mathbf{b} \geq \mathbf{0}$?

Jak se změní \mathbf{b} po ekv. úpravě kolem pivotu (i, j) ?

- ▶ b_i se změní na b_i/a_{ij} .
- ▶ Pro každé $i' \neq i$ se $b_{i'}$ změní na $b_{i'} - a_{i'j}b_i/a_{ij}$.

Tato čísla musejí zůstat nezáporná, což nastane právě když (rozmyslete!)

- ▶ $a_{ij} > 0$
- ▶ Pro každé $i' \neq i$ platí $(a_{i'j} \leq 0)$ nebo $(b_i/a_{ij} \leq b_{i'}/a_{i'j})$.

0	2	6	1	0	4	4
1	1	3	0	0	2	3
0	-1	1	0	1	2	1

- ▶ Po úpravě kolem pivotu $(2, 2)$ **nebude** $\mathbf{b} \geq \mathbf{0}$, neboť $\frac{3}{1} \not\leq \frac{4}{2}$.

Bude nové bázevé řešení přípustné?

Nechť $\mathbf{b} \geq \mathbf{0}$. Jak poznáme, zda po ekv. úpravě kolem pivotu (i, j) zůstane $\mathbf{b} \geq \mathbf{0}$?

Jak se změní \mathbf{b} po ekv. úpravě kolem pivotu (i, j) ?

▶ b_i se změní na b_i/a_{ij} .

▶ Pro každé $i' \neq i$ se $b_{i'}$ změní na $b_{i'} - a_{i'j}b_i/a_{ij}$.

Tato čísla musejí zůstat nezáporná, což nastane právě když (rozmyslete!)

▶ $a_{ij} > 0$

▶ Pro každé $i' \neq i$ platí $(a_{i'j} \leq 0)$ nebo $(b_i/a_{ij} \leq b_{i'}/a_{i'j})$.

0	2	6	1	0	4	4
1	1	3	0	0	2	3
0	-1	1	0	1	2	1

▶ Po úpravě kolem pivotu $(3, 6)$ **bude** $\mathbf{b} \geq \mathbf{0}$, neboť $2 > 0$, $\frac{1}{2} \leq \frac{3}{2}$, $\frac{1}{2} \leq \frac{4}{4}$.

Nekladný sloupec

Když jsou všechny prvky v nebázovém sloupci $j \notin J$ nekladné:

- ▶ Sloupec j se nemůže stát bázovým (nelze v něm vybrat pivot).
- ▶ Existuje směr \mathbf{v} tak, že $\mathbf{x} + \alpha\mathbf{v} \in X$ pro každé $\alpha \geq 0$
(\Rightarrow mnohostěn obsahuje polopřímku, tedy je **neomezený**).

$$[\mathbf{A} \quad \mathbf{b}] = \begin{array}{ccccccc} 0 & -2 & 6 & 1 & 0 & 4 & 4 \\ 1 & -1 & 3 & 0 & 0 & 2 & 3 \\ 0 & -1 & 1 & 0 & 1 & 2 & 1 \end{array}$$

$$\mathbf{x} = \begin{array}{cccccc} 3 & 0 & 0 & 4 & 1 & 0 \end{array}$$

$$\mathbf{v} = \begin{array}{cccccc} 1 & 1 & 0 & 2 & 1 & 0 \end{array}$$

Vektor \mathbf{v} je řešením soustavy

$$\mathbf{A}\mathbf{v} = \mathbf{0}$$

$$v_j = 1$$

$$v_{j'} = 0 \quad \forall j' \notin J, j' \neq j$$

(tedy bázové složky \mathbf{v} jsou rovny minus prvky sloupce j).

Ekvivalentní úpravy účelového řádku

Lineární program

$$\min\{\mathbf{c}^T \mathbf{x} - d \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$$

reprezentujeme **simplexovou tabulkou**

$$\begin{bmatrix} \mathbf{c}^T & d \\ \mathbf{A} & \mathbf{b} \end{bmatrix}.$$

Přičti k prvnímu (účelovému) řádku tabulky lineární kombinaci ostatních řádků:

$$[\mathbf{c}'^T \quad d'] = [\mathbf{c}^T \quad d] + \mathbf{y}^T [\mathbf{A} \quad \mathbf{b}] = [\mathbf{c}^T + \mathbf{y}^T \mathbf{A} \quad d + \mathbf{y}^T \mathbf{b}]$$

Nová účelová funkce:

$$\begin{aligned} \mathbf{c}'^T \mathbf{x} - d' &= (\mathbf{c}^T + \mathbf{y}^T \mathbf{A})\mathbf{x} - (d + \mathbf{y}^T \mathbf{b}) \\ &= \mathbf{c}^T \mathbf{x} - d + \mathbf{y}^T (\mathbf{Ax} - \mathbf{b}) \\ &= \mathbf{c}^T \mathbf{x} - d \quad \text{jestliže } \mathbf{Ax} = \mathbf{b} \end{aligned}$$

Hodnota účelové funkce zůstane stejná pro všechna řešení soustavy $\mathbf{Ax} = \mathbf{b}$, tedy úloha se nezmění.

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
 (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 1 \quad -2 \quad -3 \quad -1 \quad 2 \quad 1 \quad 4 \\
 & & 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\
 & & 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\
 & & \\
 \mathbf{x} = & & 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0
 \end{array}$$

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
 (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 0 \quad -3 \quad -6 \quad -1 \quad 2 \quad -1 \quad 1 \\
 & & 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\
 & & 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\
 & & \mathbf{x} = \quad 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0
 \end{array}$$

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
 (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 0 \quad -1 \quad 0 \quad 0 \quad 2 \quad 3 \quad 5 \\
 & & 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\
 & & 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\
 & & \\
 \mathbf{x} = & & 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0
 \end{array}$$

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
(vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl} [\mathbf{c}^T & d] = & 0 \quad 1 \quad -2 \quad 0 \quad 0 \quad -1 \quad 3 \\ & & 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\ [\mathbf{A} & \mathbf{b}] = & 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\ & & 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\ & \mathbf{x} = & 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0 \end{array}$$

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
 (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 0 \quad 1 \quad -2 \quad 0 \quad 0 \quad -1 \quad 3 \\
 & & 0 \quad 2 \quad 6 \quad 1 \quad 0 \quad 4 \quad 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 \quad 1 \quad 3 \quad 0 \quad 0 \quad 2 \quad 3 \\
 & & 0 \quad -1 \quad 1 \quad 0 \quad 1 \quad 2 \quad 1 \\
 & & \\
 \mathbf{x} = & & 3 \quad 0 \quad 0 \quad 4 \quad 1 \quad 0
 \end{array}$$

► Protože $x_j = 0$ pro $j \notin J$, je $\mathbf{c}^T \mathbf{x} = 0$ a tedy $\mathbf{c}^T \mathbf{x} - d = -d$.

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$
 (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 0 & 1 & -2 & 0 & 0 & -1 & 3 \\
 & & 0 & 2 & 6 & 1 & 0 & 4 & 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 & 1 & 3 & 0 & 0 & 2 & 3 \\
 & & 0 & -1 & 1 & 0 & 1 & 2 & 1 \\
 \\
 \mathbf{x} = & & 3 & 0 & 0 & 4 & 1 & 0 &
 \end{array}$$

- ▶ Protože $x_j = 0$ pro $j \notin J$, je $\mathbf{c}^T \mathbf{x} = 0$ a tedy $\mathbf{c}^T \mathbf{x} - d = -d$.
 - ▶ Ihned vidíme, co by udělal vstup sloupce $j \notin J$ do báze:
 - ▶ když $c_j > 0$, účelová hodnota by stoupla
 - ▶ když $c_j < 0$, účelová hodnota by klesla
- (za předpokladu, že nové bázové řešení nebude degenerované, tj. $x_j > 0$).

Takto můžeme vynulovat bázové složky vektoru \mathbf{c} , tj. $c_j = 0$ pro $j \in J$ (vždy přičti k účelovému řádku násobek řádku, který má ve sloupci j jedničku).

$$\begin{array}{rcl}
 [\mathbf{c}^T & d] = & 0 & 1 & -2 & 0 & 0 & -1 & 3 \\
 & & 0 & 2 & 6 & 1 & 0 & 4 & 4 \\
 [\mathbf{A} & \mathbf{b}] = & 1 & 1 & 3 & 0 & 0 & 2 & 3 \\
 & & 0 & -1 & 1 & 0 & 1 & 2 & 1 \\
 \\
 \mathbf{x} = & & 3 & 0 & 0 & 4 & 1 & 0 &
 \end{array}$$

- ▶ Protože $x_j = 0$ pro $j \notin J$, je $\mathbf{c}^T \mathbf{x} = 0$ a tedy $\mathbf{c}^T \mathbf{x} - d = -d$.
- ▶ Ihned vidíme, co by udělal vstup sloupce $j \notin J$ do báze:
 - ▶ když $c_j > 0$, účelová hodnota by stoupla
 - ▶ když $c_j < 0$, účelová hodnota by klesla
 (za předpokladu, že nové bázové řešení nebude degenerované, tj. $x_j > 0$).
- ▶ Jestliže v některém sloupci j je $c_j < 0$ a $a_{ij} \leq 0$ pro všechna i , účelovou funkci můžeme libovolně zmenšovat \Rightarrow úloha je neomezená.

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ \hline 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ 0 & -1 & 1 & 0 & 1 & 2 & 1 \end{array} \right]$$

Iterace se provede takto:

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ \mathbf{A} & \mathbf{b} & & & & & \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ \hline 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ 0 & -1 & 1 & 0 & 1 & 2 & 1 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ \hline 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ 0 & -1 & 1 & 0 & 1 & 2 & 1 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

- 3 Udělej ekvivalentní úpravu okolo pivotu (i, j) a redukuj cenu c_j , tj.

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ 1 & 1 & 3 & 0 & 0 & 2 & 3 \\ 0 & -0.5 & 0.5 & 0 & 0.5 & 1 & 0.5 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

- 3 Udělej ekvivalentní úpravu okolo pivotu (i, j) a redukuj cenu c_j , tj.
 - ▶ nastav pivot na jedničku

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ 0 & 2 & 6 & 1 & 0 & 4 & 4 \\ 1 & 2 & 2 & 0 & -1 & 0 & 2 \\ 0 & -0.5 & 0.5 & 0 & 0.5 & 1 & 0.5 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

- 3 Udělej ekvivalentní úpravu okolo pivotu (i, j) a redukuj cenu c_j , tj.
 - ▶ nastav pivot na jedničku
 - ▶ vynuluj prvky nad a pod pivotem (vč. účelového řádku)

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -2 & 1 & 0 & 0 & -3 & 0 \\ \hline 0 & 4 & 4 & 1 & -2 & 0 & 2 \\ 1 & 2 & 2 & 0 & -1 & 0 & 2 \\ 0 & -0.5 & 0.5 & 0 & 0.5 & 1 & 0.5 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

- 3 Udělej ekvivalentní úpravu okolo pivotu (i, j) a redukuj cenu c_j , tj.
 - ▶ nastav pivot na jedničku
 - ▶ vynuluj prvky nad a pod pivotem (vč. účelového řádku)

Iterace simplexového algoritmu

Vstup i výstup iterace je simplexová tabulka taková, že:

- ▶ podmnožina sloupců \mathbf{A} tvoří standardní bázi
- ▶ $\mathbf{b} \geq \mathbf{0}$
- ▶ ceny c_j v bázových sloupcích jsou nulové (tj. ceny jsou redukované)

$$\begin{bmatrix} \mathbf{c}^T & d \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & -3.5 & 2.5 & 0 & 1.5 & 0 & 1.5 \\ 0 & 4 & 4 & 1 & -2 & 0 & 2 \\ 1 & 2 & 2 & 0 & -1 & 0 & 2 \\ 0 & -0.5 & 0.5 & 0 & 0.5 & 1 & 0.5 \end{array} \right]$$

Iterace se provede takto:

- 1 Vyber sloupec j pivotu tak, aby $c_j < 0$.
- 2 Vyber řádek i pivotu tak, aby nové bázové řešení bylo přípustné, tedy $(a_{ij} > 0) \wedge (\forall i' \neq i: (a_{i'j} \leq 0) \vee (b_i/a_{ij} \leq b_{i'}/a_{i'j}))$. Z toho plyne

$$i \in \operatorname{argmin}_{i' | a_{i'j} > 0} \frac{b_{i'}}{a_{i'j}}$$

- 3 Udělej ekvivalentní úpravu okolo pivotu (i, j) a redukuj cenu c_j , tj.
 - ▶ nastav pivot na jedničku
 - ▶ vynuluj prvky nad a pod pivotem (vč. účelového řádku)

Ukončení algoritmu

Algoritmus končí, když je splněna jedna z podmínek:

- ▶ Všechny ceny c_j jsou nezáporné (jsme v optimu).

0	0	7	1	0	1	4
0	1	3	0.5	0	2	2
1	0	0	-0.5	0	0	1
0	0	4	0.5	1	4	3

Ukončení algoritmu

Algoritmus končí, když je splněna jedna z podmínek:

- ▶ Všechny ceny c_j jsou nezáporné (jsme v optimu).

0	0	7	1	0	1	4
0	1	3	0.5	0	2	2
1	0	0	-0.5	0	0	1
0	0	4	0.5	1	4	3

- ▶ V každém sloupci j , ve kterém $c_j < 0$, je $a_{ij} \leq 0$ pro všechna i (úloha je neomezená).

0	0	7	-1	0	1	4
0	1	3	-0.5	0	2	2
1	0	0	-0.5	0	0	1
0	0	4	-0.5	1	4	3

Ukončení algoritmu

Algoritmus končí, když je splněna jedna z podmínek:

- ▶ Všechny ceny c_j jsou nezáporné (jsme v optimu).

0	0	7	1	0	1	4
0	1	3	0.5	0	2	2
1	0	0	-0.5	0	0	1
0	0	4	0.5	1	4	3

- ▶ V každém sloupci j , ve kterém $c_j < 0$, je $a_{ij} \leq 0$ pro všechna i (úloha je neomezená).

0	0	7	-1	0	1	4
0	1	3	-0.5	0	2	2
1	0	0	-0.5	0	0	1
0	0	4	-0.5	1	4	3

Algoritmus nemusí vždy skončit kvůli **cyklení**. Blandovo anticyklické pravidlo:

- ▶ Při výběru pivotového sloupce vždy vyber sloupec s nejnižším indexem.
- ▶ Při výběru pivotového řádku vždy vyber řádek s nejnižším indexem.

Pro zahájení simplexového algoritmu musíme vstupní úlohu převést na tvar

$$\min\{ \mathbf{c}^T \mathbf{x} \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$$

kde podmnožina sloupců \mathbf{A} tvoří standardní bázi a $\mathbf{b} \geq \mathbf{0}$.

Někdy je to snadné. Např. když má vstupní úloha tvar

$$\min\{ \mathbf{c}^T \mathbf{x} \mid \mathbf{Ax} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$$

kde $\mathbf{b} \geq \mathbf{0}$. Přidáním slackových proměnných \mathbf{u} ji převedeme na úlohu

$$\min\{ \mathbf{c}^T \mathbf{x} \mid \mathbf{Ax} + \mathbf{u} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0} \}$$

se simplexovou tabulkou

$$\begin{bmatrix} \mathbf{c}^T & \mathbf{0} & 0 \\ \mathbf{A} & \mathbf{I} & \mathbf{b} \end{bmatrix}$$

ve které sloupce příslušné proměnným \mathbf{u} tvoří standardní bázi \mathbf{I} .

Dvofázová simplexová metoda

Obecný případ: Vstupní úlohu lze vždy efektivně převést na tvar

$$\min\{ \mathbf{c}^T \mathbf{x} \mid \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$$

kde $\mathbf{b} \geq \mathbf{0}$ (ale \mathbf{A} nemusí obsahovat standardní bázi).

Vyřeš nejprve pomocnou úlohu

$$\min\{ \mathbf{1}^T \mathbf{u} \mid \mathbf{Ax} + \mathbf{u} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0} \}$$

se simplexovou tabulkou

$$\begin{bmatrix} \mathbf{1}^T & \mathbf{0} & 0 \\ \mathbf{A} & \mathbf{I} & \mathbf{b} \end{bmatrix}.$$

Platí $(\mathbf{1}^T \mathbf{u} = 0) \Leftrightarrow (\mathbf{u} = \mathbf{0}) \Leftrightarrow (\mathbf{Ax} = \mathbf{b})$. Proto:

- ▶ Vstupní úloha je **nepřípustná** \Leftrightarrow pomocná úloha má opt. hodnotu **kladnou**.
- ▶ Vstupní úloha je **přípustná** \Leftrightarrow pomocná úloha má opt. hodnotu **nulovou**.
 - ▶ Jestliže je opt. řešení (\mathbf{x}, \mathbf{u}) pomocné úlohy **nedegenerované**, pak všechny sloupce příslušné proměnným \mathbf{u} jsou nebázové, proto matice \mathbf{A} obsahuje standardní bázi.
 - ▶ Jestliže je opt. řešení (\mathbf{x}, \mathbf{u}) pomocné úlohy **degenerované**, některé sloupce příslušné proměnným \mathbf{u} mohou být bázové. Dalším pivotováním je možno bázi z těchto sloupců 'vyhnat'.