4EK445 Diskrétní modely pro informatiky

Jan Fábry

Fakulta informatiky a statistiky
Katedra ekonometrie

fabry@vse.cz
http://nb.vse.cz/~fabry

Únor 2020, Praha

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Úloha celočíselného programování

Obecná úloha lineárního programování (LP):

$$z_{\mathrm{LP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{R}^n_+\}. \tag{1}$$

Úloha celočíselného programování (IP):

$$z_{\mathrm{IP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{Z}^n_+\}. \tag{2}$$

$z_{\mathrm{LP}} \geq z_{\mathrm{IP}}$ neboť $\mathbb{Z}^n_+ \subset \mathbb{R}^n_+$
$P = \{x : Ax \leq b, x \in \mathbb{R}^n_+\}, S = \{x : Ax \leq b, x \in \mathbb{Z}^n_+\}, S \subset P$

Úloha smíšeně-celočíselného programování (MIP):

$$z_{\mathrm{MIP}} = \max\{c^T x + h^T y : Ax + Gy \leq b, x \in \mathbb{R}^n_+, y \in \mathbb{Z}^p_+\}. \tag{3}$$

Úloha bivalentního programování (BIP):

$$z_{\mathrm{BIP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{B}^n\}, \mathbb{B} = \{0, 1\}. \tag{4}$$

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Modelování základních úloh IP a MIP

1. Úloha výrobního plánování

Proměnné: x_i je počet kusů i-tého typu produktu

Omezující podmínky: $x_1, x_2, \ldots, x_n$ - celé

2. Řezná úloha

Proměnné: x_i je počet kusů původního materiálu rozděleného podle i-tého řezného schématu

Omezující podmínky: $x_1, x_2, \ldots, x_n$ - celé

Účelová funkce:

- minimalizace počtu rozřezaných kusů původního materiálu
- minimalizace odpadu
- maximalizace počtu výrobků sestavených z nařezaných dílů (maximalizace zisku)

3. (0-1) Úloha batohu

Formulace: Je dán rozpočet b pro investice do n uvažovaných projektů, kde a_j je hodnota nákladů na projekt j a c_j je jeho očekávaný výnos. Cílem je vybrat takový soubor projektů, který maximalizuje celkový očekávaný výnos a nepřekročí daný rozpočet.

Proměnné:

$$x_j = \begin{cases} 1 & \text{je-li projekt } j \text{ vybrán} \\ 0 & \text{jinak} \end{cases} \tag{5}$$

Model:

$$\max \sum_{j=1}^{n} c_j x_j \tag{6}$$

$$\sum_{j=1}^{n} a_j x_j \leq b \tag{7}$$

$$x_j \in \{0, 1\} \quad \text{pro } j = 1, 2, \ldots, n \tag{8}$$

4. Pokrytí, dělení a rozklad množiny

Formulace: Nechť $M = \{1, 2, \ldots, m\}$ je konečná množina úkolů a $N = \{1, 2, \ldots, n\}$ konečná množina firem zajišťujících tyto úkoly. Je dána matice A s hodnotami $a_{ij} = 1$, jestliže firma j je schopna řešit úkol i, $a_{ij} = 0$ jinak. Jestliže j-tá firma bude vybrána, bude si účtovat náklady c_j. Cílem je pokrýt všechny úkoly s minimálními celkovými náklady.

Nechť $M_j \subseteq M$ je množina úkolů, které je firma $j \in N$ schopna řešit.

Říkáme, že

- $F \subseteq N$ je pokrytím M, jestliže $\cup_{j \in F} M_j = M$
- $F \subseteq N$ je dělením M, jestliže $M_j \cap M_k = \emptyset$ pro všechna $j, k \in F, j \neq k$
- $F \subseteq N$ je rozkladem M, jestliže F je pokrytím a dělením M

4. Pokrytí, dělení a rozklad množiny

Proměnné:

$$x_j = \begin{cases} 1 & \text{jestliže firma } j \text{ je vybrána, tj. } j \in F \\ 0 & \text{jinak} \end{cases} \tag{9}$$

Model:

$$\min \sum_{j=1}^{n} c_j x_j \tag{10}$$

$$\begin{aligned} &\text{(pokrytí)} && \sum_{j=1}^{n} a_{ij} x_j \geq 1 && \text{pro } i = 1, 2, \ldots, m \\ &\text{(dělení)} && \sum_{j=1}^{n} a_{ij} x_j \leq 1 && \text{pro } i = 1, 2, \ldots, m \\ &\text{(rozklad)} && \sum_{j=1}^{n} a_{ij} x_j = 1 && \text{pro } i = 1, 2, \ldots, m \end{aligned} \tag{11}$$

$$x_j \in \{0, 1\} \quad \text{pro } j = 1, 2, \ldots, n \tag{12}$$

4. Pokrytí, dělení a rozklad množiny

Aplikace:

- Nechť $N = \{1, 2, \ldots, n\}$ je množina potenciálních lokalit pro umístění hasičských stanic. Náklady na umístění stanice v lokalitě j jsou c_j. Nechť $M = \{1, 2, \ldots, m\}$ je množina oblastí, které mají být zajištěny. Množina M_j obsahuje oblasti, které je možné zajistit z lokality j (např. jsou dosažitelné ze stanice v rámci 10 minut).
- Přiřazení posádek letům.
- Rozvržení pracovníků na směny.

5. Úloha optimálního rozmístění zařízení

Formulace: Je dána množina $M = \{1, 2, \dots, m\}$ lokalit pro umístění zařízení a množina zákazníků $N = \{1, 2, \dots, n\}$. Nechť a_i je kapacita zařízení v lokalitě i a b_j je velikost poptávky j-tého zákazníka. Za využití lokality i se účtují fixní náklady f_i. Přeprava jedné jednotky z lokality i k zákazníkovi j stojí c_{ij}. Cílem je rozhodnout, kam umístit zařízení a jaké množství přepravit mezi jednotlivými lokalitami a zákazníky tak, aby celkové náklady byly minimální.

Proměnné:

$$x_i = \begin{cases} 1 & \text{jestliže zařízení bude umístěno v lokalitě } i \\ 0 & \text{jinak} \end{cases} \tag{13}$$

$$y_{ij} = \text{množství přepravené z lokality } i \text{ k zákazníkovi } j \tag{14}$$

5. Úloha optimálního rozmístění zařízení

Model:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} y_{ij} + \sum_{i=1}^{m} f_i x_i \qquad (15)$$

$$\sum_{j=1}^{n} y_{ij} \leq a_i x_i \quad \text{pro } i = 1, 2, \ldots, m \qquad (16)$$

$$\sum_{i=1}^{m} y_{ij} = b_j \quad \text{pro } j = 1, 2, \ldots, n \qquad (17)$$

$$x_i \in \{0, 1\} \quad \text{pro } i = 1, 2, \ldots, m \qquad (18)$$

$$y_{ij} \in \mathbb{R}_+ \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{array} \qquad (19)$$

6. Kontejnerový dopravní problém

Formulace: Homogenní zboží je přepravováno přímo od m dodavatelů k n odběratelům. Nechť a_i je kapacita dodavatele i a b_j je požadavek odběratele j. Pro přepravu jsou použity kontejnery o kapacitě K; za přepravu jednoho kontejneru mezi dodavatelem i a odběratelem j se účtují náklady c_{ij}. Cílem je uspokojit požadavky všech odběratelů s minimálními přepravními náklady.

Proměnné:

$$y_{ij} = \text{objem přepravy od dodavatele } i \text{ k odběrateli } j \tag{20}$$

$$x_{ij} = \text{počet kontejnerů použitých pro přepravu zboží od dodavatele } i \text{ k odběrateli } j \tag{21}$$

6. Kontejnerový dopravní problém

Model:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{22}$$

$$\sum_{j=1}^{n} y_{ij} \le a_i \quad \text{pro } i = 1, 2, \ldots, m \tag{23}$$

$$\sum_{i=1}^{m} y_{ij} = b_j \quad \text{pro } j = 1, 2, \ldots, n \tag{24}$$

$$y_{ij} \le K x_{ij} \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{array} \tag{25}$$

$$y_{ij} \in \mathbb{R}_+ \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{array} \tag{26}$$

$$x_{ij} \in \mathbb{Z}_+ \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{array} \tag{27}$$

7. Rozšířená úloha batohu

Formulace 1: Je dána množina n předmětů, které mohou být umístěny do m kontejnerů. Je dána hmotnost předmětu j, kterou označíme w_j a jeho hodnota c_j. Kontejner i má nosnost K_i. Cílem je maximalizovat celkovou hodnotu všech vybraných položek.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže předmět } j \text{ je umístěn do kontejneru } i \\ 0 & \text{jinak} \end{cases} \tag{28}$$

7. Rozšířená úloha batohu

Model:

$$\max \sum_{i=1}^{m} \sum_{j=1}^{n} c_j x_{ij} \tag{29}$$

$$\sum_{i=1}^{m} x_{ij} \leq 1 \quad \text{pro } j = 1, 2, \dots, n \tag{30}$$

$$\sum_{j=1}^{n} w_j x_{ij} \leq K_i \quad \text{pro } i = 1, 2, \dots, m \tag{31}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro } \begin{array}{l} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{array} \tag{32}$$

7. Rozšířená úloha batohu

Formulace 2: Je dána množina n typů předmětů, které mají být převezeny s využitím m kontejnerů s identickou nosností K. Pro typ předmětu j je dána jeho hmotnost w_j a počet r_j, který má být převezen. Cílem je minimalizovat počet použitých kontejnerů k přepravě všech předmětů.

Proměnné:

$$x_i = \begin{cases} 1 & \text{je-li kontejner } i \text{ použitý} \\ 0 & \text{jinak} \end{cases} \tag{33}$$

$$y_{ij} = \begin{array}{l} \text{počet položek typu } j \\ \text{umístěných v kontejneru } i \end{array} \tag{34}$$

7. Rozšířená úloha batohu

Model:

$$\min \sum_{i=1}^{m} x_i \tag{35}$$

$$\sum_{i=1}^{m} y_{ij} = r_j \quad \text{pro } j = 1, 2, \ldots, n \tag{36}$$

$$\sum_{j=1}^{n} w_j y_{ij} \leq K x_i \quad \text{pro } i = 1, 2, \ldots, m \tag{37}$$

$$x_i \in \{0, 1\} \quad \text{pro } i = 1, 2, \ldots, m \tag{38}$$

$$y_{ij} \in \mathbb{Z}_+ \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{array} \tag{39}$$

8. Úloha perfektního párování

Formulace: Na školním výletě má být n (sudé číslo) studentů rozřazeno do dvoulůžkových pokojů. Pro potenciální spolubydlící i a j je dán index spokojenosti c_{ij}. Cílem je umístit studenty do pokojů tak, abychom maximalizovali celkovou spokojenost třídy.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže studenti } i \text{ a } j \text{ jsou spolubydlící} \\ 0 & \text{jinak} \end{cases} \qquad i < j \tag{40}$$

Model:

$$\max \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} c_{ij} x_{ij} \tag{41}$$

$$\sum_{j<i} x_{ji} + \sum_{j>i} x_{ij} = 1 \quad \text{pro } i = 1, 2, \ldots, n \tag{42}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro} \begin{array}{l} i = 1, 2, \ldots, n-1 \\ j = i+1, i+2, \ldots, n \end{array} \tag{43}$$

9. Lineární přiřazovací problém

Formulace: K dispozici máme n pracovníků schopných provést n úkolů. Každý pracovník má vykonávat právě jeden úkol. Realizace j-tého úkolu i-tým pracovníkem je ohodnocena náklady c_{ij}. Cílem je rozdělit mezi pracovníky všechny úkoly s minimální výší celkových nákladů.

Variables:

$$x_{ij} = \begin{cases} 1 & \text{jestliže pracovník } i \text{ vykonává úkol } j \\ 0 & \text{jinak} \end{cases} \tag{44}$$

9. Lineární přiřazovací problém

Model:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{45}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{pro } i = 1, 2, \ldots, n \tag{46}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \text{pro } j = 1, 2, \ldots, n \tag{47}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, n \end{array} \tag{48}$$

10. Úzkoprofilový přiřazovací problém

Formulace: Je dáno n úkolů a n paralelních strojů k jejich realizaci. Koeficient c_{ij} představuje dobu nutnou pro dokončení úkolu i strojem j. Cílem je minimalizovat dobu, za kterou budou úkoly dokončeny (všechny stroje začnou na úkolech pracovat současně).

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže úkol } i \text{ je přiřazen na stroj } j \\ 0 & \text{jinak} \end{cases} \tag{49}$$

$$T = \text{čas dokončení posledního úkolu} \tag{50}$$

10. Úzkoprofilový přiřazovací problém

Model:

$$\min \ T \tag{51}$$

$$c_{ij}x_{ij} \leq T \quad \text{pro} \begin{array}{l} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, n \end{array} \tag{52}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{pro } i = 1, 2, \ldots, n \tag{53}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \text{pro } j = 1, 2, \ldots, n \tag{54}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro} \begin{array}{l} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, n \end{array} \tag{55}$$

11. Kvadratický přiřazovací problém

Formulace: Množina n zařízení má být umístěna na n pracovišť (na každém pracovišti bude právě jedno zařízení). Koeficient c_{ij} představuje hodnotu toku produktů od zařízení i k zařízení j a d_{kl} je vzdálenost mezi pracovišti k a l. Cílem je rozmístit zařízení na pracoviště s minimálními náklady.

Variables:

$$x_{ik} = \begin{cases} 1 & \text{je-li zařízení } i \text{ umístěno na pracoviště } k \\ 0 & \text{jinak} \end{cases} \tag{56}$$

11. Kvadratický přiřazovací problém

Model:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} c_{ij} d_{kl} x_{ik} x_{jl} \tag{57}$$

$$\sum_{k=1}^{n} x_{ik} = 1 \quad \text{pro } i = 1, 2, \ldots, n \tag{58}$$

$$\sum_{i=1}^{n} x_{ik} = 1 \quad \text{pro } k = 1, 2, \ldots, n \tag{59}$$

$$x_{ik} \in \{0, 1\} \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ k = 1, 2, \ldots, n \end{array} \tag{60}$$

11. Kvadratický přiřazovací problém

Linearizace účelové funkce:

$$y_{ijkl} = \begin{cases} 1 & \text{jestliže zařízení } i \text{ je umístěno na pracoviště } k \\ & \text{a zařízení } j \text{ je umístěno na pracoviště } l \\ 0 & \text{jinak} \end{cases} \tag{61}$$

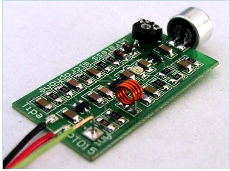
$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} c_{ij}\, d_{kl}\, y_{ijkl} \tag{62}$$

$$y_{ijkl} \geq x_{ik} + x_{jl} - 1 \quad \text{pro } i, j, k, l = 1, 2, \ldots, n \tag{63}$$

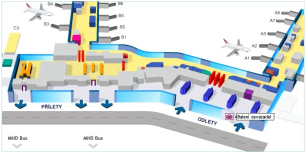
11. Kvadratický přiřazovací problém

Aplikace:

- Letovací problém



- Přiřazení letadel k odletovým branám



Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. **Úlohy na grafech**
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Úvod to teorie grafů

Graf je množina $G = \{V, E\}$, kde V je množina uzlů (vrcholů) a E množina hran.
Neorientovaná hrana je množina dvou uzlů $\{i, j\}$.
Orientovaná hrana je uspořádaná dvojice uzlů (i, j).
V neorientovaném grafu jsou všechny hrany neorientované.
V orientovaném grafu (digrafu) jsou všechny hrany orientované.
Smíšený graf obsahuje neorientované i orientované hrany.
Dva uzly spojené hranou se nazývají sousední.
Dvě hrany se společným uzlem se nazývají sousední.
Hrana a vrchol obsažený v této hraně se nazývají incidentní.
Stupeň uzlu (v neorientovaném grafu) je počet hran s ním incidentních.
Vstupní polostupeň uzlu (v orientovaném grafu) je počet incidentních hran, v nichž je tento uzel koncovým uzlem.

Úvod to teorie grafů

Výstupní polostupeň uzlu (v orientovaném grafu) je počet incidentních hran, v nichž je tento uzel počátečním uzlem.
Sled z uzlu i do uzlu j je posloupnost uzlů a hran, která začíná v uzlu i a končí v uzlu j (uzly a hrany se mohou opakovat).
Tah je sled, v němž se neopakují žádné hrany.
Cesta je tah, v němž se neopakují žádné uzly.
Cyklus je uzavřený sled (začíná a končí ve stejném uzlu).
V orientované cestě (v orientovaném grafu) je respektována orientace všech hran.
V neorientované cestě (v orientovaném grafu) nemusí být orientace hran respektována.
Neorientovaný graf je souvislý, jestliže mezi každými dvěma uzly existuje cesta.
Orientovaný graf je souvislý, jestliže mezi každými dvěma uzly existuje orientovaná nebo neorientovaná cesta.

Úlohy na grafech

Úvod to teorie grafů

Orientovaný graf je silně souvislý, jestliže mezi každými dvěma uzly existuje orientovaná cesta.
Neorientovaný graf je úplný, jestliže mezi každými dvěma uzly existuje hrana.
Strom je souvislý neorientovaný graf, v němž neexistuje cyklus.
Podgraf grafu $G = \{V, E\}$ je graf $G' = \{V', E'\}$, kde $V' \subseteq V$ a $E' \subseteq E$.
Kostra grafu G je podgraf G', kde $V' = V$, a který je stromem.
V ohodnoceném grafu jsou uzlům a/nebo hranám přiřazena čísla.
Hamiltonův cyklus v grafu je cyklus, který obsahuje každý uzel grafu právě jednou.
Eulerův cyklus v grafu obsahuje každou jeho hranu právě jednou.
Eulerův tah v grafu je tah, který obsahuje každou jeho hranu.
Eulerovský graf je graf, v němž existuje Eulerův cyklus.

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

1. Úloha hledání maximálního toku

Formulace: Nechť $G = \{V, E\}$ je digraf, v němž je pro každou hranu (i, j) definována její kapacita k_{ij}. Cílem je nalézt maximální hodnotu toku mezi zdrojem s a místem určení (stokem) d.

Proměnné:

$$x_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{64}$$

$$F = \text{hodnota celkového toku} \tag{65}$$

Model:

$$\max\ F \tag{66}$$

$$\sum_{\substack{j \in V \\ (i,j) \in E}} x_{ij} - \sum_{\substack{j \in V \\ (j,i) \in E}} x_{ji} = \begin{cases} F & \text{pro } i = s \\ 0 & \text{pro } i \in V \setminus \{s, d\} \\ -F & \text{pro } i = d \end{cases} \tag{67}$$

$$x_{ij} \leq k_{ij} \quad \text{pro } (i, j) \in E \tag{68}$$

$$x_{ij} \in \mathbb{R}_+ \quad \text{pro } (i, j) \in E \tag{69}$$

$$F \in \mathbb{R}_+ \tag{70}$$

Tokové úlohy

1. Úloha hledání maximálního toku

Alternativní model: Přidáme fiktivní orientovanou hranu ze stoku d do zdroje s, která má kapacitu $k_{ds} = M$ (vysoká konstanta).

Proměnné:

$$x_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{71}$$

Model:

$$\max\ x_{ds} \tag{72}$$

$$\sum_{\substack{j \in V \\ (i,j) \in E}} x_{ij} - \sum_{\substack{j \in V \\ (j,i) \in E}} x_{ji} = 0 \quad \text{pro } i \in V \tag{73}$$

$$x_{ij} \leq k_{ij} \quad \text{pro } (i,j) \in E \tag{74}$$

$$x_{ij} \in \mathbb{R}_+ \quad \text{pro } (i,j) \in E \tag{75}$$

Tokové úlohy

2. Úloha hledání toku s minimálními náklady

Formulace: Nechť $G = \{V, E\}$ je digraf, v němž pro každou hranu (i, j) je definována její kapacita k_{ij} a jednotkové náklady c_{ij}. Cílem je splnit požadovanou hodnotu celkového toku F_0 (ze zdroje s do stoku d) s minimálními celkovými náklady.

Proměnné:
$$x_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{76}$$

Model:
$$\min \sum_{i \in V} \sum_{\substack{j \in V \\ (i,j) \in E}} c_{ij} x_{ij} \tag{77}$$

$$\sum_{\substack{j \in V \\ (i,j) \in E}} x_{ij} - \sum_{\substack{j \in V \\ (j,i) \in E}} x_{ji} = \begin{cases} F_0 & \text{pro } i = s \\ 0 & \text{pro } i \in V \setminus \{s, d\} \\ -F_0 & \text{pro } i = d \end{cases} \tag{78}$$

$$x_{ij} \leq k_{ij} \quad \text{pro } (i, j) \in E \tag{79}$$

$$x_{ij} \in \mathbb{R}_+ \quad \text{pro } (i, j) \in E \tag{80}$$

3. Úloha hledání maximálního toku s limitovanými náklady

Formulace: Nechť $G = \{V, E\}$ je digraf, v němž pro každou hranu (i, j) je definována její kapacita k_{ij} a jednotkové náklady c_{ij}. Cílem je nalézt maximální hodnotu toku mezi zdrojem s a místem určení (stokem) d při respektování omezených celkových nákladů C_0.

Proměnné:

$$x_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{81}$$

$$F = \text{hodnota celkového toku} \tag{82}$$

Model:

$$\max\ F \tag{83}$$

$$\sum_{i \in V} \sum_{\substack{j \in V \\ (i,j) \in E}} c_{ij} x_{ij} \leq C_0 \tag{84}$$

a omezující podmínky (67) - (70)

4. Přepravní úloha (Transshipment Problem)

Formulace: Nechť $G = \{V, E\}$ je digraf se třemi množinami uzlů: množina zdrojů V_s, množina cílových uzlů V_d a množina průběžných uzlů V_t. Nechť $a_i > 0$ je velikost nabídky homogenního produktu ve zdroji $i \in V_s$ a $a_i < 0$ je velikost poptávky po produktu v cílovém uzlu $i \in V_d$. Pro průběžný uzel $i \in V_t$ platí $a_i = 0$. Pro každou hranu (i, j) je definována její kapacita k_{ij} a jednotkové náklady c_{ij}. Poptávka ve všech cílových uzlech musí být uspokojena, aniž by došlo v některém zdroji k překročení jeho nabídky. Cílem je minimalizovat celkové náklady. Předpokládejme, že velikost celkové poptávky je rovna velikosti celkové nabídky.

Předpoklady:

$$V = V_s \cup V_d \cup V_t \quad \text{a} \quad V_s \cap V_d \cap V_t = \emptyset \tag{85}$$

$$\sum_{i \in V_s} a_i + \sum_{i \in V_d} a_i = 0 \tag{86}$$

4. Přepravní úloha (Transshipment Problem)

Proměnné:

$$x_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{87}$$

Model:

$$\min \sum_{i \in V} \sum_{\substack{j \in V \\ (i,j) \in E}} c_{ij} x_{ij} \tag{88}$$

$$\sum_{\substack{j \in V \\ (i,j) \in E}} x_{ij} - \sum_{\substack{j \in V \\ (j,i) \in E}} x_{ji} = a_i \quad \text{pro } i \in V \tag{89}$$

$$x_{ij} \leq k_{ij} \quad \text{pro } (i,j) \in E \tag{90}$$

$$x_{ij} \in \mathbb{R}_+ \quad \text{pro } (i,j) \in E \tag{91}$$

5. Minimální kostra grafu

Formulace: Nechť $G = \{V, E\}$ je neorientovaný graf, v němž jsou pro každou hranu $\{i, j\}$ definovány náklady c_{ij}. Cílem je najít kostru grafu G s minimálními celkovými náklady odpovídajícími součtu nákladů vybraných hran.

Úprava grafu: Množina neorientovaných hran E je převedena na množinu orientovaných hran A takto:

Každá hrana $\{i, j\} \in E$ je nahrazena dvojicí orientovaných hran $(i, j) \in A$ a $(j, i) \in A$, $c_{ji} = c_{ij}$.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{je-li hrana } (i,j) \text{ vybrána} \\ 0 & \text{jinak} \end{cases} \tag{92}$$

$$y_{ij} = \text{hodnota toku z uzlu } i \text{ do uzlu } j \tag{93}$$

5. Minimální kostra grafu

Model:

$$\min \sum_{i \in V} \sum_{\substack{j \in V \\ (i,j) \in A}} c_{ij} x_{ij} \tag{94}$$

$$\sum_{\substack{j \in V \\ (1,j) \in A}} x_{1j} = 0 \tag{95}$$

$$\sum_{\substack{j \in V \\ (i,j) \in A}} x_{ij} = 1 \quad \text{pro } i \in V \setminus \{1\} \tag{96}$$

$$\sum_{\substack{j \in V \\ (i,j) \in A}} y_{ij} - \sum_{\substack{j \in V \\ (j,i) \in A}} y_{ji} = 1 \quad \text{pro } i \in V \setminus \{1\} \tag{97}$$

$$0 \le y_{ij} \le (|V| - 1) x_{ij} \quad \text{pro } (i,j) \in A \tag{98}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro } (i,j) \in A \tag{99}$$

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Okružní a rozvozní úlohy

Klasifikace úloh

- Typ trasy v grafu
  - Trasy jsou tvořeny uzly
    - Úloha obchodního cestujícího (TSP) - nekonečně velká kapacita vozidel
    - Rozvozní úloha (VRP) - omezená kapacita vozidel
  - Trasy jsou tvořeny hranami
    - Úloha čínského listonoše (CPP)
- Počet dep a vozidel
  - Jedno depo (výchozí místo) s jedním nebo několika vozidly
  - Více dep
- Znalost zákazníků
  - Statické úlohy - všichni zákazníci jsou známí předem
  - Dynamické úlohy - zákazníci známí předem a zákazníci s on-line požadavky
- Cíl
  - Minimalizace celkové délky tras (celkových nákladů)
  - Minimalizace celkové doby jízdy všech vozidel
  - Minimalizace nejdelší doby jízdy ze všech vozidel

1. Úloha obchodního cestujícího

Formulace: Nechť $G = \{U, E\}$ je úplný digraf se vzdáleností c_{ij} definovanou pro každou hranu (i, j) (matice C je obecně nesymetrická). Nechť uzel 1 je depo a $|U| = n$. Cílem je určit minimální Hamiltonův cyklus.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže vozidlo jede přímo} \\ & \text{z uzlu } i \text{ do uzlu } j \\ 0 & \text{jinak} \end{cases} \tag{100}$$

$$u_i = \begin{aligned} &\text{umělá proměnná zavedená v podmínkách} \\ &\text{zabraňujících vytváření parciálních cyklů} \end{aligned} \tag{101}$$

1. Úloha obchodního cestujícího

Model (Miller, Tucker, Zemlin):

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{102}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{pro } i = 1, 2, \ldots, n \tag{103}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \text{pro } j = 1, 2, \ldots, n \tag{104}$$

$$u_i + 1 - (n-1)(1 - x_{ij}) \le u_j \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 2, 3, \ldots, n \end{array} \tag{105}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, n \end{array} \tag{106}$$

$$u_i \in \mathbb{R}_+ \quad \text{pro } i = 1, 2, \ldots, n \tag{107}$$

2. Úloha obchodního cestujícího s časovými okny (TSPTW)

Formulace: Nechť je definována nesymetrická úloha TSP. Každý uzel i musí být navštíven v rámci časového intervalu $\langle e_i, l_i \rangle$. Vozidlo stráví v uzlu i daný čas S_i. Hodnota d_{ij} představuje dobu přejezdu mezi uzly i a j. Cílem je určit minimální Hamiltonův cyklus (z hlediska vzdálenosti) při respektování všech časových oken.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže vozidlo jede přímo} \\ & \text{z uzlu } i \text{ do uzlu } j \\ 0 & \text{jinak} \end{cases} \tag{108}$$

$$t_i = \text{čas příjezdu vozidla do uzlu } i \tag{109}$$

2. Úloha obchodního cestujícího s časovými okny (TSPTW)
Úprava modelu MTZ:

$$e_i \leq t_i \leq l_i \quad \text{pro } i = 2, 3, \ldots, n \tag{110}$$

$$t_1 = 0 \tag{111}$$

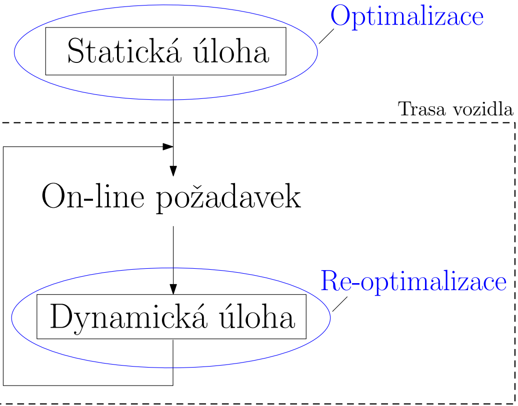
$$t_i \in \mathbb{R}_+ \quad \text{pro } i = 2, 3, \ldots, n \tag{112}$$

Proměnné u_i jsou eliminovány a omezující podmínky (105) jsou nahrazeny podmínkami

$$t_i + S_i + d_{ij} - M(1 - x_{ij}) \leq t_j \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 2, 3, \ldots, n \end{array} \; i \neq j \tag{113}$$

3. Dynamická úloha obchodního cestujícího

Ve statické verzi TSP jsou všichni zákazníci známí předem.
V dynamické verzi se během realizace optimální trasy objevují on-line požadavky dalších zákazníků.



4. Rozvozní úloha

Formulace: Nechť $G = \{U, E\}$ je úplný digraf se vzdáleností c_{ij} definovanou pro každou hranu (i, j). Nechť uzel 1 je depo, v němž je k dispozici vozidlo s kapacitou V, $|U| = n$. Každý zákazník i má požadavek o velikosti q_i. Cílem je uspokojit požadavky všech zákazníků a minimalizovat celkovou délku všech tras.

Proměnné:

$$x_{ij} = \begin{cases} 1 & \text{jestliže vozidlo jede přímo} \\ & \text{u uzlu } i \text{ do uzlu } j \\ 0 & \text{jinak} \end{cases} \tag{114}$$

$$u_i = \text{umělá proměnná zavedená pro bilanci nákladu vozidla} \tag{115}$$

Předpoklady:

$$\sum_{i=2}^{n} q_i > V \tag{116}$$

$$q_i \leq V \quad \text{pro } i = 2, 3, \ldots, n \tag{117}$$

4. Rozvozní úloha

Model:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{118}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{pro } i = 2, 3, \ldots, n \tag{119}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \text{pro } j = 2, 3, \ldots, n \tag{120}$$

$$u_i + q_j - V(1 - x_{ij}) \leq u_j \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 2, 3, \ldots, n \end{array} \tag{121}$$

$$u_i \leq V \quad \text{pro } i = 2, 3, \ldots, n \tag{122}$$

$$u_1 = 0 \tag{123}$$

$$x_{ij} \in \{0, 1\} \quad \text{pro } \begin{array}{l} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, n \end{array} \tag{124}$$

$$u_i \in \mathbb{R}_+ \quad \text{pro } i = 2, 3, \ldots, n \tag{125}$$

5. Neorientovaná úloha čínského listonoše

Formulace: Nechť $G = \{U, E\}$ je neorientovaný souvislý graf. Pro každou hranu $\{i, j\}$ jsou dány náklady c_{ij}. Cílem je najít cyklus s minimálními celkovými náklady takový, že v něm bude obsažena každá hrana alespoň jednou.

Věta: Neorientovaný graf G je eulerovský právě tehdy, když G je souvislý a všechny uzly mají sudý stupeň.

Pokud graf G není eulerovský, sestrojíme supergraf G^* grafu G takový, že G^* je eulerovský a obsahuje Eulerův cyklus, který je kratší než Eulerův cyklus v jakémkoli jiném supergrafu grafu G.

5. Neorientovaná úloha čínského listonoše

Proměnné v modelu I:

$$x_{ij} = \begin{cases} 1 & \text{jestliže hrana } \{i, j\} \text{ je zdvojena v } G^* \\ 0 & \text{jinak} \end{cases} \qquad i < j \tag{126}$$

$$y_i = \text{umělá proměnná pro vyjádření sudého/lichého čísla} \tag{127}$$

Značení v modelu I:

$U_0 \subset U$ je množina uzlů sudého stupně
$U_1 \subset U$ je množina uzlů lichého stupně

5. Neorientovaná úloha čínského listonoše

Model I:

$$\min \sum_{\substack{\{i,j\}\in E \\ i<j}} c_{ij} x_{ij} \tag{128}$$

$$\sum_{\substack{\{j,i\}\in E \\ j<i}} x_{ji} + \sum_{\substack{\{i,j\}\in E \\ j>i}} x_{ij} = 2y_i \quad \text{pro } i \in U_0 \tag{129}$$

$$\sum_{\substack{\{j,i\}\in E \\ j<i}} x_{ji} + \sum_{\substack{\{i,j\}\in E \\ j>i}} x_{ij} = 2y_i + 1 \quad \text{pro } i \in U_1 \tag{130}$$

$$x_{ij} \in \{0,1\} \quad \text{pro } \{i,j\} \in E, i<j \tag{131}$$

$$y_i \in \mathbb{Z}_+ \quad \text{pro } i \in U \tag{132}$$

5. Neorientovaná úloha čínského listonoše

Proměnné v modelu II:

$$x_{ij} = \text{počet hran } \{i,j\} \text{ v grafu } G^* \tag{133}$$

Model II:

$$\min \sum_{\{i,j\} \in E} c_{ij} x_{ij} \tag{134}$$

$$x_{ij} + x_{ji} \geq 1 \quad \text{pro } \{i,j\}, \{j,i\} \in E \tag{135}$$

$$\sum_{\{j,i\} \in E} x_{ji} = \sum_{\{i,j\} \in E} x_{ij} \quad \text{pro } i \in U \tag{136}$$

$$x_{ij} \in \mathbb{Z}_+ \quad \text{pro } \{i,j\} \in E \tag{137}$$

Okružní a rozvozní úlohy

6. „Rural“ verze úlohy čínského listonoše (doručování pošty)
Nechť $G = \{U, E\}$ je souvislý s množinou $R \subset E$ povinných hran, kterými je nutné projet alespoň jednou. Zbývající hrany v množině $E \setminus R$ jsou volitelné a mohou být využity v optimální trase.

7. Kapacitní úloha čínského listonoše (svoz odpadu)
Nechť $G = \{U, E\}$ je souvislý graf s požadavky o velikosti q_{ij} pro každou hranu $\{i, j\}$ (pro každou povinnou hranu ve verzi „Rural CPP“). Pro uspokojení požadavků je k dispozici vozidlo s omezenou kapacitou. Je nutné najít cykly, aniž by na některém z nich došlo k překročení kapacity vozidla.

8. Hierarchická úloha čínského listonoše (odklízení sněhu pluhem)
U každé hrany je dána její priorita z hlediska důležitosti příslušné komunikace.

Sylabus kurzu

Speciální omezení s logickými proměnnými

1. Nekonvexní množina přípustných řešení

Příklad: Jsou dány následující omezující podmínky. Použijte diskrétní proměnné tak, aby bylo možné úlohu s podmínkami řešit jako model MIP.

$$y_1 + y_2 \leq 40$$
$$y_1 \geq 20 \text{ nebo } y_2 \geq 10$$
$$y_1, y_2 \in \mathbb{R}_+$$

Proměnné:

$$x_1 = \begin{cases} 1 & \text{jestliže } y_1 \geq 20 \\ 0 & \text{jinak} \end{cases} \tag{138}$$

$$x_2 = \begin{cases} 1 & \text{jestliže } y_2 \geq 10 \\ 0 & \text{jinak} \end{cases} \tag{139}$$

Speciální omezení s logickými proměnnými

1. Nekonvexní množina přípustných řešení

Model:

$$y_1 + y_2 \leq 40 \tag{140}$$

$$y_1 \geq 20x_1 \tag{141}$$

$$y_2 \geq 10x_2 \tag{142}$$

$$x_1 + x_2 \geq 1 \tag{143}$$

$$y_1, y_2 \in \mathbb{R}_+ \tag{144}$$

$$x_1, x_2 \in \{0, 1\} \tag{145}$$

Speciální omezení s logickými proměnnými

2. Platnost různých soustav omezení (typ buď-nebo)

Příklad: Tři různé produkty mohou být na stroji vyráběny buď v pořadí $P_1 \rightarrow P_2 \rightarrow P_3$ nebo $P_3 \rightarrow P_2 \rightarrow P_1$. Předpokládejme, že výroba produktu P_i trvá t_i. Formulujte omezující podmínky modelující přípustnou produkci.

Proměnné:

$$y_i = \text{okamžik začátku výroby produktu } P_i \tag{146}$$

$$x = \begin{cases} 1 & \text{vyrábí-li se produkty v pořadí } P_1 \rightarrow P_2 \rightarrow P_3 \\ 0 & \text{vyrábí-li se produkty v pořadí } P_3 \rightarrow P_2 \rightarrow P_1 \end{cases} \tag{147}$$

2. Platnost různých soustav omezení (typ buď-nebo)

Model:

$$y_1 + t_1 \leq y_2 + M(1 - x) \tag{148}$$

$$y_2 + t_2 \leq y_3 + M(1 - x) \tag{149}$$

$$y_3 + t_3 \leq y_2 + Mx \tag{150}$$

$$y_2 + t_2 \leq y_1 + Mx \tag{151}$$

$$y_i \in \mathbb{R}_+ \quad \text{pro } i = 1, 2, 3 \tag{152}$$

$$x \in \{0, 1\} \tag{153}$$

Speciální omezení s logickými proměnnými

3. Platnost různých soustav omezení (splnění k z m podmínek)

Příklad: Model obsahuje soustavu m omezujících podmínek. Nechť i-tá podmínka je definována jako $a_i^T y \leq b_i$. Zajistěte, aby bylo splněno přesně k ze všech m podmínek ($k < m$).

Umělé proměnné:

$$x_i = \begin{cases} 1 & \text{je-li } i\text{-tá omezující podmínka zahrnuta v modelu} \\ 0 & \text{jinak} \end{cases} \tag{154}$$

Omezující podmínky:

$$a_i^T y \leq b_i + M(1 - x_i) \quad \text{pro } i = 1, 2, \ldots, m \tag{155}$$

$$\sum_{i=1}^{m} x_i = k \tag{156}$$

Speciální omezení s logickými proměnnými

4. Speciální omezení pro úroveň výroby

Příklad: Firma zvažuje výrobu nového produktu. V případě výroby, úroveň musí být alespoň 500 ks a nesmí překročit 1000 ks. Naformulujte odpovídající omezující podmínky.

Proměnné:

$$y = \text{úroveň výroby} \tag{157}$$

$$x = \begin{cases} 1 & \text{jestliže se firma rozhodne produkt vyrábět} \\ 0 & \text{jinak} \end{cases} \tag{158}$$

Model:

$$500x \leq y \leq 1000x \tag{159}$$

$$y \in \mathbb{Z}_+ \tag{160}$$

$$x \in \{0, 1\} \tag{161}$$

5. Plánování diskrétní úrovně výroby

Příklad: Firma zvažuje, zda vyrobit 500, 1000 nebo 2000 ks daného produktu. Naformulujte odpovídající omezující podmínky.

Proměnné:

$$y = \text{úroveň výroby} \tag{162}$$

$$x_i = \begin{cases} 1 & \text{je-li výroba na } i\text{-té úrovni} \\ 0 & \text{jinak} \end{cases} \qquad i = 1, 2, 3 \tag{163}$$

Model:

$$y = 500x_1 + 1000x_2 + 2000x_3 \tag{164}$$

$$x_1 + x_2 + x_3 = 1 \tag{165}$$

$$y \in \mathbb{Z}_+ \tag{166}$$

$$x_i \in \{0, 1\} \quad \text{pro } i = 1, 2, 3 \tag{167}$$

Speciální omezení s logickými proměnnými

6. Definice proměnné rovnající se minimu dalších proměnných

Příklad: Zapište proměnnou jako minimum n proměnných.

Proměnné:

$$w = \min(y_1, y_2, \ldots, y_n) \tag{168}$$

$$x_i = \begin{cases} 1 & \text{jestliže } w = y_i \\ 0 & \text{jinak} \end{cases} \tag{169}$$

Omezující podmínky:

$$w \leq y_i \leq w + M(1 - x_i) \quad \text{pro } i = 1, 2, \ldots, n \tag{170}$$

$$\sum_{i=1}^{n} x_i \geq 1 \tag{171}$$

7. Linearizace součinu binárních proměnných

Příklad: Linearizujte maximalizační účelovou funkci $x_1 x_2 x_3$ (všechny proměnné jsou binární).

Umělá proměnná:

$$w = x_1 x_2 x_3 \tag{172}$$

Model:

$$\max\ w \tag{173}$$

$$x_1 + x_2 + x_3 - 2 \leq w \tag{174}$$

$$w \leq x_i \quad \text{pro } i = 1, 2, 3 \tag{175}$$

$$x_i \in \{0, 1\} \quad \text{pro } i = 1, 2, 3 \tag{176}$$

$$w \in \mathbb{R}_+ \tag{177}$$

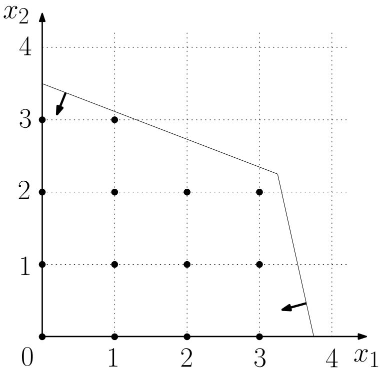
Sylabus kurzu

Vlastnosti diskrétních úloh

Množina přípustných bodů v úlohách lineárního a celočíselného programování

$$\text{(LP)} \qquad P = \{x \in \mathbb{R}^n_+ : Ax \leq b\} \tag{178}$$

$$\text{(IP)} \qquad S = \{x \in \mathbb{Z}^n_+ : Ax \leq b\} \tag{179}$$



Vlastnosti diskrétních úloh

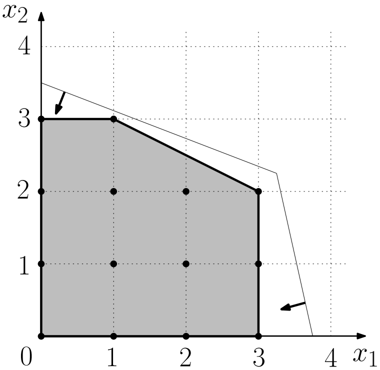
Definice: *Polyedr* $P \subseteq \mathbb{R}^n$ je množina bodů, které splňují konečný počet lineárních nerovností: $P = \{x \in \mathbb{R}^n : Ax \leq b\}$. Polyedr je *omezený*, jestliže existuje hodnota $\delta \in \mathbb{R}_+$ taková, že $P \subseteq \{x \in \mathbb{R}^n : -\delta \leq x_j \leq \delta \text{ pro } j = 1, 2, \ldots, n\}$. Omezený polyedr se nazývá *polytop*.

Definice: Nechť je dána množina $S \subseteq \mathbb{R}^n$. Bod $x \in \mathbb{R}^n$ je *konvexní kombinací* bodů z S, jestliže existuje konečná množina bodů $\{x^1, x^2, \ldots, x^t\}$ v S a vektor $\lambda \in \mathbb{R}^t_+$ takový, že $\sum_{i=1}^t \lambda_i = 1$ a $x = \sum_{i=1}^t \lambda_i x^i$.

Definice: $T \subseteq \mathbb{R}^n$ je *konvexní množina*, pokud platí: jestliže $x^1, x^2 \in T$, pak $\lambda x^1 + (1 - \lambda)x^2 \in T$ pro všechna $\lambda \in \langle 0, 1\rangle$.

Vlastnosti diskrétních úloh

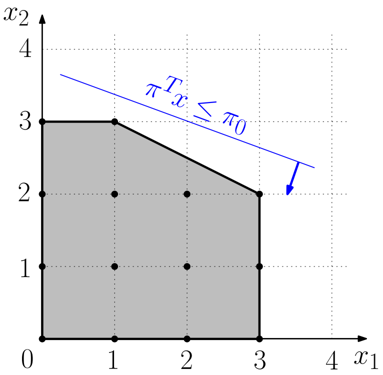
Definice: *Konvexní obal* $\mathrm{conv}(S)$ množiny S je množina všech bodů, které jsou konvexní kombinací bodů z S.



$$S \subset \mathrm{conv}(S) \subset P$$

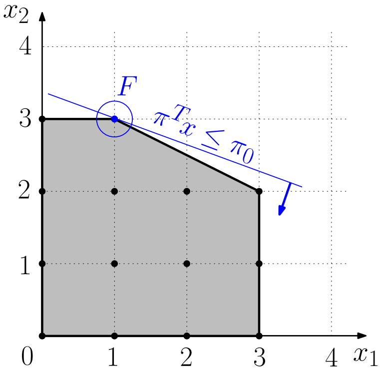
Vlastnosti diskrétních úloh

Definice: Nerovnost $\pi^T x \leq \pi_0$ se nazývá *platná nerovnost* vzhledem k S, jestliže ji splňují všechny body z S.



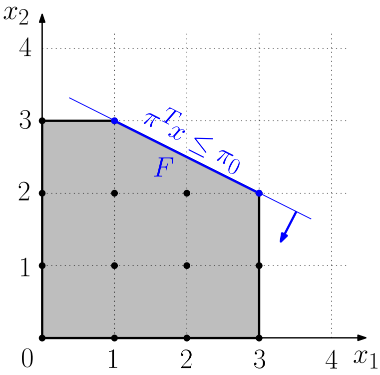
Vlastnosti diskrétních úloh

Definice: Jestliže $\pi^T x \leq \pi_0$ je platná nerovnost vzhledem k S a $\exists x^0 \in S$ takový, že $\pi^T x^0 = \pi_0$, pak se nerovnost nazývá *opěrná nerovnost* vzhledem k S. Množina $F = \{x \in \text{conv}(S) : \pi^T x = \pi_0\}$ se nazývá *opěrná stěna* množiny conv(S). Říkáme, že opěrná nerovnost $\pi^T x \leq \pi_0$ *reprezentuje* opěrnou stěnu F.



Vlastnosti diskrétních úloh

Definice: Opěrná stěna F množiny $\mathrm{conv}(S)$ se nazývá *fazeta* množiny $\mathrm{conv}(S)$, jestliže $\dim F = \dim \mathrm{conv}(S) - 1$.



Vlastnosti diskrétních úloh

Definice: Platné nerovnosti $\pi^T x \leq \pi_0$ a $\gamma^T x \leq \gamma_0$ se nazývají *ekvivalentní*, jestliže $\gamma = \lambda\pi$ a $\gamma_0 = \lambda\pi_0$ pro některé $\lambda > 0$.

Definice: Nechť $\pi^T x \leq \pi_0$ a $\gamma^T x \leq \gamma_0$ jsou dvě platné nerovnosti vzhledem k $\text{conv}(S)$, které nejsou ekvivalentní. Jestliže existuje hodnota $\lambda > 0$ taková, že $\gamma \geq \lambda\pi$ a $\gamma_0 \leq \lambda\pi_0$, pak říkáme, že $\gamma^T x \leq \gamma_0$ *dominuje (je silnější než)* $\pi^T x \leq \pi_0$. Říkáme také, že $\pi^T x \leq \pi_0$ *je dominována (je slabší než)* $\gamma^T x \leq \gamma_0$.

Pokud $\gamma^T x \leq \gamma_0$ *dominuje* $\pi^T x \leq \pi_0$, pak $\{x \in \mathbb{R}^n_+ : \gamma^T x \leq \gamma_0\} \subset \{x \in \mathbb{R}^n_+ : \pi^T x \leq \pi_0\}$.

Definice: *Maximální* platná nerovnost je taková, která není dominována žádnou jinou platnou nerovností.

Vlastnosti diskrétních úloh

Zesilování nerovností - lifting

Definice: Nechť je dána nerovnost $\sum_{j=1}^{n} \pi_j x_j \leq \pi_0$, v níž $\pi_j \in \mathbb{R}_+$ $(j = 0, 1, \ldots, n)$ a $x \in \mathbb{B}^n$. Jestliže pro nějakou hodnotu $\Delta_k > 0$ je nerovnost $\sum_{j=1}^{n} \pi_j x_j + \Delta_k x_k \leq \pi_0$ platná, pak říkáme, že je *liftovaná* z původní nerovnosti vzhledem k proměnné x_k.

Algoritmus:

Opakuj pro $k = 1, 2, \ldots, n$:

1. Nastav $x_k = 1$ a vypočti $\alpha_k = \max_{x_k \in \mathbb{B}} \sum_{j=1}^{n} \pi_j x_j \leq \pi_0$
2. Vypočti $\Delta_k = \pi_0 - \alpha_k$
3. Nahraď π_k součtem $\pi_k + \Delta_k$
4. Nerovnost $\sum_{j=1}^{n} \pi_j x_j \leq \pi_0$ je liftovaná vzhledem k proměnné x_k

Nerovnost $\sum_{j=1}^{n} \pi_j x_j \leq \pi_0$ je liftovaná vzhledem ke všem proměnným.

Vlastnosti diskrétních úloh

Zesilování nerovností - lifting

Příklad: Liftujte nerovnost $4x_1 + 5x_2 + 6x_3 + 8x_4 \leq 13$, kde $x_j \in \mathbb{B}$ $(j = 1, 2, 3, 4)$.

$x_1 = 1 \to \alpha_1 = 12 \to \Delta_1 = 1 \to \pi_1 = 5 \to 5x_1 + 5x_2 + 6x_3 + 8x_4 \leq 13$

$x_2 = 1 \to \alpha_2 = 13 \to \Delta_2 = 0 \to \pi_2 = 5 \to 5x_1 + 5x_2 + 6x_3 + 8x_4 \leq 13$

$x_3 = 1 \to \alpha_3 = 11 \to \Delta_3 = 2 \to \pi_3 = 8 \to 5x_1 + 5x_2 + 8x_3 + 8x_4 \leq 13$

$x_4 = 1 \to \alpha_4 = 13 \to \Delta_4 = 0 \to \pi_4 = 8 \to 5x_1 + 5x_2 + 8x_3 + 8x_4 \leq 13$

Vlastnosti diskrétních úloh

Zesilování nerovností - fixing

Příklad: Nechť je dána nerovnost $2x_1 + 3x_2 + 4x_3 - 15x_4 \geq 2$, v níž $x_j \in \mathbb{B}$ $(j = 1, 2, 3, 4)$.

Pokud $x_4 = 1$, nerovnost nemůže být splněna $\rightarrow$ přípustné řešení existuje jen tehdy, jestliže proměnná je zafixována $x_4 = 0$.

Příklad: Nechť je dána nerovnost $20x_1 + 5x_2 + 1x_3 - 8x_4 \geq 7$, v níž $x_j \in \mathbb{B}$ $(j = 1, 2, 3, 4)$.

Zafixování $x_1 = 1$.

Příklad: Nechť je dána rovnice $x_1 + x_2 + 3x_3 = 4$, v níž $x_j \in \mathbb{B}$ $(j = 1, 2, 3)$.

Protože $x_3 = 0$ je nepřípustné, proměnná je zafixována $x_3 = 1$.
Potom je rovnice redukována na rovnici $x_1 + x_2 = 1$.

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Sylabus kurzu

Relaxace

Definice: Nechť je dána následující úloha celočíselného programování (IP):

$$z_{\mathrm{IP}} = \max\{c^T x : x \in S \subseteq \mathbb{Z}^n_+\}. \tag{180}$$

Úloha (R)

$$z_{\mathrm{R}} = \max\{d^T x : x \in X \subseteq \mathbb{R}^n_+\} \tag{181}$$

je *relaxace* úlohy (IP), jestliže

- $S \subseteq X$
- $d^T x \geq c^T x$ pro všechna $x \in X$.

Tvrzení: Je-li úloha (R) relaxace úlohy (IP), pak $z_{\mathrm{R}} \geq z_{\mathrm{IP}}$, tj. z_{R} je horní mez pro z_{IP}.

Lineární relaxace

Definice: Nechť je dána následující úloha celočíselného programování (IP):

$$z_{\text{IP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{Z}^n_+\}. \tag{182}$$

Úloha (LP)

$$z_{\text{LP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{R}^n_+\} \tag{183}$$

je *lineární relaxace* úlohy (IP).

V případě úlohy bivalentního programování (BIP)

$$z_{\text{BIP}} = \max\{c^T x : Ax \leq b, x \in \mathbb{B}^n\}, \mathbb{B} = \{0, 1\} \tag{184}$$

je lineární relaxace definována jako

$$z_{\text{LP}} = \max\{c^T x : Ax \leq b, 0 \leq x_j \leq 1, j = 1, 2, \ldots, n\}. \tag{185}$$

Lineární relaxace

Definice: Hodnota *absolutní gap* je definována jako rozdíl

$$\mathrm{Gap} = z_{\mathrm{LP}} - z_{\mathrm{IP}} \tag{186}$$

a pro $z_{\mathrm{IP}} \neq 0$ je hodnota *relativní gap* definována jako

$$\mathrm{Gap\%} = \frac{z_{\mathrm{LP}} - z_{\mathrm{IP}}}{|z_{\mathrm{IP}}|} 100\%. \tag{187}$$

Sylabus kurzu

Metoda řezných nadrovin

Algoritmus je založen na simplexové metodě.
Řezná nadrovina (nebo jednoduše *řez*) je lineární omezující podmínka, která nevylučuje žádné přípustné celočíselné řešení.

1. Vyřeš lineární relaxaci původní úlohy.
2. Je-li optimální řešení celočíselné, jdi na krok 5.
3. Vyber proměnnou, jejíž optimální hodnota není celočíselná a sestav Gomoryho řez podle (??). Ten je přidán do simplexové tabulky jako
$$-\sum_{j=1}^{n} \pi'_j x_j + x_{n+1} = -\pi'_0, \tag{188}$$
kde x_{n+1} je přídatná proměnná.
4. Pokračuj duálním simplexovým algoritmem pro získání optimálního řešení. Jdi na krok 2.
5. Konec

Metoda větvení a mezí

Tvrzení: Je dána úloha $z_{\mathrm{IP}} = \max\{c^T x : x \in S\}$. Nechť $S = S_1 \cup S_2 \cup \ldots \cup S_K$ je dekompozice množiny S na menší množiny a nechť $z^k = \max\{c^T x : x \in S_k\}$ pro $k = 1, 2, \ldots, K$. Pak $z_{\mathrm{IP}} = \max_k z^k$.

Značení:

M je posloupnost úloh řešených ve větvích výpočetního stromu,
x^* je dosud nejlepší nalezené celočíselné řešení,
$z^* = c^T x^*$ je dosud nejlepší hodnota účelové funkce pro celočíselné řešení.

Algoritmus:

1. *Počáteční nastavení*
   $M = (LP)$, LP je lineární relaxace původní úlohy,
   x^* není definován,
   $z^* = -\infty$.

Exaktní metody

Metoda větvení a mezí

2. *Výběr řešené úlohy*
   Pokud $M = ()$, jdi na krok 5,
   jinak vyber poslední úlohu v posloupnosti M.
3. *Řešení vybrané úlohy*
   (a) Jestliže neexistuje přípustné řešení, odstraň úlohu z M a jdi na krok 2.
   (b) Je-li nalezeno optimální řešení x^0 s hodnotou účelové funkce z^0, pak
      (b1) pokud $z^0 \leq z^*$, odstraň úlohu z M a jdi na krok 2,
      (b2) pokud $z^0 > z^*$ a řešení x^0 je celočíslené, nahraď $x^* = x^0, z^* = z^0$, odstraň úlohu z M a jdi na krok 2,
      (b3) pokud $z^0 > z^*$ a řešení x^0 není celočíselné, jdi na krok 4.

Metoda větvení a mezí

4. *Větvení*

   Vyber proměnnou x_k, jejíž optimální hodnota není celočíselná. Přidej kopii poslední řešené úlohy na konec posloupnosti M spolu s omezující podmínkou

   $$x_k \leq \lfloor x_k^0 \rfloor. \tag{189}$$

   K předposlední úloze v M přidej omezující podmínku

   $$x_k \geq \lfloor x_k^0 \rfloor + 1. \tag{190}$$

   Jdi na krok 2.

5. *Konec*

   Optimální celočíselné řešení je x^*, optimální hodnota účelové funkce je z^*.

Exaktní metody

Metoda větvení a mezí

Tvrzení: Větev výpočetního stromu je uzavřena, pokud v jejím uzlu nastane jedna ze tří možností:

- úloha nemá řešení,
- optimální řešení x^0 je celočíselné,
- platí $z^0 \leq z^*$.

V případě bivalentní úlohy (v níž je n binárních proměnných), je n maximální hloubka výpočetního stromu a 2^n maximální počet jeho listů.

Sylabus kurzu

1. Úloha celočíselného programování
2. Modelování základních úloh IP a MIP
3. Úlohy na grafech
   - Tokové úlohy
   - Okružní a rozvozní úlohy
4. Speciální omezení s logickými proměnnými
5. Vlastnosti diskrétních úloh
6. Řešení úloh - metody a algoritmy
   - Relaxace
   - Exaktní metody
   - Heuristiky a metaheuristiky

Heuristiky a metaheuristiky

Přibližné metody:

- Heuristika
  Je to postup, který vede k získání dobrého řešení (blízkého optimálnímu řešení) konkrétní optimalizační úlohy.
- Metaheuristika
  Je to přístup, který lze přizpůsobit k řešení širokého okruhu úloh.

Základní principy použití přibližné metody:

- používá se k řešení úloh $\mathcal{NPC}$ a $\mathcal{NPH}$,
- nezaručuje nalezení optimálního řešení (poskytuje tzv. suboptimální řešení),
- je to polynomiální algoritmus,
- lze jej snadno upravit pro řešení konkrétní úlohy.

Heuristiky pro TSP

Klasifikace metod:

- Konstruktivní heuristiky.
- Zatřiďovací heuristiky.
- Zlepšující heuristiky.

Předpoklad: je dána matice vzdáleností mezi všemi dvojicemi uzlů (je definován úplný graf)

Metoda nejbližšího souseda

1. Vyber libovolný uzel jako výchozí pro vytvářenou trasu.
2. Najdi nejbližší uzel (dosud nevybraný) k poslednímu uzlu na trase a přidej ho na konec trasy. Pokud takový uzel neexistuje (všechny uzly již byly vybrány), pak přidej na konec trasy výchozí uzel (je vytvořen Hamiltonův cyklus) a jdi na krok 4.
3. Jdi na krok 2.
4. Konec.

Metoda výhodnostních čísel (Clarke a Wright)

1. Vypočti výhodnostní čísla

$$s_{ij} = c_{i1} + c_{1j} - c_{ij} \quad \text{pro} \begin{array}{l} i = 2, 3, \ldots, n \\ j = 2, 3, \ldots, n \end{array} \; i \neq j. \tag{191}$$

2. Vytvoř $(n-1)$ tras $(1, i, 1)$ pro $i = 2, 3, \ldots, n$ a setřiď výhodnostní čísla sestupně.

Paralelní verze

3. (Nejlepší přípustné zatřídění)
   Podle setříděných čísel postupujeme následujícím způsobem. Pro dané výhodnostní číslo s_{ij} zjistíme, zda existují dvě trasy, z nichž první obsahuje hranu $(1, j)$ a druhá hranu $(i, 1)$, které mohou být přípustným způsobem zatříděny. V takovém případě spojíme tyto dvě trasy v jednu odstraněním hran $(1, j)$, $(i, 1)$ a vložením hrany (i, j).

Heuristiky a metaheuristiky

Metoda výhodnostních čísel (Clarke a Wright)

Sekvenční verze

3. (Nastavení trasy)
   Uvažujme trasu $(1, i, \ldots, j, 1)$. Najdi první výhodnostní číslo s_{ki} nebo s_{jl} takové, že uzly k a l jsou zahrnuty v jiných trasách obsahujících hranu $(k, 1)$ nebo hranu $(1, l)$.
4. Proveď zatřídění a opakuj tento postup dokud není vytvořen Hamiltonův cyklus.

Vkládací algoritmus

1. Vyber libovolný uzel jako výchozí pro vytvářenou trasu, např. uzel 1.
2. Najdi nejvzdálenější uzel s k výchozímu uzlu a vytvoř trasu $(1, s, 1)$.
3. Do stávající trasy postupně vkládej další dosud nezahrnuté uzly (na základě minimalizace prodloužení trasy), dokud nevznikne Hamiltonův cyklus.

Metoda zatřiďování cyklů

Nechť je daný úplný graf $G = \{U, E\}$.

1. Najdi výchozí systém cyklů $\mathcal{F}$ (např. použitím metody perfektního párování; jestliže počet uzlů v U je lichý, pak jeden z cyklů bude obsahovat 3 uzly).
2. Proveď zatřídění dvou cyklů α^* a β^* s použitím následující metriky:

$$D_{\alpha^*\beta^*} = \min_{\alpha,\beta\in\mathcal{F}} D_{\alpha\beta} = \min_{\substack{i,k\in\alpha \\ j,l\in\beta}} (c_{ij} + c_{kl} - c_{ik} - c_{jl}). \tag{192}$$

   Nechť tímto zatříděním vznikne cyklus γ.
3. Odstraň cykly α^* a β^* z $\mathcal{F}$, přidej do $\mathcal{F}$ cyklus γ.
4. Jestliže γ není Hamiltonův cyklus, jdi na krok 2.

Metoda výměn (Lin a Kernighen)

Nechť je daný úplný graf $G = \{U, E\}$.

1. Předpokládejme nalezení Hamiltonova cyklu některou konstruktivní nebo zatřiďovací heuristikou.
2. Vyměň dvě nesousední hrany z trasy za jiné dvě nesousední hrany tak, aby byl opět vytvořen Hamiltonův cyklus.
3. Pokud tato výměna přináší zlepšení řešení, proveď ji.
4. Opakuj proces hledání výměn dokud je možné zlepšovat řešení. Metoda končí, jestliže již není možné řešení zlepšit.
5. Získaný Hamiltonův cyklus je lokálně optimální (2-opt) trasa.

Metaheuristiky

Značení v algoritmech:

x je přípustné řešení dané úlohy

X je množina přípustných řešení, tj. množina všech x

$N(x)$ je okolí řešení x (množina blízkých řešení)

$f(x)$ je minimalizační účelová funkce

x^* je dosud nejlepší nalezené řešení

Metoda lokálního hledání

1. Vyber výchozí řešení $x \in X$ a přiřaď $x^* = x$.
2. Definuj okolí $N(x) \subseteq X$ a ohodnoť všechna řešení.
3. Nechť x' je nejlepší řešení z $N(x)$.
   Jestliže $f(x') < f(x^*)$, pak nahraď $x^* = x'$ a $x = x'$, jinak ukonči hledání.
4. Pokud není splněno ukončovací pravidlo, jdi na krok 2.
5. Řešení x^* odpovídá lokálnímu minimu.

Metoda tabu prohledávání

TL je posloupnost zakázaných řešení (tabu seznam)
$MaxSize$ je maximálně povolený počet řešení v tabu seznamu

1. Vyber výchozí řešení $x \in X$ a přiřaď $x^* = x$.
   Nastav $TL = \{x\}$.
2. Definuj okolí $N(x) \subseteq X \setminus TL$ a ohodnoť všechna řešení.
3. Nechť x' je nejlepší řešení z $N(x)$. Nahraď $x = x'$.
   Jestliže $f(x') < f(x^*)$, pak nahraď $x^* = x'$.
4. $TL = TL \cup \{x\}$. Jestliže $|TL| > MaxSize$, pak z TL odstraň první řešení.
5. Pokud není splněno ukončovací pravidlo, jdi na krok 2.
6. Řešení x^* je nejlepší nalezené řešení.

Metoda prahové akceptace

T je hodnota prahu akceptace horších řešení
T_0 je počáteční hodnota prahu
$r \in (0, 1)$ je míra redukce prahu

1. Vyber výchozí řešení $x \in X$ a přiřaď $x^* = x$. Nastav $T = T_0$.
2. Opakuj n-krát:
   - vyber $x' \in N(x)$,
   - jestliže $f(x') - T < f(x)$, pak nahraď $x = x'$,
   - jestliže $f(x') < f(x^*)$, pak nahraď $x^* = x'$.
3. Pokud není splněno ukončovací pravidlo, proveď redukci prahu $T = rT$ a jdi na krok 2.
4. Řešení x^* je nejlepší nalezené řešení.

Genetický algoritmus

Definice:

Populace $R \subseteq X$ je konečná množina přípustných řešení.
Fitness value $f(x)$ je ohodnocení řešení $x \in R$.
Selekce je výběr určitého páru řešení (rodičů) $x, y \in R$ založený na na jejich fitness value.
Křížení je operace spojení rodičů za účelem získání jednoho nebo dvou nových řešení (*potomků*).
Mutace je náhodná změna potomka.

Cílem selekce rodičů je vybrat lepší řešení s vyšší pravděpodobností. Předpokládejme, že fitness $f(x)$ je maximalizačního typu. Pak pravděpodobnost výběru rodiče x je dána jako

$$\frac{f(x)}{\sum\limits_{\forall y \in R} f(y)} \tag{193}$$

Genetický algoritmus

Nechť rodič x je dán kódem $x_1 x_2 \ldots x_r$, rodič y kódem $y_1 y_2 \ldots y_r$ a potomek z kódem $z_1 z_2 \ldots z_r$.

- Křížení rodičů
  - Jednobodové křížení
    Potomek #1: $x_1 \ldots x_p y_{p+1} \ldots y_r$,
    Potomek #2: $y_1 \ldots y_p x_{p+1} \ldots x_r$.
  - Dvoubodové křížení
    Potomek #1: $x_1 \ldots x_p y_{p+1} \ldots y_q x_{q+1} \ldots x_r$,
    Potomek #2: $y_1 \ldots y_p x_{p+1} \ldots x_q y_{q+1} \ldots y_r$.
  - Rovnoměrné křížení
    Potomek: $z_1 z_2 \ldots z_r$, kde $z_i \in \{x_i, y_i\}$, $i = 1, 2, \ldots, r$.
- Mutace potomka
  - Jednobodová mutace
    Změněný potomek: $z_1 \ldots z_{p-1} z_p^* z_{p+1} \ldots z_r$, kde $z_p^* \neq z_p$.
  - Dvoubodová mutace
    Změněný potomek: $z_1 \ldots z_{p-1} z_q z_{p+1} \ldots z_{q-1} z_p z_{q+1} \ldots z_r$.

Mravenčí (feromonový) algoritmus

- Metoda je založena na *inteligenci mravenčích kolonií.*
- Každý mravenec se snaží najít cestu mezi mraveništěm a potravou.
- Mravenci zanechávají na cestě *feromonovou* stopu.
- Mravenci preferují cesty obsahující větší množství feromonu.
- Feromonové stopy na delších cestách se *vypařují* rychleji než na kratších cestách.
- Vypařování feromonu má velký význam, když zamezuje konvergenci k lokálnímu optimu.
- Idea algoritmu je napodobovat toto chování pomocí „simulovaných mravenců“ (umělých agentů) procházejících grafem reprezentujícím řešenou úlohu.