



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

# OPTIMALIZAČNÍ MODELY PRO STRATEGICKÉ ROZHODOVÁNÍ V ODPADOVÉM HOSPODÁŘSTVÍ

OPTIMIZATION MODELS SUPPORTING STRATEGIC DECISIONS IN WASTE MANAGEMENT

## DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. LENKA MAZLOVÁ

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. VLASTIMÍR NEVRLÝ, Ph.D.

BRNO 2023



## Zadání diplomové práce

Ústav:	Ústav matematiky
Studentka:	<b>Bc. Lenka Mazlová</b>
Studijní program:	Matematické inženýrství
Studijní obor:	bez specializace
Vedoucí práce:	<b>Ing. Vlastimír Nevrlý, Ph.D.</b>
Akademický rok:	2022/23

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

### Optimalizační modely pro strategické rozhodování v odpadovém hospodářství

#### Stručná charakteristika problematiky úkolu:

V rámci udržitelného rozvoje je kladen stále větší důraz na efektivní zpracování odpadu a využití jeho potenciálu. Cílem je zajistit takové nakládání s odpady, aby byly splněny milníky zakotvené v legislativě EU zaměřené především na omezení skládkování a podporu materiálového využití. Těchto cílů není možné dosáhnout bez vybudování adekvátní zpracovatelské infrastruktury, pro kterou je zásadní kvalitní přípravná fáze jednotlivých projektů. Projekty však nelze posuzovat izolovaně, klíčové je do všech studií vhodně implementovat konkurenční prostředí a legislativní podmínky. Studentka se zaměří na návrh optimalizační tokové úlohy v rozsahu regionálního či státního plánování. Jádrem práce je vytvoření automatizovaného výpočtového nástroje pro podporu rozhodování a rozvoje výstavby nových zařízení pro přechod na oběhové hospodářství. Hlavním rysem navrženého matematického modelu bude komplexnost zahrnující celý zpracovatelský řetězec (produkce a složení odpadu, dopravní infrastruktura, technicko–ekonomické modely a legislativní rámec), který bude modelován pomocí binárních a celočíselných proměnných speciálních typů. Očekává se implementace a testování multikomoditního modelu v programovacím jazyce Julia.

#### Cíle diplomové práce:

- Seznámení se s problematikou odpadového hospodářství.
- Prohloubení znalostí v oblasti smíšeného celočíselného programování.
- Návrh komplexního optimalizačního modelu pro multikomoditní tok v síti.
- Implementace modelu v programovacím jazyce Julia.
- Testovací výpočty pro vybrané konfigurace.

**Seznam doporučené literatury:**

WILLIAMS, H. P. Model building in mathematical programming. 5th ed. Hoboken, N.J.: John Wiley & Sons, 2013. ISBN 978-1-118-44333-0.

KLAPKA, J., J. DVORÁK a P. POPELA. Metody operačního výzkumu. Vyd. 2. Brno: VUTIUUM, 2001. ISBN 80-214-1839-7.

MD UDDIN, F. A Network Model: Minimum Cost Network Flow Problem (MCNFP) Mathematical Analysis of Minimum Cost Network Flow Problem. Neue Ausg. Saarbrücken: LAP LAMBERT Academic Publishing, 2012. ISBN 9783659104077.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2022/23

V Brně, dne

L. S.

---

doc. Mgr. Petr Vašík, Ph.D.  
ředitel ústavu

---

doc. Ing. Jiří Hlinka, Ph.D.  
děkan fakulty



## **Abstrakt**

Táto diplomová práca je zameraná na tvorbu komplexného optimalizačného modelu pre tokovú úlohu v odpadovom hospodárstve. Jej hlavným cieľom je návrh modelu úlohy zmiešaného celočíselného programovania, ktorý zahŕňa všetky aspekty od produkcie až po spracovanie odpadu, jeho implementácia v programovacom jazyku Julia a popis získaných výsledkov. Výsledný model dokáže rozvrhnúť dopravné trasy, určiť spôsob prepravy odpadu a alokovať spracovateľské a úpravné zariadenia vrátane dimenzovania ich kapacít. Tento model je následne použitý k testovacím výpočtom na reálnych dátach z Českej republiky.

## **Summary**

This diploma thesis focuses on the creation of a complex optimization model for network flow problem in waste management. The main goal is designing a model of mixed integer programming problem that includes every aspect from production to waste treatment, its implementation in the Julia programming language and the description of the obtained results. The final model can plan transport routes, determine the method of waste transportation and allocate processing and treatment facilities, including sizing their capacities. This model is then used for test calculations on real data from Czech Republic.

## **Klíčová slova**

odpadové hospodárstvo, spracovanie odpadu, toky v sieti, teória grafov, zmiešané celočíselné programovanie, programovací jazyk Julia, optimalizačný model, SOS1 premenné

## **Keywords**

waste management, waste treatment, network flow, graph theory, mixed integer programming, Julia programming language, optimization model, SOS1 variables

MAZLOVÁ, L. *Optimalizační modely pro strategické rozhodování v odpadovém hospodářství*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2023. 64 s. Vedoucí Ing. Vlastimír Nevrlý, Ph.D.



Prehlasujem, že svoju diplomovú prácu na tému *Optimalizační modely pro strategické rozhodování v odpadovém hospodářství* som spracovala samostatne pod vedením Ing. Vlastimíra Nevrlého, Ph.D. s použitím materiálov uvedených v zozname použitej literatúry.

Bc. Lenka Mazlová



Podakovanie patrí môjmu vedúcemu, Ing. Vlastimírovi Nevrlému, Ph.D., za odborné vedenie mojej diplomovej práce, užitočné rady pri jej spracovávaní a čas strávený konzultáciami. Takisto by som sa chcela poďakovať Ing. Jaroslavovi Pluskalovi za pomoc pri zostavení a implementácii modelu do programovacieho jazyku Julia. V neposlednom rade ďakujem mojej rodine za podporu počas celého štúdia a písania tejto práce.

Bc. Lenka Mazlová



# Obsah

Úvod	12
<b>1 Teória grafov</b>	<b>14</b>
1.1 Neorientované grafy	14
1.2 Orientované a ohodnotené grafy	15
1.3 Maticový zápis grafov	16
1.4 Algoritmy na hľadanie najkratšej cesty v grafe	17
<b>2 Lineárne programovanie</b>	<b>19</b>
2.1 Formulácia úlohy, základné pojmy a vlastnosti	19
2.2 Algoritmy	22
2.3 Toky v sieti	25
2.4 Zmiešané celočíselné programovanie	26
2.4.1 Typy celočíselných premenných	27
2.4.2 Algoritmy	27
2.5 Programovacie jazyky a dostupné riešiče	30
<b>3 Aplikácia matematického programovania</b>	<b>32</b>
3.1 Formulácia komplexnej úlohy	32
3.2 Označenie	32
3.3 Základný model	34
3.4 Zavedenie premenných špeciálnych typov	35
3.5 Rozšírenie modelu o reziduálne toky	36
3.6 Rozšírenie modelu o novú formu dopravy	37
<b>4 Vzorový príklad</b>	<b>39</b>
4.1 Predstavenie úlohy	39
4.2 Vstupné parametre	40
4.3 Výsledky	41
<b>5 Aplikácia modelu na reálnych dátach z Českej republiky</b>	<b>44</b>
5.1 Formulácia úlohy a vstupné dáta	44
5.2 Výpočetné problémy	45
5.3 Výsledky	46
5.3.1 Spracovateľské zariadenia	46
5.3.2 Toky odpadu	48
5.3.3 Náklady	51
<b>Záver</b>	<b>53</b>
<b>Literatúra</b>	<b>54</b>
<b>Zoznam skratiek a symbolov</b>	<b>57</b>
<b>Prílohy</b>	<b>58</b>

# Úvod

Produkcia veľkého množstva odpadu a jeho negatívny dopad na životné prostredie je narastajúcim problémom dnešnej doby. Jedným z najúčinnějších riešení tohto problému je recyklácia. Vysoká miera recyklácie je kľúčom k prechodu na obehové hospodárstvo a materiálové využitie odpadu predstavuje podstatu dlhodobu udržateľnej stratégie. Navrhovaním rôznych podporných pravidiel a regulácií sa súčasná situácia môže postupne približovať ku konceptu obehového hospodárstva, pričom prioritným opatrením je tzv. zásada 3R – opätovné využitie odpadu („reuse“), predchádzanie jeho vzniku („reduce“) a recyklácia („recycle“). Ak odpad nie je možné recyklovať, ďalším preferovaným spôsobom spracovania je jeho energetické využitie.

Strategické rozhodnutia týkajúce sa komunálneho odpadu a jeho spracovania zohrávajú dôležitú úlohu v balíčku k obehovému hospodárstvu, ktorý vznikol v rámci Európskej únie. Hlavnými cieľmi tohto balíčku je obmedziť skládkovanie komunálneho odpadu [21] a podporiť zvýšenie miery jeho separácie a recyklácie [22]. V prípade nerecyklovateľného odpadu je prioritou jeho energetické využitie. Podiel odpadu uloženého na skládkach je však vo väčšine krajín neúnosný v porovnaní s inými typmi spracovania. Zatiaľ čo sa rozvinuté krajiny v západnej Európe snažia recyklovať maximálne množstvo odpadu a využívať nerecyklovateľný odpad na výrobu energie, krajiny strednej a východnej Európy ukladajú obrovské množstvo zmesového komunálneho odpadu na skládky.

S produkciou odpadu súvisí aj využívanie obmedzených prírodných zdrojov. Fosílna palivá zabezpečujú až 85 % dnešnej globálnej spotreby energie, čo má významný vplyv na životné prostredie. Vzhľadom na neustále rastúci dopyt po energii v posledných rokoch je potrebné hľadať alternatívne zdroje energie. Vhodnou formou energie môžu byť nerecyklovateľné časti komunálneho odpadu s potenciálnym energetickým využitím, čo môže viesť k značným úsporám emisií. Súčasná kapacita zariadení využívaných k energetickému zhodnoteniu odpadu je však veľmi malá. Zmesový komunálny odpad, ktorý je vhodný na energetické využitie, ale nie je takto spracovaný, sa väčšinou ukladá na skládky. Preto je nevyhnutná výstavba nových spracovateľských zariadení.

V posledných desiatkach rokov riešilo problémy odpadového hospodárstva veľa článkov s využitím rôznych prístupov. Väčšina z nich sa však na tieto problémy zameriavala samostatne. Článok [26] sa zaoberal problémom umiestnenia prekladacích staníc, no bral do úvahy iba zber odpadu. Štúdia [23] sa zapodievala zvoznými úlohami, tj. optimálnymi trasami zberných vozidiel v mestách, chýbali jej ale dôležité prepojenia. Práca [1] predstavila viacúčelový model opisujúci umiestnenia zariadení, ktorý dokonca obsahoval spracovanie odpadu s reverzným tokom medzi zariadeniami. Tento model však nebol multikomoditný. Tieto nedostatky vyplnil článok [5], ktorý uvažoval multikomoditný model, neobsahoval ale všetky dôležité časti odpadového hospodárstva. Štúdia [13] uvažovala tak tiež materiálové a energetické využitie reziduálneho odpadu, no nebrala do úvahy náklady na zber ani prepravu odpadu.

Motiváciou pre túto diplomovú prácu bol deficit modelu, ktorý by komplexne popisoval všetky vyššie uvedené problémy. Multikomoditný model uvedený v kapitole 3 predstavuje ekonomický pohľad na zvoz a materiálové aj energetické využitie odpadu pri jeho spracovaní – zohľadňuje náklady na prevoz odpadu cestnou aj železničnou dopravou, náklady na spracovanie odpadu v existujúcich zariadeniach a náklady na prípadné vystavenie nových zariadení. Neuvažujeme ale viacúčelový model, tj. v účelovej funkcii nefiguruje



ekologické hľadisko, ktoré je v mnohých prácach vyjadrené pomocou jednotky  $\text{CO}_2\text{eq}$ , ktorá popisuje celkový dopad oxidu uhličitého a iných skleníkových plynov na globálne otepľovanie [18]. Iné kritériá je možné doplniť do vytvoreného modelu v podobe rozšírenia účelovej funkcie, prípadne ako nové obmedzenia.

Diplomová práca je štrukturovaná nasledovne. Prvé dve kapitoly tvorí matematický aparát, v ktorom sú predstavené pojmy z teórie grafov a lineárneho programovania využívané v ďalších kapitolách. V tretej kapitole je formulovaný komplexný model, ktorého využitie je v štvrtej kapitole popisované na malej úlohe. V poslednej kapitole je zostrojený model využitý pri skúmaní reálnych dát z Českej republiky. V závere sú zhrnuté postrehy získané pri tvorbe tejto práce.

# 1 Teória grafov

V tejto kapitole budú definované základné pojmy z teórie grafov, ktorá má v praxi bohaté využitie. Tieto pojmy budeme využívať v ďalších kapitolách a boli čerpané z [3], [7], [11] a [15].

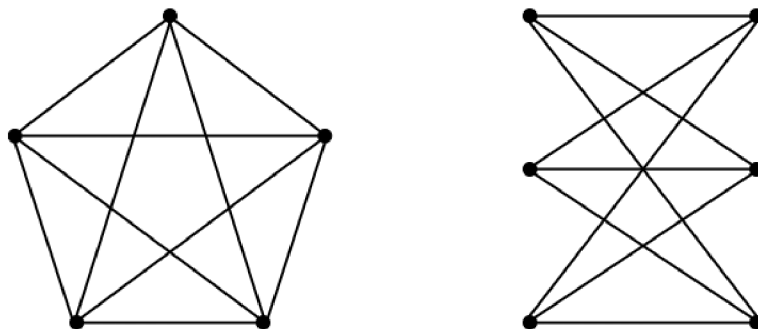
## 1.1 Neorientované grafy

**Definícia 1.1.** *Neorientovaný graf*  $G$  je dvojica  $G = (V, E)$ , kde  $V$  je neprázdna konečná množina a  $E$  je konečná množina dvojprvkových podmnožín (neusporiadaných dvojíc) množiny  $V$ . Prvky množiny  $V$  nazývame *vrcholy* alebo *uzly* a prvky množiny  $E$  nazývame *hrany*. Vrcholy  $u, v \in V$  nazývame *susedné*, ak hrana  $\{u, v\} \in E$ .

Množinu vrcholov grafu  $G$  označujeme  $V(G)$  a množinu hrán  $E(G)$ . Predstavíme si niektoré základné grafy, ktoré budeme neskôr používať.

**Definícia 1.2.** Graf  $K_n$  s  $n$  vrcholmi, ktoré sú všetky navzájom pospájané hranami, sa nazýva *úplný*.

**Definícia 1.3.** Graf  $K_{m,n}$  s  $(m + n)$  vrcholmi, ktoré môžeme rozdeliť do dvoch skupín  $M$  (s  $m$  vrcholmi) a  $N$  (s  $n$  vrcholmi) a hranami  $\{u, v\}$ , kde  $u \in M$  a  $v \in N$  nazývame *bipartitný*. Ak sú hranami navzájom pospájané všetky dvojice vrcholov  $u, v$ , kde  $u \in M$  a  $v \in N$ , tak graf  $K_{m,n}$  nazývame *úplným bipartitným grafom*.



Obr. 1.1: Úplný graf  $K_5$  (naľavo) a úplný bipartitný graf  $K_{3,3}$  (napravo)

**Definícia 1.4.** *Izomorfizmus* grafov  $G$  a  $H$  je bijektívne zobrazenie  $f : V(G) \rightarrow V(H)$ , pre ktoré platí, že dvojica vrcholov  $u, v \in V(G)$  je spojená hranou v grafe  $G$  práve vtedy, keď je dvojica  $f(u), f(v) \in V(H)$  spojená hranou v grafe  $H$ . Grafy  $G$  a  $H$  nazývame *izomorfné*, ak medzi nimi existuje izomorfizmus a píšeme  $G \simeq H$ .

**Definícia 1.5.** Graf  $H$  nazývame *podgrafom* grafu  $G$ , ak  $V(H) \subseteq V(G)$  a  $E(H) \subseteq E(G)$ , pričom každá hrana z  $E(H)$  má oba vrcholy vo  $V(H)$  a píšeme  $H \subseteq G$ .

**Definícia 1.6.** *Sled* (z vrcholu  $u$  do vrcholu  $v$ ) v grafe  $G$  je ľubovoľná postupnosť  $u = v_0, v_1, \dots, v_k = v$ , kde  $v_i$  sú vrcholy grafu  $G$  a pre každé  $i = 1, \dots, k$  je  $\{v_{i-1}, v_i\}$  hranou grafu  $G$ . *Cesta* z vrcholu  $u$  do vrcholu  $v$  v grafe  $G$  je sled, v ktorom sa každý vrchol  $v_i$  objavuje práve raz.

**Definícia 1.7.** Graf  $G$  nazývame *súvislý*, ak pre každé dva vrcholy  $u, v$  existuje v grafe  $G$  cesta z  $u$  do  $v$ . V opačnom prípade je graf *nesúvislý*.

## 1.2 Orientované a ohodnotené grafy

Existujú grafy, v ktorých má každá hrana jednoznačne určený smer. Nazývame ich orientované grafy. Smer (alebo orientácia) hrán sa pri grafickom zobrazení obvykle znázorňuje šípkami.

**Definícia 1.8.** *Orientovaný graf* alebo *digraf*  $G$  je trojica  $G = (V, E, \varepsilon)$ , kde  $V$  je neprázdna konečná množina vrcholov,  $E \subseteq V \times V$  je konečnou množinou orientovaných hrán a zobrazenie  $\varepsilon : E \rightarrow V \times V$  priradzuje každej hrane  $e \in E$  usporiadanú dvojicu vrcholov  $(u, v) \in V$ .

Usporiadanú dvojicu  $(u, v)$  tvorí *počiatočný vrchol*  $u$  a *koncový vrchol*  $v$ . Potom hovoríme, že hrana  $(u, v)$  vedie z vrcholu  $u$  do vrcholu  $v$ . Ak  $u = v$ , tak hranu nazývame *orientovanou slučkou*. Odteraz však budeme uvažovať iba grafy bez slučiek.

Z orientovaného grafu  $G$  môžeme jednoducho vyrobiť neorientovaný graf pomocou tzv. *symetrizácie* (zanedbaním orientácie hrán a nahradením násobných hrán medzi dvoma uzlami jednoduchými). V opačnom prípade hovoríme, že orientovaný graf  $G$  je *orientáciou* neorientovaného grafu  $H$ , ak je graf  $H$  symetrizáciou grafu  $G$ . Existuje veľa spôsobov ako tieto hrany orientovať. Každý neorientovaný graf má teda mnoho orientácií. Teraz si predstavíme niektoré pojmy zavedené pre neorientované grafy aj pri tých orientovaných.

**Definícia 1.9.** Hovoríme, že orientovaný graf  $G$  je (*slabo*) *súvislý*, ak je jeho symetrizácia súvislá.

**Definícia 1.10.** *Orientovaný sled* z vrcholu  $u$  do vrcholu  $v$  v orientovanom grafe  $G$  je postupnosť vrcholov  $u = v_0, v_1, \dots, v_k = v$ , v ktorej je pre každé  $i = 1, \dots, k$  usporiadaná dvojica  $(v_{i-1}, v_i)$  hranou grafu  $G$ . *Orientovaná cesta* z vrcholu  $u$  do vrcholu  $v$  v grafe  $G$  je orientovaný sled, ktorý obsahuje každý vrchol nanajvýš raz. *Cyklus* v grafe  $G$  je orientovaný sled, v ktorom je  $v_0 = v_k$ , tento vrchol je v ňom zahrnutý práve dvakrát a všetky ostatné vrcholy najviac raz.

Slabá súvislosť nehovorí o existencii orientovaných ciest v danom grafe, preto pri orientovaných grafoch často pracujeme so silnejším variantom pojmu súvislosť.

**Definícia 1.11.** Orientovaný graf  $G$  je *silno súvislý*, ak v ňom pre každú dvojicu vrcholov  $u, v$  existuje orientovaná cesta z  $u$  do  $v$  aj orientovaná cesta z  $v$  do  $u$ .

**Veta 1.12.** Slabo súvislý orientovaný graf je silno súvislý práve vtedy, keď každá jeho hrana je zahrnutá v nejakom cykle.

V praktických aplikáciách teórie grafov slúži graf ako nástroj k popisu nejakej štruktúry. Jednotlivé prvky tejto štruktúry majú často priradené hodnoty (obvykle číselné) a zadaním je realizovať určený cieľ optimálnym spôsobom. Úlohy tohto typu sa nazývajú *optimalizačné* a venovať sa im budeme v kapitole 2. Predtým si však takéto grafy musíme nadefinovať.

**Definícia 1.13.** *Ohodnotený graf* alebo *sieť*  $(G, w)$  je graf  $G$  spolu so zobrazením  $w : E(G) \rightarrow \mathbb{R}$ , ktoré sa nazýva *ohodnotenie* alebo *váha* hrán.

Niekedy je vhodné uvažovať grafy bez ohodnotenia ako špeciálne prípady ohodnotených grafov, v ktorých je váha každej hrany rovná jednej.

## 1.3 Maticový zápis grafov

Akýkoľvek graf (orientovaný aj neorientovaný) vieme popísať viacerými maticami, ktoré si teraz predstavíme. Najprv zavedieme nasledujúcu konvenciu:  $i = 1, \dots, n$  bude označovať vrcholy grafu a  $j = 1, \dots, m$  jeho hrany.

**Definícia 1.14.** Nech  $G$  je neorientovaný graf s vrcholmi  $V = \{v_1, \dots, v_n\}$  a hranami  $E = \{e_1, \dots, e_m\}$ . *Incidenčná matica*  $A(G)$  neorientovaného grafu  $G$  je reálna matica typu  $n \times m$ , definovaná vzťahom  $A(G) = (a_{ij})$ , kde

$$a_{ij} = \begin{cases} 1 & \text{ak } v_i \in e_j, \\ 0 & \text{inak.} \end{cases}$$

Podobne, ako pre neorientované grafy, môžeme zaviesť incidenčnú maticu aj pre orientované grafy.

**Definícia 1.15.** Nech  $G$  je orientovaný graf s vrcholmi  $V = \{v_1, \dots, v_n\}$  a hranami  $E = \{e_1, \dots, e_m\}$ . Potom je jeho *incidenčná matica*  $A(G)$  definovaná predpisom  $A(G) = (a_{ij})$ , kde

$$a_{ij} = \begin{cases} 1 & \text{ak } e_j \text{ vedie do vrcholu } v_i, \\ -1 & \text{ak } e_j \text{ vedie z vrcholu } v_i, \\ 0 & \text{inak.} \end{cases}$$

Jednou z dôležitých vlastností incidenčných matíc orientovaných grafov je, že v každom jej stĺpci sú práve dva nenulové prvky, z ktorých je jeden rovný 1 a druhý rovný  $-1$ . Dôvodom je samozrejme fakt, že každá hrana obsahuje práve dva vrcholy, jeden počiatočný a druhý koncový. Vlastnosť týchto matíc, ktorá je vyžadovaná pri určitých algoritmoch, sa nazýva unimodularita.

**Definícia 1.16.** Maticu  $A$  nazveme *unimodulárnou* (alebo *totálne unimodulárnou*), ak determinant akejkoľvek submatice matice  $A$  je rovný 1, 0 alebo  $-1$ .

*Poznámka.* Všetky elementy  $a_{ij}$  unimodulárnej matice  $A$  sú teda rovné 1, 0 alebo  $-1$ , keďže každý prvok je submaticou matice  $A$  rádu 1.

**Definícia 1.17.** Nech  $G$  je orientovaný graf na vrcholoch  $\{v_1, \dots, v_n\}$ . *Matica susednosti* grafu  $G$  je reálna matica typu  $n \times n$ , definovaná predpisom  $S(G) = (\sigma_{ij})$ , kde

$$\sigma_{ij} = \begin{cases} 1 & \text{ak } (v_i, v_j) \in E(G), \\ 0 & \text{inak.} \end{cases}$$

Matica susednosti neorientovaného grafu je definovaná ako matica susednosti jeho symetrickej orientácie. *Symetrická orientácia* je orientovaný graf, ktorý vznikne z neorientovaného nahradením každej hrany dvojicou protichodných orientovaných hrán.

**Definícia 1.18.** *Vážená matica susednosti* ohodnoteného orientovaného grafu  $(G, w)$  s vrcholmi  $\{v_1, \dots, v_n\}$  je matica  $W(G) = (w_{ij})$ , kde

$$w_{ij} = \begin{cases} w((v_i, v_j)) & \text{ak } (v_i, v_j) \in E(G), \\ 0 & \text{inak.} \end{cases}$$

**Definícia 1.19.** *Vzdialenosť*  $d(u, v)$  vrcholov  $u, v$  orientovaného grafu  $G$  je definovaná ako cesta s najmenším počtom hrán spájajúcich tieto dva vrcholy. Ak taká cesta neexistuje, položíme  $d(u, v) = \infty$ .

Pre neorientované grafy môžeme vzdialenosť definovať prostredníctvom symetrickej orientácie a platí nasledujúca veta.

**Veta 1.20.** Nech  $G$  je súvislý neorientovaný graf. Potom funkcia  $d(u, v)$  je *metrikou* na množine  $V(G)$ , tj. splňa nasledujúce axiomy:

1. axióm nezápornosti:  $d(u, v) \geq 0$ ,
2. axióm totožnosti:  $d(u, v) = 0 \Leftrightarrow u = v$ ,
3. axióm symetrie:  $d(u, v) = d(v, u)$ ,
4. trojuholníková nerovnosť:  $d(u, v) + d(v, w) \geq d(u, w)$ .

**Definícia 1.21.** *Matica vzdialeností* orientovaného grafu  $G$  s vrcholmi  $\{v_1, \dots, v_n\}$  je matica  $D(G)$  typu  $n \times n$  daná predpisom

$$D(G) = (d(v_i, v_j))_{i,j=1}^n.$$

**Definícia 1.22.** Nech  $M$  je množina hrán ohodnoteného orientovaného grafu  $(G, w)$ . *Váha*  $w(M)$  množiny  $M$  je súčet váh jednotlivých hrán  $e \in M$ . *Váha*  $w(P)$  cesty  $P$  je váha jej množiny hrán. Ak sú  $u, v$  vrcholmi grafu  $G$ , tak *najkratšia cesta* z  $u$  do  $v$  je každá cesta, ktorej váha je minimálna (tj. žiadna iná cesta z  $u$  do  $v$  nemá menšiu váhu). *Vážená vzdialenosť*  $d^w(u, v)$  vrcholov  $u, v$  je váha najkratšej cesty z  $u$  do  $v$ .

**Definícia 1.23.** *Matica vážených vzdialeností* ohodnoteného orientovaného grafu  $(G, w)$  s vrcholmi  $\{v_1, \dots, v_n\}$  je matica  $D^w(G)$  typu  $n \times n$  daná predpisom

$$D^w(G) = (d^w(v_i, v_j))_{i,j=1}^n.$$

## 1.4 Algoritmy na hľadanie najkratšej cesty v grafe

Medzi najznámejšie algoritmy hľadania vázenej vzdialenosti vrcholov (a teda aj najkratšej cesty) v ohodnotenom grafe patria Dijkstrov a Floyd-Warshallov algoritmus. Oba algoritmy spolu s ďalšími, sú dobre popísané v [11] a [17], odkiaľ sme čerpali.

Pre výpočet najkratšej cesty v grafe s nezápornými dĺžkami hrán sa využíva Dijkstrov algoritmus. Tento algoritmus dokáže nájsť najkratšiu cestu len medzi dvomi konkrétnymi vrcholmi, čo môže byť vo veľmi rozsiahlych grafoch pomalé, keďže by sme ho museli aplikovať na všetky dvojice vrcholov v grafe. Sústredíme sa teda na rýchlejšie a účinnejšie algoritmy.

Jedným z nich je obojsmerný Dijkstrov algoritmus, ktorý hľadá cestu v priamom smere z počiatočného vrcholu do koncového a aj v opačnom smere z koncového vrcholu do počiatočného. Z hľadiska priemerného počtu krokov je približne dvakrát rýchlejší ako obyčajný Dijkstrov algoritmus a na nájdenie najkratšej cesty by mal prehľadať len polovičný počet hrán. Zároveň môžeme tento algoritmus paralelizovať do dvoch samostatných procesov.

## 1.4 ALGORITMY NA HLADANIE NAJKRATŠEJ CESTY V GRAFE

Bližšie si popíšeme Floyd-Warshallov algoritmus, ktorý na rozdiel od vyššie spomenutého Dijkstrovho algoritmu dokáže nájsť najkratšiu cestu z každého vrcholu grafu do každého. Okrem toho dokáže pracovať aj s grafmi so zápornými dĺžkami hrán a zápornými cyklami, ktoré dokáže odhaliť. Výstupom z tohto algoritmu sú napokon dve matice, matica vážených vzdialeností  $D^n$  a matica predkov  $P^n$ .

Uvažujme ohodnotený orientovaný graf  $G = (V, E)$  s  $n$  vrcholmi. Zavedieme dve matice  $D^0$  a  $P^0$  typu  $n \times n$ , ktoré budeme postupne transformovať. Matica  $D^0$  je váženou maticou susednosti, tj. obsahuje ohodnotenie hrán grafu, kde namiesto 0 píšeme  $\infty$ . V matici  $P^0$  sú všetky prvky  $p_{ij}$  rovné  $j$ .

Floyd-Warshallov algoritmus má vždy  $n$  krokov. Východiskovými maticami sú  $D^0$  a  $P^0$ , matice  $D^k$  a  $P^k$  sú v  $k$ -tom kroku získané z matíc  $D^{k-1}$  a  $P^{k-1}$ . V  $n$ -tom kroku je matica  $D^n$  maticou vážených vzdialeností a prvky  $p_{ij}^n$  matice  $P^n$  sú rovné vrcholu, ktorý v najkratšej ceste z vrcholu  $i$  do vrcholu  $j$  nasleduje bezprostredne po vrchole  $i$ . Keďže poznáme prvky  $d_{ij}^{k-1}$  a  $p_{ij}^{k-1}$  matíc  $D^{k-1}$  a  $P^{k-1}$ , tak môžeme prvky  $d_{ij}^k$  a  $p_{ij}^k$  matíc  $D^k$  a  $P^k$  počítat nasledovne:

1. ak  $d_{ik}^{k-1} + d_{kj}^{k-1} < d_{ij}^{k-1}$ , tak  $d_{ij}^k = d_{ik}^{k-1} + d_{kj}^{k-1}$  a  $p_{ij}^k = p_{ik}^{k-1}$ ,
2. inak  $d_{ij}^k = d_{ij}^{k-1}$  a  $p_{ij}^k = p_{ij}^{k-1}$ .

Ilustračný príklad a viac o Floyd-Warshallovom algoritme môžeme nájsť v [11].



## 2 Lineárne programovanie

Táto kapitola sa bude zaoberať lineárnym programovaním, jeho podmnožinou – zmiešaným celočíselným lineárnym programovaním a spôsobmi riešenia takýchto úloh. Teória k lineárnemu programovaniu a jeho algoritmom bola prebraná z [6], [14] a [20].

### 2.1 Formulácia úlohy, základné pojmy a vlastnosti

Úloha lineárneho programovania je optimalizačný problém, v ktorom je účelová funkcia  $z$  (lineárna voči premenným) minimalizovaná alebo maximalizovaná na základe konečného počtu lineárnych obmedzení (rovníc a nerovníc). Aby takýto problém mohol byť klasifikovaný ako úloha lineárneho programovania, musí obsahovať aspoň jednu lineárnu nerovnosť.

Pod *úlohou lineárneho programovania v maticovom zápise* rozumieme

$$\max\{z = \mathbf{c}\mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}, \quad (2.1)$$

kde  $\mathbf{A}$  je matica typu  $m \times n$ ,  $\mathbf{c}$  je  $n$ -rozmerný riadkový vektor,  $\mathbf{b}$  je  $m$ -rozmerný stĺpcový vektor a  $\mathbf{x}$  je  $n$ -rozmerný stĺpcový vektor reálnych premenných. V tejto práci však budeme prevažne používať *sumačný zápis* v tvare

$$\begin{aligned} \max z &= \sum_{j=1}^n c_j x_j, \\ \sum_{j=1}^n a_{ij} x_j &\leq b_i, \quad \forall i = 1, \dots, m, \\ x_j &\geq 0, \quad \forall j = 1, \dots, n. \end{aligned} \quad (2.2)$$

Vektor  $\bar{\mathbf{x}}$ , ktorý spĺňa všetky definované obmedzenia, nazveme *prípustným riešením*. Pri maximalizačnej úlohe nazveme prípustné riešenie  $\bar{\mathbf{x}}$  *optimálnym*, ak pre všetky prípustné riešenia  $\mathbf{x}$  platí

$$\mathbf{c}\bar{\mathbf{x}} \geq \mathbf{c}\mathbf{x}. \quad (2.3)$$

Úloha lineárneho programovania môže, ale nemusí mať prípustné riešenie. *Množina* všetkých *prípustných riešení* môže byť:

- prázdna – neexistuje prípustné riešenie,
- neprázdna – existuje prípustné riešenie,
  - ohraničená – optimálna hodnota účelovej funkcie je vždy konečná,
  - neohraničená – optimálna hodnota účelovej funkcie je konečná alebo nekonečná.

Všetky obmedzenia úlohy lineárneho programovania je možné interpretovať aj geometricky. Množina riešení lineárnej rovnice sa nazýva *nadrovina*. Množina riešení lineárnej nerovnice sa nazýva *polpriestor*. Akákoľvek podmnožina  $\mathbb{R}^n$ , ktorú môžeme znázorňovať ako prienik konečnej množiny polpriestorov sa nazýva *mnohosten*. Z toho vyplýva, že množina všetkých prípustných riešení úlohy lineárneho programovania je mnohosten.

## 2.1 FORMULÁCIA ÚLOHY, ZÁKLADNÉ POJMY A VLASTNOSTI

Ďalšou dôležitou vlastnosťou tejto množiny je konvexnosť. Hovoríme, že množina  $X$  je *konvexná*, ak pre ľubovoľné dva body  $x_1, x_2 \in X$  a pre ľubovoľné  $\lambda \in [0, 1]$  platí

$$\lambda x_1 + (1 - \lambda)x_2 \in X. \quad (2.4)$$

Inými slovami, množina sa nazýva konvexná, ak úsečka spájajúca ľubovoľné dva body tejto množiny je v danej množine obsiahnutá. Ich základnou vlastnosťou je, že prienikom dvoch konvexných množín vznikne znovu konvexná množina. Z toho vyplýva, že množina všetkých prípustných riešení je naozaj konvexná.

Teraz opíšeme vzťah medzi geometrickým pojmom krajného bodu množiny prípustných riešení a algebraickým pojmom základného prípustného riešenia úlohy lineárneho programovania. Uvažujme úlohu lineárneho programovania v *standardnom tvare*

$$\begin{aligned} \max z &= \sum_{j=1}^n c_j x_j, \\ \sum_{j=1}^n a_{ij} x_j &= b_i, \quad \forall i = 1, \dots, m, \\ x_j &\geq 0, \quad \forall j = 1, \dots, n. \end{aligned} \quad (2.5)$$

Každé obmedzenie v tvare

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (2.6)$$

je možné previesť na lineárnu rovnicu pridaním tzv. *slack premennej*, ktorá je nezáporná, na jej ľavú stranu. Pre každú nerovnosť je dôležité použiť samostatnú slack premennú. Z nerovnosti (2.6) sa teda stane dvojica lineárnych obmedzení - lineárna rovnica a podmienka nezápornosti:

$$\sum_{j=1}^n a_{ij} x_j + x_{n+i} = b_i \quad \text{a} \quad x_{n+i} \geq 0, \quad (2.7)$$

kde  $x_{n+i}$  značí spomínanú slack premennú. Uvažujme, že matica  $\mathbf{A}$  má plnú hodnotu (tj. riadky matice  $\mathbf{A}$  sú lineárne nezávislé). Ak by matica  $\mathbf{A}$  nemala plnú hodnotu, tak by obmedzenia boli buď nekonzistentné (úloha by nemala riešenie) alebo redundantné (a mohli by sme ich odstrániť).

*Krajný bod* je definovaný geometricky pomocou konvexnosti. Bod  $x \in X$  je krajným bodom alebo vrcholom konvexnej množiny  $X$ , ak ho nemôžeme vyjadriť v tvare

$$x = \lambda x_1 + (1 - \lambda)x_2, \quad (2.8)$$

kde  $x_1, x_2 \in X$ ,  $x_1, x_2 \neq x$  a  $\lambda \in (0, 1)$ . Definícia krajného bodu platí pre akúkoľvek konvexnú množinu a v našom prípade najmä pre množinu prípustných riešení úlohy lineárneho programovania. Hovoríme, že dva krajné body sú *susedné*, ak sú spojené hranou množiny prípustných riešení.

*Základné riešenie* je definované algebraicky s využitím štandardného tvaru úlohy (2.5). Vektor  $\mathbf{x}$  je základným riešením, ak:

- splňa obmedzenia úlohy (2.5),



- stĺpce matice  $\mathbf{A}$  s indexami zodpovedajúcimi nenulovým zložkám vektoru  $\mathbf{x}$  sú lineárne nezávislé.

Keďže matica  $\mathbf{A}$  má plnú hodnosť, je možné rozdeliť zložky vektoru  $\mathbf{x}$  do dvoch vektorov  $\mathbf{x}_N$  a  $\mathbf{x}_B$ . Vektor  $\mathbf{x}_N$  pozostáva z  $n - m$  nebázových premenných, ktoré sú rovné nule. Vektor  $\mathbf{x}_B$  pozostáva z  $m$  bázových premenných. Množina bázových premenných sa nazýva *báza*. Bod  $\mathbf{x}$  je *základným prípustným riešením*, ak navyše spĺňa podmienku nezápornosti  $\mathbf{x} \geq 0$ . Dôkaz, že pojem krajného bodu je ekvivalentný s pojmom základného prípustného riešenia je uvedený v [14]. Je možné, že jedna alebo viac bázových premenných v základnom prípustnom riešení bude rovná nule. Vtedy bod nazveme degenerovaným vrcholom a úlohu lineárneho programovania prehlásime za *degenerovanú*. Degenerácia môže nastať, ak úloha obsahuje redundantné obmedzenia a preto takéto obmedzenia odstránime.

Pre každý problém lineárneho programovania existuje sprievodný problém, ktorý nazývame duálny. V takomto probléme sú úlohy premenných a obmedzení vymenené. To znamená, že pre každú premennú v pôvodnom alebo *primárnom probléme* existuje obmedzenie v *duálnom probléme* a pre každé obmedzenie v primárnom probléme zas premenná v duálnom probléme.

Aj keď je možné definovať duálny problém k akejkoľvek forme primárneho problému, symetria týchto dvoch problémov je najzreteľnejšia, keď je pôvodný problém v *kanonickej forme*. Minimalizačný problém je v kanonickej forme, ak sú všetky obmedzenia vo forme nerovnosti typu „ $\geq$ “ a ak sú všetky premenné nezáporné:

$$\begin{aligned} \min z &= \mathbf{c}\mathbf{x}, \\ \mathbf{A}\mathbf{x} &\geq \mathbf{b}, \\ \mathbf{x} &\geq \mathbf{0}. \end{aligned} \tag{2.9}$$

Duálny problém k primárnemu problému (2.9) má tvar

$$\begin{aligned} \max w &= \mathbf{b}^T \mathbf{y}, \\ \mathbf{A}^T \mathbf{y} &\leq \mathbf{c}^T, \\ \mathbf{y} &\geq \mathbf{0}. \end{aligned} \tag{2.10}$$

Duálny problém je teda maximalizačný problém v kanonickej forme, kde sú všetky obmedzenia vo forme nerovnosti typu „ $\leq$ “ a všetky premenné nezáporné. Súhrnná tabuľka, ktorá pomáha s prevodom primárnej úlohy na jej duálny tvar, prevzatá z [6] je uvedená v prílohe A.

Dualita sa využíva napr. na vývoj efektívnych metód na riešenie úloh lineárneho programovania. V súčasnosti sa software využívajúci metódy vnútorných bodov, predstavené v nasledujúcej podkapitole, spolieha práve na kombináciu informácií z primárneho a duálneho problému. S týmito metódami súvisí aj pojem *optimality (duality) gap*, ktorý popisuje vzdialenosť medzi optimálnym riešením primárneho a duálneho problému. Túto vzdialenosť môžeme rozdeliť na relatívnu, ktorá popisuje rozdiel medzi hodnotami účelových funkcií primárneho a duálneho problému percentuálne a absolútnu, ktorá tento rozdiel popisuje priamo. Optimality gap môže byť prakticky využitá ako stop kritérium v optimalizačných algoritmoch. Ak jej hodnota klesne pod vopred definovanú prahovú hodnotu, výpočet sa ukončí a získané riešenie považujeme za takmer optimálne.

## 2.2 Algoritmy

Všetky algoritmy, ktoré budeme ďalej preberať, majú istú výpočtovú zložitosť. Výpočtovú zložitosť daného algoritmu budeme merať pomocou funkcie  $f(n)$ , ktorá udáva časovú náročnosť výpočtu (počet krokov algoritmu) ako funkciu veľkosti vstupných dát. Algoritmus budeme považovať za *efektívny*, ak môžeme funkciu  $f(n)$  zhora odhadnúť polynómom  $p(n)$ . Ak takýto odhad neexistuje, budeme algoritmus považovať za *neefektívny*.

Problémy riešiteľné polynomiálnymi (efektívnymi) algoritmami nazývame *problémy triedy  $\mathcal{P}$* . Algoritmy, ktoré riešia takéto úlohy, pracujú deterministicky, tj. v každom kroku je jednoznačne určený nasledujúci krok.

Problémy riešiteľné nedeterministicky v polynomiálnom čase nazývame *problémy triedy  $\mathcal{NP}$*  (nedeterministicky polynomiálne). Pre tieto úlohy zatiaľ nebol nájdený efektívny algoritmus, ale zároveň nie je dokázané, že neexistuje. Pre úlohy triedy  $\mathcal{NP}$  je charakteristické, že ak sa nám podarí nájsť riešenie, tak je možné v polynomiálnom čase overiť jeho správnosť. K nájdeniu takéhoto riešenia sa používajú nedeterministické algoritmy, ktoré pracujú s náhodným generovaním. Do triedy  $\mathcal{NP}$  úloh patria aj tzv.  *$\mathcal{NP}$ -úplné úlohy*, ktoré sú považované za najobtiažnejšie. Jedná sa o také problémy, na ktoré môžeme ľubovoľný problém z triedy  $\mathcal{NP}$  polynomiálne previesť.

Platí, že  $\mathcal{P} \subset \mathcal{NP}$ , no nie je doposiaľ známe, či sa tieto dve triedy rovnajú alebo nie. Problém efektívneho riešenia  $\mathcal{NP}$ -úplných úloh, tj. otázka, či platí  $\mathcal{P} = \mathcal{NP}$  alebo  $\mathcal{P} \neq \mathcal{NP}$ , patrí medzi najdôležitejšie otvorené problémy modernej matematiky a aj cez veľké úsilie zostáva nevyriešený. Predpokladá sa ale, že tieto triedy si rovné nie sú.

Rozdelenie do týchto tried však platí len pre rozhodovacie úlohy, tj. úlohy s riešením „áno“ (1) alebo „nie“ (0). Nami riešené optimalizačné úlohy je preto nutné previesť na ekvivalentnú rozhodovaciu úlohu.

### Simplexová metóda

Jedným z najvyužívanejších algoritmov na riešenie úloh lineárneho programovania je *simplexová metóda*. Simplexová metóda je iteratívna metóda, ktorá pracuje s úlohami v štandardnom tvare. Pri aplikácii na nedegenerovanú úlohu prechádza po hranách množiny prípustných riešení z jedného základného prípustného riešenia (krajného bodu) na ďalšie, až kým nenájde optimálne riešenie.

Postup riešenia pomocou simplexovej metódy si ukážeme na minimalizačnej úlohe, keďže každá maximalizačná úloha môže byť prevedená na minimalizačnú a to nasledovne:

$$\max z(x) = - \min -z(x). \quad (2.11)$$

Uvažujme úlohu lineárneho programovania v štandardnom tvare s maticovým zápisom

$$\min\{z = \mathbf{c}\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}. \quad (2.12)$$

Nech  $\mathbf{x}$  je základné prípustné riešenie s premennými usporiadanými tak, že

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix}, \quad (2.13)$$

kde  $\mathbf{x}_B$  je vektor bázových premenných a  $\mathbf{x}_N$  je vektor nebázových premenných rovných nule. Účelovú funkciu môžeme zapísať v tvare

$$z = \mathbf{c}_B\mathbf{x}_B + \mathbf{c}_N\mathbf{x}_N, \quad (2.14)$$

kde vektor  $\mathbf{c}$  rozdelíme na vektor  $\mathbf{c}_B$  koeficientov zodpovedajúcich bázovým premenným  $\mathbf{x}_B$  a vektor  $\mathbf{c}_N$  koeficientov zodpovedajúcich nebázovým premenným  $\mathbf{x}_N$ . Obdobne môžeme zapísať obmedzenia ako

$$\mathbf{B}\mathbf{x}_B + \mathbf{N}\mathbf{x}_N = \mathbf{b}, \quad (2.15)$$

kde  $\mathbf{A} = (\mathbf{B} \ \mathbf{N})$ . Z toho dostávame

$$\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N. \quad (2.16)$$

Zmenou hodnôt nebázových premenných môžeme obdržať všetky možné riešenia  $\mathbf{A}\mathbf{x} = \mathbf{b}$ . Po dosadení do  $z$  dostávame

$$z = \mathbf{c}_B\mathbf{B}^{-1}\mathbf{b} + (\mathbf{c}_N - \mathbf{c}_B\mathbf{B}^{-1}\mathbf{N})\mathbf{x}_N. \quad (2.17)$$

Ak označíme  $\mathbf{y} = \mathbf{c}_B\mathbf{B}^{-1}$ , tak môžeme písať

$$z = \mathbf{y}\mathbf{b} + (\mathbf{c}_N - \mathbf{y}\mathbf{N})\mathbf{x}_N. \quad (2.18)$$

Vektor  $\mathbf{y}$  nazývame *vektorom simplexových multiplikátorov*. Aktuálne hodnoty bázových premenných a účelovej funkcie získame, ak položíme  $\mathbf{x}_N = 0$  a označujeme ich

$$\mathbf{x}_b = \hat{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b} \quad \text{a} \quad \hat{z} = \mathbf{c}_B\mathbf{B}^{-1}\mathbf{b}. \quad (2.19)$$

Teraz môžeme načrtnúť myšlienku simplexovej metódy. Metóda začína s bázovou maticou  $\mathbf{B}$ , ktorá zodpovedá základnému prípustnému riešeniu  $\mathbf{x}_B = \hat{\mathbf{b}} = \mathbf{B}^{-1}\mathbf{b} \geq 0$ . Postup je nasledujúci:

1. *Test optimality*

Vypočítame vektor  $\mathbf{y} = \mathbf{c}_B\mathbf{B}^{-1}$ . Vypočítame koeficienty  $\hat{\mathbf{c}}_N = \mathbf{c}_N - \mathbf{y}\mathbf{N}$ . Ak  $\hat{\mathbf{c}}_N \geq 0$ , tak je aktuálna báza optimálna. V opačnom prípade zvolíme za vstupnú premennú  $\mathbf{x}_t$  také, že  $\hat{\mathbf{c}}_t < 0$ .

2. *Krok*

Vypočítame  $\hat{\mathbf{A}}_t = \mathbf{B}^{-1}\mathbf{A}_t$  ( $\mathbf{A}_t$  je  $t$ -ty stĺpec matice  $\mathbf{A}$ ) – koeficienty obmedzenia prislúchajúce vstupnej premennej. Nájďme index  $s$  taký, že:

$$\frac{\hat{b}_s}{\hat{a}_{st}} = \min_{1 \leq i \leq m} \left\{ \frac{\hat{b}_i}{\hat{a}_{it}} : \hat{a}_{it} \geq 0 \right\}. \quad (2.20)$$

Tento predpis určuje výstupnú premennú a tzv. pivotovací vstup  $\hat{a}_{st}$ . Ak  $\hat{a}_{it} \leq 0$  pre všetky  $i$ , tak je úloha neohraničená.

3. *Aktualizácia*

Aktualizujeme bázovú maticu  $\mathbf{B}$  a vektor bázových premenných  $\mathbf{x}_B$  a postup opakujeme, dokým nedosiahneme optimálnu bázu.

V 2. kroku simplexovej metódy existuje možnosť, že množina prípustných riešení je neohraničená. Tento a mnoho ďalších prípadov, ktoré sa nedostali do tejto práce, je uvedených a vysvetlených v [14].

## Metódy vnútorných bodov

*Metódy vnútorných bodov* sú pravdepodobne najvýznamnejším posunom v lineárnej optimalizácii od vývoja simplexovej metódy. Tieto metódy majú dobrú teoretickú účinnosť a praktický výkon a preto konkurujú simplexovej metóde. Spoločným znakom všetkých metód vnútorných bodov je, že ich iterácie sú *striktne prípustné* (striktne prípustný bod množiny  $\{\mathbf{x} : \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$  je definovaný ako bod  $\mathbf{x}$ , ktorý spĺňa  $\mathbf{Ax} = \mathbf{b}$  a  $\mathbf{x} > \mathbf{0}$ ). To znamená, že na rozdiel od simplexovej metódy, kde sa pohybujeme pozdĺž hranice množiny prípustných riešení, sú body generované pomocou týchto metód vo vnútri tejto množiny. Z tohto dôvodu sú tieto metódy známe ako metódy vnútorných bodov a bolo zistené, že sú špeciálnou formou *bariérových metód* – triedy algoritmov pre nelineárnu optimalizáciu.

Väčšina metód vnútorných bodov pre úlohy lineárneho programovania spadá do jednej alebo viacerých z troch hlavných kategórií: *metódy sledovania cesty*, *metódy znižovania potenciálu* a *metódy afínného škálovania*. Metódy sledovania cesty sa pokúšajú ostať blízko tzv. centrálnej trajektórie, definovanej pomocou funkcie logaritmickéj bariéry.

Zameriame sa na metódu, ktorá bola úspešne implementovaná pri riešení rozsiahlych úloh lineárneho programovania a to *primárno-duálnu metódu sledovania cesty*. Uvažujme úlohu lineárneho programovania v štandardnom tvare (primárnu úlohu) s maticovým zápisom

$$\begin{aligned} \min z &= \mathbf{cx}, \\ \mathbf{Ax} &= \mathbf{b}, \\ \mathbf{x} &\geq \mathbf{0}. \end{aligned} \tag{2.21}$$

Predpokladáme, že matica  $\mathbf{A}$  má plnú hodnosť. Tento predpoklad nie je nevyhnutný, ale zjednodušuje ďalší postup. Ak označíme vektor duálnych slack premenných písmenom  $\mathbf{s}$ , tak môžeme príslušný duálny problém zapísať v tvare

$$\begin{aligned} \max w &= \mathbf{b}^T \mathbf{y}, \\ \mathbf{A}^T \mathbf{y} + \mathbf{s} &= \mathbf{c}^T, \\ \mathbf{s} &\geq \mathbf{0}. \end{aligned} \tag{2.22}$$

Nech  $\bar{\mathbf{x}}$  je prípustným riešením (2.21) a nech  $(\bar{\mathbf{y}}, \bar{\mathbf{s}})$  je prípustným riešením (2.22). Tieto body sú optimálnym riešením (2.21) a (2.22) práve vtedy, keď spĺňajú komplementárne podmienky

$$x_j s_j = 0, \quad \forall j = 1, \dots, n. \tag{2.23}$$

Hlavnou myšlienkou primárno-duálnej metódy je prechádzať cez sériu striktne prípustných primárnych a duálnych riešení, ktoré sa čoraz viac približujú k splneniu komplementárnych podmienok (2.23). Pri každej iterácii sa teda pokúšame nájsť vektory  $\mathbf{x}(\mu)$ ,  $\mathbf{y}(\mu)$  a  $\mathbf{s}(\mu)$ , ktoré pre nejaké  $\mu > 0$  spĺňajú

$$\begin{aligned} \mathbf{Ax} &= \mathbf{b}, \\ \mathbf{A}^T \mathbf{y} + \mathbf{s} &= \mathbf{c}^T, \\ x_j s_j &= \mu, \quad \forall j = 1, \dots, n \\ \mathbf{x}, \mathbf{s} &\geq \mathbf{0}. \end{aligned} \tag{2.24}$$

Hodnota parametra  $\mu$  sa následne zníži a celý proces sa opakuje, kým sa nedosiahne konvergencia. Ak  $\mu > 0$ , tak podmienka  $x_j s_j = \mu$  zaručuje, že  $\mathbf{x} > \mathbf{0}$  a  $\mathbf{s} > \mathbf{0}$  a teda, že iterácie sú striktne prípustné. Tieto podmienky tiež obmedzujú duality gap, keďže

$$\mathbf{c}\mathbf{x} - \mathbf{b}^T \mathbf{y} = \mathbf{x}^T \mathbf{s} = n\mu. \quad (2.25)$$

Algoritmus sa teda pokúša nájsť postupnosť primárnych a duálnych prípustných riešení so znižujúcou sa duality gap. Ak by duality gap bola nulová, tak by nájdené riešenie bolo optimálne.

Pri tomto algoritme je najobťažnejšie vyriešiť rovnice (2.24). Aby sme ušetrili výpočty, nebudeme ich riešiť presne, ale iba približne. Pri každej iterácii teda budeme mať len odhad riešenia (2.24). To, ako získať daný odhad a aké modifikácie tohto algoritmu sa využívajú v praxi je možné nájsť v [14].

## 2.3 Toky v sieti

Dôležitou triedou úloh lineárneho programovania s veľkým využitím sú *toky v sieti*. Poznatky o tokoch v sieti boli čerpané z [25].

Uvažujme digraf  $G = (V, E)$  s kapacitami hrán  $h_{ij}$  pre všetky  $(i, j) \in E$ , požiadavkou  $b_i$  (pozitívne prítoky alebo negatívne odtoky) v každom uzle  $i \in V$  a jednotkovými tokovými nákladmi  $c_{ij}$  pre každé  $(i, j) \in E$ . Úlohou tzv. *Minimum Cost Network Flow problému* je nájsť prípustný tok, ktorý splní všetky požiadavky pri minimálnych nákladoch. Formulácia problému je nasledujúca:

$$\begin{aligned} \min z &= \sum_{(i,j) \in E} c_{ij} x_{ij}, \\ \sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} &= b_i, & \forall i \in V, \\ 0 \leq x_{ij} &\leq h_{ij}, & \forall (i, j) \in E, \end{aligned} \quad (2.26)$$

kde  $x_{ij}$  značí tok na hrane  $(i, j)$ ,  $V^+(i) = \{k \in V : (i, k) \in E\}$  je množina vrcholov, do ktorých vedie hrana z  $i$  a  $V^-(i) = \{k \in V : (k, i) \in E\}$  je množina vrcholov z ktorých vedie hrana do  $i$ . Je zrejmé, že na to, aby bol problém prípustný, musí byť celkový súčet všetkých požiadaviek nulový. Špeciálnymi prípadmi tejto úlohy sú nasledujúce dva problémy.

### Problém maximálneho toku

Uvažujme digraf  $G = (V, E)$ , dva odlišné uzly  $s, t \in V$  a nezáporné kapacity  $h_{ij}$  pre všetky  $(i, j) \in E$ . Úlohou *problému maximálneho toku* je nájsť cestu medzi uzlami  $s$  a  $t$  s maximálnym tokom. Po pridaní spätnej hrany z uzla  $t$  do uzla  $s$  môžeme tento problém formulovať ako

$$\begin{aligned} \max z &= x_{ts}, \\ \sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} &= 0, & \forall i \in V, \\ 0 \leq x_{ij} &\leq h_{ij}, & \forall (i, j) \in E. \end{aligned} \quad (2.27)$$

## Problém najkratšej cesty

Uvažujme digraf  $G = (V, E)$ , dva odlišné uzly  $s, t \in V$  a nezáporné náklady  $c_{ij}$  pre všetky  $(i, j) \in E$ . Úlohou *problému najkratšej cesty* je nájsť cestu medzi uzlami  $s$  a  $t$  s minimálnymi nákladmi. Tento problém môže byť formulovaný ako

$$\begin{aligned}
 \min z &= \sum_{(i,j) \in E} c_{ij} x_{ij}, \\
 \sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} &= 1, & i = s, \\
 \sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} &= 0, & \forall i \in V \setminus \{s, t\}, \\
 \sum_{k \in V^+(i)} x_{ik} - \sum_{k \in V^-(i)} x_{ki} &= -1, & i = t, \\
 x_{ij} &\geq 0, & \forall (i, j) \in E, \\
 x &\in \mathbb{Z}^{|E|},
 \end{aligned} \tag{2.28}$$

kde  $x_{ij} = 1$ , ak je hrana  $(i, j)$  súčasťou cesty s minimálnymi nákladmi (najkratšej cesty).

## 2.4 Zmiešané celočíselné programovanie

Pomocou zmiešaného celočíselného programovania je možné formulovať a riešiť širokú škálu praktických optimalizačných problémov. Medzi takéto problémy patrí napríklad rozpis vlakov alebo plánovanie výroby. Pri spracovávaní tejto podkapitoly sme využili [19], [24] a [25].

Predpokladajme, že máme úlohu lineárneho programovania (2.2). Ak pridáme podmienku, že určité premenné musia nadobúdať celočíselné hodnoty, dostávame *úlohu zmiešaného celočíselného programovania* s účelovou funkciou

$$\max z = \sum_{j=1}^n c_j x_j + \sum_{k=1}^p h_k y_k \tag{2.29}$$

a obmedzeniami

$$\begin{aligned}
 \sum_{j=1}^n a_{ij} x_j + \sum_{k=1}^p g_{ik} y_k &\leq b_i, & \forall i = 1, \dots, m, \\
 x_j &\geq 0, & \forall j = 1, \dots, n, \\
 y_k &\in \mathbb{Z}_0^+, & \forall k = 1, \dots, p,
 \end{aligned} \tag{2.30}$$

kde  $\mathbf{G} \ni g_{ik}$  je matica typu  $m \times p$ ,  $\mathbf{h}$  je  $p$ -rozmerný riadkový vektor a  $\mathbf{y}$  je  $p$ -rozmerný stĺpcový vektor celočíselných premenných. Ak všetky premenné musia nadobúdať celočíselné hodnoty, tak dostávame *úlohu celočíselného programovania* a ak všetky premenné musia nadobúdať iba hodnoty 0 a 1, tak dostávame *úlohu binárneho celočíselného programovania*.

Ďalším typom úlohy je tzv. *problém kombinatorickej optimalizácie*. Obvykle je daná konečná množina  $N = \{1, \dots, n\}$ , váhy  $c_j$  pre každé  $j \in N$  a množina  $\mathcal{F}$  všetkých



prípustných podmnožín množiny  $N$ . Našou úlohou je teda nájsť prípustnú podmnožinu  $S$  s minimálnou váhou

$$\min_{S \subseteq N} z = \sum_{j \in S} c_j, \quad (2.31)$$

$$S \in \mathcal{F}.$$

Problémy kombinatorickej optimalizácie sú veľmi často využívané a môžu byť formulované ako úlohy celočíselného alebo binárneho celočíselného programovania.

### 2.4.1 Typy celočíselných premenných

V úlohách zmiešaného celočíselného programovania sa môžeme stretnúť s viacerými typmi premenných. Medzi ne patria celočíselné, binárne a SOS premenné.

*Celočíselné premenné* sa využívajú na vyjadrenie množstva, ktoré môže byť len v celých číslach, ako napríklad vozy, zariadenia alebo ľudia.

Rozhodovacie premenné sa často používajú na označenie toho, ktoré z množstva možných rozhodnutí by sa malo urobiť. Zvyčajne môžu nadobúdať iba dve hodnoty, 0 a 1, preto ich voláme *binárne premenné*. Jedným z príkladov je rozhodovanie o výstavbe zariadení, kde  $\delta = 1$  značí, že by sa zariadenie malo postaviť a  $\delta = 0$  zas, že by sa postaviť nemalo. Binárne premenné obyčajne označujeme pomocou gréckeho písmena  $\delta$  a latinské písmená ponechávame ostatným premenným.

*SOS (Special Ordered Set) premenné* sa delia na dva typy – SOS typu 1 a SOS typu 2. SOS1 je usporiadaná množina premenných (spojitých alebo celočíselných), z ktorých je nanajvyš jedna nenulová. Tieto premenné využívame napríklad pri výbere kapacity zariadenia v danom meste, kedy chceme vystavať zariadenie s jedinou kapacitou vhodnou pre množstvo obyvateľov v danej obci. V binárnom prípade môžeme SOS1 premenné  $\delta_1, \dots, \delta_n$  zapísať pomocou dvoch obmedzení:

$$\sum_{i=1}^n \delta_i \leq 1, \quad (2.32)$$

$$\delta_i \in \{0, 1\}, \quad \forall i \in \{1, \dots, n\}.$$

SOS2 je usporiadaná množina premenných, z ktorých sú najviac dve nenulové a ak sú nenulové práve dve, tak musia nasledovať bezprostredne za sebou.

### 2.4.2 Algoritmy

Na rozdiel od lineárneho programovania so simplexovou metódou, pre celočíselné programovanie neexistuje univerzálny algoritmus vhodný pre všetky typy úloh. Rôzne algoritmy sa osvedčujú pri rôznych typoch problémov. Môžeme ich rozdeliť do dvoch hlavných kategórií – exaktné a heuristické metódy. *Exaktné metódy* zaručujú nájdenie optimálneho riešenia, no doba behu programu sa dramaticky zvyšuje s veľkosťou problému a často sa dajú prakticky vyriešiť len malé alebo stredne veľké problémy. Preto sa pri väčších problémoch využívajú *heuristické metódy*, ktoré nájdu dostatočne dobré riešenia v rozumnom čase.

### Exaktné metódy

Jednou skupinou využívaných exaktných algoritmov sú *metódy sečných nadrovín*. Zvyčajne riešia problém celočíselného programovania ako problém lineárneho programovania s vynechaním požiadavky na celočíselnosť. Ak je výsledné riešenie celočíselné, tak je aj celočíselným optimom. V opačnom prípade sa k problému pridávajú ďalšie obmedzenia až pokiaľ nenájde celočíselné riešenie, ktoré však nemusí existovať. Tieto metódy sa môžu zdať matematicky pomerne elegantné, no pri veľkých problémoch sa veľmi neosvedčili.

Zatiaľ najpoužívanejšími exaktnými algoritmi na riešenie praktických celočíselných problémov sú *metódy vetiev a hraníc* (branch and bound). Rovnako ako pri metódach sečných nadrovín riešime daný problém ako problém lineárneho programovania tzv. relaxáciou požiadavky na celočíselnosť. Ak je výsledné riešenie celé číslo, problém je vyriešený. Ak nie, vykonávame stromové prehľadávanie. Uvažujme problém

$$z = \max \sum_{j=1}^n c_j x_j, \quad (2.33)$$

kde  $\mathbf{x} \in S$ . Množinu  $S$  môžeme rozložiť na podmnožiny  $S = S_1 \cup \dots \cup S_K$  a zaviesť  $z^k = \max \sum_{j=1}^n c_j x_j$ , kde  $\mathbf{x} \in S_k$  pre  $k = 1, \dots, K$ . Potom  $z = \max_k z^k$ . Zavedieme aj pojmy horná hranica  $z^k$  (ozn.  $\bar{z}^k$ ) a dolná hranica  $z^k$  (ozn.  $\underline{z}^k$ ). Potom horná hranica  $z$  je  $\bar{z} = \max_k \bar{z}^k$  a dolná hranica  $z$  je  $\underline{z} = \max_k \underline{z}^k$ . Metóda vetiev a hraníc rozdeľuje problém na sériu menších, jednoduchších podproblémov, ktoré sú následne spojené, aby vyriešili pôvodný problém. Takýto prístup sa nazýva „*rozdeľ a panuj*“ a jeho typickou reprezentáciou je tzv. *enumeráčny strom*. Existujú minimálne tri spôsoby ako tento strom orezať a tým zmenšiť priestor možných riešení a tak zlepšiť účinnosť metódy:

1. orezanie podľa optimality:  $\bar{z}^t = \underline{z}^t$ ,
2. orezanie podľa hranice:  $\bar{z}^t \leq \underline{z}$ ,
3. orezanie podľa neprípustnosti:  $S_t = \emptyset$ .

Detailnejší opis tohto algoritmu je uvedený v [25].

### Heuristické metódy

Vzhľadom na to, že veľa praktických problémov patrí do triedy zložitosti  $\mathcal{NP}$ , musíme pri riešení diskretných optimalizačných problémov použiť heuristické alebo aproximačné algoritmy. Dôvodov, prečo zvoliť práve heuristiky je niekoľko:

- riešenie je získané rýchlo (v priebehu pár sekúnd alebo minút),
- riešená úloha je tak veľká alebo komplikovaná, že sa nedá formulovať ako problém celočíselného programovania alebo zmiešaného celočíselného programovania rozumnej veľkosti,
- aj keď je úloha formulovaná ako problém zmiešaného celočíselného programovania, tak je veľmi náročné alebo nemožné nájsť (dobré) prípustné riešenie pomocou metódy vetiev a hraníc,



- pre určité problémy kombinatorickej optimalizácie je jednoduché nájsť prípustné riešenia preskúmaním alebo znalosťou štruktúry problému.

Najprv si predstavíme *hladný algoritmus*. Uvažujme, že skúmanú úlohu môžeme zapísať vo forme problému kombinatorickej optimalizácie:

$$\min_{S \subseteq N} \{c(S) : v(S) \geq k\}. \quad (2.34)$$

Potom je postup nasledovný:

1. Nech  $S^0 = \emptyset$  a  $t = 1$ .
2. Nech  $j_t = \arg \min \frac{c(S^{t-1} \cup \{j_t\}) - c(S^{t-1})}{v(S^{t-1} \cup \{j_t\}) - v(S^{t-1})}$ .
3. Ak je predchádzajúce riešenie  $S^{t-1}$  prípustné a účelová funkcia sa neznížila, zastavíme algoritmus pri  $S^G = S^{t-1}$ .
4. Inak, nech  $S^t = S^{t-1} \cup \{j_t\}$ . Ak už je riešenie prípustné a účelová funkcia je neklesajúca alebo  $t = n$ , zastavíme algoritmus pri  $S^G = S^t$ .
5. Inak, ak  $t = n$ , tak nebolo nájsť žiadne prípustné riešenie a algoritmus zastavíme.
6. Inak, nech  $t = t + 1$  a vrátime sa do kroku 2.

Teraz si predstavíme *algoritmus lokálneho vyhľadávania*. Na popísanie tohto algoritmu je jednoduchšie formulovať problém kombinatorickej optimalizácie ako

$$\min_{S \subseteq N} \{c(S) : g(S) = 0\}, \quad (2.35)$$

kde  $g(S) \geq 0$  predstavuje mieru neprípustnosti množiny  $S$ . S využitím obmedzenia  $v(S) \geq k$  predstaveného vyššie môžeme túto mieru vyjadriť ako  $g(S) = (k - v(S))^+$ . Pre algoritmus lokálneho vyhľadávania musíme definovať riešenie, lokálne okolie  $Q(S)$  pre každé riešenie  $S \subseteq N$  a cieľovú funkciu  $f(S)$  ktorá sa buď môže rovnať  $c(S)$ , ak je  $S$  prípustné a inak nekonečnu, alebo zloženej funkcii v tvare  $c(S) + \alpha g(S)$ , ktorá sa skladá z váženej kombinácie hodnoty účelovej funkcie a kladného násobku  $\alpha$  miery neprípustnosti množiny  $S$ . Postupujeme nasledovne:

1. Vyberieme počiatočné riešenie  $S$ .
2. Hľadáme množinu  $S' \in Q(S)$  takú, že  $f(S') < f(S)$ .
3. Ak žiadna neexistuje, tak algoritmus zastavíme.  $S^H = S$  je lokálne optimálne riešenie.
4. V opačnom prípade nech  $S = S'$  a celý postup opakujeme.

Vhodná voľba lokálneho okolia závisí od štruktúry problému. Jediným problémom tohto algoritmu je, že dosiahneme len lokálne minimum a je potenciálne možné nájsť aj lepšie riešenie. Na to slúži *tabu vyhľadávanie* alebo *algoritmus simulovaného žihania*, ktoré sú popísané napr. v [25].

### Kombinácia exaktných a heuristických metód

V poslednej dobe je snaha spojiť exaktné a heuristické metódy, čo môže zaistiť optimálne riešenie problémov v kratšom čase alebo zlepšenie samotných heuristických metód. Dve hlavné kategórie do ktorých delíme tieto prístupy sú *kolaborujúce* a *integračné* kombinácie algoritmov. Pod kolaboráciou rozumieme, že algoritmy si vymieňajú informácie, no nie sú súčasťou jeden druhého. Algoritmy teda môžu byť vykonávané následne, paralelne alebo môžu byť prepletené. Pod integráciou rozumieme, že jeden algoritmus je obsiahnutý v druhom a je mu podriadený. Jednou z integračnej kombinácie algoritmov je využitie heuristiky na získanie počiatočného prípustného riešenia a hraníc pri metóde vetiev a hraníc. Tieto a ďalšie kombinácie algoritmov sú uvedené v [19].

## 2.5 Programovacie jazyky a dostupné riešiče

Pri tvorení tejto práce bol použitý programovací jazyk Julia a komerčný riešič pre lineárne programovanie CPLEX. V tejto podkapitole budú uvedené dôvody výberu a porovnanie s ostatnými jazykmi a riešičmi. Informácie boli získané z [2], [4], [12] a [16].

Programovací jazyk Julia je flexibilný dynamický jazyk vhodný pre vedecké účely s výkonom porovnateľným s tradičnými staticky písanými jazykmi ako je napr. C. Poskytuje jednoduchosť a expresívnosť pre vysokoúrovňové numerické výpočty, rovnako ako jazyky R, MATLAB alebo Python, ale podporuje taktiež všeobecné programovanie. Medzi ďalšie výhody Julie oproti porovnateľným systémom patrí, že:

- je bezplatný open-source projekt s viac než 1000 prispievateľmi licencovaný MIT,
- používateľom definované typy sú rovnako rýchle a kompaktné ako vstavané typy,
- je navrhnutý pre paralelnosť a rozdelenie výpočtu.

Výber programovacieho jazyka Julia bol podmienený hlavne jednoduchou implementáciou získaných výsledkov do grafickej podoby, keďže tento jazyk nebol vytvorený len pre optimalizáciu a matematické programovanie. Pri použití čiste modelovacieho jazyka GAMS (The General Algebraic Modeling System) by grafická implementácia tak ľahká nebola. Ďalším rozhodovacím faktorom bola bezplatnosť modelovacieho jazyka pre matematickú optimalizáciu zabudovaného v Julii – JuMP (Julia for Mathematical Programming). Tak ako modelovacie jazyky AIMMS (Advanced Interactive Multidimensional Modeling System) a AMPL (A Mathematical Programming Language) aj JuMP podporuje množstvo open-source a komerčných riešičov pre rôzne triedy problémov, vrátane lineárneho, zmiešaného celočíselného a nelineárneho programovania.

IBM® ILOG® CPLEX® Optimizer (často označovaný len ako CPLEX) je jeden z najviac pokrokových optimalizačných riešičov, hlavne pre lineárne, zmiešané celočíselné a kvadratické programovanie. CPLEX dokáže predložiť presné a logické rozhodnutia pre rôzne problémy (plánovanie, alokácia zdrojov) s využitím výkonných algoritmov, podporuje všetky veľkosti optimalizačných modelov s neobmedzeným množstvom rozhodovacích premenných a obmedzení a je škálovateľný.

Komerčný riešič CPLEX bol jednoznačnou voľbou vďaka jeho komplexnosti pri riešení úloh matematického programovania. Jedným z dôvodov bolo aj to, že tento produkt spoločnosti IBM je bezplatne k dispozícii akademickým pracovníkom a študentom. Porovnanie schopností troch najznámejších komerčných riešičov – CPLEX, GuRoBi a XPRESS môžeme nájsť v článku [2].

# 3 Aplikácia matematického programovania

Táto kapitola sa bude zaoberať zostavením a postupným vylepšovaním optimalizačného modelu, ktorý bude slúžiť na podporu rozvoja výstavby nových spracovateľských zariadení pre prechod na obehové hospodárstvo. Jedná sa o tokovú úlohu zmiešaného celočíselného programovania, teoreticky opísanú v kapitole 2.

## 3.1 Formulácia komplexnej úlohy

Majme množinu obcí, ktoré produkujú isté frakcie komunálneho odpadu (zmesový komunálny odpad, plast, sklo, papier, biologicky rozložiteľný odpad) a množinu existujúcich zariadení, nachádzajúcich sa v týchto obciach, v ktorých odpad môžeme buď uložiť (skládky) alebo spracovať (spalovne, prekladacie stanice, triediace linky). Majme taktiež množinu potenciálnych zariadení, ktoré v definovaných obciach môžeme vystavať.

Definujeme orientovaný graf, ktorého uzly tvoria spomínané obce a zariadenia a orientované hrany sú tvorené cestnou a železničnou infraštruktúrou medzi týmito uzlami. Hrany predstavujúce cesty môžeme rozdeliť do niekoľkých skupín: hrany spájajúce obce navzájom, hrany spájajúce obce so zariadením v nich a hrany spájajúce zariadenia medzi sebou. Prvému typu hrán je priradená váha, ktorá predstavuje vzdialenosť medzi prepojenými uzlami. Druhý typ hrán je ohodnotený váhami rovnými nule. Uvažujeme teda, že zariadenia sa nachádzajú v konkrétnych obciach, prípadne pár kilometrov od obce, čo zanedbávame. Posledný typ hrán predstavuje ohodnotené hrany, cez ktoré sa preváža spracovaný odpad medzi zariadeniami (výmet z triediacich liniek do spalovní, struska zo spalovní na skládky, zlisovaný odpad z prekladacích staníc do tzv. koncoviek – spalovní alebo skládok). Ohodnotenie týchto hrán je dané najkratšou cestou medzi obcami, pod ktoré tieto zariadenia spadajú. Hrany predstavujúce železničné trasy spájajú spracovateľské zariadenia a slúžia na prevoz zlisovaného odpadu naloženého v prekladacích staniciach do koncoviek.

Cieľom výsledného matematického modelu je minimalizovať náklady spojené s prepravou a spracovaním jednotlivých frakcií odpadu a s výstavbou nových spracovateľských zariadení. Je vhodné dodať, že produkcia odpadu a náklady na jeho prevoz a spracovanie sú vyjadrené pre ročnú prevádzku.

## 3.2 Označenie

Pred začiatkom návrhu modelu je potrebné zaviesť označenie všetkých množín, vstupných parametrov a premenných, ktoré budeme používať.

### Množiny:

$i \in I$  množina všetkých uzlov

$i \in I^{ex} \subset I$  množina uzlov predstavujúcich obce a existujúce zariadenia (spalovne, prekladacie stanice, triediace linky, skládky) v nich

### 3 APLIKÁCIA MATEMATICKÉHO PROGRAMOVANIA

$i \in I^n \subset I$	množina uzlov reprezentujúcich prekladacie stanice, v ktorých je možné nakladať zlisovaný odpad na nákladné vlaky
$i \in I^p \subset I$	množina uzlov predstavujúcich potenciálne spracovateľské zariadenia (spalovne, prekladacie stanice, triediace linky)
$i \in I^s \subset I$	množina uzlov reprezentujúcich skládky
$j \in J$	množina všetkých hrán
$j \in J^x \subset J$	množina hrán medzi obcami navzájom a medzi obcami a zariadeniami v nich
$j \in J^y \subset J$	množina hrán spájajúcich zariadenia navzájom
$o \in O$	množina typov odpadu
$o \in O^l \subset O$	množina zlisovaných typov odpadu (zlisovaný zmesový komunálny odpad, zlisovaný plast)
$o \in O^n \subset O$	množina nespracovaných frakcií odpadu (zmesový komunálny odpad, plast, sklo, papier, biologicky rozložiteľný odpad)
$o \in O^s \subset O$	množina spracovaných typov odpadu (struska, zlisovaný odpad, výmet)
$r \in R$	množina trás v železničnej dopravnej sieti
$t \in T$	množina kapacít potenciálnych spracovateľských zariadení

#### Vstupné parametre:

$B_{i,\tilde{o},o}$	matica, ktorej prvky predstavujú podiel pôvodnej hmotnosti nespracovaného odpadu $\tilde{o}$ , z ktorého vznikne reziduum $o$ v uzle $i$
$M_{i,j}$	incidenčná matica pre prepravu odpadu medzi obcami navzájom a medzi obcami a zariadeniami v nich
$N_{i,j}$	incidenčná matica pre reziduálne toky medzi zariadeniami
$Z_{r,i,o}$	matica pre železničnú dopravnú sieť, viď (3.1)
$c_{i,o}^{ex}$	cena za spracovanie tony odpadu $o$ v existujúcom zariadení v uzle $i$ [Kč/t]
$c_i^p$	cena za výstavbu potenciálneho spracovateľského zariadenia v uzle $i$ [Kč]
$c_o^{ph}$	náklady na pohonné hmoty pre smetiarske vozidlá prepravujúce odpad $o$ [Kč/t.km]
$c_o^{ph,s}$	náklady na pohonné hmoty pre vozidlá prepravujúce spracovaný odpad $o$ [Kč/t.km]
$d_j$	dĺžka hrany $j$ [km]
$e_r$	maximálny možný počet železničných cyklov $r$ počas jedného roka [-]

### 3.3 ZÁKLADNÝ MODEL

$k_i^{ex}$	kapacita existujúceho zariadenia v uzle $i$ [t]
$k_t^p$	možné kapacity potenciálnych spracovateľských zariadení [t]
$k_{i,o}^u$	kapacita odpadu $o$ v uzle $i$ (povolenie spracovania odpadu $o$ v uzle $i$ ) [t]
$k_r^z$	kapacita jedného nákladného vlaku absolvujúceho trasu $r$ [t]
$p_{i,o}$	koeficient pomeru odpadu $o$ v zariadení v uzle $i$ (vyjadruje, koľkokrát sa odpad $o$ započítava do kapacity spracovateľského zariadenia) [-]
$w_{i,o}$	ročná produkcia odpadu $o$ v uzle $i$ [t]

#### Premenné:

$r_{i,o}$	reziduum $o$ (struska, zlisovaný odpad, výmet) vzniknuté v uzle $i$ [t]
$x_{j,o}$	množstvo odpadu $o$ (zmesový komunálny odpad, plast, sklo, papier, biologicky rozložiteľný odpad) prevezeného hranou $j$ medzi obcami navzájom a medzi obcami a zariadeniami v nich [t]
$y_{j,o}$	množstvo odpadu $o$ (struska, zlisovaný odpad, výmet) prevezeného hranou $j$ medzi zariadeniami [t]
$z_{i,o}$	množstvo odpadu $o$ (zlisovaný odpad) naloženého alebo vyloženého z nákladného vlaku v uzle $i$ [t]
$\gamma_r$	počet absolvovaní trasy $r$ počas jedného roka – celočíselná premenná [-]
$\delta_{i,t}$	rozhodnutie o výstavbe spracovateľského zariadenia (spalovne, prekladacej stanice, triediacej linky) s kapacitou $t$ v uzle $i$ – binárna SOS1 premenná [-]

Pred tvorbou samotného modelu je potrebné isté pedspracovanie dát. Zostavíme teda incidenčné matice  $M_{i,j}$  a  $N_{i,j}$  z dostupných informácií o prepojení uzlov orientovanými hranami a maticu  $B_{i,\bar{o},o}$ . Maticu  $Z_{r,i,o}$  zostrojíme nasledovne:

$$z_{r,i,o} = \begin{cases} 1 & \text{ak sa v uzle } i \text{ na trase } r \text{ môže odpad } o \text{ vykladať,} \\ -1 & \text{ak sa v uzle } i \text{ na trase } r \text{ môže odpad } o \text{ nakladať,} \\ 0 & \text{inak.} \end{cases} \quad (3.1)$$

### 3.3 Základný model

Prvotný model, v ktorom neuvažujeme potenciálne spracovateľské zariadenia, je tvorený účelovou funkciou (3.2), ktorá zohľadňuje jedine cenu za spracovanie odpadu v existujúcich zariadeniach:

$$\min \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex}. \quad (3.2)$$



Prvými dvoma obmedzeniami sú bilančné nerovnice pre všetky existujúce uzly a typy odpadu. Nerovnica (3.3) povoľuje alebo zakazuje spracovanie istých typov odpadu v určených spracovateľských zariadeniach a ich uloženie na skládke. Taktiež zakazuje spracovanie všetkých typov odpadu v obciach. Pri zakázaní spracovania odpadu sa  $k_{i,o}^u$  volí rovné nule, zatiaľ čo pri povolení sa tento parameter rovná nejakej veľkej konštante (tzv. big M).

$$\sum_{j \in J^x} M_{i,j} x_{j,o} + w_{i,o} + r_{i,o} \leq k_{i,o}^u, \quad \forall i \in I^{ex}, \forall o \in O. \quad (3.3)$$

Obmedzenie (3.4) zaručuje, že množstvo odpadu, ktoré sa môže spracovať alebo uložiť v danom uzle nepresiahne jeho kapacitu, pričom uzly predstavujúce obce majú kapacitu nulovú. Keďže výhrevnosť plastu v spaľovniach je omnoho vyššia ako výhrevnosť ostatných frakcií odpadu, použijeme koeficient  $p_{i,o}$ , ktorý vyjadruje kolkokrát sa daný odpad započítava do kapacity spracovateľského zariadenia.

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + w_{i,o} + r_{i,o} \right) \leq k_i^{ex}, \quad \forall i \in I^{ex}. \quad (3.4)$$

Rovnica (3.5) vyjadruje, aké množstvo rezidua (strusky, zlisovaného odpadu alebo výmetu) vznikne v uzle  $i$  po spracovaní odpadu  $\tilde{o}$ .

$$\sum_{\tilde{o} \in O} \sum_{j \in J^x} M_{i,j} x_{j,\tilde{o}} B_{i,\tilde{o},o} = r_{i,o}, \quad \forall i \in I^{ex}, \forall o \in O. \quad (3.5)$$

Nerovnice (3.6)–(3.8) sú obmedzeniami nezápornosti produkcie odpadu, množstva odpadu prevezeného hranou a množstva vzniknutého rezidua.

$$\sum_{j \in J^x} M_{i,j} x_{j,o} + w_{i,o} + r_{i,o} \geq 0, \quad \forall i \in I^{ex}, \forall o \in O, \quad (3.6)$$

$$x_{j,o} \geq 0, \quad \forall j \in J^x, \forall o \in O, \quad (3.7)$$

$$r_{i,o} \geq 0, \quad \forall i \in I^{ex}, \forall o \in O. \quad (3.8)$$

Účelovú funkciu (3.2) môžeme modifikovať zohľadnením ceny za prevoz odpadu, ktorá závisí na množstve odpadu prevezeného cez jednotlivé hrany a takisto aj na ich dĺžke. Cenu, vyjadrenú ako súčin týchto dvoch faktorov, môžeme ďalej vynásobiť konštantou  $c_o^{ph}$  vyjadrujúcou náklady na pohonné hmoty pre rôzne frakcie odpadu.

$$\min \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex} + \sum_{j \in J^x} \sum_{o \in O} x_{j,o} d_j c_o^{ph}. \quad (3.9)$$

### 3.4 Zavedenie premenných špeciálnych typov

Model uvedený v podkapitole 3.3 berie do úvahy len spracovateľské zariadenia, ktoré už v obciach existujú. Môže sa stať, že tieto zariadenia nemajú dostatočnú kapacitu, aby spracovali odpad vyprodukovaný v prilahlých obciach, preto je vhodné premýšľať o rozšírení kapacity existujúcich zariadení alebo o zatiaľ neexistujúcich zariadeniach, ktoré

### 3.5 ROZŠÍRENIE MODELU O REZIDUÁLNE TOKY

v týchto obciach môžeme vystavať. Použijeme teda SOS1 premenné, ktoré vyjadrujú rozhodnutie o ich výstavbe v určenej obci.

Účelovú funkciu (3.9) zo základného modelu rozšírime o náklady spojené s výstavbou novej spalovne, prekladacej stanice alebo triediacej linky s kapacitou  $k_t^p$ .

$$\min \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex} + \sum_{i \in I^p} \sum_{t \in T} k_t^p \delta_{i,t} c_i^p + \sum_{j \in J^x} \sum_{o \in O} x_{j,o} d_j c_o^{ph}. \quad (3.10)$$

S novými uzlami bude obmedzenie (3.3) z predchádzajúcej podkapitoly platiť nielen pre všetky obce a existujúce zariadenia, ale aj pre tie novo vystavané. K druhému obmedzeniu (3.4) pribudne obdobná bilančná rovnica (3.11), prislúchajúca novým spracovateľským zariadeniam, ktoré môžu nadobúdať jednu z vopred určených kapacít, definovaných parametrom  $k_t^p$ .

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + w_{i,o} + r_{i,o} \right) \leq \sum_{t \in T} k_t^p \delta_{i,t}, \quad \forall i \in I^p. \quad (3.11)$$

SOS1 premenné  $\delta_{i,t}$  nadobúdajú binárne hodnoty. Hodnota 0 znamená, že v uzle  $i$  nie je vystavané nové spracovateľské zariadenie, zatiaľ čo hodnota 1 značí, že v uzle  $i$  je vystavané nové spracovateľské zariadenie s kapacitou určenou indexom  $t$ . Obmedzenia (3.12) a (3.13) zaisťujú vlastnosti SOS1 premenných, spomínané v podkapitole 2.4.1.

$$\sum_{t \in T} \delta_{i,t} \leq 1, \quad \forall i \in I^p, \quad (3.12)$$

$$\delta_{i,t} \in \{0, 1\}, \quad \forall i \in I^p, \quad \forall t \in T. \quad (3.13)$$

### 3.5 Rozšírenie modelu o reziduálne toky

Keďže v spalovniach vzniká struska, ktorú už následne nemôžeme ďalej spracovať, musíme ju dopraviť na skládku. Aby sme tento vedľajší produkt po spracovaní odpadu nemuseli zdĺhavo prevážať po definovanej dopravnej sieti medzi obcami, vytvorili sme hrany, vedúce zo spalovní priamo na skládku. Takéto hrany sme vytvorili aj medzi prekladacími stanicami a koncovými zariadeniami, kde sa zlisovaný odpad spaľuje alebo ukladá. Do spalovní taktiež vedú hrany z triediacich liniek, po ktorých putuje výmet. Účelovú funkciu (3.10) upravíme tak, aby zohľadňovala cenu za spracovanie odpadu, ktorý bol na skládku privezený dodatočnými hranami a takisto aj náklady na tento prevoz.

$$\begin{aligned} \min \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex} + \sum_{i \in I^{ex}} \sum_{j \in J^y} \sum_{o \in O} N_{i,j} y_{j,o} c_{i,o}^{ex} + \sum_{i \in I^p} \sum_{t \in T} k_t^p \delta_{i,t} c_i^p + \\ + \sum_{j \in J^x} \sum_{o \in O} x_{j,o} d_j c_o^{ph} + \sum_{j \in J^y} \sum_{o \in O} y_{j,o} d_j c_o^{ph,s}. \end{aligned} \quad (3.14)$$

Pridaním hrán sa zmenia aj obmedzenia (3.3), (3.4), (3.5), (3.6) a (3.11), do ktorých musíme pridať toky odpadu cez dodatočné hrany.



$$\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + w_{i,o} + r_{i,o} \leq k_{i,o}^u, \quad \forall i \in I, \quad \forall o \in O, \quad (3.15)$$

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + w_{i,o} + r_{i,o} \right) \leq k_i^{ex}, \quad \forall i \in I^{ex}, \quad (3.16)$$

$$\sum_{\bar{o} \in O} \left( \sum_{j \in J^x} M_{i,j} x_{j,\bar{o}} + \sum_{j \in J^y} N_{i,j} y_{j,\bar{o}} \right) B_{i,\bar{o},o} = r_{i,o}, \quad \forall i \in I, \quad \forall o \in O, \quad (3.17)$$

$$\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + w_{i,o} + r_{i,o} \geq 0, \quad \forall i \in I, \quad \forall o \in O, \quad (3.18)$$

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + w_{i,o} + r_{i,o} \right) \leq \sum_{t \in T} k_t^p \delta_{i,t}, \quad \forall i \in I^p. \quad (3.19)$$

Nerovnicu (3.7) obmedzujúcu nezápornosť množstva odpadu prevezeného hranou musíme nahradiť štyrmi obdobnými obmedzeniami. Obmedzenia (3.20) a (3.21), týkajúce sa premenných  $x_{j,o}$ , povoľujú prevoz nespracovaného odpadu medzi obcami a zariadeniami a zakazujú prevoz takéhoto odpadu medzi zariadeniami navzájom. Obmedzenia (3.22) a (3.23), vzťahujúce sa na premenné  $y_{j,o}$ , zas zakazujú prevoz už spracovaného odpadu medzi obcami a zariadeniami a povoľujú jeho prevoz zo spalovní na skládky, z prekladacích staníc do koncoviek a z triediacich liniek do spalovní.

$$x_{j,o} \geq 0, \quad \forall j \in J^x, \quad \forall o \in O^n, \quad (3.20)$$

$$x_{j,o} = 0, \quad \forall j \in J^x, \quad \forall o \in O^s, \quad (3.21)$$

$$y_{j,o} \geq 0, \quad \forall j \in J^y, \quad \forall o \in O^s, \quad (3.22)$$

$$y_{j,o} = 0, \quad \forall j \in J^y, \quad \forall o \in O^n. \quad (3.23)$$

### 3.6 Rozšírenie modelu o novú formu dopravy

V predchádzajúcich podkapitolách bol modelovaný prevoz odpadu realizovaný jedine pomocou cestnej dopravy. Teraz tento model obohatíme o prevoz odpadu v kontajneroch umiestnených na súpravách nákladných vlakov. Účelová funkcia (3.24) sa znovu rozrastie o jeden člen reprezentujúci náklady za prepravu odpadu novou formou dopravy.

$$\begin{aligned} \min \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex} + \sum_{i \in I^{ex}} \sum_{j \in J^y} \sum_{o \in O} N_{i,j} y_{j,o} c_{i,o}^{ex} + \sum_{i \in I^p} \sum_{t \in T} k_t^p \delta_{i,t} c_i^p + \\ + \sum_{j \in J^x} \sum_{o \in O} x_{j,o} d_j c_o^{ph} + \sum_{j \in J^y} \sum_{o \in O} y_{j,o} d_j c_o^{ph,s} + \sum_{r \in R} \gamma_r c_r^z. \end{aligned} \quad (3.24)$$

Obmedzenia (3.15), (3.16), (3.17), (3.18) a (3.19) musia byť doplnené o člen, ktorý vyjadruje množstvo odpadu prevezeného medzi uzlami po jednotlivých železničných trasách.

### 3.6 ROZŠÍRENIE MODELU O NOVÚ FORMU DOPRAVY

$$\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \leq k_{i,o}^u, \quad \forall i \in I, \quad \forall o \in O, \quad (3.25)$$

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \right) \leq k_i^{ex}, \quad \forall i \in I^{ex}, \quad (3.26)$$

$$\sum_{\bar{o} \in O} \left( \sum_{j \in J^x} M_{i,j} x_{j,\bar{o}} + \sum_{j \in J^y} N_{i,j} y_{j,\bar{o}} + \sum_{r \in R} Z_{r,i,\bar{o}} z_{i,\bar{o}} \right) B_{i,\bar{o},o} = r_{i,o}, \quad \forall i \in I, \quad \forall o \in O, \quad (3.27)$$

$$\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \geq 0, \quad \forall i \in I, \quad \forall o \in O, \quad (3.28)$$

$$\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \right) \leq \sum_{t \in T} k_t^p \delta_{i,t}, \quad \forall i \in I^p. \quad (3.29)$$

Bilančná rovnica (3.30) zaisťuje, že množstvo odpadu naloženého na nákladný vlak sa rovná množstvu odpadu z nákladného vlaku vyloženého.

$$\sum_{i \in I} Z_{r,i,o} z_{i,o} = 0, \quad \forall r \in R, \quad \forall o \in O. \quad (3.30)$$

Množstvo odpadu, ktoré je na vlak naložené, musí taktiež spĺňať aj kapacitné obmedzenie, definované nerovnicou (3.31). Keďže prvky matice  $Z_{r,i,o}$  sú rovné  $-1$ , ak je  $i$  uzol, v ktorom sa odpad na nákladné vlaky nakladá, musíme v tomto obmedzení pred sumu položiť znamienko mínus, aby bola splnená podmienka nezápornosti.

$$0 \leq - \sum_{i \in I^n} \sum_{o \in O^l} Z_{r,i,o} z_{i,o} \leq k_r^z \gamma_r, \quad \forall r \in R. \quad (3.31)$$

Obmedzenie (3.32) vyjadruje maximálny počet absolvovaní železničných trás počas jedného roka. Od premennej popisujúcej počet trás požadujeme taktiež celočíselnosť, vyjadrenú obmedzením (3.33).

$$\gamma_r \leq e_r, \quad \forall r \in R, \quad (3.32)$$

$$\gamma_r \in \mathbb{Z}_0^+, \quad \forall r \in R. \quad (3.33)$$

Ako posledná sa k vyššie uvedeným obmedzeniam pridá nerovnica (3.34), ktorá je obmedzením nezápornosti pre novú premennú  $z_{i,o}$ .

$$z_{i,o} \geq 0, \quad \forall i \in I, \quad \forall o \in O. \quad (3.34)$$

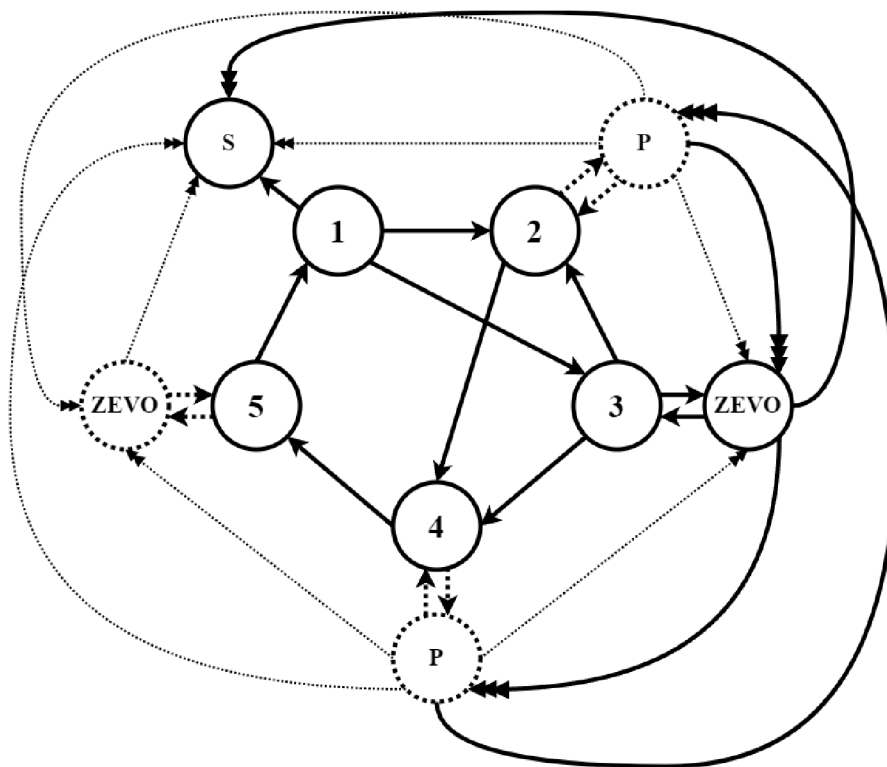
Výsledný model pozostávajúci z finálnej účelovej funkcie a všetkých obmedzení je spolu s označením k dispozícii v prílohe B.

## 4 Vzorový príklad

V tejto kapitole si uvidíme malú úlohu, na ktorú aplikujeme optimalizačný model zostrojený v kapitole 3 a popíšeme jeho výsledky. Ukážka funkčnosti modelu na malom množstve uzlov a hrán je vhodná pre lepšie pochopenie celej úlohy formulovanej v podkapitole 3.1.

### 4.1 Predstavenie úlohy

Uvažujme zoskupenie piatich obcí, vyobrazené na obr. 4.1, v ktorých sa produkuje zmesový komunálny odpad a plasty. Vytvorený odpad sa následne pomocou smetiarskych vozidiel zváža na skládku (S) nachádzajúcu sa v obci č. 1 alebo do spalovne (ZEVO) umiestnenej v obci č. 3. Keďže existujúca spalovňa nemá dostatočne veľkú kapacitu, aby zaistila spracovanie odpadu vyprodukovaného vo všetkých obciach a našou úlohou je obmedziť skládkovanie, musíme rozmýšľať o dodatočných spracovateľských zariadeniach v ostatných obciach. K existujúcej spalovni teda pridáme potenciálnu spalovňu (ZEVO), ktorá môže byť vystavaná v obci č. 5. Prevoz nespracovaného odpadu je nákladnejší ako prevoz lisovaného odpadu, preto uvažujeme výstavbu prekladacích staníc (P) v obciach č. 2 a 4. V tomto medzizariadení sa odpad zlisuje a takto predspracovaný môže putovať do koncoviek – spalovní a skládok, pomocou cestnej alebo železničnej dopravy. Železničná trasa spája obe prekladacie stanice, ktoré zároveň fungujú ako nakládky na nákladné vlaky, spolu s existujúcou spalovňou, ktorá slúži ako vykládka. Tieto vlaky prevážajú kontajnery naplnené zlisovaným odpadom, ktoré sú vo vykladacej stanici pretiahnuté na nákladné automobily so špeciálnym systémom, slúžiacie na prevoz do spalovne.



Obr. 4.1: Orientovaný graf zodpovedajúci malej úlohe

## 4.2 VSTUPNÉ PARAMETRE

Na obr. 4.1 sú obce a existujúce zariadenia znázornené ako celistvé kruhy, zatiaľ čo potenciálna spaľovňa a prekladacie stanice zas kruhmi s prerušovanou hranicou. Existujúce hrany medzi uzlami sú vyznačené celistvými šípkami a tie, ktoré vedú z alebo do potenciálnych spracovateľských zariadení sú zakreslené prerušovanými šípkami. Podľa modelu z predchádzajúcej kapitoly existujú dva typy hrán:

1. primárne, po ktorých sa realizuje prevoz nespracovaného odpadu, sú znázornené jednoduchými šípkami,
2. sekundárne, cez ktoré sa preváža spracovaný odpad, sú vyznačené dvojitými šípkami.

Železničné trasy sú zakreslené trojitými šípkami. Keďže v matici  $Z_{r,i,o}$  nie je určené poradie, v ktorom zariadenia za sebou nasledujú, znázornenie na obr. 4.1 je len ilustračné.

## 4.2 Vstupné parametre

Ročná produkcia odpadu v obciach je uvedená v tabuľke 4.1. Dĺžka primárnych hrán je uvedená v tabuľke 4.2, dĺžka sekundárnych hrán odpovedá dĺžke najkratšej cesty medzi obcami, v ktorých sa zariadenia, medzi ktorými túto hranu vedieme, nachádzajú.

obec	produkcia [t]	
	zmesový komunálny odpad	plasty
1	12 800	5 100
2	35 000	16 500
3	14 200	4 900
4	70 600	31 700
5	20 000	8 000
spolu	152 600	66 200

Tabuľka 4.1: Produkcia odpadu

počiatočná obec	koncová obec	vzdialenosť [km]
1	2	14,7
1	3	32,1
2	4	17,9
3	2	8,6
3	4	42,4
4	5	17,6
5	1	22,5

Tabuľka 4.2: Vzďalenosťi obcí

Kapacitné obmedzenie skládky je 1 000 kt a existujúcej spaľovne je 100 kt. Pri výstavbe nových zariadení sa model môže rozhodovať medzi 4 kapacitami spaľovní aj prekladacích staníc – 10 kt, 20 kt, 30 kt a 40 kt. Pri železničných trasách uvažujeme prevoz nákladným vlakom s 25 plošinovými vozňami, kde každý môže prevážať tri kontajnery s kapacitami po 9 t, tj. jedným železničným cyklom je možné previesť až 675 ton odpadu.

Pri spracovaní odpadu v spaľovniach dochádza k likvidácii odpadu. Z 15% pôvodnej hmotnosti zmesového komunálneho odpadu a z 20% pôvodnej hmotnosti plastu vznikne struska. V prekladacích staniciach sa privezený odpad lisuje, ale hmotnosť ostáva rovnaká, tj. 100% pôvodnej hmotnosti. Tieto hodnoty sú potrebné k naplneniu matice  $B_{i,\tilde{o},o}$ , kde index  $\tilde{o}$  značí z akého odpadu vznikne reziduum  $o$ .

Náklady na prevoz odpadu sa líšia v závislosti na prevážanom odpade. Pri nespracovanom odpade cena predstavuje 20 Kč/t.km, zatiaľ čo pri prevoze strusky alebo zlisovaného odpadu len 2,5 Kč/t.km. Náklady na uskutočnenie jednej železničnej trasy sú 120 000 Kč za jeden cyklus. Tieto náklady môžeme rozdeliť na prevádzkové a fixné. Medzi prevádzkové náklady patrí cena za nakládku a vykládku odpadu, náklady na elektrinu a služby a mzdy zamestnancov. Fixné náklady zahŕňajú prenájom alebo kúpu železničnej

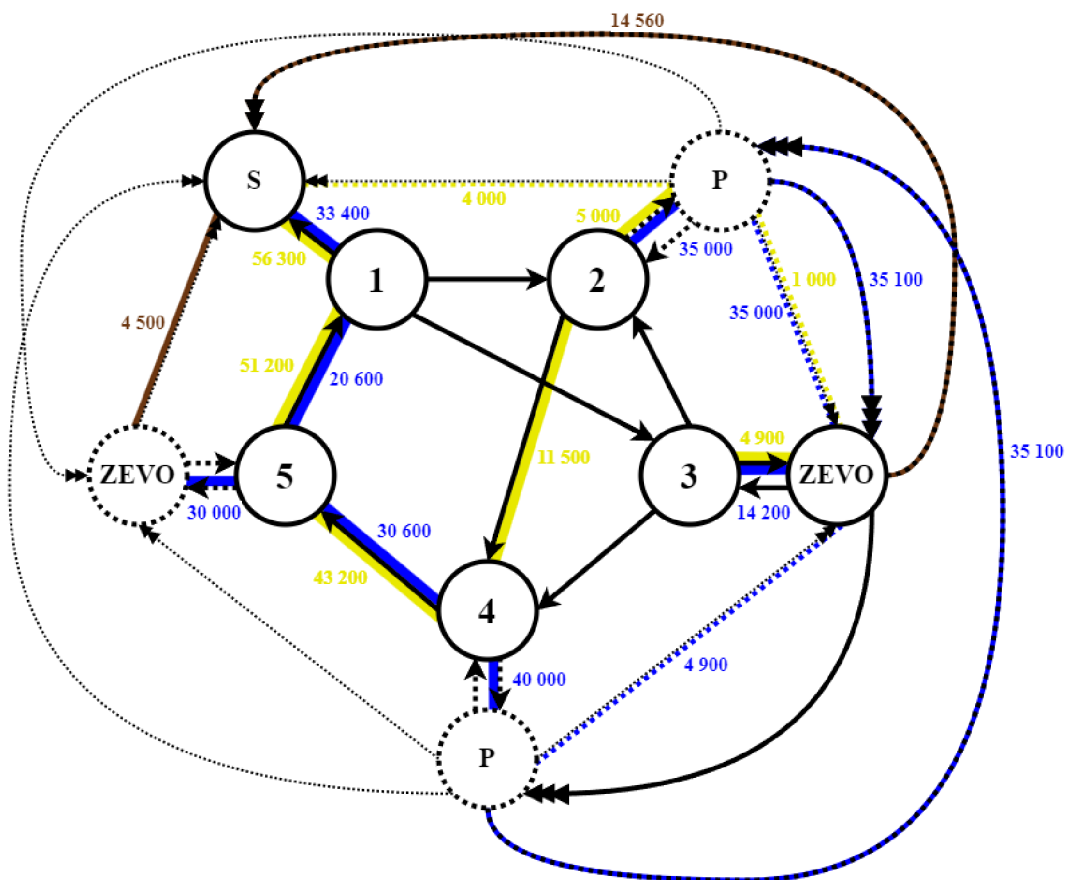
súpravy, tj. lokomotívy, kontajnerov a samotných vozov. Cena za spracovanie akéhokoľvek odpadu v spalovni je 2 500 Kč/t. Pri cene 3 700 Kč/t za uloženie odpadu na skládke berieme ohľad na obmedzenie skládkovania, tj. zohľadňujeme najvyšší možný skládkovací poplatok 1 850 Kč/t spolu s infláciou až do roku 2035. Náklady na výstavbu spracovateľských zariadení s rastúcou kapacitou pomerovo klesajú a sú uvedené v tabuľke 4.3. Jedná sa o ročné náklady s fixným investičným horizontom čo znamená, že v tejto cene je započítaná už aj cena za spracovanie odpadu.

kapacita [kt]	cena za výstavbu [milióny Kč]	
	spalovňa	prekladacia stanica
10	37,07	5,47
20	65,97	5,67
30	101,05	5,87
40	144,13	6,07

Tabuľka 4.3: Cena za výstavbu spracovateľských zariadení s rôznymi kapacitami

### 4.3 Výsledky

Po zadaní všetkých vstupných parametrov z podkapitoly 4.2 do modelu z kapitoly 3, implementovaného v programovacom jazyku Julia, sme získali výsledné toky odpadu znázornené na obr. 4.2.



Obr. 4.2: Výsledné toky odpadu



### 4.3 VÝSLEDKY

Množstvo prevezeného zmesového komunálneho odpadu je vyznačené modrými linkami, množstvo plastu žltými linkami. Ak je tento odpad vopred zlisovaný, tak je znázornený prerušovanými linkami v príslušných farbách. Množstvo strusky prepravenej zo spalovni na skládku je zakreslené hnedou linkou. Farebné čísla pri týchto čiarach znázorňujú koľko odpadu (v tonách) bolo prevezeného po daných hranách.

Z obrázku a z výsledkov modelu v programovacom jazyku Julia si môžeme všimnúť niekoľko vecí. Na spracovanie odpadu boli vystavané všetky tri potenciálne zariadenia – v obciach č. 2 a 4 prekládky s kapacitou 40 kt a v obci č. 5 spaľovňa s kapacitou 30 kt. Z grafu je zrejmé, že kapacita nových zariadení bola naplno využitá. Rovnako bola naplnená aj kapacita existujúcej spaľovne v obci č. 3, do ktorej bolo podľa tabuľky 4.4 privezených 95 100 t odpadu, z čoho 4 900 t tvoril plast, ktorý sa do kapacity každej spaľovne zarátava dvakrát. Ďalej si môžeme všimnúť, že železničný cyklus bol realizovaný  $52\times$  a na nákladných vlakoch bolo prevezené maximálne možné množstvo odpadu – 35 100 t. Naloženie odpadu bolo realizované v prekladacej stanici v obci č. 4 a vyloženie prebehlo v spaľovni v obci č. 3. Ak to železničná infraštruktúra umožňuje, tak je možné vyhnúť sa nakládku v prekladacej stanici v obci č. 2, čím sa znížia náklady na jeden cyklus prepočítaním koeficientu  $c_r^z$  v technicko-ekonomickom modeli. Takisto môžeme pozorovať, že v existujúcej spaľovni v obci č. 3 vzniklo z 89 200 t privezeného zmesového komunálneho odpadu 13 380 t strusky a z 5 900 t plastu zas 1 180 t strusky. V spaľovni v obci č. 5 s kapacitou 30 000 t, do ktorej bol dovezený jedine zmesový komunálny odpad, vzniklo 4 500 t strusky.

V nasledujúcej tabuľke je uvedené celkové množstvo odpadu spracovaného, resp. uloženého v koncových zariadeniach, skratka ZKO znamená zmesový komunálny odpad:

zariadenie	obec	množstvo odpadu [t]					
		ZKO	plast	struska	ZKO (lis)	plast (lis)	spolu
skládku	1	33 400	56 300	19 060	0	4 000	112 760
spaľovňa	3	14 200	4 900	0	75 000	1 000	95 100
spaľovňa	5	30 000	0	0	0	0	30 000

Tabuľka 4.4: Celkové množstvo odpadu v koncových zariadeniach

Z posledného stĺpca môžeme pozorovať, že veľké množstvo odpadu bolo poslané na skládku. Cieľom tejto práce je však skládkovanie obmedziť, preto je nutné v pôvodnom modeli vykonať nejaké zmeny.

Jednou zo zmien by mohlo byť pridanie obmedzenia, ktoré dodržiava podmienky uvedené v legislatíve EÚ hovoriace o množstve odpadu uloženého na skládkach,

$$\sum_{i \in I^s} \sum_{o \in O} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} \right) \leq \alpha \sum_{i \in I} \sum_{o \in O} w_{i,o}, \quad (4.1)$$

kde  $\alpha$  predstavuje podiel odpadu vyprodukovaného vo všetkých obciach, ktorý môže na skládkach skončiť.

V smernici [21] z roku 2018 sa popri zavedení obmedzenia skládkovania všetkého odpadu, ktorý je vhodný k recyklácii alebo ďalšiemu materiálovému alebo energetickému využitiu, uvádza aj obmedzenie podielu komunálneho odpadu uloženého na skládkach na 10% do roku 2035. Ak by sme však v našom modeli zvolili  $\alpha = 0,10$ , tak by model nemal prípustné riešenie, keďže maximálna spoločná kapacita oboch spaľovní predstavuje 140 000 t, pričom celkové množstvo vyprodukovaného odpadu je viac ako 200 000 t.

Tieto problémy by sa pri úlohe s reálnymi dátami dali vyriešiť niekoľkými spôsobmi. Prvým z týchto spôsobov je brať do úvahy aj iné spracovateľské zariadenia (napr. triediace linky), slúžiace na materiálové využitie privezeného odpadu. Tento nový typ zariadenia by mal pri modelovaní podobné obmedzenia ako spalovne a prekladacie stanice. Ďalším spôsobom by mohlo byť povolenie výstavby väčších spalovní, v ktorých by sa mohlo spracovať omnoho viac odpadu ako v súčasnosti.

V úlohe s reálnymi dátami by sme mohli uvažovať ešte o jednom doplňujúcom obmedzení. Môže sa stať, že niektorá z obcí má uzatvorenú zmluvu s inou obcou, v ktorej stojí, že určité množstvo odpadu vyprodukované v obci A musí byť spracované v konkrétnom zariadení v obci B. V prípade našej malej úlohy by to mohla byť napr. obec č. 1, v ktorej sa nenachádza spracovateľské zariadenie a obec č. 3, v ktorej sa nachádza spalovňa. Takéto obmedzenie by teda udávalo minimálne množstvo určitej frakcie odpadu prevezené hranou vedúcou z obce č. 1 do obce č. 3. Všeobecne

$$x_{j,o} \geq m_{j,o}, \quad \forall j \in J^x, \forall o \in O, \quad (4.2)$$

kde  $m_{j,o}$  je rovné nule, ak na hranu  $j$  a odpad  $o$  nie sú kladené žiadne nároky. Analógia platí aj pre zazmluvnený prevoz spracovaného odpadu z jedného zariadenia do druhého.

$$y_{j,o} \geq m_{j,o}, \quad \forall j \in J^y, \forall o \in O. \quad (4.3)$$

# 5 Aplikácia modelu na reálnych dátach z Českej republiky

Motiváciou pre vznik tejto práce bol deficit multikomoditného modelu, ktorý by opisoval celý proces zaobchádzania s odpadom od jeho produkcie až po jeho spracovanie. Dostupné články a akademické práce na tému odpadového hospodárstva sa vždy zameriavali jedine na čiastkové problémy. S využitím týchto zdrojov vznikol model, ktorý tieto časti prepojil a tým sa stal súčasťou širšieho cieľa zautomatizovať možnosť počítania dopravných úloh v odpadovom hospodárstve.

## 5.1 Formulácia úlohy a vstupné dáta

Každá obec v Českej republike vyprodukuje isté množstvo komunálneho odpadu, ktoré je potrebné spracovať alebo uložiť. V rámci našej prípadovej štúdie uvažujeme 6 258 obcí, z ktorých sme vylúčili štyri vojenské obvody, ktoré neevidujú produkciu odpadu – Boletice, Březina, Hradiště a Libavá. V ostatných 6 254 obciach sa zameriavame na ročnú produkciu troch frakcií komunálneho odpadu – zmesový komunálny odpad, plast a papier. Vo všetkých obciach bolo spoločne vyprodukovaných 2 786 kt zmesového komunálneho odpadu, 264 kt plastu a 587 kt papiera. Vyprodukovaný odpad je následne pomocou smetiarskych vozidiel zvášaný z obcí do zariadení, ktorých funkcie si teraz predstavíme.

Zmesový komunálny odpad môže putovať do prekladacích staníc, ktoré slúžia k redukcii nákladov na prepravu odpadu zmenšením jeho objemu zlisovaním. V našej úlohe uvažujeme potenciálne stanice, tj. pred použitím ich musíme vystavať. Každá z nich sa môže nachádzať v jednej z 206 obcí s rozšírenou pôsobnosťou (ORP). Zoznam týchto obcí je možné nájsť v [10]. Kapacita prekladacích staníc môže byť rovnako ako v prípade malej úlohy 10 kt, 20 kt, 30 kt alebo 40 kt. Náklady na výstavbu, ktoré zahŕňajú aj spracovanie odpadu sú uvedené v tabuľke 4.3.

Nespracovaný papier a plast sú zvášané do triediacich liniek k dotriedeniu. Pri tomto procese vznikajú z 50% pôvodnej hmotnosti plastov a 10% pôvodnej hmotnosti papiera tzv. výmety, ktorých potenciál po upravení na tuhé alternatívne palivá (TAP) môže byť ďalej energeticky využitý. V každej ORP sa nachádza jedna triediaca linka, ktorú považujeme za vystavanú, priradujeme jej neobmedzenú kapacitu a náklady za spracovanie odpadu v nej zanedbávame.

Zlisovaný aj nezlisovaný zmesový komunálny odpad a výmet z triediacich liniek môže byť ďalej spracovaný v zariadeniach na energetické využitie odpadov (ZEVO). Tieto zariadenia predstavujú významný zdroj tepelnej a elektrickej energie, pričom v nich po spaľovaní vzniká z 20% pôvodnej hmotnosti odpadu reziduum nazývané struska. Na rozdiel od vyššie spomenutých zariadení je ich podstatne menej a môžeme ich rozdeliť na vystavané (do tejto kategórie zaraďujeme aj zariadenia, ktorých projekty existujú a sú v procese schvaľovania) a potenciálne. Prehľad všetkých lokalít ZEVO spolu s ich kapacitou a cenou za spracovanie, resp. vystavanie, je uvedený v tabuľke 5.1.

Posledné zariadenia, ktoré v tejto práci berieme do úvahy, sú skládky, kde sa odpad nespracováva, ale ukladá. Tu môže končiť nespracovaný zmesový komunálny odpad, jeho zlisovaná podoba z prekladacích staníc a struska vyprodukovaná v ZEVO. Skládky sú zriadené v 116 z 206 ORP a uvažujeme, že majú neobmedzenú kapacitu. Prvotný skládkovací



poplatok bol nastavený na 3 700 Kč/t. Táto výška vychádza z nového zákona o odpadoch [8] z roku 2020, v ktorom sa počíta s postupným navyšovaním poplatku zo súčasnej hodnoty až na uvedenú sumu.

obec	stav	kapacita [kt]	cena za spracovanie [Kč/t]	cena za výstavbu [milióny Kč]
Brno	vystavané	352	3 000	–
České Budějovice	v pláne	160	3 613	–
Chotíkov (Plzeň)	vystavané	95	3 000	–
Liberec	vystavané	96	3 000	–
Most	v pláne	150	3 345	–
Praha	vystavané	394	3 000	–
Štětí (Mělník)	v pláne	320	2 818	–
Čeperka	potenciálne	150	–	500,2
Písek	potenciálne	50	–	169,7
Tábor	potenciálne	40	–	134,5
Ždár nad Sázavou	potenciálne	100	–	365,3

Tabuľka 5.1: Prehľad ZEVO

Každá obec je spojená dvojicou orientovaných hrán (prvou vedúcou z nej a druhou vedúcou do nej) jedine s obcami, ktoré sú v jej bezprostrednej blízkosti. Tieto hrany predstavujú cestnú infraštruktúru, po ktorej sa nespracovaný odpad preváža v smetiarskych vozidlách. Náklady na takúto prepravu činia v priemere okolo 20 Kč/t.km pre všetok nespracovaný odpad, no všeobecne sa môžu líšiť na základe prepravovanej frakcie odpadu. Hrany, ktoré navzájom spájajú všetky zariadenia medzi sebou sú definované nasledovne:

- z prekladacích staníc vedú hrany do ZEVO a na skládky,
- z triediacich liniek vedú hrany, po ktorých putuje výmet, jedine do ZEVO,
- zo ZEVO vedú hrany len na skládky, kde môže byť struska uložená.

Tieto hrany taktiež reprezentujú cesty, no spracovaný odpad sa po nich vozí v nákladných autách a cena za prepravu je podstatne nižšia – 2,5 Kč/t.km. Kvôli rozsiahlosti úlohy a časovej náročnosti výpočtu (bližšie popísané v podkapitole 5.2) sme z prípadovej štúdie vylúčili hrany predstavujúce železničné trasy.

## 5.2 Výpočetné problémy

Pri práci s veľkým množstvom dát je veľká pravdepodobnosť, že narazíme na určité výpočetné problémy. V prípadovej štúdii s dátami z celej Českej republiky sme sa stretli predovšetkým s časovou náročnosťou výpočtu a preplnením pamäte v programovacom jazyku Julia.

Prvým problémom bolo importovanie dát, uložených v tabuľkách v softvéri Excel, do Visual Studio Code, v ktorom prebiehala editácia kódu. Jednou z týchto tabuliek bola matica vážených vzdialeností všetkých obcí, ktorú tvorilo  $6\,258 \times 6\,258 = 39\,162\,564$  prvkov. Tento problém bol vyriešený predspracovaním dát priamo v Exceli s využitím makra v objektovo orientovanom programovacom jazyku Visual Basic for Applications

## 5.3 VÝSLEDKY

(VBA). Makro slúžilo na vytvorenie incidenčnej matice vo vektorovej forme a následné priradenie vzdialenosti obcí k príslušným hranám v incidenčnej matici.

Druhým a najpodstatnejším problémom bola tvorba obrovskej úlohy v modelovacom jazyku JuMP. Po aplikovaní modelu zostrojeného v kapitole 3 (z ktorého sme vylúčili železničnú infraštruktúru) na spomínané dáta vznikla úloha s desiatkami tisíc premenných a stovkami tisíc obmedzení. Už len pri prvej bilančnej nerovnici (3.15), ktorá musí platiť pre všetky uzly (6 258 obcí a 539 zariadení) a všetkých 6 typov odpadu, vznikne vyše 40 000 obmedzení. Keďže JuMP vytvára každé obmedzenie samostatne, časová náročnosť tvorby jedného modelu bola niekoľko hodín, pričom pri zmene vstupných parametrov bolo nutné model tvoriť odznova. Riešením by mohla byť predpríprava dát do obvyklého tvaru tvoreného maticou dát a vektorom premenných, ktorý by bol menej pamäťovo náročný.

Tretím a posledným problémom bola časová náročnosť optimalizácie úlohy. Pri takej rozsiahlosti úlohy, s akou sa stretávame pri pracovaní s dátami z celej Českej republiky by nájdenie optimálneho riešenia trvalo hodiny, možno dni. Z tohto dôvodu sme sa rozhodli použiť parameter nazývaný relatívna optimality gap, nastavený na 10%. Po prekročení tejto hranice sa výpočet ukončil a riešič CPLEX poskytol riešenie, ktoré sme považovali za dostatočne optimálne.

## 5.3 Výsledky

Táto podkapitola bude venovaná konečným výsledkom úlohy zmiešaného celočíselného programovania aplikovanej na dátach zo zvozu odpadu v Českej republike.

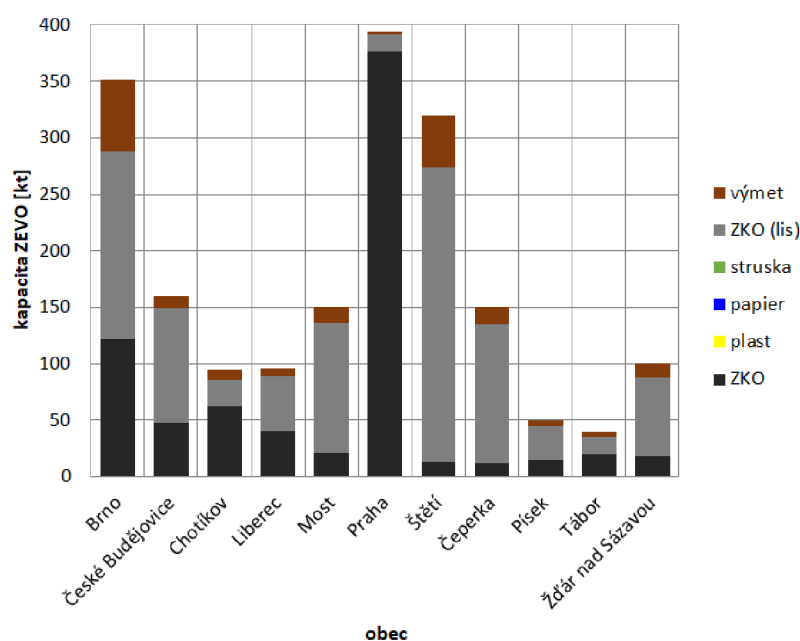
Cieľom našej práce bolo vytvoriť výpočetný nástroj, ktorý slúži na podporu rozhodovania o výstavbe nových spracovateľských zariadení za účelom obmedziť skládkovanie a umožniť postupný prechod na obehové hospodárstvo. Prvou požiadavkou, ktorú sme kládli na výstup modelu bola teda výstavba potenciálnych zariadení, hlavne potenciálnych ZEVO, v ktorých sa odpad energeticky využíva. Pri nastavení skládkovacieho poplatku na 3 700 Kč/t sme zistili, že model nerozhodol o výstavbe ani jedného nového zariadenia. Dôvodom bola skutočnosť, že náklady na prepravu odpadu do zariadení, spracovateľský poplatok (resp. náklady na výstavbu nového zariadenia), náklady na prepravu spracovaného odpadu a jeho spracovateľský poplatok niekoľkonásobne prevyšovali cenu za uloženie nespracovaného odpadu na skládke. Po postupnom zvyšovaní skládkovacieho poplatku sme dospeli až k cene 5 450 Kč/t, pri ktorej bol model nútený nejaké prekladacie stanice a ZEVO postaviť. Ďalšie výsledky patria k úlohe, v ktorej je skládkovací poplatok zvýšený práve na túto cenu.

### 5.3.1 Spracovateľské zariadenia

K 206 triediacim linkám a 7 existujúcim ZEVO bolo vystavaných 130 prekladacích staníc s rôznymi kapacitami. Zoznam ORP, v ktorých sa tieto stanice nachádzajú, spolu s príslušnými kapacitami, je uvedený v prílohe C. Vystavané boli aj všetky 4 potenciálne ZEVO v obciach Čeperka, Písek, Tábor a Žďár nad Sázavou.

Po prezretí grafu na obr. 5.1 a preskúmaní výsledkov získaných z Julie môžeme konštatovať, že kapacita všetkých – existujúcich aj novo vystavaných ZEVO, bola plne využitá. Môžeme vidieť, že v ZEVO boli spracované len tri povolené typy odpadu – zmesový komunálny odpad, lisovaný zmesový komunálny odpad a výmet, ktorý vznikol v trie-

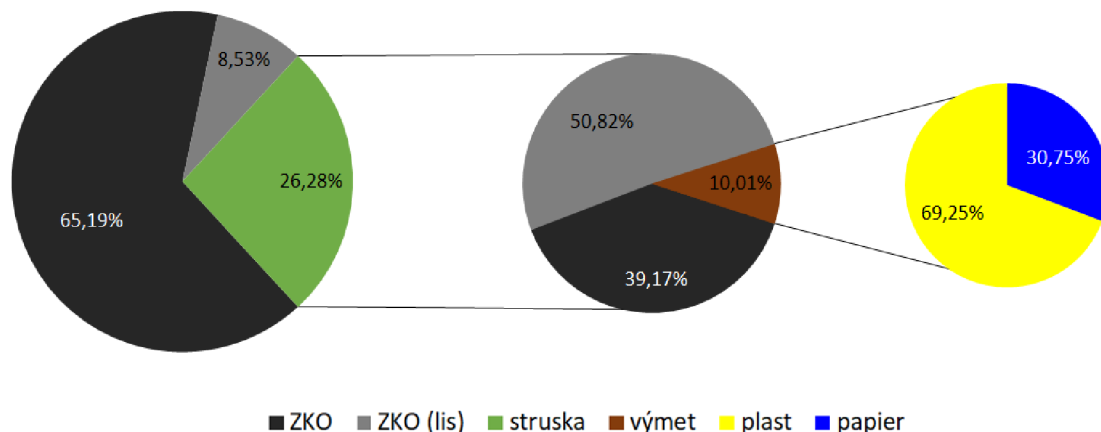
diacich linkách z plastu a papiera. Čo sa týka zloženia odpadu spracovaného v ZEVO, vo väčšine zariadení tvoril najväčší podiel zlisovaný zmesový komunálny odpad. Dôvodom bude pravdepodobne fakt, že v okolí týchto zariadení bolo postavených niekoľko prekladacích staníc a prevoz zlisovaného odpadu je menej nákladný ako prevoz nespracovaného odpadu. Jedinou výnimkou je Praha, kde najväčší podiel tvorí nespracovaný zmesový komunálny odpad. To bude zapríčinené tým, že dĺžka hrany, vedúcej od producenta (obce) k spracovateľovi (ZEVO) je v prípade Prahy nastavená na nulu, t.j. náklady za prepravu nespracovaného zmesového komunálneho odpadu sú taktiež nulové. Tento jav je možné eliminovať rozdelením Prahy (a iných väčších miest, v ktorých sa nachádzajú spracovateľské zariadenia) na viac uzlov, ktoré môžu reprezentovať napr. mestské časti. Na tomto grafe môžeme taktiež pozorovať veľké množstvo výmetu spracovaného v ZEVO v Brne. Dôvodom je zvoz tohto rezidua zo všetkých triediacich liniek nachádzajúcich sa vo východnej časti Českej republiky, viď 5.6.



Obr. 5.1: Využitie kapacity ZEVO v jednotlivých obciach

Všetok odpad spracovaný v ZEVO sa v podobe strusky zväžia na skládky. Pri skládkovacom poplatku 5 450 Kč/t bolo na skládkach uložených 1 451 kt z celkových 3 638 kt odpadu. Percentuálne zloženie tohto odpadu môžeme vidieť na prvom koláčovom grafe na obr. 5.2. Takmer dve tretiny (946 kt) tvorí nespracovaný zmesový komunálny odpad. Dôvodom je nedostatočné množstvo (resp. malá kapacita) zariadení na energetické využitie tejto frakcie odpadu v porovnaní s množstvom zmesového komunálneho odpadu, ktoré sa v Českej republike ročne vyprodukuje. Zhruba štvrtinu (381 kt) odpadu uloženého na skládkach tvorí struska, ktorej pomerové zloženie je vyobrazené na druhom grafe na obr. 5.2. Najväčšiu časť odpadu spracovaného v ZEVO, v ktorom struska vzniká, tvorí zlisovaný zmesový komunálny odpad pochádzajúci z prekladacích staníc. Približne 10% strusky však tvorí výmet z triediacich liniek, ktorého zloženie môžeme pozorovať na poslednom koláčovom grafe na obr. 5.2. Prevažnú časť výmetu tvorí plast, pretože zo separovaného papiera je v triediacich linkách väčšia výťažnosť odpadu vhodného k materiálovému využitiu.

### 5.3 VÝSLEDKY



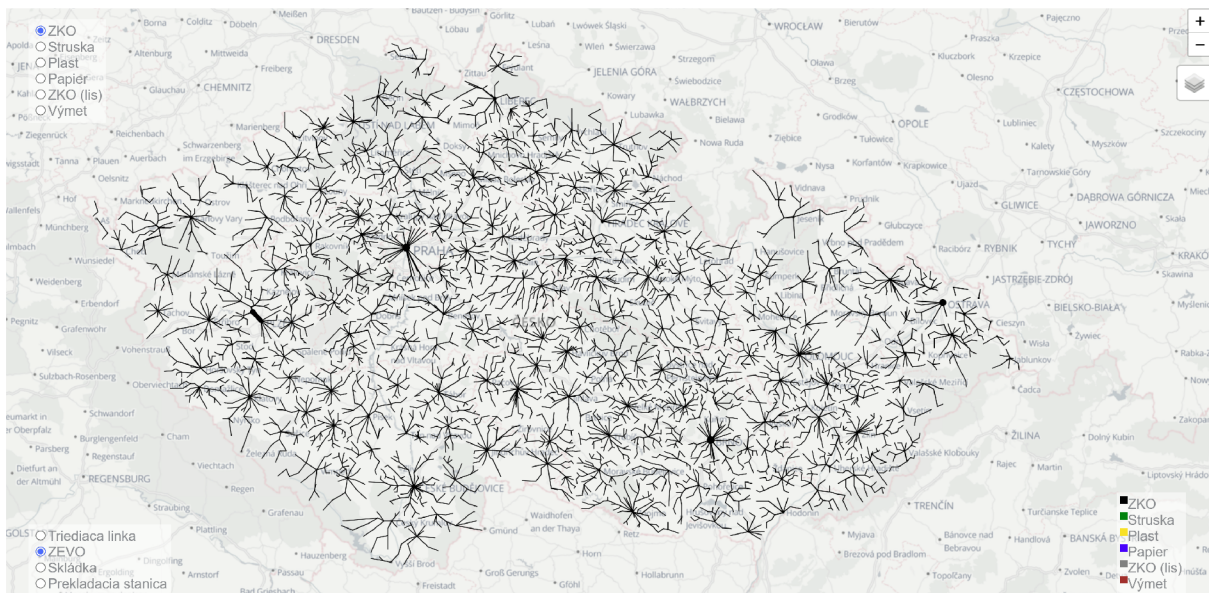
Obr. 5.2: Zloženie odpadu uloženého na skládkach

#### 5.3.2 Toky odpadu

Mapy, vykreslené v tejto časti práce, boli zhotovené pomocou vizualizačného softvéru, ktorý bol predmetom diplomovej práce [9]. Tento softvér umožňuje navoliť si typ odpadu, ktorého toky sa zobrazia, pomocou prepínača v ľavom hornom rohu obrazovky. Používateľ má taktiež možnosť vyobrazenia koláčových grafov predstavujúcich percentuálne zastúpenie odpadov spracovaných v konkrétnych zariadeniach pomocou prepínača v ľavom dolnom rohu obrazovky. Legenda sa nachádza v pravom dolnom rohu obrazovky.

Najprv popíšeme toky nespracovaného odpadu, následne toky spracovaného odpadu. Rozmiestnenie spracovateľských zariadení je vyobrazené na mapách v prílohe D.

Na obr. 5.3 sú graficky znázornené toky zmesového komunálneho odpadu.



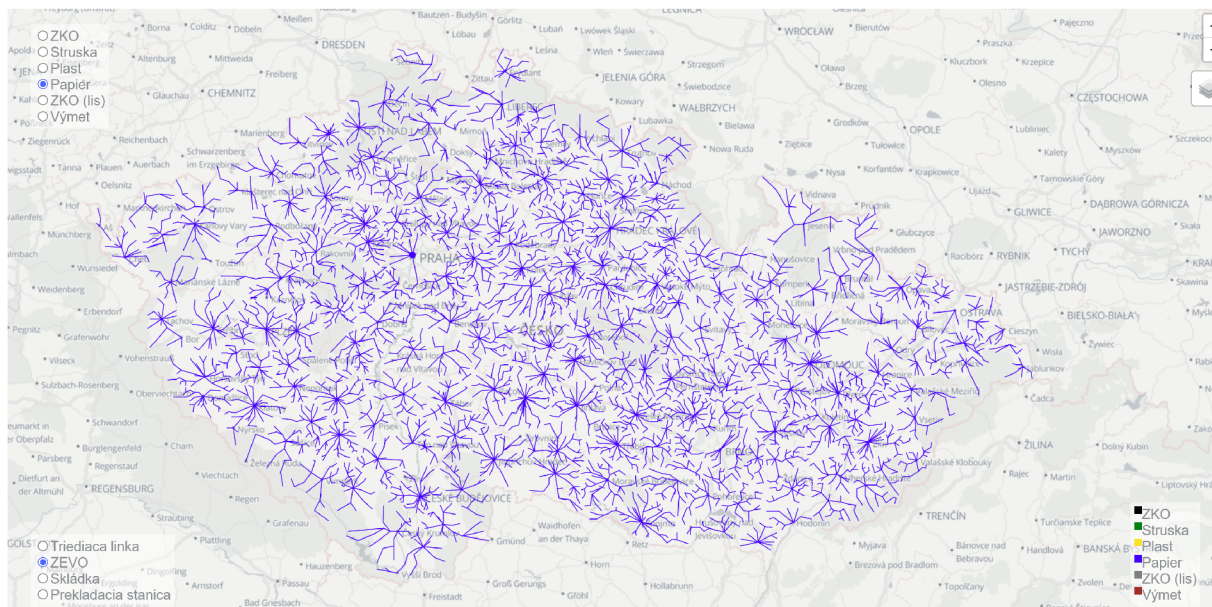
Obr. 5.3: Toky zmesového komunálneho odpadu



## 5 APLIKÁCIA MODELU NA REÁLNYCH DÁTACH Z ČESKEJ REPUBLIKY

Z obrázku je zrejmé, že táto frakcia odpadu má tendenciu zhlukovať sa v najbližších spracovateľských zariadeniach – prekladacích staniciach (viď obr. D.2), ZEVO (viď obr. 5.5) a skládkach (viď obr. D.3). Pri obciach so spalovňami s veľkou kapacitou (Brno, České Budějovice, Praha a Štětí) môžeme pozorovať rozsiahle toky, ktoré predstavujú zvoz z veľkého množstva malých príslušných obcí. V oblastiach s menšou hustotou zaľudnenia a teda menšou produkciou odpadu, ako je napr. CHKO Jeseníky v severovýchodnej časti Českej republiky, sú toky rozvetvené zas preto, lebo v týchto miestach je vystavaných veľmi málo spracovateľských zariadení. Na tejto mape môžeme pozorovať aj pár oblastí bez toku odpadu. Týmito miestami sú napr. vojenské obvody Hradiště pri Karlových Varoch a Libavá pri Olomouci, v ktorých sa produkcia odpadu nezaznamenáva.

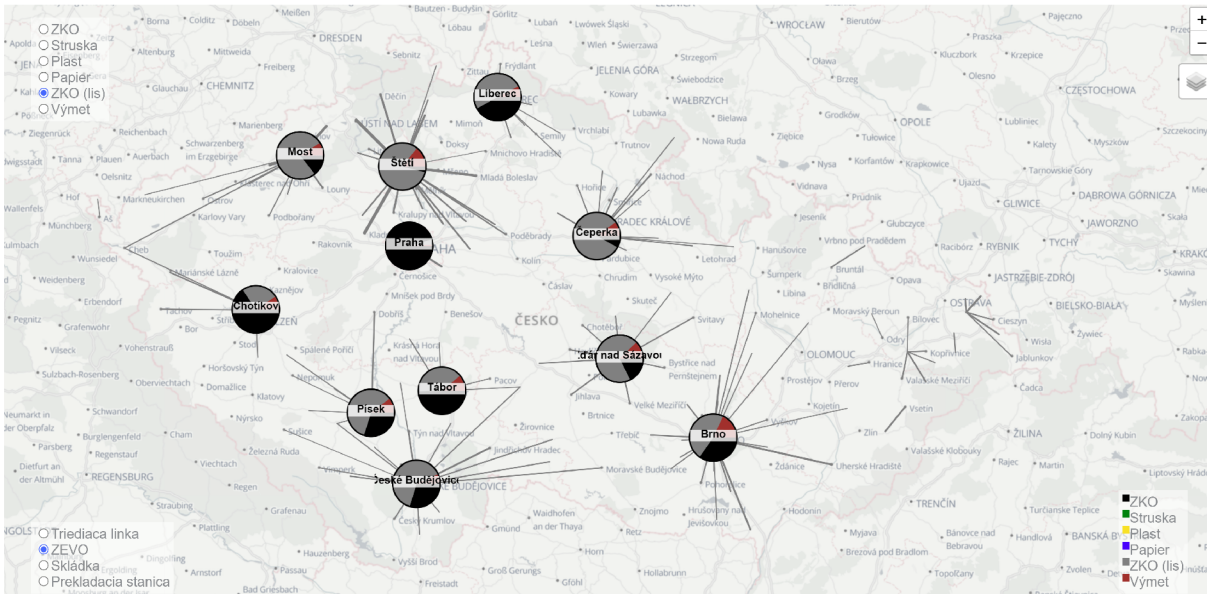
Ďalšou skúmanou frakciou je papier, ktorého toky sú znázornené na obr. 5.4. Podobne ako zmesový komunálny odpad, aj papier sa prevažne zväžá do triediacich liniek v bezprostrednej blízkosti od obce, v ktorej bol vyprodukovaný. Rozmiestnenie týchto spracovateľských zariadení je vyobrazené na mape na obr. D.1. Keďže papier a plast môžu od producenta púťovať jedine do triediacich liniek, ktoré nemajú kapacitné obmedzenie a ceny za spracovanie aj zvoz týchto dvoch frakcií sú rovnaké, mapa na tomto obrázku znázorňuje aj toky plastu.



Obr. 5.4: Toky plastu a papiera

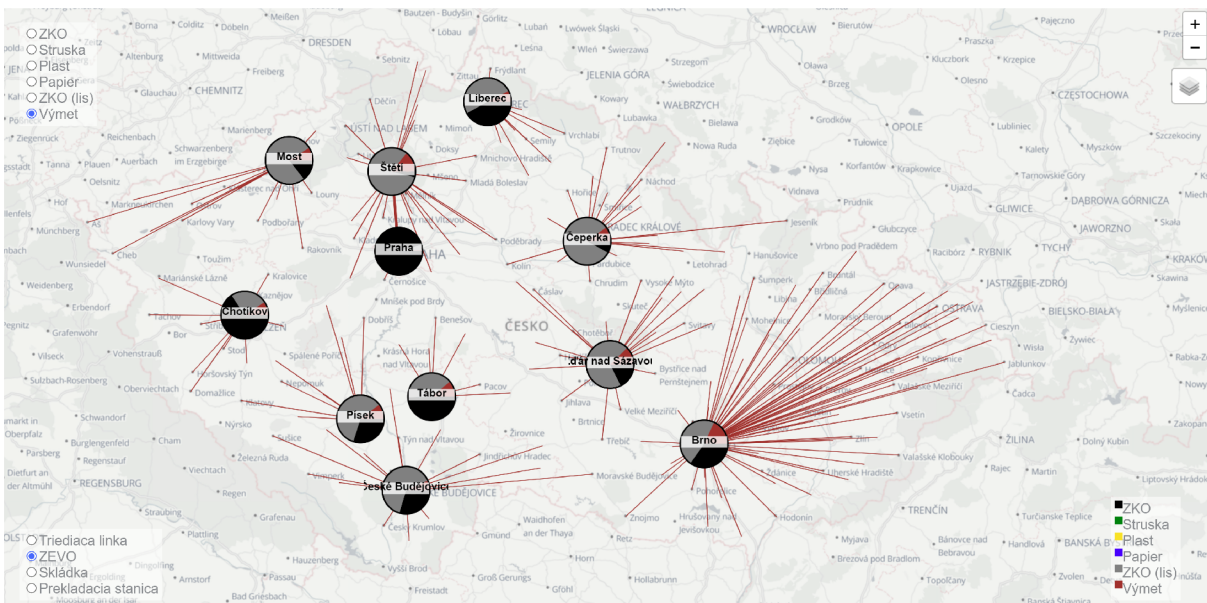
Nasledujúce dva obrázky zobrazujú toky spracovaného odpadu. Mapa na obr. 5.5 zachytáva tok zlisovaného zmesového odpadu z prekladacích staníc do koncových zariadení. Čím hrubšie sú hrany vedúce medzi týmito zariadeniami, tým väčšie množstvo odpadu je po týchto hranách prevezené. Koláčové grafy naznačujú, že zlisovaný odpad tvorí vo viacerých ZEVO po celej republike väčšinu energeticky využitého odpadu. Na tomto obrázku si môžeme všimnúť, že kvôli deficitu spalovní vo východnej časti krajiny je zlisovaný odpad v týchto miestach zväžaný na skládky. Dôvodom sú nižšie náklady na skládkovanie v porovnaní s cenou za prepravu do najbližšieho ZEVO, nachádzajúceho sa v Brne.

### 5.3 VÝSLEDKY



Obr. 5.5: Toky zlisovaného zmesového komunálneho odpadu

Posledný obrázok 5.6 predstavuje tok výmetu z triediacich línií do ZEVO. Na rozdiel od zlisovaného odpadu, výmet nemôže byť odvezený na skládky, preto je zo všetkých triediacich línií nachádzajúcich sa v severovýchode Českej republiky zvážaný do spaľovne v Brne.

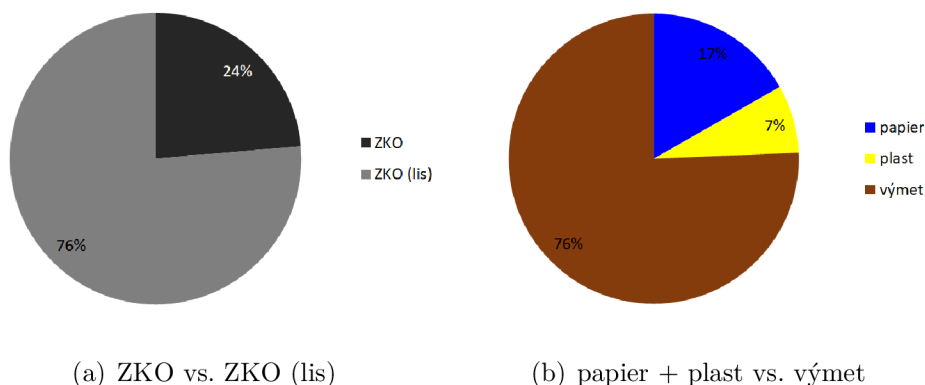


Obr. 5.6: Toky výmetu

Struska vyprodukovaná vo všetkých ZEVO je vozená na najbližšie skládky, preto je zbytočné vyobrazovať jej tok na mape. Na obr. D.3 je na koláčových grafoch možné vidieť, na ktorých skládkach toto reziduum, vyznačené zelenou farbou, končí.

Na koláčových grafoch na obr. 5.7 môžeme pozorovať pomer medzi spracovaným a nespracovaným odpadom vyjadrený pomocou súčinu vzdialeností všetkých dvojíc uzlov s množstvom odpadu, ktoré je medzi týmito uzlami prevezené. Z grafu 5.7(a) je badateľné,

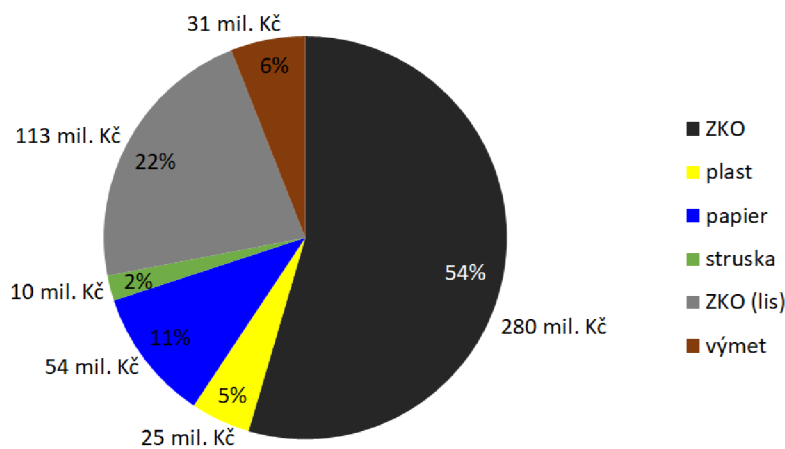
že hoci je zmesový komunálny odpad najviac produkovaným odpadom v celej krajine, tak je oproti zlisovanému zmesovému komunálnemu odpadu prevážaný na omnoho kratšie vzdialenosti. Dôvodom sú významne vyššie náklady na prepravu nespracovaného odpadu v smetiarskych vozidlách ako na prepravu spracovaného odpadu v nákladných vozidlách. Analógia platí aj pre papier a plast, ktoré sú dopravené do najbližších triediacich liniek, z ktorých výmet putuje do vzdialených ZEVO, viď 5.7(b).



Obr. 5.7: Pomer prepaveného množstva spracovaného a nespracovaného odpadu [km.t]

### 5.3.3 Náklady

Celkové ročné náklady na prevoz odpadu, jeho spracovanie a vystavanie nových spracovateľských zariadení sa vyšplhali na viac ako 15 miliárd Kč. Koláčový graf na obr. 5.8 predstavuje percentuálne zastúpenie všetkých typov odpadu v nákladoch na ich prevoz. Najväčšiu časť tohto grafu tvoria náklady na prepravu zmesového komunálneho odpadu, čo je zapríčinené tým, že tejto frakcie sa v Českej republike vyprodukuje najviac. Druhým dôvodom je cena za jeho prevoz, ktorá je 8× vyššia ako cena za prevoz jeho zlisovanej formy. Takmer štvrtinu nákladov tvorí preprava zlisovaného zmesového komunálneho odpadu, ktorý sa hromadne zväža do ZEVO, ktoré nie sú v bezprostrednej blízkosti všetkých prekladacích staníc.



Obr. 5.8: Ročné náklady na prevoz odpadu

Kľúčom k obmedzeniu skládkovania a postupnému prechodu na obehové hospodárstvo je materiálové a energetické využitie odpadu. Táto zmena so sebou prináša výrazné navýšenie nákladov pre obce. Výsledky prípadovej štúdie vykonanej na reálnych dátach zo zvozu v rámci Českej republiky naznačujú potrebu zvýšenia skládkovacieho poplatku na minimálne 5 450 Kč/t. Toto opatrenie podmieni výstavbu nových spracovateľských zariadení, ktoré pomôžu odbremeniť skládky od množstva odpadu, ktoré je na ne v súčasnosti ukladané. Jedným z krokov k zníženiu ekonomickej záťaže na obce je tvorba zväzkov, ktoré tieto obce združujú s cieľom komplexného plánovania pre zaistenie efektívnejšieho systému zberu a nakladania s odpadom. Mapy uverejnené v tejto kapitole by mohli napomôcť k optimálnemu spájaniu obcí do týchto združení.



# Záver

Cieľom tejto práce bolo zostrojenie komplexného optimalizačného modelu, ktorý opisuje tok odpadu v sieti tvorenej obcami a spracovateľskými zariadeniami, jeho implementácia v programovacom jazyku Julia a testovanie na rozsiahlej reálnej úlohe.

Prvé dve kapitoly tvorili matematický aparát celej práce. Prvá kapitola bola venovaná popisu teórie grafov, v ktorej sme sa oboznámili s pojmi ako sú orientovaný či ohodnotený graf, ich maticovými zápismi a algoritmami na hľadanie najkratšej cesty v grafe. Druhá kapitola sa zaoberala lineárnym programovaním a algoritmami riešiacimi tento typ úloh. Najväčší dôraz bol kladený na podmnožinu lineárneho programovania – zmiešané celočíselné programovanie, využívané v celom zvyšku práce. Na konci kapitoly boli uvedené dôvody výberu programovacieho jazyka Julia pri tvorbe tejto práce.

Ďalšie kapitoly predstavovali praktickú časť diplomovej práce. Tretia kapitola tvorila jadro práce, kde bola formulovaná riešená úloha a zostrojený základný model, ktorý bol postupne rozširovaný. Prvotný model, ktorý uvažoval len existujúce spracovateľské zariadenia, sa postupne rozrastal o potenciálne spracovateľské zariadenia popísané pomocou celočíselných premenných špeciálnych typov, reziduálne toky a novú formu dopravy. V prílohe B bol následne spísaný finálny model. V nasledujúcej kapitole bol zostrojený model testovaný a vyladený na fiktívnej úlohe skladajúcej sa z malého počtu uzlov a hrán. Získané výsledky tu boli graficky vyobrazené a taktiež slovne popísané. Súčasťou tejto kapitoly boli aj návrhy na zlepšenie modelu zostrojeného v kapitole tretej, obmedzujúce skládkovanie a tým pádom podporujúce výstavbu nových spracovateľských zariadení. Posledná kapitola sa venovala prípadovej štúdii uskutočnenej na dátach zo zvozu odpadu v Českej republike. Súčasťou kapitoly boli aj výpočetné problémy, na ktoré sme narazili pri práci s veľkým množstvom dát. Na konci kapitoly boli výsledky vykreslené na mapách pomocou vizualizačného softvéru, zostrojeného v rámci diplomovej práce [9].

Výsledky prípadovej štúdie naznačujú začiatok výstavby a väčšieho využitia ZEVO až pri nastavení skládkovacieho poplatku na 5 450 Kč/t. Táto hodnota je významne vyššia ako v pláne uverejnenom Ministerstvom životného prostredia Českej republiky. V skutočnosti bude nutné nájsť kompromis medzi udržaním cenovej hladiny na prípustnej hodnote a splnením bodov zakotvených v smerniciach, ktoré sú súčasťou balíčka k obehovému hospodárstvu.

Výstupom z tejto práce je funkčný model popisujúci zvoz a spracovanie odpadu implementovaný v programovacom jazyku Julia, ktorý môže byť v budúcnosti využitý pri rozhodovaní o výstavbe nových spracovateľských zariadení, podporujúcich prechod na obehové hospodárstvo. Tento model je možné ďalej rozvíjať napr. obmedzením na toky odpadu medzi krajinami, vynútením toku po niektorých hranách podľa reálnej situácie a zmlúv alebo zmenou uhlu pohľadu účelovej funkcie, ktorú môžeme uvažovať z pohľadu investora, konkrétneho producenta odpadu alebo koncového zariadenia. K ekonomickému pohľadu na celú situáciu môže byť pridaný taktiež environmentálny pohľad v podobe obmedzení alebo úpravou účelovej funkcie.

# Literatúra

- [1] AMIN, Saman Hassanzadeh a Guoqing ZHANG. A multi-objective facility location model for closed-loop supply chain network under uncertain demand and return. *Applied Mathematical Modelling* [online]. 2013, **37**(6), 4165-4176 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.apm.2012.09.039>
- [2] ANAND, Rimmi, Divya AGGARWAL a Vijay KUMAR. A comparative analysis of optimization solvers. *Journal of Statistics and Management Systems*. 2017, **20**(4), 623-635. Dostupné z: <http://dx.doi.org/10.1080/09720510.2017.1395182>
- [3] BERGE, Claude. *The Theory of Graphs*. Dover Publications, 2001. ISBN 0-486-41975-4.
- [4] BEZANSON, Jeff, Alan EDELMAN, Stefan KARPINSKI a Viral B SHAH. Julia: A fresh approach to numerical computing. *SIAM Review*. SIAM, 2017, **59**(1), 65-98. Dostupné z: <https://epubs.siam.org/doi/10.1137/141000671>
- [5] BOCCIA, Maurizio, Teodor Gabriel CRAINIC, Antonio SFORZA a Claudio STERLE. Multi-commodity location-routing: Flow intercepting formulation and branch-and-cut algorithm. *Computers & Operations Research* [online]. 2018, **89**, 94-112 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.cor.2017.08.013>
- [6] COTTLE, Richard W. a Mukund N. THAPA. *Linear and Nonlinear Optimization*. 253. New York, NY: Springer New York, 2017. ISBN 1493970534.
- [7] ČADA, Roman, Tomáš KAISER a Zdeněk RYJÁČEK. *Diskrétní matematika*. Plzeň: Západočeská univerzita v Plzni, 2004, 170 s. ISBN 80-7082-939-7.
- [8] ČESKO. Zákon č.541/2020 Sb. o odpadech. In: *Sbírka zákonů České republiky*. Dostupné z: <https://www.zakonyprolidi.cz/cs/2020-541>
- [9] ČERVENKA, Roman. *Webová aplikace pro vyhodnocování investičních záměrů v odpadovém hospodářství*. Brno, 2023. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automatizace a informatiky. Vedoucí práce Ing. Ladislav Dobrovský.
- [10] ČESKÝ STATISTICKÝ ÚŘAD [ČSÚ]. *Základní charakteristika SO ORP* [online]. [cit. 2023-5-17]. Dostupné z: [https://vdb.czso.cz/vdbvo2/faces/cs/index.jsf?page=vystup-objekt&pvo=RS007DORP&z=T&f=TABULKA&katalog=32895&c=v467~3\\_\\_RP2020#w=](https://vdb.czso.cz/vdbvo2/faces/cs/index.jsf?page=vystup-objekt&pvo=RS007DORP&z=T&f=TABULKA&katalog=32895&c=v467~3__RP2020#w=)
- [11] DEMEL, Jiří. *Grafy a jejich aplikace*. Praha: Academia, 2002, 257 s. ISBN 80-200-0990-6.
- [12] DUNNING, Iain, Joey HUCHETTE a Miles LUBIN. JuMP: A Modeling Language for Mathematical Optimization. *SIAM Review*. 2017, **59**(2), 295-320. Dostupné z: <https://epubs.siam.org/doi/10.1137/15M1020575>

- [13] GIUGLIANO, Michele, Stefano CERNUSCHI, Mario GROSSO a Lucia RIGAMONTI. Material and energy recovery in integrated waste management systems. An evaluation based on life cycle assessment. *Waste Management* [online]. 2011, **31**(9), 2092-2101 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.wasman.2011.02.029>
- [14] GRIVA, Igor, Stephen G. NASH a Ariela SOFER. *Linear and nonlinear optimization*. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics, 2009, xxii, 742 s. : il. ISBN 978-0-89871-661-0.
- [15] HLINĚNÝ, Petr. *Základy teorie grafů* [online]. 1. vyd. Brno: Masarykova univerzita, 2010 [cit. 2023-02-08]. Elportál. Dostupné z: <http://is.muni.cz/elportal/?id=878389>. ISSN 1802-128X.
- [16] *ILOG CPLEX Optimization Studio - IBM ILOG CPLEX Optimizer* [online]. US: IBM Corporation, 2023 [cit. 2023-02-24]. Dostupné z: <https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer>
- [17] MAJER, Tomáš. *Problémy trasovania v rozsiahlych dopravných sieťach* [online]. Dizertačná práca. Žilina, 2013 [cit. 2023-03-17]. Dizertačná práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky. Dostupné z: [https://www.fri.uniza.sk/uploads/phd/14f-Majer\\_praca.pdf](https://www.fri.uniza.sk/uploads/phd/14f-Majer_praca.pdf).
- [18] PLUSKAL, Jaroslav, Radovan ŠOMPLÁK, Vlastimír NEVRLÝ, Veronika SMEJKALOVÁ a Martin PAVLAS. Strategic decisions leading to sustainable waste management: Separation, sorting and recycling possibilities. *Journal of Cleaner Production* [online]. 2021, **278**, 123359 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.jclepro.2020.123359>
- [19] PUCHINGER, Jakob a Günther R. RAIDL. Combining Metaheuristics and Exact Algorithms in Combinatorial Optimization: A Survey and Classification. In: *ARTIFICIAL INTELLIGENCE AND KNOWLEDGE ENGINEERING APPLICATIONS: A BIOINSPIRED APPROACH, PT 2, PROCEEDINGS* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, s. 41-53 [cit. 2023-01-25]. ISBN 9783540263197. ISSN 0302-9743. DOI [http://dx.doi.org/10.1007/11499305\\_5](http://dx.doi.org/10.1007/11499305_5).
- [20] RYJÁČEK, Z. *Teorie grafů a diskrétní optimalizace 2*. Učební text Západočeské univerzity, 2001. Dostupné z: <https://zcu.arcao.com/kma/tgd1/TGD1-skripta.pdf>.
- [21] Směrnice Evropského parlamentu a Rady (EU) 2018/850 ze dne 30. května 2018, kterou se mění směrnice 1999/31/ES o skládkách odpadů. In: *Úřední věstník*. Dostupné z: <https://eur-lex.europa.eu/legal-content/CS/TXT/?uri=celex:32018L0850>
- [22] Směrnice Evropského parlamentu a Rady (EU) 2018/851 ze dne 30. května 2018, kterou se mění směrnice 2008/98/ES o odpadech. In: *Úřední věstník*. Dostupné z: <https://eur-lex.europa.eu/legal-content/CS/TXT/?uri=CELEX%3A32018L0851>

- [23] SON, Le Hoang a Amal LOUATI. Modeling municipal solid waste collection: A generalized vehicle routing model with multiple transfer stations, gather sites and inhomogeneous vehicles in time windows. *Waste Management* [online]. 2016, **52**, 34-49 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.wasman.2016.03.041>
- [24] WILLIAMS, H. P. *Model building in mathematical programming*. Fifth edition. Chichester: Wiley, 2013, xx, 411 stran. ISBN 978-1-118-44333-0.
- [25] WOLSEY, Laurence A. *Integer programming*. New York: Wiley, 1998, xviii, 264 stran : ilustrace. ISBN 0-471-28366-9.
- [26] YADAV, Vinay, Subhankar KARMAKAR, A.K. DIKSHIT a A.K. BHURJEE. Interval-valued facility location model: An appraisal of municipal solid waste management system. *Journal of Cleaner Production* [online]. 2018, **171**, 250-263 [cit. 2023-05-15]. Dostupné z: <https://doi.org/10.1016/j.jclepro.2017.09.233>

# Zoznam skratiek

AIMMS	Advanced Interactive Multidimensional Modeling System
AMPL	A Mathematical Programming Language
GAMS	The General Algebraic Modeling System
CHKO	chránená krajinná oblasť
JuMP	Julia for Mathematical Programming
ORP	obec s rozšírenou pôsobnosťou
SOS	Special Ordered Set
TAP	tuhé alternatívne palivá
VBA	Visual Basic for Applications
ZEVO	zariadenie na energetické využitie odpadov
ZKO	zmesový komunálny odpad

# Prílohy

## A Prevod primárneho problému na duálny

PRIMÁRNA ÚLOHA		DUÁLNA ÚLOHA	
účelová funkcia	minimalizácia	maximalizácia	účelová funkcia
účelová funkcia	koefficienty	konštanty	pravá strana
pravá strana	konštanty	koefficienty	účelová funkcia
obmedzenie	$a_{i1}x_1 + \dots + a_{in}x_n \geq b_i$	$y_i \geq 0$	premenná
obmedzenie	$a_{i1}x_1 + \dots + a_{in}x_n \leq b_i$	$y_i \leq 0$	premenná
obmedzenie	$a_{i1}x_1 + \dots + a_{in}x_n = b_i$	$y_i$ voľná	premenná
premenná	$x_j \geq 0$	$y_1a_{1j} + \dots + y_ma_{mj} \leq c_j$	obmedzenie
premenná	$x_j \leq 0$	$y_1a_{1j} + \dots + y_ma_{mj} \geq c_j$	obmedzenie
premenná	$x_j$ voľná	$y_1a_{1j} + \dots + y_ma_{mj} = c_j$	obmedzenie

Tabuľka A.1: Prevod primárneho problému na duálny

## B Kompletný matematický model

### Označenie

#### Množiny:

$i \in I$	množina všetkých uzlov
$i \in I^{ex} \subset I$	množina uzlov predstavujúcich obce a existujúce zariadenia (spaľovne, prekladacie stanice, triediace linky, skládky) v nich
$i \in I^n \subset I$	množina uzlov reprezentujúcich prekladacie stanice, v ktorých je možné nakladať zlisovaný odpad na nákladné vlaky
$i \in I^p \subset I$	množina uzlov predstavujúcich potenciálne spracovateľské zariadenia (spaľovne, prekladacie stanice, triediace linky)
$i \in I^s \subset I$	množina uzlov reprezentujúcich skládky
$j \in J$	množina všetkých hrán
$j \in J^x \subset J$	množina hrán medzi obcami navzájom a medzi obcami a zariadeniami v nich
$j \in J^y \subset J$	množina hrán spájajúcich zariadenia navzájom
$o \in O$	množina typov odpadu
$o \in O^l \subset O$	množina zlisovaných typov odpadu (zlisovaný zmesový komunálny odpad, zlisovaný plast)
$o \in O^n \subset O$	množina nespracovaných frakcií odpadu (zmesový komunálny odpad, plasty, sklo, papier, biologicky rozložiteľný odpad)
$o \in O^s \subset O$	množina spracovaných typov odpadu (struska, zlisovaný odpad, výmet)
$r \in R$	množina trás v železničnej dopravnej sieti
$t \in T$	množina kapacít potenciálnych spracovateľských zariadení

#### Vstupné parametre:

$B_{i,\tilde{o},o}$	matica, ktorej prvky predstavujú podiel pôvodnej hmotnosti nespracovaného odpadu $\tilde{o}$ , z ktorého vznikne reziduum $o$ v uzle $i$
$M_{i,j}$	incidenčná matica pre prepravu odpadu medzi obcami navzájom a medzi obcami a zariadeniami v nich
$N_{i,j}$	incidenčná matica pre reziduálne toky medzi zariadeniami
$Z_{r,i,o}$	matica pre železničnú dopravnú sieť, vid' (3.1)
$c_{i,o}^{ex}$	cena za spracovanie tony odpadu $o$ v existujúcom zariadení v uzle $i$ [Kč/t]

$c_i^p$	cena za výstavbu potenciálneho spracovateľského zariadenia v uzle $i$ [Kč]
$c_o^{ph}$	náklady na pohonné hmoty pre smetiarske vozidlá prepravujúce odpad $o$ [Kč/t.km]
$c_o^{ph,s}$	náklady na pohonné hmoty pre vozidlá prepravujúce spracovaný odpad $o$ [Kč/t.km]
$d_j$	dĺžka hrany $j$ [km]
$e_r$	maximálny možný počet železničných cyklov $r$ počas jedného roka [-]
$k_i^{ex}$	kapacita existujúceho zariadenia v uzle $i$ [t]
$k_t^p$	možné kapacity potenciálnych spracovateľských zariadení [t]
$k_{i,o}^u$	kapacita odpadu $o$ v uzle $i$ (povolenie spracovania odpadu $o$ v uzle $i$ ) [t]
$k_r^z$	kapacita jedného nákladného vlaku absolvujúceho trasu $r$ [t]
$p_{i,o}$	koeficient pomeru odpadu $o$ v zariadení v uzle $i$ (vyjadruje, koľkokrát sa odpad $o$ započítava do kapacity spracovateľského zariadenia) [-]
$w_{i,o}$	ročná produkcia odpadu $o$ v uzle $i$ [t]

### Premenné:

$r_{i,o}$	reziduum $o$ (struska, zlisovaný odpad, výmet) vzniknuté v uzle $i$ [t]
$x_{j,o}$	množstvo odpadu $o$ (zmesový komunálny odpad, plasty, sklo, papier, biologicky rozložiteľný odpad) prevezeného hranou $j$ medzi obcami navzájom a medzi obcami a zariadeniami v nich [t]
$y_{j,o}$	množstvo odpadu $o$ (struska, zlisovaný odpad, výmet) prevezeného hranou $j$ medzi zariadeniami [t]
$z_{i,o}$	množstvo odpadu $o$ (zlisovaný odpad) naloženého alebo vyloženého z nákladného vlaku v uzle $i$ [t]
$\gamma_r$	počet absolvovaní trasy $r$ počas jedného roka – celočíselná premenná [-]
$\delta_{i,t}$	rozhodnutie o výstavbe spracovateľského zariadenia (spaľovne, prekladacej stanice, triediacej linky) s kapacitou $t$ v uzle $i$ – binárna SOS1 premenná [-]



## Model

$$\begin{aligned}
\min \quad & \sum_{i \in I^{ex}} \sum_{j \in J^x} \sum_{o \in O} M_{i,j} x_{j,o} c_{i,o}^{ex} + \sum_{i \in I^{ex}} \sum_{j \in J^y} \sum_{o \in O} N_{i,j} y_{j,o} c_{i,o}^{ex} + \sum_{i \in I^p} \sum_{t \in T} k_t^p \delta_{i,t} c_i^p + \\
& + \sum_{j \in J^x} \sum_{o \in O} x_{j,o} d_j c^{ph} + \sum_{j \in J^y} \sum_{o \in O} y_{j,o} d_j c^{ph,s} + \sum_{r \in R} \gamma_r c_r^z \\
\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} & \leq k_{i,o}^u, \quad \forall i \in I, \quad \forall o \in O \\
\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \right) & \leq k_i^{ex}, \quad \forall i \in I^{ex} \\
\sum_{o \in O} p_{i,o} \left( \sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} \right) & \leq \sum_{t \in T} k_t^p \delta_{i,t}, \quad \forall i \in I^p \\
\sum_{\tilde{o} \in O} \left( \sum_{j \in J^x} M_{i,j} x_{j,\tilde{o}} + \sum_{j \in J^y} N_{i,j} y_{j,\tilde{o}} + \sum_{r \in R} Z_{r,i,\tilde{o}} z_{i,\tilde{o}} \right) B_{i,\tilde{o},o} & = r_{i,o}, \quad \forall i \in I, \quad \forall o \in O \\
\sum_{i \in I} Z_{r,i,o} z_{i,o} & = 0, \quad \forall r \in R, \quad \forall o \in O^l \\
0 \leq - \sum_{i \in I^n} \sum_{o \in O^l} Z_{r,i,o} z_{i,o} & \leq k_r^z \gamma_r, \quad \forall r \in R \\
\gamma_r & \leq e_r, \quad \forall r \in R \\
\sum_{j \in J^x} M_{i,j} x_{j,o} + \sum_{j \in J^y} N_{i,j} y_{j,o} + \sum_{r \in R} Z_{r,i,o} z_{i,o} + w_{i,o} + r_{i,o} & \geq 0, \quad \forall i \in I, \quad \forall o \in O \\
x_{j,o} & \geq 0, \quad \forall j \in J^x, \quad \forall o \in O^n \\
x_{j,o} & = 0, \quad \forall j \in J^x, \quad \forall o \in O^s \\
y_{j,o} & \geq 0, \quad \forall j \in J^y, \quad \forall o \in O^s \\
y_{j,o} & = 0, \quad \forall j \in J^y, \quad \forall o \in O^n \\
z_{i,o} & \geq 0, \quad \forall i \in I, \quad \forall o \in O \\
r_{i,o} & \geq 0, \quad \forall i \in I, \quad \forall t \in T \\
\delta_{i,t} & \in \{0, 1\}, \quad \forall i \in I^p, \quad \forall t \in T \\
\sum_{t \in T} \delta_{i,t} & \leq 1, \quad \forall i \in I^p \\
\gamma_r & \in \mathbb{Z}_0^+, \quad \forall r \in R
\end{aligned}$$

## **C Zoznam ORP s novými prekladacími stanicami**

### **Kapacita 10 kt**

Aš, Bílina, Bílovec, Blatná, Blovice, Broumov, Bučovice, Bystřice nad Pernštejnem, Český Krumlov, Český Těšín, Dačice, Dobříš, Dvůr Králové nad Labem, Frenštát pod Radhoštěm, Frýdlant, Frýdlant nad Ostravicí, Havlíčkův Brod, Hlinsko, Holešov, Holice, Horažďovice, Hořice, Hranice, Humpolec, Hustopeče, Chotěboř, Ivančice, Jablunkov, Jaroměř, Jihlava, Jilemnice, Kaplice, Konice, Kopřivnice, Kostelec nad Orlicí, Králíky, Kraslice, Krnov, Kuřim, Litoměřice, Litomyšl, Lovosice, Mariánské Lázně, Mělník, Mikulov, Milevsko, Mnichovo Hradiště, Mohelnice, Moravské Budějovice, Moravský Krumlov, Náměšť nad Oslavou, Nepomuk, Nová Paka, Nové Město na Moravě, Nové Město nad Metují, Nový Bor, Nový Bydžov, Nymburk, Odry, Orlová, Ostrov, Pacov, Pelhřimov, Podbořany, Poděbrady, Pohořelice, Prachatice, Přeštice, Rosice, Roudnice nad Labem, Sedlčany, Semily, Slavkov u Brna, Soběslav, Sušice, Svitavy, Šlapanice, Šternberk, Tachov, Tanvald, Telč, Tišnov, Trhové Sviny, Třeboň, Turnov, Týn nad Vltavou, Valašské Meziříčí, Varnsdorf, Velké Meziříčí, Vimperk, Vodňany, Votice, Vyškov, Zábřeh, Žamberk, Žatec, Železný Brod, Židlochovice

### **Kapacita 20 kt**

Blansko, Boskovice, Brandýs nad Labem-Stará Boleslav, Břeclav, Česká Lípa, Český Brod, Děčín, Cheb, Jablonec nad Nisou, Jindřichův Hradec, Kadaň, Karviná, Kralupy nad Vltavou, Litvínov, Louny, Lysá nad Labem, Náchod, Neratovice, Rožnov pod Radhoštěm, Říčany, Slaný, Strakonice, Třinec, Uherské Hradiště, Vsetín

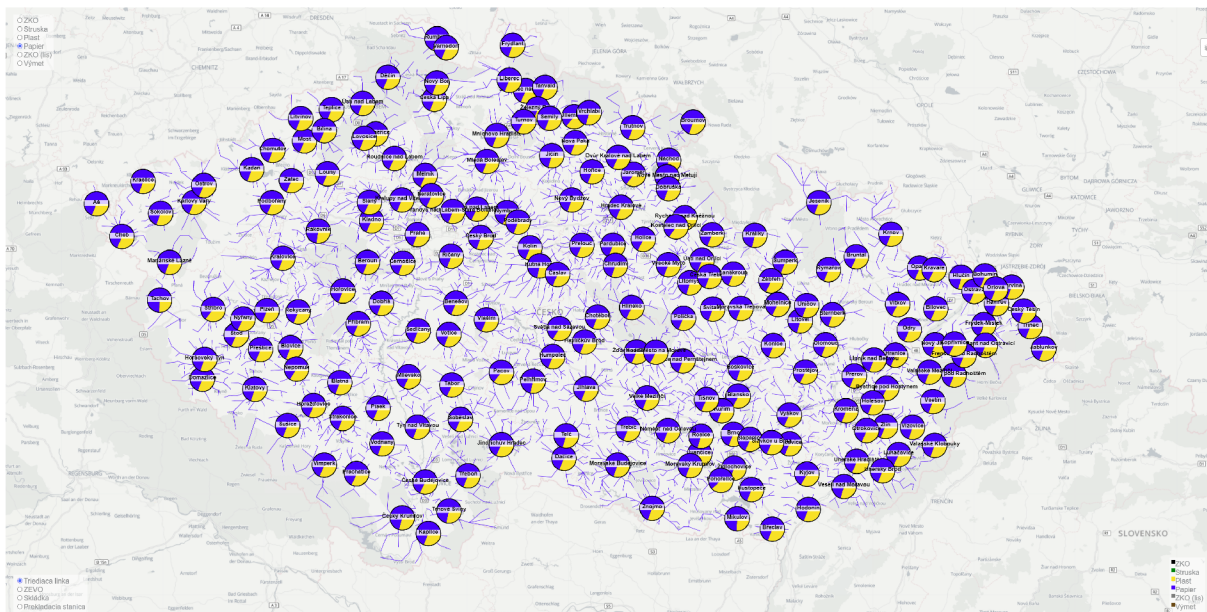
### **Kapacita 30 kt**

Hradec Králové, Chomutov, Mladá Boleslav, Pardubice, Teplice

### **Kapacita 40 kt**

Kladno, Ústí nad Labem

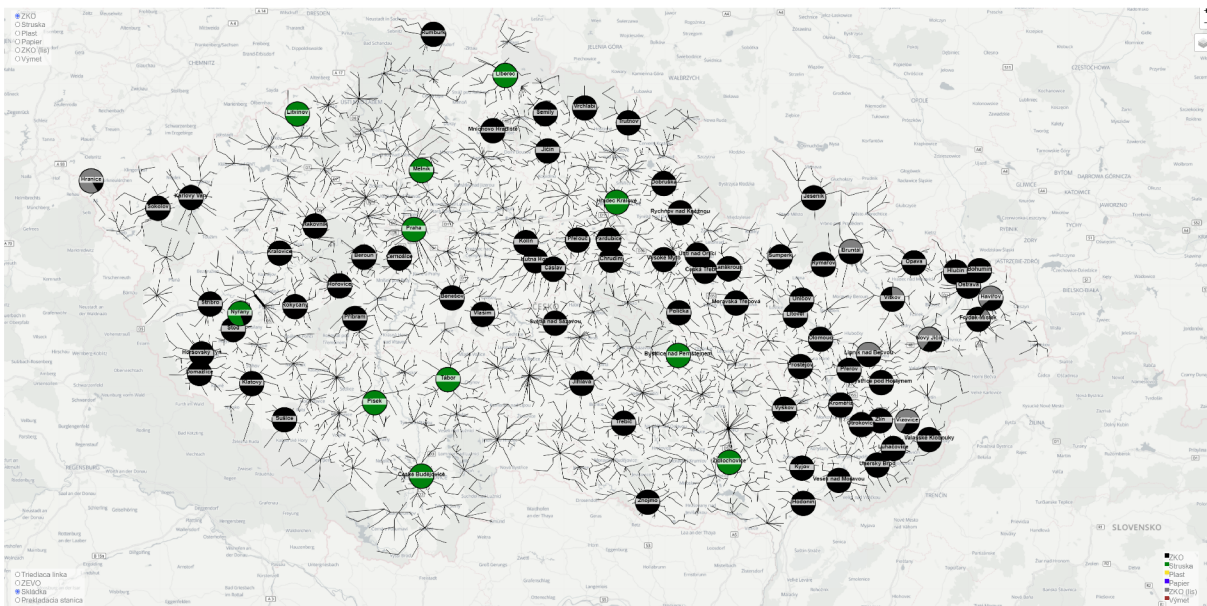
## D Mapy spracovateľských zariadení



Obr. D.1: Mapa triediacich liniek



Obr. D.2: Mapa prekladacích staníc



Obr. D.3: Mapa skládok