ŽILINSKÁ UNIVERZITA V ŽILINE

AUTOREFERÁT
DIZERTAČNEJ PRÁCE

Žilina, apríl 2019

RNDr. Zuzana Borčinová

Žilinská univerzita v Žiline
Fakulta riadenia a informatiky

RNDr. Zuzana Borčinová

Autoreferát dizertačnej práce

ROBUSTNÉ MODELY V DISTRIBUČNÝCH ÚLOHÁCH

na získanie titulu **"Philosophiae doctor"v skratke PhD.)**
v študijnom programe doktorandského štúdia
aplikovaná informatika

v študijnom odbore:
9.2.9 aplikovaná informatika

Žilina, apríl 2019

Dizertačná práca bola vypracovaná v externej forme doktorandského štúdia **na Katedre matematických metód a operačnej analýzy, Fakulte riadenia a informatiky Žilinskej univerzity v Žiline.**

Predkladateľ: **RNDr. Zuzana Borčinová**
Žilinská univerzita v Žiline
Fakulta riadenia a informatiky
Katedra matematických metód a operačnej analýzy

Školiteľ: **doc. RNDr. Štefan Peško, CSc.**
Žilinská univerzita v Žiline
Fakulta riadenia a informatiky
Katedra matematických metód a operačnej analýzy

Oponenti:

Autoreferát bol rozoslaný dňa: ....

Obhajoba dizertačnej práce sa koná dňa .... o .... h. pred komisiou pre obhajobu dizertačnej práce schválenou odborovou komisiou v študijnom odbore **9.2.9 Aplikovaná informatika, v študijnom programe Aplikovaná informatika** vymenovanou dekanom Fakulty riadenia a informatiky Žilinskej univerzity v Žiline dňa

....

prof. Ing. Karol Matiaško, PhD.
predseda odborovej komisie
študijného programu Aplikovaná informatika
v študijnom odbore 9.2.9 Aplikovaná informatika

Fakulta riadenia a informatiky
Žilinská univerzita v Žiline
Univerzitná 8215/1
010 26 Žilina

1 ÚVOD

Názvom *distribučné úlohy* označujeme úlohy súvisiace s riadením prepravy tovaru, prípadne osôb a distribúcie služieb od dodávateľov k odberateľom s cieľom minimalizovať náklady. Patria k nim napríklad dopravné úlohy, ktoré riešia zostavenie optimálneho plánu rozvozu, resp. zvozu komodít, lokačné úlohy zaoberajúce sa optimálnym rozmiestnením obslužných stredísk, priraďovacie úlohy a podobne. Z dôvodu neustále rastúcej potreby šetriť palivo a čas, prípadne znižovať emisie sú tieto úlohy predmetom intenzívneho štúdia v oblasti operačného výskumu a matematického programovania.

Pri formulácii distribučných úloh sú dodávatelia definovaní svojimi kapacitami a odberatelia svojimi požiadavkami, pričom sa predpokladá, že všetky vstupné údaje sú pevne dané (*deterministické*). V praxi, však, nie sú tieto predpoklady zaručené, t. j. niektoré z údajov majú charakter náhodných veličín (*stochastické*). Na riešenie stochastických optimalizačných úloh sa používa stochastické modelovanie, ktoré súvisí s vytváraním a riešením stochastických modelov, zodpovedajúcich reálnej situácii s určitou pravdepodobnosťou, spolu s možnosťou robiť simulácie. Pomocou stochastického modelu sa dajú simulovať rozličné scenáre s rôznymi parametrami a pozorovať správanie modelu. Na základe analýzy chovania modelu je potom možné odvodiť informácie potrebné pre rozhodovanie. Pri tvorbe analytického stochastického modelu sa vždy vychádza z predpokladu, že pravdepodobnosť (rozdelenie, charakteristiky atď.), ktorou sa riadia náhodné parametre, je známa alebo sa dá presne odhadnúť. V mnohých prípadoch je ťažké a niekedy aj nemožné presne odhadnúť rozdelenie pravdepodobnosti neistých vstupných údajov. Chybný odhad alebo nesprávna voľba pravdepodobnostného rozdelenia náhodných premenných môže viesť k chybám v stochastickom modeli. Táto skutočnosť je motiváciou k použitiu *robustných optimalizačných metód*, ktoré sú imúnne voči neurčitosti vstupných dát, čiže pri malých zmenách vstupných parametrov sa zachová výstup v blízkosti pôvodného výstupu a ani pri veľkých zmenách na vstupe nedôjde ku dramatickým zmenám, či dokonca katastrofálnym dôsledkom na výstupe. Pri robustnej optimalizácii je neistota modelovaná ako množina spojitých alebo diskrétnych hodnôt parametrov nazývaných scenáre. Hlavná myšlienka je vykonať optimalizáciu nad súborom scenárov a poskytnúť robustné riešenie, ktoré je kvalitné bez ohľadu na scenár (t. j. aj pri najhoršom možnom scenári).

Medzi najviac skúmané distribučné úlohy patrí *okružná dopravná úloha* – Vehicle Routing Problem (VRP). Úlohou VRP je naplánovať optimálne okružné trasy pre flotilu vozidiel, sústredených v centrálnom depe, aby obslúžili skupinu zákazníkov, pričom celkové prepravné náklady budú minimálne. K tejto úlohe sa v praktických aplikáciách často pridávajú ďalšie obmedzenia. Napríklad obsluha sa môže týkať súčasne rozvozu aj zberu komodít, pričom náklad vozidla po celej trase nesmie prekročiť jeho kapacitu, celková dĺžka trasy nesmie byť väčšia než predpísaný limit, zákazníci musia byť obslúžení v určených časových oknách, vozidlá flotily môžu byť heterogénne atď. Niektoré charakteristiky, ako sú požiadavky alebo časy jázd, sa môžu dynamicky meniť. Nový trend riešenia VRP spočíva v zohľadnení neurčitosti vstupných parametrov, pričom sa hľadá robustné riešenie – Robust Vehicle Routing Problem (RVRP). Výskum sa zameriava na okružné dopravné úlohy s neistými požiadavkami, s neistými zákazníkmi, s časovými oknami a s neistými časmi jázd a podobne. Táto práca je zameraná na *kapacitnú okružnú dopravnú úlohu* – Capacitated Vehicle Routing Problem (CVRP) a riešenie jej robustnej verzie *robustnej CVRP s neistými požiadavkami*.

1.1 Definícia kapacitnej okružnej dopravnej úlohy

CVRP môžeme definovať ako problém teórie grafov nasledovne [84]. Nech $\vec{G} = (V, \vec{H}, c)$ je úplný hranovo ohodnotený digraf, kde $V = \{0, 1, 2, \ldots, n\}$ je množina vrcholov a $\vec{H} = \{(i, j), \forall i, j \in V, i \neq j\}$ je množina orientovaných hrán. Vrcholy $i = 1, 2, \ldots, n$ reprezentujú zákazníkov s nezápornými požiadavkami d_i a vrchol 0 predstavuje depo. Každá hrana $(i, j) \in \vec{H}$ je ohodnotená nezápornou cenou c_{ij}, ktorá vyjadruje prepravné náklady medzi vrcholmi i a j. Budeme predpokladať, že matica ohodnotení hrán je symetrická, t. j. že platí $c_{ij} = c_{ji}$ pre všetky $i, j \in V$ a spĺňa trojuholníkovú nerovnosť $c_{ij} + c_{jk} \leq c_{ik}$ pre všetky $i, j, k \in V$. V depe je k dispozícii p homogénnych vozidiel, každé s kapacitou Q. Predpokladáme, že $p \geq p_{min}$, kde p_{min} je minimálny počet vozidiel potrebných na uspokojenie požiadaviek všetkých zákazníkov. Hodnota p_{min} sa dá určiť vyriešením *úlohy o batohu* – Bin Packing Problem (BPP) korešpondujúcej s CVRP. Vzhľadom na to, že BPP je NP-ťažký problém, často sa namiesto p_{min} používa dolná hranica $\left\lceil \frac{\sum_{i=1}^{n} d_i}{Q} \right\rceil$. Navyše predpokladáme, že $d_i \leq Q$ pre každé $i = 1, 2, \ldots, n$.

Cieľom CVRP je nájsť p cyklov, reprezentujúcich okružné trasy vozidiel, s minimálnou cenou, ktorá je definovaná ako súčet cien hrán patriacich cyklom, pričom:

1) každá trasa musí začínať a končiť v depe,
2) každý zákazník bude navštívený práve raz,
3) na žiadnej trase nebude prekročená kapacita vozidla.

1.2 Matematický model kapacitnej okružnej dopravnej úlohy

Nasledujúci matematický model CVRP je založený na tzv. *tokovej formulácii* – vehicle flow formulation, ktorá pre každú hranu $(i,j) \in \vec{H}$ používa bivalenú premennú x_{rij}, indikujúcu, či vozidlo r, $r \in \{1,2,\dots,p\}$ v optimálnom riešení prechádza hranou (i,j) alebo nie, t. j.

$$x_{rij} = \begin{cases} 1, & \text{ak trasa } r \text{ v optimálnom riešení obsahuje hranu } (i,j), \\ 0, & \text{inak.} \end{cases} \tag{1}$$

CVRP formulujeme ako nasledujúcu úlohu bivalentného lineárneho programovania [45]:

Model CVRP 1

Minimalizujte

$$\sum_{r=1}^{p} \sum_{i=0}^{n} \sum_{\substack{j=0 \\ i\neq j}}^{n} c_{ij} x_{rij}, \tag{2}$$

za podmienok

$$\sum_{r=1}^{p} \sum_{\substack{i=0 \\ i\neq j}}^{n} x_{rij} = 1, \qquad \forall j \in \{1,\dots,n\}, \tag{3}$$

$$\sum_{j=1}^{n} x_{r0j} = 1, \qquad \forall r \in \{1,\dots,p\}, \tag{4}$$

$$\sum_{\substack{i=0 \\ i\neq j}}^{n} x_{rij} = \sum_{i=0}^{n} x_{rji}, \qquad \forall j \in \{0,\dots,n\},\ r \in \{1,\dots,p\}, \tag{5}$$

$$\sum_{i=0}^{n} \sum_{\substack{j=1 \\ i\neq j}}^{n} d_j x_{rij} \le Q, \qquad \forall r \in \{1,\dots,p\}, \tag{6}$$

$$\sum_{r=1}^{p} \sum_{i\in S} \sum_{\substack{j\in S \\ i\neq j}} x_{rij} \le |S| - 1, \qquad \forall S \subseteq \{1,\dots,n\}, \tag{7}$$

$$x_{rij} \in \{0,1\}, \qquad \forall r \in \{1,\dots,p\},\ i,j \in \{0,\dots,n\}, i \neq j. \tag{8}$$

Účelová funkcia (2) minimalizuje celkové prepravné náklady. Obmedzujúce podmienky (3) zaručujú, že každý zákazník bude obslúžený práve jedným vozidlom. Podmienky toku (4) a (5) zaisťujú, že každé vozidlo opustí depo jediný raz a že počet vozidiel prichádzajúcich ku každému zákazníkovi a do depa je rovný počtu vozidiel odchádzajúcich. Kapacitné podmienky (6) garantujú, že na žiadnej trase súčet požiadaviek všetkých zákazníkov neprekročí kapacitu obslužného vozidla. Podmienky (7) sú anticyklické podmienky a zabezpečujú, že v riešení nebudú cykly, ktoré neprechádzajú depom. Posledné sú obligatórne podmienky (8) určujúce definičný obor premenných.

1.3 Všeobecná definícia robustnej optimalizácie

Pojem *robustný* má pri optimalizácii viacero významov. V praxi niekedy bývajú vstupné údaje neisté, nepresné alebo sa menia v čase, čo môže ovplyvniť optimálne riešenie nájdené pre aktuálne hodnoty parametrov tak, že sa stane suboptimálnym alebo dokonca neprípustným. *Robustné riešenie* je riešenie odolné voči odchýlkam vstupných dát.

Pri riešení optimalizačných problémov s neistými dátami sa používajú dva prístupy – stochastická a robustná optimalizácia. Dôležitým predpokladom pri *stochastickej optimalizácii* (SO) je, že rozdelenie pravdepodobnosti neistých údajov musí byť známe. Stochastické programovanie zaviedol Dantzig (1955) [20], ktorý použil dve metódy na riešenie stochastických problémov: programovanie s pravdepodobnostnými obmedzeniami (chance-constrained programming) a stochastické programovanie s rekurziou (stochastic programming with recourse). Pri programovaní s pravdepodobnostnými obmedzeniami je prípustnosť riešenia v súlade s obmedzeniami zaručená s určitou pravdepodobnosťou. V stochastickom programovaní s rekurziou sú niektoré obmedzenia relaxované a zahrnuté do účelovej funkcie s predpokladom, že ak dôjde k porušeniu týchto podmienok v dôsledku náhodnej udalosti, budú prvotné rozhodnutia rekurzívne opravené. Stochastické modely sú silné, avšak majú dva hlavné nedostatky: rozdelenie pravdepodobnosti náhodných premenných musí byť známe a riešenie je prípustné len pre niektoré realizácie náhodných udalostí.

Robustná optimalizácia (RO) naopak nepredpokladá, že rozdelenie pravdepodobnosti neistých údajov je známe, ale namiesto toho vychádza z predpokladu, že neisté dáta patria do tzv. množiny neurčitostí [6]. Výsledkom RO je riešenie, ktoré je prípustné pre všetky realizácie náhodných udalostí v danej množine. Vo všeobecnosti, cieľom RO je optimalizovať najhorší prípad zo všetkých možných hodnôt náhodných premenných.

Všeobecná formulácia robustnej optimalizácie je:

Minimalizujte

$$f_0(\mathbf{x}),$$

za podmienok

$$f_i(\mathbf{x},\mathbf{u}_i) \leq 0, \qquad \forall \mathbf{u}_i \in \mathscr{U}_i, i = 1,\ldots,m, \tag{9}$$

kde $\mathbf{x} \in R^n$ je vektor rozhodovacích premenných, $\mathbf{u}_i \in R^k$ sú *vektory scenárov* (scenario vectors) a $\mathscr{U}_i \subseteq R^k$ sú *množiny neurčitostí* (uncertainty sets). Cieľom (9) je nájsť riešenia $\mathbf{x}^*$ s najmenšou cenou spomedzi všetkých tých riešení, ktoré sú prípustné pre všetky realizácie scenárov $\mathbf{u}_i$ z množiny $\mathscr{U}_i$. Ak je $\mathscr{U}_i$ jednoprvková, potom zodpovedajúca podmienka nemá žiadnu neurčitosť. Intuitívne, tento prístup ponúka určitú mieru ochrany prípustnosti riešenia pre problémy s parametrami, ktoré nie sú presne známe.

Robustný optimalizačný prístup má oproti stochastickému niekoľko výhod [79]. Po prvé, v mnohých prípadoch je ľahšie definovať množinu neurčitostí než odhadnúť rozdelenie pravdepodobnosti. Po druhé, robustný prístup za určitých podmienok významne nezvyšuje zložitosť problému.

1.4 Robustná lineárna optimalizácia

Pre robustnú verziu lineárneho optimalizačného problému je formulácia nasledovná [9]:

Minimalizujte

$$\mathbf{c}^\top \mathbf{x},$$

za podmienok

$$\mathbf{A}\,\mathbf{x} \leq \mathbf{b}, \qquad \forall \mathbf{a}_1 \in \mathscr{U}_1,\ldots,\mathbf{a}_m \in \mathscr{U}_m, \tag{10}$$

kde $\mathbf{a}_i$ reprezentuje i-ty riadok v matici neurčitostí $\mathbf{A}$ a obsahuje hodnoty z množiny $\mathscr{U}_i \subseteq R^n$. Potom $\mathbf{a}_i^\top x \leq b_i, \forall \mathbf{a}_i \in \mathscr{U}_i$ platí práve vtedy, keď

$$\max_{a_i \in \mathscr{U}_i} \mathbf{a}_i^\top x \leq b_i, \qquad \forall i, \tag{11}$$

čo je podproblém, ktorý treba vyriešiť. Jeho štruktúra určuje zložitosť riešenia robustného optimalizačného problému.

2 SÚČASNÝ STAV RIEŠENIA PROBLEMATIKY

V roku 1959 Dantzig a Ramser vo svojej práci "The truck dispatching problem-[21] prvýkrát uviedli matematickú formuláciu a algoritmické riešenie úlohy, ktorej cieľom bolo minimalizovať celkovú dĺžku trasy homogénnych vozidiel pri zásobovaní skupiny čerpacích staníc palivom z centrálneho zdroja. O päť rokov neskôr, v roku 1964, Clarke a Wright [17] zovšeobecnili túto úlohu ako optimalizačný problém, ktorý sa stal známy ako *kapacitná okružná dopravná úloha* (CVRP) a navrhli prvú efektívnu heuristiku na jej riešenie. Pre širokú škálu praktických aplikácií sa CVRP stala jedným z najviac študovaných kombinatorických optimalizačných problémov, o čom svedčí aj veľké množstvo vedeckých publikácií. Autori predstavili niekoľko rôznych exaktných a heuristických metód riešenia VRP. Všeobecný prehľad výskumu môže čitateľ nájsť napr. v práci [53] alebo [83].

Dvomi hlavnými typmi CVRP s neistými parametrami sú stochastická CVRP (Stochastic Vehicle Routing Problem, SVRP) a robustná CVRP (Robust Vehicle Routing Problem, RVRP). V probléme SVRP sa najčastejšie objavujú tri stochastické parametre: stochastické požiadavky, stochastickí zákazníci a stochastické časy. K výskumu VRP so stochastickými požiadavkami významne prispeli napr. Bertsimas (1992) [12], Laporte a kol. (2002) [54], Christiansen a Lysgaard (2007) [40], Secomandi a Margot (2009) [75], Sun (2014) [78]. VRP so stochastickými zákazníkmi sa venovali Bertsimas (1988) [8], Waters (1989) [86]. Problém VRP so stochastickými zákazníkmi a stochastickými požiadavkami formalizoval Bertsimas (1992) [12] a prvý exaktný algoritmus na jej riešenie navrhli Gendreau, Laporte a Séguin (1995) [29]. Sungur a kol. (2010) [81] prezentoval model a algoritmus na riešenie VRP so stochastickými zákazníkmi a stochastickými časmi obsluhy. Riešením VRP so stochastickými časmi sa zaoberali Kenyon a Morton (2003) [49], Li, Tian a Leung (2010) [57].

V prípade RVRP je najviac študovaný prípad s neistými požiadavkami. Sungur a kol. (2008) [80] boli prví, ktorí skúmali tento problém. V ich práci sú vektory požiadaviek konštruované ako odchýlka od očakávanej hodnoty, ktorá prislúcha rôznym ohraničeným množinám. Robustná formulácia zohľadňuje neisté požiadavky v podmienkach, podobne ako zaviedli Ben-Tal a Nemirovski (1998) [6]. Moghaddam a kol. (2012) [63] sa s neistou distribúciou požiadaviek zákazníkov vysporiadali pomocou robustnej optimalizácie tak, že percento poruchy od nominálnej hodnoty požiadaviek definovali v podmienkach, pričom cieľom bolo minimalizovať cestovné náklady. Inú štúdiu RVRP s neistými požiadavkami prezentovali Gounaris a kol. (2013) [33]. Autori vyvinuli robustné kapacitné nerovnosti (Robust Rounded Capacity Inequalities, RCI). V modeloch sú požiadavky zákazníkov ako náhodné premenné na pravých stranách podmienok a určujú minimálnu cenu dodacieho plánu, ktorý je prípustný pre všetky realizácie predpokladaných požiadaviek. V novšej práci Gounaris a kol. (2014) [34] uviedli robustnú formuláciu VRP s neistými požiadavkami a implementovali metaheuristiku s použitím dvoch tried množín neurčitosti. Výskumu VRP s neistými časmi jázd modelovaných pomocou robustných scenárov sa venovali Han a kol. (2013) [41]. Obmedzili počet neistých parametrov, ktorým je dovolené odchýliť sa od nominálnej hodnoty (Bertsimas a Sim, 2003) [10]. Potom pre každú realizáciu (scenár) určili robustnú trasu a minimalizovali najhorší prípad zo všetkých scenárov. Využili dvojstupňové rekurzívne stochastické programovanie riešené metódou BB. Výsledky ukazujú, že takýto prístup získa dobré riešenia, keď penalizácia v účelovej funkcii je malá. Práce autorov Toklu a kol. (2013) [82] sú zamerané na VRP s neistými prepravnými cenami modelovanými ako intervaly. Na minimalizáciu celkovej ceny prepravy použili algoritmus ACO, kde poruchy sú zohľadnené v koeficientoch účelovej funkcie smerom k horným hraniciam intervalov. RVRP s neistými časmi jázd skúmali Solano-Charis a kol. (2015) [77]. Množinu cien hrán nahradili množinou diskrétnych scenárov s cieľom nájsť množinu trás s použitím lexikografického min-max kritéria. Ordóñez (2010) [67] sa vo svojej práci zameral na VRP s viacerými neistými parametrami – s neistými požiadavkami, časmi jázd, cenami a zákazníkmi. Autor popisuje rôzne robustné modely v závislosti od pôvodu neurčitosti, formulácie VRP a korelácie medzi neurčitými koeficientami. Neisté prepravné ceny sú zavedené v účelovej funkcii a neisté požiadavky a časy jázd v podmienkach. Neisté údaje sú modelované ako konvexné a ohraničené množiny neurčitostí a riešenie problému je založené na optimalizačnom prístupe podľa Ben-Tal a Nemirovského (1998) [6]. Riešenie VRP s neistými časmi jázd a požiadaviek je predmetom práce autorov Lee a kol. (2012) [55], pri ktorom na obmedzenie súčtu odchýliek časov jázd a odchýlky požiadaviek od nominálnych hodnôt definovali budžet neurčitosti podľa Bertsimas a Sim (2003) [10]. Robustnosť riešenia sa dosiahne hľadaním prípustného riešenia pre každý čas jazdy a požiadavku z množín neurčitosti s minimálnym časom prepravy. Vychádza z formulácie založenej na rozklade množiny (set partitioning formulation) a používa metódu generovania stĺpcov.

3 ZVOLENÁ METODIKA PRÁCE A METÓDY SKÚMANIA

3.1 Spôsob modelovania neistých dát

Neisté požiadavky môžu byť modelované rozličnými spôsobmi, napr. ako stochastické premenné so známym rozdelením pravdepodobnosti [12] alebo ako fuzzy premenné [25]. Pri robustnej optimalizácii sa však predpokladá, že rozdelenie pravdepodobnosti nie je známe a nedá sa ani odhadnúť.

V našej práci sme použili metodológiu, ktorú zaviedli Ben-Tal a Nemirovski [6], [7] pre robustnú lineárnu, kvadratickú a konvexnú optimalizáciu a pre celočíselné programovanie ju rozšírili Bertsimas a Sim [10]. Táto metodológia predpokladá, že neisté parametre patria danej ohraničenej množine neurčitostí. Napríklad, pre problém lineárneho programovania s neistým parametrom a je množina neurčitostí $\mathscr{U}_a$ ohraničená mnohostenom, určeným sústavou podmienok vyjadrených lineárnymi nerovnicami:

$$\mathscr{U}_a = \{(1+\xi_k)\,a_0 : |\xi_k| \leq \varepsilon\},$$

kde $0 \leq \varepsilon \leq 1$. To znamená, že realizácia $a_k = (1+\xi_k)\,a_0$ neistého parametra v k-tom scenári je daná pomocou odchýlky od jeho nominálnej hodnoty a_0, pričom ξ_k je náhodná premenná s rovnomerným rozdelením z intervalu $\langle -\varepsilon, \varepsilon \rangle$.

Neisté požiadavky zákazníkov d_i^k sme modelovali ako náhodné premenné,

$$d_i^k \in \langle d_i^0 - \varepsilon\, d_i^0, d_i^0 + \varepsilon\, d_i^0 \rangle \quad \forall i \in \{1, \ldots, n\},$$

kde d_i^0 sú očakávané (nominálne) požiadavky a n je počet zákazníkov.

3.2 Matematický model

Existuje viacero rôznych matematických formulácií CVRP, z ktorých by sa dal odvodiť matematický model RVRP. Pri výbere je však potrebné zohľadniť obtiažnosť riešenia výsledného RVRP. V matematickej formulácii CVRP 1, ktorú sme uviedli v predchádzajúcej kapitole, počet nerovností v anticyklických podmienkach v (2) rastie exponenciálne s počtom zákazníkov. Preto sme navrhli nový, omnoho efektívnejší model CVRP 2, formulovaný ako úloha zmiešaného lineárneho programovania s polynomiálnym počtom anticyklických podmienok.

Okrem toho, v metodológii RO sa preferujú podmienky obsahujúce neisté parametre vo forme nerovností, nakoľko splniť rovnosť pre všetky hodnoty neistých parametrov je oveľa zložitejšie. Aj z tohto hľadiska je náš model CVRP 2 vhodnou formuláciou.

3.3 Metódy generovania robustných riešení

Podľa robustnej optimalizačnej metodológie, robustné riešenie RVRP je riešenie, ktoré je optimálne pre najhorší scenár z množiny $\mathscr{U}_d$. Takéto riešenie je *imúnne* voči neistým vstupným dátam, pretože ak je prípustné pre najhorší prípad, tak potom je prípustné pre všetky možné prípady. Kľúčovým krokom pri riešení RVRP teda je identifikovať najhorší scenár.

3.3.1 Stratégie na nájdenie najhoršieho scenára

- *Stratégia maximálneho scenára* definuje maximálny scenár, v ktorom má každý zákazník najväčšiu požiadavku zo všetkých scenárov [80]. Následným vyriešením inštancie CVRP s maximálnym scenárom dostaneme robustné riešenie, ktoré síce môže mať väčšiu cenu, než optimálne riešenie CVRP s očakávaným (nominálnym) scenárom, ale na druhej strane predchádza neuspokojeným požiadavkám v prípade, že nastane iný scenár. Nevýhodou tejto stratégie je, že aj keď jednotlivé scenáre sú prípustné, maximálny scenár nemusí byť vždy prípustný. Z tohto dôvodu sme navrhli nasledujúcu stratégiu.

- *Stratégia najhoršieho prípustného scenára* vyberie u každého zákazníka do najhoršieho scenára čo najväčšiu požiadavku zo všetkých scenárov tak, aby výsledný scenár bol prípustný. Najhorší prípustný scenár získame riešením úlohy bivalentného lineárneho programovania. Riešenie inštancie CVRP s najhorším prípustným scenárom je vždy prípustné a v prípade, že sa najhorší prípustný scenár zhoduje s maximálnym scenárom, je aj robustné. Ak sa najhorší prípustný scenár nezhoduje s maximálnym, robustnosť riešenia nie je síce zaručená, ale neuspokojené požiadavky sú minimalizované.

Nájsť robustné riešenie znamená vyriešiť inštanciu CVRP s najhorším scenárom. Na riešenie CVRP sme modifikovali niektoré metódy známe z literatúry a okrem toho sme navrhli dve nové metódy - exaktnú iteratívnu metódu a lexikografickú metódu.

3.3.2 Metódy riešenia CVRP

- *Exaktná iteratívna metóda* je metódou lokálneho prehľadávania s najlepším zlepšením, ktorá iteratívne zlepšuje východiskové prípustné riešenie nahradením niektorých hrán inými, čo dosiahneme exaktným riešením jednoduchšieho problému.

- *Lexikografická metóda* je metóda, ktorá nájde optimálne riešenie CVRP, pričom bude využitá len najmenšia nutná kapacita vozidiel. Nevyužitá kapacita vozidiel predstavuje rezervu pre prípad, že sa očakávané požiadavky náhodne zvýšia, výsledné riešenie je teda do istej miery robustné.
- *Paralelný mikrogenetický algoritmus* je metaheuristická metóda, ktorá je variantom genetického algoritmu s veľmi malou populáciou. Paralelný mikrogenetický algoritmus využíva benefity mikrogenetického algoritmu a paralelizácie výpočtov. Výsledkom je suboptimálne riešenie.

3.4 Kritérium robustnej optimality

3.4.1 Neuspokojené požiadavky

Ak by sme ignorovali premenlivosť požiadaviek a použili by sme deterministické optimálne riešenie, mohlo by dôjsť k tomu, že keď sa požiadavky zmenia tak, že na niektorej trase bude prekročená kapacita vozidla, bude toto riešenie neprípustné.
Neuspokojené požiadavky sú požiadavky, ktoré na jednotlivých trasách prekročia kapacitu vozidla. Označme g_d neuspokojené požiadavky, ktoré vzniknú pri použití deterministického optimálneho riešenia a g_r neuspokojené požiadavky pri riešení získanom robustnou optimalizáciou. Ukazovateľ

$$\gamma = \frac{g_d - g_r}{\sum_{i=1}^{n} d_i^0} \tag{12}$$

vyjadruje zníženie relatívneho množstva neuspokojených požiadaviek pri nájdenom riešení v porovnaní s optimálnym riešením, kde $\sum_{i=1}^{n} d_i^0$ je súčet očakávaných požiadaviek. Na základe tohto ukazovateľa môžeme porovnať riešenia získané rôznymi stratégiami, pričom väčšia hodnota γ znamená lepšie riešenie.

3.4.2 Zvýšenie ceny riešenia

Na porovnanie riešení nájdených rôznymi metódami z hľadiska ceny použijeme ukazovateľ ζ, ktorý predstavuje relatívne zvýšenie ceny z_r nájdeného riešenia voči cene z_d optimálneho deterministického riešenia:

$$\zeta = \frac{z_r - z_d}{z_d}. \tag{13}$$

Tento pomer nám dáva informáciu, o čo relatívne vyššie budú prepravné náklady, ak použijeme riešenie, ktoré predchádza neuspokojeným požiadavkám, namiesto deterministického optimálneho riešenia. Z toho vyplýva, že čím je hodnota ζ menšia, tým je riešenie lepšie.

3.5 Testovacie inštancie

Na testovanie a overovanie našich algoritmov a modelov sme využili klasické inštancie CVRP z repozitára dostupného na [90]. Inštancie sú zoskupené v sériách, z ktorých sme zvolili tieto:

- **Séria A** (Augerat [3]) - Vrcholy sú umiestnené na náhodných pozíciách v mriežke (0,100)x(0,100). Požiadavky zákazníkov sú náhodné hodnoty z rovnomerného rozdelenia na intervale (1,30), avšak desatina z nich je vynásobená tromi.
- **Séria B** (Augerat [3]) - Podobne ako set A, ale zákazníci sú umiestnení v klastroch.
- **Séria E** (Christifides a Eilon [43]) - Informácie o ich generovaní nie sú známe.
- **Séria P** (Augerat [3]) - Dáta boli vytvorené modifikáciou inštancií zo sérií A, B a E zmenou ich kapacít a následne aj zmenou požadovaného počtu vozidiel

Zvolené inštancie sme modifikovali na inštancie RVRP zohľadnením neistých požiadaviek zákazníkov. Pre každú inštanciu sme vygenerovali množinu neurčitostí $\mathscr{U}_d$ s 5 scenármi s danou maximálnou odchýlkou ε tak, že v každom scenári požiadavky zákazníkov sú náhodné hodnoty z intervalu $\langle d_i^0 - \varepsilon d_i^0, d_i^0 + \varepsilon d_i^0 \rangle$, kde d_i^0 je očakávaná požiadavka zákazníka i z pôvodnej inštancie CVRP.

3.6 Softvérové nástroje

Všetky navrhnuté metódy sme implementovali v jazyku Python 3.4 [88] s využitím jeho modulov: NumPy, NetworkX, Matplotlib, mpi4py. Úlohy lineárneho programovania sme riešili pomocou optimalizačného softvéru Gurobi 6.5 [89].

4 VÝSLEDKY PRÁCE

4.1 Nový model CVRP

Počet nerovností v anticyklických podmienkach v (2) rastie exponenciálne s počtom zákazníkov. Nový model CVRP, ktorý predstavíme, je formulovaný ako úloha zmiešaného lineárneho programovania s polynomiálnym počtom anticyklických podmienok. Použijeme rozhodovacie premenné x_{ij}, kde

$$x_{ij} = \begin{cases} 1, & \text{ak optimálne riešenie obsahuje hranu } (i,j), \\ 0, & \text{inak.} \end{cases} \tag{14}$$

Aby sme zabezpečili spojitosť trasy a zabránili vzniku nežiadúcich cyklov, definujeme doplnkovú premennú y_i, $d_i \leq y_i \leq Q$ pre $i \in V - \{0\}$, ktorá vyjadruje využitú kapacitu (náklad) vozidla pri odchode od zákazníka i [48].
Pre každú dvojicu zákazníkov $i, j \in V - \{0\}, i \neq j$ vypočítame úsporu s_{ij}, ktorú získame ak spojíme cykly $0 \to i \to 0$ a $0 \to j \to 0$ do jedného použitím hrany $i \to j$, t. j.

$$s_{ij} = c_{i0} + c_{0j} - c_{ij} \tag{15}$$

podobne, ako v Clarke-Wrightovej metóde [17]. Potom, namiesto minimalizácie celkovej ceny, budeme maximalizovať celkovú úsporu pri preprave.

Pre zjednodušenie modelovania každú prípustnú trasu $0 \to v_1 \to v_2 \to \cdots \to v_k \to 0$ nahradíme cestou z vrcholu 0 do vrcholu v_k, t. j. $0 \to v_1 \to v_2 \to \cdots \to v_k$.

4.1.1 Model CVRP 2

Maximalizujte

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} s_{ij} x_{ij}, \tag{16}$$

za podmienok

$$\sum_{j=1}^{n} x_{0j} = p, \tag{17}$$

$$\sum_{i=1}^{n} x_{i0} = 0, \tag{18}$$

$$\sum_{\substack{i=0 \\ i \neq j}}^{n} x_{ij} = 1, \qquad \forall j \in \{1, \ldots, n\}, \tag{19}$$

$$\sum_{\substack{j=1 \\ i \neq j}}^{n} x_{ij} \leq 1, \qquad \forall i \in \{1, \ldots, n\}, \tag{20}$$

$$y_i + d_j x_{ij} - Q(1 - x_{ij}) \leq y_j, \qquad \forall i, j \in \{1, \ldots, n\}, i \neq j, \tag{21}$$

$$d_i \leqq y_i \leq Q, \qquad \forall i \in \{1, \ldots, n\}, \tag{22}$$

$$x_{ij} \in \{0, 1\}, \qquad \forall i, j \in \{0, \ldots, n\}, i \neq j. \tag{23}$$

V tejto formulácii účelová funkcia (16) maximalizuje celkovú úsporu pri preprave. Podmienky (17) a (18) stanovujú, že z depa vyjde presne p hrán, a do depa nevojde žiadna hrana, (19) a (20) určujú počet vozidiel prichádzajúcich a odchádzajúcich k/od zákazníkov. Skupina podmienok (21) predstavuje súčasne anticykliace podmienky a podmienky kontinuity toku, ktoré zaručujú, že riešenie neobsahuje cykly neincidentné s depom a tiež, že využitá kapacita vozidla je neklesajúca (v prípade zberu komodít) postupnosť v závislosti od požiadaviek zákazníkov prislúchajúcich tej istej trase. Podmienky (22) upresňujú dolné a horné ohraničenie y_i a (23) sú obligatórne podmienky.

4.2 Nová exaktná iteratívna metóda na riešenie CVRP

Naša exaktná metóda na riešenie CVRP [15] je postavená na koncepte *k-optimálnosti* [56]. Hlavná myšlienka spočíva v iteratívnom zlepšovaní východiskového prípustného riešenia S nahradením niektorých hrán inými, čo dosiahneme exaktným riešením jednoduchšieho problému pomocou modelu CVRP 2. Túto metódu môžeme teda zaradiť medzi metódy lokálneho prehľadávania s najlepším zlepšením.

4.2.1 Algoritmus CVRP 3

Krok 1: Pomocou rýchlej heuristickej metódy nájdi počiatočné prípustné riešenie S_0.
Polož $S = S_0$.

Krok 2: Vytvor zoznam E hrán prislúchajúcich riešeniu S, ktoré nie sú incidentné s depom.
Nech $m = |E|$.

Krok 3: Polož hodnoty $t = 1$, $m_1 = m - \delta$ a $m_2 = m - 1$, kde δ je vopred definovaná celočíselná hodnota $1 < \delta < m$.

Krok 4: Pomocou CVRP 2 nájdi optimálne riešenie S_t také, aby množina E_t hrán prislúchajúcich riešeniu S_t, ktoré nie sú incidentné s depom obsahovala minimálne m_1 a maximálne m_2 hrán z E, t.j. do modelu CVRP 2 pridáme podmienky:

$$\sum_{(i,j)\in E} x_{ij} = z, \tag{24}$$

$$m_1 \leq z \leq m_2, \tag{25}$$

kde celočíselná premenná z určuje, koľko hrán z E sa nachádza v množine E_t.

Krok 5: Ak je riešenie S_t lepšie než riešenie S, tak polož $S = S_t$ a pokračuj krokom 2.
Inak polož hodnoty $t = t + 1$, $m_1 = m_1 - \delta$ a $m_2 = m_2 - \delta$.

Krok 6: Ak $m_1 < 0$, polož $m_1 = 0$. Ak $m_2 \geq 0$, choď na krok 4, inak Stop.

Algoritmus CVRP 3 mení počet hrán vymieňaných hrán počnúc $k = 1$ a končiac $k = m$, čiže konečné riešenie je m-optimálne. Preto je táto metóda exaktná. Z našich experimentov vyplynulo, že naša metóda je schopná v prijateľnom čase exaktne vyriešiť aj inštancie s relatívne väčším počtom zákazníkov. Prípadne, ak by bol výpočet veľmi zdĺhavý, je možné ho prerušiť, pričom získame k-optimálne riešenie (posledné riešenie v CVRP 3).

4.3 Riešenie CVRP pomocou paralelného mikrogenetického algoritmu

Mikrogenetický algoritmus (μGA) je verzia genetického algoritmu (GA), ktorý hľadá najlepšie prípustné (suboptimálne) riešenie problému na princípe darwinovskej evolúcie. Na rozdiel od klasického GA, kde je populácia zložená z veľkého počtu jedincov (30 – 200), μGA používa mikropopuláciu s 3 – 5 jedincami (Goldberg, 1989 [35]), a preto je výpočtový čas v porovnaní s GA významne kratší a aj pamäťová náročnosť na uchovanie populácie je omnoho menšia. Samozrejme, malá populácia vývojom rýchlo stratí svoju rôznorodosť (*diverzitu*), t.j. jedinci sa po krátkom čase začnú navzájom podobať a vývoj stagnuje. Môže to znamenať, že došlo k *predčasnej konvergencii*, čiže riešenie uviazlo v lokálnom extréme. Aby evolučný proces mohol ďalej pokračovať, populácia sa reštartuje, pričom sa zachová len najlepší jedinec (*elitárstvo*) a zvyšní štyria sa náhodne vygenerujú. Elitárstvo zaručí, že najlepší jedinec v nasledujúcej generácii nebude horší od predchádzajúceho.

4.3.1 Mikrogenetický algoritmus μGA

Krok 1: Vytvor náhodne populáciu s 5 jedincami (inicializácia) a choď na krok 3.

Krok 2: Vytvor populáciu so 4 jedincami náhodne a 1 najlepšieho jedinca z predchádzajúcej generácie (reštart).

Krok 3: Vypočítaj fitness jedincov.

Krok 4: Urči najlepšieho jedinca - toho uchovaj do nasledujúcej generácie (elitárstvo).

Krok 5: Pomocou turnajového výberu zvoľ 2 páry jedincov (rodičov) na reprodukciu.

Krok 6: Vykonaj kríženie a potomkov pridaj do novej generácie.

Krok 7: Vypočítaj fitness jedincov.

Krok 8: Skontroluj, či došlo k strate diverzity a ak nie, choď na krok 4.

Krok 9: Ak nebolo splnené ukončovacie kritiérium, choď na krok 2., inak Stop.

Ukončovacím kritériom môže byť napríklad dosiahnutie predpísaného počtu generácií počítaných od začiatku vývoja alebo od posledného zlepšenia hodnoty fitness najlepšieho jedinca. Výsledným riešením je najlepší jedinec v aktuálnej generácii.

Jedinec S v našom μGA predstavuje riešenie CVRP a jeho fitness $f(S)$ je rovná cene tohto riešenia. Jedinca budeme reprezentovať ako vektor (*chromozóm*), ktorého zložky (*gény*) sú zákazníci v takom poradí, v akom sú obslúžení na jednotlivých trasách, bez oddeľovačov jednotlivých trás, t.j. $S = (S_1, S_2, \ldots, S_n)$, kde $S_i \in \{1, \ldots, n\}$ a $S_i \neq S_j$ ak $i \neq j$. Jedná sa o tzv. *permutačné kódovanie jedinca*. V prípade potreby (napríklad pri výpočte hodnoty fitness $f(S)$) sa dá chromozóm S dekódovať pomocou procedúry *split* (Prins, 2004 [71]). Na výber jedincov pre reprodukciu sme použili metódu *binárneho turnajového výberu* (Goldberg, 1990 [36]) a pre kríženie operátor *kríženia s čiastkovým mapovaním* (Partially Mapped Crossover, PMX) [37].

Efektívnosť výpočtu môžeme zvýšiť jeho paralelizáciou, t.j. rozložením výpočtového výkonu na viac procesorov. V literatúre možno nájsť niekoľko stratégií paralelizácie GA, napr. [19], [76]. Náš paralelný mikrogenetický algoritmus, ktorý sme navrhli pre riešenie CVRP pracuje na princípe hrubozrnného modelu, avšak namiesto podpopulácií veľkej populácie, sme použili mikropopulácie a proces migrácie sme nahradili dosadením *nasadeného jedinca*.

4.3.2 Paralelný mikrogenetický algoritmus pμGA

Najprv modifikujeme algoritmus μGA tak, že vstupom bude vektor reprezentujúci nasadeného jedinca a pri inicializácii (v kroku 1.) sa vytvorí populácia s nasadeným jedincom a 4 náhodne vygenerovanými jedincami. Každý procesor bude vykonávať μGA so svojou vlastnou mikropopuláciou, až kým nebude splnené ukončovacie kritérium. Túto periódu izolovaného vývoja mikropopulácie nazveme *epocha*.

Inicializácia prvého nasadeného jedinca

Prvý nasadený jedinec môže byť vytvorený buď náhodným vygenerovaním alebo použitím rýchlej konštruktívnej heuristiky, napr. Clarke-Wrightovej metódy.

Úlohy, ktoré vykonáva procesor počas jednej epochy

Krok 1: Prijmi nasadeného jedinca.

Krok 2: Inicializuj populáciu s 5 jedincami: 1 nasadený a 4 náhodne vygenerovaní jedinci.

Krok 3: Opakuj μGA kým nie je splnené ukončovacie kritérium.

Krok 4: Odovzdaj najlepšieho jedinca.

Jeden z procesorov je určený ako manažér úloh (*root*). Jeho úlohou je na začiatku každej novej epochy rozposlať (všetkým procesorom rovnakého) nasadeného jedinca a po jej skončení vyzbierať a vyhodnotiť výsledky od všetkých procesorov.

Operácie, ktoré vykonáva manažér úloh

Krok 1: Inicializuj prvého nasadeného jedinca.

Krok 2: Odošli nasadeného jedinca jednotlivým procesorom (*broadcast*).

Krok 3: Po skončení epochy prijmi výsledky od všetkých procesorov (*gather*).

Krok 4: Urči najlepší zo všetkých prijatých výsledkov a označ ho ako nového nasadeného jedinca.

Krok 5: Ak nebolo splnené ukončovacie kritérium, choď na krok 2., inak Stop.

Výpočet sa ukončí po stanovenom počte epoch a výsledkom bude posledný nasadený jedinec.

4.4 Lexikografický model CVRP

V našom modeli CVRP 2 sa využíva presne p vozidiel, pričom predpokladáme, že celkový dopyt nepresahuje kapacitu všetkých vozidiel, t.j. $\sum_{i=1}^{n} d_i \leq pQ$. Avšak v niektorých praktických situáciách celková kapacita všetkých vozidiel nepostačuje na to, aby boli uspokojené všetky požiadavky, hoci nie je menšia než celkový dopyt zákazníkov. Ak by sme predsa-len chceli využiť presne p vozidiel (napr. preto, že depo zamestnáva p vodičov), musíme určiť, najmenej akú kapacitu Q_{min} musia mať vozidlá, aby boli schopné obslúžiť všetkých zákazníkov. Minimálnu potrebnú kapacitu Q_{min} určíme pomocou *modifikovaného problému kontajnerov* (mBPP), ktorý môžeme formulovať nasledovne: Treba rozmiestniť n objektov, každý s kladnou hmotnosťou d_i, $i = 1,\dots,n$, do p kontajnerov, $1 \leq p \leq n$, s čo najmenšou rovnakou kapacitou tak, aby celková hmotnosť objektov v žiadnom kontajneri nepresiahla jeho kapacitu.

Problém mBPP budeme modelovať pomocou bivalentných rozhodovacích premenných b_{ij}, kde

$$b_{ij} = \begin{cases} 1, & \text{ak objekt } j \text{ je umiestnený v kontajneri } i, \\ 0, & \text{inak} \end{cases} \tag{26}$$

a celočíselnej premennej q, $q > 0$, ktorá reprezentuje kapacitu kontajnera. Cieľom je minimalizovať q.

4.4.1 Model mBPP

Minimalizujte

$$q \tag{27}$$

za podmienok

$$\sum_{i=1}^{p} b_{ij} = 1, \qquad \forall j \in \{1,\dots,n\}, \tag{28}$$

$$\sum_{j=1}^{n} b_{ij} \geq 1, \qquad \forall i \in \{1,\dots,p\}, \tag{29}$$

$$\sum_{j=1}^{n} b_{ij} d_j \leq q, \qquad \forall i \in \{1,\dots,p\}, \tag{30}$$

$$b_{ij} \in \{0,1\}, \qquad \forall i \in \{1,\dots,p\}, \forall j \in \{0,\dots,n\}. \tag{31}$$

Účelová funkcia (27) minimalizuje kapacitu kontajnera, resp. inými slovami hornú hranicu hmotnosti nákladu, ktorý do kontajnera smieme naložiť. Obmedzujúce podmienky (28) zaručujú, že každý objekt bude umiestnený do práve jedného kontajnera, nerovnice (29) zas zabezpečujú, že každý kontajner bude použitý. Ďalšie podmienky (30) sú kapacitné a (31) sú obligatórne podmienky.

4.4.2 Definícia lexikografickej CVRP

Podobne ako v CVRP, máme množinu n zákazníkov s istými požiadavkami, ktoré sú nedeliteľné. Označme D celkový dopyt všetkých zákazníkov, teda $D = \sum_{i=1}^{n} d_i$. Nech počet vozidiel s rovnakou kapacitou, ktorými disponuje depo je p. *Lexikografická kapacitná dopravná okružná úloha* (LCVRP) má nájsť p prípustných trás tak, aby celkové prepravné náklady boli minimálne a pritom

- každá trasa začínala aj končila v depe,
- každý zákazník bol obslúžený práve jedným vozidlom,
- na žiadnej trase nebola prekročená najmenšia nutná kapacita Q_{min}.

Daný problém môžeme riešiť dvojakým spôsobom. Prvá možnosť je vypočítať minimálnu kapacitu vozidla Q_{min} riešením korešpondujúceho mBPP a následne vyriešiť CVRP s $Q = Q_{min}$. Druhý spôsob, ako zobrať do úvahy požiadavku minimalizovať kapacitu vozidla, je použiť ďalšiu rozhodovaciu premennú q reprezentujúcu kapacitu vozidla (rovnako ako v mBPP) a upraviť účelovú funkciu pridaním člena $M\,q$, kde M je veľké kladné číslo, tzv. penalizačná konštanta. Takto budeme mať možnosť v prvom rade minimalizovať kapacitu vozidla a popritom minimalizovať aj cenu riešenia.
Pretože v našom modeli CVRP 2 účelová funkcia namiesto minimalizácie prepravných nákladov maximalizuje celkovú úsporu, LCVRP sformulujeme matematicky takto:

4.4.3 Model LCVRP

Maximalizujte

$$\sum_{i=1}^{n}\sum_{j=1,i\neq j}^{n} s_{ij}x_{ij} - Mq, \tag{32}$$

za podmienok

$$\sum_{j=1}^{n} x_{0j} = p, \tag{33}$$

$$\sum_{i=0,i\neq j}^{n} x_{ij} = 1, \qquad \forall j \in \{1,\dots,n\}, \tag{34}$$

$$\sum_{j=1,i\neq j}^{n} x_{ij} \le 1, \qquad \forall i \in \{1,\dots,n\}, \tag{35}$$

$$y_i + d_j x_{ij} - D(1-x_{ij}) \le y_j, \qquad \forall i,j \in \{1,\dots,n\}, i\neq j, \tag{36}$$

$$d_i \le y_i \le q, \qquad \forall i \in \{1,\dots,n\}, \tag{37}$$

$$q \ge \max\{\lceil D/p \rceil, \max\{d_i\}\} \tag{38}$$

$$x_{ij} \in \{0,1\}, \qquad \forall i,j \in \{0,\dots,n\}, i\neq j. \tag{39}$$

Podmienky (33), (34), (35), (37) a (39) sú rovnaké ako v modeli CVRP 2. Nerovnice (36) vyjadrujú anticyklické podmienky a súčasne podmienky kontinuity toku a podmienka (38) vymedzuje dolnú hranicu premennej q. Z experimentov vyplynulo, že veľkosť penalizačnej konštanty M v účelovej funkcii (32) ovplyvňuje správnosť výsledkov a závisí od vstupných údajov. Z tohto dôvodu bolo potrebné nastaviť hodnotu M nasledovne:

$$M = \left\lceil \frac{\max\{s_{ij}\}(n-p)}{\min\{d_i\}} \right\rceil, \tag{40}$$

kde $n-p$ je počet hrán, ktoré patria riešeniu a nie sú incidentné s depom.

4.5 Riešenie robustnej okružnej dopravnej úlohy s neistými požiadavkami zákazníkov

Problém robustnej okružnej dopravnej úlohy (RVRP) s neistými požiadavkami zákazníkov spočíva v tom, že kým v CVRP sú požiadavky zákazníkov vopred exaktne dané (deterministické), v RVRP majú povahu náhodných premenných, pričom však rozdelenie pravdepodobnosti týchto náhodných premenných nie je známe. Za predpokladu, že trasy vozidiel musia byť určené ešte predtým, než budú známe aktuálne hodnoty požiadaviek zákazníkov a nebudú sa môcť dodatočne meniť, môže náhodné správanie požiadaviek spôsobiť, že navrhnuté prípustné riešenie sa stane neprípustným (lebo ak na niektorej trase bude prekročená kapacita vozidla, tak niektorí zákazníci nebudú obslúžení).

4.5.1 Diskrétne scenáre

Neistotu budeme modelovať pomocou *diskrétnych scenárov* tak, že v každom scenári budú definované aktuálne požiadavky zákazníkov. Cieľom je nájsť *robustné riešenie*, čiže riešenie (s minimálnymi prepravnými nákladmi), ktoré bude "dobré"pre každý scenár, t. j. v každom scenári budú uspokojené všetky požiadavky.

4.5.2 Formálna definícia problému

Pri formulácii RVRP budeme predpokladať, že neisté požiadavky zákazníkov patria danej ohraničenej množine diskrétnych scenárov

$$\mathscr{U}_d = \{\mathbf{d}^k \in R^n | k \in \{1,\dots,s\}\},$$

kde $\mathbf{d}^k$ je vektor požiadaviek v scenári k a s je počet scenárov. Budeme uvažovať takú množinu $\mathscr{U}_d$, v ktorej sú vektory $\mathbf{d}^k$ konštruované pomocou odchýlok od vektora $\mathbf{d}^0$ očakávaných hodnôt, t. j. pre požiadavku d_j^k zákazníka j v scenári k platí

$$d_j^k \in \langle d_j^0 - \varepsilon d_j^0, d_j^0 + \varepsilon d_j^0 \rangle \quad \forall j \in \{1, \ldots, n\},$$

kde ε je konštanta, ktorá určuje maximálnu možnú relatívnu odchýlku od nominálnej hodnoty, pričom

$$(1+\varepsilon) \sum_{i=1}^{n} d_i^k \leq pQ \text{ pre } \forall k \in \{1, \ldots, s\}.$$

RVRP hľadá riešenie, ktoré je prípustné pre všetky scenáre z množiny $\mathscr{U}_d$. To znamená, že formuláciu RVRP s neistými požiadavkami môžeme odvodiť z formulácie CVRP (Model CVRP 2) ako nasledujúcu úlohu MILP:

Model RVRP 1

Maximalizujte

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} s_{ij} x_{ij}, \tag{41}$$

za podmienok

$$\sum_{j=1}^{n} x_{0j} = p, \tag{42}$$

$$\sum_{\substack{i=0 \\ i \neq j}}^{n} x_{ij} = 1, \qquad \forall j \in \{1, \ldots, n\}, \tag{43}$$

$$\sum_{\substack{j=1 \\ i \neq j}}^{n} x_{ij} \leq 1, \qquad \forall i \in \{1, \ldots, n\}, \tag{44}$$

$$y_i + d_j x_{ij} - Q(1 - x_{ij}) \leq y_j, \qquad \forall d \in \mathscr{U}_d, \forall i, j \in \{1, \ldots, n\}, i \neq j, \tag{45}$$

$$d_i \leq y_i \leq Q, \qquad \forall d \in \mathscr{U}_d, \forall i \in \{1, \ldots, n\}, \tag{46}$$

$$x_{ij} \in \{0, 1\}, \qquad \forall i, j \in \{0, \ldots, n\}, i \neq j. \tag{47}$$

4.5.3 Stratégie na riešenie RVRP

Pretože RVRP je NP-ťažký kombinatorický problém, riešiť exaktne úlohu formulovanú v modeli RVRP 1 je možné len pre inštancie s malým počtom zákazníkov n a scenárov s. Riešenie sa však dá zjednodušiť na základe toho, že množina scenárov $\mathscr{U}_d$ je ohraničená. Pre takúto množinu sa dá dokázať, že výsledný RVRP problém je inštanciou CVRP [80].

Stratégia maximálneho scenára

Nech $d_j^m = \max\{\max_k d_j^k, d_j^0\}$ pre $j \in \{1, \ldots, n\}, k \in \{1, \ldots, s\}$. Potom pre definovanú množinu $\mathscr{U}_d$ môžeme podmienky (45) nahradiť podmienkami

$$y_i + d_j^m x_{ij} - Q(1 - x_{ij}) \leq y_j, \quad \forall i, j \in \{1, \ldots, n\}, i \neq j. \tag{48}$$

To znamená, že optimálne riešenie úlohy so scenárom $\mathbf{d}^m$ bude prípustné pre všetky scenáre z množiny $\mathscr{U}_d$, a tak robustné riešenie môžeme hľadať optimalizáciou inštancie CVRP s vektorom požiadaviek $\mathbf{d}^m$. Stačí teda riešiť nasledujúcu úlohu MILP odvodenú z modelu CVRP 2:

Model RVRP 2

Maximalizujte

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} s_{ij} x_{ij}, \tag{49}$$

za podmienok

$$\sum_{j=1}^{n} x_{0j} = p, \tag{50}$$

$$\sum_{\substack{i=0\\i\neq j}}^{n} x_{ij} = 1, \qquad \forall j \in \{1,\dots,n\}, \tag{51}$$

$$\sum_{\substack{j=1\\i\neq j}}^{n} x_{ij} \leq 1, \qquad \forall i \in \{1,\dots,n\}, \tag{52}$$

$$y_i + d_j^m x_{ij} - Q(1-x_{ij}) \leq y_j, \qquad \forall i,j \in \{1,\dots,n\}, i \neq j, \tag{53}$$

$$d_i^m \leq y_i \leq Q, \qquad \forall i \in \{1,\dots,n\}, \tag{54}$$

$$x_{ij} \in \{0,1\}, \qquad \forall i,j \in \{0,\dots,n\}, i \neq j. \tag{55}$$

V závislosti od povahy scenárov, môže nastať situácia, keď scenár $\mathbf{d}^m$ nie je prípustný, hoci jednotlivé scenáre z množiny $\mathscr{U}_d$ sú prípustné alebo dokonca, keď robustné riešenie neexistuje. V takýchto prípadoch sa pokúsime nájsť také riešenie, pri ktorom bude čo najmenej neuspokojených požiadaviek.

Stratégia najhoršieho prípustného scenára

Hlavným cieľom robustnej optimalizácie vo všeobecnosti je optimalizovať najhorší prípad zo všetkých možných scenárov. Ako sme ukázali v predchádzajúcej časti, môžeme za najhorší scenár považovať vektor požiadaviek $\mathbf{d}^m$ so zložkami $d_j^m = \max\{\max_k d_j^k, d_j^0\}$, t. j. s najväčšími požiadavkami zákazníkov zo všetkých scenárov. Avšak tento scenár nemusí byť prípustný. Preto budeme hľadať *najhorší prípustný scenár*, čiže taký vektor požiadaviek $\mathbf{d}^h$, ktorého zložky $d_j^h \in \{d_j^k, k = 0,1,\dots,s\}$ budú čo najväčšie, resp. ich súčet bude čo najväčší.

Použijeme rozhodovacie premenné t_{ijk}, kde

$$t_{ijk} = \begin{cases} 1, & \text{ak vozidlo } i \text{ obslúži požiadavku } d_j^k, \\ 0, & \text{inak.} \end{cases} \tag{56}$$

Našu úlohu nájsť najhorší prípustný scenár sformulujeme ako nasledujúcu úlohu bivalentného lineárneho programovania:

Model RVRP 3

Maximalizujte

$$\sum_{i=1}^{p}\sum_{j=1}^{n}\sum_{k=0}^{s} t_{ijk} d_j^k, \tag{57}$$

za podmienok

$$\sum_{j=1}^{n}\sum_{k=1}^{s} t_{ijk} \geq 1, \qquad \forall i \in \{1,\dots,p\}, \tag{58}$$

$$\sum_{i=1}^{p}\sum_{k=0}^{s} t_{ijk} = 1, \qquad \forall j \in \{1,\dots,n\}, \tag{59}$$

$$\sum_{j=1}^{n}\sum_{k=0}^{s} t_{ijk} d_j^k \leq Q, \qquad \forall i \in \{1,\dots,p\}, \tag{60}$$

$$t_{ijk} \in \{0,1\}, \qquad \forall i \in \{1,\dots,p\}, \forall j \in \{0,\dots,n\}, \forall k \in \{0,\dots,s\}. \tag{61}$$

Účelová funkcia (57) maximalizuje celkovú sumu uspokojených požiadaviek. Podmienky (58) zabezpečujú, že každé vozidlo bude využité, t. j. obslúži aspoň jedného zákazníka. Skupina podmienok (59) zas stanovuje, že u každého zákazníka bude obslúžená požiadavka z práve jedného scenára a práve jedným vozidlom. Podmienky (60) sú kapacitné podmienky zaručujúce, že nebude prekročená kapacita žiadneho vozidla a (61) sú obligatórne podmienky.

Stratégia najmenšej nutnej kapacity
Príčinou toho, že scenár $\mathbf{d}^m$ nie je prípustný, je nepostačujúca celková kapacita vozidiel pQ. Jednou z možností, ako sa vysporiadať s týmto problémom, je použitie lexikografického modelu LCVRP, pomocou ktorého nájdeme optimálne riešenie inštancie so scenárom $\mathbf{d}^m$ s využitím najmenšej nutnej kapacity vozidiel Q_{min}. Minimálnu potrebnú kapacitu získame riešením nasledujúcej úlohy odvodenej z modelu mBPP:

Model RVRP 4
Minimalizujte

$$q \tag{62}$$

za podmienok

$$\sum_{i=1}^{p} b_{ij} = 1, \qquad \forall j \in \{1, \dots, n\}, \tag{63}$$

$$\sum_{j=1}^{n} b_{ij} \geq 1, \qquad \forall i \in \{1, \dots, p\}, \tag{64}$$

$$\sum_{j=1}^{n} b_{ij} d_j^m \leq q, \qquad \forall i \in \{1, \dots, p\}, \tag{65}$$

$$b_{ij} \in \{0, 1\}, \qquad \forall i \in \{1, \dots, p\}, \forall j \in \{0, \dots, n\}, \tag{66}$$

kde celočíselná premenná q, $q > 0$ v účelovej funkcii (62) modeluje minimálnu nutnú kapacitu Q_{min} a bivalentné premenné b_{ij} modelujú rozhodnutie, či vozidlo i obslúži zákazníka j.

4.5.4 Výsledky experimentov

Experimenty sme vykonali so 6 inštanciami z [90] s malým počtom zákazníkov (15 až 22) a so 6 inštanciami s väčším počtom zákazníkov (33 až 100). Najprv sme tieto inštancie modifikovali tak, aby v nich boli zahrnuté neisté požiadavky. Z deterministických (očakávaných) požiadaviek sme pre každú inštanciu vygenerovali 4 množiny $\mathscr{U}_d$ obsahujúce po 5 scenárov požiadaviek so zadanou maximálnou odchýlkou ε, konkrétne $\varepsilon \in \{0,05; 0,1; 0,15; 0,2\}$.
Postup pri jednotlivých stratégiách
Stratégia maximálneho scenára (rvrp 1):

1. výpočet maximálneho scenára $\mathbf{d}^m$ so zložkami $d_j^m = \max\{\max_k d_j^k, d_j^0\}$ pre $j \in \{1, \dots, n\}, k \in \{1, \dots, s\}$,
2. vyriešenie inštancie CVRP so scenárom $\mathbf{d}^m$ a kapacitou Q.

Stratégia najhoršieho prípustného scenára (rvrp 2):

1. exaktný výpočet najhoršieho prípustného scenára $\mathbf{d}^h$ použitím modelu RVRP 3,
2. vyriešenie inštancie CVRP so scenárom $\mathbf{d}^h$ a kapacitou Q.

Stratégia najmenšej nutnej kapacity (rvrp 3):

1. exaktný výpočet najmenšej nutnej kapacity Q_{min} pri scenári $\mathbf{d}^m$ použitím modelu RVRP 4,
2. vyriešenie inštancie CVRP so scenárom $\mathbf{d}^m$ a kapacitou Q_{min}.

Na vyriešenie inštancií CVRP sme pri malom počte zákazníkov použili exaktnú metódu (algoritmus CVRP 3 s parametrom $\delta = 2$) a pri väčšom počte zákazníkov heuristickú metódu (algoritmus pμGA s parametrami: 50 generácií bez zlepšenia, 5000 epoch). Inicializačné riešenie pre uvedené metódy sme v stratégiách *rvrp 1* a *rvrp 3* získali pomocou Clarke-Wrightovej metódy [17] a v stratégii *rvrp 2* riešením modelu RVRP 3.
Všetky stratégie robustnej optimalizácie sme implementovali v jazyku Python 3.4 [88]. Exaktné metódy sme riešili pomocou solvera Gurobi 6.5 [89] na PC s parametrami Intel Xeon 32 jadier, 2,4 GHz, 256 GB RAM. Paralelné výpočty sme realizovali na univerzitnom klastri Klaster UNIZA s nasledovnou špecifikáciou [1]: 46 výpočtových uzlov (2 x 6 jadrový Intel Xeon, 2.27 GHz, 96 GB RAM), 2 výpočtové uzly (2 x 10 jadrový Intel Xeon, 2.27 GHz, 256 GB RAM).

[1] https://nic.uniza.sk/hpc/doku.php?id=hpc:klaster

Kvalitu nájdeného riešenia z hľadiska neuspokojených požiadaviek sme ohodnotili pomocou ukazovateľa γ, ktorý predstavuje zníženie relatívneho množstva neuspokojených požiadaviek pri použití nájdeného riešenia namiesto optimálneho deterministického riešenia (12).

Ďalším ukazovateľom, ktorý sme použili na ohodnotenie nájdeného riešenia z hľadiska ceny, je relatívne zvýšenie ceny nájdeného riešenia voči cene optimálneho riešenia (13).

4.5.5 Analýza výsledkov

Neuspokojené požiadavky

Vo všetkých prípadoch, keď bol maximálny scenár $\mathbf{d}^m$ prípustný, stratégia *rvrp 2* našla pomocou modelu RVRP 3 najhorší prípustný scenár $\mathbf{d}^h$ zhodný s maximálnym scenárom ($\mathbf{d}^h = \mathbf{d}^m$). Preto sú výsledky získané stratégiami *rvrp 1* a *rvrp 2* pri inštanciách riešených exaktne rovnaké. Pri heuristickom riešení sa výsledky líšia, čo však vyplýva z povahy heuristických metód. Obom stratégiám sa podarilo nájsť robustné riešenia, ktoré sú schopné uspokojiť všetky požiadaky zákazníkov vo všetkých scenároch. Rovnako aj riešenia nájdené stratégiou *rvrp 3* sú všetky robustné.

Inštancie, pri ktorých bol maximálny scenár neprípustný, sú v tabuľkách zvýraznené. Tieto inštancie sa vyznačujú pomerne tesným kapacitným obmedzením ($\tau \geq 0,94$). Najmenšia kapacita nutná na to, aby bol scenár $\mathbf{d}^m$ prípustný, je v týchto prípadoch $Q_{min} \geq Q$. Z tohto dôvodu stratégia *rvrp 1* zlyhala. Na rozdiel od nej, stratégie *rvrp 2* a *rvrp 3* našli riešenia, ktoré síce neboli všetky robustné ($g_r \geq 0$), ale množstvo neuspokojených požiadaviek bolo prijateľné. Heuristickou metódou obe stratégie vygenerovali len robustné riešenia ($g_r = 0$).

Zaujímavé je teda porovnanie výsledkov stratégií *rvrp 2* a *rvrp 3* z hľadiska neuspokojených požiadaviek v tých prípadoch, keď stratégia *rvrp 1* nebola úspešná. Graf 1a zobrazuje zníženie relatívneho množstva neuspokojených požiadaviek γ pri použití riešení získaných stratégiami *rvrp 2* a *rvrp 3*. Z predchádzajúcej kapitoly vieme, že čím väčšia je hodnota γ, tým je riešenie lepšie. Stratégia *rvrp 2* bola lepšia v dvoch prípadoch a *rvrp 3* v štyroch prípadoch z jedenástich. V piatich prípadoch boli rovnaké.

Cena riešenia

Aj pri tomto hľadisku porovnáme osobitne výsledky pre inštancie, pri ktorých bol maximálny scenár neprípustný. Graf 1b vizualizuje relatívne zvýšenie ceny ζ pri stratégiách *rvrp 2* a *rvrp 3*, pričom menšia hodnota ζ znamená lepšie riešenie. Stratégia *rvrp 2* bola lepšia v šiestich prípadoch, *rvrp 3* v troch a v dvoch prípadoch z jedenástich boli rovnaké.

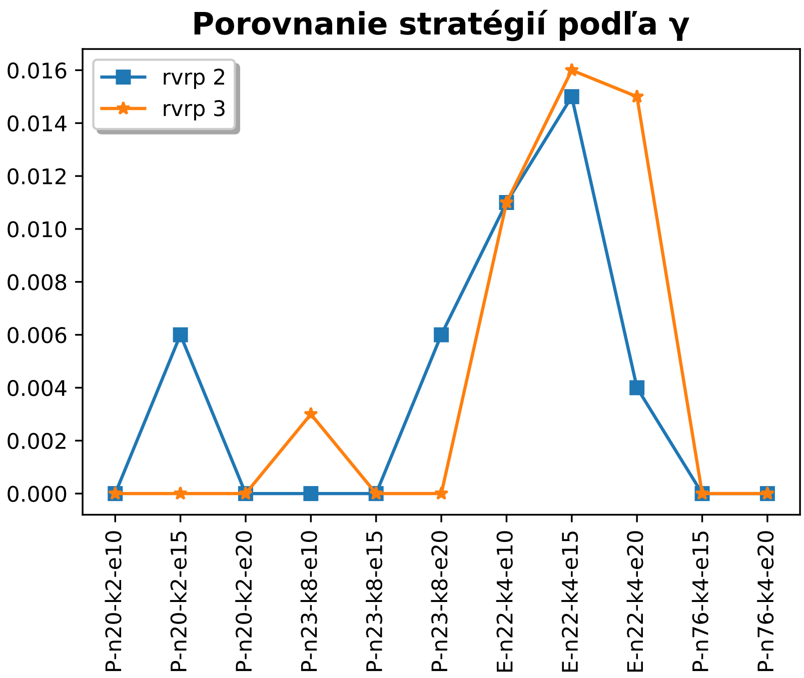
K zvýšeniu ceny o viac ako 50% došlo v riešeniach získaných stratégiou *rvrp 3* v troch prípadoch, kým u stratégie *rvrp 2* len v jednom.

Pri ostatných inštanciách sme najprv porovnali výsledky pre inštancie s malým počtom zákazníkov (riešené exaktne), a potom pre inštancie s väčším počtom zákazníkov (riešené heuristicky). Opäť platí, že exaktné riešenia získané stratégiami *rvrp 1* a *rvrp 2* sú rovnaké, preto sú v grafe 2a zobrazené len výsledky stratégií *rvrp 2* a *rvrp 3*. Heuristické výsledky všetkých troch stratégií sú rozdielne, ako môžeme vidieť v grafe 2b.

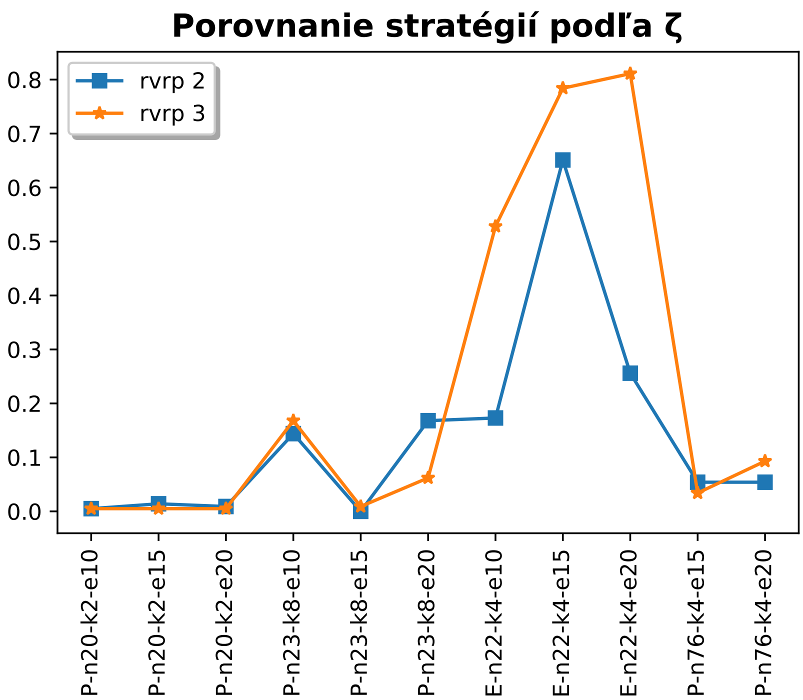
Najhoršie výsledky z hľadiska ceny dosiahla stratégia *rvrp 3*, ktorá bola najhoršia vo všetkých prípadoch okrem dvoch. Stratégia *rvrp 2* bola v porovnaní so stratégiou *rvrp 1* lepšia v deviatich prípadoch, kým *rvrp 1* v ôsmich z dvadsiatich dvoch prípadov riešených heuristicky. Vo zvyšných prípadoch boli rovnaké. Môžeme teda hovoriť o približne rovnakej úspešnosti. Riešenia s relatívnym zvýšením ceny väčším ako 50% sme dostali len stratégiou *rvrp 3*, a to v dvoch prípadoch. V riešeniach stratégií *rvrp 1* a *rvrp 2* sa cena zvýšila nanajvýš o 15,4%.

Cena versus neuspokojené požiadavky

Pri rozhodovaní, ktoré riešenie je lepšie, treba nájsť vyvážený kompromis medzi týmito dvomi hľadiskami. Napríklad pri inštancii E-n22-k4-e15 stratégia *rvrp 2* našla riešenie s cenou 619 a množstvom neuspokojených požiadaviek 13, naopak stratégia *rvrp 3* našla riešenie s väčšou cenou 669, ale so žiadnymi neuspokojenými požiadavkami. Na jednej strane, nižšia cena znamená väčší aktuálny profit, na druhej strane menej neuspokojených požiadaviek vedie k väčšej spokojnosti zákazníkov a tým aj k vyššiemu profitu v budúcnosti.

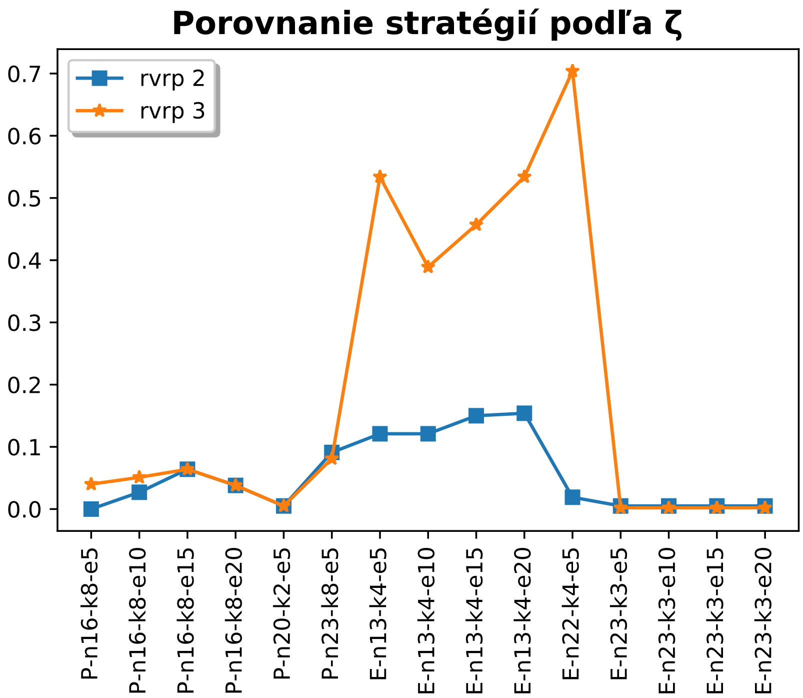


(a) Zníženie relatívneho množstva neuspokojených požiadaviek γ

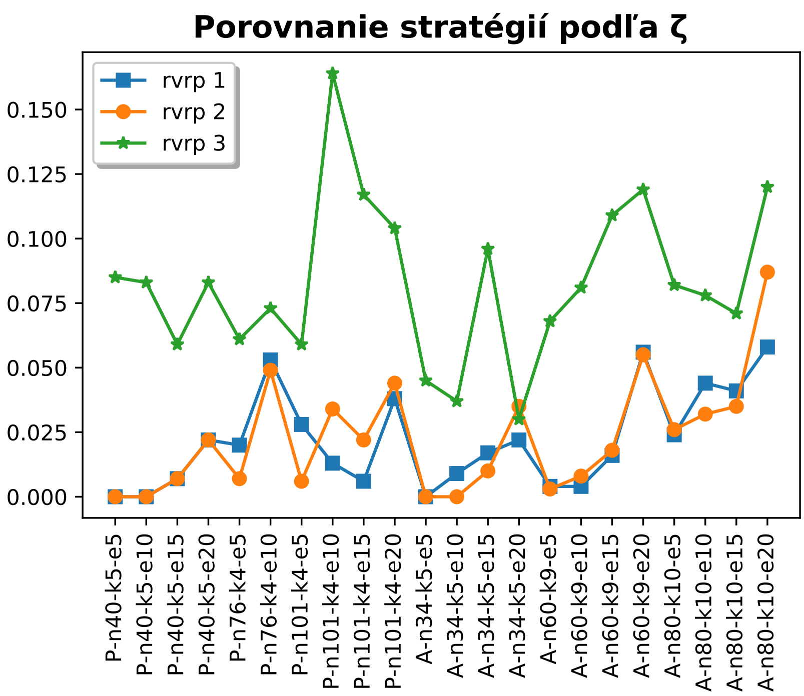


(b) Relatívne zvýšenie ceny ζ

Obr. 1: Inštancie s neprípustným maximálnym scenárom



(a) Inštancie riešené exaktne



(b) Inštancie riešené heuristicky

Obr. 2: Inštancie s prípustným maximálnym scenárom

5 ZÁVER

V našej práci sme sa zaoberali riešením distribučných úloh s neistými parametrami pomocou robustnej optimalizácie. Vzhľadom na široký rozsah tejto problematiky sme sa zamerali na kapacitnú okružnú dopravnú úlohu s neistými požiadavkami zákazníkov. Cieľom bolo navrhnúť robustné modely na riešenie tejto úlohy.

V úvode sme opísali teoretický základ s účelom definovať a vysvetliť základné pojmy skúmanej problematiky. Druhá kapitola obsahuje prehľad exaktných a heuristických metód používaných pri riešení CVRP. Osobitná časť tejto kapitoly je venovaná CVRP s neistými parametrami a zahŕňa prehľad metód so stochastickým i robustným prístupom. Zdrojom bola prevažne zahraničná literatúra, nakoľko literatúra venovaná problematike robustnej optimalizácie v slovenskom, resp. českom jazyku prakticky neexistuje.

Ďalej nasleduje opis zvolenej metodiky práce a metód skúmania. Opísali sme spôsob modelovania neistých požiadaviek zákazníkov pomocou diskrétnych scenárov a dve stratégie na nájdenie najhoršieho scenára. Navrhli sme vhodnú deterministickú formuláciu CVRP, ktorú sme potom použili pri odvodení robustnej formulácie. Taktiež sme predstavili dve exaktné a jednu heuristickú metódu na riešenie CVRP, aby sme ich neskôr využili pri riešení robustnej CVRP. Okrem toho sme objasnili kritériá, podľa ktorých sme posudzovali úspešnosť navrhnutých robustných stratégií. Táto kapitola obsahuje aj špecifikáciu testovacích inštancií a softvérových nástrojov použitých pri našich experimentoch.

V kapitole venovanej výskumu sme prezentovali náš prístup k riešeniu robustnej CVRP s neistými požiadavkami zákazníkov. Najskôr sme navrhli a implementovali nový model CVRP 2 na riešenie CVRP založený na tokovej formulácii, ktorý sa vyznačuje polynomiálnym počtom anticyklických podmienok. Tento model je formuláciou vhodnou pre odvodenie takej formulácie RVRP, ktorú nebude príliš zložité riešiť.
Aby sme boli schopní v prijateľnom čase exaktne riešiť aj úlohy s väčším počtom zákazníkov, vyvinuli sme efektívnu exaktnú metódu CVRP 3 postavenú na koncepte k-optimálnosti. Naša metóda iteratívne zlepšuje počiatočné prípustné riešenie nahradením niektorých hrán inými. Zlepšenie riešenia pritom dosiahne exaktným riešením jednoduchšieho problému, vďaka čomu nájde optimálne riešenie omnoho rýchlejšie než CVRP 2. Navyše, ak by bol výpočet príliš zdĺhavý, môžeme ho prerušiť a získame tak k-optimálne riešenie.
Vzhľadom na to, že CVRP patrí do skupiny NP-ťažkých problémov, nie je možné pomocou tohto modelu exaktne riešiť úlohy s veľkým rozsahom, a preto sa na ich riešenie využívajú heuristické a metaheuristické metódy. My sme na tento účel navrhli a experimentálne overili paralelný mikrogenetický algoritmus pμGA, ktorý profituje z výhod mikrogenetického algoritmu a paraleného výpočtu. Pri paralelizácii mikrogenetického algoritmu sme vychádzali z princípu hrubozrnného modelu, ale namiesto podpopulácií veľkej populácie sme použili mikropopulácie a proces migrácie sme nahradili dosadením nasadeného jedinca.
Nový prístup k riešeniu CVRP predstavuje návrh lexikografického modelu LCVRP, ktorého cieľom je nájsť riešenie s minimálnou cenou, pričom bude využitá len najmenšia nutná kapacita vozidiel. Takéto riešenie má síce väčšiu cenu v porovnaní s cenou riešenia pri využití plnej kapacity, ale na druhej strane nevyužitá kapacita vozidiel môže slúžiť ako rezerva pre prípad, že dôjde k neočakávanému nárastu požiadaviek zákazníkov. Vďaka tomu je riešenie LCVRP do určitej miery robustné.
V poslednej fáze výskumu sme sa venovali robustnej CVRP s neistými požiadavkami zákazníkov, ktorej cieľom je nájsť také riešenie, ktoré bez ohľadu na to, aký scenár nastane, uspokojí všetkých zákazníkov. Niekedy však také riešenie neexistuje, a preto sme v našom prístupe k riešeniu tejto úlohy hľadali riešenie s minimálnymi neuspokojenými požiadavkami a minimálnou cenou. Navrhli sme tri robustné stratégie na riešenie robustnej CVRP s neistými požiadavkami zákazníkov – stratégiu maximálneho scenára *rvrp 1*, stratégiu najhoršieho prípustného scenára *rvrp 2* a stratégiu najmenšej nutnej kapacity *rvrp 3*. Experimentálne výsledky dosiahnuté aplikáciou týchto stratégií sme porovnali z hľadiska ceny i z hľadiska neuspokojených požiadaviek. Na základe analýzy výsledkov bola z hľadiska vyváženého kompromisu medzi cenou a neuspokojenými požiadavkami najúspešnejšou stratégia *rvrp 2*.

Na záver zhrnieme vedecký a praktický prínos práce.

5.1 Vedecký prínos práce

Za hlavný vedecký prínos tejto práce možno považovať:

- prehľad súčasného stavu riešenia distribučných úloh s využitím robustnej optimalizácie,
- teoretický návrh prístupu k riešeniu problematiky robustných modelov,
- softvérové riešenia a verifikáciu navrhnutých prístupov,

- inovatívne metódy na riešenie deterministickej i robustnej CVRP, využiteľné aj v iných oblastiach kombinatorickej optimalizácie:
  - novú iteratívnu exaktnú metódu na riešenie CVRP,
  - efektívny paralelný mikrogenetický algoritmus na riešenie CVRP,
  - lexikografický model CVRP,
  - nový prístup k riešeniu robustnej CVRP s neistými požiadavkami zákazníkov.

5.2 Praktický prínos práce

Navrhnuté modely nájdu praktické využitie v oblastiach ako sú napríklad:

- miestna preprava tovaru (zásobovanie),
- kuriérske služby,
- mestský zber komunálneho a separovaného odpadu,
- mobilné gastronomické služby (cateringový servis),
- školská autobusová doprava,
- zvoz recyklovateľných obalov.

6 SUMMARY

Distribution problems are concerned with the transfer of goods or services between manufacturing facilities, distribution centers and customers. Usually, we assume that input data (transportation costs, service times, demands etc.) is precisely known at the time ofproblem solving. However, in many real-life applications data are subject to uncertainty due to their random nature, measurement errors or other reasons. This random behavior of the problem data could cause the proposed feasible solution to become infeasible. Robust optimisation seeks the solution that is immune to this uncertainty under assumption that probability distribution is unknown. Instead we assume that the uncertain problem parameters belong to bounded uncertainty set. The objective of robust optimisation is to find the best solution which is feasible for any realization of the data from the given uncertainty set. In this work we consider the robust capacitated vehicle routing problem in which a set of uncertain customers' demands has to be served by a fleet of homogenous vehicles departing from a depot. We proposed three robust strategies which aim to minimise transportation costs and unsatisfied demands for the specific uncertainty set. Our experimental results show that obtained solutions can minimise unmet demands while incurring a small extra cost over deterministic optimal solution.

Keywords: robust optimisation, capacitated vehicle routing problem, uncertain customers' demands, integer and mixed linear programming, parallel micro genetic algorithm

Literatúra

[1] ALAJMI, A., WRIGHT, J.: Selecting the most efficient genetic algorithm sets in solving unconstrained building optimization problem. International Journal of Sustainable Built Environment, vol. 3, no. 1, 18–26, 2014.

[2] ALBA, E., DORRONSORO, B.: Solving the vehicle routing problem by using cellular genetic algorithms, in: Evolutionary Computation in Combinatorial Optimization (EvoCOP), Lecture Notes in Computer Science, vol. 3004, Springer-Velag, Berlin, 11–20, 2004.

[3] AUGERAT, P., BELENGUER, J., BENAVENT, E., CORBERÁN, A., NADDEF, D., RINALDI, G.: Computational results with a branch and cut code for the capacitated vehicle routing problem. Tech. Rep. 949-M, Universite Joseph Fourier, Grenoble, France, 1995.

[4] BALDACCI R., HADJICONSTANTINOU E., MIGNOZZI A.: An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. Oper. Res. 52:723–738, 2004.

[5] BEASLEY, J. E.: Route first-cluster second methods for vehicle routing. Omega, 11(4), 403–408, 1983.

[6] BEN-TAL, A., NEMIROVSKI, A.: Robust convex optimization. Mathematics of Operations Research, 23(4), 769–805, 1998.

[7] BEN-TAL, A., NEMIROVSKI, A.: Robust Solutions of Uncertain Linear Programs, OR Letters v. 25, 1-13, 1999.

[8] BERTSIMAS, D.: Probabilistic combinatorial optimization problems. Ph.D. thesis, Massachusetts Institute of Technology, Dept. of Mathematics, 1988.

[9] BERTSIMAS, D., BROWN, D. B., CARAMANIS, C.: Theory and applications of robust optimization. SIAM Rev. 53(3), 464–501, 2011.

[10] BERTSIMAS, D., SIM, M.: Robust discrete optimization and network flows. Mathematical Programming Series B, 98(1-3), 49–71, 2003.

[11] BERTSIMAS, D., SIMCHI-LEVI, D.: A new generation of vehicle routing research: robust algorithms, addressing uncertainty. Operations Research, 44(2), 286–304, 1996.

[12] BERTSIMAS, D.: A vehicle routing problem with stochastic demand. Operations Research, 40(3), 574–585, 1992.

[13] BORČINOVÁ, Z.: Two models of the capacitated vehicle routing problem. In: Croatian operational research review. - ISSN 1848-0225. - Vol. 8, no. 2, s. 463-469, 2017.

[14] BORČINOVÁ, Z.: Solving the capacitated vehicle routing problem using a parallel micro genetic algorithm. In: 2018 IEEE Workshop on Complexity in Engineering (COMPENG) - ISBN 978-1-5386-5338-8. - s. [1-6], 2018.

[15] BORČINOVÁ, Z., PEŠKO, Š.: New exact iterative method for the capacitated vehicle routing problem. In: Communications : scientific letters of the University of Žilina. - ISSN 1335-4205. - Vol. 18, no. 3, s. 19-21, 2016.

[16] BORČINOVÁ, Z., PEŠKO, Š.: The lexicographical capacitated vehicle routing problem. In: SOLI 2017 : IEEE international conference on Service operations and logistics, and informatics : September 18-20, 2017 Bari, Italy - ISBN 978-1-5090-5847-1. - [4] s., 2017.

[17] CLARKE, G., WRIGHT, J. W.: Scheduling of vehicles from a central depot to a number of delivery points. Operations Research, 12(4), 568–581, 1964.

[18] COCO, A. A., SOLANO-CHARRIS, E. L., SANTOS, A. C., PRINS, C., NORONHA, T.: Robust optimization criteria: state-of-the-art and new issues. Technical Report UTT-LOSI-14001. ISSN:2266-5064. Université de Technologie de Troyes, 2014a.

[19] CRAINIC, T.: Parallel solution methods for vehicle routing problems. In: Golden, B., Raghavan, S., Wasil, E. eds., The Vehicle Routing Problem: Latest Advances and New Challenges. Springer, New York, 171–198, 2008.

[20] DANTZIG, G. B.: Linear Programming under uncertainty. Management Science, 1(3-4), 197–206, 1955.

[21] DANTZIG, G. B., RAMSER, J. H.: The truck dispatching problem. Management Science, 6(1), 80–91, 1959.

[22] DORIGO, M., BLUM, C.: Ant colony optimization theory: A survey. Theoret Comput Sci, 344 (2–3), pp. 243–278, 2005.

[23] DESSOUKY, M., ORDÓÑEZ, F., JIA, H., SHEN, Z. Rapid distribution of medical supplies. Patient flow: Reducing delay management in health care systems. Springer, USA, 309–339, 2005.

[24] DRESHER, M.: Games of Strategy. Prentice-Hall, 1961.

[25] ERBAO, C., MINGYONG, L.: A hybrid differential evolution algorithm to vehicle routing problem with fuzzy demands. Journal of Computational and Applied Mathematics. 231. 302-310, 2009.

[26] ERBAO, C., MINGYONG, L., HONGMING, Y.: Open vehicle routing problem with demand uncertainty and its robust strategies. Expert Systems with Applications: An International Journal. 41. 3569-3575, 2014.
[27] FUKASAWA, R., LONGO, H., LYSGAARD, J., ARAGÃO, M. P., REIS, M., UCHOA, E., WERNECK, R. F.: Robust Branch-and-Cut-and-Price for the Capacitated Vehicle Routing Problem. Mathematical Programming Series A, 106(3), 491–511, 2006.
[28] GABREL, V., MURAT, C., THIELE, A.: La pw-robustesse: pourquoi un nouveau critère de robustesse et comment l'appliquer? In: 14ème congrès de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF), 2013.
[29] GENDREAU, M., LAPORTE, G., SEGUIN, R.: An exact algorithm for the vehicle routing problem with stochastic demands and customers. Transportation Science, 29(2), 143–155, 1995.
[30] GENDREAU, M., LAPORTE, G., SEGUIN, R.: Stochastic vehicle routing. European Journal of Operational Research, 88(1), 3–12, 1996.
[31] GENDREAU, M., POTVIN, J. Y.: Handbook of Metaheuristics, Second Edition, Springer, New York, 2010
[32] GILLETT, B. E., MILLER, L. R.: A heuristic algorithm for the Vehicle-Dispatch problem. Operations Research, 22(2), 340–349, 1974.
[33] GOUNARIS, C. E., WIESEMANN, W., FLOUDAS, C. A.: The Robust Capacitated Vehicle Routing Problem Under Demand Uncertainty. Operations Research, 61(3), 677–693, 2013.
[34] GOUNARIS, E. C., REPOUSSIS, P. P., TARANTILIS, C. D., WIESEMANN, W., FLOUDAS, C. A.: An Adaptive Memory Programming framework for the Robust Capacitated Vehicle Routing Problem. Transportation Science. doi:10.1287/trsc.2014.055, 2014.
[35] GOLDBERG, D. E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, MA, 1989.
[36] GOLDBERG, D. E.,KORB, B., DEB, K.: Messy genetic algorithms: Motivation, analysis, and first results. Complex Systems, 3, 493–530, 1990.
[37] GOLDBERG, D. E., LINGLE, R.: Alleles Loci and the TSP. Proceedings of the First International Conference on Genetic Algorithms and Their Application, Lawrence Erolaum, New Jersy, 54–159, 1985.
[38] GROËR, Ch., GOLDEN, B., WASIL, E. A.: A Parallel Algorithm for the Vehicle Routing Problem. INFORMS Journal on Computing. 23. 315–330, 2011.
[39] CHEPURI, K., HOMEM de MELLO, H.. Solving the vehicle routing with stochastic demands using the cross entropy method. Annals of Operations Research, 135:153–181, 2005.
[40] CHRISTIANSEN, C., LYSGAARD, J.: A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands, Operation Research Letters, 35(6), 773–781, 2007.
[41] HAN, J., LEE, C., PARK, S.: A Robust Scenario Approach for the Vehicle Routing Problem with Uncertain Travel Times. Transportation Science, 48(3), 373–390, 2013 .
[42] HOLLAND, H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. University of Michigan Press, 1975.
[43] CHRISTOFIDES, N., EILON, S.: An algorithm for the vehicle dispatching problem. Journal of the Operational Research Society, Vol. 20, No. 3, pp. 309–318, 1969.
[44] CHRISTOFIDES, N., MIGNOZZI, A., TOTH, P.: The vehicle routing problem. Pages 315–338 of: Combinatorial Optimization. Wiley, Chichester, 1979.
[45] JANÁČEK, J., JÁNOŠÍKOVÁ, Ľ., KOHÁNI, M.: Modelovanie a optimalizácia, EDIS vydavateľstvo ŽU, 2013.
[46] JUAN, A., GRASMAN, S., FAULIN, J., RIERA, D., MÉNDEZ, C., RUIZ, B.: Applying Simulation and Reliability to Vehicle Routing Problems with Stochastic Demands. CEUR Workshop Proceedings, 589, 2009.
[47] KALAÏ, R., LAMBORAY, C., VANDERPOOTEN, D.: Lexicographic α-robustness: An alternative to min-max criteria. European Journal of Operational Research, 220(3), 722–728, 2012.
[48] KARA, I.: Tightening Bounding Constraints of the Miller-Tucker-Zemlin Based Formulation of the Capacitated Vehicle Routing Problems and Some Extensions, Proc. of the 2nd intern. conference on Manufacturing Engineering, Quality and Production Systems, edited by C. Panait et al., WSEAS Press, Constantza, pp.137-142, 2010.
[49] KENYON, A. S., MORTON, D.: Stochastic vehicle routing with random travel times. Transportation Science, 37(1), 69–82, 2003.
[50] KOUVELIS, P., YU, G.: Robust discrete optimization and its applications. Kluwer Academic Publishers, Norwell, MA, 1997.
[51] KRISHNAKUMAR, K.: Micro-Genetic Algorithms for Stationary and Non-Stationary Function Optimization. SPIE Proceedings: Intelligent Control and Adaptive Systems, Vol. 1196, 289–296, 1989.

[52] L. GORISSEN, B., YANIKOĞLU, İ, den HERTOG, D.: A Practical Guide to Robust Optimization. Omega. 53, 124-137, 2015.
[53] LAPORTE, G.: Fifty Years of Vehicle Routing. Transportation Science, 43(4), 408–416, 2009.
[54] LAPORTE, G., LOUVEAUX, F., HAMME, L. V.: An Integer L-Shaped Algorithm for the Capacitated Vehicle Routing Problem with Stochastic Demands. Operations Research, 50(3), 415–423, 2002.
[55] LEE, C., LEE, K., PARK, S.: Robust vehicle routing problem with deadlines and travel time/demand uncertainty. Journal of the OR Society, 63, 1294–1306, 2012 .
[56] LIN, S., KERNIGHAN, B. W.: An Effective Heuristic Algorithm for the Traveling-Salesman Problem. Operations Research. 21 (2): 498–516, 1973.
[57] LI, X., TIAN, P., LEUNG, S. C.H.: Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm, International Journal of Production Economics, Elsevier, vol. 125(1), 137-145, 2010.
[58] LYSGAARD, J., LETCHFORD, A. N., EGLESE, R. W.: A new branch-and-cut algorithm for the capacitated vehicle routing problem. Mathematical Programming, 100(2), 423–445, 2004.
[59] MARKOVIC, H., CAVAR, I., CARIC, T.: Using data mining to forecast uncertain demands in stochastic vehicle routing problem. In Proceedings of the 13th International Symposium on Electronics in Transport (ISEP), Slovenia, 2005.
[60] MARTELLO, S.,TOTH, P.: Knapsack Problems: Algorithms and Computer Implementations, Wiley, Chichester, 1990.
[61] MARTIN, W. N., LIENING, J., COHOON, J. P.: Island (migration) models: Evolutionary algorithms based on punctuated equilibria. Evolutionary Computation 2: Advanced Algorithms and Operators, T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds. London, U.K.: IOP Publishing, vol. 2, 101–124, 2000.
[62] MESTER, D., BRÄYSY, O.: Active-guided evolution strategies for large-scale capacitated vehicle routing problems. Computers & Operations Research, 34(10), 2964–2975, 2007.
[63] MOGHADDAM, B. F., RUIZ, R., SADJADI, S. J.: Vehicle routing problem with uncertain demands: An advanced particle swarm algorithm. Computers & Industrial Engineering, 62(1), 306–317, 2012 .
[64] MOLE, R. H., JAMESON, S. R.: A Sequential Route-Building Algorithm Employing a Generalised Savings Criterion. Operational Research Quarterly, 27(2), 503–511, 1976.
[65] NEUMANN, J. V.: Zur Theorie der Gesellschaftsspiele. Mathematische Annalen, 100(1), 295–320, 1928.
[66] OCHI, L. S., VIANNA, D. S., DRUMMOND, L. M. A., VICTOR, A. O.: A Parallel Evolutionary Algorithm for the Vehicle Routing Problem with Heterogeneous Fleet. Future Generation Computer Systems, 14(3): 285–292, 1998.
[67] ORDÓÑEZ, F.: Robust Vehicle Routing. Chap. 8, pages 153–178, 2010.
[68] OSMAN, I.: Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. Annals of Operations Research, 41(4), 421–451, 1993.
[69] PALÚCH, S., PEŠKO, Š.: Kvantitatívne metódy v logistike, EDIS vydavateľstvo ŽU, 2006.
[70] PECIN, D., PESSOA, A., POGGI, M., UCHOA, E.: Improved Branch-Cut-and-Price for Capacitated Vehicle Routing. Pages 393–403 of: Integer Programming and Combinatorial Optimization. Lecture Notes in Computer Science, vol. 8494. Springer International Publishing, 2014.
[71] PRINS, C.: A simple and effective evolutionary algorithm for the vehicle routing problem. Computers & Operations Research, 31(12), 1985–2002, 2004.
[72] PRINS, C., LACOMME, P., PRODHON, C.: Order-first split-second methods for vehicle routing problems: A review. Transportation Research Part C, 40, 179–200, 2014.
[73] RIEMANN, M., DOERNER, K., HARTL, R. F.: Analyzing a Unified Ant System for the VRP and Some of its Variants. Applications of Evolutionary Computing, 2611:300– 310, 2003.
[74] ROY, B.: Robustness in operational research and decision aiding: A multi-faceted issue. European Journal of Operational Research, 200(3), 629–638, 2010.
[75] SECOMANDI, N., MARGOT, F.: Reoptimization Approaches for the Vehicle-Routing Problem with Stochastic Demands. Operations Research, 57(1), 214–230, 2009.
[76] SEKAJ, I.: Evolučné výpočty a ich využitie v praxi, IRIS, Bratislava, 2005.
[77] SOLANO-CHARRIS, E. L., PRINS, C., SANTOS, A. C.: Local Search Based Metaheuristics for the Robust Vehicle Routing Problem with Discrete Scenarios. Applied Soft Computing, 32, 518–531, 2015.
[78] SUN, L.: A New Robust Optimization Model for the Vehicle Routing Problem with Stochastic Demands. Journal of Interdisciplinary Mathematics, 17(3), 287–309, 2014.
[79] SUN, L., WANG, B.: Robust Optimisation Approach for Vehicle Routing Problems with Uncertainty, Mathematical Problems in Engineering, vol. 2015, Article ID 901583, 8 pages, 2015.
[80] SUNGUR, I., ORDÓÑEZ, F., DESSOUKY, M.: A robust optimization approach for the capacitated vehicle routing problem with demand uncertainty. IIE Transactions, 40(5), 509–523, 2008 .
[81] SUNGUR, I., REN, Y., ORDóñEZ, F., DESSOUKY, M., ZHONG, H.: A model and algorithm for the courier delivery problem with uncertainty. Transportation Science, 44(2), 193–205, 2010.

[82] TOKLU, N. E., MONTEMANNI, R., GAMBARDELLA, L. M.: An ant colony system for the capacitated vehicle routing problem with uncertain travel costs. Pages 32–39 of: IEEE Symposium on Swarm Intelligence (SIS), 2013.
[83] TOTH, P., VIGO, D. : Vehicle routing: Problems, Methods and Applications, Second Edition. Philadelphia: SIAM, 2014.
[84] TOTH, P., VIGO, D.: Models, Relaxations and Exact Approaches tor Capacitated Vehicle Routing Problem, Discrete Applied Mathematics, 123: 487-512, 2002.
[85] YANG, W., MATHUR, K., BALLOU, R.: Stochastic vehicle routing problem with restocking. Transportation Science, 34(1):99–112, 2000.
[86] WATERS, C.: Vehicle-scheduling problems with uncertainty and omitted customer. Journal of the Operational Research Society, 4(12), 1099–1108, 1989 .
[87] ZACHARIADIS, E. E., KIRANOUDIS, C. T.: A strategy for reducing the computational complexity of local search-based methods for the vehicle routing problem. Computers & operations research, 37 (12), 2089-2105, 2010.
[88] The Python Language Reference, 1990 - 2016, Python Software Foundation, `http://docs.python.org/3/`.
[89] Gurobi Optimizer Reference Manual, version 6.5, 2016, Gurobi Optimization, Inc., `http://www.gurobi.com`.
[90] Branch Cut and Price Resource Web, Vehicle Routing Data Sets USA, `http://www.coin-or.org/SYMPHONY/branchandcut/VRP/data/`.
[91] Matplotlib 3.0, 2002 - 2018,The Matplotlib development team,`https://matplotlib.org/`.
[92] Message Passing Interface (MPI) standard for the Python programming language, `https://bitbucket.org/mpi4py/mpi4py`.
[93] NetworkX 2.2, 2004 - 2018, NetworkX Developers, `https://networkx.github.io/documentation/stable/`

Vlastné publikácie

BORČINOVÁ, Z.: Two models of the capacitated vehicle routing problem. In: Croatian operational research review. - ISSN 1848-0225. - Vol. 8, no. 2, s. 463-469, 2017.

BORČINOVÁ, Z.: Solving the capacitated vehicle routing problem using a parallel micro genetic algorithm. In: 2018 IEEE Workshop on Complexity in Engineering (COMPENG) - ISBN 978-1-5386-5338-8. - s. [1-6], 2018.

BORČINOVÁ, Z., PEŠKO, Š.: New exact iterative method for the capacitated vehicle routing problem. In: Communications : scientific letters of the University of Žilina. - ISSN 1335-4205. - Vol. 18, no. 3, s. 19-21, 2016.

BORČINOVÁ, Z., PEŠKO, Š.: The lexicographical capacitated vehicle routing problem. In: SOLI 2017 : IEEE international conference on Service operations and logistics, and informatics : September 18-20, 2017 Bari, Italy - ISBN 978-1-5090-5847-1. - [4] s., 2017.