

ŽILINSKÁ UNIVERZITA V ŽILINE

**AUTOREFERÁT
DIZERTAČNEJ PRÁCE**

Žilina apríl 2021

Ing. Maroš Janovec

Žilinská univerzita v Žiline
Fakulta riadenia a informatiky

Ing. Maroš Janovec

Autoreferát dizertačnej práce

**Metódy pre riešenie úlohy návrhu turnusov pre elektrobusy
vo verejnej doprave**

na získanie akademického titulu „**philosophiae doctor**“ (v skratke **PhD.**)
v študijnom programe doktorandského štúdia

inteligentné informačné systémy

v študijnom odbore:
informatika

Žilina apríl 2021

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

Predkladateľ: Ing. Maroš Janovec
Katedra matematických metód a operačnej analýzy
Fakulta riadenia a informatiky
Žilinská univerzita v Žiline

Školiteľ: doc. Ing. Michal Koháni, PhD.
Katedra matematických metód a operačnej analýzy
Fakulta riadenia a informatiky
Žilinská univerzita v Žiline

Oponent: doc. Ing. Dušan Teichmann, Ph.D.
Ústav logistiky a managementu dopravy
Fakulta dopravní
České vysoké učení technické v Praze

Oponent: prof. Ing. Ivan Brezina, CSc.
Katedra operačného výskumu a ekonometrie
Fakulta hospodárskej informatiky
Ekonomická univerzita v Bratislave

Autoreferát bol rozoslaný dňa:

Obhajoba dizertačnej práce sa koná dňa **19.8.2021** o **9:00** h. pred komisiou pre obhajobu dizertačnej práce schválenou pracovnou skupinou odborovej komisie v študijnom odbore **informatika v študijnom programe inteligentné informačné systémy**, vymenovanou dekanom Fakulty riadenia a informatiky Žilinskej univerzity v Žiline dňa

prof. Ing. Karol Matiaško, PhD.
predseda pracovnej skupiny odborovej komisie
v študijnom odbore **informatika**
v študijnom programe **inteligentné informačné systémy**

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

Obsah

1	Úvod do problematiky	4
2	Definícia úlohy.....	6
3	Výsledky práce.....	7
3.1	Testovacie dáta	7
3.2	Exaktné metódy na riešenie úlohy návrhu turnusov el. autobusov	8
3.2.1	<i>Matematický model úlohy návrhu turnusov elektrických autobusov</i>	<i>8</i>
3.2.2	<i>Výsledky experimentov</i>	<i>12</i>
3.3	Metóda redukcie vstupov s využitím riešenia pomocou IP solvera	13
3.3.1	<i>Výber potenciálnych dvojíc spojov na spojenie</i>	<i>13</i>
3.3.2	<i>Výber najvhodnejšej dvojice spojov na spojenie.....</i>	<i>13</i>
3.3.3	<i>Výsledky experimentov</i>	<i>14</i>
3.4	Metóda založená na metóde generovania stĺpcov	15
3.4.1	<i>Riešenie hlavného problému</i>	<i>16</i>
3.4.2	<i>Riešenie pod-problému.....</i>	<i>16</i>
3.4.3	<i>Výsledky experimentov</i>	<i>18</i>
3.5	Riešenie pomocou algoritmu Grouping Genetic Algorithm (GGA)	19
3.5.1	<i>Reprezentácia riešenia a návrh operátorov.....</i>	<i>20</i>
3.5.2	<i>Výsledky experimentov</i>	<i>22</i>
4	Záver	23
	Literatúra	24
	Vlastné publikácie	26

1 Úvod do problematiky

V posledných rokoch začínajú prechádzať do popredia elektrické vozidlá. Keďže nemajú žiadne emisie CO₂ a zároveň majú nízke prevádzkové náklady, začínajú byť zaujímavou alternatívou ku konvenčným vozidlám. Tento trend sa týka aj verejnej dopravy. V blízkej dobe môžeme očakávať výmenu flotily konvenčných vozidiel za flotilu elektrických vozidiel. Vzhľadom na to, že špecifikácia elektrických vozidiel sa v niektorých ohľadoch značne líši od konvenčných vozidiel, je nutné riešiť problém návrhu dopravného systému pre elektrické vozidlá inak ako pre konvenčné vozidlá.

Úloha, ktorou sa zaoberá táto práca, je úloha návrhu turnusov elektrických vozidiel. Keďže je táto úloha pomerne všeobecná a nakoľko rôzne typy vozidiel majú rôzne vlastnosti, budú sa líšiť aj problémy týkajúce sa konkrétneho typu vozidla. Keďže najpoužívanejším typom vozidla vo verejnej doprave je konvenčný autobus, je možné ho nahradiť elektrickým autobusom, ktorý sa konvenčnému autobusu z pohľadu použitia veľmi podobá. Verejná doprava je zároveň miestom, kde by mohli elektrické autobusy priniesť značné úspory nákladov na prevádzku a zároveň pomohli znížiť emisie a tak zlepšiť ovzdušie v mestách. Našou úlohou teda bude riešiť úlohu návrhu turnusov elektrických autobusov vo verejnej doprave.

Základom úlohy je základná úloha návrhu turnusov. Pre popísanie základnej úlohy návrhu turnusov s jedným depom a jedným typom vozidiel existuje niekoľko matematických modelov v závislosti od pohľadu na daný problém. My si uvedieme jeden z nich. K tomu máme zadefinované nasledujúce množiny. N je množina všetkých spojov, D_0 je ranné depo, D_n je večerné depo, K je množina všetkých vozidiel, $F(i)$ je množina možných nasledovníkov spoja $i \in N$ z časového hľadiska a $B(i)$ je množina možných predchodcov spoja $i \in N$ z časového hľadiska. Okrem týchto premenných si definujeme premennú x_{ij}^k , ktorý predstavuje rozhodnutie, či obslúžime spoj j hneď po spoji i vozidlom k , alebo nie.

$$\min \sum_{k \in K} \sum_{j \in F(D_0)} x_{D_0 j}^k \quad (1.1)$$

$$\text{z. p.} \quad \sum_{k \in K} \sum_{i \in B(j)} x_{ij}^k = 1 \quad \forall j \in N \quad (1.2)$$

$$\sum_{i \in B(j)} x_{ij}^k = \sum_{l \in F(j)} x_{jl}^k \quad \forall j \in N, k \in K \quad (1.3)$$

$$x_{ij}^k \in \{0,1\} \quad \forall k \in K, i \in N \cup D_0 \cup D_n, j \in F(i) \quad (1.4)$$

Účelová funkcia (1.1) vyjadruje počet potrebných vozidiel. Podmienky (1.2) zabezpečujú, že každý spoj bude obslúžený práve jedným vozidlom a podmienky (1.3) nám zabezpečujú, že po obslúžení spoja bude vozidlo pokračovať ďalej k ďalšiemu spoju. Podmienky (1.4) nám vyjadrujú binárnosť premennej x_{ij}^k .

Pri aplikácii elektrických autobusov je nutné zvážiť ich špecifiká pri úprave úlohy návrhu turnusov. Prvou skupinou problémov sú problémy súvisiace s batériou. Základným obmedzením, ktoré prináša batéria je obmedzenie dojazdu vozidla. Približný dojazd súčasných elektrických autobusov sa pohybuje od 100 do 300 km [1], čo nie je dost' na pokrytie celodennej prevádzky vozidla. Ďalším aspektom batérie je jej životnosť. Počas používania batérie dochádza k takzvanej degradácii, čo je proces, pri ktorom sa znižuje maximálna kapacita batérie. Tento proces trvá väčšinou niekoľko rokov, pričom sa podľa [2] zníži kapacita lítium-iónovej batérie na 70-80% jej pôvodnej kapacity počas 5-7 rokov používania.

Druhá skupina problémov sa týka vonkajších vplyvov. Táto skupina problémov opäť ovplyvňuje batériu a spotrebu vozidla. Jedným z príkladov je tvar trasy elektrického autobusu. Pri stúpaní bude spotreba určite vyššia ako pri jazde po rovnej trase. Naopak pri jazde nadol je nutné brzdiť, čo dáva možnosť na využitie regeneratívnych bŕzd na dobíjanie elektrického autobusu [3]. Ďalším veľmi dobrým príkladom vonkajšieho vplyvu je počasie a špecificky vonkajšia teplota. Bolo zistené, že pri nízkych teplotách batéria stráca kapacitu [17], čo samozrejme môže do značnej miery môže ovplyvniť turnusy.

Posledná skupina problémov sa viaže k nabíjaniu. Asi najväčším problémom vyplývajúcim z nabíjania je časová náročnosť tohto procesu. Dotankovanie klasického autobusu trvá niekoľko minút, naproti tomu dobitie batérie elektrického autobusu môže zabráť až niekoľko hodín v závislosti od použitej technológie nabíjania [3][4]. Nemaľým problémom je aj počet nabíjajúcich staníc, ktoré sú k dispozícii, keďže vybudovanie každej stanice prináša

nemalé náklady. Preto by sa malo vybudovať iba toľko nabíjajúcich staníc, koľko je nutné. Pokiaľ by sa však vybudovalo príliš málo staníc, mohlo by sa stať, že sa vytvorí pred nimi front elektrických autobusov čakajúcich na nabitie a v dôsledku toho by mohlo dôjsť ku kolapsu verejnej dopravy. Respektíve by bolo nutné zapojiť väčší počet elektrických autobusov na pokrytie spojov.

V súčasnosti existujú dva hlavné smery nabíjajúcich technológií. Sú to systémy s výmenou batérií a systémy priebežného nabíjania. Výmena batérií je systém, kde máme k dispozícii elektrický autobus s vymeniteľnou batériou a zároveň je k dispozícii infraštruktúra na ich výmenu za nabitú batériu. Tento systém bol testovaný napríklad v [6][7][8]. My sme sa však zamerali na systém s priebežným nabíjaním. Priebežné nabíjanie je systém nabíjania, kde sa elektrický autobus nabíja počas prevádzky. Toto sa môže diať v depe, na koncových staniciach alebo počas spojov na vybraných zastávkach. Zároveň je možné mať stanice umiestnené na viacerých miestach, alebo len na jednom. Pri tomto type je teda možné riešiť úlohy, kde máme presne zadané lokácie nabíjajúcich staníc alebo riešime aj ich umiestňovanie. Na ich riešenie sa dá využiť simulačný prístup ako v prácach [9][10] alebo prístup modelovania a pomocou optimalizačných metód [12][13][14]. Väčšinou sa však autori zameriavajú na aplikácie, kde sú všetky nabíjačky umiestnené v depe [12], respektíve blízko depa [13][14].

2 Definícia úlohy

Dôležitou súčasťou riešenia je definícia problému a definícia predpokladov s ktorými pracujeme, ktoré budú použité aj pri formulácii matematického modelu úlohy. Naše predpoklady sa líšia od predpokladov iných autorov, hlavne v oblasti definície nabíjajúcej infraštruktúry. Podmienky úlohy návrhu turnusov pre elektrické autobusy môžeme rozdeliť do troch skupín.

Základné podmienky úlohy návrhu turnusov sú nasledovné.

- Obsluhuje každého spoja jediným vozidlom.
- Zabezpečenie, že každé vozidlo obsluhuje iba možnú sekvenciu spojov - v jednom čase obsluhuje iba jeden spoj (vylúčenie paralelnej obsluhy spojov).

Ďalšie podmienky špecifické pre elektrické autobusy sú nasledovné.

- Kapacita batérie musí byť dostatočná na prejdienie k nabíjačke počas celého turnusu.
- Na vybraných miestach sa môže batéria nabíť. Pri nabíjaní sa však nesmie presiahnuť maximálna kapacita batérie.
- Na každej prípojke sa v jednom čase môže nabíjať iba jedno vozidlo.

Poslednou časťou sú podmienky týkajúce sa nabíjajúcej infraštruktúry, ktoré vychádzajú z nášho návrhu úlohy:

- Nabíjačky sú rozmiestnené v dopravnej sieti na rôznych miestach.
- Každá nabíjačka má iba jedinú prípojku.

- Proces nabíjania je lineárny.
- Počet použitých elektrických autobusov musí byť minimálny.

3 Výsledky práce

V tejto časti práce si predstavíme nami navrhnutý matematický model úlohy na riešenie úlohy návrhu turnusov elektrických autobusov, jednu metódu založenú na metóde generovania stĺpcov a jednu heuristickú metódu. V každej časti sú uvedené aj výsledky experimentov pre konkrétnu metódu riešenia, ako aj ich zhodnotenie. Zároveň sú tu uvedené špecifikácie stroja a dát použitých pri experimentoch.

3.1 Testovacie dáta

Na otestovanie navrhnutých metód a ich porovnanie sú použité datasety vygenerované z dát poskytnutých prevádzkovateľom verejnej dopravy DPMŽ v meste Žilina. Poskytnuté dáta obsahujú informácie o spojoch obsluhovaných naftovými autobusmi a trolejbusmi počas jedného pracovného dňa. Z poskytnutých dát sme vytvorili 10 úloh vyseparovaním spojov niektorých liniek, ktoré obsahujú rôzne počty obsluhovaných spojov. Špecifikácia datasetov je nasledovná:

- DS1 – 49 spojov na linke 26
- DS2 – 77 spojov na linke 27
- DS3 – 83 spojov na linkách 26 a 29
- DS4 – 105 spojov na linke 20, 29, 30 a 31
- DS5 – 135 spojov na linke 20, 26, 29 a 30
- DS6 – 160 spojov na linke 26, 27 a 29
- DS7 – 245 spojov na linke 21, 22 a 27
- DS8 – 415 spojov na všetkých autobusových linkách
- DS9 – 494 spojov na všetkých trolejbusových linkách
- DS10 – 927 spojov na všetkých autobusových a trolejbusových linkách

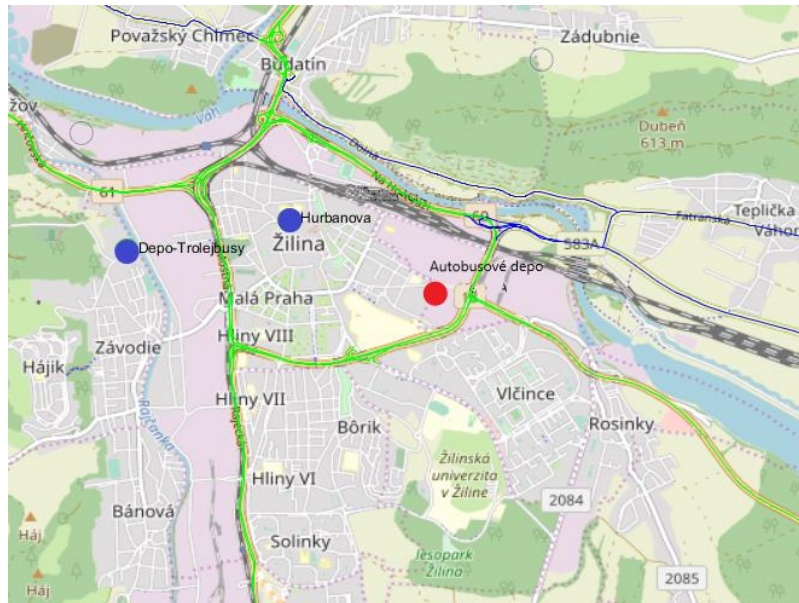
Na simuláciu rôznych podmienok počas prevádzky, ktoré sa týkajú maximálnej kapacity batérie a spotreby energie, sme špecifikovali tri rôzne scenáre. Ich špecifikáciu môžeme vidieť v tabuľke 3.1.

Tabuľka 3.1. Sezónne scenáre

<i>Scenár</i>	<i>Spotreba energie</i>	<i>Maximálna kapacita batérie</i>
<i>Jar</i>	0,8 kWh/km	140 kWh
<i>Leto</i>	1,08 kWh/km (+ 35%)	140 kWh
<i>Zima</i>	1,08 kWh/km (+ 35%)	105 kWh (-25%)

Dôležité je umiestnenie depa, odkiaľ budú elektrické autobusy ráno odchádzať a večer doňho prichádzať. Pri každom datasete sa ako ranné aj večerné depo berie autobusové depo v meste Žilina, ktoré je vyznačené ako červený bod na mape na obrázku 3.1. Na tomto obrázku sú vyznačené aj umiestnenia nabíjačiek, pričom dve sa nachádzajú v depe pre trolejbusy a jedna

je v centre mesta na Hurbanovej ulici, čo približne zodpovedá aktuálnemu umiestneniu nabíjačiek v meste.



Obr. 3.1. Umiestnenie nabíjačiek (modré body) a depo (červený bod) v meste Žilina

3.2 Exaktné metódy na riešenie úlohy návrhu turnusov el. autobusov

Medzi základné metódy riešenia optimalizačných úloh patria metódy založené na modelovaní problému a jeho následnom riešení pomocou exaktných optimalizačných metód, akou je napríklad metóda vetiev a hraníc.

Základnou časťou exaktných metód je teda popísanie problému pomocou matematického modelu. V súčasnosti neexistuje jednoznačná štandardná formulácia úlohy návrhu turnusov elektrických autobusov vo verejnej doprave. Všetky matematické modely, ktoré túto úlohu popisujú stavajú na rôznych predpokladoch v závislosti od špecifického charakteru skúmaného variantu úlohy. Ako sme už spomínali v predchádzajúcich kapitolách, existujú rôzne technológie nabíjania, rôzne prístupy k nabíjaniu a rôzne predpoklady týkajúce sa nabíjacej infraštruktúry. Zároveň závisí od autorov, akú mieru abstrakcie úlohy použijú.

Preto sme sa rozhodli vytvoriť vlastný matematický model úlohy návrhu turnusov elektrických autobusov vo verejnej doprave, ktoré budú zodpovedať našej predstave úlohy, štandardným podmienkam úlohy návrhu turnusov, ako aj predpokladom týkajúcich sa nabíjacej infraštruktúry a technológie nabíjania.

V nasledujúcej časti si formálne zadefinujeme úlohu návrhu turnusov elektrických autobusov vo verejnej doprave popíšeme nami navrhnutý matematický model.

3.2.1 Matematický model úlohy návrhu turnusov elektrických autobusov

Majme množinu spojov N , ktoré treba obslužiť. Ranné depo je reprezentované vrcholom D_0 . Večerné depo je reprezentované množinou vrcholov označených ako D_n , ktorá obsahuje jeden vrchol pre každý obsluhovaný spoj.

Ďalej definujeme množinu R reprezentujúcu všetky dostupné nabíjačky, pričom každá z nich má iba jednu nabíjaciu prípojku. Na každej nabíjačke $r \in R$ máme k dispozícii množinu všetkých nabíjacích udalostí T^r , ktoré reprezentujú časový interval, počas ktorého je možné nabíjať elektrický autobus. V našom prípade je každá nabíjacia udalosť odvodená od konca korešpondujúceho spoja, a preto majú nabíjacie udalosti rôznu dĺžku. Množina T^r je usporiadaná podľa času začiatku nabíjacích udalostí.

Každého spoja spoj $i \in N$ je charakterizovaný časom začiatku spoja s_i a časom trvania spoja t_i a spotrebou c_i . Každá nabíjačka $r \in R$ je charakterizovaná rýchlosťou nabíjania q_r , ktorá definuje, koľko energie sa nabije počas jednej jednotky času (v našom prípade uvažujeme minúty). Ako sme predtým spomenuli, na každej nabíjačke $r \in R$ máme k dispozícii množinu nabíjacích udalostí T^r . Nabíjacia udalosť $t \in T^r$ je charakterizovaná časom začiatku s_{rt} . Koniec nabíjacej udalosti je naviazaný na začiatok nasledujúcej, čiže nabíjacia udalosť končí vtedy, keď začne nasledujúca nabíjacia udalosť.

Ďalšou podstatnou časťou problému sú dĺžka trvania prejazdu a spotreba energie počas prejazdu. Čas prejazdu medzi spojom i a spojom j je reprezentovaný konštantou t_{ij} a spotreba konštantou c_{ij} . Čas prejazdu medzi spojom i a umiestnením nabíjačky r je označený ako t_{ir} a spotreba na tomto prejazde ako c_{ir} . Umiestnenie nabíjačky je definované aj časom prejazdu t_{rj} od nabíjačky r k spojom j a spotrebou c_{rj} . Použitý elektrický autobus je charakterizovaný maximálnou SoC_{max} a minimálnou SoC_{min} kapacitou batérie.

Ďalej si zadefinujeme množiny reprezentujúce vzťahy medzi spojmi a nabíjacími udalosťami. Množina F_i obsahuje spoje, ktoré môžu z časovej perspektívy nasledovať za spojom i . Podobne je zadefinovaná aj množina B_i , ktorá je množinou možných prechádzajúcich spojov spoja i .

Druhým vzťahom je vzťah medzi spojom a nabíjacími udalosťami. Tu si definujeme množinu $F_{c_{ri}}$, ktorá reprezentuje možné nasledujúce nabíjacie udalosti na nabíjačke r k spojom i . Podobne si zadefinujeme aj množinu $B_{c_{ri}}$, ktorá je množinou všetkých možných predchádzajúcich nabíjacích udalostí na nabíjačke r k spojom i .

Pre každú nabíjaciu udalosť t na nabíjačke r definujeme dve množiny $F_{i_{rt}}$ a $B_{i_{rt}}$. Tieto množiny reprezentujú možné nasledujúce a predchádzajúce spoje k nabíjacej udalosti t na nabíjačke r .

Poslednou časťou je definícia rozhodovacích premenných. Premenná x_{ij} rozhoduje, či spoj j bude obsluhovaný hneď po obslúžení spoja i . Premenná y_{irt} reprezentuje rozhodnutie o presune na nabíjaciu udalosť t na nabíjačke r hneď po obslúžení spoja i . Premenná z_{rtj} má opačný charakter a rozhoduje o presune na obsluhu spoja j po nabíjaní počas nabíjacej udalosti t na nabíjačke r . Premenná w_{rt} umožňuje prechod medzi dvomi za sebou nasledujúcimi nabíjacími udalosťami na jednej nabíjačke.

Keďže sa v našej úlohe pracuje s energiou ako palivom a potrebujeme vedieť stav nabitia batérie elektrického autobusu vo vybraných momentoch, zaviedli sme prídavné rozhodovacie premenné e_i a ε_{rt} , ktoré nám hovoria o stave nabitia batérie pred spojom i , respektíve nabíjacou udalosťou t na nabíjačke r .

Účelová funkcia

$$\min \sum_{j \in F_{D_0}} x_{D_0j} + \sum_{r \in R} \sum_{t \in F_{C_r D_0}} y_{D_0rt} \quad (3.1)$$

Účelová funkcia (3.1) sa skladá z dvoch súm, ktoré minimalizujú počet použitých elektrických autobusov vychádzajúcich z ranného depa.

Podmienky návrhu turnusov

$$\text{z. p.} \quad \sum_{i \in B_j} x_{ij} + \sum_{r \in R} \sum_{t \in B_{C_rj}} z_{rtj} = 1 \quad \forall j \in N \quad (3.2)$$

$$\sum_{j \in B_{i_{rt}}} y_{jrt} + w_{rt-1} \leq 1 \quad \forall r \in R, t \in T^r \quad (3.3)$$

$$\sum_{i \in B_j} x_{ij} + \sum_{r \in R} \sum_{t \in B_{C_rj}} z_{rtj} = \sum_{l \in F_j} x_{jl} + \sum_{r \in R} \sum_{t \in F_{C_rj}} y_{jrt} \quad \forall j \in N \quad (3.4)$$

$$\sum_{i \in B_{i_{rt}}} y_{irt} + w_{rt-1} = \sum_{j \in F_{i_{rt}}} z_{rtj} + w_{rt} \quad \forall r \in R, t \in T^r \quad (3.5)$$

Podmienky (3.2) zaručujú, že každý spoj bude obslužený práve jedným vozidlom. Podmienky (3.3) stanovujú, že každá nabíjacia udalosť bude navštívená maximálne jedenkrát, čo znamená, že iba jeden elektrický autobus môže byť nabíjaný v jednom čase na jednej nabíjačke. Podmienky (3.4) sú štandardnými podmienkami zachovania toku pre spoje. Ako podmienky zachovania toku pre každú nabíjaciú udalosť slúžia podmienky (3.5).

Podmienky spotreby energie

$$e_{D_0} = SoC_{max} \quad (3.6)$$

$$e_i \geq SoC_{min} + c_i + \sum_{j \in F_i} x_{ij} c_{ij} + \sum_{r \in R} \sum_{t \in F_{C_{ri}}} y_{irt} c_{ir} \quad \forall i \in N \quad (3.7)$$

$$e_j + c_{rj} + Mq_r(1 - z_{rtj}) \geq SoC_{min} + z_{rtj} c_{rj} \quad \forall r \in R, t \in T^r, j \in F_{i_{rt}} \quad (3.8)$$

$$e_j \leq e_i - x_{ij}(c_i + c_{ij}) + SoC_{max}(1 - x_{ij}) \quad \forall j \in N, i \in B_j \quad (3.9)$$

$$e_j \geq e_i - x_{ij}(c_i + c_{ij}) - SoC_{max}(1 - x_{ij}) \quad \forall j \in N, i \in B_j \quad (3.10)$$

$$\varepsilon_{rt} \leq e_i - y_{irt}(c_i + c_{ir}) + SoC_{max}(1 - y_{irt}) \quad \forall r \in R, t \in T^r, i \in B_{i_{rt}} \quad (3.11)$$

$$\varepsilon_{rt} \geq e_i - y_{irt}(c_i + c_{ir}) - SoC_{max}(1 - y_{irt}) \quad \forall r \in R, t \in T^r, i \in B_{i_{rt}} \quad (3.12)$$

Podmienky (3.6) slúžia na inicializáciu stavu nabitia batérie elektrického autobusu na maximálnu kapacitu na začiatku pracovného dňa. Ďalšie podmienky (3.7) zabezpečujú, že elektrický autobus bude mať dost' energie na prejdienie spoja a nasledujúceho presunu k nasledujúceho spoju alebo nabíjačke. Podobne, podmienky (3.8) zaručujú, že elektrická autobus bude mať dost' energie po nabíjaní na to, aby bol schopný prejsť k nasledujúceho spoju. Podmienky (3.9)-(3.12) sa spájajú so zachovaním energie a slúžia ako väzba medzi

premennými reprezentujúcimi stav nabitia. V prípade, že za sebou nasledujú dva spoje za sebou musia platiť podmienky (3.9) a (3.10). Podmienky (3.11) a (3.12) budú použité v prípade, že zo spoja i prejde elektrický autobus k nabíjačke r kvôli nabíjaniu počas nabíjacej udalosti t .

Podmienky nabíjania

$$e_j + c_{rj} - \varepsilon_{rt} + SoC_{max}(1 - z_{rtj}) \geq 0 \quad \forall r \in R, t \in T^r, j \in Fi_{rt} \quad (3.13)$$

$$\varepsilon_{rt+1} - \varepsilon_{rt} + SoC_{max}(1 - w_{rt}) \geq 0 \quad \forall r \in R, t \in T^r \quad (3.14)$$

$$e_j + c_{rj} - Mq_r(1 - z_{rtj}) \leq SoC_{max} \quad \forall r \in R, t \in T^r, j \in Fi_{rt} \quad (3.15)$$

$$\varepsilon_{rt+1} - Mq_r(1 - w_{rt}) \leq SoC_{max} \quad \forall r \in R, t \in T^r \quad (3.16)$$

$$e_j \leq \varepsilon_{rt} + z_{rtj} \left((s_j - t_{rj} - s_{rt})q_r - c_{rj} \right) + SoC_{max}(1 - z_{rtj}) \\ \forall j \in N, r \in R, t \in Bc_{rj} \quad (3.17)$$

$$\varepsilon_{rt+1} \leq \varepsilon_{rt} + w_{rt}(s_{rt+1} - s_{rt})q_r + SoC_{max}(1 - w_{rt}) \quad \forall r \in R, t \in T^r \quad (3.18)$$

$$e_j + c_{rj} - \varepsilon_{rt} - SoC_{max}(1 - z_{rtj}) \leq (s_{rt+1} - s_{rt})q_r \quad \forall r \in R, t \in T^r, j \in Fi_{rt} \quad (3.19)$$

V tejto časti sú zadefinované podmienky, ktoré sa vzťahujú k minimálnej a maximálnej hranici nabitej energie počas nabíjacích udalostí. Podmienky (3.13) a (3.14) definujú, že množstvo nabitej energie musí byť nezáporné. Ďalšie podmienky (3.15) a (3.16) zaručujú, že po nabíjaní nebude prekročená maximálna kapacita batérie definovaná konštantou SoC_{max} .

Podmienky (3.17), (3.18) a (3.19) limitujú maximálne množstvo dobitej energie založené na dostupnom čase pre nabíjanie počas nabíjacej udalosti. V prípade že po nabíjaní nasleduje spoj musia platiť podmienky (3.17) a (3.19) obmedzujúce čas nabíjania začiatkom nasledujúceho spoja aj nasledujúcej nabíjacej udalosti. Pre prípad, kde pokračuje elektrický autobus v nabíjaní musia platiť podmienky (3.18) obmedzujúce čas nabíjania začiatkom nasledujúcej nabíjacej udalosti.

Obligatórne podmienky

$$x_{ij} \in \{0,1\} \quad \forall i \in N \cup D_0 \cup D_n, j \in F_i \quad (3.20)$$

$$y_{irt} \in \{0,1\} \quad \forall i \in N, r \in R, t \in Fc_{ri} \quad (3.21)$$

$$z_{rtj} \in \{0,1\} \quad \forall r \in R, t \in T_r, j \in Fi_{rt} \quad (3.22)$$

$$w_{rt} \in \{0,1\} \quad \forall r \in R, t \in T_r \quad (3.23)$$

$$e_j \geq 0 \quad \forall i \in N \quad (3.24)$$

$$\varepsilon_{rt} \geq 0 \quad \forall r \in R, t \in T_r \quad (3.25)$$

Rozhodovacie premenné x_{ij} , y_{irt} , z_{rtj} and w_{rt} sú binárneho charakteru, čo je definované podmienkami (3.20), (3.21), (3.22) and (3.23). Prídavné rozhodovacie premenné e_i a ε_{rt} , ktoré udržiavajú informáciu o aktuálnom stave nabitia batérie počas turnusu sú definované podmienkami (3.24) a (3.25) ako nezáporné.

3.2.2 Výsledky experimentov

Na overenie a otestovanie navrhnutého matematického modelu sme sa rozhodli použiť štandardný IP solver Xpress IVE. Okrem otestovania matematického modelu na rôznych datasetoch sme získané výsledky porovnali aj s výsledkom riešenia štandardnej úlohy návrhu turnusov (VSP) reprezentovanej matematickým modelom (1.1)-(1.4). Zároveň sme zisťovali aj časovú náročnosť výpočtu. Časový limit na výpočet problému bol stanovený na 16 hodín, keďže pri väčších úlohách sa po 16 hodinách výpočtu začali prejavovať problémy spojené s nedostatkom pamäte RAM.

Dosiahnuté výsledky pre zimný scenár sa nachádzajú v tabuľke 3.2. Stĺpec *Rieš* predstavuje najlepšie nájdené riešenie počas výpočtu modelu, čo predstavuje počet použitých elektrických autobusov na pokrytie všetkých spojov turnusmi. Stĺpec *BB* predstavuje najlepšiu spodnú hranicu riešenia a stĺpec *Čas* predstavuje čas výpočtu modelu v sekundách. Posledný stĺpec *VSP* slúži na porovnanie výsledkov nášho matematického modelu voči prípadu, keď flotilu dostupných vozidiel budú tvoriť konvenčné autobusy.

Tab. 3.2. Výsledky experimentov s matematickým modelom v prípade zimného scenára

<i>Dataset</i>	<i>Rieš.</i>	<i>BB</i>	<i>Čas (s)</i>	<i>VSP</i>
<i>DS1</i>	4	4	10,4	4
<i>DS2</i>	4	4	40	4
<i>DS3</i>	5	5	629,3	5
<i>DS4</i>	6	6	1237	6
<i>DS5</i>	10	8	57600	8
<i>DS6</i>	11	9	57600	9
<i>DS7</i>	-	13	57600	13
<i>DS8</i>	-	26	57600	26
<i>DS9</i>	-	28	57600	28
<i>DS10</i>	-	49	57600	49

V prípade zimného scenára, kde sa okrem zvýšenej spotreby aj znížila maximálna kapacita batérie. Ako vidíme na výsledkoch pre datasety DS1-DS4, tak sa riešenie úlohy VSP a návrhu turnusov pre elektrické autobusy rovná. Tento fakt nám hovorí to, že pre úlohy definované týmito datasetmi je možné nahradiť flotilu konvenčných autobusov za flotilu elektrických autobusov v pomere 1:1. Takisto si môžeme všimnúť, že pre datasety DS1-DS4 bolo riešenie pomocou matematického modelu schopné dosiahnuť optimálne riešenie. Avšak pre datasety DS5-DS10 to pre prípad zimného scenára už nebolo možné z časového hľadiska, keďže sme dosiahli časový limit experimentu. Ako teda vidíme tak matematický model začína mať problémy s riešením pri väčších datasetoch.

V tejto časti sme predstavili nový matematický model úlohy návrhu turnusov elektrických autobusov vo verejnej doprave. Experimenty s modelom overili matematický model a preukázali použiteľnosť modelu na reálnych úlohách. Takisto sme boli schopní definovať obmedzenia veľkosti úloh, ktoré je možné pomocou matematického modelu a štandardného IP solvera riešiť. Popri tom sme našli aj faktory, ktoré vplyvujú najviac ovplyvňujú riešiteľnosť úlohy a síce spotreba energie a maximálna kapacita batérie.

3.3 Metóda redukcie vstupov s využitím riešenia pomocou IP solvera

Riešenie úlohy návrhu turnusov pomocou matematického modelu sa ukazuje ako veľmi náročné na čas, hlavne v závislosti od zväčšujúceho sa rozmeru úlohy. Preto sme sa rozhodli zamerať na algoritmus, ktorý je schopný redukovať vstupy [18][19], čiže množstvo spojov. Pri aplikácii tohto algoritmu sme schopný zredukovať počet vstupných spojov do matematického modelu, čím budeme schopný skrátiť čas výpočtu a v prípade väčších úloh dokonca nájsť veľmi dobré riešenie, ktoré by sme pri pôvodnej úlohe neboli schopní nájsť. Predstavovaný algoritmus je založený na myšlienke spájania dvoch spojov do jedného, tým, že ich zreťazíme a budeme sa k nim správať ako k jedinému spoju. Samozrejme, tento prístup môže zapríčiniť stratu optimality riešenia úlohy, ale bude nám schopný poskytnúť riešenie blízke optimálnemu, je to teda heuristický algoritmus. Náš návrh algoritmu spájania spojov sa skladá z dvoch fáz. V prvej fáze si vytvoríme možné dvojice spojov na spájanie a v druhej fáze sa zameriame na výber najvhodnejších dvojíc z výberu v prvej fáze, ktoré majú byť spojené.

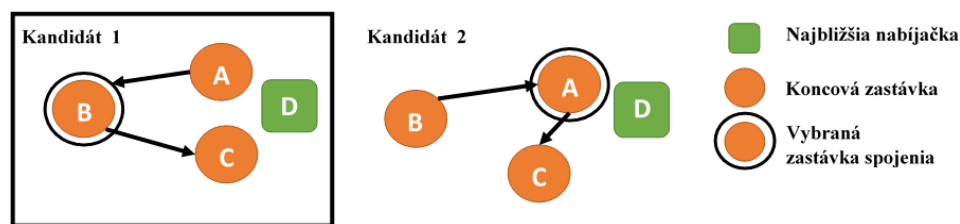
3.3.1 Výber potenciálnych dvojíc spojov na spojenie

V prvej fáze vyberáme potenciálne dvojice, ktoré by mohli byť spojené na základe niekoľkých kritérií. Prvým kritériom je výber na základe koncových zastávok. Tieto koncové zastávky, v ktorých dôjde k spojeniu, musia byť umiestnené v okrajových častiach mesta.

Ďalej si musíme definovať kritérium na výber vhodných kandidátov (dvojíc spojov) na spojenie. Prvým pravidlom je, že prvý z dvojice spojov musí končiť na zastávke nachádzajúcej sa v okrajovej časti. Zároveň druhý spoj z dvojice musí začínať na tej istej zastávke ako prvý skončil. Druhé pravidlo je založené na myšlienke, že dva spoje nasledujú v krátkom časovom slede za sebou. Veľkosť tohto časového intervalu je definovaná parametrom *MaxTime*. Po aplikácii oboch kritérií výberu dostávame zoznam potenciálnych dvojíc spojov na spojenie.

3.3.2 Výber najvhodnejšej dvojice spojov na spojenie

V druhej fáze algoritmu na spájanie spojov sa zameriame na výber najvhodnejšej dvojice spojov na spojenie spomedzi dvojíc získaných v prvej fáze.



Obr. 3.2: Príklad výberu kandidáta podľa kritéria BCC2

Prvým navrhnutým kritériom je vybrať tú dvojicu za sebou nasledujúcich spojov, ktorá má medzi sebou najkratší čas. Toto kritérium sme označili ako BCC1 (Best candidate criterion). Myšlienka druhého navrhnutého kritéria je spojená s elektrickými autobusmi a ich vzdialenosťou od nabíjajúcich staníc. Podľa druhého kritéria (BCC2) si najskôr vyberáme tie dvojice spojov, ktorých bod spojenia (zastávka) je čo najďalej od najbližšej nabíjačky.

Príkladom rozhodovania môže byť situácia na obrázku 3.2, kde máme k dispozícii dvoch kandidátov na spojenie a máme z nich vybrať toho vhodnejšieho. Na základe kritéria

BCC2 si vyberieme kandidáta 1. Po aplikovaní druhej fázy algoritmu redukcie dostávame redukovaný zoznam spojov, v ktorom sa nachádzajú všetky spoje (spojené aj nespojené).

3.3.3 Výsledky experimentov

Pri experimentoch sme testovali vplyv redukcie na výpočtový čas a výsledky riešenia úlohy pomocou matematického modelu v porovnaní s riešením pôvodnej (neredukovanej) úlohy. Testovali sme niekoľko rôznych redukcí. Pre parameter *MaxTime* sme testovali hodnoty 5 a 15 minút maximálneho času medzi spojmi. Ďalej sme testovali obe kritériá výberu kandidátov BCC1 a BCC2 pre obe hodnoty parametra *MaxTime*.

V tabuľke 3.3 nájdeme výsledky redukcie spojov pre jednotlivé parametre reprezentované stĺpcami tabuľky, ako aj stĺpec *Original*, ktorý obsahuje počet spojov pôvodnej úlohy. Každá redukcia je označená názvom reprezentujúcim veľkosť parametra *MaxTime*, za ktorým nasleduje typ aplikovaného kritéria pre výber najvhodnejšieho kandidáta.

Tab. 3.3. Výsledky redukcie vstupov pre jednotlivé datasety

<i>Dataset</i>	<i>Redukcia vstupov</i>				<i>Original</i>
	<i>5_BCC1</i>	<i>5_BCC2</i>	<i>15_BCC1</i>	<i>15_BCC2</i>	
<i>DS1</i>	42	42	31	31	49
<i>DS2</i>	60	60	44	43	77
<i>DS3</i>	63	63	52	51	83
<i>DS4</i>	123	123	96	94	160
<i>DS5</i>	81	81	73	71	105
<i>DS6</i>	102	102	85	83	133
<i>DS7</i>	196	196	155	149	245
<i>DS8</i>	329	329	273	266	415
<i>DS9</i>	494	494	494	494	494
<i>DS10</i>	839	839	780	773	927

Po redukcii sme pristúpili k testovaniu výpočtu redukovaných úloh pomocou matematického modelu a IP solvera. V tabuľke 3.4 môžeme vidieť výsledky týchto experimentov pre zimný scenár. Prvý stĺpec *Rieš./BB* pri každej redukcii obsahuje najlepšie nájdené riešenie a spodnú hranicu riešenia. Druhý stĺpec reprezentuje čas výpočtu v sekundách. Posledný stĺpec *Original* reprezentuje výsledky experimentov pre pôvodnú úlohu pred redukciou.

Pri zimnom scenári boli najväčšie problémy pri pôvodnej úlohe a tento problém sa preniesol aj na redukované úlohy. Dokazuje to veľkosť úloh, ktoré sme schopný dopočítať po redukcii. Musíme podotknúť, že kritérium BCC2 je vhodnejšie, keďže v prípade redukcí *5_BCC1* a *15_BCC1* sa zhoršili spodné hranice riešení pri niektorých úlohách. Najlepšie výsledky pre zimný scenár opäť poskytuje redukcia *15_BCC2*, pričom dosahuje aj najlepšie časy výpočtu. Priemerný ušetrený čas za však znížil na 45%, avšak aj to je značná úspora času oproti pôvodnej úlohe. Môžeme si všimnúť, že v prípade datasetov *DS9* a *DS10* sa v prípade všetkých scenárov nepodarilo nájsť riešenie po žiadnej redukcii. Je to spôsobené hlavne tým, že tieto datasety obsahujú trolejbusové spoje, ktoré sa nepodarilo zredukovať a tak zostala veľkosť úlohy stále veľmi veľká.

Tab. 3.4. Výsledky experimentov pre jednotlivé redukcie pre zimný scenár s najlepším nájdeným riešením a spodnou hranicou (*Rieš./BB*) a časom výpočtu (*Čas*) v sekundách

<i>Data set</i>	<i>5_BCCI</i>		<i>5_BCC2</i>		<i>15_BCCI</i>		<i>15_BCC2</i>		<i>Original</i>	
	Rieš./BB	Čas (s)	Rieš./BB	Čas (s)	Rieš./BB	Čas (s)	Rieš./BB	Čas (s)	Rieš./BB	Čas (s)
<i>DS1</i>	4/4	6,4	4/4	6,1	4/4	3	4/4	2,5	4/4	10,4
<i>DS2</i>	4/4	19,7	4/4	18,9	5/5	5,9	4/4	4,1	4/4	40
<i>DS3</i>	5/5	69,2	5/5	69,6	5/5	44,9	5/5	32,7	5/5	629,3
<i>DS4</i>	6/6	556,2	6/6	528,1	7/6	57600	6/6	117	6/6	1237
<i>DS5</i>	9/9	57600	9/8	57600	9/8	57600	9/8	57600	10/8	57600
<i>DS6</i>	10/9	57600	10/9	57600	10/10	1329	9/9	716,2	11/9	57600
<i>DS7</i>	-/13	57600	-/13	57600	19/14	57600	17/13	57600	-/13	57600
<i>DS8</i>	-/26	57600	-/26	57600	-/26	57600	-/26	57600	-/26	57600
<i>DS9</i>	-/28	57600	-/28	57600	-/28	57600	-/28	57600	-/28	57600
<i>DS10</i>	-/49	57600	-/49	57600	-/49	57600	-/49	57600	-/49	57600

3.4 Metóda založená na metóde generovania stĺpcov

Matematický model, ktorý sme predstavili v prechádzajúcej časti je komplexný a jeho použitie spolu so štandardným IP solverom a algoritmom vetiev a hraníc je v prípade veľkých datasetov nemožné. Je to hlavne z dôvodu časovej náročnosti ako aj pamäťovej náročnosti. Pomocou modelu sme teda schopný riešiť úlohy len približne o 100-150 spojoch, čo nie je vôbec dostatočné pre praktické použitie. Preto sme sa rozhodli vyskúšať iný typ prístupu, a síce prístup na základe metódy generovania stĺpcov. Pri tejto metóde sa na problém návrhu turnusov elektrických autobusov pozeráme z iného uhla pohľadu. V prípade metódy generovania stĺpcov nebudeme matematický model priamo vytvárať turnusy, ale budeme vyberať zo všetkých možných turnusov tie, ktoré sa použijú na pokrytie spojov.

Definujeme množinu S všetkých prípustných turnusov spĺňajúcich všetky podmienky spotreby energie a nabíjania. Potom náš hlavný problém môže byť popísaný nasledujúcim matematickým modelom, kde binárna rozhodovacia premenná λ_s predstavuje rozhodnutie, či turnus $s \in S$ bude použitý v riešení alebo nie.

$$\min \sum_{s \in S} \lambda_s \quad (3.26)$$

$$\text{z. p.} \quad \sum_{s \in S} \lambda_s \left(\sum_{i \in N} a_{sij} + \sum_{r \in R} \sum_{t \in T^r} c_{srtj} \right) = 1 \quad \forall j \in N \quad (3.27)$$

$$\sum_{s \in S} \lambda_s \left(\sum_{i \in N} b_{sirt} + d_{sr(t-1)} \right) \leq 1 \quad \forall r \in R, t \in T^r \quad (3.28)$$

$$\lambda_s \in \{0,1\} \quad \forall s \in S \quad (3.29)$$

Účelová funkcia (3.26) minimalizuje počet použitých turnusov, čo zodpovedá minimalizácii počtu použitých elektrických autobusov, nakoľko jeden turnus je obsluhovaný jedným elektrickým autobusom. Podmienka (3.27) slúži ako záruka, že každý spoj bude obslužený. Konštanty a_{sij} sú 1 v prípade, že spoj j je obslužený hneď po spoji i počas turnusu

s a konštanty c_{srtj} sú 1, ak je spoj j obslúžený hneď po nabíjaní počas nabíjacej udalosti t na nabíjačke r počas turnusu s . Podmienka (3.28) reprezentuje obmedzenie, že počas jednej nabíjacej udalosti sa môže nabíjať iba jeden elektrický autobus. Konštanty b_{sirt} v tejto podmienke predstavujú prechod medzi spojom i a nabíjacou udalosťou t na nabíjačke r počas turnusu s . Konštanty $d_{sr(t-1)}$ sú pokračovaním nabíjania medzi nabíjacou udalosťou t a predchádzajúcou nabíjacou udalosťou $t-1$ na rovnakej nabíjačke r počas turnusu s . Podmienky (3.29) sú podmienkami binárnosti pre každú premennú λ_s .

Tento pohľad na problém je špecifický pre použitie metódy generovania stĺpcov. Keďže metóda generovania stĺpcov je iteratívna metóda, ktorá pozostáva z opakovaného riešenia dvoch problémov musíme zadefinovať dve úlohy a síce: obmedzený hlavný problém a podproblém. Následne bude celkové riešenie úlohy prebiehať tak, že sa vyrieši obmedzený hlavný problém, čím získame k dispozícii konštanty potrebné pre riešenie podproblému, ktorý zodpovedá úlohe hľadania zlepšujúceho prípustného turnusu

3.4.1 Riešenie hlavného problému

Hlavný problém definovaný matematickým modelom (3.26)-(3.29). Avšak, počet všetkých prípustných turnusov je obrovský a aj pre malé úlohy je časovo náročné nájsť všetky prípustné turnusy. Preto sa hlavný problém transformuje na obmedzený hlavný problém popísaný matematickým modelom (3.30)-(3.33), ktorý je identický s hlavným problémom s tým rozdielom, že úlohu rieši len na obmedzenej množine turnusov P , ktorá je podmnožinou všetkých prípustných turnusov S .

$$\min \sum_{s \in P} \lambda_s \quad (3.30)$$

$$\text{z. p.} \quad \sum_{s \in P} \lambda_s \left(\sum_{i \in N} a_{sij} + \sum_{r \in R} \sum_{t \in T^r} c_{srtj} \right) \geq 1 \quad \forall j \in N \mid \pi_j \geq 0 \quad (3.31)$$

$$\sum_{s \in P} \lambda_s \left(\sum_{i \in N} b_{sirt} + d_{sr(t-1)} \right) \leq 1 \quad \forall r \in R, t \in T^r \mid \sigma_{rt} \leq 0 \quad (3.32)$$

$$\lambda_s \in \{0,1\} \quad \forall s \in P \subset S \quad (3.33)$$

Pri riešení obmedzeného hlavného problému musíme začať prípustným riešením úlohy návrhu turnusov elektrických autobusov. Pre počiatkové riešenie sme zvolili kyvadlové jazdy, kde je každý spoj obslúžený iným elektrickým autobusom a každý turnus tvorí iba obsluha jediného spoja. Riešením LP-relaxácie obmedzeného hlavného problému zároveň získame duálne multiplikátory π_j pre každý spoj a σ_{rt} pre každú nabíjaciu udalosť, pričom toto riešenie je realizované IP solverom.

3.4.2 Riešenie podproblému

V podprobléme potrebujeme nájsť prípustnú sekvenciu spojov a nabíjacích udalostí, ktoré by neporušili podmienky spotreby energie a nabíjania batérie a zároveň bude čo najvýhodnejší vzhľadom na získané duálne multiplikátory, čiže taký, ktorý má zápornú redukovanú cenu. Redukovaná cena turnusu (stĺpca) je našom prípade nasledovná:

$$c_s^* = 1 - \sum_{j \in N} \pi_j \left(\sum_{i \in N} a_{sij} + \sum_{r \in R} \sum_{t \in T^r} c_{srtj} \right) - \sum_{r \in R} \sum_{t \in T^r} \sigma_{rt} \left(\sum_{i \in N} b_{sirt} + d_{sr(t-1)} \right) \quad (3.34)$$

Čiže v pod-probléme sa snažíme minimalizovať redukovanú cenu turnusu. Táto úloha môže byť vyjadrená modelom s účelovou funkciou (3.35) a prídavnou podmienkou (3.36), ktoré zabezpečuje, že z ranného depa vyjde iba jediné vozidlo. Zároveň musíme zahrnúť aj podmienky toku vozidla, podmienky spotreby energie ako aj podmienky nabíjania (3.4)-(3.25). Tieto podmienky môžeme použiť z predchádzajúceho matematického modelu, pričom aj definícia rozhodovacích premenných je identická.

$$\begin{aligned} \min \quad & 1 - \sum_{i \in N} \sum_{j \in F_j} \pi_j x_{ij} - \sum_{i \in N} \sum_{r \in R} \sum_{t \in F_{c_{ri}}} \pi_j y_{irt} \\ & - \sum_{j \in N} \sum_{r \in R} \sum_{t \in B_{c_{rj}}} \sigma_{rt} z_{rtj} - \sum_{r \in R} \sum_{t \in T^r} \sigma_{rt} w_{r(t-1)} \end{aligned} \quad (3.35)$$

$$\text{z. p.} \quad \sum_{j \in F_{D_0}} x_{D_0j} + \sum_{r \in R} \sum_{t \in F_{c_{rD_0}}} y_{D_0rt} = 1 \quad (3.36)$$

Prvou metódou na riešenie pod-problému (metóda SP-mod) je základná implementáciou metódy generovania stĺpcov pomocou matematických modelov a riešenia pod-problému pomocou štandardného IP solvera. Táto metóda bola použitá za účelom porovnania ostatných navrhnutých metód. Pre riešenie pod-problému sme navrhli štyri metódy, ktoré nadväzujú na seba a každá ďalšia metóda je vylepšením tej predchádzajúcej.

Prvou navrhnutou metódou na riešenie pod-problému je metóda SP-Yen, ktorá využíva na riešenie pod-problému algoritmus na hľadanie k-najkratších ciest, keďže úlohu pod-problému je možné transformovať na hľadanie najkratšej prípustnej cesty, pričom prípustnosť je ovplyvnená podmienkami spotreby energie a nabíjania.

Základnou myšlienkou na hľadanie najkratšej prípustnej cesty je generovanie najkratších ciest z ranného do večerného depa a následné otestovanie, či sú prípustné. Na generovanie najkratších ciest sme sa rozhodli použiť algoritmus na hľadanie k-najkratších ciest. Pre implementáciu algoritmu sme použili Yenov algoritmus, pretože hľadá najkratšie cesty iteratívnym spôsobom. Maximálny počet ciest, ktoré sú testované, je obmedzený parametrom $kMax$. Ak je k-ta najkratšia cesta prípustná, tak je pridaná k množine v obmedzenom hlavnom probléme. V prípade, že po otestovaní všetkých $kMax$ -najkratších ciest nenájdeme ani jednu prípustnú cestu, tak sa spustí matematický model na nájdenie najkratšej prípustnej cesty.

Na otestovanie či je najkratšia cesta prípustná je použitá heuristika, ktorá postupne prechádza všetky vrcholy cesty a upravuje aktuálny stav nabitia batérie. Ak je vrcholom nabíjacia udalosť, je nabitie maximálne možné množstvo energie.

Ďalšia metóda SP-YenM využíva algoritmus generovania viacerých turnusov z jedného pod-problému inšpirovaná metódami v [12] a [15]. Metóda SP-YenM funguje podobne ako predchádzajúca metóda SP-Yen s tým rozdielom, že vyhladáva aj ďalšie cesty po nájdení

najkratšej prípustnej cesty. Tieto turnusy sú komplementárnymi k prvému nájdenému turnusu, čím získame niekoľko turnusov počas jediného podproblému. Konvergencia riešenia pomocou tejto metódy sa ukazuje byť omnoho rýchlejšia ako v prípade vygenerovania len jediného turnusu. Je to hlavne z dôvodu, že komplementárne turnusy tvoria dohromady časť dobrého riešenia a preto vedú účelovú funkciu obmedzeného problému rýchlo stlačiť nadol.

Tretia metóda SP-TYenM využíva redukovaný graf spojení medzi spojmi, v ktorom sa nenachádzajú nabíjacie udalosti, spolu s heuristikou na pridávanie nabíjajúcich udalostí do turnusu. Zároveň využíva aj hľadanie niekoľkých turnusov počas jediného podproblému.

Táto metóda funguje veľmi podobne ako prvá metóda a skladá sa z dvoch fáz. V prvej fáze sa hľadá najkratšia cesta len na digrafe tvorenom spojmi a ranným a večerným depom. Po nájdení najkratšej cesty nastupuje druhá fáza, v ktorej sa snažíme vytvoriť prípustný turnus z tejto najkratšej cesty pomocou pridávania nabíjajúcich udalostí do turnusu. Samozrejme, ak najkratšia cesta spĺňa podmienky pre spotrebu energie, tak nie je nutné pridávanie nabíjajúcich udalostí.

Druhá fáza sa vykonáva pomocou heuristiky na pridávanie nabíjajúcich udalostí, kde sa najprv nájde problematický spoj, ktorý porušuje podmienku spotreby energie. Následne sa postupuje smerom k začiatku turnusu a pridávajú sa nabíjacie udalosti, ktoré je možné pridať medzi jednotlivé spoje, až kým nie je turnus prípustný. Ak sa nepodarí vytvoriť prípustný turnus, vraciame sa k prvej fáze a testujeme ďalšiu najkratšiu cestu.

Posledné dve metódy SP-TYenMD1 a SP-TYenMD2 pracujú rovnako ako metóda SP-TYenM, ale je pridaná heuristika na odstránenie spojov z turnusu. Heuristika na odstraňovanie spojov je aplikovaná iba v prípade, že heuristika na pridávanie nabíjajúcich udalostí je neúspešná pri generovaní prípustného turnusu pre všetkých $kMax$ najkratších ciest. Základná myšlienka, za ktorou je odstraňovanie je, že po odstránení spoja vznikne v turnuse časová medzera v dĺžke odstráneného spoja, čo poskytne možnosť na nabíjanie elektrického autobusu, ktorý bude môcť následne pokračovať v turnuse. Heuristika na odstraňovanie spojov pracuje nasledovne. Najprv heuristika aplikuje heuristiku na pridávanie nabíjajúcich udalostí a vytvorí turnus obsahujúci nabíjacie udalosti, aj keď tento turnus je neprípustný. Ďalej v tomto turnuse nájde spoj, ktorý spôsobuje porušenie podmienky pre spotrebu energie. Tento spoj je následne odstránený z čistého turnusu bez nabíjajúcich udalostí a je opätovne aplikovaná heuristika na pridávanie nabíjajúcich udalostí. Maximálny počet odstránených spojov je ohraničený parametrom $dMax$. Vytvorili sa dve stratégie odstraňovania, a sice sekvenčné odstraňovanie v metóde SP-TYenMD1 a cyklické odstraňovanie v metóde SP-TYenMD2.

3.4.3 Výsledky experimentov

Pre spustenie experimentov bolo nutné nastaviť niektoré parametre definované pri metódach. Čas pre výpočet jedného experimentu bol stanovený na 24 hodín. Ďalej pre výpočet jedného podproblému bol stanovený maximálny čas tri hodiny, v prípade, že sa použije výpočet pomocou matematického modelu. Parameter $kMax$, ktorý definuje maximálny počet najkratších ciest vygenerovaných algoritmom k-najkratších ciest vo všetkých metódach kde sa používa, bol zafixovaný na hodnotu $kMax = 300$. Nakoniec, parameter $dMax$, ktorý definuje maximálny počet spojov odstraňovaných v metódach SP-TYenMD1 a SP-TYenMD2, bol

stanovený na $dMax = 5$ pre datasety DS1-DS8. Pre datasety DS9 a DS10 bol parameter stanovený na $dMax = 7$, kvôli veľkosti úlohy.

Tabuľka 3.5: Výsledky experimentov s pri zimnom scenári

<i>Dataset</i>	<i>Najl. výsledok /BB</i>	<i>SP-mod</i>		<i>SP-TYenMD1</i>		<i>SP-TYenMD2</i>	
		<i>Rieš.</i>	<i>Čas (s)</i>	<i>Rieš.</i>	<i>Čas (s)</i>	<i>Rieš.</i>	<i>Čas (s)</i>
<i>DS1</i>	4/4	5	315	5	3,3	5	4,6
<i>DS2</i>	4/4	10	49564	4	1,2	4	1,2
<i>DS3</i>	5/5	14	40531	6	28,6	6	23,7
<i>DS4</i>	6/6	23	97162	7	29,7	7	38
<i>DS5</i>	10/8	79	92507	9	116	9	120
<i>DS6</i>	11/9	123	65441	10	151,6	10	146
<i>DS7</i>	-/13	-	-	15	399,6	14	358
<i>DS8</i>	-/26	-	-	27	361	27	394
<i>DS9</i>	-/28	-	-	31	2558	31	2674
<i>DS10</i>	-/49	-	-	56	31409	123	45105

V tabuľke 3.5 nájdeme výsledky pre prípad zimného scenára pre najlepšie dve metódy SP-TYenMD1 a SP-TYenMD2 a ich porovnanie so spodnou hranicou riešenia získanou pomocou matematického modelu ako aj porovnanie s metódou riešenia pod-problému len pomocou matematického modelu (SP-mod). Ako môžeme vidieť v tabuľke 3.5, tak pri použití metódy SP-mod je možné riešiť len veľmi malý rozsah úloh a zároveň čas výpočtu je veľmi vysoký. Pri tejto metóde nastali problémy už pri datasete DS2, kde je dosiahnutý výsledok viac ako dvojnásobný so spodnou hranicou riešenia. Naproti tomu je vidno, že metódy SP-TYenMD1 a SP-TYenMD2 značne zlepšili možnosti metódy generovania stĺpcov. Vidíme to jednak na dosiahnutých výsledkoch, kde bola metóda dosiahnuť veľmi dobré výsledky s nízkym rozdielom oproti spodnej hranici. Zároveň čas výpočtu bol drasticky znížený a môžeme povedať, že metóda generovania stĺpcov je vhodná na použitie aj pri reálnych úlohách. Pri porovnaní oboch metód SP-TYenMD1 a SP-TYenMD2 vidíme, že metóda SP-TYenMD2 dosiahla v prípade datasetu DS7, avšak zlyhala v prípade datasetu DS10. Pri testovaní na ostatných scenároch sa však vyskytli aj opačné prípady a teda nemôžeme povedať, ktorá z týchto dvoch metód je lepšia.

Vo všeobecnosti môžeme povedať, že náš návrh metódy generovania stĺpcov a špeciálne metódy SP-TYenMD1 a SP-TYenMD2 sú veľmi vhodné na riešenie úlohy návrhu turnusov elektrických autobusov, keďže dosiahli veľmi dobré výsledky a čas výpočtu bol značne skrátený v porovnaní s riešením pomocou matematického modelu, zvlášť v prípade úloh väčšieho rozsahu. Môžeme teda túto metódu odporúčať aj na úlohy väčšieho rozsahu a praktické využitie v systémoch na podporu rozhodovania.

3.5 Riešenie pomocou algoritmu Grouping Genetic Algorithm (GGA)

Jednou z najpoužívanejších metaheuristických metód je genetický algoritmus. Patrí medzi algoritmy, ktoré pracujú s populáciou riešení (chromozómy). Táto populácia sa v priebehu algoritmu mení, pričom sa zo starej populácie vytvorí nová na základe niekoľkých

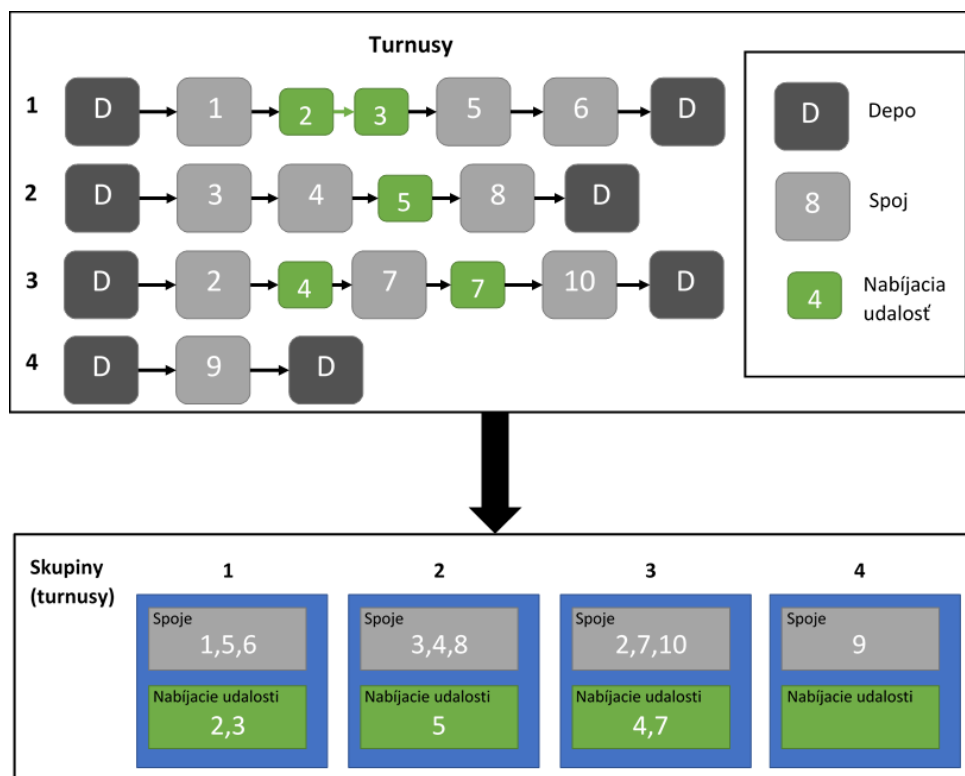
operácií, ktoré z riešení v pôvodnej populácii vytvoria nové riešenia. Veľkosť populácie je štandardne daná parametrom a počas behu algoritmu sa nemení. Pre našu úlohu sme sa rozhodli použiť metódu Grouping Genetic Algorithm (GGA), ktorá je špecifickým variantom genetického algoritmu vhodným pre úlohy deliace množinu prvkov (spoje) na disjunktné skupiny (turnusy).

Na základe [16] sa Grouping Genetic Algorithm vyznačuje špecifickou reprezentáciou riešenia. Ďalej obsahuje tri rôzne operátory, ktoré vytvárajú novú populáciu. Sú to kríženie, mutácia a inverzia. Operátory kríženia a mutácie principiálne pracujú identicky ako pri štandardnom genetickom algoritme. Operátor inverzie je novým operátorom, ktorý však nemení samotné riešenie, iba jeho štruktúru.

3.5.1 Reprezentácia riešenia a návrh operátorov

V našom prípade riešenie (chromozóm) predstavuje kompletný návrh turnusov pre elektrické autobusy. Základnú myšlienku návrhu predstavuje obrázok 3.3. Riešenie sa skladá z turnusov pre jednotlivé elektrické autobusy. Každý turnus teda predstavuje skupinu (gén), ktorú tvoria obsluhované spoje a navštívené nabíjačky. Zároveň musí platiť, že každý turnus v riešení je prípustný, čiže musí spĺňať podmienky pre nabíjanie a spotrebu energie.

V každej skupine sa nachádzajú dva zoznamy. Prvý zoznam obsahuje spoje obsluhované počas turnusu. Druhým zoznamom, je zoznam využitých nabíjajúcich udalostí počas daného turnusu reprezentovaného skupinou. Platí, že každý spoj musí byť obslužený a teda riešenie musí obsahovať všetky spoje. Pre nabíjacie udalosti musí platiť, že každá nabíjacia udalosť môže byť v rámci riešenia využitá iba jedenkrát. Fitness riešenia je reprezentovaná hodnotou účelovej funkcie.

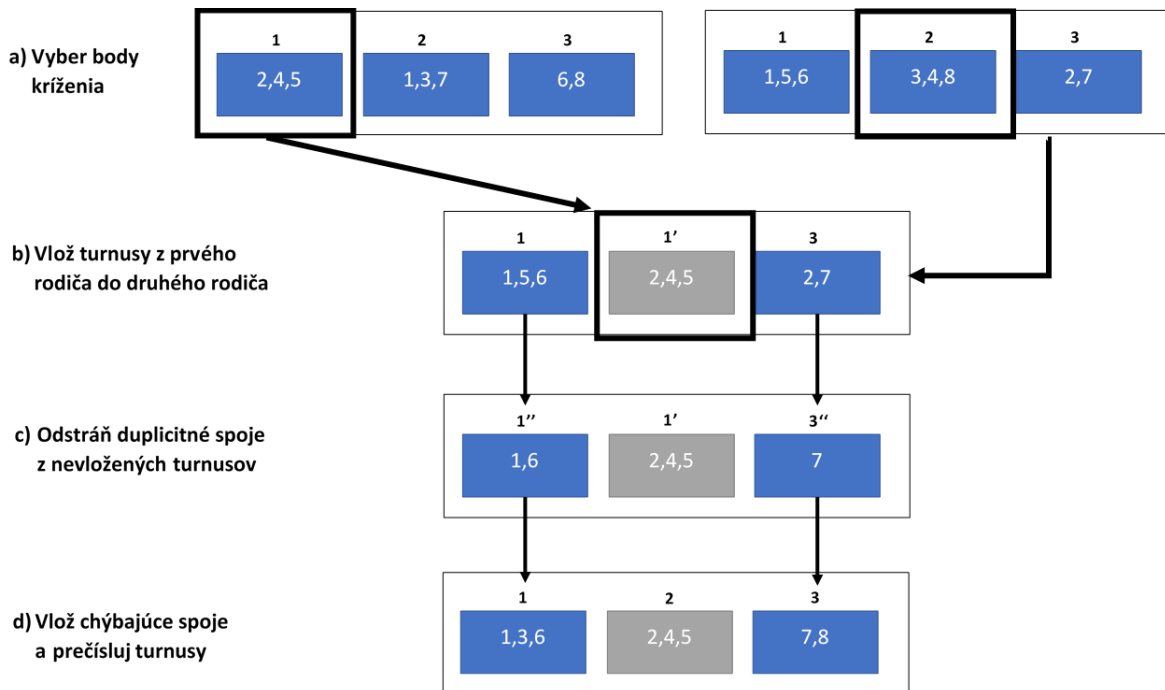


Obr. 3.3. Schéma reprezentácie riešenia pre Grouping Genetic Algorithm

Dôležitým parametrom metaheuristiky je parameter *PopSize*, ktorý predstavuje veľkosť populácie. Na jej inicializáciu je použitá vkladacia heuristika, ktorá postupne vkladá spoje do jednotlivých turnusov, pričom kontroluje prípustnosť vloženia spoja do turnusu aj potenciálnym pridaním potrebných nabíjacích udalostí. Ak nie je možné spoj vložiť do žiadneho turnusu, vytvorí sa nový turnus, do ktorého je tento spoj vložený.

$$p_i = \frac{\frac{1}{f_i}}{\sum_{j=1..popSize} \frac{1}{f_j}} \quad (3.37)$$

Kríženie je operácia, pri ktorej skombinujeme niekoľko riešení za účelom získania nového riešenia. Vo väčšine prípadov sa vyberajú práve dve riešenia nazývané aj ako rodičia. Prvou časťou kríženia je výber rodičov pre kríženie – selekcia. V našej implementácii sme sa rozhodli použiť pre selekciu metódu Roulette-Wheel-Selection. Pri použití tejto metódy transformujeme fitness f_i riešenia i na pravdepodobnosť p_i pre každé riešenie pomocou vzorca (3.37), pričom platí že suma všetkých pravdepodobností riešení je rovná jednej. Získané pravdepodobnosti prenesieme na interval 0-1, čím ho rozdelíme na intervaly reprezentujúce jednotlivé riešenia. Po vygenerovaní náhodného čísla sa vyberie to riešenie, do ktorého intervalu toto číslo padlo.



Obr. 3.4. Schéma kríženia

Na kombináciu riešení v krížení sme použili metódu uvedenú v [16]. Základný princíp heuristiky je možné popísať v nasledujúcich krokoch, ktoré sú zobrazené aj na obrázku 3.4. Najprv sa pre oboch rodičov (riešenia) náhodne vybrané dva body kríženia (obr. 3.4a). Následne je vybraná sekcia z prvého rodiča $1'$ vložená do druhého rodiča namiesto vybranej sekcie 2 druhého rodiča (obr. 3.4b), čím vzniká nový potomok. V potomkovi sa môže stať, že obsahuje duplicity spojov a niektoré spoje neobsahuje vôbec. Duplicity sú teda z potomka odstránené,

pričom sa vyhýbame vlozenej sekcii. Počas tohto procesu sa môže modifikovať niektoré gény - z génu 1 sa teda stane gén 1'' a z génu 3 sa stane gén 3'' (obr. 3.4c). Zároveň je aplikovaná heuristika na vkladanie nabíjacích udalostí na zabezpečenie prípustnosti turnusov. V ďalšom kroku sa vkladajú spoje, ktoré sú vynechané (obr. 3.4d) – do génu 1'' sme vložili spoj číslo 3 a do génu 3'' spoj číslo 8. Tento postup sa opakuje, až kým nie je vytvorená nová populácia.

Mutácia pracuje iba s jediným riešením za účelom jeho zmeny a vytvorenia nového riešenia. Mutácia je realizovaná výmennou heuristikou vymieňajúcou sekvencie spojov medzi dvomi vybranými turnusmi. Z prvého vybraného turnusu je náhodne vybraná sekvencia, ktorá sa bude vymieňať za sekvenciu spojov druhého vybraného turnusu, pričom tieto sekvencie sa odohrávajú približne v rovnakom časovom okne. Zároveň sa dbá na to, aby bola výmena prípustná z pohľadu spotreby energie. Operácia mutácia je ovplyvňovaná dvomi parametrami – počet mutácií *MutNum* a pravdepodobnosť vykonania mutácie *MutProb*.

Inverzia nemení riešenie ale iba jeho štruktúru za účelom zvýšiť diverzitu oprácie kríženia. Realizuje sa ako náhodná výmena poradia dvoch skupín – turnusov v rámci riešenia. Ovpływujú ju opätovne dva parametre – počet inverzií *InvNum* a pravdepodobnosť vykonania inverzie *InvProb*.

3.5.2 Výsledky experimentov

Pre otestovanie navrhutej metaheuristiky a ohodnotenie jej výkonnosti sme vykonali niekoľko experimentov so základným nastavením parametrov ako aj experimenty s nastavovaním parametrov. Po experimentoch s nastavením parametrov sme zvolili nasledovné hodnoty parametrov:

- *PopSize* = 150
- *MutNum* = 10
- *MutProb* = 0,5
- *InvNum* = 10
- *InvProb* = 0,8

Pre každý dataset bolo vykonaných 10 replikácií. Pre každú replikáciu bol stanovený časový limit 1200 sekúnd, v prípade datasetu DS10 to bolo 1800 sekúnd. Pre prípad druhého kritéria zastavenia bola zvolená hranica 200 výmen populácií bezo zmeny najlepšieho nájdeného riešenia.

Tabuľka 3.6 obsahuje výsledky pre zimný scenár po nastavení parametrov. Sledovaný je najlepší, najhorší dosiahnutý výsledok, ako aj počet, koľkokrát bol dosiahnutý najlepší výsledok. Zároveň je uvedený aj najlepší, najhorší a priemerný čas výpočtu replikácie. V prípade datasetov DS1-DS4 môžeme vidieť, že dosiahnuté sú identické s optimálnym riešením vo všetkých replikáciách. Vidíme však rozdiely v ostatných datasetoch. Pre výsledky datasetov DS5-DS7 sa znovu zvýšila variabilita výsledkov, čiže nedosahujeme najlepší nájdený výsledok až tak často. V prípade datasetov DS9 a DS10 sa už podstatne vzdalujeme od spodnej hranice riešenia. Toto zhoršenie sa vzťahuje k náročnejším podmienkam fungovania, keďže v zimnom scenári je zvýšená spotreba energie a zároveň nižšia maximálna kapacita batérie. Čas výpočtu sa zvyšuje v závislosti od veľkosti datasetu. Je to ukázané hlavne v prípade datasetu

DS10, kde sa metaheuristika ukončila z dôvodu vyčerpania času a nie z dôvodu dosiahnutia maximálneho počtu výmen populácií bezo zmeny najlepšieho nájdeného riešenia.

Tabuľka 3.6. Výsledky metaheuristiky GGA pre zimný scenár s upravenými nastaveniami

<i>Dataset</i>	<i>Grouping Genetic Algorithm – upravené nastavenie</i>					
	Najlepší výsledok	Najhorší výsledok	Počet najlepších v repl.	Najlepší čas (s)	Najhorší čas(s)	Priemerný čas (s)
<i>DS1</i>	4	4	20	1,27	2,45	1,37
<i>DS2</i>	4	4	20	4,75	4,97	4,82
<i>DS3</i>	5	5	20	4,51	4,98	4,70
<i>DS4</i>	6	6	20	7,64	10,67	8,37
<i>DS5</i>	8	9	2	11,09	18,34	11,70
<i>DS6</i>	9	10	14	17,32	33,82	21,99
<i>DS7</i>	13	14	5	53,47	99,05	62,72
<i>DS8</i>	26	26	20	209,44	258,99	222,68
<i>DS9</i>	30	32	1	629,84	1251,93	944,56
<i>DS10</i>	54	56	5	1801,01	1819,99	1806,66

Pri experimentoch s metaheuristikou sa ukázalo, že je veľmi výkonná, najmä na úlohách malého rozsahu, kde dosahovala vždy optimálne výsledky. Pri úlohách väčšieho rozsahu síce nie vždy dosiahla optimálny výsledok, avšak aj najhorší výsledok sa príliš nelíšil od spodnej hranice riešenia. Pri experimentoch sa ukázalo, že nastavenie parametrov metaheuristiky značne ovplyvňuje stabilitu výsledkov. Na základe časov výpočtu a výsledkov môžeme túto metaheuristiku odporučiť na riešenie úlohy návrhu turnusov elektrických autobusov vo verejnej doprave.

4 Záver

Táto práca sa venovala úlohe návrhu turnusov elektrických autobusov vo verejnej doprave, ktorá je variantom tradičnej úlohy návrhu turnusov. Keďže sa v úlohe využívajú elektrické autobusy, je nutné ju upraviť a zahrnúť špecifiká spojené s elektrickými autobusmi, ktorými sú obmedzený dojazd a nutnosť nabíjania [4][5][7], ktoré trvá dlhší časový úsek. Nabíjačky na ktorých je možné nabíjanie sú rozmiestnené v dopravnej sieti. Riešením tejto úlohy riešime jeden z problémov pri transformácii flotily konvenčných autobusov na flotilu elektrických autobusov, pričom táto transformácia sa bude zrejme vyžadovať vo viacerých mestách kvôli trendu znižovania emisií. Zároveň prevádzka elektrických autobusov prinesie zníženie operačných nákladov, keďže cena elektriny je momentálne omnoho nižšia ako cena palív.

Na modelovanie úlohy sme formulovali matematický model špecifický pre naše zadanie, ktorý spĺňal všetky potrebné podmienky týkajúce sa spotreby energie a nabíjania ale aj štandardné podmienky návrhu turnusov, keďže pri prácach iných autorov sme sa nestretli so špecifikom rozmiestnenia nabíjajúcich staníc v dopravnej sieti, ale len na jednom mieste, najčastejšie v depe [11][12][13][14]. Následne sme s týmto modelom vykonali numerické experimenty, pričom sme testovali exaktné riešenie úlohy pomocou štandardného IP solvera Xpress IVE. Na základe experimentov sme zistili exaktné riešenia testovaných úloh, respektíve

spodnú hranicu riešenia, čo nám poslúžilo ako porovnávacie kritérium pri ostatných skúmaných metódach. Ďalej bolo navrhnutých niekoľko metód na riešenie zadanej úlohy. Tieto metódy boli následne skúmané z pohľadu kvality získaných výsledkov ako aj z pohľadu výpočtového času. Prvou metódou bola metóda redukcie vstupov s využitím riešenia úlohy pomocou matematického modelu a IP solvera. Táto metóda predspracuje vstupné dáta heuristickým spôsobom, čím je schopná zredukovať množstvo vstupov odosielaných do matematického modelu. Týmto spôsobom sa podarilo riešiť úlohy väčšieho rozsahu a zároveň znížiť čas výpočtu potrebný na riešenie úloh.

Druhou skúmanou metódou bola metóda založená na metóde generovania stĺpcov, ktorá pracuje s iným pohľadom na problém a na riešenie častí problému využíva exaktné metódy. Pri tejto metóde sme navrhli niekoľko alternatív riešenia pod-problému ktorý je jednou z fáz metódy generovania stĺpcov, založených na algoritme hľadania k-najkratších ciest s využitím niekoľkých heuristických algoritmov. Pri experimentoch sa ukázalo, že metóda generovania stĺpcov je schopná dosiahnuť dobré výsledky aj na úlohách reálneho rozsahu, pričom čas výpočtu bol výrazne nižší ako v prípade použitia exaktných metód.

Poslednou navrhnutou metódou bola metaheuristika Grouping Genetic Algorithm, ktorá je špecifickou odnožou genetického algoritmu. Pri tejto metóde boli navrhnuté špecifické operácie spracúvajúce populáciu, ako aj testovanie nastavenia parametrov metaheuristiky. Ukázalo sa že nastavenie parametrov značne ovplyvňuje stabilitu algoritmu a rýchlosť výpočtu niektorých fáz algoritmu. Samotná metóda sa ukázala ako veľmi vhodná na použitie pri reálnych úlohách, nielen z pohľadu získaných výsledkov, ale aj z pohľadu výpočtového času.

Navrhnuté metódy sa ukázali ako vhodné na použitie aj na riešenie úloh reálneho rozsahu. Tento fakt, ako aj získané výsledky nám poskytujú možnosť využiť tieto metódy pri návrhu nástrojov potrebných pri zavádzaní flotily elektrických autobusov do prostredia verejnej dopravy, ako aj ich operatívneho a strategického riadenia. V budúcnosti by bolo vhodné širšie preskúmanie navrhnutých metód, ako aj testovanie na úlohách väčšieho rozsahu a zároveň ich kombinácia s metódami navrhujúce nabíjaciu infraštruktúru, čo môže uľahčiť prechod z konvenčných vozidiel na ekologickejšie elektrické vozidlá v prostredí verejnej dopravy.

Literatúra

- [1] ZeEUS project. (2016) *Zeeus ebus report: An overview of electric buses in Europe*. [Online]. Available at: <http://zeeus.eu/uploads/publications/documents/zeeus-ebus-reportinternet.pdf>
- [2] A. Hoke, A. Brissette, D. Maksimović, A. Pratt and K. Smith. *Electric vehicle charge optimization including effects of lithium-ion battery degradation*. 2011 IEEE Vehicle Power and Propulsion Conference, Chicago, IL, 2011, pp. 1-8. doi: 10.1109/VPPC.2011.6043046
- [3] Mikko Pihlatie, VTT . *Planning of electric bus systems* . Presentation. Presented on Latin American webinar: Centro Mario Molina Chile & UNEP. 2017. [Online] http://movelatam.org/wp-content/uploads/2017/09/VTT_electric_bus_system_planning.pdf

- [4] National Academies of Sciences, Engineering, and Medicine. 2018. *Battery Electric Buses—State of the Practice*. Washington, DC: The National Academies Press. <https://doi.org/10.17226/25061>.
- [5] Rogge, Matthias; Wollny, Sebastian; Sauer, Dirk U. 2015. *Fast Charging Battery Buses for the Electrification of Urban Public Transport—A Feasibility Study Focusing on Charging Infrastructure and Energy Storage Requirements*. *Energies* 8, no. 5: 4587-4606.
- [6] Kim, Jeongyong; Song, Inho; Choi, Woongchul. 2015. *An Electric Bus with a Battery Exchange System*. *Energies* 8, no. 7: 6806-6819.
- [7] Mirchandani, Pitu ; Madsen, Oli B.G. ; Adler, Jonathan. 2012. *Scheduling and location issues in transforming service fleet vehicles to electric vehicles*. 12th International Conference on Advanced Systems for Public Transport, Santiago, Chile, 16 p.
- [8] Zhu Chao, Chen Xiaohong, *Optimizing Battery Electric Bus Transit Vehicle Scheduling with Battery Exchanging: Model and Case Study*, *Procedia - Social and Behavioral Sciences*, Volume 96, 2013, Pages 2725-2736, ISSN 1877-0428, <https://doi.org/10.1016/j.sbspro.2013.08.306>.
- [9] Moataz Mohamed, Hany Farag, Nader El-Taweel, Mark Ferguson. *Simulation of electric buses on a full transit network: Operational feasibility and grid impact analysis*. *Electric Power Systems Research*, Volume 142, 2017, Pages 163-175, ISSN 0378-7796, <https://doi.org/10.1016/j.epsr.2016.09.032>.
- [10] T. Paul and H. Yamada. *Operation and charging scheduling of electric buses in a city bus route network*. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, 2014, pp. 2780-2786. doi: 10.1109/ITSC.2014.6958135
- [11] Posthoorn, C. *Vehicle Scheduling of Electric City Buses: A Column Generation Approach*. (Master thesis). 2016. Faculty of Electrical Engineering, Mathematics and Computer Science. Technische Universiteit Delft. <http://resolver.tudelft.nl/uuid:0a0b2596-e908-475c-b8d7-48e4b6bbb37d>
- [12] van Kooten Niekerk, Marcel & Akker, J.M. & Hoogeveen, J.A.. (2017). *Scheduling electric vehicles*. *Public Transport*. 9. 10.1007/s12469-017-0164-0.
- [13] Sassi, Ons & Oulamara, Ammar. (2014). *Electric Vehicle Scheduling and Optimal Charging Problem: Complexity, Exact and Heuristic Approaches*. *International Journal of Production Research*. 55. 10.1080/00207543.2016.1192695.
- [14] Matthias Rogge, Evelien van der Hurk, Allan Larsen, Dirk Uwe Sauer. *Electric bus fleet size and mix problem with optimization of charging infrastructure*. *Applied Energy*, Volume 211, 2018, Pages 282-295, ISSN 0306-2619, <https://doi.org/10.1016/j.apenergy.2017.11.051>.
- [15] Jaroslav Janacek, Michal Kohani, Matyas Koniorczyk, Peter Marton, *Optimization of periodic crew schedules with application of column generation method*, *Transportation Research Part C: Emerging Technologies*, Volume 83, 2017, Pages 165-178, ISSN 0968-090X, <https://doi.org/10.1016/j.trc.2017.07.008>.

- [16] Michael Mutingi, Charles Mbohwa. (2017) *Grouping Genetic Algorithms - Advances and Applications*. Studies in Computational Intelligence 666, Springer, ISBN 978-3-319-44393-5
- [17] H. Abidi, K. Hassine and F. Mguis. (2018) *Genetic Algorithm for Solving a Dynamic Vehicle Routing Problem with Time Windows*. 2018 International Conference on High Performance Computing & Simulation (HPCS), Orleans, France, pp. 782-788, doi: 10.1109/HPCS.2018.00126.
- [18] Zeng, Yong & Liu, Da-Cheng & Li, Ju-Xuan & Hou, Xiang-Yu. (2013). *A Study of Grouping Heuristics on Vehicle Scheduling Problem Based on Changeable Expenditure Coefficient Model*. TELKOMNIKA Indonesian Journal of Electrical Engineering. 11. 10.11591/telkomnika.v11i1.1913.
- [19] Y. Hou, Y. Yang, Z. Liu and L. Sun. (2016) *A discontinuous coordinated charging strategy for electric vehicles*. 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), Hefei, China, pp. 1099-1102, doi: 10.1109/ICIEA.2016.7603746.

Vlastné publikácie

- [1] Maroš Janovec. *Exaktný prístup k turnusovaniu električkových autobusov*. In: Využitie kvantitatívnych metód vo vedeckovýskumnej činnosti a v praxi. Zborník príspevkov z medzinárodného vedeckého seminára. 1. vyd. Bratislava : Vydavateľstvo EKONÓM, 2019. s. 74-82. ISBN 978-80-225-4617-1
- [2] Maroš Janovec, Michal Koháni (2019). *Exact approach to the electric bus fleet scheduling*. Transportation Research Procedia, Volume 40, Pages 1380-1387, ISSN 2352-1465, <https://doi.org/10.1016/j.trpro.2019.07.191>.
- [3] M. Janovec and M. Koháni (2019). *Battery Degradation impact on the Electric Bus Fleet Scheduling*. International Conference on Information and Digital Technologies (IDT), Zilina, Slovakia, pp. 190-197. doi: <https://doi.org/10.1109/DT.2019.8813693>
- [4] Janovec, M. and Kohani, M. (2020). *Comparison of Continuous and Discontinuous Charging Models for the Electric Bus Scheduling Problem*. In Proceedings of the 9th International Conference on Operations Research and Enterprise Systems - Volume 1: ICORES, ISBN 978-989-758-396-4, pages 179-186. DOI: <https://doi.org/10.5220/0008962901790186>
- [5] Janovec M., Kohani M. (2020) *Data Reduction Algorithm for the Electric Bus Scheduling Problem*. In: Neufeld J.S., Buscher U., Lasch R., Möst D., Schönberger J. (eds) Operations Research Proceedings 2019. Operations Research Proceedings (GOR (Gesellschaft für Operations Research e.V.)). Springer, Cham. https://doi.org/10.1007/978-3-030-48439-2_98