EKONOMICKÁ UNIVERZITA V BRATISLAVE

FAKULTA HOSPORÁRSKEJ INFORMATIKY

Evidenčné číslo:103003/I/2019/421500290281

Viackriteriálna optimalizácia prepravných trás

Diplomová práca

2019 **Bc. Barbora Nováková**

EKONOMICKÁ UNIVERZITA V BRATISLAVE

FAKULTA HOSPORÁRSKEJ INFORMATIKY

Viackriteriálna optimalizácia prepravných trás

Diplomová práca

Študijný program: Informačný manažment

Študijný odbor: Kvantitatívne metódy v ekonómii

Školiace pracovisko: Katedra operačného výskumu a ekonometrie

Vedúci záverečnej práce: Ing. Pavel Gežík. PhD

Bratislava 2019 **Bc. Barbora Nováková**

1406721678

Ekonomická univerzita v Bratislave
Fakulta hospodárskej informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Bc. Barbora Nováková
Študijný program:	informačný manažment (Jednoodborové štúdium, inžiniersky II. st., externá forma)
Študijný odbor:	3.3.24 Kvantitatívne metódy v ekonómii
Typ záverečnej práce:	Inžinierska záverečná práca
Jazyk záverečnej práce:	slovenský
Sekundárny jazyk:	anglický

Názov: Viackriteriálna optimalizácia prepravných tras

Anotácia: Väčšina úloh riešiacich problematiku prepravných trás sa zameriava na optimalizáciu na základe jedného kritéria, najčastejšie spojeného s minimálnou prejazdenou vzdialenosťou. Toto kritérium sa v dobe informatizácie a komplexnejšej optimalizácie prepravy stáva nepostačujúce a do rozhodovania o voľbe trasy vstupujú viaceré kritéria. Často sa jedná o kritéria spojené s časom a ekonomickou výhodnosťou danej trasy.
Zaradenie viacerých kritérií do rozhodovania o voľbe trasy komplikuje a dopĺňa štandardné prístupy a algoritmy.

Vedúci:	Ing. Pavel Gežík, PhD.
Katedra:	KOVE FHI - Katedra operačného výskumu a ekonometrie FHI
Vedúci katedry:	prof. Mgr. Juraj Pekár, PhD.
Dátum zadania:	11.11.2016

Dátum schválenia: 16.11.2016

prof. Ing. Michal Fendek, PhD.
vedúci katedry

Poďakovanie

Touto cestou by som sa chcela poďakovať vedúcemu mojej diplomovej práce Ing. Pavlovi Gežíkovi, Phd. za jeho ochotu, ústretovosť, rady a pripomienky, ktoré napomohli k vypracovaniu tejto diplomovej práce.

Čestné vyhlásenie

Čestne vyhlasujem, že záverečnú prácu som vypracovala samostatne a že som uviedla všetku použitú literatúru.

V Bratislave dňa 3.5.2019 ....

Bc. Barbora Nováková

ABSTRAKT

Nováková, Barbora, Bc.: Viackriteriálna optimalizácia prepravných trás – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky; Katedra operačného výskumu a ekonometrie.- Bratislava: FHI EU, 2019, počet strán 67

Cieľom záverečnej práce je analýza okružných úloh s dodatočnými kritériami a prezentácia vybraného algoritmu, pomocou ktorého sú riešené vybrané úlohy z reálneho sveta.

Práca je rozdelené do piatich kapitol. Obsahuje 2 prílohy a 15 obrázkov a 9 tabuliek. Prvá kapitola sa zaoberá prehľadom súčasného stavu riešenej problematiky doma na Slovensku a v zahraničí. V druhej kapitole sa charakterizuje cieľ. V tretej kapitole sú prezentované úlohy optimalizácie dopravných úloh. Viackriteriálna optimalizácia prepravných trás, možnosti jej riešenia a použité algoritmy sú prezentované vo štvrtej kapitole. Posledná kapitola je zameraná na prezentáciu získaných výsledkov, ich porovnanie a zhodnotenie.

Kľúčové slová: okružné úlohy, algoritmus Clark-Wright, dodatočné ohraničenia, kapacita, časové okná

ABSTRACT

Nováková, Barbora, Bc.: Multicriterial optimization of transportation routes – University of Economics in Bratislava, Faculty of Economic Informatics; Department Of Economics and Econometrics. – Supervisor: Ing. Pavel Gežík, PhD. – Bratislava: FHI EU, 2019, 67p.

The aim of the final thesis is the is to analyze the vehicle roouts with additional criteria and to present the selected algorithm, whitch are used to solve selected problems from the real world.

The thesis is divided into five chapter. In constains 2 attachmenst, 15 charts and 9 tables. The first chapter is devoted to presents the tasks of optimization of traffics problems. The second chapter describes the aim. The thirs chapter presents the tasks of opitimization of traffic problems. Multi-criterial optimization of transport routes, the possibilities of its solution and algorithms are presented in the fourth chapter. The last chapaer is focused on the presentation of the results, theis comparison and evaluation.

Kľúčové slová: vehicle routing problem, Clark-Wright algorithm, additional constraints, capacity, time windows

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

V dnešnej dobe čas znamenajú peniaze. To platí pre firmy ale aj bežných ľudí. Každodenne sa firmy aj ľudia stretávajú s množstvom situácií, ktoré sú spojené s rozhodovaním v oblasti prepravy osôb a tovarov. Množstvo logistických operácií, optimalizácia plánovania trás a času potrebného v distribučných procesoch ale aj riadenie vozových parkov alebo samotných pracovníkov sú zložité procesy, ktorých plánovanie ovplyvňuje výšku nákladov.

Dynamickosť trhu a ich vzájomné prepojenie je úzko spojené s riadením a optimalizáciou prepravných trás, preto je nesmierne dôležité efektívne plánovanie a optimalizácia. Nesprávne rozhodnutia nielen v oblasti logistiky môžu ovplyvniť všetky procesy od obstarania materiálu až po doručenie hotového výrobku alebo služby konečnému spotrebiteľovi. Majú vplyv aj na výšku kapitálu viazaného v zásobách, personál pracujúci v danej oblasti a v neposlednom rade sa zvyšujú riziká spojené s týmito procesmi. Náročnosť týchto procesov dnes vyžaduje okrem znalostí pracovníkov aj pomoc systémov na podporu rozhodovania a plánovania. Často sa stáva, že pracovník nie je schopný pri množstve objednávok optimálne splniť danú úlohu, napríklad rozviezť tovar všetkým zákazníkom. To vplýva na zníženie efektivity, zvýšenie nákladov a samozrejme aj na „goodwill“ dobré meno spoločnosti nielen v očiach konečného spotrebiteľa ale aj subdodávateľov a obchodných partnerov.

Spojením pokročilej analýzy, modelovania dát a iných techník možno vytvoriť aplikácie, ktoré budú spĺňať nielen jedno kritérium, ktoré si zvolíme ako napríklad najnižšia prejazdená vzdialenosť alebo najnižšia spotreba PHM ale možno tak vyhovieť viacerým požiadavkám a ich kombinácii. Najpokročilejšie logistické aplikácie vedia trasu naplánovať, splniť všetky obmedzenia a tak v konečnom dôsledku zvýšiť efektivitu s čím je úzko spojený zisk a úspech v konkurenčnom prostredí.

Optimalizácia prepravných trás sa však dotýka aj každodenného života jednotlivcov. Jazda autom či na bicykli do práce, prípadne plánovanie dovolenky. Optimalizácia trasy pomocou navigácie je jednoduchá.

Stačí zadať body, ktoré chceme navštíviť a výsledok sa objaví rovno pred našimi očami. Počítače dokážu takúto úlohu vyriešiť za zlomok sekundy.

Jednoduché úlohy dokonca vieme vyriešiť aj bez pomoci technických prostriedkov ale zložité úlohy by sme nevyriešili. A ak aj áno tak čas na ich riešenie by bol veľmi dlhý. Či už by sme sa rozhodli riešiť problém pomocou počítača alebo nie budeme na to potrebovať vhodný algoritmus.

V prvej kapitole, budú prezentované okružné úlohy a ich modifikácie. Predstavený bude algoritmus pomocou, ktorého budú riešené praktické príklady. Tiež bude obsahovať historický prehľad a súčasné trendy z oblasti riešenia okružných úloh.

Cieľom tejto práce je poskytnúť prehľad v oblasti okružných úloh a prezentácia vybraného algoritmu na konkrétnych úlohách s dodatočnými ohraničeniami. Medzi ciele tejto práce sa tiež zaraďuje upozornenie na potrebu riešenia okružných úloh v prostredí domácností.

V tretej kapitole budú prezentované algoritmy a metódy riešenia okružných úloh. Podrobnejšie bude popísaný algoritmus Clarka-Wrighta a jeho modifikácia pri riešení úloh s dodatočnými ohraničeniami.

Vo štvrtej kapitole budú prezentované výsledky a postup pre riešenie okružných úloh. Tie budú porovnané a vyhodnotené. Nna základe získaných výsledkov budú navrhnuté ďalšie možnosti riešenia.

1. Súčasný stav riešenej problematiky v SR a zahraničí

Optimalizácia prepravných trás môže obsahovať viacero skupín problémov. Jednou z nich je úloha hľadania najkratšej cesty. Tento problém sa považuje za jeden z najznámejších problémov v rámci optimalizácie prepravy. Jeho riešenie má podstatu v nájdení najkratšej cesty z jedného vrcholu v grafe do iného vrcholu. Takúto úlohu môžeme ešte rozdeliť na dva typy. Prvým z nich je problém *Single-source shortest-path problem* (problém hľadania najkratšej cesty z jedného uzla v sieti), ktorý predstavuje hľadanie najkratšej cesty z východiskového uzla do všetkých uzlov v sieti. Na jeho riešenie sa používa Dantzigov algoritmus, Fordov-Fulkersenov algoritmus alebo Dijkstrov algoritmus. Úloha výpočtu najkratších ciest medzi všetkými uzlami v sieti nazývaná tiež *All-pairs shortest-path problem* (problém hľadania najkratšej cesty medzi všetkými uzlami) sa zase rieši pomocou Floydovho algoritmu alebo algoritmu pre minimálne sčítanie matíc.

Druhým typom úloh je hľadanie najkratšej okružnej cesty. V tomto prípade treba určiť, ktorý z uzlov v sieti bude východiskovým a zároveň aj koncovým pričom musí byť splnená podmienka, že každým uzlom v sieti prejdeme práve raz. Dôležitá je ale aj minimalizácia dĺžky trasy. Takéto úlohy sa využívajú pri plánovaní umiestňovania obslužných centier a môžu sa označovať ako *úlohy obchodného cestujúceho.* Na riešenie problému obchodného cestujúceho doteraz nebol nájdený exaktný polynomiálny algoritmus. Existuje však množstvo metód ako optimalizačné, heuristické, metaheuristické a kombinované algoritmy, ktoré dokážu vyriešiť problém obchodného cestujúceho v prijateľnom čase.

Veľmi často riešenými úlohami sú úlohy riešiace rozvoz a zvoz označované ako *Vehicle Routing Problem* (okružné úlohy) – VRP. Môžu sa zaradiť medzi úlohy okružných jázd alebo úlohy obchodného cestujúceho. Riešia aj úlohu umiestnenia obslužného centra, tak aby náklady boli čo najnižšie.[4]

Optimalizácia prepravných trás sa väčšinou spája so splnením viac ako jednej podmienky.

Úlohy obchodného cestujúceho môžu byť doplnené o časové okná, spätný zber, určené časy doručenia a iné dodatočné podmienky. Na riešenie takýchto problémov sa potom využívajú modely viackriteriálneho programovania.

Matematické problémy týkajúce sa problému obchodného cestujúceho boli v roku 1800 spracované írskym matematikom Sirom Williamom Rowanom Hamiltonom a britským matematikom Thomasom Penyngtonom Karkmanom. Všeobecná forma úlohy obchodného cestujúceho bola študovaná matematikmi už v 30tych rokoch 20.storočia, Karlom Mengerom vo Viedni a Harvarde. Problém neskôr podporili aj Hassle Whitnsy a Merrill Flood v Princetone.

Problém obchodného cestujúceho je veľmi starým problémom zasahujúcim do informatiky a je rozšírením problému Hamiltonovského cyklu. Dotýka sa aj oblastí ako teória zložitosti. Riešenie tohto problému nemožno nájsť v polynomiálnom čase čo znamená, že čas potrebný na nájdenie riešenia rastie exponenciálne s lineárne sa zvyšujúcim počtom vrcholov. Keď by sme teda chceli nájsť riešenie úlohy, ktorá by mala za úlohu navštíviť 12 miest, mali by sme 20 miliónov riešení.

Doteraz však nebol nájdený žiadny efektívny algoritmus a niektorý odborníci sa domnievajú, že ho nebude možné nájsť nikdy.

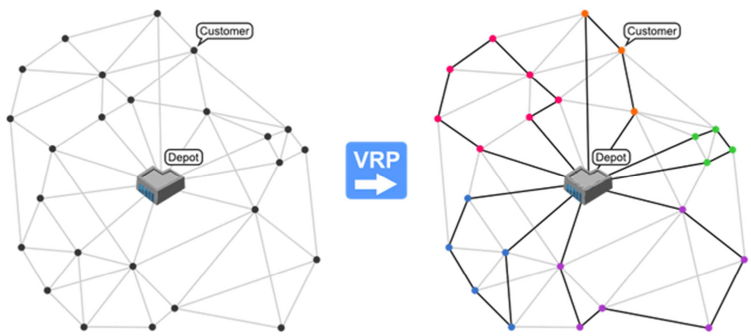
Úlohy obchodného cestujúceho sa zaraďujú medzi okružné úlohy. Sú častým praktickým problémom, ktorý hľadá riešenie napríklad pri probléme odvozu odpadu, rozvoze pekárenských výrobkov, donáškových službách, plánovaní práce strojov a iných.[5]

Problém prepravných úloh ďalej v práci označovaný ako VRP (okružná úloha) je jednou z najnáročnejších úloh viakriteriálnej optimalizácie. Jeho úlohou je nájdenie optimálnej trasy pre vozidlá pri rozvoze a zvoze tovaru zákazníkom. O tomto probléme sa diskutuje už viac ako 50 rokov a jeho náročnosť a úzka spojitosť s reálnymi problémami zabezpečuje pretrvávajúci výskum v oblasti viackriteriálnej optimalizácie prepravných trás. Zaraďuje sa medzi kombinačné, optimalizačné a celočíselné programovacie problémy.

Od roku 1959 (Dantzig) sa problém VRP stal dôležitou súčasťou logistiky, distribúcie a prepravy. Problém smerovania vozidiel je všeobecný názov, ktorý sa venuje celej skupine problémov, pri ktorých je potrebné určiť súbor trás pre vozidlá, ktoré vychádzajú z jedného alebo viacerých dep. Cieľom je minimalizovať náklady na rozvoz tovaru a zároveň splnenie požiadaviek zákazníkov.

Na obrázku vidíme typický vstup pre problém VRP a jeden z jeho možných výstupov:

Obrázok 2: Okružná úloha a jej možné riešenie



ZDROJ: http://neo.lcc.uma.es/vrp/vehicle-routing-problem/

Rozvozné úlohy riešia distribúciu medzi skladmi a zákazníkmi pomocou dopravnej siete a dostupných vozidiel. Ďalšími cieľmi týchto úloh môže byť zvoliť optimálny počet vozidiel alebo optimálnu trasu z rôznych hľadísk ako nákladovosť, čas, počet zastávok, vyriešenie resp. obslúženie všetkých objednávok a iné. Výsledkom úspešne vyriešenej úlohy je potom rozvozný plán a trasy pre jednotlivých vodičov. Okružné úlohy sa dajú riešiť pomocou viacerých metód. Medzi exaktné metódy patria Clark-Wrightova metóda, algoritmus založený na základe zhody (Matching based savings algorithm) a zlepšovacie techniky viacerých ciest (Multi-route). Ďalším spôsobom sú dvojfázové algoritmy, ktoré riešia problém rozdelením na dva komponenty. Jedným komponentom je vytvorenie trasy a druhým zoskupenie vrcholov. Na riešenie s primárnym zoskupením vrcholov a sekundárnym vytvorením trasy možno použiť Fische a Jaikumar algoritmus, Petal algoritmus alebo Sweep algoritmus. Tieto úlohy možno riešiť aj pomocou metaheuristík, ktoré kladú dôraz na hlboké skúmanie. Patria sem napríklad mravčie kolónie, deterministické žíhanie, simulované žíhanie, tabu vyhľadávanie ale aj programovanie s ohraničujúcimi podmienkami.[6]

Okružné úlohy môžeme tiež nazývať aj problémom smerovania vozidiel. Ako už bolo spomínané, spočíva v navrhovaní trás pre vozidlá s cieľom minimalizovať náklady.

Tento problém má ústredné miesto v riadení distribúcie a logistiky a každodenne ho používajú desiatky tisíc prepravcov po celom svete. Problematika sa vyskytuje vo viacerých formách z dôvodu rôznych obmedzení, s ktorými sa stretávame v praxi. Viac ako 50 rokov tento problém priťahoval pozornosť veľkej skupiny pracovníkov z oblasti operačného výskumu. Čiastočne k tomu prispieva ekonomický význam problému, ale aj výzvy, ktoré sa s riešením tohto problému spájajú. Napríklad TSP je špeciálnym prípadom VRP, ktorý môže riešiť tisíce vrcholov. VRP sa odlišuje náročnosťou a je definovaný ako NP-tažký problém. Napríklad v jednoduchom prípade riešenia úlohy s kapacitným obmedzením je so zvyšujúcim sa počtom uzlov úloha náročnejšia a nájdenie jej riešenia sa stáva problematickejším.

CVRP (Capacited Vehicle Routing Problem) úloha s obmedzením kapacity bola predstavená v roku 1959 Dantzigom a Ramserom. Títo autori navrhli jednoduchú heuristiku založenú na zhode pre jej riešenia a ilustrovali ju na príklade veľkosti hračky. V ďalších rokoch sa postupne objavovali heuristiky založené na rôznych princípoch, ako sú úspory, geografická blízkosť, zákaznícke zhody, zlepšenia v rámci trasy, medzipodnikové zlepšenia. Najznámejšou heuristikou je heuristika založená na úsporách od Clarka a Wrighta, ktorý ju predstavili v roku 1964. Táto heuristika dokázala odolať požiadavkám rýchlosti, jednoduchosti a primerane dobrej presnosti nájdeného riešenia. Vývoj presných algoritmov sa začal v roku 1981 vydaním dvoch článkov. V prvom bol navrhnutý algoritmus založený na dynamickom programovaní s relaxáciou stavového priestoru a v druhom boli navrhnuté matematické formulácie využívajúce q-cesty a najkratšie stromy. O niekoľko rokov neskôr v roku 1984 bol navrhnutý prvý prístup založený na princípe rezných nadrovín a celočíselného modelu. Tieto prístupy sa potom stali základom pre vznik novších algoritmov.

Odvtedy boli navrhnuté rôzne exaktné algoritmy založené na matematickom programovaní. Niektoré formulácie obsahujú premenlivé veličiny a často sa riešia metódami vetvenia a rezov. Okružné úlohy tiež môžu byť formulované ako problémy rozdelenia množín. V roku 2006 Fukasawa a spol. využili postupy založené na tejto metodike pri riešení VRP a tieto implementácie sa považujú za jedny z najúspešnejších. Rovnako ich využil aj Baldicci a spol, v roku 2008.

Vývoj modernej heuristiky pre riešenie úloh VRP sa začal spolu s nástupom metaheuristík. Dokonca môžeme povedať, že snaha o riešenie týchto úloh mateheuristikou stimulovalo rast a pochopenie niektorých heuristických konceptov, ktoré teraz poznáme. Prvotný výskum v tejto oblasti však bol roztriešteny a postupom času sa niektoré algoritmy začali racionalizovať. Za najlepšie metaheuristiky možno považovať také, ktoré súčasne vykonávajú rozsiahle a hlboké vyhľadávania riešení a naraz riešia niekoľko variantov problému. Vo všeobecnosti buď uplatňujú viac operátorov, ako pri adaptívnom vyhľadávaní, alebo kombinujú genetické algoritmy s miestnym vyhľadávaním, ako v prípade hybridného genetického algoritmu, ktorý v roku 2012 navrhol Vidal a spol.[7]

Od roku 2016 sa pozornosť zameriava okrem klasických úloh VRP s dodatočnými ohraničeniami aj na takzvaný „green" (zelený) VRP, ktorý rieši najmodernejšie rozhodovacie problémy v oblasti smerovania v súvislosti s environmentálnym aspektom. Cieľom je motivovať výskumníkov z disciplín ako je operačný výskum, manažment vedy, logistika a veda o životnom prostredí aby spolupracovali pri riešení úloh a výskume problému zelených VRP. Literatúra v tejto oblasti stále nie je postačujúca. V teoretickej rovine sa navrhujú idealizované modely, ktorých cieľom je minimalizácia medzier v reálne využívaných aplikáciách. Tieto medzery poskytujú príležitosti pre podnikové aplikácie, ktoré si ako cieľ vytýčili efektívnu a udržateľnú prevádzku. Environmentálny aspekt by sa mal brať do úvahy v navrhovaní riešenia akéhokoľvek variantu VRP, pretože udržateľnosť organizácií je základom pri ich rozvoji a činnosti. Multiobjektívne programovanie sa stáva zaujímavou alternatívou, ktorá sa implementuje vzhľadom na potrebu zohľadnenia stratégie znižovania vplyvov na životné prostredie. [8]

Nové trendy v oblasti VRP, ktoré sa stali populárne v posledných rokoch prezentuje vo svojom príspevku „*New shades of the Vehicle Routing Problem: Emerging problem formulations and Computational Intelligence solution methods*" (Nové odtiene problému okružných úloh: Formulácia problému a metódy riešenia) Jacek Mandziuk. V rámci svojho výskumu zozbieral v priebehu rokov 2015 až 2017 viac ako 200 prác v tlačenej forme a 200 prác v elektronickej forme zaoberajúcich sa problémom VRP. Z celkového počtu zozbieraných prác bolo vybraných 200, ktoré boli podrobne preštudované za účelom sledovania nových formulácii a inovatívnych riešení v oblasti VRP. Väčšina zozbieraných príspevkov pochádzala z časopisov. Tabuľka 1 uvádza rozdelenia publikácií riešiacich problém VRP podľa rokov a tabuľka 2 najčastejšie vyberané časopisy na publikáciu moderných prístupov v oblasti VRP. [9]

Tabuľka 1: Rozdelenie publikácií v oblasti VRP podľa rokov[1]

Rok publikácie	***Počet publikácií***
2017	30
2016	20
2015	15
2014	8
2013	8
2010-2012	7
2000-2009	12
1959-1999	15

Tabuľka 2:Časopisy v ktorých sa najčastejšie publikovali články z oblasti VRP[1]

Názov časopisu	***Počet publikácií***
European Journal of Operational Research	20
Transportation Research	8
Computers & Operations Research	7
Information Science	5
Applied Sofr Computing	5
Networks	5
Comtupers and Industrial Engineering	5
Transportation Science	5
Expert Systems with Applications	5
Journal of the Operational Research Society	5
IEEE Transactions on Cybernetic	3
IEEE Transactions on Intelligent Transportation Systems	3
Operation Research	3
IEEE Transactions On Systems Man and Cybernetics: Systems	2
IEEE Systems Journal	2
IEEE Transactions on Evolutionary Computation	2

Na základe vykonanej analýzy publikovaných dokumentov sa zistilo, že záujem o riešenie niektorých formulácií VRP narastá. Je však potrebné podotknúť, že tento rozvoj sa nepovažuje za neúspech predchádzajúcich zistení a riešení. Zakladá sa na meniacich sa podmienkach v spoločnosti a snahe prispôsobiť sa aktuálne riešeným témam. Oblasti VRP o ktoré je v poslednej dobe väčší záujem ako o tradičné formulácie a osvedčené postupy a metódy sú uvedené v tabuľke 3.

[1]ZDROJ:https://www.researchgate.net/publication/329960346_New_Shades_of_the_Vehicle_Routing_Problem_Emerging_Problem_Formulations_and_Computational_Intelligence_Solution_Methods, platné ku dňu 29.4.2019

Tabuľka 3:Počet príspevkov s vybranou tematikou v rokoch 2000 -2017[2]

	Kľúčové slová					
ROK	*last mile delivery*	*crowndshipping*	*green vehicle routing*	*autonomous delivery vehicle*	*bike sharing*	*UAV delivey*
Prvá zmienka	2008	2016	2008	*2004*	2012	2010
2000-2009	3	-	2	4	-	-
2010-2012	3	-	4	2	4	4
2013	2	-	6	2	2	1
2014	2	-	7	1	8	1
2015	0	-	10	4	14	1
2016	6	2	19	1	15	8
2017	6	2	19	7	23	11

1.1 Vehicle Routing Problem (VRP) a jeho varianty

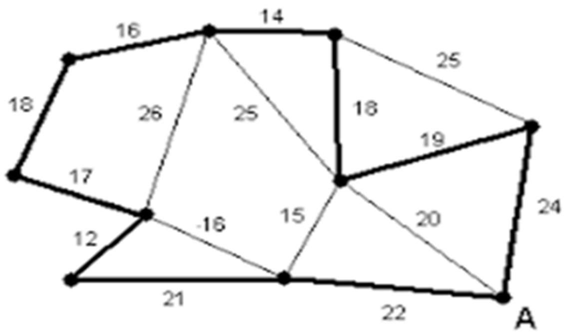
Vozidlá sa väčšinou pohybujú v dopravnom systéme. Či už vo vzduchu, po cestách, koľajniciach alebo vode, tento dopravný systém môžeme nazvať dopravnou sieťou. Dopravná alebo distribučná sieť je prezentovaná na vybranej zemepisnej oblasti orientovaným grafom. Graf sa skladá z dvoch základných prvkov a to uzol a hrana. Tieto prvky sú medzi sebou prepojené. Hrany tvoria spojnice medzi uzlami. Každý z uzlov je charakterizovaný určitými vlastnosťami, ktoré sa zväčša označujú číslom. Uzly sa spájajú s charakteristikami ako kapacita uzla a to v prípade, že je uzol vnímaný ako dodávateľ. V prípade odberateľa označuje uzol požiadavku. Ďalšou charakteristikou uzla môže byť náročnosť na obsluhu, spoľahlivosť, náklady a iné. Tie sa potom odvíjajú od charakteru riešenej úlohy. Dopravné siete sa modelujú pomocou grafov a digrafov pričom si môžeme stanoviť ich priechodnosť. Tá môže byť obojsmerná ale aj jednosmerná.

Základom pre riešenie okružných úloh je problém obchodného cestujúceho - The Traveling Salesman Problem (TSP). Je to najjednoduchší a pravdepodobne aj najznámejší problém riešiaci okružné úlohy. Cieľom tejto úlohy je minimalizácia celkovej dĺžky trasy, nákladov alebo vzdialenosti pri navštívení všetkých vrcholov v grafe tak, že každé mesto navštívi jeden krát. Z každého mesta sa môže dostať do každého iného mesta buď priamo alebo cez iné mesto. Všetky hrany - cesty sú ohodnotené číslom vyjadrujúcim čas, vzdialenosť, cenu a podobne. Počet vrcholov je známy a označuje sa *n*.

[2]ZDROJ:https://www.researchgate.net/publication/329960346_New_Shades_of_the_Vehicle_Routing_Problem_Emerging_Problem_Formulations_and_Computational_Intelligence_Solution_Methods, platné ku dňu 29.4.2019

Ďalej poznáme vzdialenosti medzi vrcholmi d_{ij} *(i,j=1,2, ...n),* z ktorých určíme maticu najkratších vzdialeností medzi všetkými vrcholmi označovanú D =$\{d_{ij}\}$,(i,j = 1,2, ...n).

Obrázok 2 - TSP



ZDROJ*:* https://www.ipaslovakia.sk/sk/ipa-slovnik/stihla-logistika?css=desktop

Formulácia úlohy

Tak ako úlohu najkratšej cesty v sieti aj úlohu okružnej cesty môžeme formulovať pomocou matematického programovania. Keďže sa jedná o špecifický typ priraďovacieho problému, jeho premenné sú bivalentné to znamená, že nadobúdajú hodnoty $x_{ij} \in \{0,1\}$, pričom *i,j = 1,2, ...n.* Premenná sa rovná 1 ak sa cesta realizuje, 0 ak sa cesta na príslušnej trase nerealizuje. Účelová funkcia predstavuje celkovú prejdenú vzdialenosť a má tvar:

$$f\ (x,y) = \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij}$$

Takáto formulácia okružnej cesty sa nazýva Truckehova formulácia. Z tejto formulácie vyplýva, že každý model obsahuje *n.n* binárnych premenných a *(n-1)+(n+1)+ (n-1).(n+2)* ohraničujúcich podmienok. Pri úlohe s ôsmimi uzlami by to predstavovalo 64 binárnych premenných a 56 ohraničujúcich podmienok. Toto riešenie by bolo časovo veľmi náročné a v prípade, že by úloha mala viac vrcholov takmer nerealizovateľné v reálnom čase. Preto sa na riešenie vyvinulo množstvo optimalizačných a heuristických metód, ktoré poskytujú prijateľné riešenie v dostupnom čase. Sú to napríklad metóda vetiev a hraníc alebo algoritmus najbližšieho suseda.[4]

1.1.1 Všeobecná okružná úloha bez obmedzení

VRP všeobecne rieši nájdenie najlepšej trasy alebo plánu pri obsluhe zákazníkov alebo prevoze pasažierov, tovarov alebo iných atribútov medzi jednotlivými mestami. Pritom sa snaží rešpektovať všetky zadané ohraničujúce podmienky a optimalizovať náklady spojené s riešením problému. V tejto časti sa budeme venovať opisu smerovania vozidiel inak povedané okružným dopravným úlohám. VRP je všeobecný názov používaný pre všetky typy úloh, ktoré riešia pohyb vozidiel v sieti. Základným predpokladom riešenia distribučného problému je zabezpečenie, že vozidlá, ktoré sú k dispozícii sú umiestnené v centrálnom zariadení (depe) odkiaľ môžu v danom časovom období uspokojiť požiadavky zákazníkov. Cieľom základného tvaru úlohy VRP je teda minimalizácia distribučných nákladov pri preprave. Tiež by sme ho mohli opísať ako problém hľadania optimálneho spôsobu dodania alebo zberu z depa k zákazníkom. Preto sa tento najzákladnejší typ VRP môže nazvať plánovaním vozidiel, dispečingu, alebo problém s doručovaním.

Matematický model úlohy, ktorá rieši problém obsluhy viacerých zákazníkov pri dodržaní určitých požiadaviek môžeme definovať na grafe $G=(V,A)$, kde $V=(v_1, \dots, v_N)$ predstavuje množinu vrcholov a $A = \{(i, j) : i, j \in V, i \neq j\}$ prezentuje množinu hrán prípadne oblúkov. V_0 predstavuje sklady (depá) kde sa nachádza najviac m identických vozidiel, pričom každé vozidlo má kapacitu Q. Každý vrchol v grafe okrem skladu (depa) má nezáporný dopyt q_i, nezáporný čas obsluhy. V niektorých prípadoch je počet vozidiel určený a priori. V iných je zase počet vozidiel rozhodovacou premenou. Základ úlohy VRP spočíva v nájdení okružnej trasy pre každé vozidlo, pričom:

- vozidlo vždy začína a končí v depe
- každý zákazník je obslúžený práve jedným vozidlom
- celkový dopyt akejkoľvek trasy nepresahuje Q
- celkové náklady sú minimalizované

Pre matematickú formuláciu riešenia VRP sme vybrali vzorce, ktoré definoval Fisher a Jaikumar (1981). Ako prvé definujeme premenné potrebné pre matematickú formuláciu.

Parametre: m: počet dostupných vozidiel

N: počet zákazníkov, ktorí majú byť obslúžení. Označení sú od 1 po n.

Depo má číslo 0.

Q_k: kapacita vozidla k

Q_i: dopyt zákazníka

D_{ij}: vzdialenosť medzi zákazníkov i a j

Rozhodovacie premenné:

y_{ik}: binárna rozhodovacia premenná, ktorá môže mať hodnotu 1 vtedy a len vtedy, ak riešenie že vozidlo k ide od zákazníka i k zákazníkovi j je súčasťou riešenia, v opačnom prípade má hodnotu 0

x_{ijk}: za predpokladu $j \geq i$ je binárna rozhodovacia premenná, ktorá môže mať hodnotu 1 vtedy a len vtedy, ak zákazník j je obslúžený s rovnakej trasy ako zákazník i pomocou vozidla k, v opačnom prípade sa rovná 0

Formulácia:

$$Min \qquad f(X) = \sum_{i=0}^{n} \quad \sum_{j=0}^{n} \sum_{k=1}^{K} d_{ij} x_{ijk}$$

$$\sum_{i=0}^{n} q_i y_{ik} \leq Q_k \qquad\qquad k \in \{1, \dots m\}$$

$$\sum_{k=1}^{K} y_{ik} = \left\{ \begin{array}{ll} k & i = 0 \\ & y_{ik} = \{0, alebo\ 1\} \qquad \mathrm{i} \in \{0, \dots \mathrm{m}\} \qquad \mathrm{k} \in \{1, \dots \mathrm{m}\} \\ 0 & i = 1, \dots, n \end{array} \right\}$$

ohraničujúce podmienky

$$\sum_{i=0}^{n} x_{ijk} = y_{jk} \quad j \in \{0, \dots n\} \qquad\qquad k \in \{1, \dots m\}$$

$$\sum_{i=0}^{n} x_{ijk} = y_{jk} \quad i \in \{0, \dots n\} \qquad\qquad k \in \{1, \dots m\}$$

$$\sum_{j=0}^{n} x_{ijk} \leq | S | -1 \quad \forall S \in \{1, \dots n\} \;\; k \in \{1, \dots m\} \;\; 2 \leq | S | \leq n-1$$

$$x_{ijk} = \{0 \; alebo \; 1\} \qquad i, j \in \{0, \dots n\} \qquad k \in \{1, \dots m\}$$

Účelová funkcia minimalizuje všetkými vozidlami prejdenú vzdialenosť. Ohraničujúce obmedzenia zabezpečujú, že kapacita vozidla nebude prekročená, že všetky vozidlá začnú aj skončia svoju trasu v depe a že každý zákazník má priradené jedno vozidlo, ktorým bude obslúžený. Ostatné podmienky zabezpečujú, že zákazník bude navštívený len jeden krát. Tieto obmedzenia sú rovnaké ako pri riešení úlohy obchodného cestujúceho.

Počas a vďaka výskumu riešenia úloh VRP v posledných rokoch sa objavujú a ukazujú nové deriváty okružných úloh. Tieto zistenia sú spôsobené predovšetkým neustálou prácou výskumníkov na úlohách distribúcie a dopravy v spoločnostiach.

V tejto časti postupne predstavíme odvodené úlohy pri riešení VRP, zabezpečené dodatočnými ohraničeniami. Základný VRP pracuje s predpokladmi ako sú homogénna skupina vozidiel, jedna okružná jazda pre jedno vozidlo, jedno navštívenie zákazníka a podobne. Po zavedení určitých prekážok do tejto základnej úlohy VRP sa tieto predpoklady odstránia a tým sa zvýši náročnosť a zložitosť problému. Tieto úlohy sa už môžu označiť ako NP-hard. Z toho vyplýva, že vo väčšine prípadov sa zavedenie každého dodatočného obmedzenia rieši samostatne bez integrácie. V ďalších častiach budú predstavené jednotlivé dodatočné ohraničenia, ktoré zavádzané samostatne.[4,10]

1.1.2 VRP s obmedzením kapacity a typu vozidla (CVRP)

Najzákladnejšou verziou VRP je kapacitné obmedzenie vozidla. CRVP môžeme popísať ako problém so zákazníkmi, ktorí majú požiadavky na doručenie dodávky zo skladu v určitom množstve, pričom je stanovená aj kapacita vozidla. Keďže kapacita vozidla je obmedzená, vozidlo sa musí pravidelne vracať do skladu aby opätovne naložilo tovar. V prípade úlohy CVRP však nemôžeme rozdeliť dodávky jednotlivých zákazníkov. Z toho dôvodu je riešením súbor vykonaných ciest vozidla, pričom každý zákazník je navštívení práve raz a je mu doručené požadované množstvo. Cieľ spočíva v nájdení súboru ciest s minimálnymi požiadavkami na čas. Gendreau a kol. (1999) navrhli riešenie úlohy VRP s heterogénnou skupinou vozidiel.

V takomto prípade heterogénneho vozového parku disponujeme vozidlami s rôznymi kapacitami a nákladmi na ich používanie. Takéto úlohy riešia aj optimálne zloženie vozového parku a odkazujú sa na riešenie úloh s heterogénnou flotilou a Heterogeneous Fleet Vehicle Routing Problem (HVRP) ale aj na zmiešané úlohy s obmedzenou flotilou Fleet Size and Mix Vehicle Routing Problem (FSMVRP). Úlohy s heterogénnym vozovým parkom (VRPHE), kde počet vozidiel istého typu *t* vo vozovom parku je obmedzený sa zameriava na optimalizáciu využitia vozového parku.

1.1.3 VRP s časovými oknami (VRPTW)

VRPTW môžeme popísať ako ten istý problém VRP s dodatočným obmedzením, ktorým je časové okno. Časové okno predstavuje časové obdobie, v ktorom sa musí uskutočniť dodávka, teda interval v ktorom musí byť konkrétny zákazník obslúžení. Interval v depe nazývame plánovací horizont.

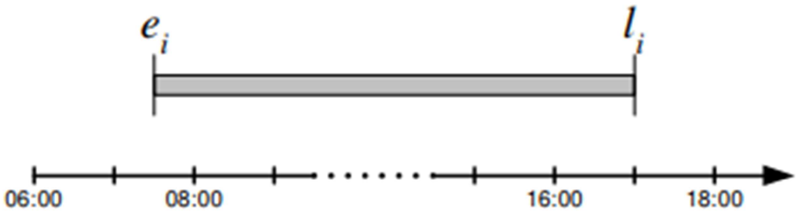
Časové okná majú tri hlavne charakteristiky a to:

- najskorší povolený čas príchodu označovaný aj ako otváracia doba e_i
- posledný povolený čas príchodu označovaný aj ako čas uzávierky l_i
- definícia tvrdosti časového okna, či je považované za mäkké alebo tvrdé[3]

Príklad zobrazený na obrázku znázorňuje požiadavku zákazníka *i* na doručenie medzi 07:30 a 17:00. Na rozlíšenie medzi skutočným časom príchodu a určeným časom príchodu premenná a_i označuje skutočný čas príchodu vozidla do uzla *i*. Ak by skutočný čas príchodu do uzla i označovaný a_i, bol skorší ako najskôr možný príchod do uzla označený e_i, muselo by vozidlo čakať určitý čas w_i, ktorý vypočítame ako $w_i=max\{0, e_i – a_i\}$.

[3]https://repository.up.ac.za/bitstream/handle/2263/26439/02chapter2.pdf?sequence=3&isAllowed=y

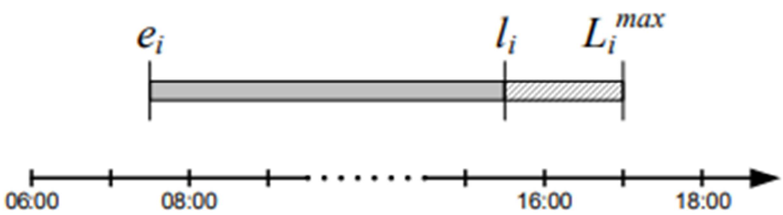
Obrázok 3: Tvrdé časové okno



ZDROJ:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.879.8176&rep=rep1&type=pdf

Ak je stanovený najskorší a najneskorší povolený príchod, uvádza sa časové okno ako obojstranné. Ak nie sú povolené žiadne príchody mimo uvedených parametrov, časové okno označujeme ako tvrdé. Tvrdé časové okno je zobrazené na obrázku 3. Ak je dodávka povolená aj mimo špecifikovaného časového okna, tak sa časové okno považuje za mäkké a zákazník *i* môže penalizovať oneskorenie dodatočnými nákladmi α_i. Zákazník *i* môže určiť maximálne oneskorenie L_i^{max}. Na obrázku 4 je zobrazené mäkké časové okno, kde môžeme vidieť že zákazník *i* určil čas dodávky medzi 07:30 a 15:30 ale určil aj interval oneskorenia do 17:00.

Obrázok 4: Mäkké časové okno

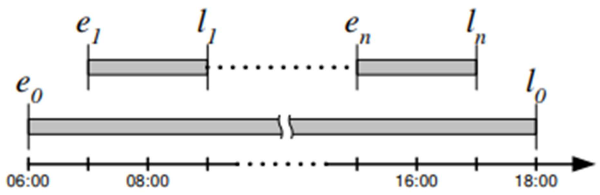


ZDROJ:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.879.8176&rep=rep1&type=pdf

Tvrdé časové okno preto môžeme považovať za špecifický typ mäkkého časového okna, kde sa čas oneskorenia rovná nule, $L_i^{max} = 0$. Ak vozidlo príde po poslednom povolenom čase príchodu l_i ale pred maximálnym omeškaním L_i^{max}, omeškanie u zákazníka *i* môžeme vypočítať ako s $L_i = max\{0, a_i - l_i\}|a_i \leq L_i^{max}$. Omeškanie je penalizované doplnením takzvaného trestného výrazu do minimlizačnej funkcie.

Časové okná je možné špecifikovať aj pre depo. V takomto prípade sa určujú otváracie, prevádzkové hodiny. V tomto prípade sú na obrázku 5 zobrazené časové okná pre zákazníka aj pre depo. V prípade depa je však časové okno určené od 06:00 do 18:00, zatiaľ čo obsluha prvého zákazníka môže nastať medzi 07:00 a 09:00 a obsluha posledného zákazníka v časovom okne od 15:00 do 17:00.

Obrázok 5: Časové okno pre depo

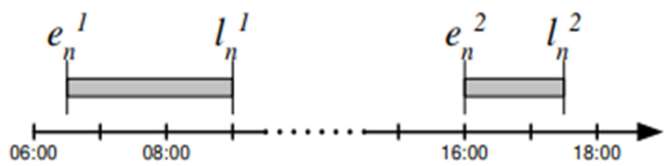


ZDROJ:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.879.8176&rep=rep1&type=pdf

Ak zákazník určí viac časových okien, symbol indexovania *a* sa zavedie ako horný index k najskoršiemu a poslednému povolenému času príchodu, kde $a \in \{1, 2, \dots, A\}$ v ktorom *A* označuje maximálny povolený počet časových okien každého zákazníka.

Príklad kedy zákazník *n* požaduje doručenie buď v čase od 06:30 do 09:00 alebo v čase od 16:00 do 17:00 je zobrazený na obrázku 6.

Obrázok 6 - Viac časových okien



ZDROJ:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.879.8176&rep=rep1&type=pdf

Takýto prípad je typický pre zákazníkov, ktorý žiadajú doručenie domov mimo otváracích hodín.

1.1.4 Multi depot VRP

Ďalšou z možných modifikácií VRP je pridanie skladov, z ktorých sú zákazníci obsluhovaný. Pre získanie ešte nižších finančných a časových nákladov sa v realite často krát vyskytujú úlohy s viacerými skladmi. Takéto úlohy označujeme ako Multi-Depot Vehicle Routing Problem. Tieto úlohy predpokladajú existenciu vozového parku, ktorý nie je založený na jednom vozidle v jednom depe, ale na viacero vozidlách vo viacero depách. Tak ako v predchádzajúcich prípadoch na riešenie VRP úloh s dodatočnými ohraničeniami existuje viacero metód.

V tomto prípade použitia algoritmu Clarka-Wirghta je potrebné najprv zákazníkov rozdeliť do skupín a priradiť im depo. Zákazníci sú potom obsluhovaný vozidlom z prideleného depa a do tohoto depa sa vozidlá aj vracajú.[12]

1.1.5 Pick up and delivery VRP

Táto modifikácia úlohy VRP sa môže rozdeliť na 3 podskupiny. Všetky tri prípady sa zaraďujú medzi úlohy označované ako NP-hard. Prvým prípadom je VRP with backhouls, druhým je VRP with backhouls and mexidload a posledným je prípad VRP with simultaneous pickup and delivery.

VRP with backhouls - okružné úlohy so spätným zberom. Zákazníci sú rozdelený do dvoch skupín. Jedna skupina má požiadavky na dodanie tovaru a druhá skupina má požiadavky na vyzdvihnutie tovaru. Pri riešení sú najprv vybavený zákazníci, ktorí očakávajú dovoz tovaru a až potom nasleduje vyzdvihnutie tovaru. Toto poradie je určené aj z dôvodu, že nakladanie tovaru počas prvej etapy by mohlo spôsobiť vykladanie neefektívnym. Ak by sme museli pred vyložením tovaru vyložiť už naložený tovar. Čas obsluhy by sa tak predĺžil. Cieľom je nájsť optimálnu trasu pri minimalizácii nákladov tak, aby boli obslúžení všetci zákazníci - tovar bol doručený aj vyzdvihnutý a nebola prekročená kapacita vozidla.

VRP with backhoul and mexidload - je podobná úloha ako VRP so spätnou nakládkou. Zákazníci sú rovnako rozdelený na dve skupiny. V tomto prípade však neberieme do úvahy predpoklad, že najprv musia byť tovary vyložené a následne naložené. Dodávky a vyzdvihnutia môžu byť vykonávané v ľubovoľnom poradí. Tento fakt však spôsobuje kolísavé zaťaženie vozidla.

Stále však zostáva zachovaná požiadavka nájsť optimálne riešenie pri dodržaní kapacity vozidla a minimalizácii nákladov spojených s rozvozom a zvozom tovaru.

VRP with pickup and delivery - je problém učenia smerovania vozidla pri súčasnom odbere a doručení tovaru zákazníkovi. V tomto prípade sa však nedelia na dve skupiny ale každý zákazník reprezentuje aj odberateľa aj dodávateľa, pričom každý požaduje doručenie určitého množstva a vyzdvihnutie určitého množstva tovaru. Úlohu môžeme nazvať aj kompletnou službou, kde počas jednej návštevy je vykonaná dodávka aj vyzdvihnutie. Súčasná nakládka a vykládka môže tak môže znamenať zníženie manipulačných nákladov. Riešenie úlohy je prípustné v prípade, že je počas celej jazdy dodržaná kapacita vozidla a rovnako aj podmienky prípustnosti vykladania a nakladania komodít.[11]

1.1.6 Multi comodity VRP

Vo svete však neexistuje len jeden druh výrobku. Existuje mnoho druhov a typov výrobkov, ktoré zákazníci požadujú. Multi comodity VRP sa teda snaží uspokojiť rôzne požiadavky rôznych druhov výrobkov tak aby boli náklady minimálne. Existuje viacero spôsobov ako sa môžu tieto úlohy rozdeliť. Prvým typom úloh je viacero druhov tovarov, ktoré sú rozvážané v tom istom vozidle.

Druhým typom je viacero druhov tovarov ale každý druh je rozvážaný pomocou samostatne, pomocou vlastného vozidla, teda sa predpokladá heterogénny vozový park.[14]

1.1.7 VRP s neistotou

V predchádzajúcich častiach boli popísané úlohy VRP, ktoré nezohľadňovali neistotu v riešení. V realite sa často krát objavuje situácia kedy nemáme istotu a preto sa niektorí začali zaujímať aj o zavedenie neistoty do riešenia VRP. Napríklad Powell (2003) sa zameriava na výskum smerovacích a plánovacích algoritmov, ktoré zachytávajú neistotu v plánovaných budúcich rozhodnutiach. Táto oblasť výskumu je mimoriadne mladá. Niektorí autori pracujú na úlohách, v ktorých vozidlá rozvážajú a zbierajú náhodné množstvá tovarov, obsluhujú neurčité množstvo zákazníkov a dokonca ich aj navštevujú v náhodných časoch. V súvislosti s týmito podmienkami potom vznikajú určité pokuty v závislosti od presahujúcej dĺžky trasy, ktorá bola prejdená a označuje sa konštantou B.

Typickým príkladom je trasa prejdená mimo pracovnej doby, čo znamená pre šoféra prácu nadčas, ktorá by mu mala byť preplatená. Laporte s spol. (1992) navrhli účinný spôsob ako počítať s určitými obmedzenými nákladmi spojenými s prácou nadčas. Ak sa tieto predpokladané náklady súvisiace s prácou nadčas vezmú do úvahy, riešenie úlohy VRP sa znova stáva úlohou stochastického programovania.

Pri riešení sa postupuje v dvoch krokoch. Prvým je určenie počtu požadovaných vozidiel ako aj ich trás. V druhom kroku keď sú náhodne vybrané časy jázd a obsluhy, sa zadávajú penále za presahujúci čas pri preprave a obsluhe. Funkcia rovnako ako v predchádzajúcich častiach minimalizuje celkové náklady, ktoré zahŕňajú fixné náklady vozidla, cestovné náklady a aj predpokladané penalizačné náklady spojené s prekročením doby trvania trasy.[4,11]

1.1.8 Time-depend travel time

Pri riešení úloh VRP v reálnom svete by sme mali predpokladať aj možnosť výskytu nepredvídateľných udalostí ako napríklad dopravné nehody, prípadne poruchy vozidla. Avšak existujú aj do určitej miery predvídateľné udalosti, ktorými môžu byť napríklad preťaženie počas dopravnej špičky.

Môžeme teda povedať, že zohľadnenie týchto skutočností v modelovaní nás približuje k reálnemu plánu smerovania vozidiel a rozvozu tovaru.

Hill a Benton (1992) skúmajú dva hlavné prístupy pri odhadovaní vzdialenosti medzi dvoma uzlami i a j označenej d_{ij}. Tieto prístupy sú pomenované podľa Minkowskeho vzdialenosti a Pythagoreanovej vzdialenosti.

Vzhľadom na skutočnosť, že automobily sú najčastejšie využívaným cestným dopravným prostriedkom sa dnes v mnohých aj menších ale hlavne vo veľkých mestách čas prepravy predlžuje a to hlavne v ranných a poobedných hodinách počas pracovných dní. Ak by sa kapacita ciest zväčšila mohlo by plánovanie trasy aj v týchto obdobiach pracovať bez zohľadnenia faktu, že sa čas jazdy predlžuje. Cestné siete sú však nerovnomerne preťažené a tak bez zohľadnenia preťaženosti by čas jazdy z miesta i na miesto j nebol rovnaký.

Výskumom týkajúcim sa zohľadnenia nielen cestovného času ako funkcie vzdialenosti, ale aj denného času zahrnutého do výpočtu sa zaoberal Malandraki a Daskin (1992). Ichous a spol. (2003) začali s výskumom časovo závislých problémoch v 50-tych rokoch. Zamerali sa na úlohy Time Dependent Traveling Salesnam Problem (TDTSP) a jej podobné. Výskum problému Time Dependent Vehicle Routing Problem (TDVRP) robili Hell a spol. (1998). Nasledujúci výskum tohto problému bol realizovaný Hillom a Bentonom (1992). Ten sa zaoberá rýchlosťou jazdy, ktorá odráža rýchlosť jazdy v okolí uzlov. Všetky posiaľ uvádzané prístupy, ktoré riešili závislosť na čase pracujú s udalosťami, kedy jedno vozidlo prejde spolu s iným vozidlom po tej istej hrane. Hoci prvé vozidlo štartovalo skôr, druhé daný úsek prejde v kratšom čase ako prvé aj keď štartovalo neskôr.[4,11]

1.1.9 VRP s viacnásobným plánovaním

V reálnom svete nemôžeme počítať len s možnosťou, že keď vozidlo obslúži zákazníkov už nebude pracovať. Vozidlo vo väčšine prípadov znova navštívi sklad, doplní zásoby a vydá sa obslúžiť nových zákazníkov na novej trase. Takého úlohy sa definujú ako úlohy s viacnásobným použitím označované *Vehicle routing Problem with Multiple use of vehicles* (VRPM). Tento typ úloh zadefinoval Taillard a kol. (1996). Inou úrovňou takýchto úloh je reálnejšie riešenie, ktoré predpokladá že všetci zákazníci nemôžu byť obslúžený ale sa maximalizuje ich počet.

Sú to úlohy označované ako *Multiple Tour Maximum Collection Problem* (MTMCP). Brandao and Mercer (1997) riešili tieto úlohy viacerých ciest doplnené o časové okná. Tento viacúčelový problém smerovania označovaný ako *Milti-Trip Vehicle Routing Problem* (MTVRP) je špeciálnym prípadom viacnásobného plánovania.[15]

1.2 Metódy riešenia prepravných trás

V tejto časti sú uvedené a popísané najčastejšie používané techniky na riešenie úloh VRP. Môžeme ich rozdeliť na tri skupiny. Exaktné metódy, metaheuristiky a heuristiky. V každej z týchto skupín sa nachádza niekoľko prístupov. V nasledujúcom texte sú popísané najpoužívanejšie metódy riešenia úloh VRP. Väčšina z prístupov sa však zaraďuje do skupiny heuristických a metaheuristických metód z dôvodu, že pomocou exaktných prístupov sa nedá zaručiť nájdenie riešenia v dostupnom čase.

Naopak heuristické a metaheuristické metódy riešenia nám dokážu poskytnúť optimálne riešenie v akceptovateľnom čase.

1.2.1 Exaktné metódy

Inak povedané aj kombinatorické algoritmy sú metódy, ktorých výsledkom je optimálne riešenie. Počas riešenia postupne prechádzajú všetkými možnosťami až kým sa nedostanú k optimálnemu riešeniu. Časová náročnosť týchto algoritmov je však veľká a preto sa využívajú len na riešenie jednoduchých úloh. Medzi exaktné metódy sa zaraďujú:

- metóda vetiev a hraníc - tento algoritmus používa stratégiu rozdeľuj a panuj. Celú úlohu najprv rozdelí na podúlohy a tie rieši samostatne. Nájdené riešenia následne porovná a vyberie optimálne riešenie,
- metóda rezných nadrovín - pracuje na princípe znižovania množiny prípustných riešení zavádzaním dodatočných ohraničení. Využíva sa pri riešení celočíselných úloh,
- metóda dynamického programovania - pri tomto spôsobe riešenia sa úloha preformuluje na úlohu lineárneho - matematického programovania, kde sa pomocou určených rovností, nerovností a účelovej funkcie hľadá optimálne riešenie.

1.2.2 Metaheuristické metódy

Metaheuristické metódy sa považujú za najkvalitnejšie zo všetkých metód. Je to vďaka tomu, že nie sú viazané na riešenie konkrétnych problémov ale sa zameriavajú na všeobecné riešenie a postupy pri hľadaní optimálneho riešenia.

Metaheuristické metódy fungujúce na princípe pamäte:

- Tabu Search - je jednou z najpoužívanejších heuristík. Pracuje s krátkodobou pamäťou, vďaka ktorej sa presúva do susedných riešení, ktoré sú lepšie. Takto predchádza aj zacykleniu.
- VNS - Variable neighbourhood seach - pomocou tejto metódy sa riešia úlohy kombinatorických a globálnych optimalizačných problémov. Zakladá sa na skúmaní okolia aktuálneho riešenia aby tak mohlo nájsť optimálne riešenie.

- Adaptívna pamäť - pomocou uchovávania navštívených riešení a ich komponentov vytvára nové optimálne riešenia. Tie potom využíva na aktualizáciu pamäte.

Metaheuristické metódy založené na evolučných algoritmoch:

- Genetické algoritmy - sú založené na teórii biologickej evolúcie, na Darwinovej teórii, ktorá spočíva vo výbere, mutácii, a krížení najsilnejších jedincov kolónie. Tí vo svojich chromozómoch nesú kód, ktorý sa dedí a tak sa každá generácia stáva lepšou.

- Mravčie kolónie - táto metóda sa zakladá na spolupráci pri hľadaní optimálneho riešenia. Mravce skúmajú cesty a pomocou pachu si mravce nechávajú signály. Vďaka sile pachovej stopy vedia prísť k potrave a teda nájsť optimálne riešenie.

1.2.3 Heuristické metódy

Heuristické metódy nezaručujú získanie optimálneho riešenia. Vykonávajú relatívne obmedzené skúmanie v rámci hľadania riešenia. Poskytujú však riešenie blízke optimálnemu riešeniu v prípustnom čase.

Konštruktívne metódy:

- Clark a Wright algoritmus - je jedným z najznámejších heuristických metód používaných pre riešenie úloh VRP. Pracuje na princípe úspor a využíva sa na riešenie úloh, kde počet vozidiel nie je stanovený pevne.

- Matching Baed algoritmus - tento prístup zavádza do riešenia štandardného úsporného algoritmu zlúčenie, kedy sa pri každej iterácii vypočítava úspora získaná zlúčením dvoch ciest

- Multi-route zlepšovacie heuristiky - pokúšajú sa nájsť optimálne riešenie pomocou zamieňania vrcholov a hrán medzi jednotlivými plánovanými trasami vozidiel. Pričom existuje viacero prístupov k týmto výmenám. Napr. Thompson a Psaraftis, Van Breedam, Kinderwater a Savelsberg.

Dvojfázové algoritmy: problém je rozdelený na dve časti.

- Cluster first, route second - v prvom kroku rozdeľujú zákazníkov a až následne priraďujú trasy. Napríklad Sweep algoritmus rozdeľuje v prvom kroku zákazníkov geometricky, Fisher a Jaikumar algoritmus rozdeľuje zákazníkov podľa generalizovaného problému riadenia. Medzi tieto algoritmy patrí aj Taillard a Petal algoritmus, ktorý je prirodzeným rozšírením Sweep algoritmu.

- Route-First Cluster-Second algoritmus - pracujú na opačnom princípe a teda najprv vyberajú trasu a potom zoskupujú vrcholy. Tento algoritmus pracuje s neobmedzeným počtom vozidiel a v druhej fáze rieši úlohy najkratšej cesty v čase. [6]

1.3 Metódy úspor – algoritmus Clarka-Wrighta

Heuristických metód riešenia úloh VRP je mnoho. Jednou z najznámejších sa však stala metóda Clark-Wright, ktorú poznáme pod názvom metóda úspor. Bola vyvinutá v roku 1964 a bola prvou metódou, ktorá sa postupne rozvíjala. V prvých rokoch bola známa ako Wright-Fletcher-Clarke algoritmus (1968) alebo Fletcher-Clarke-Wright algoritmus. Fletcher a Clarke v roku 1963 vydali článok Conference of Operational Research Society held in Nottingham, kde na ilustratívnom príklade použili vlastný algoritmus, ktorý dosahoval lepšie výsledky ako algoritmus Dantziga a Ramsera. Prístup Dantziga a Ramsera spočíval v hľadaní dvojíc a následnom spájaní už nájdených dvojíc. Clark a Wright zakladali na znížení vzdialenosti medzi dvojicami uzlov a spájaní všetkých zákazníkov.

Špecifický príklad uvedený v ich príspevku z roku 1964 zahŕňal 30 zákazníkov obsluhovaných z depa na predmestí v Manchestri. Výsledné riešenie úlohy pomocou ich algoritmu sa zmenilo z pôvodných 10 trás, v ktorých bolo prejdených 1766 míľ na 8 trás s celkovou dĺžkou 1427 míľ.

Niekoľko autorov, sa časom zameralo na modifikovanie úsporných algoritmov. Snažili sa prispôsobiť vzorec úspor, urýchliť výpočtový čas a zlepšovať proces zlučovania trás. Algoritmus tiež prispôsobovali modifikovaným úlohám VRP. Medzi základné úpravy úsporných algoritmov môžeme zaradiť:

- AVRP – asymetrická okružná úloha nepredstavuje žiadne ťažkosti. Rozšírenie algoritmu predstavuje len zmenu symetrickej matice vzdialeností a matice úspor na asymetrickú. Táto modifikácia sa často vyskytuje v úlohách mestkých oblastí, kde sa môžu nachádzať jednosmerky a iné dopravné obmedzenia. Týmto rozšírením sa zaoberal v roku 1996 Vigo.
- MDVRP – multi depot okružná úloha predstavuje rozšírenie o viacero skladov. Pôvodne tento problém riešili Tillman a Cain (1972). Uvažovali o množstve úspor, ktoré by sa dosiahlo pridaním depa a ich prepojením s odbernými miestami v ich blízkosti. Úspory pri obsluhe z jedného skladu tak boli nižšie ako úspory pri obsluhe z druhého skladu.
- VTPRW – k riešeniu úloh s časovými oknami bolo pristupované z viacerých hľadísk. Prvotný prístup Solomona (1987) mal zväčšiť úspory metódou paralelného sporenia. Balakrishnan (1993) sa zase zameriaval na riešenie mäkkých časových okien. Ďalšími autormi, ktorý modifikovali úsporné metódy pri riešení VRPTW sú Arkinson (1994,1998) a Bräysy(2002).
- VRPB – úlohy so spätným zberom. Prvé prístupy k tomuto problému rozšírením metódy úspor boli definované v roku 1984 Deifom a Bodinom. Základom je doplnenie obmedzenia, ktoré zabezpečí uskutočnenie všetkých dodávok pre zberom. V roku 2007 Wassan predstavil algoritmus, ktorý vyvinul spolu s Osmanom (2002), ktorého základom je heuristika vkladania (SIH) a heuristika ukladania (SAH).
- VRPPD – úlohy s rozvozom a zvozom. Metóda úspor je rovnako základom riešenia tohto problému. V roku 2003 sa ním zaoberal Gronalt a spol. Autori prezentovali 4 algoritmy. Prvým bol priama adaptácia metódy úspor, ktorá sa však nedala využiť z dôvodu viacerých skladov. Druhý sa nazýva *opportunity savings algorithm*, ktorý spolu s tretím zohľadňujú náklady spojené s výberom kombinácií objednávok. Tretí algoritmus je označovaný ako simultánne úspory a posledný je kombináciou druhého a tretieho prístupu. [23]

Okrem týchto modifikácií existujú mnohé ďalšie ako napríklad okružné úlohy s náhodnými požiadavkami a náhodnými časom.

2. Cieľ práce

Táto diplomová práca sa bude zameriavať na riešenie vybraného typu prepravných trás doplnených o dodatočné ohraničenia a ich aplikáciu na konkrétnom príklade pomocou už známeho algoritmu.

Cieľom teoretickej časti tejto diplomovej práce je poskytnúť komplexný pohľad na riešenie problematiky dopravný trás. Predstavíme v nej históriu okružných úloh ale aj súčasnú situáciu a trendy z oblasti riešenia okružných úloh a algoritmov založených na úsporách. Je daný prehľad najviac skúmaných okružných problémov v posledných rokoch. Okružné úlohy definujeme pomocou matematickej formulácie a podrobnejšie popíšeme okružné úlohy s vybranými dodatočnými ohraničeniami. V časti metodika predstavíme postup riešenia okružných úloh. Podrobnejšie sa budeme venovať algoritmu Clarka – Wrighta a jeho modifikácii, ktorý bude v praktickej časti využitý na nájdenie okružných jázd.

Cieľom praktickej časti tejto práce je aplikovanie algoritmu Clarka-Wrighta na okružné úlohy s dodatočnými ohraničeniami. Konkrétne na úlohy s kapacitným ohraničením, s viacerými depami a úlohy s časovými oknami. Modelová situácia v praktickom príklade bude vychádzať z prostredia domácností, ktoré riešia okružné úlohy bez znalosti o tom, že tieto úlohy riešia. Získané výsledky budú porovnané a vyhodnotené.

Prepravné trasy a problém ich hľadania sa na podnikovej úrovni v oblasti logistiky rieši každodenne a neodmysliteľne k nej aj patria. Oblasť osobného života však do tejto problematiky nie je zapojená až v takom rozsahu akoby sa žiadalo a preto je v neposlednom rade cieľom tejto práce poukázať na potrebu riešenia prepravných a okružných trás na úrovni obyvateľstva.

3 Metodika a metódy skúmania

V tejto časti bude podrobnejšie popísaný algoritmu Clarka – Wrighta, ktorý je v praktickej časti využitý na hľadanie riešenia. Bude predstavená jeho formulácia a postupnosť krokov. Následne jeho modifikácia na riešenie okružnej úlohy s časovými oknami. Posledná časť opisuje postupnosť krokov pri K-means clusteringu, pomocou ktorého sme v praktickej časti rozdelili všetky uzly na dve skupiny.

3.1 Clark – Wright algoritmus

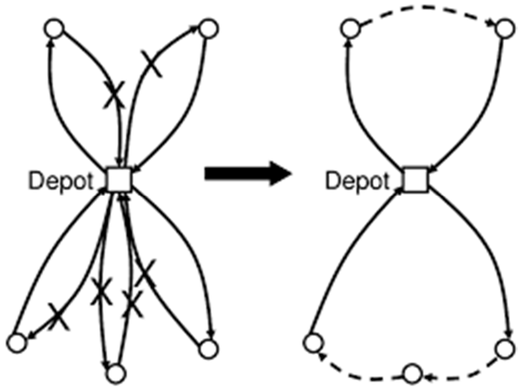
Metóda riešenia úloh Clark-Wright je konštruktívna metóda, ktorá pracuje na princípe hľadania úspor. Tieto úspory získava prostredníctvom spájania dvoch ciest, pričom práve spojením vzniká úspora. Patrí medzi najrozšírenejšie metódy používané na riešenie dopravných úloh. Prvý krát bola publikovaná v roku 1968 pod názvom *Clark - Fretcher –Wright* algoritmus, anglickými vedcami. V tom období bola lepšou alternatívou k algoritmu Dantzia a Ramsera.

Heuristický prístup spočíva v postupnom spracovaní východiskového riešenia, pričom toto riešenie predstavuje cestu *sklad – zákazník – sklad.* Sklad v tomto prípade prezentuje uzol s indexom nula. V praxi sa často používa aj názov depo. Zákazníkov prezentujú ostatné uzly v sieti. Ďalším krokom je nájdenie úspor medzi dvoma uzlami s ohľadom na stanovené ohraničenia. Ak sa tieto dva uzly dajú spojiť do jednej trasy, vzniká tak úspora a celková trasa sa skracuje. Množstvom vykonaných iterácií sa tak postupne dostávame k optimálnej trase. Algoritmus sa končí ak už nie je možné vykonať spojenie trás bez možnosti úspory. Algoritmus umožňuje len spojenie takých dvoch okružných jázd v uzloch, ktoré boli v pôvodných jazdách spojené so skladom. Existujú dva prístupy k riešeniu dopravných úloh pomocou tohto algoritmu. Sekvenčná verzia rieši problém budovaním len jednej trasy až pokým nie je splnená niektorá z ohraničujúcich podmienok, napríklad naplnená kapacita vozidla alebo už neexistuje ďalšia možnosť úspory. Paralelná verzia vytvára viacero trás naraz podľa vzniknutých úspor.

Pri riešení oboch verzií sekvenčnej aj paralelnej je prvý krok algoritmu rovnaký a spočíva v nájdení úspor pre všetky dvojice uzlov. V druhom kroku sekvenčnej verzie vyberáme najväčšiu úsporu a tak získavame nové riešenie. Takýmto spôsobom pokračujeme až do bodu kedy neexistuje ďalšia vhodná cesta.

Pri paralelnej verzii v kroku dva usporiadame úspory a postupne ich prechádzame zhora nadol. Pre každú úsporu potom vyberieme trasu medzi dvoma uzlami tak aby neboli porušené ohraničujúce podmienky.[19]

Obrázok 8 – Príklad riešenia pomocou algoritmu Clarka-Wrighta



ZDROJ: https://www.researchgate.net/figure/Example-showing-the-use-of-the-Clarke-and-Wright-Savings-Algorithm-for-the-Vehicle_fig6_35940168

3.1.1 Formulácia základného algoritmu Clarka-Wrighta

Pre to aby sme mohli daný algoritmus doplniť o dodatočné ohraničenia je potrebné poznať jeho princíp. Základný tvar algoritmu zohľadňuje iba kapacitné obmedzenie vozidla. Postup pri riešení môžeme rozdeliť do nasledovných krokov:

1.krok: východiskovým prípustným riešením je trasa $u_1 - u_2 - u_1 - u_3 - u_1 - u_4 - u_1 \dots u_1 - u_n - u_l$, ktorú označíme ako T. Definujeme zobrazenie Q pre r okružných jázd tak, že Q_r udáva súčet všetkých požiadaviek q_i , ktoré r-tá jazda uspokojuje. Ďalej definujeme zobrazenie F_i pre všetky nestrediskové uzly, a ktoré im priraďuje číslo okružnej jazdy, do ktorej príslušný uzol patrí. V prvom kroku sú definované

$$Q_r = q_r \text{ a } F_i = i.$$

2.krok: pre všetky neusporiadané dvojice uzlov (u_i ,u_j) prepravnej siete, kde $i \neq j$ a i, $j \neq 0$, vypočítame koeficienty

$$l_{ij} = d_{i0} + d_{j0} - d_{ij} , \qquad i,j=1,1,...,n.$$

3.krok: určíme najvyššiu kladnú hodnotu l_{ij} a prejdeme k bodu 4. Ak také l_{ij} neexistuje, výpočet sa končí a výsledným riešením je trasa určená postupnosťou T.

4.krok: V prípade, že platí nerovnosť $Q(F_i)+ Q(F_j) \leq K$, položíme $l_{ij} = 0$ a zlúčime príslušné dve jazdy. Číslo novej jazdy r určíme vzťahom

$$F_r = min\{F_i,F_j\}.$$

Pre všetky uzly k zo spojených jázd $F_k = r$ platí $Q_r = Q(F_i) + Q(F_j)$.

5.krok: ak po spojení uzlov u_i,u_j aspoň jeden z nich už nebude spojený zo strediskom u_0, položíme všetky hodnoty $l_{rs} = 0$, kde buď r alebo s sa rovná číslu označenia takéhoto uzla. Takisto sa budú rovnať nule tie koeficienty l_{rs} , kde uzly u_r aj u_s patria do novovzniknutej cesty. V oboch prípadoch sa potom vrátime k bodu 2. [19]

3.1.2 VRP s časovými oknami

Ak sa doručenie u niektorých alebo všetkých zákazníkov musí uskutočniť v rámci určeného časového okna, úloha VRP sa stáva VRP s časovými oknami. Časové okná môžu byť mäkké, ale s nedodržaním času následne môžu byť spojené dodatočné náklady.

Riešenie úloh VRP pomocou algoritmu Clarka-Wrighta s dodatočným ohraničení časových okien je založené na už spomínanom základe tohto algoritmu. Základný tvar VRP bol v predchádzajúcich kapitolách doplnený o kapacitné ohraničenie. Ak chceme do úlohy doplniť ďalšie ohraničenie týkajúce sa časových okien je potrebné vybraný algoritmus upraviť. Úprava spočíva v doplnení ohraničujúcich podmienok, ktoré pri riešení musia byť splnené. Kontrola prípustnosti riešenia potom bude spočívať v nasledujúcich etapách.

Kontrola prípustnosti kapacity vozidla: Nech Q_k je kapacita vozidla, ktoré vykonáva r − tú trasu, nech Q_r je množstvo tovaru, ktoré požadujú všetci zákazníci zaradení do r − tej trasy a nech q_i je množstvo tovaru, ktoré požaduje zákazník i, ktorý má byť zahrnutý do r − tej trasy. Aby zaradenie nového uzla i do trasy nenarušilo prípustnosť riešenia, musí platiť nasledujúci vzťah

$$Q_k \geq Q_r + q_i$$

Ak tento vzťah neplatí je porušená prípustnosť daného riešenia a preto daná dvojica uzlov nemôže byť zaradená do riešenia. Algoritmus potom pokračuje výberom ďalšej dvojice v poradí s najvyššou úsporou. Ak v tomto kroku je splnená podmienka prípustnosti uzly môžu byť zaradené do trasy.

Kontrola prípustnosti časových okien: obsluha u zákazníka, ktorý požaduje dodanie tovaru v určitom presne vymedzenom čase môže začať v čase b_i. Tento čas sa nachádza v časovom okne, ktoré je definované najskorším možným začiatkom časom obsluhy e_i a najneskorším možným časom obsluhy l_i. V prípade, že vozidlo smeruje priamo z uzla i do uzla j, začiatok obsluhy b_j zákazníka j určíme ako:

$$b_j = max\ e_j;\ b_i + o_i + t_{ij} \leq l_i - o_j$$

kde t_{ij} sú priame náklady medzi uzlami i a j, teda čas, ktorý je potrebný na realizáciu cesty medzi nimi a o_i je čas obsluhy zákazníka. Časy b_i pre $i = 1,2, \dots , n$, v ktorých začína obsluha zákazníka sú rozhodujúcimi premennými.

Okrem kontroly prípustnosti musíme pri modifikácii algoritmu počítať aj s predpokladmi ako je počet vozidiel. V tomto prípade bude počet vozidiel určený na základe získaného riešenia. Okrem toho je potrebné definovať ďalšie rozhodovacie premenné, ktorými sú čas kedy vozidlo opustí sklad e_0 a čas kedy sa najneskôr môže vozidlo vrátiť do skladu l_0.

Modifikovaný algoritmus Clarka-Wrighta sa v prvých dvoch krokoch zhoduje so základným tvarom algoritmu. Rozdiel je len v tom, že namiesto jednej podmienky prípustnosti overujeme dve podmienky. V prípade, že by jedna z nich nebola dodržaná trasa nemôže byť skonštruovaná. [18]

3.1.3 Formulácia algoritmu multi-depot

VRP je veľmi dôležitým problémov v oblasti kombinatorickej optimalizácie. Jeho základný tvar predpokladá jedno depo, z ktorého sa vykonáva obsluha zákazníkov. Problém s viacerými depami, je rozšírením základného problému VRP. MDVRP predstavuje spojenie dvoch druhov úloh. Jednou je umiestnenie nového depa a druhou je klasická úloha VRP, teda smerovanie vozidiel.

Na riešenie takýchto úloh existuje tak ako v prípade základného modelu VRP viacero algoritmov.

Vzhľadom ku skutočnosti, že algoritmus Clarka-Wrighta sa na riešenie úloh s viacerými depami nadá použiť, musíme túto úlohu rozdeliť na dve časti. Prvým krokom je rozdelenie uzlov pomocou K-means algoritmu. V druhom kroku sa úloha rieši rovnako ako v prípade základného modelu VRP.[20]

Metóda K-means zaraďuje medzi metódy zhlukovej analýzy. Jednoduchým spôsobom klasifikuje vstupné dáta do vopred definovaných zhlukov *k*. Úlohou tohto algoritmu je roztriediť jednotlivé uzly v sieti do klastrov tak, aby bola minimalizovaná vzdialenosť medzi depom a ostatnými uzlami.

Základnými charakteristikami K-means metódy je efektivita a potreba špecifikovať počet zhlukov. Časovú náročnosť tejto metódy môžeme určiť ako O(tkn), kde n je počet vstupných údajov, k je počet zhlukov a t je počet iterácií. Medzi jej nevýhody patrí slabosť pri spracovaní zašumelých údajov.

Algoritmus môžeme popísať v nasledovných krokoch:

1.krok: zo všetkých uzlov v sieti vyberieme tie, ktoré budú reprezentovať depo

2.krok: každý z nasledujúcich uzlov priradíme k tomu zhluku, ktorého depo je k nemu najbližšie

3. krok: ak sú všetky uzly priradené k nejakému depu, priemerom prepočítame nové centrá zhlukov a to tak, že vzdialenosť medzi všetkými objektmi zhluku a novým centrom zhluku bude minimalizovaná

4.krok: opakujeme kroky 2 a 3, kým sa centrá zhlukov prestanú meniť.[21]

4 Výsledky práce a diskusia

V tejto časti práce je rozpracovaná problematika okružných jázd s rôznymi typmi dodatočných obmedzení. Rozdelená je do troch častí. Prvú časť môžeme nazvať výskumnou, teda sa v nej zisťuje, či je o riešenie tejto problematiky zameranej na optimalizáciu času, a zjednodušenie rozhodovania v spojení s aktivitami a povinnosťami detí záujem zo strany rodičov. V druhej časti je uvedené riešenie dopravnej úlohy s dodatočnými ohraničeniami pomocou algoritmu Clarka-Wrighta. Je v nej predstavený základný model a modifikácie algoritmu. Vo formulácii úlohy ide o riešenie plánu rozvozu detí na mimoškolské aktivity vo vybraný deň počas týždňa. Sú popísané vybrané typy prepravných trás s dodatočnými obmedzeniami, ktoré sa najčastejšie vyskytujú v spojení s riešením časového rozvrhu voľno časových aktivít. Posledná tretia časť je diskusia.

V každodennom živote sa aj obyčajní ľudia zaoberajú viackriteriálnou otimalizáciou. Často krát si to ani neuvedomujú ale riešia dopravné úlohy s viacerými ohraničeniami. Úloha rozvozu a zvozu či okružné úlohy sú každodenne využívané kuriérskymi a inými spoločnosťami, ktoré na riešenie využívajú sofistikované programy, ktoré im plánujú trasu po ktorej majú ísť. Tieto softvéry však bežnému človeku, ktorý nepracuje v takejto alebo podobnej oblasti nie sú dostupné. Preto človek, ktorý rieši takúto úlohu, musí pred začatím samotnej jazdy stráviť určitý čas plánovaním. Samotné úlohy síce nemajú veľmi veľký rozmer ale ak by sme každý deň venovali len päť minút takémuto plánovaniu za dlhšiu dobu to je dostatočne dlhý čas, ktorý nás vedie o tomu aby sme sa touto problematikou zaoberali.

4.1 Prieskum trhu

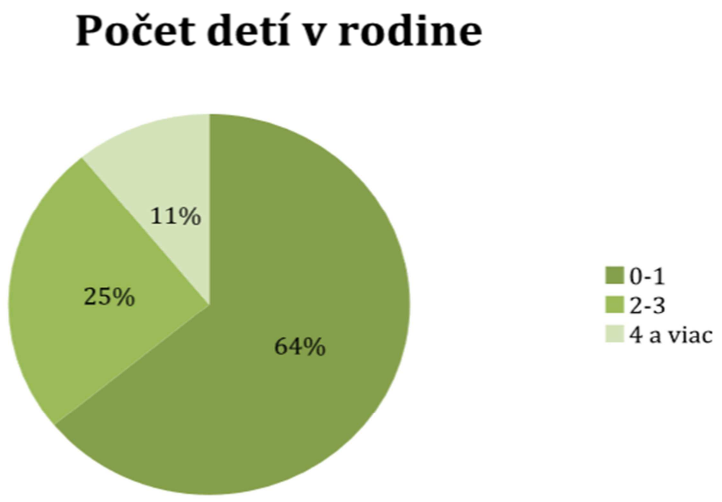
Na začiatku riešenia akéhokoľvek problému je potrebné si definovať, či riešená problematika je aktuálna, či má spoločnosť záujem o nové pohľady na daný problém. Okrem toho je však veľmi dôležité aj zistiť, či sa touto problematikou už niekto zaoberal, prípadne, či daný problém už má vynájdené riešenie, ktoré uspokojuje potreby trhu. Keďže v dnešnej dobe sú aplikácie na podporu rozhodovania a time menežmentu využívané takmer všetkými používateľmi mobilných zariadení je priam nevyhnutné zmapovať už existujúce aplikácie, ktoré sa touto problematikou zaoberajú.

Na trhu existuje nespočetné množstvo aplikácií, ktoré ľuďom pomáhajú s time manažmentom a prepravou. Vo väčšine prípadov sú to však aplikácie, ktoré tieto dva typy úloh riešia samostatne. Jedny riešia time manažment a druhé prepravu. Takzvané time management application však slúžia predovšetkým na zaznamenávanie času, stráveného jednotlivými činnosťami a plánovanie činností. Napríklad aplikácia Instant od spoločnosti Emberify je aplikácia bežiaca na pozadí mobilného zariadenia, ktorá monitoruje počet odomknutí zariadenia počas dňa a čas strávený používaním iných aplikácií nainštalovaných v zariadení prípadne ostatných činností vykonávaných počas dňa na vybranom zariadení. Iný typ aplikácie pre time management je napríklad klasický kalendár, do ktorého si môžeme zaznamenávať jednotlivé úlohy a pripomienky. Takýchto aplikácií je na trhu veľké množstvo. Potom sú tu aplikácie, ktoré synchronizujú viacero iných programov. Jednou z nich je napríklad aplikácia Remember the milk. Táto aplikácia dokáže spojiť programy ako Gmail, Google Calendar, Evernote, Twitter a mnohé iné. Tak ako aj väčšina kalendárov upozorňuje vopred na zadané úlohy a udalosti.

Všetky takéto a podobné aplikácie sa však zameriavajú len na málo kritérií z ktorých prioritnou je čas, kedy udalosť nastane alebo kedy treba danú úlohu vykonať. Neriešia problém nájdenia trasy medzi jednotlivými miestami aktivít. Pre náš problém teda nie sú vhodné lebo nám neposkytujú možnosť výberu viacerých kritérií. Síce je čas aktivity pre nás najdôležitejším faktorom, pretože zmeškanie krúžku či vyučovania by znamenalo nesplnenie úlohy, nie menej dôležitým kritériom resp. obmedzením je kapacita vozidla.

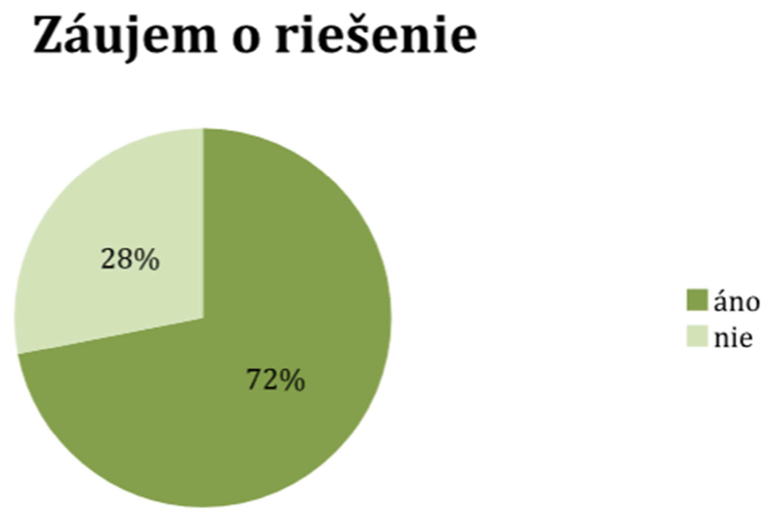
V prvom kroku tejto práce bol vypracovaný dotazník, ktorým sa zistilo, že rodičia detí navštevujúcich prvý stupeň základných škôl by mali záujem vyskúšať mobilnú aplikáciu, ktorá by im pomohla pri každodennom rozhodovaní. Toto zistenie nás teda presvedčilo že touto témou je vhodné sa zaoberať aj napriek skutočnosti ako je nízky počet detí v rodinách, s čím sa môže spájať aj nižší čas potrebný na rozhodovanie. Na nasledujúcich grafoch je zobrazené percentuálne vyhodnotenie vybraných otázok z dotazníka. Prvý graf zobrazuje počet detí v jednotlivých rodinách a druhý graf záujem o riešenie danej problematiky. Dotazník, bol daný rodičom na základných školách v Bratislave. Je súčasťou prílohy číslo 1.

Obrázok 9 - Výsledky dotazníka - otázka číslo 1



ZDROJ: vlastné spracovanie

Obrázok 10 - Výsledky dotazníka - otázka číslo 6



ZDROJ: vlastné spracovanie

4.2 Charakteristika skúmanej oblasti a návrh riešenia

Mária Letecká, manželka a matka 4 detí je majiteľkou účtovnej spoločnosti. Jej manžel pracuje ako právnik. So svojou rodinou žijú v Bratislave. Všetky ich deti sú školopovinné. Tri z nich sú žiakmi prvého stupňa základnej školy a najstarší syn je žiakom siedmeho ročníka základnej školy. Každé z ich detí navštevuje voľnočasové aktivity v škole ale aj mimo nej.

Vzhľadom k tomu, že tri z ich detí sú vo veku kedy ešte necestujú samé mestkou hromadnou dopravou, a rodinný dom v ktorom bývajú sa nachádza približne 15 minút chôdzou od najbližšej zastávky mestkej hromadnej dopravy, musí svoje deti voziť autom. Alternatívou by bolo využitie niektorej zo služieb na rozvoz detí, to však v ich prípade neprichádza do úvahy z dôvodu finančnej náročnosti. Tento fakt spôsobuje, že sa musia v priebehu školského roka počas pracovného týždňa každé ráno zaoberať otázkou, kto odvezie deti ráno do školy, kto ich vyzdvihne zo školy, či ona sama alebo jej manžel, kto z nich odvezie deti na mimoškolské krúžky, kto ich znova vyzdvihne a následne privezie domov.

K dispozícii majú dve autá. Jedno má manžel, druhé manželka. Manželkino vozidlo je 5 miestne, manželovo vozidlo je 7 miestne. V prípade potreby sa teda môže celá šesť členná rodina odviezť v jednom vozidle. Každé z jej detí navštevuje viacero krúžkov počas týždňa.

Čas, ktorý pani Mária a jej manžel trávia plánovaním, či už na týždennej báze, alebo počas jednotlivých dní sa nedá presne určiť. Okrem rozvozu detí musia dohliadať na pracovné úlohy a keďže sa obaja nachádzajú v pozícii vedúcich pracovníkov s nepresne stanovenou pracovnou dobou často krát sa ich pracovné stretnutia a iné povinnosti prekrývajú s časmi, kedy je potrebné odviezť deti. Na základe vykonaného prieskumu však môžeme hovoriť v priemere o 15 minútach denne, ktoré súvisia s dohadovaním, plánovaním, sledovaním hodiniek, a iných povinností, ktoré v ten deň rodičia musia splniť. S určitosťou však môžeme povedať, že aplikácia, ktorá by im umožnila toto každodenné plánovanie vynechať, resp. znížiť ho len na čas, ktorý je potrebný na zadanie vstupných informácií na začiatku školského roka by im ušetrila počas jedného školského roka, viac ako 48,5 hodiny času čo je viac ako 2 dni. Pri tomto výpočte sme vychádzali z informácie, že školských dní počas ktorých deti navštevujú školu, teda nepočítame s prázdninami a voľnými dňami, je 194.

Na ich ulici žijú ešte ďalšie štyri rodiny, ktoré majú školopovinné deti a podobný rozhodovací problém riešia rovnako každé ráno, počas školského roka. Okrem detí, ktoré navštevujú ZŠ majú aj deti, ktoré navštevujú predškolské zariadenia a rovnako musí byť postarané o ich odvoz. Keďže sa rodičia poznajú a majú medzi sebou dobré vzťahy dohodli sa, že si s rozvozom detí do školy a na krúžky pomôžu.

Do samotnej úlohy, ktorá je riešená teda vstupuje 15 uzlov a 13 detí. V prílohe číslo 2 je uvedená matica najkratších vzdialeností medzi adresami, ktoré musia počas týždňa navštevovať podľa rozvrhu.

Na to aby sme mohli okružnú úlohu riešiť pomocou aplikácie, musíme si zvoliť algoritmus, pomocou ktorého budeme hľadať optimálne riešenie. V tejto práci sme sa rozhodli na riešenie úlohy použiť algoritmus Clarka-Wrighta.

Vstupné údaje:

Depom, teda uzlom u_0 je adresa na ktorej bývajú všetci susedia. Je to Rebarborová ulica v Bratislave. Rozvoz detí sa zabezpečuje do 10 inštitúcií, ktoré sú prezentované uzlami u_1 až u_{10} , pričom všetky inštitúcie sa nachádzajú v Bratislave. Sú na nasledovných adresách:

u_0. Rebarborová ulica

u_1. Beňadická ulica

u_2. Gercenova ulica

u_3. Teplická ulica

u_4. Strečnianska ulica

u_5. Pri Suchom mlyne

u_6. Jelačičova ulica

u_7 Železničiarska ulica

u_8. Viedenská cesta ulica

u_9. Sklenárova ulica

u_{10}. Palkovičova ulica

u_{11}. Junácka ulica

Preprava sa bude uskutočňovať pomocou osobných áut, ktoré majú k dispozícii obyvatelia ulice a kapacita vozidla bude pevne stanovená. Počet vozidiel sa bude odvíjať od získaného riešenia.

Ďalej sú uvedené riešenia konkrétneho dopravného problému rozvozu detí na mimoškolské aktivity. Konkrétne je to plán rozvozu detí vo štvrtok.

Pri výpočte konkrétnych riešení bol použitý programovací jazyk Java od spoločnosti Oracle Corporation, ktorý je objektovo orientovaným jazykom použiteľným na všetkých platformách. Konkrétne JAVA SE (Standard Edition), ktorá je typickou inštaláciou Javy pre domáce počítače.

4.3 Okružná úloha bez kapacitného obmedzenia

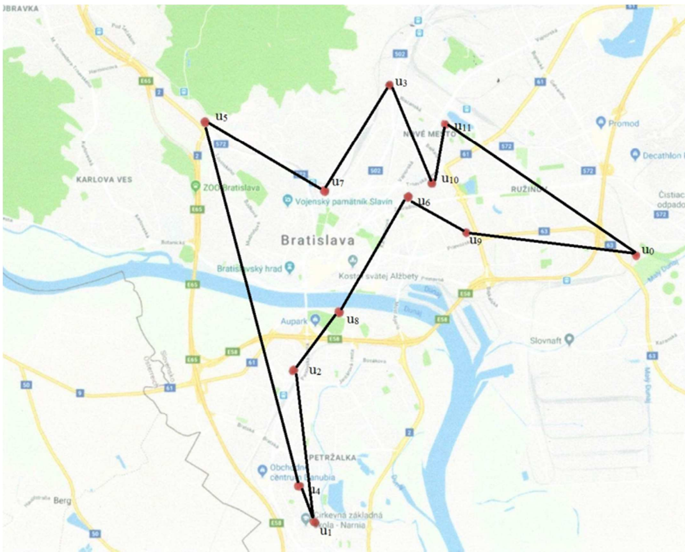
Základným tvarom okružnej úlohy bez akýchkoľvek obmedzení je také riešenie úlohy, v ktorom treba navštíviť všetky uzly v sieti pri minimalizácii času a dodržaní požiadaviek zákazníkov. Kapacita vozidla je neobmedzená a tak sa všetky deti môžu viesť spolu. Takéto riešenie však nie je postačujúce a preto danú úlohu v ďalších častiach doplníme o dodatočné ohraničenia.

Zákazníkov respektíve adresy, ktorých potrebujeme obslúžiť sú: Rebarborová ulica, kde všetky rodiny bývajú, ktorá zároveň v tomto prípade bude prezentovať depo, Beňadická ulica – základná škola, Gercenova – základná škola, Teplická – základná škola, Strečnianska – základná umelecká škola, Pri Suchom mlyne – športové centrum, Jelačičova – základná škola, Železničiarska – špeciálne pracovisko pre riešenie porúch reči, Viedenská cesta – jazyková škola, Sklenárova - telocvičňa, Palkovičova – materská škola, Junácka – plaváreň.

Ak by sme chceli všetky tieto miesta navštíviť, pričom by sme trasu absolvovali mimo dopravnej špičky a riadili by sme sa pomocou navigácie od spoločnosti Google potrebovali by sme na jej prejdenie 1 hodinu. Celková trasa by merala 50 km. Toto riešenie však bolo získané len náhodným plánovaním trasy a teda ho nemôžeme považovať za optimálne. Riešená úloha nezohľadňuje žiadne dodatočné ohraničenia ako kapacita, časové okná alebo iné. V prípade, že by sme túto trasu chceli prejsť v pracovný deň po 15-tej hodine čas potrebný na jej prejdenie by sa zvýšil z pôvodnej 1 hodiny na 1 hodinu a 26 minút. Na základe tohto zistenia, teda budeme pracovať v ďalších úlohách s časom, ktorý bude zohľadňovať bežnú dopravnú situáciu v meste počas pracovného dňa. Priemerná rýchlosť vozidla bude 34,7 km za hodinu.

Na uskutočnenie takejto okružnej trasy by však obyvatelia ulice potrebovali dopravný prostriedok, ktorý by mohol odviezť všetky deti. Takýto prostriedok by bol neefektívny z dôvodu, že by nemohli byť akceptované dodatočné ohraničenia časových okien alebo viacerých dep.

Obrázok 11- Okružná trasa bez obmedzení



ZDROJ: vlastné spracovanie

Úloha bez dodatočných obmedzení riešená pomocou algoritmu Clarka-Wrighta pri daných požiadavkách uzlov bude určovať trasu v nasledovnom poradí: $u_0 - u_9$- $u_6 - u_8 - u_2 - u_1 - u_4 - u_5 - u_7 - u_3 - u_{10} - u_{11} - u_0$. Riešenie je zobrazené na obrázku číslo 11 vyššie. Východiskovým uzlom je Rebarborová ulica a trasa pokračuje nasledovne: Sklenárova – Jelačičova - Viedenská cesta – Gercenova - Beňadická – Strečnianska – Pri Suchom mlyne – Železničiarska – Teplická – Palkovičova – Junácka - Rebarborová. Celková prejdená vzdialenosť predstavuje 48 km. Čas potrebný na prejdenie tejto trasy by bol pri bežnej premávke 1 hodina a 22 minút.

4.4 Riešenie úlohy VRP s kapacitným obmedzením

Kapacitné ohraničenie VRP patrí medzi najjednoduchšie modifikácie okružných úloh. V tejto úlohe teda budeme hľadať optimálne riešenie pre rozvoz detí také, aby nebola prekročená kapacita vozidla.

Budeme predpokladať, že rodiny majú klasické päť miestne automobily ale jedno miesto bude je pridelené šoférovi, takže kapacita vozidla bude 4 osoby.

Pre riešenie problému rozvozu detí do školy a na krúžky budeme pracovať aj s nasledujúcimi predpokladmi:

- k dispozícii je jeden sklad – v tomto prípade jedna adresa, z ktorej sa budú deti vyzdvihovať – miesto ich bydliska Rebarborová ulica,

- počet vozidiel sa bude odvíjať od získaného riešenia, kapacita je vopred známa a u všetkých vozidiel rovnaká,

- budeme predpokladať, že rýchlosť vozidla je konštantná a teda vozidlo ide rovnakou rýchlosťou,

- kapacita vozidiel nemôže byť prekročená,

- úlohou je minimalizovať počet vozidiel, potrebných a obsluhu všetkých zákazníkov – navštívenie všetkých adries, tak aby boli dodržané ohraničenia a zároveň minimalizovaný čas a vzdialenosť potrebná na rozvoz.

Výsledkom úlohy sú 4 okružné jazdy:

- Rebarborová – Viedenská cesta – Beňadická – Strečnianska – Pri Suchom mlyne – Železničiarska – Junácka - Rebarborová; celková dĺžka trasy je 38 km a čas potrebný na prejdenie trasy predstavuje 1 hodinu a 6 minút.
- Rebarborová – Gercenova – Jelačičova - Rebarborová; celková dĺžka trasy je 23km, čo činí 40 minút.
- Rebarborová – Sklenárova – Palkovičova – Rebarborová; celková dĺžka trasy je 14 km čo činí 24 minút.
- Rebarborová – Teplická - Rebarborová; celková dĺžka trasy je 16 km, čo činí 28 minút.

Zo získaného riešenia vyplýva, že na to aby rodičia mohli rozviezť svoje deti budú potrebovať 4 vozidlá s kapacitou 5 osôb, pričom jedno sedadlo je určené pre šoféra. V prípade ak by mali k dispozícii vozidlá, ktoré majú 7 miest riešenie by sa zmenilo nasledovne:

- Rebarborová – Viedenská cesta – Gercenova – Beňadická – Strečnianska – Pri Suchom mlyne – Železničiarska – Junácka - Rebarborová; celková dĺžka trasy je 39 km, čo činí 1 hodinu a 7 minút.

- Rebarborová – Teplická – Palkovičova – Rebarborová; celková dĺžka trasy je 17 km, čo činí 29 minút.

- Rebarborová – Jelačičova - Sklenárova – Rebarborová; celková dĺžka trasy je 14 km, čo činí 24 minút.

Ak porovnáme získané riešenia môžeme povedať, že obyvateľom na ulici Rebarborová by sme odporučili zaobstaranie vozidiel s väčšou kapacitou, ktoré sú dnes bežne dostupné. Tak by sa znížil počet šoférov, keďže sa znížil aj počet trás. Ostatní rodičia by mohli ušetrený čas využiť v prospech svojich iných záujmov. Ušetrený čas v súčte predstavuje počas jedného pracovného dňa 38 minút. V konečnom dôsledku by takéto rozhodnutie malo pozitívny vplyv aj na dopravnú situáciu v meste a životné prostredie.

Výpočet riešenia prebiehal pomocou vyššie spomínaného Algoritmu Clarka-Wrighta. Pomocou programovacieho jazyka Java, sme zapísali kroky algoritmu, ktorý následne spracoval zadané vstupné údaje. Výstupom tejto operácie bola matica vzdialeností medzi jednotlivými uzlami, matica úspor, utriedený zoznam úspor v tvare [(uzol.uzol) - úspora] a zoznam nájdených ciest. Takýmto spôsobom sme postupovali aj pri riešení úlohy bez kapacitného ohraničenia. Pri zadávaní vstupných hodnoť kapacita mala hodnotu väčšiu ako počet detí, a parameter kapacita vozidla nijako neovplyvňoval hľadanie riešenia.

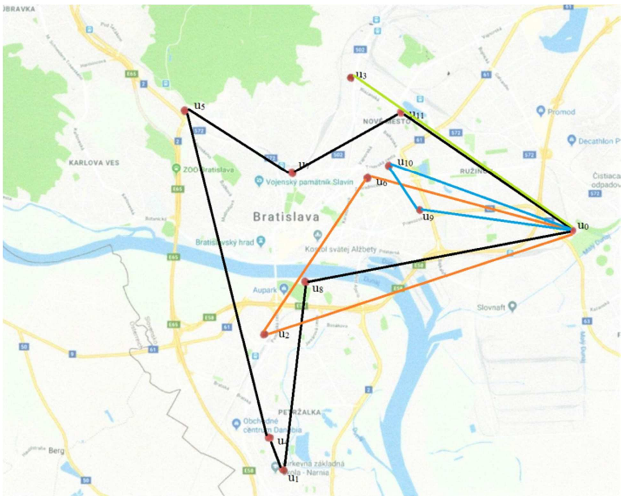
Obrázok 12- Výsledné riešenie okružnej úlohy s kapacitným ohraničením

```
Spare Matrix - Matica uspor


20
11,10
25,19,10
15,14,13,14
11,11,10,11,12
14,13,13,13,15,12
17,18,10,16,13,10,13
9,8,9,8,8,8,9,6
11,9,11,9,11,10,11,7,8
9,8,11,8,11,9,11,7,8,10

List of circles ordered by spare value
[[(4,1) - 25], [(2,1) - 20], [(4,2) - 19], [(8,2) - 18], [(8,1) - 17], [(8,4) - 16], [(5,1) - 15], [(7,5) -
15], [(5,2) - 14], [(5,4) - 14], [(7,1) - 14], [(5,3) - 13], [(7,2) - 13], [(7,3) - 13], [(7,4) - 13],
[(8,5) - 13], [(8,7) - 13], [(6,5) - 12], [(7,6) - 12], [(3,1) - 11], [(6,1) - 11], [(6,2) - 11], [(6,4) -
11], [(10,1) - 11], [(10,3) - 11], [(10,5) - 11], [(10,7) - 11], [(11,3) - 11], [(11,5) - 11], [(11,7) -
11], [(3,2) - 10], [(4,3) - 10], [(6,3) - 10], [(8,3) - 10], [(8,6) - 10], [(10,6) - 10], [(11,10) - 10],
[(9,1) - 9], [(9,3) - 9], [(9,7) - 9], [(10,2) - 9], [(10,4) - 9], [(11,1) - 9], [(11,6) - 9], [(9,2) - 8],
[(9,4) - 8], [(9,5) - 8], [(9,6) - 8], [(10,9) - 8], [(11,2) - 8], [(11,4) - 8], [(11,9) - 8], [(10,8) - 7],
[(11,8) - 7], [(9,8) - 6]]
********** RESULTS **********
Kazdy okruh zacina a konci nultym vrcholom.
Circle 1:
0 8 2 1 4 5 7 11 0
Circle 2:
0 6 3 10 0
Circle 3:
0 9 0
```

ZDROJ: vlastné spracovanie

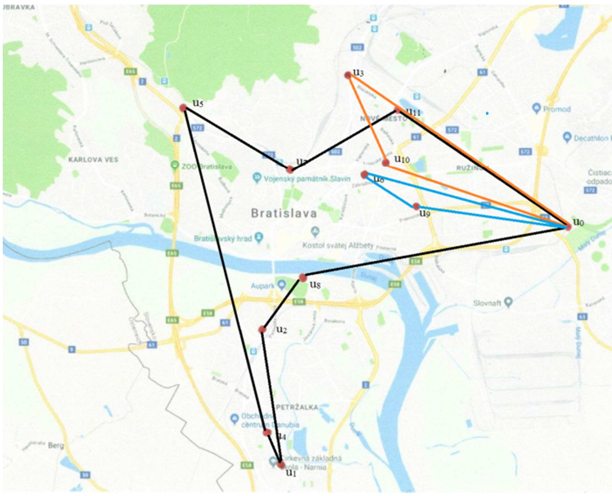
Dĺžku trasy sme potom vypočítali pomocou matice vzdialeností, sčítaním vzdialeností medzi uzlami v nájdenej trase. Po prepočte, priemernou rýchlosťou vozidla sme dostali čas potrebný na prejdenie trasy.

Obrázok 13- Trasy vozidla s kapacitou 4 osoby



ZDROJ: vlastné spracovanie

Obrázok 14- Trasy vozidla s kapacitou vozidla 6 osôb



ZDROJ: vlastné spracovanie

4.5 Riešenie úlohy VRP s časovými oknami

Princíp zavedenia časových okien do okružnej úlohy spočíva v určení presne vymedzených časov kedy môže byť daný zákazník obslúžený. Okrem vyššie uvedených premenných teda do modelu vstupujú nové premenné, ktoré súvisia práve s týmito časovými oknami. Každé časové okno má svoj začiatok a koniec. Práve začiatok a koniec sa teda stávajú premennými vstupujúcimi od modelu. Začiatok časového okna budeme označovať e_i a koniec časového okna l_i. Ďalšou premennou je *bi*, ktorá prezentuje skutočný začiatok obsluhy zákazníka *i*. Premenná o_i reprezentuje dĺžku času obsluhy zákazníka *i*.

Tak ako v prípade všeobecného modelu iba s kapacitným obmedzením aj v tomto prípade musíme skontrolovať prípustnosť kapacity vozidla. Vzhľadom na doplnenie kritéria s časovými oknami sa táto kontrola dopĺňa o kontrolu prípustnosti časových okien.

Riešená úloha bude mať nasledujúce predpoklady:

- štartovací uzol je len jeden a nemá určené časové okno, obsluha sa začína pred navštívením prvého uzla,
- matica najkratších vzdialeností je v tomto prípade prepočítaná z kilometrov na čas. Priemerná rýchlosť vozidla je 34,7 km,
- všetky časové okná sú určené vopred vzhľadom na aktivity, ktoré na daných adresách prebiehajú,
- ku všetkým uzlom sú priradené časové okná,

- časy obsluhy jednotlivých uzlov sú stanovené podľa dĺžky aktivity, ku každému začiatku je pripočítaných 5 minút rezerva
- je presne určená kapacita vozidiel pričom všetky vozidlá majú rovnakú kapacitu, a kapacita nemôže byť prekročená.

Pri riešení úloh s časovými oknami sme postupovali podobne ako v prípade úloh s kapacitným ohraničením. Algoritmus, sme upravil tak aby zohľadňoval časové okná. Výstup v tomto prípade vyzeral rovnako, avšak pred samotným výpočtom sme upravili maticu aj vzdialeností. V predchádzajúcom prípade jej hodnoty prezentovali počet prejdených kilometrov. Pre úlohu s časovými oknami sme tieto hodnoty prepočítali na čas, ktorý je potrebný na prejdenie trasy. Po tejto úprave majú všetky parametre rovnaký význam a teda môžeme začať s výpočtom.

Získané výsledné trasy nám poskytujú len informáciu o roztriedení uzlov do trás a ich poradí. Celkovú vzdialenosť sme museli vypočítať nasledovným spôsobom. Čistý čas potrebný na prejdenie trasy sa rovná súčtu všetkých vzdialeností medzi uzlami v trase. Keďže výsledné údaje sú teraz v minútach musíme hodnotu spätne previesť na kilometre. Depo v tomto prípade nemá stanovené časové okno. Pri určovaní času, kedy vozidlo musí vyštartovať teda vychádzame z údaju o otvorení časového okna, ktoré je obsluhované ako prvé a údaju o dĺžke trasy potrebnej na prejdenie z depa do tohto uzla v minútach. Podobným spôsobom vypočítame aj návrat vozidla do depa a celkový čas strávený mimo depa.

Riešenie okružnej úlohy s časovými oknami sa skladá z nasledovných trás:

Tabuľka 6: Okružná cesta číslo 1: $u_0 - u_2 - u_7 - u_6 - u_8 - u_{14} - u_{13} - u_0$

Adresa	Požiadavky	Dĺžka trasy v minútach	Začiatok	Koniec	Doba obsluhy
Rebarborová	0	-	-	-	0
Gercenova	1	19	7:30	8:00	15
Železničiarska	2	10	13:30	14:00	30
Jelačičova	1	3	14:30	15:30	60
Viedenská cesta	1	9	16:00	17:00	60
Jedenásta	1	12	7:30	19:00	60
Spojná	1	7	7:30	19:00	60
Rebarborová	0	14	-	-	0

ZDROJ: vlastné spracovanie

Celková prejdená vzdialenosť na tejto trase predstavuje 42,7 km. Vozidlo musí vyštartovať o 7 hodine a 6 minúte a vráti sa do o 19 hodine a 14 minúte.

Tabuľka 7: Okružná cesta číslo 2: uzly $u_0 - u_{10} - u_0$

Adresa	Požiadavky	Dĺžka trasy v minútach	Začiatok časového okna	Koniec časového okna	Doba obsluhy
Rebarborová	0	-	-	-	0
Palkovičova	2	10	7:30	8:00	15
Rebarborová	0	10	-	-	0

ZDROJ: vlastné spracovanie

Táto trasa pozostáva len z obsluhy jedného uzla. Prejdená vzdialenosť predstavuje dĺžku 11,5 km s čistým časom prepravy 20 minút. Vozidlo môže vyštartovať o 9 minút neskôr ako vozidlo, ktoré obsluhuje prvú trasu.

Jeho návrat však je skorší a to o 8 hodine a 4 minúte. Vzhľadom k tomu, že všetky ostatné trasy začínajú v ranných hodinách toto vozidlo už nie je viac použité.

Tabuľka 8: Okružná cesta číslo 3: $u_0 - u_3 - u_9 - u_{11} - u_0$

Adresa	Požiadavky	Dĺžka trasy v minútach	Začiatok časového okna	Koniec časového okna	Doba obsluhy
Rebarborová	0	-	-	-	0
Teplická	2	14	7:30	8:00	15
Sklenárova	1	7	15:30	16:00	90
Junácka	1	5	16:30	17:15	45
Rebarborová	0	10	-	-	0

ZDROJ: vlastné spracovanie

Okružná cesta číslo 3 sa skladá z obsluhy troch uzlov. Rovnako ako prechádzajúce okružné jazdy začína obsluhou uzla, ktorého časové okno začína o 7:30. Štart tohto vozidla však nie je rovnaký. Vozidlo musí vyraziť o 7 hodine a 11 minúte. Návrat vozidla je o 17:25. Dĺžka trasy je 20,8 km a celkový čas, ktorý vozidlo strávi mimo depa je 10 hodín a 14 minút.

Tabuľka 9: Okružná cesta číslo 4: $u_0 - u_1 - u_{12} - u_{15} - u_5 - u_4 - u_0$

Adresa	Požiadavky	Dĺžka trasy v minútach	Začiatok časového okna	Koniec časového okna	Doba obsluhy
Rebarborová	0	-	-	-	0
Beňadická	3	24	7:30	8:00	15
Kutlíkova	1	9	7:30	19:00	60
Na Riviére	1	15	7:30	19:00	60
Pri Suchom mlyne	1	7	16:00	17:00	60
Strečnianska	1	16	17:30	18:15	45
Rebarborová	0	21	-	-	0

ZDROJ: vlastné spracovanie

Posledná okružná cesta obsluhuje 5 uzlov. Táto sa začína najskôr zo všetkých okružných jázd o 7 hodine a 1 minúte. Celková trasa meria 53 km. Vozidlo sa vráti do depa o 18 hodine a 36 minúte.

Kapacita vozidla bola vo všetkých okružných jazdách dodržaná a jej hodnota bola alebo 7 osôb. Rovnako boli dodržané aj časové okná, v ktorých prebieha obsluha. Celková prejdená vzdialenosť predstavuje 128 km, celkový čas strávený na cestách predstavuje 34 hodín a 46 minút.

4.6 Riešenie úlohy multi pedot

V prípade, že by sme úlohu pozmenili a pracovali s predpokladmi, že rodičia detí pracujú a majú svoje aktivity mimo adresy bydliska, a to na rôznych adresách v meste musíme okružnú úlohu riešiť ako multi depot úlohu, teda úlohu s viacerými depami. Keďže rodičia sa prispôsobujú deťom a ich aktivitám, musia sa prispôsobiť aj trasy a tak do úlohy vstupujú viaceré uzly označujúce sa ako depo. V poobedných hodinách rodičia končia v práci, prípadne končia ich aktivity a tak po deti nepôjdu z Rebarborovej ulice ale z rôznych adries v meste. Táto skutočnosť spôsobí, že depo nebude len jedno ale môže ich byť viac. Pomocou algoritmu Clarka-Wrighta sa však úlohy s viacerými depami riešiť nedajú. Tento algoritmus je zameraný len na riešenie úloh s jedným depom. V takomto prípade teda ešte pred začatím hľadania optimálnej trasy musia byť jednotlivé uzly v sieti rozdelené, podľa počtu východzích uzlov.

Do úlohy pribudnú nasledujúce uzly:

u_{12} Kutlíkova ulica

u_{13} Spojná ulica

u_{14} Jedenásta ulica

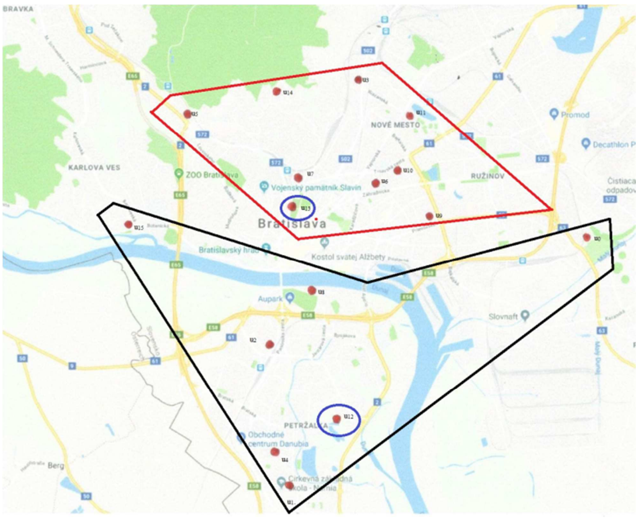
u_{15} Na Riviére

Na rozdelenie uzlov sa používajú rôzne techniky. V tomto prípade sme zvolili techniku K-means clustering a pomocou nej sme množinu všetkých uzlov, ktorých je 15 vrátane východzích uzlov rozdelili na dve skupiny. Prvým depom je uzol číslo 12 na Kutlíkovej ulici a druhým depom je uzol číslo 13 na Spojnej ulici. Klastrovaním sme rozdelili uzly siete nasledovne:

klaster 1: $u_0 - u_1 - u_2 - u_4 - u_8 - u_{12} - u_{15}$

klaster 2: $u_3 - u_5 - u_6 - u_7 - u_9 - u_{10} - u_{11} - u_{13} - u_{14}$

Obrázok 15- Rozdelenie uzlov na skupiny podľa blízkosti k depu



ZDROJ: vlastné spracovanie

Východiskovým uzlom v prvom klastri je Kutlíkova ulica a budeme ju označovať u_0, ostatné uzly budú mať označenie u_1 až u_6. Analogicky zmeníme aj označenia uzlov v druhom klastri.

u_0. Kutlíkova ulica	u_0. Spojná ulica
u_1. Rebarborová ulica	u_1. Teplická ulica
u_2. Beňadická ulica	u_2. Pri Suchom mlyne
u_3. Gercenova ulica	u_3. Jelačičova ulica
u_4. Strečnianska ulica	u_4. Železničiarska ulica
u_5. Viedenská cesta	u_5. Sklenárova ulica
u_6. Na Riviére	u_6. Palkovičova ulica
	u_7. Junácka ulica
	u_8. Jedenásta ulica

Matica všetkých vzdialeností teda bola rozdelená na dve matice, podľa klastrov nasledovne:

Tabuľka 4: Adresy v meste – klaster 1

D12	0	1	2	3	4	5	6
0	0	9	5	4	2	5	9
1	9	0	14	11	12	9	14
2	5	14	0	5	1	6	9
3	4	11	5	0	4	8	6
4	2	12	1	4	0	5	8
5	5	9	6	8	5	0	5
6	9	14	9	6	8	5	0

ZDROJ: vlastné spracovanie

Tabuľka 5: Adresy v meste – klaster 2

D13	0	1	2	3	4	5	6	7	8
0	0	4	4	3	2	4	3	4	4
1	4	0	6	4	3	4	3	3	4
2	4	6	0	5	4	8	6	6	3
3	3	4	5	0	2	3	2	3	5
4	2	3	4	2	0	5	3	3	3
5	4	4	8	3	5	0	3	3	7
6	3	3	6	2	3	3	0	2	5
7	4	3	6	3	3	3	2	0	5
8	4	4	3	5	3	7	5	5	0

ZDROJ: vlastné spracovanie

Po rozdelení všetkých uzlov v sieti môžeme pokračovať v riešení okružnej úlohy pomocou algoritmu Clarka-Wrighta. Samotný výpočet trasy prebieha rovnako ako v prípade úlohy s obmedzením kapacity.

Okružná úloha pre klaster 1 s kapacitou vozidla 4 osoby má nasledovné riešenie:

- Kutlíkova - Rebarborová - Viedenská cesta - Na Riviére - Gercenova - Strečnianska - Kutlíkova; celková dĺžka trasy je 35 km, čo činí 1 hodinu.

- Kutlíkova – Beňadická – Kutlíkova; celková dĺžka trasy je 10 km, čo činí 17 minút.

Okružná úloha pre klaster 1 s kapacitou vozidla 6 osôb má nasledovné riešenie:

- Kutlíkova - Rebarborová - Viedenská cesta - Na Riviére - Gercenova - Beňadická - Strečnianska - Kutlíkova; celková dĺžka trasy je 37 km, čo činí 1 hodinu a 4 minúty.

Okružná úloha pre klaster 2 s kapacitou vozidla 4 osoby má nasledovné riešenie:

- Spojná - Palkovičova - Teplická - Junácka - Sklenárova - Jelačičova - Jedenásta - Spojná; celková dĺžka trasy je 24 km, čo činí 41 minút.

- Spojná - Pri Suchom mlyne - Spojná; celková dĺžka trasy je 8 km, čo činí 14 minút.

- Spojná - Železničiarska - Spojná; celková dĺžka trasy je 4 km, čo činí 7 minúty.

Okružná úloha pre klaster 2 s kapacitou vozidla 6 osôb má nasledovné riešenie:

- Spojná - Pri Suchom mlyne - Spojná; celková dĺžka trasy je 24 km, čo činí čo činí 41 minút.

- Spojná - Železničiarska - Spojná; celková dĺžka trasy je 8 km, čo činí čo činí 7 minút.

Po porovnaní získaných riešení vidíme, že v oboch prípadoch by sa rodinám oplatilo vlastniť vozidlá s väčšou kapacitou. Tak ako nám použitie vybraného algoritmu nájde okružné trasy a tým ušetrí prejdené kilometre aj použitie väčšieho vozidla má za následok úsporu. V okružnej úlohe pre 1 klaster by sme použitím väčšieho vozidla ušetrili 8 km a 13 minút. V úlohe pre 2 klaster úspora predstavuje 4 kilometre a 7 minút.

4.7 Diskusia

Na základe získaných riešení môžeme povedať, že použitie algoritmu Clarka-Wrighta pri riešení okružných úloh s dodatočnými ohraničeniami nám dokáže poskytnúť akceptovateľné riešenie. Vzhľadom ku skutočnosti, že vozidlá využívané na osobnú prepravu majú obmedzenú kapacitu, v reálnom svete sa najčastejšie stretávame s 5 miestnymi vozidlami, je toto ohraničenie smerodajným.

Vo firmách riešiacich rovnaké alebo podobné úlohy sú často krát vozidlá s rôznymi veľkosťami a podľa množstva prepravovaného tovaru sa výber vozidla prispôsobuje.

V kapitole 4.4 sme pomocou vybraného algoritmu počítali okružné úlohy s rôznymi kapacitnými ohraničeniami. Zistili sme, že ak by rodiny mali vozidlá aj s väčšou kapacitou, konkrétne 7 osôb, celkový čas potrebný na prejdenie trasy by sa znížil o 23,07% a celková prejdená trasa by sa znížila o 24,05%.

Podobné výsledky sme získali aj pri riešení úlohy s viacerými depami v časti 4.6. Pri použití vozidla s väčšou kapacitou by sme celkovú vzdialenosť znížili o 14,84% a celkový čas o 18,24%. V tomto prípade sme okrem algoritmu Clarka-Wrighta použili na získanie riešenia aj algoritmus K-means, z dôvodu, že algoritmus Clarka-Wrighta takéto úlohy nevie riešiť. V ďalšej práci by sme preto odporučili na riešenie použiť iný algoritmus, ktorý by okružnú úlohu s viacerými depami vedel riešiť.

Ďalšou modifikáciou algoritmu bolo pridanie časových okien. Všetky časové okná sú definované ako tvrdé časové okná. Ak by to tak nebolo, meškanie by spôsobilo nestihnutie danej aktivity. Keďže sa v našom prípade nejedná o tovar, nemôžeme počítať s mäkkými časovými oknami. V prípade distribúcie tovaru alebo materiálu sa predpokladá, že na sklade majú ešte určité množstvá a tak môžu byť použité aj mäkké časové okná. V okružných úlohách riešených v časti 5.5 sme zistili, že ak chceme dodržať všetky časové okná a kapacitné ohraničenie, musíme sa vydať na 4 okružné jazdy. Ak ich porovnáme s výsledkami z časti 5.4 môžeme vidieť, že napriek použitiu vozidla s väčšou kapacitou musíme vykonať rovnaký počet jázd ako v prípade len kapacitného ohraničenia so stanovenou kapacitou 5 osôb.

Po porovnaní všetkých výsledkov, môžeme povedať, že okružné úlohy a ich modifikácie majú miesto aj v prostredí domácností. Algoritmus, ktorý sme použili nám poskytol rýchle s prijateľné riešenie, avšak pre bežného človeka je tento problém stále zložitým.

Záver

Rozhodovanie v oblasti riadenia prepravy je problém, s ktorým sa stretáva väčšina firiem aj obyvateľstva. O význame možnosti získať na podporu rozhodovania, riadenia a optimalizácie prepravy potrebný systém, svedčí aj veľkosť tohto odvetia a pozornosť, ktorá sa venuje optimalizácii prepravných trás.

Táto diplomová práca sa zameriavala na riešenie okružných úloh pomocou algoritmu Clarka- Wrighta a jeho modifikácií. Ako sme si mohli všimnúť riešenie úloh s časovými oknami sa dá dosiahnuť pomocou modifikácie tohto algoritmu. Ak však riešime úlohu s viacerými depami, tento algoritmus už nie je najvhodnejší, z dôvodu, že pred jeho využitím úlohu musíme upraviť. Tento krok by pri reálnom využití mohol celý proces hľadania riešenia spomaliť. Preto by sme na riešenie úloh s viacerými depami odporučili otestovať iný algoritmus, ktorý by poskytoval prijateľné riešenie.

Hlavným prínosom tejto diplomovej práce je popis a objasnenie témy okružných úloh a ich modifikácií. Priblíženie a definovanie algoritmu Clarka-Wrigha a jeho prezentácia na konkrétnom príklade z reálneho života. Takto prezentované okružné úlohy by boli ľahko zrozumiteľné aj neodbornej verejnosti. Tá by mohla rýchlo pochopiť výhody plynúce z viackriteriálnej optimalizácie prepravných trás, ich plánovnia a metód na ich riešenie. Spolu s dopytom po vhodnom programe, ktorý by riešil okružné úlohy s dodatočnými obmedzeniami by tak vznikla aj pracovná príležitosť pre vývojárov a iných členov odbornej verejnosti. Ďalším prínosom tejto práce je upozornenie na oblasť osobnej prepravy, ktorej sa nevenuje až taká veľká pozornosť. V osobnej preprave sa vo väčšine prípadov využíva na plánovanie trasy navigácia, ktorá však nezohľadňuje dodatočné ohraničenia ako kapacita, časové okná a iné.

Vzhľadom na aktuálne trendy v spoločnosti môžeme povedať, že riešenie okružných úloh na úrovni domácností je priam než žiaduce. Zo získaných výsledkov vyplýva, že úsporou času stráveného na cestách klesá aj počet automobilov čo prispieva k zlepšeniu dopravnej situácie v meste. Rovnako klesá aj počet emisií, ktoré sú dnes v alarmujúcich číslach.

Zoznam použitej literatúry

[1]EUROEKONÓN. *Logistika*. 2018. [online]. Dostupné na: https://www.euroekonom.sk/obchod/logistika

[2] CIGÁNEKOVÁ, M. 2017. *Štíhla logistika.* [online]. [cit. 4.3.2019]. Dostupné na: https://www.ipaslovakia.sk/sk/ipa-slovnik/stihla-logistika

[3] EUROEKONÓN.SK. *Metódy vhodné pre logistiku*. 2008. [online]. Dostupné na: https://www.euroekonom.sk/obchod/logistika/metody-vhodne-pre-logistiku/

[4] JOUBERT, J.W. 2007. *The Vehicle Routing Problem: origins and variants*. 2007. Dostupné na: https://repository.up.ac.za/bitstream/handle/2263 /26439/02chapter2.pdf?sequence=3&isAllowed=y

[5] TSP. *History of the TSP*. 2007. [online]. Dostupné na: http://www.math.uwaterloo.ca/tsp/history/index.html

[6] THE VRP WEB. *Solutions techniqies for VRP*. 2006. [online]. Dostupné na: http://www.bernabe.dorronsoro.es/vrp/

[7] LAPORTE, G. – TOTH, P. – VIGO, D. 2013. *Vehicle Routing: historical perspective and recent contributions*. Dostupné na: https://www.researchgate.net/publication/257806102_Vehicle_routing_Historical_perspective_and_recent_contributions

[8] TORO, E. – ESCOBAR, A. – GRANADA, M. 2016. [online]. *Literature reviev on the vehicle routing problem in the green transportation context.* ISSN 1909-2474. Dostupné na: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1909-24742016000100021

[9] MAŃDZIUK, J. 2018. *New Shades of the Vehicle Routing problem: Emerging Problem Formulations and Compotational Intelligence Solution Methods.* [online]. [cit. 15.4.2019]. Dostupné na: https://www.researchgate.net/publication/329960346_New_Shades_of_the_Vehicle_Routing_Problem_Emerging_Problem_Formulations_and_Computational_Intelligence_Solution_Methods

[10] THE VRP WEB. *Vehicle routing Problemś Formulation*. 2006. [online]. Dostupné na: http://www.bernabe.dorronsoro.es/vrp/

[11] GAJPAL, Yuvraj. 2008. *Algorithm for vehicle routinf problem with pickup and delivery*. [rigorózna práca]. Ontario: McMaster University. 2008. 224 s. Dostupné na: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.879.8176&rep=rep1&type=pdf

[12] BRUCE, L. 1975. *Vehicle routing problems: Formulations and heuristic solution techniques*. [technický report]. Massachusetts: Institute od technology. 1975. 28s. Dostupné na: https://apps.dtic.mil/dtic/tr/fulltext/u2/a013639.pdf

[13] BREZINA, I. – PEKÁR, J. 2014. *Operačná analýza v podnikovej praxi.* Bratislava: Vydavateľstvo EKONÓM, 2014. 282 s. ISBN 978-80-225-4012-4

[14] ARCHETTI, C. – CAMPBELL, A. – SPARANZA, M. 2016. *Multicommodity vs. Single-Commodity Routing. Transportation Science.* 50. 461-472s. Dostupné na: https://pdfs.semanticscholar.org/6782/2b82af099c47cad9aa8fc6d3fa8ad138e448.pdf

[15] CATTARUZZA, D. – ABSI, N. – FEILLET, D. 2016. *Vehicle routing problems with multiple trips*. 4OR: A Quarterly Journal of Operations Research, Springer Verlag, 2016, 14 (3), pp.223-259. ffemse01250603f. Dostupné na: https://hal-emse.ccsd.cnrs.fr/emse-01250603/document

[16] HRABIK-CHOVANOVÁ, H. – SAKÁL, P. 2011. *Operačná analýza časť I.* Trnava: AlimniPress, 2011. 242 s. ISBN 978-80-8096-151-0.

[17] EHRGOTT, M. 2005. *Multicriteria Optimization. Second edition.* Auckland: Springer Berlin Heidelberg New York, 2005. 323 s. ISBN 3-540-21398-8. Dostupné na: http://www.math.hcmus.edu.vn/~nvthuy/om/Multicriteria%20Optimization.pdf/bBEZOD KAZU

[18] ŠKERLÍKOVÁ Zuzana. 2012. *Okružné úlohy s dodatočnými ohraničeniami* [dizertačná práca]. [cit. 4.3.2019]. Bratislava: Ekonomická univerzita v Bratislave. 2012. 103 s.

[19] BREZIN, I. *Kvantitatívne metódy v logistike.* [cit. 7.3.2019]. Prvé vydanie. Ekonóm, 2003. 294 s. ISBN: 80-225-1735-6.

[20] SINGANAMALA, P. – RADDY, K. – VENKATARAMAIAH, P. 2018. *Solution to a Multi Depot Vehicle Routing Problem Using K-means Algorithm, Clarke and Wright Algorithm and Ant Colony Optimization.* In: International Journal of Applied Engeneering

Research. 2018. ISSN: 0973-4562. Dostupné na: https://www.ripublication.com/ ijaer18/ijaerv13n21_58.pdf

[21] KRIŠKA, Jozef. *Ústav informatiky a softwérového inžinierstva STU*, Ilkovičova 3, Bratislava. *Hierarchická a nehierarchická zhluková analýza.* [cit. 18.4.2019]. Dostupné na: http://www2.fiit.stuba.sk/~kapustik/ZS/Clanky0607/kriska/index.html

[22] LI, XIAOYAN. 2015. *Capacitated Vehicle Routing Problem with Time Windows.* [diplomová práca]. Arizona: Arizona state University, 2015. 79s. Dostupné na: https://pdfs.semanticscholar.org/acc2/785529ecd91c12387d8b8e08567c1ce9ab50.pdf

[23] RAND, R. 2009. *The life and rimes of the Savings Method for Vehicle Routing Problems.* In: RN Orion. 2009. ISSN 0529-191-X. Dostupné na: http://citeseerx.ist.psu.edu /viewdoc/download?doi=10.1.1.885.2969&rep=rep1&type=pdf

[24] TOTH, P. – VIGO, D. 2014. *Vehicel ruting Problems, Methods, and Applications.* 2. vyd. Philadelphia. Applied Methematics and the Mathematival Optimization Society. 2014. 463 s. ISBN 978-1-611973-58-7. Dostupné na: https://books.google.sk/books? id= VUzUBQAAQBAJ&pg=PA162&lpg=PA162&dq=PDVRP+model&source=bl&ots=MY_ Aspq_B1&sig=ACfU3U2L9aPr1nbBbS3JrBF01ZcKTo7Y6g&hl=sk&sa=X&ved=2ahUK EwjSupOWyrnhAhUSLVAKHaRnBuoQ6AEwAXoECAkQAQ#v=onepage&q&f=false

Prílohy

Príloha číslo 1 : Dotazník pre rodičov detí navštevujúcich prvý stupeň ZŠ

Dotazník pre rodičov detí ZŠ navštevujúcich prvý stupeň ZŠ

1. Počet detí vo vašej rodine
   a. 0-1
   b. 2-3
   c. 4 a viac
2. Počet vozidiel, ktoré má vaša rodina k dispozícii
   a. 0
   b. 1
   c. 2 a viac
3. Počet krúžkov, ktoré vaše deti navštevujú mimo školy
   a. 0-1
   b. 2-3
   c. 3 a viac
4. Priemerný týždenný čas strávený rozvozom detí a plánovaním mimoškolských aktivity
   a. 0-1 hodina
   b. 2-3 hodiny
   c. 3 a viac hodín
5. Využívate pri plánovaní rodinného harmonogramu mobilné aplikácie?
   a. Áno
   b. Nie
6. Boli by ste ochotný vyskúšať aplikáciu, kt. by Vám takého plánovanie a rozvoz detí uľahčila? (upozornila vás na termíny, časy, aktuálnu dopravu...)
   a. Áno
   b. Nie
7. Nastala už vo vašej rodine situácia, že z dôvodu nedostatku, či zlej organizácie času sa Vaše dieťa nemohlo zúčastniť mimoškolského krúžku?
   a. Áno
   b. Nie

Meno: JÚLIA Ročník: 2 ZŠ: GERCENOVA

Príloha číslo 2: Matica vzdialeností

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	24	19	14	21	19	10	14	15	9	10	10	15	14	17	24
1	2	0	9	19	2	17	15	14	10	17	15	19	9	15	19	15
2	3	9	0	15	7	14	10	10	3	14	14	15	7	9	14	10
3	5	19	15	0	17	10	7	5	12	7	5	5	17	7	7	15
4	7	2	7	17	0	15	14	12	9	15	15	17	3	12	17	14
5	9	17	14	10	15	0	9	7	12	14	10	10	15	7	5	7
6	10	15	10	7	14	9	0	3	9	5	3	5	12	5	9	12
7	12	14	10	5	12	7	3	0	7	9	5	5	14	3	5	10
8	14	10	3	12	9	12	9	7	0	14	14	14	9	7	12	9
9	15	17	14	7	15	14	5	9	14	0	5	5	12	7	12	19
10	17	15	14	5	15	10	3	5	14	5	0	3	12	5	9	14
11	19	19	15	5	17	10	5	5	14	5	3	0	14	7	9	15
12	21	9	7	17	3	15	12	14	9	12	12	14	0	12	17	15
13	22	15	9	7	12	7	5	3	7	7	5	7	12	0	7	10
14	24	19	14	7	17	5	9	5	12	12	9	9	17	7	0	12
15	26	15	10	15	14	7	12	10	9	19	14	15	15	10	12	0