

# GA-GED VR: SOFTWARE PRO SESTAVU OKRUŽNÍCH JÍZD

## GA-GED VR: SOFTWARE FOR VEHICLE ROUTING

Miroslav Slivoně, Bedřich Rathouský, Hana Císařová, Jaromír Široký<sup>1</sup>

---

*Anotace:* V článku je nastíněn postup tvorby software pro sestavu tras vozidel provádějících rozvoz či svoz zásilek. Při sestavě software byl kladen akcent především na využitelnost vznikajícího prostředí pro řešení reálných úloh, čemuž odpovídá výběr použitých algoritmů (snadná rekonfigurace pro různé varianty základní úlohy) i způsob zadávání vstupních dat (sestava modelu sítě z GIS mapových podkladů)

*Klíčová slova:* úloha okružních jízd, svoz, rozvoz, genetické algoritmy, GIS mapy

*Summary:* The text outlines the process of design of software for building of delivery / pick-up vehicle routes. This software product accentuates the usability for real task solution – the algorithms (easily reconfigurable for different basic problem variants) and data entry (network model is constructed from GIS maps) serve this purpose.

*Key words:* vehicle routing problem, pick-up, delivery, genetic algorithms, GIS maps

### 1. ÚVOD

Problém optimalizace tras svozu a rozvozu je úlohou, která má pro dopravce klíčový význam. Racionální plánování jízd může přinést značné úspory – udává se, že velikost úspory nákladů po zavedení prostředí umožňujícího optimalizaci tras vozidel činí v konkrétních podmínkách 5 až 20 % [1].

Text tohoto článku popisuje genezi a prezentuje dosud implementované funkce vznikajícího software GA-GED VR, který umožňuje řešit vybrané modifikace základní úlohy okružních jízd (především s využitím genetických algoritmů, dále jen GA) a k sestavě modelu dopravní sítě využívá standardní geodata (GIS mapové podklady). Software byl vytvořen ve vývojovém prostředí Turbo Delphi 2006.

### 2. V SOUČASNOSTI IMPLEMENTOVANÉ TYPY ÚLOH

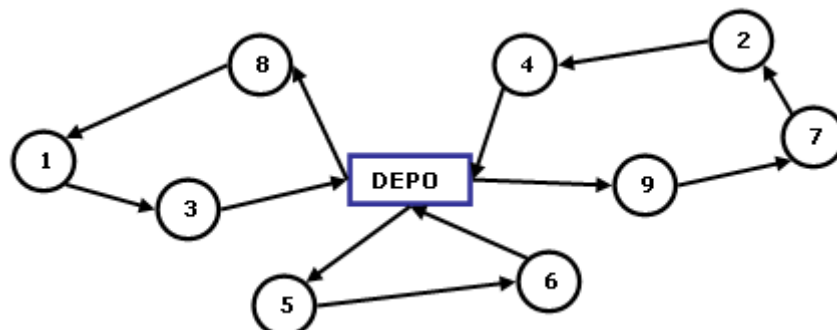
#### 2.1 Kapacitně omezená úloha okružních jízd (CVRP)

Klasickou kapacitně omezenou úlohu okružních jízd (Capacitated Vehicle Routing Problem, dále CVRP) je možné verbálně formulovat například takto: Je dáno umístění střediska  $s$  a množina zákazníků  $J$ ; přičemž jsou známy vzájemné vzdálenosti  $d_{ij}$  mezi uvažovanými objekty (tj. zákazníky a depem) na dopravní síti. Dále jsou známy požadavky jednotlivých zákazníků  $b_j$ . K obsluze zákazníků je k dispozici množina vozidel  $V$ , každé

---

<sup>1</sup> Ing. Miroslav Slivoně, Ing. Bedřich Rathouský, Ing. Hana Císařová, doc. Ing. Jaromír Široký, Ph.D., Univerzita Pardubice, DFJP, Katedra technologie a řízení dopravy, Studentská 95, 532 10 Pardubice, Tel. +420 466 036 198, E-mail: [Miroslav.Slivone@upce.cz](mailto:Miroslav.Slivone@upce.cz)

o kapacitě  $C_r$ . Každé vozidlo vyjede z depa a opět se do něj musí vrátit, každé vozidlo smí být použito pouze jednou a každý zákazník musí být uspokojen jedinou jízdou vozidla. Cílem je navrhnout takovou množinu tras vozidel, aby jejich celková délka byla minimální. Poměrně běžná je substituce vzdáleností  $d_{ij}$  časovými dostupnostmi  $t_{ij}$  – pak se nehledají trasy nejkratší, ale trasy s nejmenší délkou trvání jízdy.

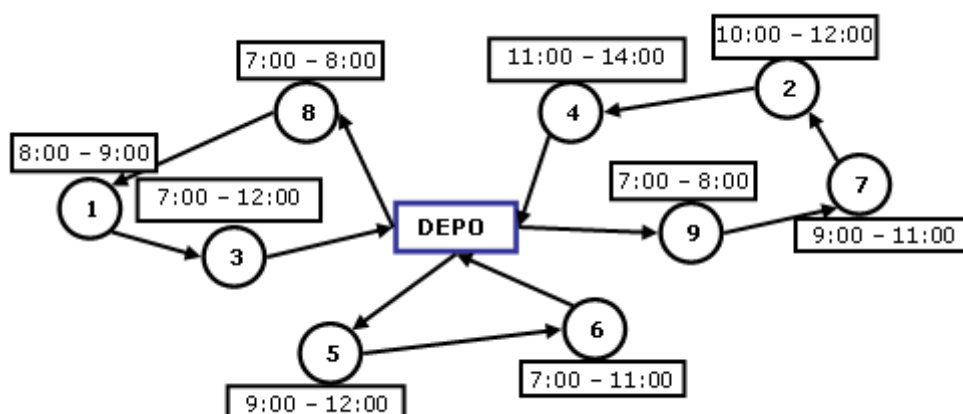


Zdroj: autoři

Obr. 1 - Schematické znázornění klasické úlohy okružních jízd

## 2.2 Úloha okružních jízd s časovými okny (VRP-TW)

Na úlohu s časovými okny (Vehicle Routing Problem with Time Windows, dále VRP-TW) je možné pohlížet jako na kombinaci úlohy okružních jízd a problému rozvrhování. Jedná se o rozšíření základní úlohy o tzv. časová okna – pro každého zákazníka je specifikován časový interval ( $e_i, l_i$ ), ve kterém je nutné zákazníka v rámci trasy obsloužit. Navíc bývá pro každého zákazníka specifikován čas obsluhy  $f_i$  a maximální doba trvání jízdy  $T_{max}$  platná pro kterékoli vozidlo. Praktická aplikace je zřejmá (omezená provozní doba ve skladu zákazníka, potřeba obdržet objednávku do určitého času apod.).



Zdroj: autoři

Obr. 2 - Schematické znázornění úlohy okružních jízd s časovými okny

### 3. POUŽITÉ ALGORITMY

#### 3.1 Clarke-Wrightova metoda

C-W algoritmus patří mezi nejstarší postupy používané k řešení klasického CVRP. Jedná se o jednoduchou heuristiku založenou na principu výhodnostních koeficientů. Její výhodou je rychlost a jednoduchost použití, nevýhodou pak relativně nízká kvalita získaných výsledků. Přesto je tento algoritmus v GA-GED VR implementován, získané řešení je možné postoupit do výchozí populace genetického algoritmu.

Tzv. sekvenční C-W metoda pracuje na principu stejných výhodnostních koeficientů, umožňuje však řešit úlohu s různou kapacitou vozidel a různým omezením délky trvání jízdy jednotlivých vozidel. Obě metody jsou popsány například v [2].

#### 3.2 Genetický algoritmus GVR

Genetické algoritmy slouží k vyhledání suboptimálního řešení složitých kombinatorických úloh, mezi které patří i úlohy o sestavě okružních jízd. Pracují na principu simulace procesu evoluce v přírodě, který je opakován až do doby, dokud není dosaženo požadované hodnoty účelové funkce, předem definovaného počtu generací či maximálního času vymezeného na hledání řešení úlohy.

V programu GA-GED VR byly implementovány genetické algoritmy založené na reprezentaci a operátorech GVR [3], [4], [5]. Důvody pro volbu GVR jsou následující:

- díky dvoustupňové reprezentaci úlohy je GVR značně univerzální – bez větších úprav umožňuje úspěšně řešit CVRP, VRP-TW a zřejmě i další varianty úloh,
- umožňuje rychle nacházet nová řešení – operátory křížení a mutace produkují nová řešení, jejichž oprava (tj. získání přípustného řešení) není časově náročná,
- velmi dobrá kvalita produkovaných řešení ověřená na standardních datech [6], [7].

Parametry genetického algoritmu (počet generací, časové stop kritérium, pravděpodobnosti použití genetických operátorů) lze měnit v prostředí programu, výsledné trasy mohou být zobrazeny jak v textové podobě, tak graficky v mapě.

### 4. ZADÁVÁNÍ A ZPRACOVÁNÍ VSTUPNÍCH DAT

#### 4.1 Konstrukce modelu dopravní sítě z geodat

Myšlenka využití standardních mapových dat určených pro GIS stála hned na počátku vzniku software GA-GED VR, protože jinak by bylo využití programu pro řešení reálných úloh značně omezené a úsilí věnované konstrukci relevantního modelu dopravní sítě by muselo být značné.

V současnosti je implementováno načítání dat formátu ESRI Shapefile, tedy souborů s příponami SHP, SHX a DBF. Pro načítání dat je využita volně dostupná knihovna Shapefile C Library [8] a její API pro Delphi [9].

Ke konstrukci modelu dopravní sítě jsou v ideálním případě potřeba 3 vrstvy dat – vrstva obsahující uzly dopravní sítě, vrstva obsahující úseky dopravní sítě a vrstva obsahující souřadnice center obcí. Pro sestavu modelu silniční sítě České republiky byla využita GIS data Silniční databanky Ostrava (uzly a úseky) [10] a Územně identifikačního registru (obce) [11], všechna data obsahují souřadnice objektů v systému S-JTSK.

Při načítání údajů o uzlech dat jsou kromě jejich geografických souřadnic využitelné atributy jako je jméno (název obce, čtvrti), kód uzlu, územní příslušnost uzlu (NUTS regiony), povaha uzlu (typ křižovatky apod.). Program umožňuje zvolit záznam v DBF souboru, který má být s daným atributem asociován.

The screenshot shows a software interface for selecting node attributes. It consists of several rows, each with a label, a checkbox, a dropdown menu, and a button. The first row is for 'NODE NAME' with a dropdown set to 'ICZUJ\_TEXT' and a button 'Assign centres'. The second row is for 'MUNICIPAL CODE' with a dropdown set to 'ICZUJ' and a button 'Harmonize codes'. The third row is for 'DESCRIPTION' with a dropdown set to 'CHAR\_UZLU' and an example value of '5'. The fourth row is for 'REGION' with a dropdown set to 'ADM1' and an example value of 'CZ0413'. The fifth row is for 'Show record No.:' with a dropdown set to '1'. There are also buttons for 'Load population' and 'Harmonize codes'.

Zdroj: autoři

Obr. 3 - Prostředí programu – výběr atributů uzlu

Při načítání údajů o úsecích jsou využitelné atributy jako: souřadnice začátku a konce úseku a mezilehlých bodů (typ polyline), délka úseku, orientace úseku a třída komunikace úseku. Protože údaje o orientaci úseku a třídě komunikace mohou být kódovány různým způsobem, je nutné, aby uživatel programu tyto údaje rozklíčoval – přiřadil hodnotám objevujícím se v DBF hodnoty, které používá GA-GED VR.

V případě orientace úseků lze zadat 3 hodnoty – úsek průjezdný oběma směry, úsek průjezdný jen od počátečního uzlu ke koncovému, úsek průjezdný jen od koncového uzlu k počátečnímu.

Pro každou třídu komunikace lze zvolit její číselné označení (1-9), tloušťku a barvu čáry pro vykreslení do mapy a průměrnou rychlost dosahovanou na dané třídě komunikace (z té jsou poté dopočítány typické časy průjezdu hranou).

Pokud jsou načteny uzly, zjistí program shodu (tolerance je volitelná) koncových vrcholů úseků s uzly a přiřadí příslušné uzly úsekům.

Select ARC NAME  SILNICE

Example: 21012

Select LENGTH  DELKA\_US Multiply the length by: 1.0

Example: 1180

Select ORIENTATION  DOPR\_SMERY Assign orientation

Example: 0

Select CATEGORY  KOD\_TR\_KOM Assign categories

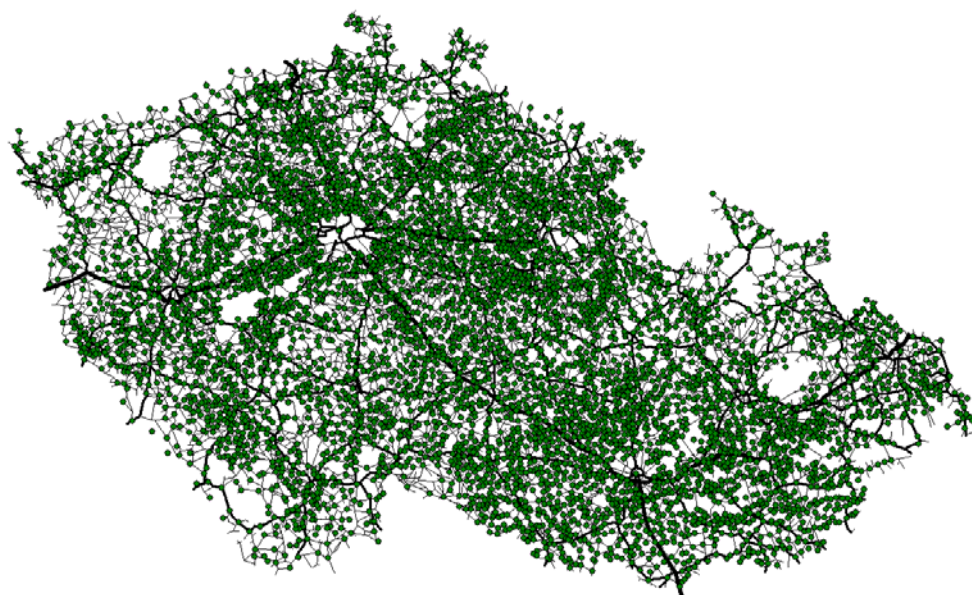
Example: 4

Show record No.: 1

Zdroj: autoři

Obr. 4 - Prostředí programu – výběr atributů úseku

Vrstva obsahující souřadnice center obcí není nutná pro běh programu, umožňuje však lepší orientaci uživatele v mapě. Tato vrstva není propojena s předchozími dvěma vrstvami – zákazníka (příp. středisko) je možné lokalizovat podle názvu obce (střed zobrazované oblasti mapy je automaticky posunut na souřadnice příslušné obce), ale poté je nutné vybrat uzel sítě nejlépe odpovídající poloze zákazníka nebo se spokojit s nabídnutým uzlem, který je nejbližší souřadnicím centra obce.



Zdroj: autoři

Obr. 5 - Načtená silniční síť ČR – zobrazeny úseky a centra obcí

Vytvořený model dopravní sítě je ukládán do textového souboru s příponou CSV, data lze prohlížet v MS Excel a editovat v libovolném textovém editoru jako prostý text.

## 4.2 Implementace algoritmů k vyhledání nejkratších cest

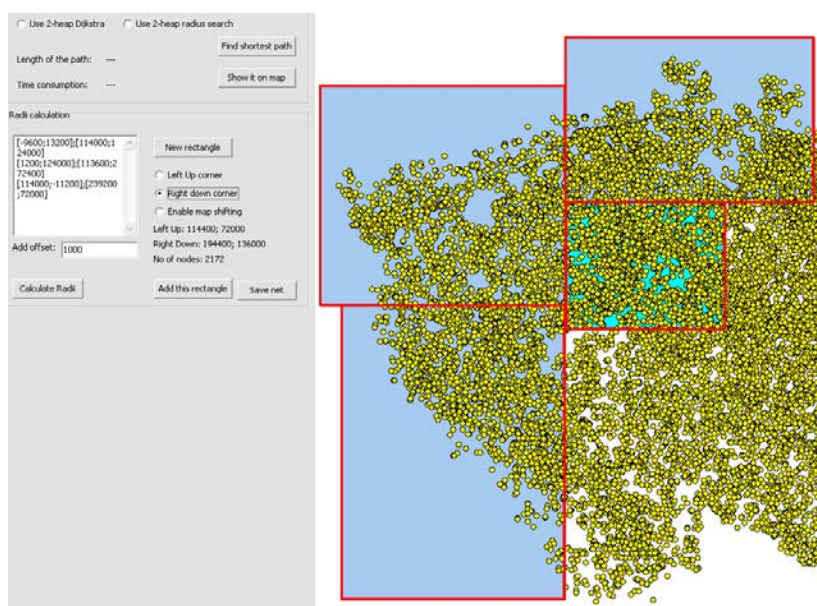
Pro potřeby běhu každého algoritmu pro řešení VRP je zapotřebí znát vzájemné vzdálenosti (případně časové dostupnosti) všech zákazníků a střediska. V relativně malých sítích, kde počet uzlů nepřesahuje několik málo tisíc, není problém přímo spočítat celou distanční matici a potřebné vzdálenosti z ní vybrat. V případě reálné dopravní sítě s desítkami či stovkami tisíců uzlů je tento postup nemyslitelný – výpočet takové distanční matice by trval příliš dlouho (i za použití postupů vhodných pro řídké dopravní sítě – např. Johnsonova algoritmu) a operační paměť soudobého počítače by ji stejně nedokázala pojmout.

Proto jsou algoritmem pro hledání nejkratší cesty mezi dvojicí uzlů počítány pouze ty vzdálenosti, které jsou skutečně potřeba. V programu je implementován Dijkstrův algoritmus, kde je fronta vrcholů k prozkoumání udržována v binární haldě, což jeho výpočetní složitost snižuje z původní  $O(n^2)$  na  $O((n+m) \log n)$ , kde  $n$  je počet uzlů a  $m$  počet úseků sítě.

I při použití Dijkstrova algoritmu s binární haldou by však ve skutečně rozsáhlé dopravní síti mohl trvat výpočet potřebných vzdáleností příliš dlouho – např. pro výpočet vzájemných vzdáleností jen mezi 100 zákazníky a 1 střediskem je na síti obsahující orientované úseky zapotřebí vyhledat 10 100 nejkratších cest. Proto je implementován ještě rychlejší algoritmus využívající tzv. rádius úseku [12], který musí být pro každý úsek předem vypočítán. Rádus úseku je hodnota, která je rovna maximální Euklidovské vzdálenosti počátku nebo konce libovolné cesty, která tento úsek obsahuje, od počátečního vrcholu úseku. Tento exaktní algoritmus může pracovat cca 40 až 60-krát rychleji než běžné algoritmy (jako je Dijkstrův algoritmus nebo  $A^*$ ) a cca 2-krát rychleji než heuristický hierarchický algoritmus (uvedeno v [12], výsledky dosaženy na modelu silniční sítě Německa s cca 78 500 uzly a 241 500 úseky).

Nevýhodou algoritmu je však čas potřebný na výpočet rádií úseků, protože aby byl algoritmus exaktní, je zapotřebí vyhledat nejkratší cestu mezi jakoukoli dvojicí uzlů na síti. Výpočetní čas lze do značné míry redukovat rozdělením sítě do několika menších podsítí (obsahujících cca 2 000 uzlů) a výpočet rádií úseků v těchto podsítích. Ty úseky, pro které se podařilo nalézt konečný rádus, jsou poté ze sítě vypuštěny (obvykle se jedná o cca 85 % úseků) a pro zbytek sítě se dopočítá horní odhad velikosti rádií.

Výpočet rádií pro výše zmíněnou síť ČR o cca 22 300 uzlech a 30 100 úsecích zabere okolo 30 minut na procesoru Pentium IV o frekvenci 3 GHz; tento výpočet přirozeně stačí provést pro danou síť pouze jednou, vypočtené hodnoty jsou poté uloženy do souboru. Výběr podsítí o přibližně stejném počtu uzlů je prováděn v jednoduché grafické pomůcce.



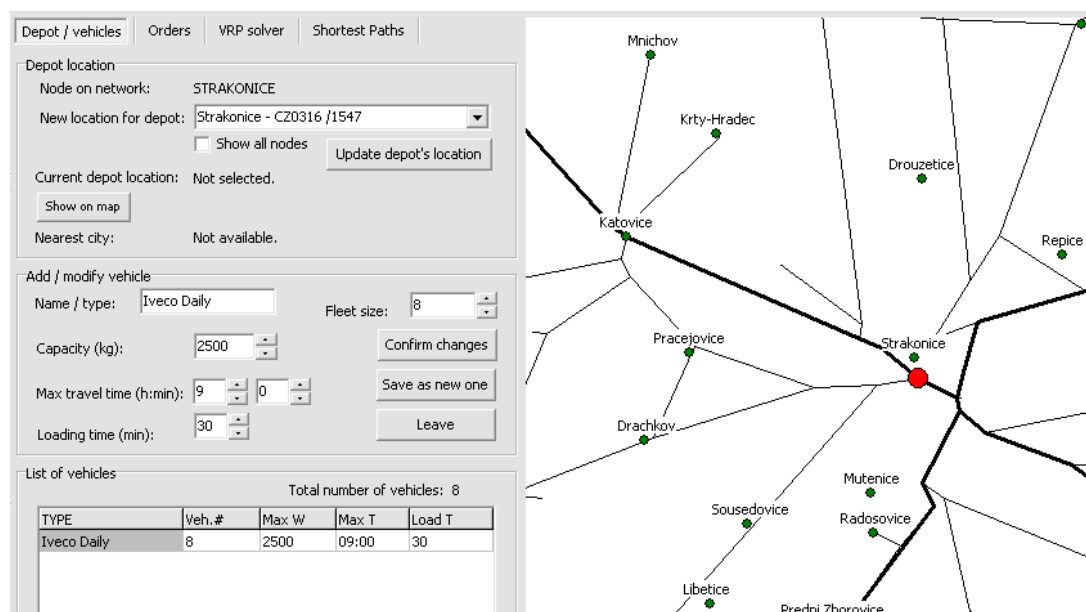
Zdroj: autoři

Obr. 6 - Prostředí programu – výběr podsítí pro výpočet rádií úseků

### 4.3 Zadávání dat o středisku, vozidlech a zákaznících

Umístění střediska – depa lze vybrat buď se seznamu obcí / uzlů (střed zobrazované oblasti mapy je automaticky posunut na souřadnice příslušné obce či uzlu) nebo vybrat přímo z mapy. V současnosti program umožňuje práci pouze s jediným střediskem.

Při zadávání vozidel je nutné daný typ vozidla pojmenovat, vyplnit jeho užitečnou hmotnost a omezit maximální trvání jeho jízdy. Není nutné každé jednotlivé vozidlo zadávat zvlášť, lze najednou zadat libovolný počet vozidel, které se ve zmíněných parametrech shodují.



Zdroj: autoři

Obr. 7 - Prostředí programu – umístění depa a zadávání údajů o vozidlech

Při vytváření nové objednávky je nutné vyplnit jméno zákazníka, vybrat jeho umístění (stejný postup jako v případě výběru umístění depa), specifikovat, zda se jedná o svoz nebo rozvoz, uvést hmotnost zásilky, nejdříve možný a nejpozdější čas obsluhy zákazníka (časové okno), čas potřebný na obsluhu zákazníka.

The screenshot shows a software interface for creating orders. On the left, there is a 'Create / modify order' form with the following fields and values:

- Node on network: (empty)
- Customer's location: Horazdovice - CZ0322 /1883
- Customer's name: Order # 28
- Parcel weight (kg): 2
- Service time (min): 6
- Earliest time (h:min): 6
- Latest time (h:min): 17

Below the form is a 'List of orders' table:

Customer	Address	D/P	Weight	T-Window	Serv. T
8467	STARE HRADISD	8467	141:07 - 141	8467	
9824	PARDUBICE	D	9824	163:44 - 163	9824
11146	SVITAVY	D	11146	185:46 - 185	11146
11027	POLICKA	D	11027	183:47 - 183	11027
11028	LITOMYSL	D	11028	183:48 - 183	11028
8412	LAZNE BOHDAN	D	8412	140:12 - 140	8412
9589		D	9589	159:49 - 159	9589
10383	BOROHRADEK	D	10383	173:03 - 173	10383
8800	KUTNA HORA	D	8800	146:40 - 146	8800

On the right, there is a network map showing nodes like Horazdovice, Maly Bor, Velky Bor, Sverac, Velke Hydčice, Hejna, Strelske Hc, and Kalenice. A thick black line highlights a path from Horazdovice to Maly Bor.

Zdroj: autoři

Obr. 8 - Prostředí programu – vytváření objednávek ke svozu / rozvozu

## 5. DALŠÍ PŘEDPOKLÁDANÝ VÝVOJ A VYUŽITÍ SOFTWARE

### 5.1 Doplnění o další varianty VRP

V současnosti je program rozšiřován o možnost řešení úlohy se současným rozvozem a svozem (Vehicle Routing Problem with Simultaneous Delivery and Pick-Up Service, VRPDP). Jedná se o variantu základní úlohy okružních jízd, pro kterou je specifické, že každé vozidlo může provádět svoz i rozvoz současně. V závislosti na konkrétním požadavku zákazníka se tedy u některých zákazníků pouze nakládá, u některých pouze vykládá a u některých se provádí obě činnosti.

Jako zajímavé rozšíření se jeví úloha s tzv. měkkými časovými okny (soft time windows) – v takové úloze je dovolené časová okna porušovat, každé takové porušení je penalizováno. Implementace do GVR genetického algoritmu se jeví jako relativně snadná.

Při použití reprezentace GVR nejsou vznikající trasy vázány přímo na jednotlivá vozidla a nezohledňuje se kapacita vozidel – ta je zohledňována až ve druhém stupni interpretace trasy. Nabízí se proto vytvoření jednoduché heuristiky, která by umožňovala řešit úlohy



s různou kapacitou vozidel (v tuto chvíli je to možné pouze s využitím sekvenční C-W metody).

## 5.2 Využití GA-GED VR při ověřování výše navrženého tarifu

V rámci řešení projektu „Optimalizace svozu a rozvozu malých zásilek s využitím silniční a železniční dopravy“ je navrhována metodika výpočtu tarifů účtovaných při rozvozu resp. svozu malých zásilek. Program GA-GED je využíván pro ověření adekvátnosti navržené výše tarifu. Celý postup lze zhruba popsat takto: Na základě znalosti počtu obyvatel v jednotlivých obcích jsou s daným rozdělením pravděpodobnosti náhodně generovány objednávky na rozvoz resp. svoz. Výsledkem jsou tak velké množství objednávek, které jsou na daném území víceméně reálně rozmístěny. Pro tyto objednávky jsou pak sestaveny trasy vozidel a podle kalkulačního vzorce spočítány náklady na tyto trasy. Výsledné náklady jsou pak porovnány s příjmy utrženými při účtování dle daného tarifu.

## 6. ZÁVĚR

Autoři se domnívají, že se podařilo připravit prostředí pro řešení úloh VRP, které umožňuje jednoduše sestavit funkční model dopravní sítě z v současné době snadno dostupných geodat a v reálném čase si poradí s hledáním nejkratších cest i ve značně rozsáhlých sítích. Implementované algoritmy na řešení CVRP a VRT-TW podávají velice uspokojivé výsledky. V současné době je GA-GED VR stále vyvíjen a testován, jsou implementovány nové typy úloh a algoritmů.

*Příspěvek vznikl za podpory řešení projektů CG932-019-520 „Optimalizace svozu a rozvozu malých zásilek s využitím silniční a železniční dopravy“ a Institucionálního výzkumu MSM 0021627505 „Teorie dopravních systémů“.*

## POUŽITÁ LITERATURA

- [1] Hall, R., Partyka, J. On the Road to Mobility – 2008 survey of vehicle routing software. Dostupné z <<http://www.lionhrtpub.com/orms/orms-2-08/vrss.html>>.
- [2] Janáček J. Optimalizace na dopravních sítích. Žilina. Žilinská univerzita v Žiline, 2006, ISBN 80-8070-586-0.
- [3] Pereira, F.B., Tavares, J., Machado, p. Costa, E. GVR: A New Genetic Representation for the Vehicle Routing Problem. In: Proceedings of the 13th Irish Conference on Artificial Intelligence and Cognitive Science, Springer-Verlag, 2002.
- [4] Pereira, F.B., Tavares, J., Machado, p. Costa, E. GVR: On the Influence of GVR in Vehicle Routing. In: Proceedings of the 2003 ACM Symposium on Applied Computing, ACM Press, 2003.

- [5] Pereira, F.B., Tavares, J., Machado, p. Costa, E. GVR: GVR Delivers It on Time. In: SEAL02 4th Asia-Pacific Conference on Simulated Evolution And Learning, 2002.
- [6] Dostupné z <<http://www.branchandcut.org/VRP/data>>.
- [7] Dostupné z <<http://web.cba.neu.edu/~msolomon/problems.htm>>.
- [8] Dostupné z <<http://shapelib.maptools.org/>>.
- [9] Dostupné z <<http://www.triplexware.huckfinn.de/shpapi.html>>.
- [10] Dostupné z <<http://www.rsd.cz/doc/Silnicni-a-dalnicni-sit/Silnicni-databanka-Ostrava/vyuziti-informacni-zakladny>>.
- [11] Dostupné z <<http://grass.fsv.cvut.cz/gwiki/FreeGeodataCZ>>.
- [12] Ertl, G.: Shortest Path Calculation in Large Road Networks. OR Spectrum, Springer-Verlag, 1998.