

Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky a kvantitativních metod

Optimalizace plánované trasy s více body
Bakalářská práce

Autor: David Sládeček
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Milan Košťák

Hradec Králové

Duben 2020

Prohlášení:

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 28.4.2020

.....
David Sládeček

Poděkování:

Děkuji vedoucímu bakalářské práce Ing. Milanu Košťákovi za metodické vedení práce a přínosné podněty při jejím vypracování. Mimo jiné bych chtěl poděkovat rodině a partnerce za podporu.

Anotace

Práce se zabývá problematikou grafových algoritmů hledajících nejkratší cestu a jejich využití v praxi v součinnosti s mapovými aplikacemi. Součástí práce je představení problému obchodního cestujícího a návrh jeho řešení za použití k-means shlukové analýzy pro jednotlivé vrcholy. Praktickým použitím navrhovaného řešení je automatizovaný výpočet trasy pro spediční společnost. V teoretické části je čtenáři přiblížena oblast logistiky a samotné spedice. V návaznosti na grafové algoritmy jsou uvedeny příklady již vytvořených aplikací, které využívají grafových algoritmů. Výsledkem práce je návrh optimalizačního skriptu využívající euklidovských vzdáleností pro vytváření shluků a výpočtu trasy. Skript na vstupu získá data v podobě informací o polohách bodů a počtu shluků, do kterých jsou body umístěny. Ze vstupních dat vytvoří přeskupenou posloupnost adres, která vytváří itinerář cesty.

Annotation

Title: Optimising a scheduled route with multiple points

The content of the thesis is the issue of graph algorithms computing the shortest path and use of these algorithms using map application to show the result. Part of the thesis is the presentation of the Travelling Salesman Problem and the design of a solution using k-means cluster analysis for the layout of vertices. The practical use of the proposed solution is an automated route calculation for delivering company. In the theoretical part, the reader is introduced to the field of logistics and delivering itself. Several applications using graph algorithms are presented, and the use of these applications is described. The result of the thesis is the design of an optimization script using Euclidean distances to create clusters and compute the route. The input of the script consists of data in the form of information about the position of the points and the number of clusters. It creates a rearranged sequence of addresses from the input data, which creates a path itinerary.

Obsah

1	Úvod.....	1
2	Cíl práce.....	3
3	Metodika zpracování.....	4
4	Logistika a silniční spedice	6
4.1	Úvod do logistiky.....	6
4.2	Využití logistiky ve spedici	7
4.3	Spediční plánování.....	7
4.3.1	Současné plánování spedice balíků	8
5	Grafové algoritmy	9
5.1	Vymezení pojmů	9
5.2	Nalezení nejkratší cesty	9
5.2.1	Bellman – Ford algoritmus.....	10
5.2.2	Dijkstrův algoritmus.....	10
5.3	Problém obchodního cestujícího.....	11
6	Řešení využívající grafové algoritmy	12
6.1	mPUD	12
6.2	Circuit Route Planner.....	12
6.3	Mapové aplikace	13
6.3.1	Google Maps.....	13
6.3.2	Mapy.cz	14
6.4	Srovnání.....	16
7	Optimalizace trasy návrh	17
7.1	Možnosti optimalizace TSP	17

7.1.1	Exaktní.....	17
7.1.2	Heuristické a aproximační optimalizace	18
7.2	Navrhovaná optimalizace.....	19
7.2.1	Nalezení shluků	20
7.2.2	Nalezení nejkratší cesty mezi shluky.....	20
8	Implementace optimalizace trasy	22
8.1	Použité technologie	22
8.2	Wireframe aplikace	23
8.3	Výpočet trasy	24
9	Shrnutí výsledků	26
9.1	Výsledná trasa	26
9.2	Výpočetní náročnost	30
10	Závěr.....	32
11	Seznam použité literatury	34

Seznam obrázků

Obrázek 1 Ukázka webové aplikace Maps Google v anglické lokalizaci.....	13
Obrázek 2 Ukázka z webové aplikace Mapy.cz v anglické lokalizaci.....	14
Obrázek 3 Ukázka mobilní aplikace Mapy.cz s turistickou mapu.....	15
Obrázek 4 Diagram řešení výsledné trasy.....	21
Obrázek 5 Návrh vzhledu aplikace s označením funkčních prvků.....	23
Obrázek 6 JavaScriptový kód s cyklem pro rozdělení bodů do clusteru podle volených centroidů.....	25
Obrázek 7 Vizualizace referenční mapy ve webové aplikaci Mapy.cz.....	29
Obrázek 8 Vizualizace vypočtené trasy se šesti shluky ve webové aplikaci Mapy.cz.....	30

Seznam tabulek

Tabulka 1 Referenční trasa řidiče.....	26
Tabulka 2 Vypočtená trasa se čtyřmi shluky.....	27
Tabulka 3 Vypočtená trasa s pěti shluky.....	27
Tabulka 4 Vypočtená trasa s šesti shluky.....	28
Tabulka 5 Srovnání jednotlivých tras.....	29

1 Úvod

Pouze s obtížemi si většina obyvatel dokáže představit, že se dopravuje jinak než za pomoci využití stále rostoucí silniční sítě. Ta se může na jednu stranu zdát, že je nedostatečná, ale zvyšující se zájem o silniční dopravu naznačuje, že i přes nedostatky, které mají komunikace v Česku, je to nejčastější volba mezi dopravci pro přepravu zásilek. Kvůli zvýšené hustotě provozu na silnicích, která je zapříčiněna lidmi, denně dojíždějící do zaměstnání, je situace pro dopravce náročnější. Do jisté míry lze běžným lidským uvažováním odhadnout, kde lze očekávat větší nápor řidičů, ale stále je prvním bodem při řešení přepravy, ať již lidí, či zboží, správně naplánovaná trasa.

Ze sociodemografické analýzy pro město Hradec Králové (Centrum EP, 2013, s 134.) vypracované v roce 2013 můžeme vyčíst, že v roce 2011 denně téměř 13000 obyvatel dojíždělo do Hradce Králové do zaměstnání a necelých 3000 obyvatel za prací odjíždělo mimo město. Počet osob na silnicích je tedy opravdu velký a mimo optimální trasy je třeba uvažovat i s možnými dopravními komplikacemi v podobě kolon nebo jiných omezení.

Nejpalčivější oblastí, která vyžaduje co možná nejlepší nejen pokrytí, ale také nejméně nákladnou trasu, jsou spediční společnosti. Mezi soukromé spediční firmy, jakými jsou například Amazon, PPL, DPD a jiné lze zařadit i národní podnik Česká pošta, která rovněž vyžaduje úsporu prostředků a času pro poskytování zejména listovních služeb. Všechny tyto společnosti mají jedno společné, a to přepravu z jednoho místa do mnoha míst až ke koncovému zákazníkovi. O služby dopravců je veliký zájem. Kupříkladu společnost PPL v předvánočním období, které je pro všechny doručovací společnosti nejnáročnější, doručí až 200 000 zásilek za jediný den (Zdroj: PPL.cz). Čas je tedy nejcennější komoditou, se kterou je třeba nakládat co možná nejlépe.

Doposud jsou v této práci zmíněny pouze varianty dopravy, kdy dopravní společnost doručuje zásilky fyzickým případně právnickým osobám na příslušné adresy. S touto problematikou, tedy přeprava zboží a materiálu, se ovšem setkává i kterákoliv jiná společnost, která přerozděluje vlastní skladové zásoby mezi jednotlivými pobočkami. V této práci se ovšem na tuto oblast nezaměříme a soustředíme se na

spediční společnosti, které v rámci města, regionu nebo jiné zájmové oblasti potřebují vyřešit optimální trasu pro rozvoz zásilek, které jsou jim svěřeny do přepravy.

Pro řešení průjezdu trasy mezi všemi adresami, které řidič musí absolvovat lze snadno uvažovat jako o problému nalezení nejkratší cesty v grafu. V takovém případě jsou vrcholy grafu jednotlivé adresy a hrany grafu samotná silniční síť. Pokud řidič vytváří trasu pouze s několika zastávkami, je zbytečné uvažovat o nějaké automatizaci nebo optimalizaci této trasy. Naopak v případě několika desítek až stovek zastávek je už prakticky nemožné bez nějaké pomoci vytvořit trasu, která by měla požadované parametry. Pro takové případy již saháme po automatizačních nástrojích, které nám v menší nebo větší míře pomohou optimální trasu získat.

2 Cíl práce

Jak již bylo zmíněno v úvodu práce, primárním cílem je navrhnout nástroj, v tomto případě single page aplikaci, česky jednostránkovou aplikaci, která by ulehčila práci komukoliv, kdo potřebuje plánovat trasu s větším množstvím zastávek. Cílovým uživatelem jsou primárně řidiči spedičních společností, kteří mají oproti běžnému uživateli několikanásobně více zastávek. Nástroj navržený v této práci poslouží pro optimalizaci referenční trasy uskutečněnou kurýrem společnosti PPL.

Teoretická část se nejprve zaměřuje na pojem logistiky a silniční spedice. Dále jsou zmíněny grafové algoritmy vhodné pro zamýšlený nástroj. Podrobněji je popsán na problému obchodního cestujícího, kterým lze optimalizovat nalezení nejkratší cesty v grafu. Pro optimalizaci lze použít několik přístupů, které jsou v jednotlivých kapitolách představeny. Součástí teoretické části je přehled současných nástrojů a aplikací zaměřujících se na vytváření tras s více či méně průjezdními body, včetně vzájemného porovnání. V závěru teoretické části je popsáno navrhované řešení, které využívá poznatků z kapitol předešlých.

Praktická část se následně zaměřuje na návrh optimalizačního nástroje. Návrhu předchází analýza potenciálních požadavků na daný nástroj. V rámci návrhu a analýzy jsou představeny i technologie užité při tvorbě nástroje. Výsledek, vzniklý z navrhovaného řešení, je porovnáván s konkrétní trasou řidiče spediční společnosti PPL. Při posuzování kvality výsledku dosaženého nástrojem je přihlédnuto k několika kritériím. Hlavním kritériem je samotná trasa. Na ni lze pohlížet jak z časového, tak vzdálenostního hlediska. V obecnosti není ekvivalentní tvrzení, že kratší trasa znamená méně času nutného k uskutečnění trasy. Stejně důležitým kritériem je samotná výpočetní náročnost. V závěru praktické části je ukázán dosažený výsledek nejen ze vzorové trasy, ale rovněž výsledků vzniklých při testování.

3 Metodika zpracování

V rámci zpracování se pokusíme vyspecifikovat konkrétní problematiku, která pokrývá obsah práce. V úvodní kapitole a kapitole popisující cíl práce je opakovaně zmiňován nástroj pro spediční společnost. Každá taková společnost má jiné požadavky a při návrhu je nutné k těmto požadavkům přihlížet. Spedice je široká oblast, která je popsána v dalších kapitolách. Rozvoz zásilek je pouze jeho dílčí částí a na tuto problematiku můžeme nahlížet z různých úhlů. Jedním z možných pohledů, je pohlížet na rozvoz balíčků komplexně, tedy rozvoz a svoz zásilek, jak popisuje (Veenstra, 2017), kde je problematika obchodního cestujícího řešena s rozšířením o systém vykládání a nakládání jednotlivého vozu.

Pro rozvoz, který je bližší tématu práce, je podobná problematika distribuce lékařských prostředků v Mexiku (Lechuga, 2018), kde jsou blízké nemocnice sjednoceny do jednoho uzlu a tím se redukuje počet vrcholů pro hlavní trasu. Nemocnice tvořící jednotlivý vrchol jsou pak obslouženy stejným způsobem. Tedy problém nalezení cesty zúžíme pouze pro daný vrchol, který je reprezentován větším počtem nemocnic.

V předchozím odstavci Lechuga pojednává o řešení rozvozu materiálu pro jednotlivé nemocnice, které jsou shlukovány do skupin. V obecnosti se může v rámci shlukování jednat o libovolná lineární, plošná nebo prostorová data, která můžeme seskupovat. Jedním z možných přístupů, jak seskupovat data, je k-means shluková analýza, kterou ve své knize popisuje Hartigan (Hartigan, 1975, s. 84-112). Shlukování dat je zde popsáno na výživových hodnotách vybraných potravin. Jednotlivé shluky jsou vytvářeny podle energetické hodnoty potravin a obsahu proteinu. Shluková k-means analýza se sestává z přerozdělování vstupních hodnot mezi daný počet shluků na základě velikosti chyby v blízkém sousedství. Zjednodušeně lze výsledek shlukové analýzy za pomoci k-means analýzy popsat jako seskupování blízkých hodnot do určitého počtu shluků. Jednotlivé shluky jsou vytvářeny v okolí vybraných hodnot, centroidů, a proto tento algoritmus řadíme mezi centroidní.

Mimo již zmíněnou k-means analýzu, existují i jiné přístupy, konkrétně hierarchické a hustotní.

Nástroj, který je náplní této bakalářské práce, je problematikou zúžen pouze na fázi doručování bez následného nakládání přepravovaných zásilek. Tím je ovšem vytvořena sekundární skupina potenciálních uživatelů, kterými jsou společnosti zajišťující služeb messengeru nebo osoby, plánující vlastní trasu pro soukromé účely, kdy plánují velké množství zastávek a nejsou si jisti vlastním řešením navržené trasy.

Pro jednoduchost není v práci zohledněn problém provozu na silnicích. Trasa, která je nástrojem vytvořena, by se tak měla blížit celkovou vzdáleností k optimu, ovšem bez přihlédnutí na současný stav komunikace. Ať již se jedná o zvýšený provoz, uzavírky některých silnic nebo poplatky spojené s využitím komunikace.

Pro lepší porozumění problematice logistiky a samotné přepravy zboží a materiálu jsou v teoretické části práce uvedeny základní poučky, které jsou v rámci řízení podniku a jeho logistiky vyučovány zejména na dopravních univerzitách. Práce se zaměřuje především na problémy, které souvisí s přepravou a doručováním. Zajímavým tématem, který souvisí se spedicí, je například optimální hustota skladů. Samotná hustota skladů je jedním z ukazatelů dobré spediční společnosti.

4 Logistika a silniční spedice

V následující kapitole je přiblížena problematika logistiky včetně základních pojmů, které jsou dále používány. Současně je vysvětlen rozdíl mezi spedicí a logistikou. V závěru kapitoly je nastíněno, jakým způsobem spediční společnosti plánují a organizují trasu pro rozvoz zásilek. Kapitola vychází z (Bakešová, 2008) a (Svoboda, 2004).

4.1 Úvod do logistiky

S logistikou se můžeme setkat již v dávných dobách, kdy lidé budovali první společenství. Ukázkou prvotní logistické činnosti bylo skolení mamuta. Takto velké zvíře nelze zastavit lidskou silou a bylo tedy nutné vykopat jámu, do které byl následně mamut nahnán a v dané jámě zabit. Logistika zde spočívá v procesu, kdy kopáči jam potřebovali nástroje pro svou práci, jiní odstraňovali vykopanou zeminu a další jámu maskovali. Každá tato skupina musela být nějakým způsobem živena. V neposlední řadě se pak řešilo zpracování a transport uloveného mamuta.

Postupem času se logistika zakořenila do lidského života. Bez správného logistického řešení nebylo možné řídit malé osady, ani velká města. Za pomoci nauky o logistice pak dávní vojevůdci dobývali území vzdálená od svých domovin. Z toho tedy můžeme historicky oddělit logistické větve na civilní a vojenskou, která se každá ubírala vlastním směrem. Přesto fungovaly paralelně a často se vzájemně doplňovaly.

Prvotní zmínky o distribuci stavebního materiálu a způsobu zásobování stravy pro dělníky nalezneme v Číně 6 tisíc let před naším letopočtem. Daleko přesnější a podrobnější záznamy nalezneme v oblasti dnešního Egypta 4 až 3 tisíce let před naším letopočtem v období staveb pyramid. Takto bychom mohli pokračovat až do novodobých dějin. Ve vojenství se pojem logistika začal častěji objevovat až po druhé světové válce. Toto odvětví logistiky zahrnuje široké spektrum činností od administrativy až po samotné vyzbrojování jednotek. Termín byl později spjat i s hospodářskou činností.

Logistika je tedy procesem zajištění optimálního přesunu zboží, které zahrnuje samotnou tvorbu, řízení a organizaci materiálu, zajištění finančních a informačních toků a dalších s nimi spjatými. Logistické řešení napomáhá k naplnění a dosažení kýženého oběhu zboží.

4.2 Využití logistiky ve spedici

V předcházející podkapitole je zmíněna v příkladu s mamutem přeprava zpracovaného úlovku. Spedice má s přepravou opravdu mnoho společného, ale samotný transport zboží z jednoho bodu do druhého není pro spedici výstižné. Nutno říct, že spedice není to samé, co logistika. Lze tvrdit, že spedice využívá některých metod logistiky, ale rozhodně mezi těmito pojmy neexistuje ekvivalence. Spedice označuje službu zasílání, kterou firma poskytuje jiným. Ve stručnosti se tedy jedná o službu, kdy je firma poptávána na přepravu zboží za úplatu. Spedice je ve své podstatě mikro logistickým odvětvím, která je součástí celého procesu logistiky.

Ve spedici tak využíváme logistických přístupů, tedy uspořádání jednotlivých kroků do posloupnosti, která umožňuje optimalizovat prostředky nutné k celému procesu. Stejně jako logistika i spediční společnosti musí pružně reagovat na požadavky trhu. Reakce na změnu je spjata s krátkodobým i dlouhodobým plánováním.

S plánováním je spjata mnoho dílčích částí, jedním z nich je i hustota skladů. Hustotou zde není myšleno, jak moc je sklad využit v rámci jedné budovy, ale jaká je síť skladů v zájmové oblasti. Pomocí různých metodik dokáže firma nebo spediční společnost určit, zda existuje nadbytek nebo nedostatek skladových prostor.

4.3 Spediční plánování

Každý krok logistického procesu vyžaduje určité náklady a je potřeba zajistit dostatek prostředků pro jejich realizaci. V poslední době největší náklady padají na přímé logistické náklady, mezi které patří i doprava. Každá společnost, která se zabývá samotnou spedicí nebo komplexním logistickým procesem, hledá způsoby, jak tyto náklady snižovat.

Ohledně spedice a jejího plánování tak mluvíme o optimalizaci přepravovaných zásilek. Podle dispozic dopravce tak můžeme kombinovat silniční, železniční a leteckou síť. Dalším faktorem, který dopomůže k lepší organizaci, je správné zahuštění svozových dep, které nadále působí jako počáteční bod pro rozvoz zásilek do blízkých oblastí. Tím se docílí rychlejšího dodání většího objemu zásilek oproti místům, která jsou od svozového depa více vzdáleny.

4.3.1 Současné plánování spedice balíků

Proces přepravy zásilek uskutečněných spedičními společnostmi konkrétního depa můžeme rozdělit na tři časová okna. V prvním časovém okně začínající zhruba okolo šesté hodiny ranní jsou veškeré zásilky, které byly dovezeny z ostatních skladů, vyloženy na pás a podle informací uvedených na balíčku jsou přerozděleny řidičům.

V druhém časovém okně mezi osmou a šestnáctou hodinou jsou zásilky, rozdělené mezi řidiče, rozváženy po hlavních trasách. Každý řidič má přidělenou oblast, obvykle městskou část, kterou pravidelně navštěvuje. V této oblasti si na základě vlastních zkušeností vytvoří vlastní ideální trasu, kterou si v malé míře upravuje pro potřeby konkrétního rozvozu. Na doručovací trasu pak plynuje navazuje svozová trasa, na které řidič navštíví místa, kde převezme balíčky do přepravy a následně se vrací na depo.

V třetím časovém okně mezi šestnáctou a šestou hodinnou ranní jsou v druhé vlně rozváženy balíčky do oblastí s obvykle řidší rozvážkou. Zásilky, které jsou dopraveny na depo, jsou následně třízeny a nakládány do kamionů, které jsou po naplnění vypraveny do ostatních dep.

5 Grafové algoritmy

Kapitola grafové algoritmy pojednává o grafech z matematického hlediska. Nalezneme zde definici grafu společně s definicemi jeho částí a jak jsou tyto části dále popisovány. Součástí kapitoly je také přehled algoritmů pracujících s grafy, které poslouží jako teoretický podklad pro samotný nástroj vyhledávání trasy. V závěru si představíme, problém obchodního cestujícího, který řeší, jakým způsobem nejkratší cestu hledat.

5.1 Vymezení pojmů

Pojem graf a hrana grafu je dle Matouška v knize Kapitoly diskrétní matematiky (Matoušek, 2000 s. 95) popsán následovně:

Obyčejný graf G je uspořádanou dvojicí

$$G = \{V, E\}$$

pro kterou platí, že V je libovolná množina vrcholů a E představuje množinu hran daného grafu. Každá hrana je určena dvěma vrcholy.

$$E \subseteq P_2(V)$$

$$e = \{v_i, v_j\}, v_i, v_j \in V, e \in E$$

Následující pojmy odpovídají definicím uvedené Milkovou (Milková, 2013 s. 17) ve studijních materiálech pro výuku Diskrétní matematiky vyučované na Univerzitě Hradec Králové.

Sled S je konečnou posloupností

$$S = \{v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k\}, i = 0, \dots, k, e_i = \{v_{i-1}, v_i\}$$

Tah je takový sled, ve kterém se žádná hrana nevyskytuje dvakrát.

Jako **cesta** se označuje otevřený tah (nekončí v počátečním vrcholu), ve jsou všechny vrcholy různé.

Kružnici nazýváme uzavřenou cestu. Tedy takou, která končí v počátečním vrcholu.

5.2 Nalezení nejkratší cesty

Pro automatizované hledání nejkratší cesty v grafu bylo v minulosti vytvořeno několik algoritmů, které pracují s různými postupy. Ty nejznámější si nyní představíme. Kromě přehledu některých známých algoritmů si představíme i problém obchodního

cestujících, z anglického Travelling Salesman Problem (dále TSP), který je spjat s řešením nejkratší cesty.

5.2.1 Bellman – Ford algoritmus

Obecný Bellman-Ford algoritmus je dle Cormena (Cormen, 2009 s. 650) algoritmus, řešící problém nejkratší cesty z počátečního bodu i v případě, že váha hrany je záporná. Bellman-Fordův algoritmus vrací pravdivostní hodnoty, zda se ve tvořené cestě tvoří nebo netvoří záporná smyčka, které lze dosáhnout z počátečního vrcholu. Pokud je taková smyčka v grafu odhalena, algoritmus ohlásí neexistenci řešení. V opačném případě algoritmus sestrojí nejkratší váženou cestu.

Algoritmus využívá relaxace hran, která spočívá v aktualizaci vah hran po přidání dalšího vrcholu do cesty. Váha hran se může pouze zmenšovat. Pro ukázkou fungování si ukážeme pseudokód uvedený Cormenem (Cormen, 2009 s. 651).

```
BELLMAN-FORD( $G, w, s$ )
1:   INITIALIZE-SINGLE-SOURCE( $G, s$ )
2:   for  $i \leftarrow 1$  to  $|V[G]| - 1$ 
3:       do for each edge  $(u, v) \in E[G]$ 
4:           do RELAX( $u, v, w$ )
5:   for each edge  $(u, v) \in E[G]$ 
6:       do if  $d[v] > d[u] + w(u, v)$ 
7:           then return FALSE
8:   return TRUE
```

5.2.2 Dijkstrův algoritmus

Algoritmus, vytvořený nizozemským informatikem Edsgerem W. Dijkstrem v roce 1956, je ve stejné publikaci (Cormen, 2009 s. 658 a dále) popsán jako algoritmus řešící nalezení nejkratší cesty ve váženém orientovaném grafu, kdy žádná z vah hran grafu není záporná, tedy $w(e_i, e_j) \geq 0$. Oproti algoritmu Bellman – Ford je při vhodné implementaci Dijkstrův algoritmus rychlejší při výpočtu. V obecnosti je rychlejší než Bellman – Ford, ale požadavkem je, aby všechny hrany byly nezáporné.

Množina vrcholů, která je určena k navštívení, využívá prioritní fronty Q , kdy priorita je určena vahou příslušné hrany. Algoritmus udržuje množinu vrcholů S , která je nejkratší cestou z počátečního vrcholu přes doposud navštívené vrcholy. Po počáteční inicializaci délek z výchozího bodu s opakovaně vybírá vrchol $u \in V - S$ s nejmenší

vahou hrany. Následně je tento vrchol přidán do množiny S a relaxují se hrany vycházející z přidaného vrcholu. Algoritmus končí v momentě, kdy je vstupní množina vrcholu prázdná. Následuje ukázka pseudokódu z publikace Cormena (Cormen, 2009 s. 659).

```
Dijkstra (G, w, s)
1:   INITIALIZE_SINGLE_SOURCE(G, s)
2:   S = {}
3:   Q = G.V
4:   while Q ≠ {}
5:       u = EXTRACT_MIN(Q)
6:       S = S ∪ {u}
7:       for each vertex v ∈ G.Adj[u]
8:           RELAX(u, v, w)
```

5.3 Problém obchodního cestujícího

V následující podkapitole se seznámíme s problémem obchodního cestujícího (TSP). Problém je v odborných kruzích dostatečně známý a v následující kapitole je tedy uveden jeho stručný popis a jeho zařazení v příslušné kategorii problémů.

TSP zobecňuje problém hledání nejkratší cesty. Algoritmus si klade za cíl nalézt nejkratší možnou cestu mezi všemi dosažitelnými vrcholy grafu. Problém je formulován tak, že máme n měst, mezi kterými vedou cesty. O těchto cestách víme, jak jsou dlouhé a kolik času zabere přesun mezi jednotlivými městy. Úkolem obchodního cestujícího je navštívit každé z těchto měst a následně se vrátit do svého výchozího města v co nejkratším čase. Tím tedy vytvoříme nejkratší cestu mezi všemi městy.

V matematice se o TSP dle Cormena (Cormen, 2009 s. 311 a dále) bavíme jako o deterministickém problému hledající nejkratší cestu spojující množinu vrcholů. Problém řadíme mezi NP-úplné problémy.

S TSP částečně souvisí problém čínského listonoše, který v grafu hledá eulerovský tah. Black ve slovníku algoritmů (Black, 1999) uvádí, že optimalizace problému Sedm mostů města Královce prezentována roku 1960 čínským matematikem Kwanem byla na jeho počest pojmenována jako problém čínského listonoše. Návrh na pojmenování doporučil Alan J. Goldman roku 1962, kdy byl vydán anglický překlad.

6 Řešení využívající grafové algoritmy

Následující kapitola obsahuje přehled několika aplikací a nástrojů, které ve větší či menší míře využívají grafové algoritmy a pracují s mapou pro návrh trasy.

6.1 mPUD

Aplikace mPUD je vytvořená společností eMan jako zakázková aplikace pro společnost E.ON Distribuce a.s. Mobilní aplikace mPUD poskytuje možnost spravovat a organizovat pracovní příkazy pro techniky, kteří monitorují stav zařízení během distribuce elektrické energie.

V praktickém použití aplikace umožňuje technikovi naplánovat si jednotlivé výjezdy k zařízením a optimalizovat tak pořadí vyřízení jednotlivých úkonů. Mezi vymoženosti aplikace patří možnost pořízení snímku měřícího zařízení, případně závady a spojit tuto informaci s příslušným výjezdem. Kromě fotodokumentace výjezdu si technik může přidat k výjezdu doplňující informace, zejména seznam vybavení a nářadí potřebné k úkonu. Díky této aplikaci pak lze rozhodnout, zda je v časových možnostech technika obsloužit dané stanoviště nebo je nutné na nové místo vyslat jinou skupinu servisních techniků.

6.2 Circuit Route Planner

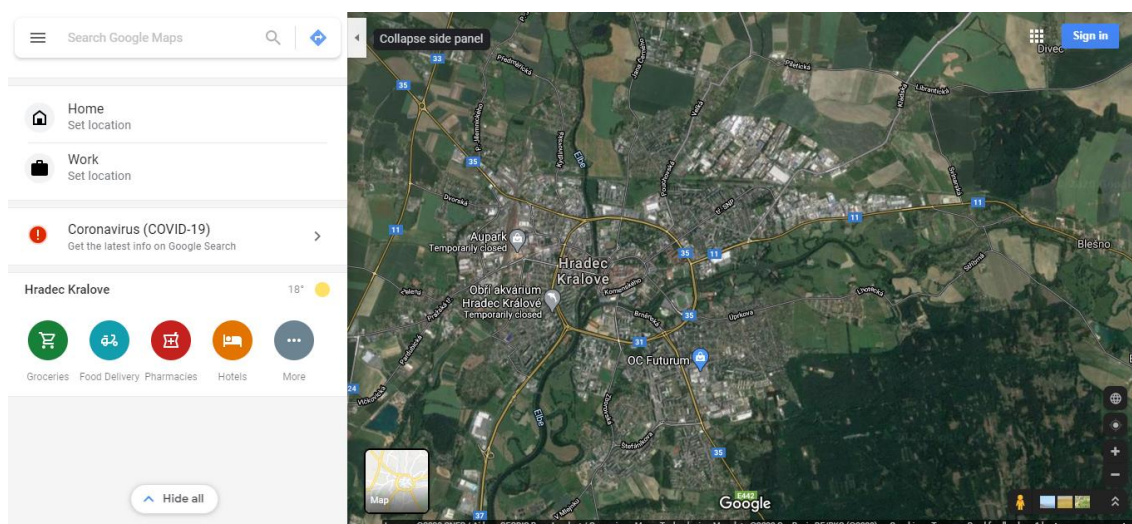
Mezi uživatele této mobilní aplikace využívané především ve Spojených státech patří spediční společnosti jako UPS, Amazon, FedEx, DHL a United States Postal Service. Hlavním přínosem této aplikace je možnost seřazení zadaných zásilek, které jsou seřazeny i s přihlédnutím na současnou dopravní situaci. Na blogu společnosti Santiago uvádí (Santiago, 2020), že používáním aplikace lze ušetřit 15-20 % času při využívání aplikace Circuit Route Planner. Aplikace je intuitivní na ovládání, ale v základní verzi je uživatel omezen pouze na 7denní zkušební dobu a následně je po vypršení zkušební doby účtován poplatek 19,99 amerických dolarů (přibližně 500 Kč) měsíčně. Přínosem je propojení s aplikací Waze a s aplikací Google Maps.

6.3 Mapové aplikace

Mezi nejznámější mapové aplikace, které mají mnoho uživatelů a nachází uplatnění jak v soukromém, tak firemní sektoru, jsou Maps od společnosti Google a Mapy.cz od společnosti Seznam CZ.

6.3.1 Google Maps

Současná dominance mapové aplikace vytvořená společností Google, s více než 5 miliardami instalací na platformách Android (Zdroj: Google Play), se samozřejmě promítla i do Česka. Tato dominance však na českém trhu nemá dlouhého trvání. Důvodem je existence mapové aplikace společnosti Seznam, Mapy.cz. Se současným počtem instalací na zařízení s Android přesahujícím jeden milion (Zdroj: Google Play) se nemůže takovému gigantu rovnat. Přesto si ovšem stále nachází své příznivce. Samotná aplikace Google Maps umožňuje vyhledávat na celé mapě světa a plánovat trasu s více průjezdy. Díky sběru dat z mobilních zařízení je aplikace schopná na základě statistických dat navrhnout změnu trasy v závislosti na provozu. Za zmínku v souvislosti s aplikací Maps stojí i aplikace Google Earth, vyvinuta stejnou společností, která poskytuje detailní pohled na planetu Země.



Obrázek 1 Ukázka webové aplikace Maps Google v anglické lokalizaci

Zdroj: maps.google.com

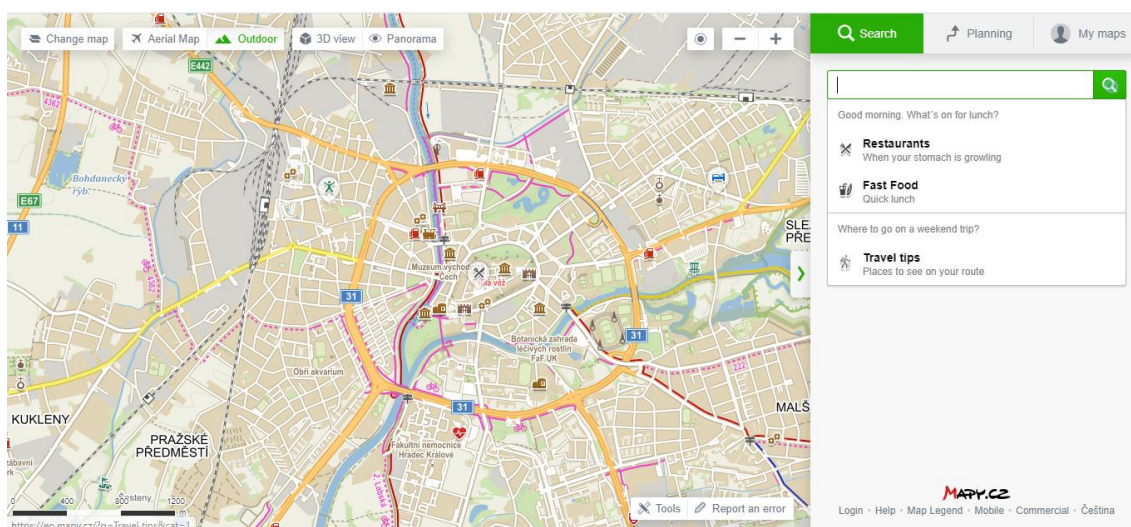
Mimo zmíněné funkce aplikace dále umožňuje zobrazit několik typů map, jako jsou letecké nebo silniční. Aplikace je využívána i ostatními aplikacemi společnosti

Google. Uživateli je za pomoci Maps umožněno vyhledávání hotelů včetně cen za ubytování, restaurací s recenzemi od návštěvníků nebo prodejny a provozovny se základními informacemi jakými jsou otevírací doba, kontakt a internetové stránky. Mapovou aplikaci využívá i služba Google Flights, která umožňuje vyhledání a zakoupení letenek včetně nalezení ubytování v daných destinacích.

Po přihlášení lze ukládat vytvořené trasy, místa a organizovat je. Přínosná je možnost zadání adresy domova a práce, pro pohodlnější tvorbu trasy. K zobrazovaným hotelům, restauracím, obchodům a podnikům lze z uživatelského účtu přidávat recenze a fotografie z daných míst.

6.3.2 Mapy.cz

Mapy.cz je velmi oblíbená aplikace mezi cyklisty a pěšími turisty. Aplikace nabízí přehledně zmapované trasy pro silniční i pěší dopravu. Turisté nejvíce ocení režim turistické mapy se všemi potřebnými značkami a barevným značením jednotlivých naučných i poznávacích tras po celé České republice.

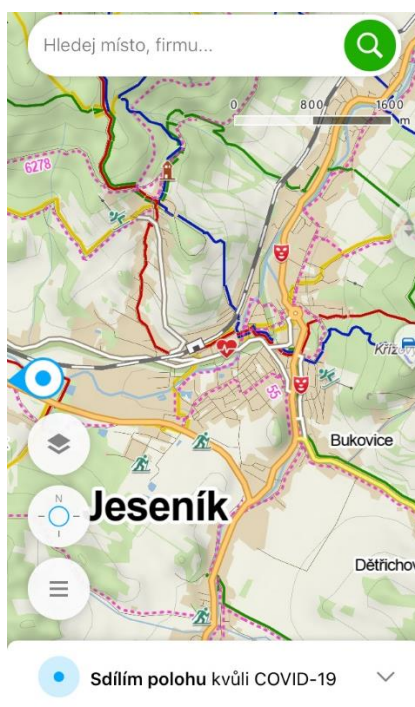


Obrázek 2 Ukázka z webové aplikace Mapy.cz v anglické lokalizaci
Zdroj: Mapy.cz

Stejně jako konkurenční aplikace Maps disponují Mapy.cz několika režimy zobrazení map. Ve výchozím nastavení je zobrazena mapová vrstva uvedená na obrázku výše a odpovídá vzhledu tištěné mapy. Mimo tuto základní mapu je možné zobrazit letecké mapy různého stáří, kdy první letecké snímky jsou z roku 2003. Dalšími režimy jsou již zmíněná turistická mapa a zimní mapa obsahující informace o lyžařských

střediscích v Česku. Pro užší zájmové skupiny je k dispozici historická mapa vycházející z podkladů map z období Rakouska-Uherska nebo geografická mapa se zajímavými informacemi o jednotlivých místech v Česku.

Mezi největší přínosy zejména mobilní verze aplikace je možnost uložení mapových podkladů do telefonu pro využití mapy bez nutnosti internetu a aplikace krokoměru. Velkou předností Mapy.cz jsou podrobnější mapy pro pěší trasy, cyklistiku a veřejnou dopravu včetně vlakových a autobusových spojů. Pro plánování trasy například v horských oblastech je do výsledného času započteno i převýšení a uživatel tak má lepší představu o trase. Stejně jako aplikace Maps je součástí aplikace panoramatické prohlížení oblastí, které je získáváno všesměrovou kamerou, která je umístěna na vozu projíždějící jednotlivé komunikace.



Obrázek 3 Ukázka mobilní aplikace Mapy.cz s turistickou mapu

Zdroj: Snímek obrazovky mobilního telefonu

Mimo osobního použití je aplikace využívána jako pomoc při hledání nakažených pacientů v souvislosti s epidemií viru Covid-19. Po schválení sdílení polohy uživatele v mobilní aplikaci jsou shromažďovány informace o pohybu uživatele. V případě potvrzení nákazy u daného jedince jsou data o pohybu předána krajské hygienické

stanici a v součinnosti s tzv. chytrou karanténou je vytvořená vzpomínková mapa, která usnadní vyhledání osob, které mohly být nakaženy.

6.4 Srovnání

První dvě zmíněné aplikace využívající trasování jsou sice každá pro jiného koncového zákazníka a s vlastní přidanou hodnotou, ale obě mají společné využití optimalizace trasy některým z dostupných algoritmů. Aplikace mPUD je ukázkovým příkladem softwaru na míru. Veškeré požadavky a specifikace jsou získány od zadavatele a použití v soukromém sektoru je tedy nemožné. Produkt Circuit Rout Planner je naopak použitelný i pro veřejnost na platformách Android a iOS. Jeho největší nevýhodou je velmi nízký počet tras v bezplatné zkušební verzi a placená verze se tak hodí pouze pro člověka, který se zabývá přepravou osob, zboží nebo doručování zásilek.

Obě mapové aplikace jsou bezplatně dostupné pro širokou veřejnost a jsou dostupné ve formě webové i mobilní aplikace pro všechny běžné platformy. Mimo to disponují vlastním API pro vývojáře. API je zkratkou pro aplikační programový interface, tedy rozhraní, které umožňuje komunikaci mezi stránkou a uživatelem.

API umožňuje využití funkcionalit jednotlivých aplikací, které vývojář využívá ve svém projektu. Stejně jako jakákoliv jiná součást softwaru, i API podléhá licenčním ujednáním. Ohledně poskytovaného API pro aplikaci Mapy.cz a Maps můžeme vidět jeden podstatný rozdíl. Zatímco Seznam poskytuje své API bezplatně, společnost Google umožňuje bezplatný provoz pouze do určité výše objemu dotazů. Přesto je vývojář, který má v plánu API od Google využívat, nucen při registraci uvést údaje o platební kartě, ze které jsou následně strhávány poplatky za provoz.

Pokud porovnáme funkcionalitu obou mapových aplikací, lze říct, že pro příležitostného uživatele mají totožnou funkcionalitu. Bez obtíží naleznou vyhledávané místo pomocí adresy nebo zjistit adresu v bodě kliknutím do mapy. Rovněž umožňují vyhledávání zájmových bodů v okolí současné polohy i vybraného bodu. V neposlední řadě lze využít možnosti plánování trasy s více zastávkami s možností dodefinování podmínek. Těmito podmínkami mohou být například preferovaná trasa. Uživatel tak může volit mezi rychlou nebo krátkou trasou, zvolit typy komunikace, například vyhýbání se zpoplatněným silnicím a dalším omezením.

7 Optimalizace trasy návrh

Jak již bylo zmíněno dříve, k nalezení nejkratší cesty lze použít Dijkstraův nebo Bellman-Fordův algoritmus v závislosti na vstupních podmínkách. Problému optimalizace trasy se věnuje TSP, který je výchozím problémem pro řešení v této práci. V podkapitolách návrhu optimalizace jsou ukázány možnosti, jak lze TSP optimalizovat a následně je představen způsob, který je použit pro optimalizaci trasy vytvořenou navrhovaným nástrojem.

7.1 Možnosti optimalizace TSP

Pro optimalizaci TSP lze využít dvou základních přístupů, jedna z možností přístupu je exaktní, kdy lze očekávat výpočetní náročnost a pro velký objem dat je nutno počítat s požadavkem na dostatečný výpočetní výkon přístroje. Druhou kategorií optimalizace jsou heuristické přístupy, které nepředpokládají dokonalou správnost řešení, ale jejich přínosem je získání přijatelného výsledku s nižším nárokem na výpočetní výkon a celkový čas nutný k dosažení výsledku.

7.1.1 Exaktní

Řešení hrubou silou (brutal force)

Nejjednodušším přístupem pro nalezení nejkratší cesty je projít veškeré permutace vrcholů a jejich hran. Tento přístup je vhodný pouze pro malý počet vrcholů. Již pro 10 vrcholů je nutno prozkoumat 3 628 800 permutací, pro vyšší počet je to již příliš velká zátěž pro výpočetní systém a v rozumném čase už nejsme schopni dosáhnout výsledku.

Lineární optimalizace

Metoda řešení pomocí lineární optimalizace se opírá o techniku lineárního programování, která hledá optimální, pro náš případ minimální řešení za pomocí soustavy lineárních rovnic s určitým počtem neznámých z dané množiny. Obecné optimalizační řešení užitím lineárních rovnic popisuje Vanderbei (Vanderbei, 2014) v úvodních kapitolách. Kde se čtenář seznámí s principem řešení lineárních rovnic z pohledu matematiky. V pozdějších kapitolách je uvedeno řešení těchto rovnic užitím algoritmů.

7.1.2 Heuristické a aproximační optimalizace

Nejbližší soused

Metoda, kterou ve své práci popisuje Hlatká (Hlatká, 2018), pojednává o jedné z jednodušších metod pro řešení optimalizace. Na počátku je náhodně vybrán počáteční vrchol. V implementacích se často vybírá vrchol s nejnižším indexem. Následně se vybraný vrchol přidá do množiny již navštívených a následně je vybrána hrana vycházející z toho vrcholu s nejmenší hodnotou. Hrana s nejnižším ohodnocením nyní spojuje počáteční vrchol s vrcholem, který je tak nově součástí navrhnuté trasy T . V dalších krocích pak hledáme vrcholy, které jsou nejbližší k poslednímu přidanému vrcholu vytvářené cesty. Algoritmus končí v momentě, kdy jsou navštíveny všechny vrcholy. Posledním krokem je dokončení cyklu, kdy je spojen poslední bod trasy s bodem počátečním.

Přístup mravenčí kolonie

Na optimalizaci TSP se později použilo i poznatků, které byly vypořazovány z chování mravenčích kolonií. Mravenci se na základě emitovaných pachů rozhodují o nejkratší cestě. Tento princip naznačuje například Pinteá (Pinteá, 2017), který popisuje užití multi agentového řešení. Chování agenta odpovídá vypořazovanému chování u mravenců. Mravenci jsou schopni nalézt nejkratší cestu mezi hnízdem a potravou za pomoci umístování aromatických sekretů. V grafové reprezentaci je tak vrchol reprezentován místem, kde mravenec tuto značku umístí. Mravenec se na základě síly tohoto feromonu rozhoduje, která trasa je pro něj výhodnější. Pro velkou skupinu mravenců tak největší množství mravenců zvolí skutečně nejkratší cestu. Ostatní mohli s jistou mírou pravděpodobnosti zvolit trasu, která nebyla nejkratší například z důvodu překážky, která stála v cestě.

Jako řešení tohoto problému je pak trasa, kterou prošel některý z mravenců, protože již víme, že některý z mravenců skutečně našel nejkratší trasu. Pro skutečné nalezení nejkratší cesty pak můžeme porovnávat množinu tras jednotlivců. Následně lze iterativně projít všechny cesty a porovnávat jejich celkovou délku nebo zvolit cestu s nejčastějším výskytem a považovat ji za nejkratší.

Shlukování vrcholů

Posledním typem heuristické optimalizace, kterou si představíme je shlukování blízkých vrcholů. Tento princip řešení společně s užitím modifikovaného Dijkstrova algoritmu popisuje Lechuga (Lechuga, 2018) při návrhu řešení zásobování nemocnic z centrálního skladu ve střední Americe, konkrétně v Mexiku. V práci je představeno několik způsobů řešení zásobování a jedním z nich je clustering, česky shlukování.

V rámci tohoto řešení je na základě analýzy určeno kritérium, podle kterého se jednotlivé nemocnice shlukují. V obecnosti lze říct, že čas strávený obslužením seskupených nemocnic by měl být menší nebo roven času, který zásobovací vůz stráví cestou ze skladu do hlavního vrcholu daného shluku. Hlavní vrchol je vybrán z množiny vrcholů shluku, jehož vzdálenost z počátečního případně předešlého vrcholu je nejmenší.

7.2 Navrhovaná optimalizace

V následující kapitole je podrobně představen návrh optimalizačního nástroje, který je v rámci této práce vytvářen.

Mezi požadavky na navrhovaný nástroj je mimo samotné vytvoření trasy také jednoduché zadávání vstupních dat. Uživatel si průjezdní místa volí pomocí vyhledávání adres nebo kliknutím do mapy. Nástroj rovněž umožňuje nahrání trasy do navigace vozidla případně zobrazení trasy v navigaci v mobilním telefonu.

Pro řešení je užito heuristického přístupu shlukování vrcholů v kombinaci s metodou nejbližších sousedů, které byly popsány výše. Důvod použití těchto přístupů je jejich jednoduchost v kombinaci s nižším nárokem na výpočetní kapacitu. Dalším důvodem je možnost užití tohoto řešení pro různě vzdálené body. V případě, že body jsou od sebe více vzdálené, tak velikost konkrétního shluku může být menší a lze předpokládat menší počet vstupních vrcholů. Naopak při velkém množství vrcholů lze očekávat, že velikost shluků je větší. A jednotlivé shluky tak opět vytvoří menší počet vrcholů, na kterou lze teoreticky použít i metodu hrubé síly a vybrat nejideálnější permutaci.

Celý algoritmus je logicky rozdělen do dvou částí, a to vytvoření shluků jednotlivých bodů a následné nalezení nejkratší cesty mezi jednotlivými shluky. Vstupem

je množina vrcholů, kdy na první pozici množiny se nachází pivot, který následně slouží jako počáteční a zároveň koncový bod.

7.2.1 Nalezení shluků

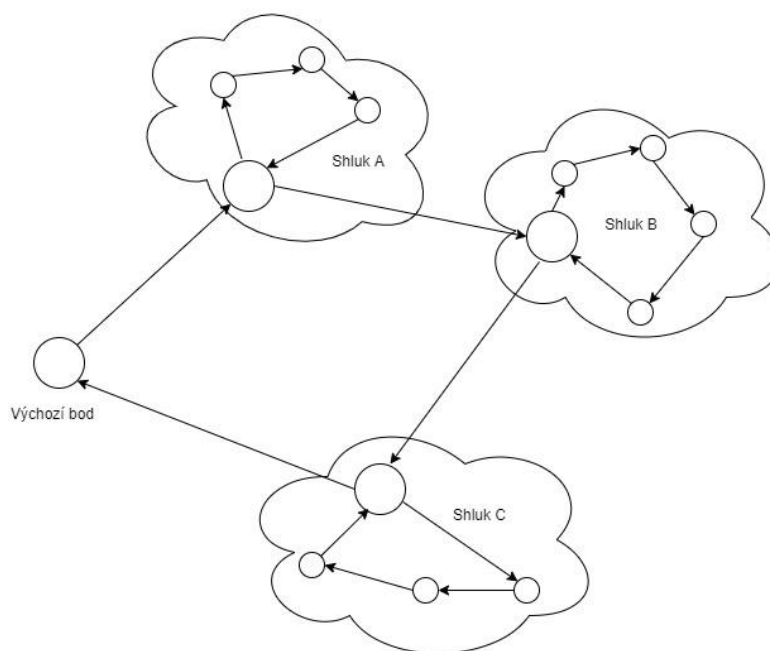
Každý vrchol, který je součástí vstupní množiny, obsahuje informaci o své poloze na mapě světa. Za pomoci této hodnoty tak nejen určíme vzdálenost mezi všemi vrcholy, ale zároveň máme představu o tom, zda jsou některé body dostatečně blízko u sebe, aby byly přidány do jednotlivých shluků. Takto vytvořené shluky jsou reprezentovány jako podmnožina vstupních vrcholů s vlastním pivotem, který se nachází nejbližší k počátku. Pro komplexnost řešení lze v rámci vstupní množiny požadovat parametr, který je rozhodovacím kritériem, zda daný uzel přidat do shluku či nikoliv.

V teoretické úvaze lze tento krok přeskočit a pro úsporu času při výpočtu určení trasy jednotlivé vrcholy shlukovat náhodně a vytvořit tak shluky se stejnou velikostí. Náhodné seskupení sice ušetří výpočetní čas, ale tato úspora je vykoupena sníženou šancí na získání optimální trasy.

V rámci řešení je pro jednotlivé vytváření shluků využito k-means přístupu. Jedná se analytickou metodu, která na základě vstupní váhy shlukuje jednotlivé body do množin. Tento přístup lze aplikovat na lineární, plošná i prostorová data.

7.2.2 Nalezení nejkratší cesty mezi shluky

V druhé fázi algoritmu hledáme cluster, jehož pivot má nejkratší vzdálenost od počátku stejně jako je tomu na začátku v Dijkstrově algoritmu. Následně je cluster konkrétního pivotu samostatně zpracován. V úvahu připadá možnost opětovného shlukování, které by ovšem mohlo výpočetně zatížit celý výpočet. Lepší variantou je tedy užití metody nejbližších sousedů, případně užít hrubé síly, pokud je počet uzlů ve shluku dostatečně malý.



Obrázek 4 Diagram řešení výsledné trasy

Zdroj: vlastní nákres

Když je cluster obslužen a všechny vrcholy jsou navštíveny, je program vrácen zpět a následně je vyhledán další nejbližší pivot. V posledním kroku je přidána do cesty hrana spojující poslední navštívený vrchol posledního zpracovávaného clusteru s počátečním bodem.

8 Implementace optimalizace trasy

Následující kapitola se již zaměřuje na reálné vytváření nástroje. V jednotlivých podkapitolách jsou ukázány použité technologie a principy. Mimo technologií a principů, je popsán proces samotné implementace. V závěru kapitoly je uveden scénář pro používání nástroje.

8.1 Použité technologie

Již dopředu je zjevné, že výsledný nástroj potřebuje získávat mapová data a informace o silniční síti. V rámci šetření bylo původně uvažováno o využití mapového API společnosti Google. Vzhledem k nedávné změně politiky poskytování a používání mapového API bylo od tohoto návrhu upuštěno. Pomyslným vítězným poskytovatelem mapových podkladů se tak stalo API české společnosti Seznam a.s. s jejich aplikací Mapy.cz, která je otevřená dalšímu použití včetně poskytnutí rozsáhlé a přehledné dokumentace.

Jednotlivé zastávky jsou reprezentovány v JSON formátu. Z jednotlivých záznamů pak získáváme data potřebná pro výpočet optimální trasy. Tyto záznamy jsou uloženy v MongoDB databázi. Databáze umožňuje vytvoření cloudového serveru, který má ovšem omezení. Uživatel nemá oprávnění některých akcí. Nejvíce problematickou je možnost vkládání nových dokumentů do externího cloudového úložiště. Databáze tak funguje pouze v režimu čtení.

Kostra navrhované aplikace je vytvořena pomocí Spring.io konfigurátoru, který umožňuje předdefinovat potřebné knihovny v závislosti na použití. Pro účely nástroje jsou vhodné ovladače pro správu webu a komunikaci s databází MongoDB.

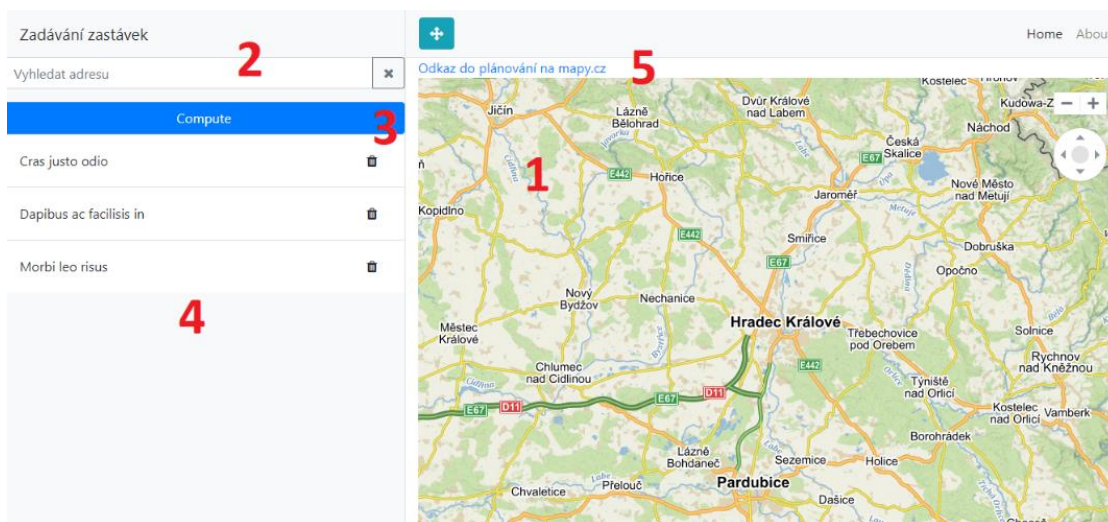
Pro API, které slouží k navigaci a ovládání aplikace, lze uvažovat o řešení stojící na JavaScriptu a knihovně Node.js případně využití jazyku Java, který je rovněž multiplatformní.

Pro stylizaci stránky a používání funkčních prvků na stránce jako jsou tlačítka, vstupy formulářů a jiné je použit framework Bootstrap poskytující širokou škálu předdefinovaných prvků s možností úpravy.

Nosný kód je vytvořen v JavaScriptu. V rámci přípravy na pozdější migraci do webové aplikace, je skript součástí konzolové aplikace, která ze vstupního souboru vytvoří trasu. Vzdálenosti mezi jednotlivými body jsou počítány na základě euklidovské, tedy vzdušné, vzdálenosti.

8.2 Wireframe aplikace

Následující kapitola slouží jako stručný popis scénáře pro používání optimalizačního nástroje.



Obrázek 5 Návrh vzhledu aplikace s označením funkčních prvků
Zdroj: Vlastní návrh

1. Hlavní mapa pro zobrazení výsledné trasy
2. Vstup pro zadávání zastávek
3. Tlačítko pro výpočet trasy zadaných bodů
4. Seznam zastávek (první v seznamu je počáteční bod, který je zároveň i cílový)
5. Odkaz pro přesměrování na mapy.cz

Scénář pro běžnou práci s nástrojem je následující. Uživatel pomocí vyhledávače najde jednotlivé zastávky, které se mu následně zobrazují v seznamu pod vyhledávačem. Jiným způsobem zadání bodu je vybrání bodu kliknutím do mapy. Následně stiskne tlačítko pro výpočet trasy, která je následně zobrazena v náhledové mapě, nad kterou je uvedený odkaz na samotnou aplikaci Mapy.cz, kde může být spuštěna samotná navigace. Uživatel musí mít na paměti, že nejdůležitější je výběr prvního bodu, který

odpovídá počátku trasy. Pro přívětivější použití lze uvažovat o možnosti vybrat dodatečně libovolný bod jako počáteční bez vazby na pořadí přidání.

Volba prvního bodu je důležitá zejména proto, že skript, který nástroj využívá, očekává výchozí bod na první pozici.

8.3 Výpočet trasy

V této podkapitole je čtenář seznámen se samotným výpočtem výsledné trasy za použití skriptu, který je součástí konzolové aplikace.

Vstupem pro zjištění trasy je JSON soubor obsahující zastávky včetně výchozího bodu a číslo odpovídající počtu shluků, do kterých jsou jednotlivé zastávky rozděleny. Zastávka ve formě JSON objektu obsahuje informaci o geografické poloze ve formátu řetězce sestávající se ze zeměpisné šířky a délky, adresy a pořadí vložení do seznamu, pro okamžité nalezení výchozí pozice, která se nachází na pozici v poli s indexem 0.

Implementace kopíruje samotný návrh výpočtu. Prvním krokem je rozdělení adres do shluků. Samotné rozdělení je pro výsledný výpočet trasy stěžejní a rozhoduje o vhodnosti konečného výsledku. Rozdělení do shluků využívá k-means cluster analýzy. Jednotlivé centroidy jsou v prvním kroku zvoleny rovnoměrně v poli a v následných krocích jsou upravovány podle dosažených výsledků.

Samotný výpočet trasy navazuje na vytvořené shluky. Nejprve jsou pro jednotlivé shluky určeny pivoty. Následně je vytvořena matice sousednosti výchozího bodu a pivotů všech shluků. Poté jsou vytvořeny matice sousednosti jednotlivých shluků. V tento moment je vytvářena trasa stejným postupem, jak je uvedeno v návrhu. Tedy vybrání nejbližšího pivota a následně obslužení shluku daného pivota.

Kompletní podoba skriptu se vzorovými daty je umístěna v online repozitáři na platformě GitHub: <https://github.com/dawissl/BP/tree/master/scheduler-skript>.


```

for (let i = 0; i < nodes.length; i++) {
  //dont add start node to clusters
  if (nodes[i].position === 0) continue;
  let oldDist = -1.0;
  let range;
  for (let j = 0; j < k; j++) {
    range = distance(centroids[j], nodes[i]);

    if (oldDist === -1.0) {
      oldDist = range;
      newClusterIndex = j;
    } else if (range <= oldDist) {
      oldDist = range;
      newClusterIndex = j;
    }
  }

  clusters[newClusterIndex].push(nodes[i]);
}

```

Obrázek 6 JavaScriptový kód s cyklem pro rozdělení bodů do clusteru podle volených centroidů

Zdroj: Výňatek z vlastního skriptu

9 Shrnutí výsledků

Jedním z cílů práce bylo za pomoci optimalizačního nástroje nalézt optimalizovanou trasu z referenčních dat. V tabulkách není uveden výchozí bod, který odpovídá adrese svozového depa společnosti PPL v blízkosti Hradce Králové. Depo se nachází na adrese Březhradská 199, Březhrad. Předpokládáme, že řidič se na konci své trasy opět vrací do výchozího bodu a v následujících tabulkách tedy tato adresa není uvedena.

9.1 Výsledná trasa

Tabulka 1 Referenční trasa řidiče

Pořadí zastávky	Adresa pro doručení
1	U Hřiště 61/6, Hradec Králové
2	Pohřebačka 128, Opatovice nad Labem
3	K. Čapka 319, Čeperka
4	Podůlšany 61, Podůlšany
5	Sedlice 44, Praskačka
6	Osice 63, Osice
7	Syrovátka 134, Syrovátka
8	Dobřenice 114, Dobřenice
9	Michnovka 12, Kratonohy
10	Kratonohy 26, Kratonohy
11	Homyle 69, Boharyně
12	Libčany 96, Libčany
13	Libčany 241, Libčany
14	Lhota pod Libčany 215, Lhota pod Libčany
15	Lhota pod Libčany 113, Lhota pod Libčany

Při tomto pořadí trasy řidič urazí trasu v délce 63,4 km a čas trasy odpovídá 1:35 h. Za předpokladu spotřeby vozu 8 l/100 km, a ceně pohonných hmot 30 Kč za litr je výsledný náklad po haléřovém zaokrouhlení roven 152 Kč.

V následujících třech tabulkách si ukážeme výslednou trasu vytvořenou za použití nástroje pro variantu rozdělení bodů do 4, 5 a 6 shluků.

Tabulka 2 Vypočtená trasa se čtyřmi shluky

Pořadí zastávky	Adresa pro doručení
1	U Hřiště 61/6, Hradec Králové
2	Pohřebačka 128, Opatovice nad Labem
3	K. Čapka 319, Čeperka
4	Sedlice 44, Praskačka
5	Podůlšany 61, Podůlšany
6	Libčany 241, Libčany
7	Libčany 96, Libčany
8	Lhota pod Libčany 113, Lhota pod Libčany
9	Lhota pod Libčany 215, Lhota pod Libčany
10	Osice 63, Osice
11	Syrovátka 134, Syrovátka
12	Homyle 69, Boharyně
13	Kratonohy 26, Kratonohy
14	Dobřenice 114, Dobřenice
15	Michnovka 12, Kratonohy

Trasa vytvořená za pomoci čtyř shluků je v součtu společně s výchozí a cílovou adresou dlouhá 93,2 km a řidič na této trase stráví téměř dvě hodiny.

Tabulka 3 Vypočtená trasa s pěti shluky

Pořadí zastávky	Adresa pro doručení
1	U Hřiště 61/6, Hradec Králové
2	Pohřebačka 128, Opatovice nad Labem
3	K. Čapka 319, Čeperka
4	Sedlice 44, Praskačka
5	Podůlšany 61, Podůlšany
6	Osice 63, Osice
7	Syrovátka 134, Syrovátka

8	Dobřenice 114, Dobřenice
9	Libčany 241, Libčany
10	Libčany 96, Libčany
11	Lhota pod Libčany 113, Lhota pod Libčany
12	Lhota pod Libčany 215, Lhota pod Libčany
13	Homyle 69, Boharyně
14	Kratonohy 26, Kratonohy
15	Michnovka 12, Kratonohy

Využitím pěti shluků se vzdálenost i čas oproti řešení se čtyřmi shluky snížila. Výsledný čas strávený na cestě vychází na 1:48 h s ujetými 85,6 km.

Tabulka 4 Vypočtená trasa s šesti shluky

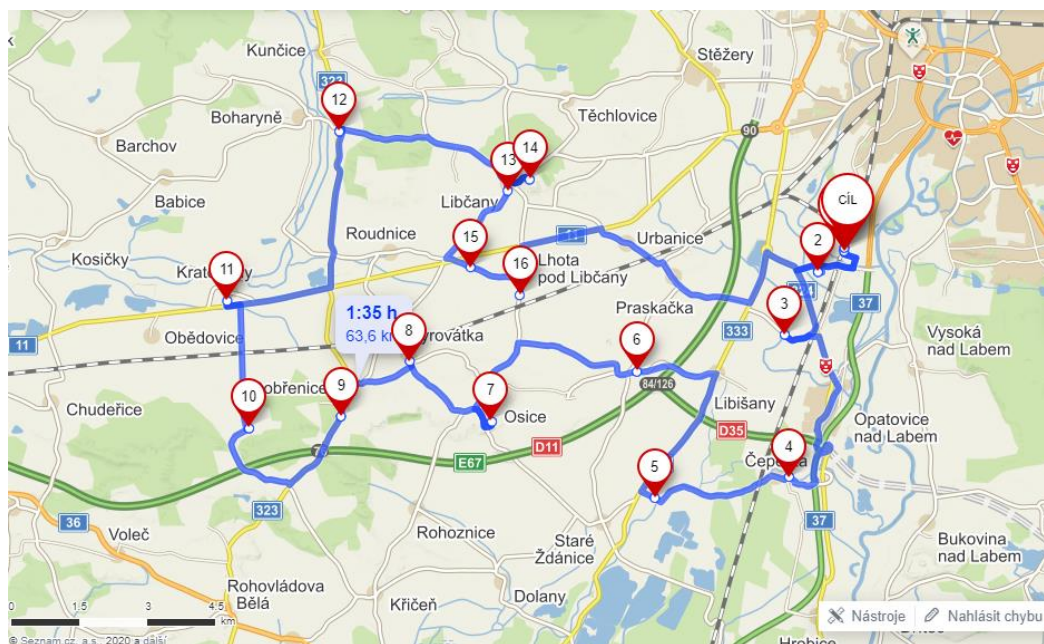
Pořadí zastávky	Adresa pro doručení
1	U Hřiště 61/6, Hradec Králové
2	Pohřebačka 128, Opatovice nad Labem
3	K. Čapka 319, Čeperka
4	Podůlšany 61, Podůlšany
5	Sedlice 44, Praskačka
6	Lhota pod Libčany 113, Lhota pod Libčany
7	Lhota pod Libčany 215, Lhota pod Libčany
8	Osice 63, Osice
9	Syrovátka 134, Syrovátka
10	Libčany 96, Libčany
11	Libčany 241, Libčany
12	Homyle 69, Boharyně
13	Kratonohy 26, Kratonohy
14	Dobřenice 114, Dobřenice
15	Michnovka 12, Kratonohy

Při šesti shlucích, do kterých jsou vstupní data rozdělena, lze očekávat, že shluky jsou tvořeny dvojicí nebo trojicí adres. Výsledná trasa přináší další úsporu času a méně ujetých kilometrů s výsledkem 78,9 km ujetých za 1:43 h.

Tabulka 5 Srovnání jednotlivých tras

Trasa	Vzdálenost (km)	Čas (h)
Referenční	63,4	1:35
4 shluky	93,2	1:59
5 shluků	85,6	1:48
6 shluků	78,9	1:43

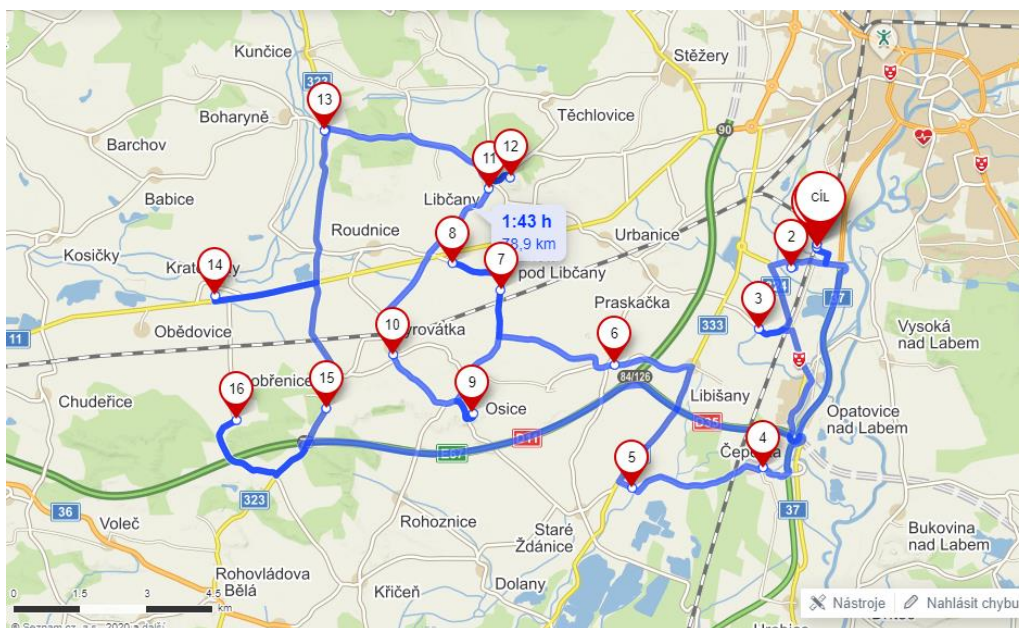
Z výsledků uvedených v tabulce 5 vidíme, že skutečná trasa, kterou řidič absolvoval, je z hlediska času i vzdálenosti nejvýhodnější. Jednotlivé trasy, které jsou porovnávány s trasou referenční, jsou získané za pomoci původního skriptu konzolové aplikace, které pro měření vzdáleností využívá euklidovské vzdálenosti. Z tabulky lze říct, že i přes využití euklidovské vzdálenosti, která může být zavádějící pro potřeby silniční sítě, již při pěti shlukách získáváme z časového hlediska podobnou trasu, jako je referenční.



Obrázek 7 Vizualizace referenční mapy ve webové aplikaci Mapy.cz

Zdroj: Mapy.cz

Nejlépeším z vytvořených výsledků se jeví trasa, která vytváří šest shluků ze vstupních dat. Při stejných hodnotách spotřeby a ceny pohonných hmot jako pro referenční trasu je výsledný náklad na trasu roven po haléřovém zaokrouhlení 189 Kč. Rozdíl nákladů mezi oběma trasami je tedy 37 Kč.



Obrázek 8 Vizualizace vypočtené trasy se šesti shluky ve webové aplikaci Mapy.cz
Zdroj: Mapy.cz

Z obrázku 8 je patrné, že řidič při absolvování vypočtené trasy využívá dálnici. Maximální povolená rychlost na dálnici je sice vyšší než na běžné komunikaci mezi obcemi, ale v tomto případě se jedná o náklad navíc v podobě dálniční známky. Při volbě trasy mimo dálnici se trasa časově prodlouží na 1:50 h, ale vzdáleností se trasa zkrátí na 77 km.

9.2 Výpočetní náročnost

V této podkapitole se pokusíme odhadnout výpočetní složitost daného řešení a porovnat jej se složitostmi dílčích algoritmů užitých pro výpočet trasy. Celková složitost výpočtu je odvozena z dílčích částí výpočtu, tedy vytvoření clusterů a následné hledání optimální trasy.

Pro určení asymptotické složitosti výpočtu hledání trasy ve vstupních datech je předpokládáno, že pro počet vstupních adres n , existuje k matic, k = počet clusterů, s rozměrem $M_k = (n - x_i) * (n - x_i)$ a matice počátku s pivoty o rozměru $(k + 1) * (k + 1)$. Z algoritmu je patrné, že nejbližšího pivota pro zpracování nalezneme nejhůře na indexu k . Pro odbavení příslušného shluku procházíme $(n - x_i)^2$ pozic. V součtu tedy

získáváme složitost $(n - x_i)^2 + (k + 1)^2$ a tedy výsledná složitost hledání spadá do kategorie $O(n^2)$.

K-means shlukování je obecně řazeno mezi NP-úplné úlohy. Dabura (Dabura, 2018) ve svém popisu implementace uvádí složitost analýzy jako kvadratickou. Lze předpokládat, že algoritmus stejného typu má rovněž kvadratickou složitost.

Celkově tedy získáváme složitost $2 * n^2$, a tedy asymptoticky $O(n^2)$ složitost.

10 Závěr

V rámci práce bylo cílem navrhnout single page aplikaci, která by sloužila jako nástroj pro optimalizaci trasy mezi jednotlivými zastávkami. Prvotní nástroj v podobě konzolové aplikace s možností zvolit počet shluků jednotlivých zastávek, je využit k vytvoření testovacích dat, která jsou použita k porovnání s výchozí trasou. Referenční trasa, se kterou jsou výsledky porovnávány, odpovídá odpolední rozvážce řidiče společnosti PPL. Ta má obvykle nižší počet balíčků a je tak vhodná pro účel práce a demonstraci funkčnosti navrženého řešení.

S přihlédnutím na využití vzdušných vzdáleností mezi zastávkami jsou vytvořené cesty smysluplné. Jejich použitelnost je ovlivněna zejména počtem shluků. Samotné užívání přesto vyžaduje jistou znalost cílových adres, a jejich okolí, pro ideální volbu počtu shluků. Nižší počet shluků může způsobit, že se v jedné množině sjednotí body, které výrazně ovlivní výsledek. Naopak velký počet shluků zvyšuje výpočetní náročnost a vede k řešení hrubou silou.

Samotný skript lze použít pro jiný nástroj, který by pouze zpracovával vstupní data. Data by mohla být získávána pomocí čtečky čárových kódů, které jsou umístěné na jednotlivých zásilkách. Tento přístup by výrazně napomohl k automatizaci při vytváření trasy. Řidič při přejímání zásilek jednotlivé balíky skenuje, aby mohly být spárovány se záznamy v databázi. Nevznikala by tak nadbytečná práce při vytváření trasy.

Při transformaci původní konzolové aplikace na plnohodnotnou aplikaci autor práce narazil na několik problémů, které v konečném důsledku znamenaly její nedokončení. Mezi důvody lze zařadit využití databáze MongoDB, která neumožňuje nad cloudovými daty vykonávat operaci zápisu z lokálního zařízení a použitelnost API pro aplikaci Mapy.cz ve vlastním kódu. Veškerý výpočet trasy je tak nutno řešit až po načtení stránky a stisknutí tlačítka. Problematické je samotné získání vzdáleností mezi body, respektive velký počet volání. Pro vzorových 16 zastávek včetně výchozího bodu je v konfiguraci se 6 shluky uskutečněno 72 volání na zjištění vzdálenosti při konstrukci clusterů, celkový počet volání by se tak při stejné konfiguraci zvedl na 177. Pro každou dvojici zastávek je třeba získat vzdálenost v obou směrech, a to z důvodu silničních

omezení. Příkladem jsou například jednosměrné ulice nebo přikázané směry jízdy. Počet volání může být ještě vyšší, pokud využijeme silniční vzdálenost i pro vytváření shluků adres.

Za předpokladu nahrazení a zprovoznění API pro určení vzdálenosti jiným než euklidovským řešením, bychom získávali přesnější data. Nutno ovšem podotknout, že i dosavadní výsledky užitím euklidovských vzdálenosti přinesly uspokojivé výsledky.

11 Seznam použité literatury

- [1] VEENSTRA, M, Roodbergen, *KJ, Vis*, IFA & Coelho, L 2017, 'The pickup and delivery traveling salesman problem with handling costs', *European Journal of Operational Research*, vol. 257, no. 1, pp. 118-132.
- [2] PÉREZ LECHUGA, *G.*, Optimal logistics strategy to distribute medicines in clinics and hospitals. *J.Math.Industry* 8, 2 (2018). <https://doi.org/10.1186/s13362-018-0044-5>.
- [3] BAKEŠOVÁ, Miroslava a Vladimír KŘEŠŤAN. *Základy logistiky*. Jihlava: Vysoká škola polytechnická Jihlava, 2008. ISBN 978-80-87035-08-5.
- [4] SVOBODA, Vladimír. *Dopravní logistika*. Praha: Vydavatelství ČVUT, 2004, 115 s. ISBN 80-01-02914-X.
- [5] CORMEN, Thomas H. *Introduction to algorithms*. 3rd ed. Cambridge, Mass.: MIT Press, c2009. ISBN 9780262533058.
- [6] HLATKÁ, M. (2017). APLIKACE METODY OPERAČNÍHO VÝZKUMU PRO ÚLOHU OPTIMALIZACE ROZVOZOVÉ TRASY. *Mladá Veda*, 5(8), 68-75
- [7] PINTEA, C., Pop, P.C. & Chira, C. The generalized traveling salesman problem solved with ant algorithms. *Complex Adapt Syst Model* 5, 8 (2017). <https://doi.org/10.1186/s40294-017-0048-9>
- [8] MATOUŠEK, Jiří. *Kapitoly z diskrétní matematiky*. Vyd. 2., opr., (V nakl. Karolinum 1.). Praha: Karolinum, 2000. ISBN 80-246-0084-6.
- [9] MILKOVÁ, Eva. *Teorie grafů a grafové algoritmy*. Hradec Králové: Gaudeamus, 2013. ISBN 978-80-7435-267-6.
- [10] HARTIGAN, JOHN A. *Clustering Algorithms (Probability & Mathematical Statistics)*. New York: John Wiley, 1975. ISBN 0-471-35645-X.
- [11] Objem balíků před Vánoce stoupá až o 100 %. Kurýři PPL v prosinci rozvezou více než 200 tisíc zásilek denně. PPL CZ s.r.o. [online]. Říčany, c2009-2020 [cit. 2020-03-20]. Dostupné z: https://www.ppl.cz/main.aspx?cls=art&art_id=590
- [12] *Mapy* [online]. Praha, c2020 [cit. 2020-04-24]. Dostupné z: <https://play.google.com/store/apps/details?id=com.google.android.apps.maps>
- [13] *Mapy.cz* [online]. Praha, c2020 [cit. 2020-04-24]. Dostupné z: <https://play.google.com/store/apps/details?id=cz.seznam.mapy>

- [14] E.ON MOBILNÍ PORTÁL ÚDRŽBY [online]. Praha 7, c2020 [cit. 2020-04-24]. Dostupné z: <https://www.eman.cz/nase-prace-detail/eon-mpud/>
- [15] BLACK, Paul E. Chinese postman problem. Dictionary of Algorithms and Data Structures [online]. CRC Press, 1999 [cit. 2020-04-24]. Dostupné z: <https://xlinux.nist.gov/dads/HTML/chinesePostman.html>
- [16] SANTIAGO, Jonathan. Searching for Route Dispatch Software? Before You Buy, Here's What to Consider [online]. Huntingdon, c2020 [cit. 2020-04-24]. Dostupné z: <https://blog.getcircuit.com/route-dispatch-software/>
- [17] VANDERBEI, Robert J. Linear Programming: Foundations and Extensions [online]. 4th ed. New York: Springer, Boston, MA, 2014 [cit. 2020-04-26]. ISBN 978-1-4614-7630-6. Dostupné z: <https://doi.org/10.1007/978-1-4614-7630-6>
- [18] Dokumentace API Mapy.cz. JSDOC: SeznamTříd [online]. [cit. 2020-04-26]. Dostupné z: <https://api.mapy.cz/doc/index.html>
- [19] CENTRUM EP. Sociodemografická analýza města Hradce Králové [online]. Hradec Králové, 2013, s. 272 [cit. 2020-04-26]. Dostupné z: https://www.hradeckralove.org/assets/File.ashx?id_org=4687&id_dokumenty=63115
- [20] DABBURA, Imad. K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks. Towards Data Science [online]. 2018 [cit. 2020-04-26]. Dostupné z: <https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a>

Zadání bakalářské práce

Autor: David Sládeček

Studium: I1700139

Studijní program: B1802 Aplikovaná informatika

Studijní obor: Aplikovaná informatika

Název bakalářské práce: **Optimalizace plánované trasy s více body**

Název bakalářské práce AJ: Optimising a scheduled route with multiple points

Cíl, metody, literatura, předpoklady:

Cíl: Seznámit se s problematikou a principy grafových algoritmů pro hledání nejkratších cest a implementovat single page aplikaci pro silniční spediční služby, která tyto principy uplatňuje.

Rámcová osnova:

Úvod

Prozkoumání problematiky silniční spedice, využití grafových algoritmů a jejich možné optimalizace pro návrh trasy

Vytvoření přehledu současných řešení využívající grafových algoritmů

Návrh implementace pro silniční spediční službu využívající grafových algoritmů pro hledání nejkratší cesty

Implementace navrhovaného řešení

Návrh a provedení testování a srovnání se současnými využívanými metodami

Závěr a zhodnocení dosažených výsledků

1. Hlatká, M. (2017). APLIKACE METODY OPERAČNÍHO VÝZKUMU PRO ÚLOHU OPTIMALIZACE ROZVOZOVÉ TRASY. *Mladá Věda*, 5 (8), 68-75.
2. Jeromin, B. & Körner, F. *Mathematical Programming* (1987) 37: 254.
3. M. Veenstra, K.J. Roodbergen, I.F. Vis, L.C. Coelho *The pickup and delivery traveling salesman problem with handling costs, European Journal of Operational Research*, 257 (1) (2017), 118-132

scholar.google.com

Garantující pracoviště: Katedra informatiky a kvantitativních metod,
Fakulta informatiky a managementu

Vedoucí práce: Ing. Milan Košťák

Datum zadání závěrečné práce: 14.1.2018