



**ČESKÉ VYSOKÉ  
UČENÍ TECHNICKÉ  
V PRAZE**

**F3**

**Fakulta elektrotechnická  
Katedra kybernetiky**

**Bakalářská práce**

# **Obohacování grafů dopravních sítí o kontextové informace z OpenStreetMaps**

**Petr Michalík**

**Studijní program: Otevřená informatika**

**Obor: Informatika a počítačové vědy**

**Květen 2020**

**Vedoucí práce: doc. Ing. Michal Jakob, Ph.D.**



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Michalík** Jméno: **Petr** Osobní číslo: **474564**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra kybernetiky**  
Studijní program: **Otevřená informatika**  
Studijní obor: **Informatika a počítačové vědy**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Obohacování grafů dopravních sítí o kontextové informace z OpenStreetMaps**

Název bakalářské práce anglicky:

**Enriching Transport Network Graphs with Contextual Information from OpenStreetMaps**

Pokyny pro vypracování:

Graf dopravní sítě (dále dopravní graf) je datová struktura používána zejména pro plánování tras v navigačních systémech. Dopravní graf kromě topologie sítě typicky zachycuje i vlastnosti jednotlivých úseků (např. kvalitu povrchu). Graf ale typicky opomíjí prostorový kontext jednotlivých úseků (např. zda-li daný chodník vede přes park nebo podél rušné silnice), což jsou přitom informace, které jsou pro řadu úloh na dopravními sítěmi důležité (mohou např. významně ovlivnit výběr pěší nebo cyklistické trasy). Cílem práce je vytvoření nástroje na automatické obohacování dopravních grafů o kontextové informace z mapových podkladů s primárním účelem využití v navigačních systémech.

Při řešení problému postupujte následovně:

1. Seznamte se s mapovými podklady OpenStreetMap a jejich strukturou.
2. Prostudujte existující přístupy obohacování dopravních grafů o kontextové informace.
3. Navrhněte nástroj na automatické obohacování dopravního grafu o kontextové informace v okolí jeho hran a uzlů.
4. Implementujte navržený nástroj pro získávání kontextových informací z mapových podkladů v OpenStreetMaps.
5. Vyhodnoťte funkčnost navrženého nástroje a diskutujte výsledky.

Seznam doporučené literatury:

- [1] M. Raubal, M., and S. Winter. 2002. Enriching wayfinding instructions with local landmarks. In International conference on geographic information science. Berlin, pp. 243-259.
- [2] N. Runge, P. Samsonov, D. Degraen, and J. Schöning. 2016. No more Autobahn! Scenic Route Generation Using Google Street View. In Proceedings of the 21st International Conference on Intelligent User Interfaces (IUI '16). Association for Computing Machinery, New York, NY, USA, pp. 147-151.
- [3] H. Hochmair and G. Navratil. 2008. Computation of scenic routes in street networks. In Geospatial Crossroads@ GI\_Forum'08: Proceedings of the Geoinformatics Forum Salzburg, pp. 124-133.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**doc. Ing. Michal Jakob, Ph.D., centrum umělé inteligence FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **10.01.2020**

Termín odevzdání bakalářské práce: **22.05.2020**

Platnost zadání bakalářské práce: **30.09.2021**

doc. Ing. Michal Jakob, Ph.D.  
podpis vedoucí(ho) práce

doc. Ing. Tomáš Svoboda, Ph.D.  
podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.  
podpis děkana(ky)

### III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.  
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta



## Poděkování / Prohlášení

Chtěl bych poděkovat Ing. Pavolu Žileckému za přínosné konzultace, pravidelnou podporu a cenné a zkušené rady týkající se mé bakalářské práce.

Dále bych chtěl poděkovat svému vedoucímu práce doc. Ing. Michalu Jakobovi, Ph.D., který mi téma této práce navrhl, téma se mnou prodiskutoval a motivoval mě k tvorbě této práce.

Rád bych také poděkoval své rodině za podporu nejenom během tvorby této práce.

Tato práce závisí na otevřených datech OpenStreetMap, proto bych chtěl závěrem poděkovat i všem přispěvatelům OpenStreetMap.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 20. května 2020

.....

## Abstrakt / Abstract

Většina navigačních služeb dokáže plánovat nejkratší nebo nejrychlejší trasy. Tyto trasy ovšem nemusí být vhodné pro chodce nebo cyklisty, kteří spíše preferují cesty vedoucí parkem nebo podél vody za cenu, že cesta vede mírnou oklikou. Cílem této práce bylo vytvoření nástroje pro automatické obohacování cest v grafu dopravní sítě o informace, které by vypovídaly o kvalitě cesty z hlediska jejího okolí. Nástroj získá všechny objekty v okolí dané cesty a každou cestu následně ohodnotí podle předem určených kategorií (zeleň, voda, rušné silnice). Tyto výsledky jsou následně vizualizovány v interaktivní webové aplikaci. Zároveň byly porovnány výsledné hodnoty pro několik vybraných cest oproti ručně určeným očekávaným hodnotám. Podle vizualizace i porovnání hodnot jsme došli k závěru, že navržený nástroj funguje pro většinu případů podle očekávání. Výsledek této práce umožňuje hlouběji analyzovat kvality cest nebo navázat na tuto práci plánovačem tras vhodných nejenom pro chodce a cyklisty.

**Klíčová slova:** OpenStreetMap, obohacení, cesta, trasa, zeleň, voda, silnice, okolí

Conventional navigation can find the shortest or fastest route. Those routes might not be suitable for pedestrians or cyclists that might prefer ways leading through a park or by the water at the cost of a slight detour. This work aims to create a tool for automatic enrichment of every way in a transport network graph with such data that could convey information about nearby objects. The tool collects nearby objects around every way and then evaluates every way for each of the predefined categories (greenery, water, busy roads). The results were visualized in an interactive web application, and result values for a few selected ways were compared with predefined expected values. Based on this visualization and this comparison, we conclude that this method works in most cases as expected. The result of this work can be used for a more in-depth analysis of road networks or planning routes more suitable for both pedestrians and cyclists.

**Keywords:** OpenStreetMap, enrichment, way, route, greenery, water, road, surroundings

**Title translation:** Enriching Transport Network Graphs with Contextual Information from OpenStreetMaps

# Obsah /

<b>1 Úvod</b> .....	1
1.1 Motivace .....	1
1.2 Cíl .....	2
<b>2 Přehled literatury</b> .....	3
2.1 Mapové podklady .....	3
2.2 Obrázky .....	4
2.3 Lidé .....	5
<b>3 Specifikace problému</b> .....	6
3.1 Mapové podklady .....	6
3.2 Graf dopravní sítě .....	7
3.3 Obohacení grafu .....	7
3.3.1 Okolí .....	7
3.3.2 Kategorie .....	7
3.4 Výsledek obohacení grafu .....	8
<b>4 Řešení problému</b> .....	9
4.1 Hodnoty hran .....	9
4.1.1 Funkce pro ohodnocení hrany .....	9
4.1.2 Viditelnost .....	9
4.1.3 Strany .....	10
4.2 Body zájmu .....	10
<b>5 Implementace</b> .....	11
5.1 Architektura .....	11
5.2 OpenStreetMap .....	11
5.3 Výběr cest .....	12
5.4 Algoritmy a knihovny .....	13
5.5 Postup .....	15
5.6 Kategorie .....	16
5.6.1 Rušné silnice .....	16
5.6.2 Zeleň .....	16
5.6.3 Vodní plochy .....	17
5.6.4 Body zájmu .....	17
5.7 Dělení cest .....	18
5.8 Překážky .....	19
5.9 Druhy cest .....	20
5.10 Formát dat .....	21
5.11 Vizualizace dat .....	21
5.11.1 Webová aplikace .....	22
<b>6 Vyhodnocení</b> .....	24
6.1 Velikost okolí .....	24
6.2 Testovací scénáře .....	24
6.3 Výsledky .....	25
6.3.1 Zeleň .....	26
6.3.2 Vodní plochy .....	26
6.3.3 Rušné silnice .....	27
6.3.4 Překážky .....	28
6.3.5 Navigační instrukce .....	28
6.4 Diskuze .....	29
<b>7 Závěr</b> .....	30
7.1 Další postup .....	30
<b>Literatura</b> .....	32
<b>A Snímky</b> .....	35
<b>B Testovací případy</b> .....	37
<b>C Schéma Java tříd</b> .....	39

## Tabulky / Obrázky

<b>5.1.</b> OSM tagy silnic .....	17	<b>5.1.</b> Architektura řešení.....	11
<b>5.2.</b> OSM tagy zeleně .....	17	<b>5.2.</b> Vybrané cesty .....	13
<b>5.3.</b> OSM tagy vodních ploch .....	17	<b>5.3.</b> Strany cesty .....	20
<b>5.4.</b> OSM tagy bodů zájmu.....	18	<b>5.4.</b> Část GeoJSON souboru .....	22
<b>5.5.</b> OSM tagy překážek .....	19	<b>5.5.</b> Webová aplikace .....	23
<b>6.1.</b> Velikosti okolí .....	24	<b>6.1.</b> Vizualizace velikostí okolí.....	25
<b>6.2.</b> Test hodnot pro zeleň.....	26	<b>A.1.</b> Rušné silnice.....	35
<b>6.3.</b> Test hodnot pro vodu.....	27	<b>A.2.</b> Zeleň .....	35
<b>6.4.</b> Test hodnot pro silnice.....	27	<b>A.3.</b> Voda .....	36
<b>6.5.</b> Test hodnot pro překážky .....	28	<b>C.4.</b> Schéma aplikace.....	39

# Kapitola 1

## Úvod

### 1.1 Motivace

Běžné navigace jsou schopné nalézt nejrychlejší nebo nejkratší cestu, která nemusí být pro chodce vždy vhodná nebo bezpečná. Řešením tedy může být poskytnutí plánovacímu algoritmu dodatečnou informací o okolí libovolného bodu nebo cesty.

Podle článku *Enriching Wayfinding Instructions with Local Landmarks*[1] lze obohacený graf využít také k vytváření navigačních instrukcí, které jsou blíže lidské komunikaci a vnímání. Lidé se často orientují podle okolních budov nebo jiných záchytných bodů. Navigační instrukce pak může vypadat například jako „U Staroměstské radnice pokračujte rovně k paláci Kinských“ místo běžné instrukce „Jděte rovně 100 metrů“.

Dalším důvodem, proč je důležité se zabývat plánováním vhodnějších pěších tras, je výsledek práce *Green streetscape and walking* [2], podle kterého velké otevřené prostory se zelení nebo vodními plochami (např. pláže, parky, stromořadí, křižovatky, ...) pozitivně ovlivňují průměrnou denní dobu chůze. Naopak náměstí nebo široké ulice zkracují denní dobu chůze. Tyto závěry byly učiněny na základě statistiky z mobilních měření od 127 dobrovolníků v Barceloně. Chůze má pozitivní účinky na fyzické zdraví člověka, snižuje úroveň deprese nebo i snižuje porodní rizika. Tyto závěry potvrzuje také TED-Ed video *What happens if you cut down all of a city's trees?* [3], které se zabývá významností zeleně ve spojení s městy.

Mapové podklady jsou data, která tvoří dnešní digitální mapy, nejvýznamnějšími poskytovateli jsou Google Mapy, Seznam Mapy nebo OpenStreetMap<sup>1</sup>. „OpenStreetMap tvoří komunita uživatelů, kteří přidávají a udržují data o silnicích, cestách, kavárnách, železničních stanicích a mnohém dalším po celém světě.“[4] Protože se v mapových podkladech OpenStreetMap nachází téměř všechny silnice, chodníky, cesty a ostatní pozemní komunikace, lze data z OpenStreetMap využít k vytvoření grafu dopravní sítě.

Dopravní síť je tvořena dvojicí množin dopravních uzlů (křižovatek) a dopravních cest (komunikací). Graf dopravní sítě je reprezentace dopravní sítě pomocí orientovaného grafu. Přesná specifikace je uvedena v kapitole 3.3.

Většina mapových podkladů obsahuje informace o jednotlivých objektech (cestách, zastávkách, obchodech) bez informace o jejich okolí. Pokud se tedy nacházíme v libovolném uzlu, známe jeho přesnou geografickou pozici, všechny informace o tomto uzlu (povrch, třída apod.), ale neznáme nic o objektech, které se nachází v jeho blízkém okolí. Lze tedy získat informaci, že se nacházíme v uzlu, který reprezentuje rozcestí, ale nevíme, zda se poblíž nachází hlučné dálnice nebo městský park. Z tohoto důvodu je dobré znát nejen informace o daném uzlu ale i nějakou informaci o okolí. Touto informací může být např. množství zeleně v okolí nebo kvalita ovzduší v daném místě.

Daty, kterým lze graf dopravní sítě obohatit, mohou být například panoramatické nebo letecké snímky okolí nebo data z OpenStreetMap. Pokud tedy plánovací algorit-

<sup>1</sup> <https://www.openstreetmap.org/>

mus dostane dodatečné informace o okolí, může učinit lepší rozhodnutí a nalézt vhodnější trasu. Vhodnější trasou může být například cesta vedoucí parkem namísto cesty kolem rušné dálnice nebo případně lze vytvořit trasu podle preferencí uživatele. Přestože tedy takové cesty budou zpravidla delší než nejkratší cesta, mohou být výrazně lepší z hlediska charakteru okolí cesty.

## 1.2 Cíl

Cílem této práce je vytvořit nástroj pro automatické obohacování hran v grafu dopravní sítě, který pro každou hranu uloží váhu každé kategorie (zeleň, voda, rušné silnice), a výsledky tohoto nástroje bude možné využít při plánování pěších nebo cyklistických tras. Čím vyšší je váha (hodnota) dané kategorie, tím vyšší zastoupení objektů se vyskytuje kolem cesty.

Prvním krokem je sestavení grafu dopravní sítě obsahující hrany, které jsou určeny nebo vhodné pro chodce nebo cyklisty (a naopak se vyhnout například dálnicím). Data z OpenStreetMap nemají grafovou strukturu, je tedy nutné tyto data přetvořit do podoby grafu dopravní sítě, kde uzlu odpovídá křižovatka cest a hraně odpovídá cesta z OpenStreetMap.

Dalším krokem je získání všech objektů v okolí dané hrany. Tato informace není v grafu dopravní sítě obsažena, je tedy nutné prohledat okolí a získat pouze ty objekty, které se nachází ve stanovené vzdálenosti od hrany. Tyto objekty mohou být body, hrany, polygony nebo skupina několika objektů (relace).

Získané objekty jsou následně využity pro určení množství zeleně, vodních ploch a rušných silnic v okolí dané hrany a tato informace je přidána ke každé hraně. Následně dojde k nalezení všech bodů (křižovatek), ve kterých dochází ke křížení dvou a více hran. V každé křižovatce se vybere jeden nejvýraznější orientační bod, který bude sloužit pro vytvoření obohacené navigační instrukce. Mezi každou dvojicí hran, které se navzájem kříží, vznikne textová navigační instrukce obsahující vybraný orientační bod.

Nakonec se výsledky této práce vizualizují a vyhodnotí se funkčnost navrženého nástroje v několika testovacích scénářích, které odpovídají zvoleným kategoriím.

## Kapitola 2

### Přehled literatury

Graf dopravní sítě lze obohatit o informace z různých datových zdrojů. Nejčastějším zdrojem dat jsou mapové podklady, které obsahují informace o většině okolních objektů. Dále je možné využít k obohacení grafu obrázky z míst poblíž cest, například panoramatické snímky Google Street View nebo databáze s fotografiemi (obsahující geografické souřadnice polohy). Informace pro obohacování lze také získávat od lidí. Těmito zdroji mohou být hodnocení různých míst podél cest, upozornění na možné nebezpečí v daném místě nebo zpracovaná data z elektrokardiogramu získaná v průběhu cesty.

Tyto přístupy se navzájem nevyklučují, je tedy možné použít všechny způsoby obohacování grafu dopravní sítě současně.

### 2.1 Mapové podklady

V článku *A System for Generating Customized Pleasant Pedestrian Routes Based on OpenStreetMap Data* [5] jsou využívána data z OpenStreetMap pro vytvoření nástroje, který pro každou cestu přidává informaci o charakteru okolí v okruhu 50 metrů. Těmito informacemi (kategoriemi) jsou množství zeleně, počet veřejných míst (obchody, bary, kavárny, restaurace, ...) a hlučnost dané cesty (podle okolních silnic). Součástí práce je i plánovač trasy, který vytváří trasy s ohledem na délku i kvalitu cest. Uživatel pak může zadat váhy jednotlivým kategoriím a dostat trasu, která není nejkratší ale pravděpodobně příjemnější. V rámci výsledků vyhodnocení plánovače většina lidí tuto funkci ocenila, ovšem někteří účastníci postrádali další kategorie nebo vizualizaci jednotlivých kategorií. Mezi navrhované kategorie, které by lidé uvítali, patří např. bezpečnost (v noci se chci vyhnout neosvětleným místům), šířka chodníku nebo sluneční stín.

V článku *Computation of scenic routes in street networks* (Hochmair et al.) [6] se zaměřují na plánování tras podle atraktivních míst, například parky nebo vodní plochy. Cena každé cesty, označená jako vnímaná délka cesty, je určena jako geometrická délka cesty  $d$  vynásobena faktorem atraktivity  $f$  ( $0 \leq f < 1$ ). Tímto se tedy problém převede na hledání cesty v grafu s nejmenší cenou. Kolem každého atraktivního místa se vytvoří jedna nebo více kruhových vrstev, které ovlivňují atraktivitu okolních cest. Cesty není vhodné dělit na hranicích kruhových vrstev, protože se tím znatelně zvýší počet hran grafu, čímž se zvýší náročnost hledání cesty s nejmenší cenou. Řešením je přiřazovat ceny celým nerozděleným segmentům. Nejvhodnější volbou je tedy 1 kruhová vrstva (více kruhových vrstev neposkytuje znatelné zlepšení) s poloměrem 25 metrů.

Pěší trasy se nemusí plánovat jenom s ohledem na pohodlí chodce, ale je nutné zvážit i bezpečnost vybrané trasy, tento pohled na plánování cest je popsán v článku *A computational model of pedestrian road safety* [7]. V tomto modelu jsou pěší cesty obohacovány o informaci možného rizika dopravní nehody. Faktory jsou typy přechodů (světelná křižovatka, přechod s tlačítkem pro chodce, řízené dopravním policistou nebo zebra) a typ cesty včetně rizika přecházení této cesty (riziko přecházení dálnice je mnohem vyšší než u běžné silnice). Každá cesta je tak ohodnocena podle její bezpečnosti.

Dále lze graf dopravní sítě obohatit také o výškovou informaci. Tímto problémem se zabývá článek *Finding outdoor boundaries for 3D space-based navigation* [8]. Zde se pro navigaci využívá 3D prostor, který je vytvořen z 2D map a digitálního terénního modelu pro výškovou informaci. Tento přístup umožňuje plánovat cesty vedoucí pod mostem nebo skrze náměstí, trávníky, parkoviště a jiné souvislé oblasti a vyhnout se tak cestě, která zbytečně vede oklikou.

Informace o okolí cesty lze využít také k obohacení navigačních instrukcí. Tímto přístupem se zabývá článek *Enriching Wayfinding Instructions with Local Landmarks* [1]. V této práci jsou hledány výrazné orientační body kolem cest. Kvalita těchto orientačních bodů je určena vizuální přitažlivostí (velikost, tvar, barva), kulturní nebo historickou významností a strukturální významností (křižovatka, most, hranice čtvrti). Všem vlastnostem jsou přiřazeny váhy, lze tak jednoznačně vybrat nejvýraznější orientační bod z okolí cesty. Tento orientační bod je poté využit při vytváření bohatší navigační instrukce, která je blíže lidské komunikaci.

Existuje služba Walk Score<sup>1</sup>, která se nezabývá plánováním tras pro chodce, ale hodnotí městské oblasti podle toho, jak daleko jsou nejbližší restaurace, obchody, parky, školy atd. a zda je potřeba pro vyřízení všech běžných záležitostí (nákup apod.) používat auto, kolo nebo městskou hromadnou dopravu anebo lze kamkoliv dojít pouze pěšky. Walk Score se proto snaží propagovat místa, kde je vše dostupné jen pěšky.

## 2.2 Obrázky

Zdrojem dat pro obohacování dopravní sítě mohou být také snímky z Google Street View. V článku *No more Autobahn!* [9] jsou tyto snímky využívány ke klasifikaci dálnic podle jejich vizuálního okolí. Automatický systém *Autobahn* rozdělí mapové podklady z vybrané oblasti do buněk o velikosti 1 km<sup>2</sup>. V každé buňce poté vybere cestu nejbližší středu buňky a zde vybere odpovídající snímek z Google Street View. Každý snímek je následně klasifikován pomocí konvoluční neuronové sítě (CNN) podle 5 kategorií: sightseeing (např. památky), příroda (např. lesy, údolí), zemědělská pole, voda (např. oceán, přístav) a hory. Tato metoda tedy řeší i případy, kdy například dálnice vede podél vodní plochy a výhled z ní blokují domy, stromy nebo tunel.

V závěru 24 dobrovolníků ve věku od 18 do 49 let hodnotilo pomocí Likertovy škály tyto trasy i trasy generované z článku Hochmaira et al. [6]. Přestože účastníci dotazníku vyjádřili preference pro trasy vygenerované pomocí systému *Autobahn*, rozdíl mezi oběma trasami nebyl příliš významný. Zároveň výsledky ukázaly preferenci vylepšených tras oproti běžným nejkratším trasám pro oba systémy, lze tak usoudit, že oba systémy splnily svůj záměr.

Další možný přístup je popsán v článku *The shortest path to happiness: recommending beautiful, quiet, and happy routes in the city* [10], kde jsou pro doporučování příjemných tras využívány lidské emoce, které vzbudí místa na trasách v Londýně a Bostonu. Jednotlivá místa na trasách jsou ohodnocena podle lidského vnímání krásy, tichosti a podle toho, zda dané místo činí člověka šťastného. Jako zdroj těchto hodnocení byly použity fotografie ze služby Flickr z míst, která obsahovala geografické souřadnice. Fotografie byly ohodnoceny podle počtu zhlédnutí, oblíbenosti, komentářů a tagů. Poté je možné uživateli nabídnout 3 různé trasy podle těchto daných kritérií, které jsou v průměru delší jen o 12% než nejkratší vypočtená cesta.

V článku *Mapping road safety features from streetview imagery: A deep learning approach* [11] jsou všechny dálnice obohaceny informacemi z hlediska bezpečnosti dané

<sup>1</sup> <https://www.walkscore.com/>



silnice pomocí panoramatických snímků z Google Street View. Bezpečnostními prvky jsou podélné čáry s akustickým efektem (*rumble strip*), betonové zábrany nebo kovová svodidla. Tyto bezpečnostní prvky jsou detekovány ve snímcích pomocí konvoluční neuronové sítě. Výsledkem je oklasifikovaná dopravní síť velkých dálnic v Alabamě s informací o bezpečnostních prvcích v jednotlivých úsecích dálnic.

## 2.3 Lidé

V článku *Knowledge extraction from crowdsourced data for the enrichment of road networks* [12] jsou zdrojem dat pro obohacení dopravní sítě uživatelská data z různých míst. Těmito zdroji jsou například data s geografickou informací ze stránek Foursquare, Yelp, Twitter, Flickr nebo záznamy projetých tras z OpenStreetMap nebo Endomondo. Dále jsou také analyzovány příspěvky z blogů nebo jiné texty, které obsahují například recenze různých míst. Pokud nějaké místo obsahuje spoustu recenzí a fotografií, pak se pravděpodobně jedná o zajímavé místo. Tyto informace jsou následně využity při tvorbě turistických tras a porovnány s profesionálními doporučenými trasami pro turisty. Výsledkem jsou trasy, které jsou navzájem velmi podobné.

Pro obohacování map lze alternativně použít senzorní měření, např. ECG (elektrokardiogram), EDA (elektrická aktivita kůže), srdeční tep nebo teplotu. V práci *Urban Emotions and Cycling Experience – enriching traffic planning for cyclists with human sensor data* [13] bylo dvanáct dobrovolníků z Bostonu monitorováno při jízdě na kole po městě pomocí ECG a EDA. Pokud tyto senzory zaznamenaly zvýšení stresové aktivity, došlo k zaznamenání příčiny. Mezi častými příčinami byla blízko projíždějící auta, dlouhé čekání u křižovatek, špatný stav silnic nebo naopak i kladné emoce, jako například tzv. „zelená vlna“.

# Kapitola 3

## Specifikace problému

V této práci jsou kromě dopravní sítě důležité také reálné objekty (stromy, parky, obchody apod.). Některé objekty lze reprezentovat jako body v prostoru (například stromy, zastávky nebo lavičky), jiné objekty lze reprezentovat jako úsečky v prostoru (například dopravní cesty, vodní toky, ploty nebo zdi). Složitější objekty lze poté reprezentovat pomocí mnohoúhelníků (polygonů), tyto objekty mohou být například budovy, parky, administrativní území, pole nebo louky.

### 3.1 Mapové podklady

**Definice 3.1.** Mapové podklady jsou tvořeny  $n$ -ticí  $W = (V, E, P, f, g)$ , kde

- $V$  je neprázdná konečná množina vrcholů (uzlů)
- $E$  je konečná množina hran  $e = (u, v)$ , kde  $u, v \in V$  jsou *krajní vrcholy* hrany  $e$ 
  - vrchol  $u$  je *počáteční vrchol* hrany  $e$  (značeno  $PV(e) = u$ )
  - vrchol  $v$  je *koncový vrchol* hrany  $e$  (značeno  $KV(e) = v$ )
  - jsou zakázány *orientované smyčky*, tj.  $u \neq v$
  - jsou zakázány také *paralelní hrany*, tj.

$$\forall e_1, e_2 : (PV(e_1) = PV(e_2) \wedge KV(e_1) = KV(e_2)) \Rightarrow e_1 = e_2$$

- $P$  je množina *polygonů* (mnohoúhelníků)

- cyklus je posloupnost vrcholů

$$v_1, e_1, v_2, e_2, \dots, v_{k-1}, e_{k-1}, v_k$$

taková, že pro každé  $i = 1, 2, \dots, k - 1$  platí  $v_i = PV(e_i)$ ,  $v_{i+1} = KV(e_i)$  a platí, že  $k > 1$  a vrcholy  $v_i$  se neopakují, s výjimkou  $v_1 = v_k$ .

- *polygon* je  $n$ -tice cyklů  $(A, \dots)$ ,  $A$  je *vnější* cyklus a všechny ostatní cykly jsou *vnitřní* cykly (reprezentující díry v polygonu)
  - vnější cyklus má body definované proti směru hodinových ručiček (v kladném směru, podle pravidla pravé ruky) a všechny ostatní cykly mají body definované po směru hodinových ručiček
- funkce  $f$  přiřazuje vrcholům  $v \in V$  body eukleidovské roviny
  - funkce  $g$  přiřazuje hranám  $e \in E$  úsečky, tj. pro hranu  $e$  je množina bodů v eukleidovské rovině

$$M = \{ \alpha f(PV(e)) + (1 - \alpha) f(KV(e)) \mid \alpha \in [0, 1] \}$$

Při odstranění omezení  $\alpha \in [0, 1]$  (a nahrazením  $\alpha \in \mathbb{R}$ ) bychom dostali přímku procházející krajními vrcholy hrany. [14][15]

**Definice 3.2.** Máme vrchol  $v \in V$  a polygon  $p \in P$ . Vrchol  $v \in V$  leží *uvnitř* polygonu  $p$  (značeno  $v \in p$ ), pokud bod  $f(v)$  z eukleidovské roviny leží uvnitř polygonu  $p$ . Bod  $f(v)$  leží uvnitř polygonu, jestliže počet průsečíků hraničních úseček všech cyklů  $e \in P$  s polopřímkou, která vede z bodu  $f(v)$  do libovolného směru, je liché číslo. Pokud je počet průsečíků sudé číslo, bod  $f(v)$  leží mimo polygon. [16]

## 3.2 Graf dopravní sítě

**Definice 3.3.** Dopravní síť je reprezentována *orientovaným grafem*  $G = (V_G, E_G)$ , kde

- $V_G$  je podmnožina vrcholů z mapových podkladů, tedy  $V_G \subseteq V$  (vrcholy odpovídají dopravním uzlům)
- $E_G$  je podmnožina hran z mapových podkladů  $E_G \subseteq E$  (hrany odpovídají dopravním cestám)

Potom graf  $G$  nazveme *grafem dopravní sítě*. Orientovaný graf je zvolen z důvodu, že v orientovaném grafu má každá hrana jednoznačně definován počáteční a koncový vrchol. Tato vlastnost je později důležitá při rozlišování levé a pravé strany hrany grafu a při vytváření navigačních instrukcí, kdy jsou tvořeny instrukce pro původní i opačný směr hrany.

## 3.3 Obohacení grafu

Obohacení grafu je problém definovaný jako  $n$ -tice  $P = (G, W, D, K)$ :

- $G = (V_G, E_G)$  je orientovaný graf dopravní sítě
- $W = (V, E, P, f, g)$  jsou mapové podklady
- $D$  je parametr určující velikost okolí každé hrany
- $K$  je trojice *kategorií*  $(k_z, k_v, k_s)$ , podle kterých jsou hrany hodnoceny
  - $k_z$  je kategorie zeleně (lesy, parky, trávníky)
  - $k_v$  je kategorie vodních ploch (řeky, potoky, rybníky)
  - $k_s$  je kategorie rušných silnic

### 3.3.1 Okolí

Pro získání všech objektů v okolí vybrané hrany je potřeba definovat způsob, jakým určit objekty patřící do okolí dané hrany. Kritériem výběru je minimální vzdálenost mezi hranou a objektem.

**Definice 3.4.** Máme objekty  $A \in W$  a  $B \in W$ . Body  $a \in A$ ,  $b \in B$  jsou *nejbližší body* mezi objekty  $A$  a  $B$ , jestliže platí  $(\forall x \in A)(\forall y \in B) : d(a, b) \leq d(x, y)$ , kde  $d$  je eukleidovská vzdálenost (norma) mezi 2 body.

**Definice 3.5.** *Minimální vzdálenost* mezi 2 objekty  $A, B$  je  $\text{dist}(A, B) = d(a, b)$ , kde body  $a \in A$  a  $b \in B$  jsou nejbližší body mezi objekty  $A$  a  $B$ .

Nyní lze tedy pomocí minimální vzdálenosti získat všechny objekty v okolí hrany.

**Definice 3.6.** *Okolí hrany*  $e \in E$  je množina  $M = \{x \in W \mid \text{dist}(e, x) \leq D\}$ , kde  $D$  je parametr určující velikost okolí. Z tohoto okolí lze následně určit kvalitu hrany ve vybraných kategoriích.

### 3.3.2 Kategorie

Kategorie lze vnímat jako funkce, které vyberou z okolí hrany  $M$  pouze objekty patřící do vybrané kategorie ( $k \in K : M \rightarrow N, N \subseteq M$ ).

K výběru podmnožiny objektů pro vybranou kategorii je potřeba o každém objektu znát dodatečnou informaci (například zda se jedná o les, silnici apod.). V případě OpenStreetMap je tato informace obsažena v tagách daného objektu (blíže popsáno v kapitole 5.6).

### 3.4 Výsledek obohacení grafu

Výsledek obohacení grafu  $P = (G, W, D, K)$  je dvojice  $S = (C, I)$ , kde

- množina  $C$  obsahuje váhy kategorií pro všechny hrany grafu  $G$ , informace o okolí hrany  $e_1 \in E_G$  je vektor vah  $c_1 = (h_1, \dots, h_n)$ , kde  $h_i \in [0, 1]$  odpovídá váze (hodnotě)  $i$ -té kategorie  $k_i \in K$
- množina  $I$  obsahuje bod zájmu  $i \in W$  pro každou dvojici hran  $(e_1, e_2)$ , které se protínají ve společném bodě (křižovatce)
  - $e_1, e_2 \in E_G$

# Kapitola 4

## Řešení problému

V grafu  $G = (V_G, E_G)$  jsou hrany  $e \in E$  navzájem nezávislé. Hodnotu  $c_i \in C$  každé hrany lze tak určit bez ohledu na ostatní hrany. Navigační instrukce  $i \in I$  je vytvořena pro dvojici hran  $(e_1, e_2)$ , je tedy nutné znát celou množinu  $E_G$ .

### 4.1 Hodnoty hran

Máme hranu  $e \in E_G$ , množinu objektů  $M$  v okolí hrany  $e$  a kategorii  $k_i \in K$ . Poté získáme pouze okolní objekty z dané kategorie  $N$  jako  $k_i(M) = N$ ,  $N \subseteq M$ .

Zbývá už jen vyrobit konkrétní funkci, která pro podmnožinu objektů  $N$  vrátí číslo  $h_i$  v rozsahu  $[0, 1]$ . Těchto způsobů výběru (ohodnocení podmnožiny) může být několik.

#### 4.1.1 Funkce pro ohodnocení hrany

V této práci je zvolen následující postup. Pokud pro danou kategorii objektů není podmnožina prázdná, hrana má hodnotu 1, jinak má hodnotu 0. Tento přístup ale nedokáže vůbec rozlišit relativní umístění objektu z dané kategorie vůči celé hraně. Pokud se nějaký objekt z kategorie nachází pouze poblíž jednoho z konců hrany, pak má hrana hodnotu 1, přestože z opačného konce daný objekt nemusí být vůbec viditelný. Řešením je hranu rozdělit na několik kratších úseků a každý úsek vyhodnotit zvlášť. Potom je nutné stanovit funkci pro ohodnocení celé hrany na základě těchto kratších úseků. Funkcí pro ohodnocení celé hrany je opět několik, například maximum, minimum, (vážený) aritmetický průměr, medián atd. Protože je výhodné, aby výsledná hodnota odrážela relativní délku hrany, po kterou daná vede podél objektů z vybrané kategorie, je zvolen aritmetický průměr (sekce 5.7). Tento celkový přístup má výhodu ve své stabilitě, protože nezáleží přesně na vzdálenosti mezi hranou a objektem, stačí pouze, že se někde poblíž hrany vyskytuje daný objekt. Naopak lze očekávat problémy s hodnotami pro objekty na hranici okolí.

#### 4.1.2 Viditelnost

Dále jsou pro každou kategorii vybírány z podmnožiny pouze objekty, které jsou z dané hrany (dopravní cesty) viditelné, proto je nutné zavést kategorii překážek. Překážka nemusí vést podél celé hrany, je tedy nutné vyhodnocovat viditelnost v rámci každého segmentu. Viditelnost objektu z hrany lze zjišťovat několika způsoby. Lze vytvářet několik paprsků v různých úhlech a určit, jaká část paprsků se protíná s překážkou a jaká část naopak projde až k objektu. Tento přístup lze vylepšit vytvářením paprsků procházející pouze skrze vrcholy překážek a získat tak okolní viditelnou plochu.[17] Protože při zjišťování vzdálenosti mezi hranou a okolními objekty jsou vždy nalezeny 2 nejbližší body, lze spojnicí těchto 2 bodů využít pro určení viditelnosti objektu. Pokud se nejkratší spojnice protíná s překážkou, pak není objekt z hrany viditelný. Tento způsob nemusí vždy fungovat podle očekávání, některé problémy tohoto přístupu jsou popsány v sekci 6.3.4.

### 4.1.3 Strany

Přestože výše uvedený postup dokáže rozlišit relativní délku hrany, po kterou je daný objekt z hrany viditelný, nedokáže rozlišit, zda se objekt nachází vlevo, vpravo nebo na obou stranách hrany (dopravní cesty). Protože graf dopravní sítě má orientované hrany, jsou levá a pravá strana hrany definovány podle směru od počátečního vrcholu hrany ke koncovému vrcholu hrany.

**Definice 4.1.** Máme hranu z grafu dopravní sítě  $e \in E_G$  a vrchol z mapových podkladů  $v \in V$ . Potom relativní pozice vrcholu  $v$  vůči hraně  $e$  je určena znaménkem determinantu matice  $M = [f(KV(e) - f(PV(e)) \quad f(v) - f(PV(e))]$ , označováno též jako analogie vektorového součinu ve 2D (*Cross Product (MathWorld)*[18] v rovnici 8). Pokud znaménko matice  $M$  je kladné, potom vrchol  $v$  leží vlevo od hrany  $e$ , a pokud je znaménko matice  $M$  záporné, potom vrchol  $v$  leží vpravo od hrany  $e$ . Pokud je determinant matice  $M$  roven 0, pak daný vrchol  $v$  leží v hraně  $e$ .

Díky této definici lze tak získat relativní pozici všech okolních objektů vzhledem k vybrané hraně. Pak lze také získat informaci, zda hrana prochází skrze daný objekt nebo mezi více objekty z jedné kategorie. Pokud najdeme 2 různé vrcholy  $v_1, v_2 \in V$  a vrcholy  $v_1, v_2$  leží v okolí hrany  $e$  ( $v_1, v_2 \in M$ ), kde vrchol  $v_1$  leží vlevo od hrany  $e$  a vrchol  $v_2$  leží vpravo od hrany  $e$ , potom hrana prochází skrze daný objekt (případně mezi objekty). Tento postup je podrobněji rozebrán v sekci 5.9.

## 4.2 Body zájmu

Kategorie bodů zájmu není hodnocena číselně, ale je využívána k obohacení navigačních instrukcí v křižovatkách (průniku dvou a více hran). Pokud se 2 různé hrany protínají, pak průsečík (bod  $K$ ) je křižovatkou. Podmnožina bodů zájmu je  $P \subseteq W$  a objekt  $i \in P$  je bod zájmu.

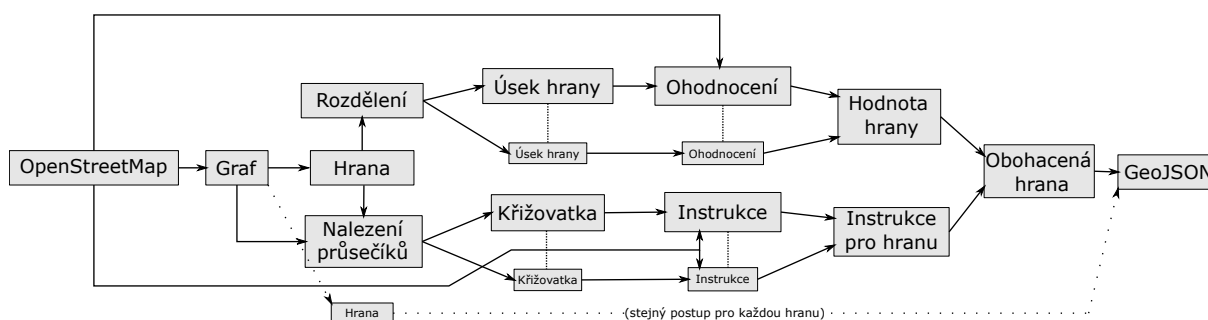
Kolem křižovatky může být několik bodů zájmu, pro instrukci je vhodné zvolit pouze jeden z těchto bodů. Tento bod lze vybrat na základě velikosti, barvy, historické významnosti, viditelnosti apod.

Nejdůležitější vlastností je pravděpodobně vzdálenost od křižovatky, vybraný bod zájmu  $i \in P$  má tak nejmenší minimální vzdálenost od dané křižovatky ( $\forall b \in P : \text{dist}(i, k) \leq \text{dist}(b, k)$ ). V sekci 5.6.4 jsou tyto body zájmu využity k obohacení navigačních instrukcí o orientační bod.

# Kapitola 5

## Implementace

### 5.1 Architektura



**Obrázek 5.1.** Diagram postupu zpracování dat z OpenStreetMap do souboru ve formátu GeoJSON

Na obrázku 5.1 je popsán postup řešení obohacování grafu dopravní sítě. Nejprve se z dat OpenStreetMap získá graf dopravní sítě (sekce 5.3), poté se každá hrana rozdělí na několik úseků (sekce 5.7) a každý úsek se ohodnotí zvlášť (sekce 5.6). Zároveň jsou využity body zájmu (sekce 5.6.4) k obohacení běžných navigačních instrukcí o orientační body. Výsledky jsou poté uloženy do souboru ve formátu GeoJSON (sekce 5.10).

### 5.2 OpenStreetMap

OpenStreetMap jsou mapové podklady celého světa, které jsou tvořeny komunitou dobrovolníků, kteří přidávají a upravují data o cestách, kavárnách, obchodech, parcích, zastávkách městské hromadné dopravy a mnohém dalším po celém světě. OpenStreetMap tvoří svobodná data pod licencí ODbL<sup>1</sup> poskytovaná nadací OpenStreetMap Foundation<sup>2</sup>. [4]

Datový model OpenStreetMap rozlišuje 3 základní datové prvky: uzly, cesty a relace. Zároveň každý prvek může obsahovat několik tagů (značek). Uzel (bod) je základní prvek, který vždy obsahuje svůj unikátní identifikátor (ID), zeměpisnou délku a šířku. Cesta (linie) je uspořádaný seznam uzlů. Pokud je cesta uzavřená (počáteční bod je stejný jako koncový bod), pak se může jednat o plochu (rozlišeno pomocí tagu `area`). Relace je seznam uzlů a cest (členů). Relace se nejčastěji používají pro popsání nějakého vztahu mezi jednotlivými členy relace (např. trasy, územní hranice nebo složitější polygony). Tag (značka) je dvojice klíče a hodnoty (značeno jako `klíč=hodnota`). Každý datový prvek (uzel, cesta, relace) může obsahovat libovolný počet různých tagů (nemůže ale obsahovat například 2 tagy se stejným klíčem). Přestože klíče a hodnoty mohou být

<sup>1</sup> <https://opendatacommons.org/licenses/odbl/>

<sup>2</sup> <https://www.openstreetmap.org/copyright>

libovolné textové řetězce, existují konvence pro jejich použití. Například budovy se označují klíčem `building`, pozemní komunikace klíčem `highway`, obchody klíčem `shop` apod. Tyto pravidla jsou podrobně popsána na OpenStreetMap Wiki<sup>1</sup>. [19]

Protože OpenStreetMapy jsou tvořeny lidmi, mapové informace nemusí vždy odpovídat realitě, některé informace mohou být chybné nebo mohou zcela chybět. Některé oblasti mohou být detailně zmapované, některé zase naopak příliš málo. Například les může být popsán pomocí páru `natural=forest` nebo `landuse=forest` a travnaté plochy mohou být popsány pomocí tagů `natural=grassland`, `landuse=grass` nebo `landuse=meadow`. Některé plochy mohou být chybně označeny, například tag `natural=village_green`, který má označovat střed vesnice (náves), je často používán pro různé druhy smíšené vegetace<sup>2</sup>.

Přístup do OpenStreetMap lze především pomocí editoru *iD* na oficiální stránkách OpenStreetMap, který ovšem vyžaduje určitou minimální znalost dat a pro běžného uživatele proto nemusí být nejvhodnější. Existují ale také neoficiální aplikace pro úpravu dat v OpenStreetMapách. Jednou z nich je mobilní aplikace *StreetComplete*<sup>3</sup>, která cílí na běžné uživatele a umožňuje doplňování chybějících údajů v OpenStreetMapách bez předchozí znalosti dat a tagů. Lze tedy předpokládat, že OpenStreetMapy se v průběhu času díky spoustě dobrovolníků (nebo importu dat z jiných zdrojů) budou rozšiřovat a vylepšovat.

Souřadnice objektů v OpenStreetMapách jsou uvedeny ve stupních, pro výpočet vzdáleností mezi objekty je vhodné používat jako jednotky metry. Je nutné proto definovat projekci, která převede souřadnice ve stupních na souřadnice v metrech. Protože neexistuje žádná univerzální nejlepší projekce, je nutné zvolit takový typ projekce, který zachovává vzdálenosti a velikosti ploch, tou je například Lambertova azimutální projekce. Systémem souřadnic je zvolena Evropská projekce LAEA, označená jako EPSG:3035<sup>4</sup>, která zaručuje přesnost 1 metr a je určena pro použití v oblasti Evropy, v jiných oblastech je tedy nutné použití jiného systému souřadnic.

### 5.3 Výběr cest

Graf dopravní sítě určený pro automobilovou dopravu nejčastěji obsahuje všechny dopravní komunikace od dálnic, rychlostních silnic až po místní komunikace. Protože obohacený graf dopravní sítě má větší význam pro chodce nebo cyklisty, je tedy výběr cest přizpůsoben zejména pro tyto druhy dopravy.

Plánování pěších cest je jednodušší z hlediska, že všechny cesty lze považovat za obousměrné, na rozdíl od plánování v silniční síti, kde se vyskytují omezení v podobě jednosměrných silnic. Komplikace ale nastávají díky faktu, že se lidé mohou pohybovat pěšky mnohem volněji než při jízdě autem. Mají totiž možnost chodit například přes trávníky (pokud je to v daném místě povoleno), volně chodit po pěších zónách nebo skrze pěší zkratky, které se nemusí vyskytovat v mapových podkladech.

Z OpenStreetMap jsou tak vybrány silnice III. třídy, rezidenční silnice, cyklostezky, pěší a obytné zóny, chodníky, schody, pasáže, stezky, pěšinky a jiné nezpevněné cesty. Zároveň tyto cesty musí být veřejně dostupné cesty, jsou tak vyjmuty všechny soukromé cesty nebo cesty přístupné jen pro zákazníky (například zoologická zahrada, parkoviště atd.). V tabulce `planet_osm_line` se nalézají také relace (autobusové linky,

<sup>1</sup> [https://wiki.openstreetmap.org/wiki/Map\\_Features](https://wiki.openstreetmap.org/wiki/Map_Features)

<sup>2</sup> [https://wiki.openstreetmap.org/wiki/Cs:Tag:landuse%3Dvillage\\_green](https://wiki.openstreetmap.org/wiki/Cs:Tag:landuse%3Dvillage_green)

<sup>3</sup> <https://github.com/westnordost/StreetComplete>

<sup>4</sup> <https://epsg.io/3035>



elektrické vedení apod.) rozlišené pomocí záporného znaménka identifikátoru, vybírám proto pouze objekty s kladným identifikačním číslem. SQL dotaz pro získání těchto cest včetně všech požadavků nad tabulkou všech hran je vypsán v bloku 5.2. Detaily databáze a použitých knihoven jsou popsány v sekci 5.4.

Z těchto vybraných cest je následně sestaven graf dopravní sítě, ve kterém každá vybraná cesta je dopravní cestou a křižovatky více cest jsou dopravní uzly.

```
SELECT osm_id, way
FROM planet_osm_line
WHERE highway IN ('tertiary','residential','cycleway','pedestrian',
'footway','living_street','unclassified','tertiary_link','service',
'track','steps','path','corridor')
AND (foot IS NULL OR foot NOT IN ('no','private','customers'))
AND (bicycle IS NULL OR bicycle NOT IN ('no','private','customers'))
AND (access IS NULL OR access NOT IN ('no','private','customers'))
AND osm_id >= 0;
```

**Obrázek 5.2.** SQL dotaz pro získání cest

## 5.4 Algoritmy a knihovny

OpenStreetMapy obsahují mapové podklady celého světa, pro tuto práci je vybrán pouze obdélníkový výřez městské čtvrti Praha 6 Dejvice a blízkého okolí. Tento mapový výřez se nachází na okraji Prahy, proto obsahuje silnice různých tříd, spoustu zeleně (včetně Letenských sadů a části Stromovky) i vodní oblasti (řeka Vltava a Výzkumný ústav vodohospodářský).

Data z OpenStreetMap lze nejčastěji stáhnout ve formátu XML nebo v binárním formátu PBF. Výhodou formátu XML je jejich čitelnost pro člověka, nevýhodou může být vyšší velikost souboru. Je možné stáhnout veškerá data celého světa z OpenStreetMap, celková velikost takového souboru ovšem převyšuje 49 GB ve formátu PBF a 84 GB ve formátu XML<sup>1</sup>. Pro účel této práce je zbytečné stahovat všechna data a poté vytvářet menší výřez.

Naštěstí existují další služby, které nabízí ke stažení již menší výřezy světa, např. služba Geofabrik<sup>2</sup>, která umožňuje stažení výřezů jednotlivých světových kontinentů nebo konkrétních států, které lze oříznout například pomocí níže zmíněného nástroje Osmosis. Oficiální web OpenStreetMap<sup>3</sup> umožňuje stažení (exportování) menších mapových výřezů (limit 50 000 bodů), případně pomocí služby Overpass API<sup>4</sup> je možné stahovat větší mapové oblasti. Výřez městské čtvrti Praha 6 Dejvice a okolí je proto získán pomocí služby Overpass API ve formátu XML.

Existuje spousta knihoven v různých programovacích jazycích pro práci s daty z OpenStreetMap. Některé nástroje čtou data přímo ze souboru a jiné naopak pracují s daty v databázi a některé nástroje umožňují obojí.

Osmosis<sup>5</sup> dokáže pracovat s daty v souborech i v databázích. Nástroj osm2pgsql<sup>6</sup> načte data ze souboru a ty následně uloží do PostgreSQL databáze s PostGIS rozšířením. Přestože oba nástroje dokáží přečíst data ze souboru, které poté načtou do databáze, způsoby uložení OSM tagů v databázi a reprezentace geometrických objektů jsou mírně

<sup>1</sup> <https://planet.openstreetmap.org/>

<sup>2</sup> <https://download.geofabrik.de/>

<sup>3</sup> <https://www.openstreetmap.org/>

<sup>4</sup> <https://overpass-api.de/>

<sup>5</sup> <https://wiki.openstreetmap.org/wiki/Osmosis>

<sup>6</sup> <https://github.com/openstreetmap/osm2pgsql>

odlišné. Osm2pgsql vytvoří tabulky pro uzly, hrany a polygony a nejběžnější tagy uloží jako sloupce tabulky. Při uvedení přepínače `-h` osm2pgsql uloží ostatní tagy do sloupce *tags* jako datový typ *hstore*. Osmosis vytvoří 3 tabulky pro uzly, hrany a polygony, ale tagy uloží buď do zvláštní tabulky se sloupci identifikátor, klíč a hodnota nebo je všechny uloží do datového typu *hstore* (variantu uložení tagů lze zvolit). Osm2pgsql reprezentuje všechny objekty pomocí datového typu *geometry* z PostGIS rozšíření, zatímco Osmosis takto reprezentuje jenom jednotlivé uzly a složitější objekty reprezentuje jako cizí klíče odkazující na primární klíče uzlů. Oba nástroje zároveň automaticky vytvoří pro tabulky i sloupce s geometrickými daty indexy pro rychle vyhledávání.

Pro geometrické operace je tedy vhodnější importovat data pomocí osm2pgsql, protože všechny objekty z OpenStreetMap jsou reprezentovány pomocí datového typu *geometry*. Naopak výhodou nástroje Osmosis je možnost zpětného uložení dat z databáze do souboru.

Protože některé cesty se navzájem kříží, jsou tyto cesty rozděleny v křižovatkách pomocí nástroje pgRouting<sup>1</sup> (použita verze 3.0.0 kvůli opravám funkce *pgr\_nodeNetwork*). Tento nástroj je sice určen hlavně pro plánování v navigačních grafech, ale obsahuje i funkce, které dokáží rozdělit cesty v křižovatkách (bohužel nerozezná mimoúrovňové křížení) a převést obecný graf na graf, který je vhodnější pro plánování cest v grafu.

Programovacím jazykem pro implementaci obohacování grafu dopravní sítě je jazyk Java. Pro komunikaci mezi Javou a PostgreSQL databází je využit JDBC driver, který zpracovává SQL dotazy, posílá je do databáze a vrací výsledky v podobě množiny řádků (ResultSet). Protože geometrické datové typy nejsou základním JDBC driverem podporovány, je přidáno také rozšíření<sup>2</sup> JDBC driveru, které přidává podporu geometrických i geografických datových typů do Javy. Tato knihovna ovšem neumožňuje složitější geometrické operace (například nejmenší vzdálenost mezi 2 objekty) přímo v Javě, je tedy nutné pro každou operaci sestavit SQL dotaz. Pokud se vyskytuje v SQL dotazu geometrický objekt<sup>3</sup>, dochází k převodu tohoto objektu na WKT řetězec<sup>4</sup>.

Protože tato práce opakovaně využívá různé geometrické operace, je tento přístup časově náročný. Proto jsou všechny geometrické operace řešeny pomocí knihovny JTS<sup>5</sup>, která obsahuje všechny potřebné funkce ekvivalentní PostGIS funkcím. Datové typy geometrických objektů používaných v PostGIS JDBC driveru a knihovně JTS jsou různé, je tedy nutné převádět všechny objekty do JTS geometrických objektů, ve kterých lze provádět geometrické operace. Tento převod je ovšem mnohonásobně rychlejší než převod na WKT řetězec, poslání a vyhodnocení SQL dotazu a následné získání výsledku. Na některé operace (například nalezení všech objektů v databázi poblíž dané cesty) je ale stále nutné použít PostGIS funkce.

Další použitou knihovnou je GSON<sup>6</sup>, která dokáže serializovat Java objekty do formátu JSON a také pro řetězec v tomto formátu dokáže vytvořit odpovídající Java objekt. Tato knihovna je využívána při ukládání výsledku do souboru ve formátu GeoJSON.

<sup>1</sup> <https://pgrouting.org/>

<sup>2</sup> <https://github.com/postgis/postgis-java/tree/master/jdbc>

<sup>3</sup> <https://github.com/postgis/postgis-java/blob/master/jdbc/src/main/java/org/postgis/Geometry.java>

<sup>4</sup> <http://docs.opengeospatial.org/is/18-010r7/18-010r7.html>

<sup>5</sup> <https://locationtech.github.io/jts/>

<sup>6</sup> <https://github.com/google/gson>

Poslední knihovnou je ukazatel průběhu<sup>1</sup>, který zobrazuje počet již zpracovaných cest, celkový počet cest a odhad potřebného času pro zpracování všech vybraných cest z databáze.

Výsledkem programu (nástroje) v Javě může být seznam identifikátorů všech objektů v okolí každé cesty. Protože ale nalezení všech okolních objektů podle souřadnic je srovnatelně rychlé jako hledání podle uloženého pole identifikátorů okolních objektů, nevyplatí se ukládat samotný seznam okolních objektů. Hodnotnějším výstupem jsou pak číselné hodnoty shrnující kvalitu objektů v okolí vybrané cesty.

Výstupem je tak soubor ve formátu GeoJSON, který obsahuje všechny cesty společně s vypočtenými vlastnostmi. Tento soubor lze poté vizualizovat například pomocí nástroje Kepler.gl<sup>2</sup>, který dokáže vykreslit všechny cesty a obarvit je třeba podle jednotlivých vypočtených vlastností.

Protože možnosti této aplikace jsou poměrně omezené, je pro tuto práci vytvořena webová aplikace (dostupná na adrese <http://prague-ways.netlify.app>), která sama stáhne a načte výstupní soubor ve formátu GeoJSON a data vizualizuje. Aplikace využívá službu Mapbox<sup>3</sup> (konkrétně knihovnu Mapbox GL JS) pro zobrazení dat a jejich vizualizaci (detaily popsány v sekci 5.11.1).

## 5.5 Postup

Před samotným obohacováním grafu je potřeba zajistit, že existuje databáze, která obsahuje data z OpenStreetMap. Protože nástroje Osmosis i osm2pgsql používají různá schémata databáze i různé názvy tabulek, je možné načíst data ze souboru pomocí obou nástrojů do jedné společné databáze, ale z důvodu, že data z databáze nejsou ukládána zpět do souboru, použití nástroje Osmosis není v tomto případě nezbytné.

Prvním krokem je nalezení všech identifikátorů cest uložených v databázi. Ty jsou v OpenStreetMapách uloženy jako hrany a jsou charakterizovány klíčem `highway` s hodnotou, která vyjadřuje třídu dané cesty nebo její účel (např. rezidenční cesta), vybrané cesty jsou popsány v sekci 5.3. Všechny nalezené identifikátory cest jsou poté uloženy do seznamu, který lze paralelně zpracovat pomocí Java Streamu.

Pro každý identifikátor cesty se nalezne geometrická reprezentace, pomocí které pak lze hledat všechny okolní objekty. Protože uzly, hrany i polygony jsou uloženy v oddělených tabulkách, je potřeba hledat v každé této tabulce zvlášť. Pro nalezení okolních objektů je použita PostGIS funkce `ST_DWithin`, která pro 2 geometrické datové typy a vzdálenost rozhodne, zda jsou od sebe vzdálené v rámci dané maximální vzdálenosti. Tato funkce zároveň automaticky používá databázové indexy pro rychlejší vyhledávání. Nastavením vhodné velikosti okolí se zabývá sekce 6.1. Výsledkem jsou tedy databázové záznamy všech objektů, které jsou v okolí dané cesty. Nyní lze podle těchto okolních objektů ohodnotit danou cestu. Každý objekt v OpenStreetMapách obsahuje několik párů klíč hodnota (blíže sekce 5.6), které popisují vlastnosti daného objektu.

Protože některé cesty v OpenStreetMapách mohou měřit několik stovek metrů, může se charakter okolí v průběhu cesty výrazně měnit. Vhodným ukazatelem by proto byla hodnota vyjadřující relativní délku úseku vedoucí podél daného objektu v okolí (blíže popsáno v sekci 5.7).

Navíc existují různé druhy zeleně, je tedy potřeba rozlišovat například mezi travnatou plochou a lesem. Proto jsou objekty v rámci každé kategorie rozdělené podle vah

<sup>1</sup> <https://github.com/ctongfei/progressbar/>

<sup>2</sup> <https://kepler.gl/>

<sup>3</sup> <https://www.mapbox.com/>

(popsáno v sekci 5.6). V případě některých cest může nastat situace, že například potok není z cesty viditelný, protože mezi cestou a potokem stojí budova. Je proto výhodné detekovat různé překážky podél cest. Je také důležité rozlišit, zda cesta vede podél parku nebo skrz park. Proto jsou pro každou kategorii vytvořeny 2 hodnoty popisující tyto stavy. Takových kategorií lze pro jednu cestu vytvořit více, např. počet okolní zeleně, vodních ploch, silnic, budov atd. Zároveň informaci o okolních objektech je použita pro vytváření bohatších navigačních instrukcí v křižovatkách cest (sekce 5.6.4).

Tyto výsledné hodnoty jsou nakonec uloženy jako soubor ve formátu GeoJSON. Následně lze tyto data vizualizovat v mapě pomocí nástroje Kepler.gl nebo pomocí knihovny Mapbox GL JS<sup>1</sup>.

## 5.6 Kategorie

Podle pozice nebo tvaru okolního objektu nelze rozhodnout, o jaký druh objektu se jedná. V OpenStreetMapách jsou informace o objektech uloženy pomocí tagů, které jsou vždy ve tvaru **klíč=hodnota**. Na základě těchto tagů lze určit, o jaký druh objektu (zeleně, obchod, silnice apod.) se jedná. Všechny objekty lze tedy roztrždit do kategorií (skupin) objektů podle informací v těchto tagách. Informace o objektech jsou vždy získávány pouze podle obsažených informací, nikoliv podle chybějící informace (například pokud pro zahradu chybí tag `garden:type`, pak zahrada může i nemusí být veřejně přístupná).

V této práci jsou zavedeny následující kategorie objektů (zmíněny již v kapitole 3 Specifikace problému), které ovlivňují pozitivně i negativně kvalitu každé cesty. Každá kategorie navíc rozlišuje všechny druhy objektů podle ručně určených vah (desetinné číslo v rozmezí  $[0, 1]$ ).

Nejvyšší váhu 1 mají objekty, které jsou nejvýraznější nebo nejvýznamnější z dané kategorie, váze 0.99 odpovídají objekty, které jsou významné, ale v porovnání s objekty s maximální vahou mohou mít nepatrně menší význam. Objekty s vahou 0.01 jsou takové objekty, které nelze zanedbat a stále patří do dané kategorie, ale může se jednat o méně výrazné objekty (například trávník) nebo o objekty s nejistým vlivem na okolí (například silnice III. třídy v případě kategorie rušných silnic). Dále implicitně platí, že všechny nevybrané objekty pro danou kategorii mají váhu 0. Tyto váhy je přirozeně možné určit odlišně a ovlivnit tak relativní význam objektů.

### 5.6.1 Rušné silnice

V OpenStreetMapách jsou všechny silnice označeny klíčem `highway`. V článku *A System for Generating Customized Pleasant Pedestrian Routes Based on OpenStreetMap Data* [5] jsou silnice rozděleny podle jejich hlučnosti na 2 skupiny. V první skupině (hlučné) jsou dálnice, silnice pro motorová vozidla, silnice I. a II. třídy a ve druhé skupině (méně hlučné) jsou umístěny silnice III. třídy, rezidenční silnice a pěší zóny. Toto rozdělení bylo vytvořeno podle délky silnice a naměřených hodnot hluku v městě Heidelberg v Německu.

Jako rušné silnice jsou vybrány všechny cesty od dálnic až po silnice III. třídy. Tagy cest s jejich vahami jsou vypsány v tabulce 5.1.

### 5.6.2 Zeleně

V OpenStreetMapách existuje spousta objektů, které obsahují zeleně. Všechny tagy, které odpovídají zeleni, jsou vypsány v tabulce 5.2. V tabulce nejsou zahrnuty například

<sup>1</sup> <https://github.com/mapbox/mapbox-gl-js/>

Váha	Klíč	Hodnoty
1.00	highway	motorway, motorway_link, trunk, trunk_link, primary, primary_link, secondary, secondary_link
0.01	highway	tertiary, tertiary_link

**Tabulka 5.1.** Váhy klíčů a hodnot pro silnice

hřiště nebo stadiony, protože nemusí nutně obsahovat zeleň nebo nemusí být veřejně přístupné.

Je rozdíl, zda cesta vede podél trávníku nebo parku, proto mají travnaté plochy menší váhu než parky nebo lesy. Zároveň ale trávníky nemají nulovou váhu, protože cesty vedoucí podél nich jsou bohatší z hlediska zeleně než cesty, které kolem sebe nemají vůbec žádnou zeleň.

Váha	Klíč	Hodnoty
1.00	natural	wood
	landuse	forest
	leisure	park, nature_reserve
0.99	natural	tree_row, scrub, heath
	landuse	meadow, orchard, vineyard
	leisure	playground
0.01	garden:type	botanical, castle, community, monastery
	natural	grassland
	landuse	farmland, village_green, grass, greenfield

**Tabulka 5.2.** Váhy klíčů a hodnot pro zeleň

### 5.6.3 Vodní plochy

Podobně jako u zeleně existuje i pro vodní plochy a oblasti s nimi spojenými spousta tagů. Všechny tagy, které patří do této kategorie, jsou vypsány v tabulce 5.3. Tato tabulka neobsahuje klíč `water`, protože tento klíč by se měl (podle OpenStreetMap wiki<sup>1</sup>) vyskytovat vždy společně s hodnotou `natural=water`.

Váha	Klíč	Hodnoty
1.00	natural	water, wetland, glacier, spring, hot_spring, geyser
	landuse	basin, reservoir, pond
	waterway	river, riverbank, tidal_channel, canal, dam, weir, waterfall
0.99	waterway	stream, ditch

**Tabulka 5.3.** Váhy klíčů a hodnot pro vodní plochy

### 5.6.4 Body zájmu

Mezi okolními objekty se nachází také body zájmu (obchody, fontány, pomníky atd.). Tyto body zájmu lze využít pro lepší navigační instrukce (podobně jako v práci *Enriching Wayfinding Instructions with Local Landmarks* [1]).

Pro každou cestu jsou získány všechny křižovatky, ve kterých se tato cesta kříží s ostatními cestami. V každé z těchto křižovatek je následně vytvořena navigační instrukce. Běžná navigační instrukce vypadá například takto: „Jděte 200 metrů rovně

<sup>1</sup> <https://wiki.openstreetmap.org/wiki/Key:water>

a potom odbočte vlevo“. Tuto instrukci lze vylepšit o orientační bod. Mezi orientační body (body zájmu) patří všechny objekty, které obsahují tagy z tabulky 5.4, naopak lavičky, odpadkové koše nebo budovy bez jména (tj. obyčejné domy) nejsou dostatečně výrazné. Ze všech okolních bodů zájmu je vybrán pouze ten, který leží nejbližší dané cestě. Tento přístup lze zdokonalit pomocí snímků z daného místa, z kterých se následně určí nejvýraznější orientační bod. Díky bodu zájmu lze vyřadit informaci, kolik metrů je potřeba ujít a kdy přesně zabočit. To se může hodit v případě, že aplikace nemůže průběžně využívat informaci o aktuální poloze uživatele nebo si uživatel chce instrukce zapsat na papír a projít trasu bez navigace. Dále lze pro instrukci využít jméno ulice, na kterou se odbočuje (pokud je tato ulice pojmenována).

Klíč	Hodnoty
shop	*
public_transport	*
amenity	bank, fuel, bus_station, embassy, hospital, clinic, school, restaurant, pub, cafe, bar, fast_food, food_court, theatre, kindergarten, pharmacy, library, college, university, fountain, fire_station, police shelter, telephone, playground, grave_yard
historic	*

**Tabulka 5.4.** Klíče a hodnot pro body zájmu

V případě, že je kolem cesty bod zájmu a zároveň cílová ulice má jméno, potom instrukce vypadá například následovně: „Odbočte vlevo u banky ČSOB do ulice Kyjevská“. Pokud ale název ulice chybí nebo v okolí cesty není žádný bod zájmu, nelze navigační instrukci ničím obohatit a je možné vytvořit pouze základní směrovou instrukci.

Protože levá a pravá strana cesty závisí na směru pohybu, jsou instrukce tvořeny pro směr od počátečního ke koncovému bodu cesty i pro opačný směr. Pokud jiná cesta navazuje na danou cestu v přímém směru, pak je vytvořena instrukce pro pokračování v daném směru. Pokud by cesta navazovala na počáteční vrchol, pak je vytvořena instrukce „Otočte se“ s případnou informací o okolí. V každé křižovatce jsou tak 4 možné směry pokračování: vlevo, vpravo, rovně a zpět. Výsledky tohoto přístupu jsou diskutovány v sekci 6.3.5.

## 5.7 Dělení cest

Pro krátké úseky cest dosavadní přístup pro ohodnocení cest dostačuje. Pokud je ale cesta delší, charakter okolní krajiny se často výrazně mění v průběhu cesty. Příkladem může být chodník<sup>1</sup> vedoucí částečně kolem rušné Veletržní ulice a částečně podél Střemovky. Možným řešením by bylo všechny cesty delší než stanovená hranice (např. 50 metrů) předem rozdělit a poté je hodnotit zvlášť. Tento přístup je ovšem výpočetně náročnější (popsáno v článku od Hochmaira et al.[6]) pro vyhledávání optimálních tras i pro vizualizaci všech cest. Použitá funkce `pgr_nodeNetwork` z knihovny `pgRouting` sice spoustu cest rozdělí a vytvoří tak spoustu nových cest, ale cesty bez křížení zbytečně nerozděluje, výpočetní náročnost tedy vzroste pouze mírně.

<sup>1</sup> <https://www.openstreetmap.org/way/562284025>



Zvolen je tak následující přístup. Každá cesta se rozdělí na několik segmentů, každý segment cesty se ohodnotí zvlášť a poté na základě dílčích hodnot segmentů se ohodnotí celá cesta jednou společnou hodnotou. Hodnoty segmentů cest se určí podle kategorií z části 5.6 a výsledná hodnota cesty je následně určena jako aritmetický průměr ze všech hodnot segmentů. Protože všechny váhy kategorií jsou v intervalu od 0 do 1, průměr hodnot leží také v tomto intervalu a tato hodnota tedy vyjadřuje relativní délku, po kterou vede cesta podél vybrané kategorie objektů. Pokud ovšem cesta vede podél více objektů s různými vahami, je interpretace takového výsledku složitější.

Velikost segmentů cesty je důležitým parametrem pro přesný výsledek. Pokud by byly objekty blízko středu cesty významnější, bylo by možné cestu rozdělit nerovnoměrně. Protože ale všechny okolní objekty jsou stejně důležité, každá cesta se rovnoměrně rozdělí tak, aby každý segment byl přibližně stejně dlouhý. Délka segmentu je také důležitá pro detekci překážek mezi cestou a okolními objekty. Pokud by délka segmentu cesty byla delší než velikost okolí (50 metrů), výsledky by nebyly dostatečně přesné a detekce překážek (viditelnosti) by nemusela správně fungovat. Zároveň nemá smysl cesty dělit na příliš krátké segmenty (například 1 metr), protože by se tím značně zvýšila výpočetní náročnost. Proto je jako maximální délka segmentu zvolena vzdálenost 15 metrů. Pokud je celá cesta kratší než 15 metrů, nedojde k rozdělení cesty. Pokud je cesta delší než 15 metrů, vypočítá se počet segmentů jako  $\text{ceil}(\frac{\text{délka cesty}}{15})$  a délka jednoho segmentu je určena jako  $\frac{\text{délka cesty}}{\text{počet segmentů}}$ . Zároveň je počet segmentů omezen maximálně na 100 segmentů pro případ, že by některá cesta byla příliš dlouhá.

## 5.8 Překážky

Mezi cestou a okolním objektem (parkem, řekou) se může nacházet například plot nebo budova, která brání ve výhledu. Proto po nalezení všech okolních objektů jsou vybrány pouze ty, které jsou z dané cesty viditelné. Tato překážka nemusí vést po celé délce cesty, proto je viditelnost zjišťována jednotlivě pro každý úsek cesty (popsáno v sekci 5.7).

Pro každou trojici cesta, okolní objekt, překážka je zjištěna viditelnost následujícím způsobem. Mezi cestou a okolním objektem jsou vybrány 2 nejbližší body, z kterých je následně vytvořena spojnice těchto bodů. Pokud se tato spojnice protíná s překážkou, potom lze usoudit, že okolní objekt není z cesty vidět. Protože v případě zanedbatelné nenulové vzdálenosti se objekty fyzicky neprotínají, je kvůli numerické stabilitě počítána vzdálenost mezi spojnicí a překážkou. Pokud je tato vzdálenost menší než stanovená mez (1 cm), pak okolní objekt není z cesty vidět. Jinak je daný objekt součástí okolí (pokud patří do některé výše uvedené kategorie ze sekce 5.6).

Klíč	Hodnoty
barrier	city_wall, fence, hedge, retaining_wall, wall
building	*

**Tabulka 5.5.** Klíče a hodnot pro překážky

Všechny tagy překážek jsou uvedeny v tabulce 5.5. Tyto překážky mají stejnou váhu (rovnou 1) a všechny jsou považovány za nepropustné překážky, žádná z nich tak není například polopropustná. Libovolné budovy jsou považovány také jako překážky bez ohledu na jejich tvar nebo výšku (v tabulce zapsáno jako klíč `building` s libovolnou hodnotou).

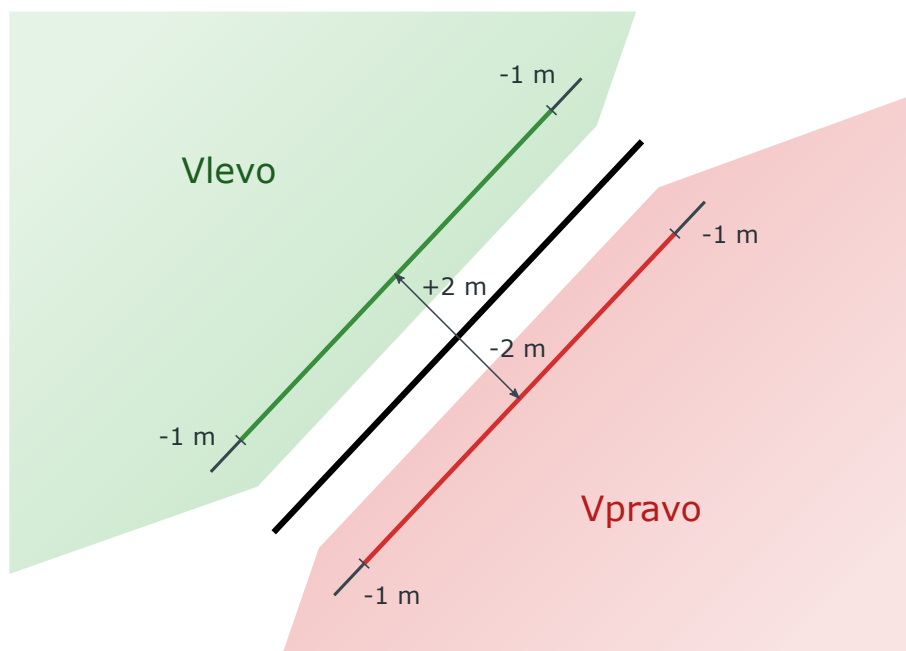
## 5.9 Druhy cest

Všechny cesty byly dosud hodnoceny podle toho, zda se kdekoliv kolem cesty nachází libovolný objekt z vybrané kategorie (zeleň, voda, silnice). Relativní pozice objektu vůči cestě ale může být důležitou dodatečnou informací pro hodnocení cest. Pokud by cesta vedla podél parku nebo skrz park, hodnoty obou cest by byly totožné, přestože cesta vedoucí skrz park je kvalitnější z hlediska zeleně než cesta vedoucí pouze podél parku.

Proto jsou pro každou kategorii vytvořeny podkategorie: cesty vedoucí podél zeleně, skrz zeleň, podél silnice, mezi silnicemi, skrz silnici, podél vody a mezi vodními plochami. Pokud cesta vede například skrz parky, pak cesta jde skrz park, ale nejde podél parku (hodnota odpovídající kategorii podél zeleně je rovna 0).

Pro rozlišení těchto podkategorií v rámci jedné kategorie je nutné zjistit, zda se vybraný objekt nachází vlevo nebo vpravo od cesty. Pokud například cesta vede podél řeky, následně přes ní a poté opět podél ní, řeka by se nacházela vlevo i vpravo od celé cesty, ale tato cesta nevede mezi řekami. Proto je relativní pozice zjišťována vůči každému segmentu cesty.

Jestli se objekt nachází vlevo nebo vpravo od dané cesty záleží na tom, jakým směrem po cestě jdeme, ale každá cesta má jednoznačně určené počáteční a koncové vrcholy, směr cesty je proto určen od počátečního vrcholu ke koncovému. Pro rozlišení, zda se vybraný objekt nachází vlevo nebo vpravo, je každý segment cesty posunut o 2 metry na obě strany a zároveň zkrácen o 1 metr (polovina vzdálenosti posunutí) na obou koncích, čímž nejsou zahrnuty objekty, které leží blíže jednomu z konců cesty. Pokud by některý posunutý segment měřil méně než 2 metry, pak je daná část nahrazena středem tohoto posunutého segmentu. Výsledná situace je pak znázorněna na obrázku 5.3.



**Obrázek 5.3.** Posunuté segmenty cesty pro zjišťování, zda objekt leží ■ vlevo nebo ■ vpravo. Barevné plochy znázorňují množinu bodů, které leží blíže k posunutému segmentu cesty než k původní cestě.



Vzdálenost posunutí byla zvolena takto, protože některé cesty<sup>1</sup> tvoří cyklus s velmi malým poloměrem. Pokud by vzdálenost posunutí byla větší než poloměr, pak by cestu nebylo možné posunout směrem do středu.

Po posunutí segmentu cesty se porovnají vzdálenosti k původnímu segmentu a k oběma posunutým segmentům. Pokud jsou všechny vzdálenosti k vybranému objektu rovné nule nebo posunutím na obě strany se vzdálenosti k vybranému objektu zmenší, pak na obou stranách leží objekt s touto váhou. Pokud se vzdálenost k vybranému objektu sníží pouze na jedné ze stran, je aktualizována maximální váha objektů pro stranu, na které objekt leží. U rušných silnic jsou navíc rozlišeny cesty vedoucí mezi silnicemi nebo vedoucí skrze silnice. Pokud cesta překříží silnici, pak cesta vede skrze silnici, jinak cesta vede mezi silnicemi. Pokud sama cesta je rušnou silnicí (konkrétně silnicí III. třídy), pak cesta vede skrze rušnou silnici. Všechny vzdálenosti jsou porovnávány s tolerancí 10% vzdálenosti posunutí.

Po probrání všech okolních objektů jsou srovnány hodnoty maximálních vah objektů na obou stranách cesty. Pokud jsou váhy různé, potom cesta vede podél objektu s vyšší vahou. Pokud se váhy rovnají, potom cesta vede skrze objekt s danou vahou. Může se také stát, že cesta vede mezi 2 parky, potom tato cesta vede skrze park, přestože se jedná o 2 různé parky. Pokud by cesta vedla mezi stromořadím a loukou, pak hodnota podkategorie „skrz zeleň“ bude 0.99, přestože se jedná o 2 různé druhy zeleně.

## 5.10 Formát dat

Výstupem nástroje (programu) v Javě je soubor ve formátu GeoJSON jako typ *FeatureCollection*. Každá cesta obsahuje původní geometrii jako *LineString* v souřadném systému daném specifikací GeoJSON (EPSG:4326<sup>2</sup>), unikátní identifikátor (*id*) a vlastnosti cesty, které jsou tvořeny hodnotami jednotlivých podkategorií (*green\_by*, *green\_through*, *water\_by*, ...), počtem segmentů cesty (*\_segment\_count*) a pro každou kategorii jsou vypsané i tagy okolních objektů (přípona *\_matched*). Zároveň jsou v souboru zahrnuty i navigační instrukce ve všech křižovatkách dané cesty. Pod klíčem *\_poi* se nachází pole všech křižovatek, každá křižovatka obsahuje geografické souřadnice, jméno nejbližšího bodu zájmu v okolí křižovatky a navigační instrukce pro všechny navazující cesty z dané křižovatky. Výhodou formátu GeoJSON je jeho čitelnost a jednoduché zpracování tohoto souboru v JavaScriptu, ve kterém je napsána webová aplikace (sekce 5.11.1). Příklad části výstupu je uveden v bloku 5.4.

## 5.11 Vizualizace dat

Ohodnocené cesty jsou uloženy ve formátu GeoJSON (popis v kapitole 5.10). Pro vizualizaci dat lze využít například knihovnu Kepler.gl nebo přímo její webovou demo aplikaci<sup>3</sup>. Po načtení dat se automaticky vytvoří nová vrstva a všechny cesty ze souboru se vykreslí stejnou barvou i se stejnou tloušťkou. Následně lze například nastavit, že barva cesty bude záviset na hodnotě některé z vybraných kategorií. Po najetí myši na libovolnou cestu se zobrazí některé vlastnosti této cesty. Nakonec lze mapu se všemi viditelnými vrstvami exportovat jako obrázek.

Přestože výsledná vizualizace dat pomocí této knihovny funguje po vzhledové stránce výborně, aplikace není příliš uživatelsky přívětivá a možnosti aplikace jsou omezené.

<sup>1</sup> <https://www.openstreetmap.org/way/396757473>

<sup>2</sup> <https://epsg.io/4326>

<sup>3</sup> <https://kepler.gl/demo>

```

...
{
  "type": "Feature",
  "geometry":
  {
    "type": "LineString",
    "coordinates":
    [
      [14.368984,50.096149],
      [14.36873,50.09676]
    ]
  },
  "properties":
  {
    "water_by": 0,
    "water_between": 0,
    "_green_matched": "landuse=village_green;",
    "roads_through": 0,
    "_segment_count": 5,
    "roads_by": 0,
    "roads_between": 0,
    "_poi":
    [
      {
        "geometry": [14.36873,50.09676],
        "poi": "",
        "next": {"4568550": "Odbočte vpravo do ulice Na Rozdílu",...}
      }
    ],
    "green_by": 0.004,
    "_roads_matched": "",
    "_water_matched": "",
    "green_through": 0.006
  },
  "id": 4892719
},
...

```

**Obrázek 5.4.** Část výstupního souboru ve formátu GeoJSON (zkráceno pro přehlednost)

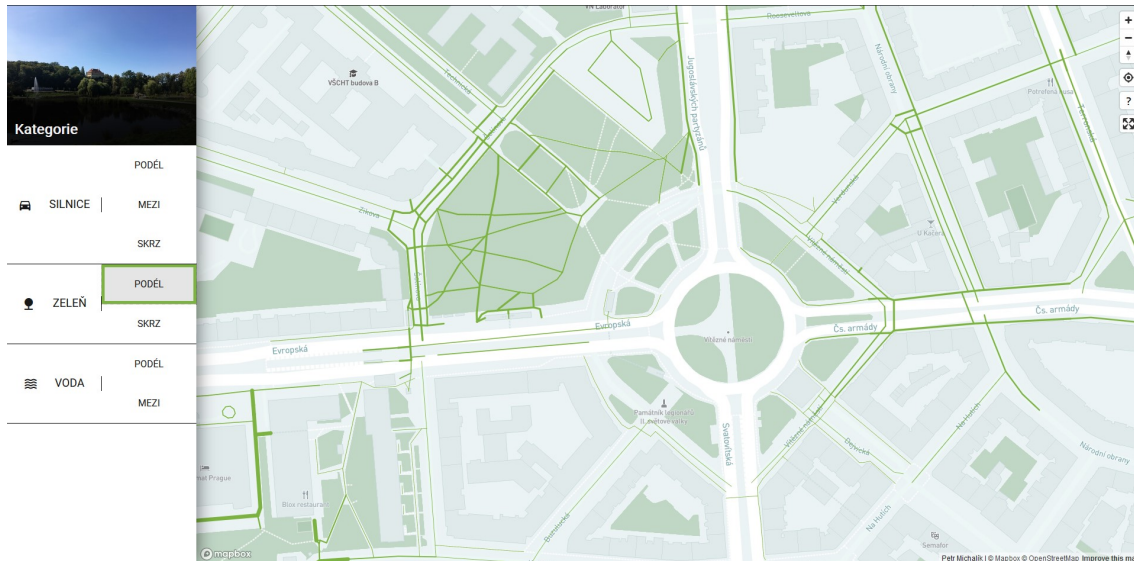
Knihovna Kepler.gl využívá pro vykreslování mapových podkladů a vykreslování vrstev dat službu Mapbox (knihovnu Mapbox GL JS). Proto byla pro účely této práce vytvořena vlastní webová aplikace pro vizualizaci obohacených cest.

### 5.11.1 Webová aplikace

Webová aplikace využívá knihovnu Mapbox GL JS<sup>1</sup> pro vykreslování mapových podkladů, přidávání vrstev s daty a stylování vykreslených cest na základě dat (data-driven styling). Aplikace obsahuje v levém panelu ovládací prvky, které zobrazují a skrývají vrstvy pro jednotlivé barevně rozlišené podkategorie. Uživatel má tedy možnost zobrazit si více vrstev z různých kategorií najednou. Všechny cesty se po vybrání libovolné kategorie obarví barvou vybrané kategorie, tloušťka každé čáry je poté určena hodnotou dané kategorie. Pokud tedy cesta bude mít tuto hodnotu rovnou nule, tloušťka čáry bude nulová a taková cesta tedy nebude viditelná. V pravé části aplikace se nad mapou nacházejí ovládací prvky mapy (přiblížení, oddálení, rotace, zobrazení přibližné aktuální geografické polohy, přepnutí do režimu celé obrazovky) včetně nápovědy pro použití aplikace (využita knihovna Intro.js<sup>2</sup>) a uložení mapy jako PNG obrázek.

<sup>1</sup> <https://github.com/mapbox/mapbox-gl-js/>

<sup>2</sup> <https://introjs.com/>



**Obrázek 5.5.** Webová aplikace

Protože soubor ve formátu GeoJSON může být příliš velký, aplikace si stáhne zipovaný soubor a pomocí knihovny JSZip<sup>1</sup> soubor extrahuje. Soubor s daty se tak zmenší přibližně na desetinu velikosti (na přibližně 1 MB), nevýhodou může být vyšší náročnost aplikace kvůli nutnosti soubor extrahovat.

Pro uložení mapy do PNG obrázku je převzata část kódu z repozitáře print-maps<sup>2</sup>, který vytvoří neviditelný HTML canvas s mapou s určenými rozměry a kvalitou (DPI). Pro uložení obrázku je použita knihovna FileSaver.js<sup>3</sup>, která z Blob objektu s obrázkem vytvoří soubor a nabídne ho uživateli ke stažení. Výstupní obrázky pro jednotlivé kategorie jsou umístěny v příloze A.

Mapa je interaktivní, je tedy možné na každou cestu z výstupního souboru kliknout a zobrazit detaily této cesty (hodnoty každé podkategorie). Zároveň je pro každou křižovatku vybrané cesty vypsána navigační instrukce využívající informaci o nejbližším bodu zájmu a názvu cílové ulice. Všechny křižovatky jsou v mapě označeny barevnými body, proto každá vypsaná instrukce obsahuje barevnou značku odpovídající křižovatky. Všechny cesty jsou obousměrné, proto se ve výstupním souboru nacházejí instrukce pro oba možné směry, ve webové aplikaci jsou ale všechny instrukce vypsány jen pro směr od počátečního bodu cesty (v cestě označen písmenem *S*) ke koncovému bodu cesty. Protože zdrojem dat jsou OpenStreetMapy, obsahují detaily i identifikátor cesty, který zároveň slouží jako odkaz na stránku OpenStreetMap.org.

Kromě výše zmíněných knihoven aplikace navíc využívá Material Icons<sup>4</sup>, font Roboto<sup>5</sup> a fotografii od Martina Kopty ze služby Unsplash<sup>6</sup>.

Aplikace je dostupná na adrese <https://prague-ways.netlify.app/>, na obrázku 5.5 je umístěn snímek této webové aplikace.

<sup>1</sup> <https://stuk.github.io/jszip/>

<sup>2</sup> <https://github.com/mpetroff/print-maps>

<sup>3</sup> <https://github.com/eligrey/FileSaver.js>

<sup>4</sup> <https://material.io/resources/icons/>

<sup>5</sup> <https://fonts.google.com/specimen/Roboto>

<sup>6</sup> <https://unsplash.com/photos/shAnqZqffYM>

# Kapitola 6

## Vyhodnocení

Při hledání blízkých objektů každé cesty je nutné určit maximální velikost okolí. Pokud zvolíme příliš malé okolí, nezískáme příliš mnoho informace o okolí dané cesty. Pokud naopak zvolíme příliš velké okolí, některé okolní objekty mohou být až příliš daleko od cesty nebo nemusí být vůbec vidět z dané cesty. Nalezením této vzdálenosti se zabývá sekce 6.1.

Každá cesta z grafu dopravní sítě obsahuje pro všechny zavedené podkategorie hodnotu v rozmezí od 0 do 1. Pro vyhodnocení přesnosti těchto hodnot byly v sekci 6.2 vytvořeny testovací scénáře.

### 6.1 Velikost okolí

Pro správné fungování nástroje je důležité stanovit rozumnou velikost okolí pro hledání objektů poblíž cest. Pro nalezení optimální hodnoty je vybrána Technická ulice<sup>1</sup> v Praze 6.

Vzdálenost [m]	Uzly	Hrany	Polygony
5	1	28	19
50	8	34	35
100	18	34	47
250	42	48	78
500	75	71	123
750	95	78	199
1000	117	86	229

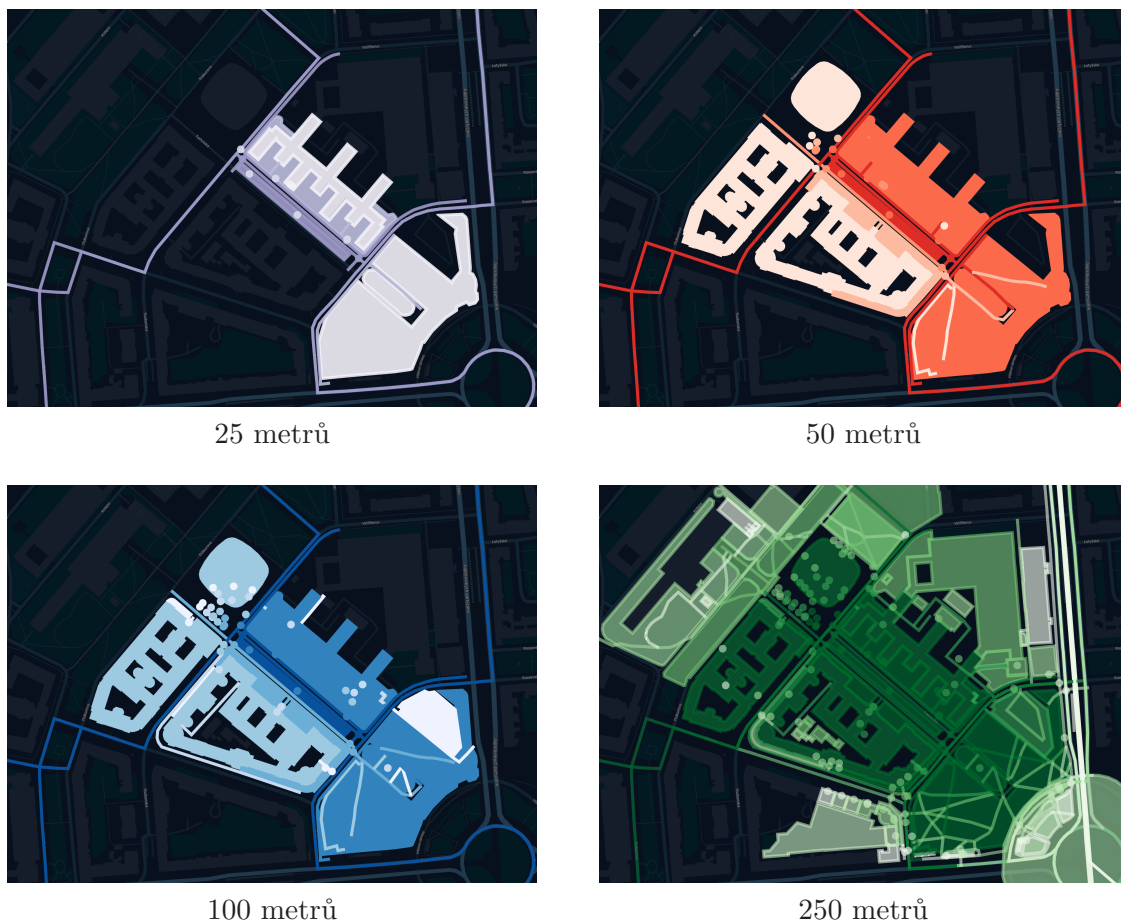
**Tabulka 6.1.** Počet typů okolních objektů pro různé velikosti okolí

V tabulce 6.1 lze samozřejmě pozorovat, že s rostoucí velikostí okolí roste i celkový počet nalezených objektů v okolí. Z tabulky ale nelze vyčíst, která velikost okolí je již příliš velká. Proto jsou všechny zachycené objekty pro různé velikosti okolí znázorněny pomocí Kepler.gl (obrázek 6.1) pro určení optimální velikosti okolí. Při velikosti okolí 25 metrů není zachyceno dostatek okolních objektů, protože mezi okolními objekty chybí např. protější budova. Naopak při velikosti 250 metrů je spousta okolních objektů příliš daleko, některé nejsou z dané ulice ani viditelné. Použitá velikost okolí je tak 50 metrů. Při této vzdálenosti jsou zachyceny všechny důležité okolní objekty, které danou cestu ovlivňují.

### 6.2 Testovací scénáře

Pro vyhodnocení přesnosti vypočtených hodnot bylo náhodně vybráno několik cest (preferovány byly spíše delší cesty) a tyto cesty byly ručně ohodnoceny ve všech podkategoriích (podél zeleně, skrz zeleň atd.).

<sup>1</sup> <https://www.openstreetmap.org/way/8588965>



**Obrázek 6.1.** Vizualizace zachycených objektů pro různé velikosti okolí pomocí Kepler.gl

Pro každou kategorii je vytvořen testovací scénář (zeleň, voda, silnice), pro zjištění funkčnosti detekce překážek je zaveden testovací scénář obsahující cesty, které kolem sebe mají nějaké překážky, a přidán je také testovací scénář pro vyhodnocení navigačních instrukcí. Protože nelze vždy určit očekávané hodnoty s úplnou přesností, odchylka mezi očekávanou a vypočtenou hodnotou může být často způsobena chybou vzniklou při ručním hodnocení cest.

Tyto výsledky jsou zapsány v tabulkách. Každá tabulka v prvním řádku hlavičky obsahuje testované kategorie, ve druhém řádku podkategorie pro kategorie z prvního řádku a první sloupec tabulky tvoří identifikátory z OpenStreetMap. Tabulka obsahuje ve sloupcích číselné hodnoty vyjadřující rozdíl mezi očekávanou hodnotou a vypočtenou hodnotou. Pokud je libovolná hodnota vyšší než 0, je očekávaná hodnota vyšší než vypočtená a naopak. V posledním sloupci tabulky je následně zapsána chyba vypočtená jako suma čtverců hodnot v řádku.

V níže uvedených tabulkách je vždy uvedeno jen 10 cest s nejvyšší chybou seřazených sestupně, všechny testované cesty včetně odchylek jsou uvedeny v příloze B.

## 6.3 Výsledky

Pro všechny kategorie vrací nástroj hodnoty v rámci odchylky od očekávaných hodnot. V každé kategorii jsou rozebrány případy cest, pro které vyšla odchylka (chyba) nejvyšší.

ID	zeleň		chyba
	podél	skrz	
27123916	0,0000	1,0000	1,0000
26392190	-0,2979	0,5296	0,3692
4818889	-0,3453	0,3553	0,2455
165956907	-0,3900	-0,0060	0,1521
141493852	0,3061	0,0000	0,0937
489293163	-0,3000	0,0030	0,0900
261944484	-0,2500	-0,0250	0,0631
259219167	0,2460	-0,0014	0,0605
485938113	-0,2458	0,0010	0,0604
180965351	0,1345	0,1984	0,0575

**Tabulka 6.2.** Tabulka odchylek hodnot pro kategorii zeleň

### 6.3.1 Zeleň

Nejčastější příčinou nepřesností v hodnotách je tag `landuse=village_green` (zmiňeno již v sekci 5.2). V případech cest s ID 26392190, 259219167 a 180965351 dochází ke střídání různých druhů zeleně s různými váhami v průběhu cesty, nelze proto v těchto případech určit očekávanou hodnotu cesty zcela přesně. Další příčinou může být výškový rozdíl mezi cestou a zelení (cesta s ID 141493852), kdy nelze určit, zda je ve skutečnosti z cesty daný objekt viditelný. Chyba u cesty v Trojské zámecké zahradě (ID 4818889) vznikla proto, že vnitřní část zahrady je označena jako `sad`, který není součástí této zámecké zahrady. Cesta s ID 27123916 vede skrze Královskou zahradu, která ovšem není označena pomocí tagu `garden:type` jako veřejná zahrada, nelze proto tento objekt rozlišit od obyčejné rezidenční zahrady.

V případě části ulice Tobruková (ID 489293163) se vpravo od cesty nachází stromy (označeno pomocí `landuse=village_green`) a křoví, podél kterých vede silnice Horoměřická a za ní se nachází les. Protože stromy, křoví ani silnice nepatří mezi překážky, dojde k započítání tohoto lesa mezi viditelnou zeleň. Pokud naopak vede potok skrze les, je velmi často tento potok viditelný (například cesta Nad Dubovým mlýnem<sup>1</sup>) a les je v tomto případě správně vyhodnocen jako objekt, který nepatří mezi překážky.

Výše popsané cesty jsou uvedeny v tabulce 6.2 včetně odchylek hodnot.

### 6.3.2 Vodní plochy

Očekávané hodnoty se ve většině případů příliš neliší od vypočtených hodnot. U cesty na okraji Letenských sadů (ID 25401468) je vidět řeka Vltava, přestože je dále než určený poloměr okolí 50 metrů od cesty. Cesta s ID 27126958 je na okraji vybraného výřezu, proto v databázi okolních objektů chybí většina objektů v okolí této cesty včetně vodní plochy (vodní nádrž). V případě cest vedoucích skrze Stromovku poblíž rybníků (cesty s ID 548739733 a 677306325) se liší hodnoty v určení relativní délky cesty vedoucí mezi nebo podél těchto rybníků.

Chodník vedoucí podél silnice Podbabská a podél Vltavy (ID 115744295), vede také částečně podél Výzkumného ústavu vodohospodářského T. G. Masaryka, který brání ve výhledu na Vltavu, rozdíl mezi očekávanou a vypočtenou hodnotou proto závisí na určení délky úseku cesty vedoucího podél tohoto areálu.

Někdy je naopak cesta vedoucí podél vody natolik zakřivená (například cesta s ID 34260112 nebo 26378083), že není snadné odhadnout očekávané hodnoty dostatečně

<sup>1</sup> <https://www.openstreetmap.org/way/261944484>



ID	voda		
	podél	mezi	chyba
25401468	0,5333	0,0000	0,2844
677306334	-0,5000	0,0000	0,2500
548739733	-0,3000	0,3000	0,1800
677306325	0,2667	-0,2667	0,1422
27126958	0,3000	0,0000	0,0900
26427962	-0,2000	0,0000	0,0400
26378083	-0,1756	0,0759	0,0366
115744295	-0,1532	0,0000	0,0235
34260112	0,1100	0,0000	0,0121
4818889	0,1000	0,0000	0,0100

**Tabulka 6.3.** Tabulka odchylek hodnot pro kategorii vodní plochy

přesně. Cesta s ID 677306334 se od vodního toku vzdaluje postupně, v určitém bodě se tedy cesta dostane do vzdálenosti větší než 50 metrů, kdy nástroj určí, že kolem tohoto bodu není žádná vodní plocha, přestože vodní tok může být stále viditelný. Zde je proto těžké určit, jaká relativní část cesty vede podél vody.

Výše popsané cesty jsou uvedeny v tabulce 6.3 včetně odchylek hodnot.

### 6.3.3 Rušné silnice

ID	podél	silnice		chyba
		mezi	skrz	
561617083	0,4500	0,0000	0,0000	0,2025
31957682	0,2000	0,0000	0,0000	0,0400
141493852	0,0204	0,1796	0,0000	0,0327
260181683	0,1667	0,0000	0,0000	0,0278
562284025	0,1302	0,0000	0,0000	0,0170
115744295	0,1064	0,0000	0,0000	0,0113
132401546	0,0200	0,1000	0,0000	0,0104
485938101	0,1000	0,0000	-0,0100	0,0101
32398652	-0,0163	0,0000	0,0767	0,0062
51277601	-0,0707	0,0000	0,0000	0,0050

**Tabulka 6.4.** Tabulka odchylek hodnot pro kategorii rušné silnice

U rušných silnic se cesty dělí na 3 podkategorie: podél rušných silnic, mezi rušnými silnicemi a skrz rušnou silnici. U chodníku (ID 561617083) vedoucího podél silnice Milady Horákové je očekávaná hodnota v podkategorii *podél silnic* vyšší než vypočtený, protože po této silnici vede i tramvajová trať, přestože se jedná pouze o silnici III. třídy. Podobný případ nastal i u cesty s ID 31957682, která částečně vede podél nábřeží Kapitána Jaroše. Délka tohoto úseku je ovšem zanedbatelná vůči délce celé cesty.

Na koncích cesty s ID 141493852 jsou segmenty vyhodnoceny jako vedoucí mezi silnicemi (vlivem Čechova a Štefánikova mostu), proto se zde hodnoty mírně liší.

U chodníku (ID 260181683) vedoucího podél Evropské ulice se v krátkém úseku nalézá zábradlí (označeno tagem `barrier=fence`), proto je v této části vliv Evropské ulice zanedbán. V tomto případě by proto bylo lepší nepovažovat zábradlí za překážku.

Chodník s ID 115744295 bezpochyby vede podél Podbabské ulice v celé své délce, hodnota podkategorie *podél silnic* by se proto měla rovnat 1. Příčinou této odchylky je chybějící část (konkrétně část Podbabské ulice s ID 46982806) v XML souboru s výřezem Prahy. Soubor byl stažen pomocí služby Overpass API, chyba tedy nejspíše vznikla zde. Ovšem žádný podobný případ se v tomto výřezu nevyskytuje, tento případ je tedy výjimka.

### 6.3.4 Překážky

Případy cest s ID 489293163 a 27123916 jsou již popsány výše v sekci 6.3.1. U cest s ID 27123916, 4818889 a 26427962 je důležité, že jsou od silnic odděleny zdmi, v těchto případech proto detekce překážek funguje podle očekávání.

Cesta V Šáreckém údolí (ID 165956907) vede poblíž lesa, před kterým stojí domy, proto lze očekávat, že tyto domy poslouží jako překážka před lesem. Protože pro detekci překážek je využita nejkratší spojnice mezi cestou a objektem, prochází v tomto případě spojnice těsně mezi domy a les je stále započítán mezi okolní objekty. Řešením této situace by mohlo být použití jiné než nejkratší spojnice mezi cestou a objektem nebo zvýšit mezní vzdálenost mezi spojnicí a překážkou (viz kapitola 5.8).

Cesta Nad Dubovým mlýnem (ID 261944484) vede na jedné straně podél Šáreckého potoka a na druhé straně podél plotu, za kterým se nachází rybník Dubák, který není přes plot vidět, proto cesta nevede mezi vodními plochami.

Cesta vedoucí na Císařském ostrově (ID 95350183) je blízko Vltavy, kolem ní se vyskytuje několik budov a plot (ID 602587741), které brání ve výhledu na Vltavu, proto cesta z tohoto důvodu nevede mezi vodními plochami.

Kolem cesty s ID 26427953 stojí městská zeď (bašta svatého Tomáše), která by měla teoreticky bránit ve výhledu na přiléhající Letenské sady. Protože je ale cesta vyvýšená, městská zeď v tomto případě neplní funkci překážky jako ostatní zábrany.

ID	zeleň		voda		silnice			chyba
	podél	skrz	podél	mezi	podél	mezi	skrz	
489293163	-0,3000	0,0030	0,0000	0,0000	-1,0000	0,0000	0,0000	1,0900
27123916	0,0000	1,0000	0,0000	0,0000	0,0000	0,0000	0,0000	1,0000
4818889	-0,3453	0,3553	0,1000	0,0000	0,0000	0,0000	0,0000	0,2555
165956907	-0,3900	-0,0060	0,1000	0,0000	-0,0060	0,0000	0,0000	0,1622
261944484	-0,2500	-0,0250	0,0100	0,0000	0,0000	0,0000	0,0000	0,0632
115744295	-0,0979	0,0000	-0,1532	0,0000	0,1064	0,0000	0,0000	0,0444
26427962	0,0000	0,0000	-0,2000	0,0000	0,0000	0,0000	0,0000	0,0400
95350183	-0,1087	0,1556	0,0444	0,0000	0,0000	0,0000	0,0000	0,0380
4709428	-0,1457	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0212
26427953	0,0000	0,0000	0,0000	0,0000	-0,1333	0,0000	0,0000	0,0178

**Tabulka 6.5.** Tabulka odchylek hodnot pro cesty s okolními překážkami

### 6.3.5 Navigační instrukce

Pro navigační instrukce bylo vybráno několik dvojic cest společně s jejich křižovatkami. Navigační instrukce fungují podle očekávání pro pojmenované silnice III. třídy s okolními body zájmu. Příkladem je křižovatka ulic Dejvická (ID 8589041) a Kyjevská (ID 4387108) potkávající se na rohu banky ČSOB, proto dojde k vytvoření instrukce „Odbočte vlevo u banky ČSOB do ulice Kyjevská“.



Pokud nepůjdeme po silnici III. třídy, ale například po chodníku nebo jiné cestě, je velmi nepravděpodobné, že tyto cesty budou pojmenované. Příkladem může být chodník (ID 109076017) vedoucí podél ulice Na Hutích, která obsahuje instrukci „Pokračujte rovně u rychlého občerstvení Bistro Kentaur“. Tato instrukce je stále dostatečně srozumitelná, protože obsahuje alespoň bod zájmu (Bistro Kentaur), přesto by jistě mohla obsahovat jméno své přilehlé ulice.

V nejhorsím případě chybí zároveň bod zájmu i název navazující cesty. Příkladem je cesta vedoucí skrze Stromovku (ID 548739733), u které lze vytvořit pouze základní instrukce (např. „Pokračujte rovně“ nebo „Odbočte vpravo“). Těchto případů, kdy chybí obě informace, je bohužel spousta. Řešením by bylo rozšířit skupinu bodů zájmu (o vodní toky apod.), přidat názvy chodníku podle přilehlých ulic nebo vytvářet v takových situacích instrukce tvaru „Za 5 laviček odbočte vpravo“.

V případě ulice Terronská (ID 40314844) je možné pokračovat v 5 různých směrech a některé směry není snadné od sebe rozlišit, proto vznikají stejné instrukce pro různé navazující cesty (například „Odbočte vpravo u školy Základní škola Náměstí Interbrigády“). Tuto situaci lze vyřešit rozšířením navigačních instrukcí (například instrukce „Odbočte mírně vpravo“) nebo upřesněním směru (například instrukce „Odbočte vpravo směrem k zastávce Lotyšská“).

## 6.4 Diskuze

V testovacích scénářích se projevil nedostatek v použití pevně určeného poloměru okolí. Pro objekt vzdálený 49 metrů dojde k započítání tohoto objektu, zatímco objekt ve vzdálenosti 51 metrů je již ignorován. Nezáleží proto, jak daleko od cesty je vybraný objekt, ale záleží pouze na tom, zda je tato vzdálenost menší než 50 metrů.

Dalším omezením mohou být data z OpenStreetMap. Pokud v mapách nějaká informace chybí nebo není přesná, projeví se tento nedostatek i ve zpracovaných datech.

Ve všech výše uvedených scénářích jsou rozebírány pouze cesty s nejvyššími chybami v jednotlivých kategoriích. Mezi vybranými cestami se ale vyskytují také cesty s nulovou nebo zanedbatelnou chybou ve všech podkategoriích, například pro cesty vedoucí skrze park bez okolních silnic nebo vody lze očekávanou hodnotu určit zcela jednoznačně. Průměrná chyba (suma čtverců rozdílů v jednotlivých kategoriích) vychází 0.095 a pro více než 90% případů je celková chyba menší než 0.25, proto lze usoudit, že ve většině případů vychází výsledné hodnoty s dostatečnou přesností s ohledem na možnost lidské chyby při určování očekávaných hodnot (detailní výsledky viz příloha B).

# Kapitola 7

## Závěr

V této práci byl vytvořen nástroj pro automatické obohacování grafu dopravní sítě. Z dat OpenStreetMap byl sestaven graf dopravní sítě s dopravními cestami, které jsou vhodné pro chodce nebo cyklisty. Mezi cestami chybí pěší zóny, které jsou nejčastěji reprezentovány pomocí polygonů. Chodci mají také mnohem více volnosti oproti autům nebo cyklistům. Vytvořený graf dopravní sítě tedy neumožňuje chodcům jít přes trávníky nebo přejít silnice mimo přechody. Tramvajové nebo autobusové zastávky lze také považovat za cesty, proto by měly být i tyto objekty přidány mezi ostatní cesty. Dále jsou některé cesty rozděleny i v částech, kde na sebe navazují pouze 2 cesty. V těchto případech by bylo výhodné takové cesty spojit (kontrakce grafu například pomocí funkce `pgr_contraction`<sup>1</sup> z knihovny `pgRouting`) a mohlo by tak dojít mírnému zrychlení nástroje.

Dále byla vhodně stanovena velikost okolí a pro každou cestu získány všechny okolní objekty. Tyto objekty jsou následně rozděleny do několika vybraných kategorií (zeleně, voda, rušné silnice) a každá cesta je podle těchto kategorií ohodnocena. Každá cesta tak obsahuje informaci o okolní zeleni, vodních plochách a okolních rušných silnicích včetně informace, zda cesta vede podél, mezi nebo skrz dané okolní objekty.

Poté byly stejné okolní objekty využity pro vytváření bohatších navigačních instrukcí v křižovatkách více cest. V těchto instrukcích je zahrnuta informace o nejbližším okolním orientačním bodu a názvu navazující ulice. Možnosti takových instrukcí jsou značně omezené a v mnoha případech se taková instrukce neliší od běžné navigační instrukce, pokud v okolí křižovatky chybí orientační bod nebo cesty nemají přiřazené jméno.

Výstupem nástroje je soubor ve formátu GeoJSON, který obsahuje všechny vybrané hrany z grafu dopravní sítě. Tento soubor je následně vizualizován ve webové aplikaci, ve které lze cesty vykreslovat podle předem určených kritérií (kategorií). Zároveň byla ověřena přesnost a správnost výsledků na několika vybraných cestách a byly diskutovány přednosti a omezení navrženého nástroje.

Výsledné snímky lze nalézt v příloze A.

### 7.1 Další postup

Výsledky této práce lze uplatnit pro zjišťování a vizualizaci kvality ovzduší, které je ovlivněno například množstvím frekventovaných silnic a množstvím zeleně nebo při plánování rozvoje města, kdy lze zkoumat dostupnost různých městských oblastí a určit, které oblasti postrádají městskou zeleně a kde přesně je v budoucnosti potřebné vybudovat třeba nové parky (popsáno v článku *Assessing the spatial distribution of urban parks using GIS* [20]).

Na tuto práci lze navázat městským plánovačem pěších tras (podobně jako např. v práci *A System for Generating Customized Pleasant Pedestrian Routes Based on OpenStreetMap Data* [5]), který lze použít pro vyhledávání cest vedoucích podél zeleně,

<sup>1</sup> <https://docs.pgrouting.org/latest/en/contraction-family.html>

vodních ploch nebo mimo rušné silnice. Protože výsledkem této práce není plánovač cest, je nutné výsledky spojit s běžným plánovačem tras, který plánuje trasy pouze dle informací o cestách (délce cesty, kvalitě povrchu), nebo použít výsledky této práce jako jedno z kritérií pro již hotový plánovač. V takovém případě by uživatel měl možnost zvolit si buď váhy pro jednotlivé kategorie nebo vybrat určitý profil, který bude zahrnovat již vybraný soubor vah pro jednotlivé kategorie. Takovým profilem může být například nejkratší cesta, cesta vedoucí skrze zeleň, vedoucí dále od rušných silnic nebo nejbezpečnější cesta.

Výsledky této práce je možné zpřesnit zpracováním informací z leteckých nebo panoramatických snímků (např. Google Street View, Mapy.cz Panorama nebo služba Mapillary<sup>1</sup>), které lze zpracovávat pomocí strojového učení (například neuronové sítě). Nevýhodou tohoto přístupu by byla vyšší výpočetní náročnost a nedostupnost panoramatických snímků v některých oblastech. Dále je možné výsledky zpřesnit výškovou informací (například z topografických dat SRTM<sup>2</sup>), případně plánovat trasy ve 3D prostoru.

V této práci byly zpracovány pouze objekty z části Prahy, celý výpočet je tak dále možné rozšířit na větší oblast, například pro celé město Prahu.

---

<sup>1</sup> <https://www.mapillary.com/>

<sup>2</sup> <https://www2.jpl.nasa.gov/srtm/>

## Literatura

- [1] Martin Raubal a Stephan Winter. *Enriching Wayfinding Instructions with Local Landmarks*. 2002.  
[http://link.springer.com/10.1007/3-540-45799-2\\_17](http://link.springer.com/10.1007/3-540-45799-2_17).
- [2] Guillem Vich, Oriol Marquet a Carme Miralles-Guasch. Green streetscape and walking. 2019, 12 50-59. DOI 10.1016/j.jth.2018.11.003.
- [3] Stefan Al a Mette Ilene Holmriis. *What happens if you cut down all of a city's trees?* 2020.  
<https://ed.ted.com/lessons/what-happens-if-you-cut-down-all-of-a-city-s-trees-stefan-al>.
- [4] *OpenStreetMap - O projektu*.  
<https://www.openstreetmap.org/about>.
- [5] Tessio Novack, Zhiyong Wang a Alexander Zipf. A System for Generating Customized Pleasant Pedestrian Routes Based on OpenStreetMap Data. *Sensors*. 2018, 18 (11), DOI 10.3390/s18113794.
- [6] Hartwig H Hochmair a Gerhard Navratil. *Computation of scenic routes in street networks*. In: *Geospatial Crossroads@ GI-Forum'08: Proceedings of the Geoinformatics Forum Salzburg*. 2008. 124–133.
- [7] Charlotte Hannah, Irena Spasic a Pdraig Corcoran. A computational model of pedestrian road safety. 2018, 121 347-357. DOI 10.1016/j.aap.2018.06.004.
- [8] Jinjin Yan, Abdoulaye A. Diakit , Sisi Zlatanova a Mitko Aleksandrov. Finding outdoor boundaries for 3D space-based navigation. *Transactions in GIS*. 2020, 24 (2), 371-389. DOI 10.1111/tgis.12613.
- [9] Nina Runge, Pavel Samsonov, Donald Degraen a Johannes Sch ning. *No more Autobahn!*. In: *Proceedings of the 21st International Conference on Intelligent User Interfaces - IUI '16*. New York, New York, USA: ACM Press, 2016. 147-151. ISBN 9781450341370.  
<http://dl.acm.org/citation.cfm?doid=2856767.2856804>.
- [10] Daniele Quercia, Rossano Schifanella a Luca Maria Aiello. *The shortest path to happiness*. In: *Proceedings of the 25th ACM conference on Hypertext and social media - HT '14*. New York, New York, USA: ACM Press, 2014. 116-125. ISBN 9781450329545.  
<http://dl.acm.org/citation.cfm?doid=2631775.2631799>.
- [11] Arpan Sainju a Zhe Jiang. *Mapping road safety features from streetview imagery: A deep learning approach*. 2019.  
<https://arxiv.org/abs/1907.12647>.
- [12] Gregor Joss , Klaus Arthur Schmid, Andreas Z fle, Georgios Skoumas, Matthias Schubert, Matthias Renz, Dieter Pfoser a Mario A. Nascimento. Knowledge extraction from crowdsourced data for the enrichment of road networks. *GeoInformatica*. 2017, 21 (4), 763-795. DOI 10.1007/s10707-017-0306-1.

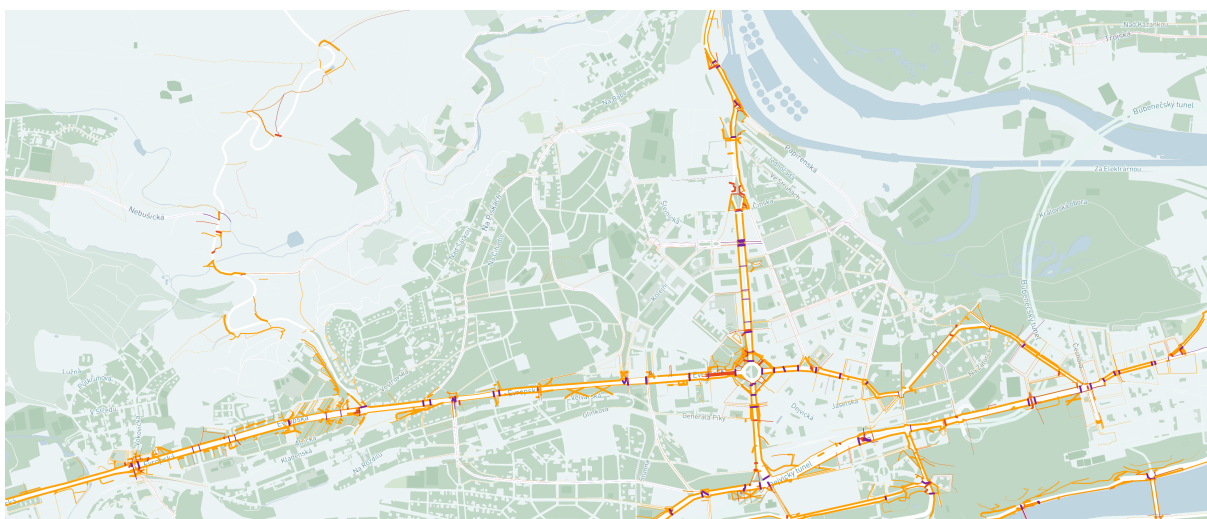
- 
- [13] Peter Zeile, Bernd Resch, Martin Loidl, Andreas Petutschnig a Linda Dörrzapf. *Urban Emotions and Cycling Experience – enriching traffic planning for cyclists with human sensor data*. 2016.  
<https://hw.oeaw.ac.at?arp=0x0033ffa1>.
- [14] Marie Demlová. *Text přednášek z letního semestru 2017/18 v jednom souboru*.  
[http://math.feld.cvut.cz/demlova/teaching/lgr/text\\_lgr\\_2018.pdf](http://math.feld.cvut.cz/demlova/teaching/lgr/text_lgr_2018.pdf).
- [15] *Graph Theory/Definitions*.  
[https://en.wikibooks.org/wiki/Graph\\_Theory/Definitions](https://en.wikibooks.org/wiki/Graph_Theory/Definitions).
- [16] Kai Hormann a Alexander Agathos. The point in polygon problem for arbitrary polygons. *Computational geometry*. 2001, 20 (3), 131–144.
- [17] Patel Amit. *2d Visibility*.  
<https://www.redblobgames.com/articles/visibility/>.
- [18] Eric Weisstein. *Cross Product*.  
<https://mathworld.wolfram.com/CrossProduct.html>.
- [19] *OpenStreetMap Wiki*.  
<https://wiki.openstreetmap.org/>.
- [20] Kyushik Oh a Seunghyun Jeong. Assessing the spatial distribution of urban parks using GIS. *Landscape and Urban Planning*. 2007, 82 (1-2), 25-32. DOI 10.1016/j.landurbplan.2007.01.014.



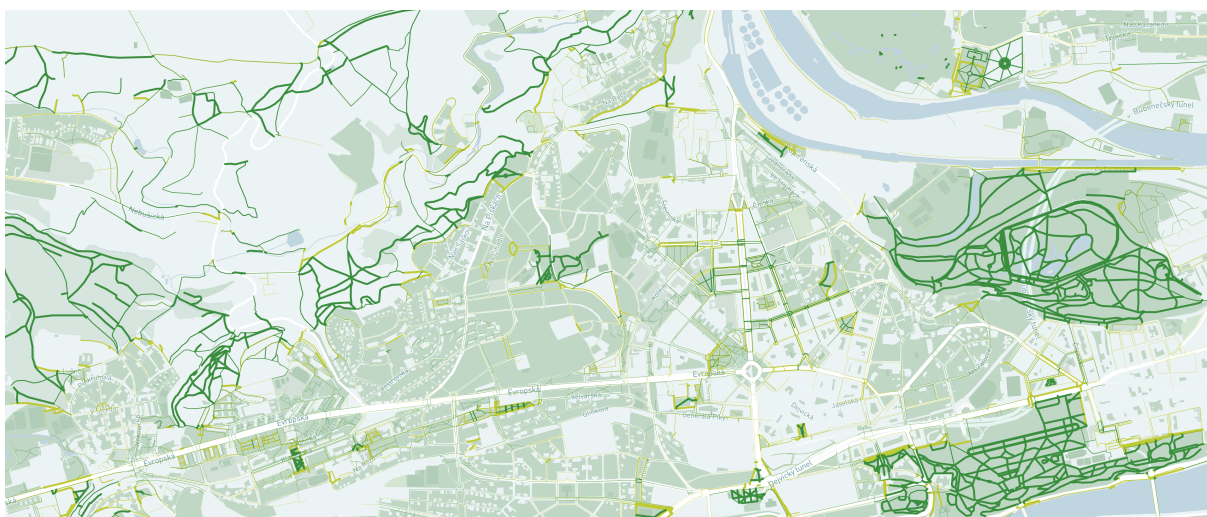
## Příloha A

### Snímky

V této příloze jsou snímky z webové aplikace pro všechny vybrané kategorie. Pro detailní zobrazení doporučujeme navštívit stránku <https://prague-ways.netlify.app/>.



**Obrázek A.1.** Cesty vedoucí ■ podél silnic, ■ mezi silnicemi nebo ■ skrze silnice



**Obrázek A.2.** Cesty vedoucí ■ podél zeleně nebo ■ skrze zeleň



**Obrázek A.3.** Cesty vedoucí ■ podél vodních ploch nebo ■ mezi vodními plochami



# Příloha B

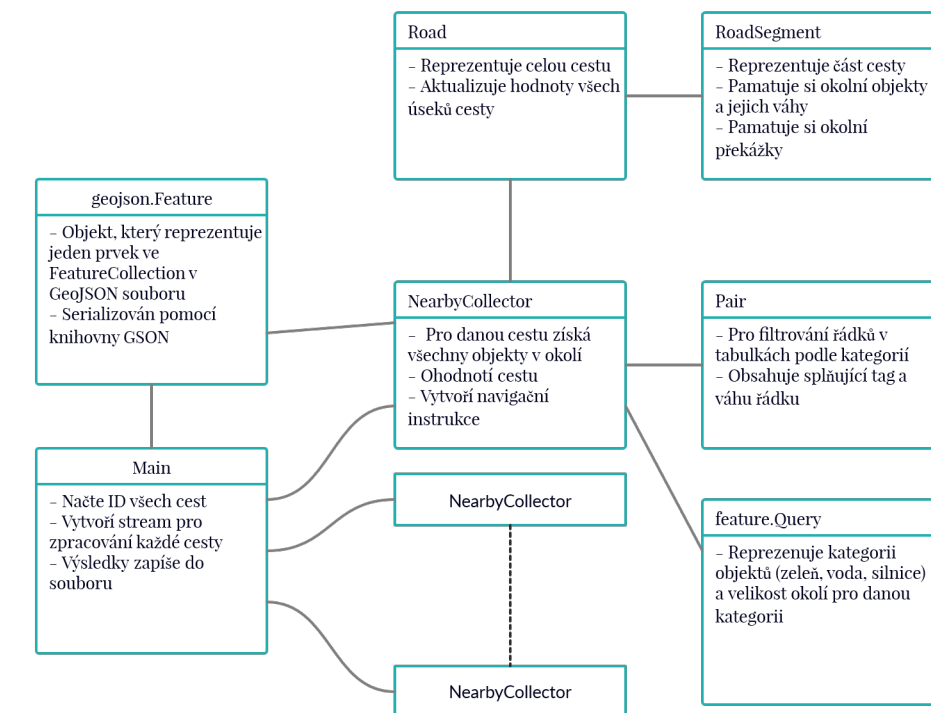
## Testovací případy

ID	zeleň		voda		silnice		chyba	
	podél	skrz	podél	mezi	podél	mezi		
550307899	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
44288821	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
25400106	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
182100634	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
8588962	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
34637560	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
23315134	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
482520327	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	
738125138	-0,0023	0,0003	0,0000	0,0000	0,0000	0,0000	0,0000	
132921507	0,0072	-0,0072	0,0000	0,0000	0,0000	0,0000	0,0001	
109076010	0,0100	-0,0100	0,0000	0,0000	0,0000	0,0000	0,0002	
25400107	0,0000	0,0000	-0,0222	0,0000	0,0000	0,0000	0,0005	
684198969	0,0086	-0,0030	0,0000	0,0000	0,0286	0,0000	0,0009	
360330376	0,0097	0,0071	0,0000	0,0000	-0,0289	0,0000	0,0010	
546733870	-0,0333	0,0000	0,0000	0,0000	0,0000	0,0000	0,0011	
544563898	0,0333	0,0000	0,0000	0,0000	0,0000	0,0000	0,0011	
25400117	-0,0244	0,0244	-0,0100	0,0000	0,0000	0,0000	0,0013	
26378080	0,0000	0,0000	0,0557	0,0000	0,0000	0,0000	0,0031	
8100529	-0,0589	-0,0394	0,0000	0,0000	0,0003	0,0010	0,0050	
482706951	0,0750	0,0000	0,0000	0,0000	0,0000	0,0000	0,0056	
286839321	-0,0651	-0,0282	0,0000	0,0000	0,0230	0,0230	0,0061	
32398652	0,0000	0,0000	0,0000	0,0000	-0,0163	0,0000	0,0062	
8660178	-0,0658	-0,0006	0,0000	0,0000	0,0468	0,0000	0,0065	
42944842	-0,0364	0,0364	-0,0727	0,0000	0,0000	0,0000	0,0079	
4709416	-0,1019	0,0100	0,0000	0,0000	0,0000	0,0000	0,0105	
26427953	0,0000	0,0000	0,0000	0,0000	-0,1333	0,0000	0,0178	
4709319	-0,0729	-0,1171	0,0000	0,0000	0,0000	0,0000	0,0190	
4709428	-0,1457	0,0000	0,0000	0,0000	0,0000	0,0000	0,0212	
360690674	0,1557	0,0043	0,0000	0,0000	-0,0091	0,0000	-0,0009	0,0243
562284025	0,1000	0,0000	0,0000	0,0000	0,1302	0,0000	0,0000	0,0270
260181683	0,0078	-0,0067	0,0000	0,0000	0,1667	0,0000	0,0000	0,0279
8588965	-0,1880	0,0020	0,0000	0,0000	0,0000	0,0000	0,0000	0,0353
95350183	-0,1087	0,1556	0,0444	0,0000	0,0000	0,0000	0,0000	0,0380
147231964	0,1968	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0387
25400091	-0,1983	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0393
26427962	0,0000	0,0000	-0,2000	0,0000	0,0000	0,0000	0,0000	0,0400

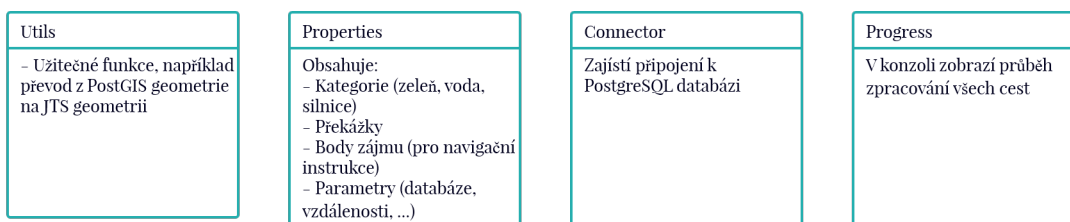
ID	zeleň		voda		silnice			chyba
	podél	skrz	podél	mezi	podél	mezi	skrz	
34260112	0,1202	-0,1250	0,1100	0,0000	0,0000	0,0100	-0,0100	0,0424
115744295	-0,0979	0,0000	-0,1532	0,0000	0,1064	0,0000	0,0000	0,0444
31957682	0,0658	-0,0526	0,0000	0,0000	0,2000	0,0000	0,0000	0,0471
26378083	0,0244	-0,1098	-0,1756	0,0759	0,0000	0,0000	0,0000	0,0492
485938101	0,1226	0,0472	-0,1681	0,0000	0,1000	0,0000	-0,0100	0,0556
25397203	-0,0821	0,1359	-0,1777	0,0000	-0,0018	0,0000	0,0000	0,0568
180965351	0,1345	0,1984	0,0000	0,0000	-0,0136	0,0000	0,0000	0,0577
132401546	0,0950	0,1956	0,0000	0,0000	0,0200	0,1000	0,0000	0,0577
259219167	0,2460	-0,0014	0,0000	0,0000	-0,0067	0,0000	-0,0009	0,0606
485938113	-0,2458	0,0010	0,0000	0,0000	0,0000	0,0100	-0,0100	0,0606
261944484	-0,2500	-0,0250	0,0100	0,0000	0,0000	0,0000	0,0000	0,0632
51277601	-0,0707	-0,0007	-0,2656	0,0000	-0,0707	0,0000	0,0000	0,0806
27126958	0,0757	0,0010	0,3000	0,0000	0,0000	0,0000	0,0000	0,0957
141493852	0,3061	0,0000	0,0000	0,0000	0,0204	0,1796	0,0000	0,1264
677306325	0,0000	0,0000	0,2667	-0,2667	0,0000	0,0000	0,0000	0,1422
165956907	-0,3900	-0,0060	0,1000	0,0000	-0,0060	0,0000	0,0000	0,1622
548739733	0,0000	0,0000	-0,3000	0,3000	0,0000	0,0000	0,0000	0,1800
561617083	0,0000	0,0000	0,0000	0,0000	0,4500	0,0000	0,0000	0,2025
677306334	0,0000	0,0000	-0,5000	0,0000	0,0000	0,0000	0,0000	0,2500
4818889	-0,3453	0,3553	0,1000	0,0000	0,0000	0,0000	0,0000	0,2555
26392190	-0,2979	0,5296	0,0000	0,0000	0,0000	0,0000	0,0000	0,3692
25401468	0,0000	0,0000	0,5333	0,0000	-0,8000	0,0000	0,0000	0,9244
27123916	0,0000	1,0000	0,0000	0,0000	0,0000	0,0000	0,0000	1,0000
489293163	-0,3000	0,0030	0,0000	0,0000	-1,0000	0,0000	0,0000	1,0900

# Příloha C

## Schéma Java tříd



### Statické třídy



### Enum



**Obrázek C.4.** Schéma vytvořené pomocí služby Creately