

**ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ**

KATEDRA VÝKONOVÉ ELEKTRONIKY A STROJŮ

BAKALÁŘSKÁ PRÁCE

**Model elektrických vozidel v mikroskopickém
simulátoru dopravy**

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta elektrotechnická

Akademický rok: 2020/2021

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Jakub TUREK**
Osobní číslo: **E18B0105P**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektrotechnika a energetika**
Téma práce: **Model elektrických vozidel v mikroskopickém simulátoru dopravy**
Zadávací katedra: **Katedra výkonové elektroniky a strojů**

Zásady pro vypracování

Student vybere simulátor dopravy vhodný pro simulace spotřeby energie elektrických vozidel případně jejich dalších elektrických parametrů. Ověří použití simulátoru a jeho výsledků na reálných datech.

1. Proveďte stručné rešerše dopravních simulátorů a zhodnoťte vhodnosti použití pro simulace spotřeby energie elektrických vozidel.
2. Popište model spotřeby elektrických vozidel ve vybraném dopravním simulátoru.
3. Navrhněte vstupní hodnoty parametrů daného modelu pro konkrétní reálné vozidlo, případně navrhněte vylepšení modelu.
4. Ověřte simulované výsledky s reálnými daty.

Rozsah bakalářské práce: **30 – 40 stran**
Rozsah grafických prací: **podle doporučení vedoucího**
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. Sevcik, J. and Prikryl, J., 2019. A Vehicle Device Tailored for Hybrid Trolleybuses and Overhead Wires Implementation in SUMO. EPiC Series in Computing, 62, pp.145-157.
2. Zbývající příslušnou literaturu si student vyhledá v dostupných pramenech podle doporučení vedoucího práce sám.

Vedoucí bakalářské práce: **Ing. Jakub Ševčík**
Regionální inovační centrum elektrotechniky

Datum zadání bakalářské práce: **9. října 2020**
Termín odevzdání bakalářské práce: **27. května 2021**



Prof. Ing. Zdeněk Peroutka, Ph.D.
děkan



Prof. Ing. Václav Kús, CSc.
vedoucí katedry

Abstrakt

Předkládaná bakalářská práce je zaměřena na simulaci elektrických vozidel v mikroskopickém simulátoru dopravy. Konkrétněji se práce zabývá simulací reálného provozu hybridních trolejbusů ve městě Plzeň na lince 12 a 13. K simulaci je použit simulační nástroj Simulation of Urban Mobility (SUMO). Výsledky základní simulace se liší od chování reálných hybridních trolejbusů, proto je tato simulace dále vylepšena. K tomu byl vyvinut skript pro tzv. režim výběhu, jehož výsledky se přibližují chování hybridních trolejbusů v reálném provozu. Na konci práce jsou uvedeny rozdíly mezi jednotlivými simulacemi a jejich porovnání s reálnými daty získané přímo od Plzeňských městských dopravních podniků (PMDP).

Klíčová slova

SUMO, TraCI, Matlab, Plzeň, simulátor, simulační nástroj, model pro elektrická vozidla, model pro hybridní vozidla, hybridní trolejbus, linka 12, linka 13, trakční měnič, vstupní soubory, výstupní soubory, režim výběhu.

Abstract

The presented bachelor thesis is focused on the simulation of electric vehicles in a microscopic traffic simulator. More specifically, the work deals with the simulation of real traffic of hybrid trolleybuses in the city of Pilsen on lines 12 and 13. The simulation tool Simulation of Urban Mobility (SUMO) is used for the simulation. The results of this simulation differ from the behavior of real hybrid trolleybuses, so this basic simulation is improved. To this end, a script respecting so-called coasting mode has been developed, the results of which approximate the behavior of hybrid trolleybuses in real traffic. At the end of the work, the differences between the individual simulations and their comparison with real data obtained directly from the Pilsen City Transport Company (PMDP) are presented.

Key words

SUMO, TraCI, Matlab, Pilsen, simulator, simulation tool, Electric model, ElectricHybrid model, trolleybus, route 12, route 13, substation, input files, output files, coasting mode.

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.



.....
podpis

V Plzni dne 26.5.2021

Jakub Turek

Poděkování

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Jakubu Ševčíkovi za cenné profesionální rady, připomínky, poskytnuté soubory a metodické vedení práce.

Obsah

OBSAH	8
ÚVOD	10
TEORETICKÁ ČÁST	11
1 ELEKTRICKÁ VOZIDLA	12
1.1 POKROČILÉ NÁSTROJE PLÁNOVÁNÍ MĚSTSKÉ HROMADNÉ DOPRAVY	13
1.2 ROZDĚLENÍ SIMULÁTORŮ PROVOZU	14
1.2.1 <i>Multi-Agent Transport Simulation (MATSim)</i>	15
1.2.2 <i>Simulation of Urban Mobility (SUMO)</i>	15
1.2.3 <i>Transportation Analysis and Simulation System (Transims)</i>	15
1.2.4 <i>SimTraffic</i>	16
1.2.5 <i>Výběr vhodného simulátoru</i>	16
2 SUMO – SIMULATION OF URBAN MOBILITY	17
2.1 MODELÝ SIMULÁTORU.....	17
2.1.1 <i>Model pro elektrická vozidla (Electric model)</i>	17
2.1.2 <i>Model pro hybridní vozidla (ElectricHybrid model)</i>	20
2.2 SÍŤOVÁ DATA	22
2.3 SIMULACE	22
2.4 DOPLNĚK TRACI	23
PRAKTICKÁ ČÁST	25
3 VSTUPNÍ SOUBORY SIMULÁTORU	26
3.1 SÍŤOVÉ VSTUPNÍ SOUBORY	26
3.2 SOUBORY PRO DEFINICI VOZIDEL, TYPŮ VOZIDEL A TRAS VOZIDEL	27
3.3 DALŠÍ VSTUPNÍ SOUBORY	28
3.4 KONFIGURAČNÍ SOUBOR	29
4 POPIS SIMULACE A ZPRACOVÁNÍ VÝSLEDKŮ	30
4.1 VÝSTUPNÍ SOUBORY A JEJICH ZPRACOVÁNÍ DO SOUBORU.....	31
4.2 ZPRACOVÁNÍ VÝSTUPNÍCH DAT	33
4.2.1 <i>Zpracování výstupních dat z trakčních měření</i>	34
4.2.2 <i>Zpracování výstupních dat z hybridních trolejbusů</i>	37
5 VYLEPŠENÍ SIMULACE	43
5.1 OPRAVA SÍTĚ	43
5.2 REŽIM VÝBĚHU	45
5.2.1 <i>Konkrétní realizace režimu výběhu</i>	45
5.2.2 <i>Porovnání výsledků z hybridních trolejbusů s předchozí simulací</i>	49
5.2.3 <i>Porovnání výsledků z trakčních měření s předchozí simulací</i>	54
5.2.4 <i>Porovnání výsledků z trakčních měření s předchozí simulací</i>	55
5.3 POROVNÁNÍ VÝSLEDKŮ S REÁLNÝMI DATY	56
ZÁVĚR	59
SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ	61
SEZNAM OBRÁZKŮ	63
SEZNAM TABULEK	64

PŘÍLOHY	1
PŘÍLOHA A – SKRIPT PRO SIMULACI BEZ REŽIMU VÝBĚHU POMOCÍ TRACI	1
PŘÍLOHA B – SKRIPT PRO SIMULACI REŽIMU VÝBĚHU POMOCÍ TRACI	2
PŘÍLOHA C – SKRIPT PRO VÝPOČET POŽADOVANÉHO ZPOMALENÍ PRO REŽIM VÝBĚHU	5

Úvod

Tato bakalářská práce se zabývá simulací elektrických vozidel, konkrétněji hybridních vozidel, v mikroskopickém simulátoru dopravy. Simulace je zaměřena na provoz hybridních trolejbusů linky 12 a 13 ve městě Plzeň. Simulace bude simulována ve dvou režimech, které jsou následně porovnány s reálnými naměřenými daty. Tyto data byly poskytnuty přímo od Plzeňských městských dopravních podniků (PMDP).

Text je rozdělen do pěti kapitol. První dvě kapitoly se zabývají teorií, která je potřebná pro praktickou část práce. První z nich se zabývá základní charakteristikou a vlastnostmi elektrických vozidel a následně jednotlivými simulátory dopravy, které jsou rozděleny do několika skupin. Tato kapitola je zakončena vhodným výběrem simulačního nástroje pro tuto bakalářskou práci. V druhé části práce jsou popsány základní principy a vlastnosti vybraného simulačního nástroje, kterým je Simulation of Urban Mobility (SUMO). Dále jsou zde obecně popsány veškeré vstupní soubory, které je nutné simulátoru dopravy dodat.

Následuje praktická část této bakalářské práce. Ta je rozdělena do tří kapitol. První z nich, tedy třetí kapitola práce, se zabývá konkrétními vstupními soubory, které jsou použity pro simulaci dopravy hybridních trolejbusů ve městě Plzeň. Je vysvětleno, k čemu soubory slouží a jaké informace obsahují. Součástí této kapitoly je i popsání veškerého nastavení dané simulace. Ve čtvrté kapitole jsou naopak popsány výstupní soubory ze simulátoru dopravy. Tyto výstupní soubory jsou následně zpracovány pomocí softwaru Matlab a poté i graficky vykresleny. Na konci této kapitoly je poukázáno na některé nepřesnosti této simulace. Tyto nepřesnosti jsou odstraňovány v poslední páté kapitole práce. Tato kapitola je zaměřena na úpravu výškopisů sítě města Plzeň, a především na přiblížení výsledků simulací blíže k reálnému chování hybridních trolejbusů. K tomu je vyvinut skript pro tzv. režim výběhu, který upravuje podobu jízdního profilu hybridních trolejbusů. Následně jsou výsledky z jednotlivých simulací porovnány a rozdíly vysvětleny. Na konci kapitoly jsou výsledky ze simulací porovnány s naměřenými daty, které obsahují informace o reálných průjezdech hybridních trolejbusů po lince 13.

TEORETICKÁ ČÁST

1 Elektrická vozidla

Elektrická vozidla (*EV*, z anglického *Electric Vehicles*) jsou motorová vozidla, která využívají ke svému pohonu elektrickou energii. Elektromobily mohou obsahovat jeden, ale i více elektrických motorů. Elektrická energie se uskládá nejčastěji v bateriích, existují však alternativy jako vodíkové palivové články. Elektrická vozidla lze rozdělit na tři skupiny. První z nich jsou elektromobily, které mají pouze elektromotor a energii k pohonu získávají z baterie. Označují se zkratkou *BEV* (*Battery Electric Vehicle*). V baterii se ukládá energie získaná především z dobíjecích stanic. Druhou skupinou jsou hybridní elektromobily, tedy *HEV* (*Hybrid Electric Vehicle*). Tyto elektrická vozidla využívají kombinaci klasického spalovacího motoru s elektromotorem. Hybridní elektromobily lze ještě rozdělit dvěma způsoby. Prvním z nich je dělení podle uspořádání hnacího ústrojí na sériové, paralelní a kombinované. Druhé dělení je podle stupně hybridizace na micro, mild, full a plug-in hybridy [6]. Třetí skupinou jsou elektrická vozidla s palivovými články se zkratkou *FCEV* (*Fuel Cell Electric Vehicle*). Dnes tyto elektromobily využívají většinou palivové články na vodík. Všechny tyto typy elektrických vozidel nezatěžují životní prostředí emisemi škodlivých plynů. Další výhodou elektromobilů je jejich nízká hlučnost a menší nároky na údržbu. Významnou sledovanou vlastností elektrických vozů je jejich dojezd čistě na elektrický pohon. Snahy o prodloužení tohoto dojezdu jsou nedílně provázány se zvyšováním účinnosti využívání elektrické energie vzhledem ke kapacitě baterie, výkonu elektromotoru, celkové hmotnosti vozu či aerodynamice a podobně.

Výzkum elektrických vozidel musí brát v úvahu spoustu důležitých věcí, vlastností a technických parametrů. Důraz je kladený například na návrh elektrické soustavy vozu, její analýzy a řízení tohoto systému. Řízením systému se snažíme o řízení elektrické energie pro zvýšení energetické úspornosti, účinnosti a životnosti baterie elektromobilu. To v praxi znamená spoustu času stráveného vývojem a laděním elektrické soustavy vozu. Zefektivněním této soustavy můžeme dosáhnout většího dojezdu na elektrický pohon.

Důležitý je také možný dopad elektrických vozidel na elektrickou distribuční síť, kdy elektrická vozidla budou hrát pravděpodobně důležitou roli při integraci obnovitelného zdroje energie do distribuční elektrické sítě. Příkladem může být ukládání energie v dobách nadměrné výroby a poskytování energie zpět do sítě během doby špičkového zatížení. Také obousměrné charakteristiky elektrických vozidel představují zvláštní výzvy a příležitosti,

tedy kromě spotřeby energie mají také schopnost poskytovat energii do sítě po omezenou dobu, například rekuperací. Společný dopad elektrických vozidel na distribuční elektrickou síť bude záviset na mnoha faktorech, včetně jmenovitých hodnot, doby nabíjení (mimo špičku nebo na vrcholu), rychlost a schéma nabíjení, rozmístění elektrických vozidel v elektrické síti, jejich stav nabití atd. Naopak, nedostačující nebo špatná správa elektrických vozidel by mohla mít negativní dopady na kvalitu dodávky energie a spolehlivost sítě **0**. Negativní dopady by mohly zahrnovat nestabilitu napětí, harmonické zkreslení, nevyváženost zatížení a přetížení elektrické rozvodné sítě.

Z důvodu elektrifikace automobilového průmyslu vznikly nové požadavky na vývoj vozidel. Vznikají proto nové nástroje a simulační programy, přes které se můžeme pokusit optimalizovat celou elektrickou soustavu vozidla, tedy energetickou účinnost, výkon, řízení teploty a podobně, a zároveň ekonomický přínos elektrických vozidel v systému dodávky elektřiny. Dále také urychlíme dodání elektromobilů na trh.

1.1 Pokročilé nástroje plánování městské hromadné dopravy

Asi je jasné, že elektrifikace automobilového průmyslu se nevyhne ani elektrifikaci městské hromadné dopravy. Jako náhrada za autobusy se spalovacími motory se nabízí použití hybridních trolejbusů, které kombinují napájení z troleje a z baterií. K tomuto použití je nutné mít vybudovanou určitou infrastrukturu s vysokou mírou zatrolejování. K tomu je vhodná třeba právě Plzeň. Je zřejmé, že to není možné bez úpravy linek a celé infrastruktury hromadné dopravy. Jako další alternativa se nabízí plně elektrické autobusy, které využívají energii pouze z baterií. Toto řešení by zvýšilo energetickou účinnost dopravy, ale bohužel s ním přichází řada problémů. Jedním z nich je právě samotná baterie a její kapacita, kvůli tomu by bylo nutné vybudovat potřebnou infrastrukturu nabíjecích stanic. Mezi další řešení by patřily trolejbusy zcela závislé na trakčním vedení. Toto řešení také přináší spoustu problémů, jako špatná ekonomika provozu na méně vytižených linkách nebo problémy při mimořádnostech. Plně elektrická a hybridní vozidla MHD, včetně hybridních trolejbusů, jsou již vyráběna a provozována, od konce roku 2017 dochází k masovému nasazování těchto vozidel.

Pro úspěšné nasazení elektrických autobusů a hybridních trolejbusů je potřeba optimalizovat parametry vozidla, zvláště způsob akumulace energie, kapacitu baterie na palubě vozidla jako kompromis mezi hmotností vozidla, jeho dojezdem, dynamikou a životností akumulčních prvků. Dále je zapotřebí optimalizovat parametry infrastruktury, zejména míru zatrolejování, dimenzování trolejového vedení a nabíjecích stanic a jejich ideální rozložení. K tomu, aby byla zajištěna dopravní obslužnost s plně elektrifikovanou městskou hromadnou dopravou (MHD) a zároveň byly náklady na pořízení a na samotný provoz elektrické dopravní dopravy co nejnižší, je třeba udělat návrh správné konfigurace vozidel a trakční infrastruktury. Je nezbytné před nasazením do zkušebního provozu otestovat konfiguraci vozidla v některém softwarovém nástroji, kde můžeme upravovat různé parametry, jako například počet cestujících, hustotu provozu, výškový profil atd. Zároveň můžeme otestovat danou konfiguraci vozidla v předpokládaných výjimečných případech jako je uzavírka, dopravní nehoda, výpadek napájení a podobně. Každý simulátor provozu je trochu odlišný a umí něco jiného, proto jsem provedl analýzu dostupných simulačních nástrojů z hlediska jejich použitelnosti pro simulaci elektrických vozidel [18].

1.2 Rozdělení simulátorů provozu

Simulátory provozu lze rozdělit na makroskopické, mezoskopické, mikroskopické a metaskopické (nanoskopické). Dělení závisí na tom, jak detailně jednotlivé vozidlo simulujeme. Makroskopické simulují pouze dopravní proudy, mikroskopické simulují každé vozidlo, mezi nimi leží mezoskopické simulátory. Poslední, metaskopické simulátory, navíc přidávají detailní informace o stavu vozidla. Simulátor, který bude využit pro tuto bakalářskou práci, musí být schopný simulovat interakce elektrického vozidla s okolním provozem a s napájecí infrastrukturou včetně komunikace. Také musí dostatečně detailně simulovat danou oblast, proto musí být dostatečně výkonný. Simulátor, který tyto vlastnosti splňuje a zároveň bude tedy použit, je simulátor mikroskopický. Existuje řada open-source nástrojů pro simulaci provozu, proto jsem vybral ty nejpoužívanější z nich. Mezi ně patří nástroj Multi-Agent Transport Simulation (MATSim) [2], dále Simulation of Urban Mobility (SUMO) [3], Transportation Analysis and Simulation System (Transims) [4] a nakonec nástroj SimTraffic [5].

1.2.1 Multi-Agent Transport Simulation (MATSim)

Mezoskopický simulační nástroj, který je vyvíjen zejména na ETH Zurich a TU Berlin ve spolupráci s dalšími subjekty. Je to simulátor v prostředí JAVA. Simuluje chování chodců, vozidel a hromadné dopravy, slouží především k simulaci každodenních aktivit obyvatel a k analýze systémů veřejné hromadné dopravy. Kinematický model bere v úvahu polohu a rychlost vozidla, přičemž spotřeba energie je počítána z průměrné rychlosti vozidla. Světelná signalizační zařízení jsou nastavena pevnými časovými úseky. Výsledky lze zobrazovat na Google Earth. V roce 2014 byla dodána podpora elektrických a hybridních vozidel a také je možné definovat dobíjecí místa nebo úseky.

1.2.2 Simulation of Urban Mobility (SUMO)

Mikroskopický a mezoskopický simulační nástroj vyvíjený od roku 2001 německým Centrem pro kosmický výzkum (DLR) v prostředí C++. Simuluje chování chodců, vozidel a hromadné dopravy, umožňuje zpracování velkých sítí. Kinematický model bere v úvahu polohu a rychlost vozidla a zároveň také sklon trasy. Spotřeba energie je počítána z rychlosti a zrychlení vozidla. Světelná signalizační zařízení jsou nastavena pevnými časovými úseky nebo řízena v dopravně-závislém signálním plánu, jehož struktura bohužel není stejná se strukturou používanou v České republice. Výsledky lze vizualizovat ve vlastním grafickém rozhraní. Je možné definovat dobíjecí místa nebo úseky. Výhodou je, že rychlost dobíjení můžeme každému vozidlu nastavit zvlášť, jak pro stání, tak i pohyb vozidla. Vozidla ale bohužel nejsou vybavena zpětnou vazbou, takže i po vybití baterie je schopno se vozidlo pohybovat.

1.2.3 Transportation Analysis and Simulation System (Transims)

Mikroskopický simulační nástroj vyvíjený od roku 2006 komunitou s podporou Federal Highway Administration. Stejně jako SUMO je tento nástroj v prostředí C++. Opět simuluje chování chodců, vozidel a hromadné dopravy. Zaměřuje se na modelování aktivit jednotlivých obyvatel a propojení obyvatel se systémem hromadné dopravy. Kinematický model bere v úvahu polohu, zrychlení a rychlost vozidla, ale počítá také i se sklonem trasy a maximální povolenou rychlostí v daném úseku. Spotřeba energie je počítána z rychlosti

a zrychlení vozidla. Světelná signalizační zařízení jsou řízena pevnými časy nebo délkou fronty. Výsledky lze vizualizovat ve vlastním grafickém rozhraní.

1.2.4 SimTraffic

Mikroskopický simulační nástroj vyvíjený od roku 2005 společností Trafficware. SimTraffic je sada optimalizačních nástrojů pro simulaci městské dopravy. Simuluje chování chodců, vozidel a hromadné dopravy. V tomto simulačním nástroji jsou jednotlivá vozidla modelována a zobrazována procházející pouliční sítí. Na rozdíl od mnoha jiných modelovacích aplikací se během provádění simulace zobrazuje animace SimTraffic. Výsledky lze zobrazit pomocí 3D prohlížeče.

1.2.5 Výběr vhodného simulátoru

Je těžké se rozhodnout mezi těmito simulačními nástroji, jelikož si jsou velmi podobné. Simulační nástroj Multi-Agent Transport Simulation (MATSim) hned zpočátku vyřazují, protože spotřeba energie se počítá jen z průměrné hodnoty. Navíc je tento nástroj v prostředí JAVA, ve kterém nejsem vůbec zblhlý. Simulační nástroj SimTraffic mi přišel příliš složitý a nejspíše by mi nevyhovoval, proto ho k mé práci také nepoužiji. Simulátory Simulation of Urban Mobility (SUMO) a Transportation Analysis and Simulation System (Transims) jsou si velice podobné. K mojí bakalářské práci jsem si vybral simulační nástroj Simulation of Urban Mobility (SUMO) především kvůli jeho rozhraní, ovládní, jeho zaměření na provoz, ale také i kvůli recenzím.

2 SUMO – Simulation of urban mobility

SUMO je open source software pro simulaci provozu. Je k dispozici od roku 2001 a vyvíjen především Německým střediskem pro letectví a kosmonautiku (DLR). Od té doby se SUMO vyvinulo v plně vybavenou sadu nástrojů pro modelování dopravních systémů včetně silničních vozidel, veřejné dopravy a chodců. Součástí softwaru je spousta podpůrných nástrojů, které automatizují klíčové úlohy pro vytvoření, provedení a vyhodnocení simulace provozu [11]. Příkladem může být importování sítě, výpočty tras, vizualizace a výpočet emisí. Software lze rozšířit o vlastní modely a také poskytuje rozhraní pro dálkové ovládání simulace například pomocí rozšíření TraCI.

Aby bylo možné simulovat provoz, musíme dodat simulátoru několik nezbytných vstupů. Mezi ty nejdůležitější patří síťová data obsahující informace o silnicích a chodnicích, dodatečné dopravní infrastruktury jako křižovatkách, semaforech a podobně a dopravní zátěž. Tyto prvky tvoří společně simulační scénář. Proces přípravy simulačního scénáře může být založený na reálných datech a je obvykle časově náročný. SUMO poskytuje velký balíček aplikací, které se snaží zjednodušit či zautomatizovat jednotlivé procesy.

2.1 Modely simulátoru

Simulátor SUMO používá určité modely pro simulaci různých typů vozidel. Mezi tyto modely patří model pro elektrická vozidla (*Electric model*), dále model pro hybridní vozidla (*ElectricHybrid model*) a emisní model (*Emissions model*). Jelikož se tato práce týká simulace hybridních trolejbusů, je pro nás důležitý model pro hybridní vozidla. Tento model vychází z modelu pro elektrická vozidla, proto zmíním oba dva tyto modely.

2.1.1 Model pro elektrická vozidla (*Electric model*)

V poslední době sledujeme několik globálních trendů ohledně mobility, které vedou k prosazování alternativních technologií pohonu. Stále více se setkáváme s použitím elektrické energie v automobilech či dopravních prostředcích. Proto byly implementovány elektrické modely i do simulátorů dopravy [13]. Pro definování elektrického vozidla musí

být vozidlo vybaveno tzv. *battery.device*, tedy nějakou baterií. Následně je vozidlo specifikováno pomocí parametrů vozidla. Tyto parametry jsou shrnuty v *Tab. 2.1*.

Tab. 2.1: Elektrické parametry vozidla

Parametr	Výchozí hodnota	Popis parametru
maximumBatteryCapacity	0 (Wh)	Maximální kapacita baterie E_{Max}
actualBatteryCapacity	0 (Wh)	Počáteční nastavení kapacity baterie
maximumPower	0 (W)	Maximální výkon, kterého může vozidlo dosáhnout
vehicleMass	1000 (kg)	Hmotnost vozidla m_{veh}
frontSurfaceArea	5 (m ²)	Čelní povrchová plocha A_{veh}
airDragCoefficient	0,6	Součinitel odporu vzduchu c_W
internalMomentOfInertia	0,01 (kg)	Relativní hmotnost rotačních částí vozidla m_{rot}
radialDragCoefficient	0,5	Součinitel radiálního odporu c_{rad}
rollDragCoefficient	0,01	Koeficient valivého odporu c_{Roll}
constantPowerIntake	100 (W)	Průměrný (konstantní) výkon vozidla P_{Const}
propulsionEfficiency	0,9	Efektivita pohonu η_{prop}
recuperationEfeficiene	0,8	Rekupační účinnost η_{recup}
stoppingTreshold	0,1 (km/h)	Minimální rychlost pro zahájení nabíjení

Změnu energie jednoho vozidla lze vypočítat sečtením jednotlivých změn energie. Konkrétněji to znamená, že energii vozidla lze vypočítat sečtením změny kinetické energie, potenciální energie a rotační energie z jednoho časového kroku do následujícího kroku [8]. energii vozidla $E_{veh}[k]$ v diskrétním časovém kroku k lze vypočítat pomocí rovnice 2.1. Známými proměnnými jsou hmotnost vozidla m , variabilní časová rychlost vozidla $v[k]$, gravitační zrychlení g , variabilní časová nadmořská výška vozidla $h[k]$ a relativní hmotnost rotačních částí m_{rot} .

$$E_{veh}[k] = E_{kin}[k] + E_{pot}[k] + E_{rot,int}[k] =$$

$$= \frac{1}{2} \cdot m \cdot v^2[k] + m \cdot g \cdot h[k] + \frac{1}{2} \cdot m_{rot} \cdot v^2[k] \quad 2.1$$

Energetické ztráty $\Delta E_{loss}[k]$ jsou způsobené odporem vzduchu, valivým odporem, odporem při jízdě po křivkové trajektorii (při jízdě zatáčkou) a konstantní spotřebou [10]. Konstantní spotřebou je myšleno to, že i kdyby vozidlo stálo na místě, stále má nějakou

vlastní konstantní spotřebu energie, například spotřebu nezbytných systémů či topení (klimatizace). Energetický zisk mezi časovými kroky k a $k+1$ můžeme vypočítat rovnice 2.2.

$$\Delta E_{gain}[k] = E_{veh}[k+1] - E_{veh}[k] - \Delta E_{loss}[k] \quad 2.2$$

$$\Delta E_{loss}[k] = \Delta E_{air}[k] + E_{roll}[k] + \Delta E_{curve}[k] + \Delta E_{const}[k] \quad 2.3$$

$$\Delta E_{air}[k] = \frac{1}{2} \cdot \rho_{air} \cdot A_{veh} \cdot c_w \cdot v^2[k] \cdot |\Delta s[k]| \quad 2.4$$

$$\Delta E_{roll}[k] = c_{roll} \cdot m \cdot g \cdot |\Delta s[k]| \quad 2.5$$

$$\Delta E_{curve}[k] = c_{rad} \cdot \frac{m \cdot v^2[k]}{r[k]} \cdot |\Delta s[k]| \quad 2.6$$

$$\Delta E_{const}[k] = P_{const} \cdot \Delta t \quad 2.7$$

Jak bylo nastíněno a je vidět v rovnici 2.3, energetické ztráty jsou tvořeny 4 složkami. První složkou jsou energetické ztráty způsobené odporem vzduchu a jejich výpočet je uveden v rovnici 2.4. Ty jsou závislé na hustotě vzduchu ρ_{air} , čelní ploše vozidla A_{veh} , koeficientu odporu vzduchu c_w . Mezi další ztráty patří ztráty způsobené valivým odporem kol. Jejich výpočet je uveden v rovnici 2.5. Tyto ztráty jsou závislé na valivém součiniteli odporu c_{roll} . Výpočet ztrát způsobené jízdou po křivkové trajektorii je uveden v rovnici 2.6. Tyto ztráty jsou závislé na dostředivé síle F_{rad} a součiniteli odporu při jízdě po křivkové trajektorii c_{rad} . Posledními ztráty jsou ztráty konstantní spotřebou. Jejich výpočet je uveden v rovnici 2.7 a závisí na konstantním výkonu pomocných pohonů P_{const} .

Množství energie, kterou vozidlo spotřebovalo nebo naopak získalo zpět, je závislé na znaménku energetických ztrát. Množství spotřebované, respektive rekuperované energie je dále ovlivněno účinností jednotlivých výkonových prvků (baterie, pohonná jednotka apod.) hybridního vozidla. Tyto ztráty jsou pro jednoduchost reprezentovány konstantní účinností pro rekuperaci η_{recup} a konstantní účinností pro pohon η_{prop} . Pokud je energetický zisk $\Delta E_{gain}[k]$ kladný, vozidlo zrychluje a změna energie baterie se vypočte

pomocí rovnice 2.9. Naopak pokud je energetický zisk $\Delta E_{gain}[k]$ záporný, vozidlo brzdí, dochází k rekuperaci a změna energie baterie se vypočte podle rovnice 2.8.

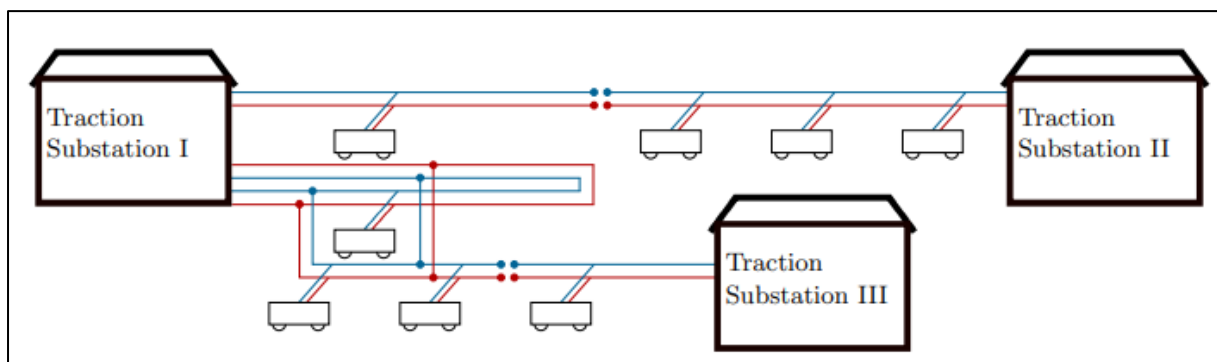
$$\Delta E_{bat}[k + 1] = \Delta E_{bat}[k] + \Delta E_{gain}[k] \cdot \eta_{recup} \quad 2.8$$

$$\Delta E_{bat}[k + 1] = \Delta E_{bat}[k] + \Delta E_{gain}[k] \cdot \eta_{prop}^{-1} \quad 2.9$$

2.1.2 Model pro hybridní vozidla (ElectricHybrid model)

Tento model rozšiřuje předchozí elektrický model o simulaci vnitřních elektrických parametrů vozidla. Mezi ně patří *overheadWireChargingPower* a *maximumPower*. Zdá se, že parametr *maximumPower* je totožný s elektrickým modelem, ale tento parametr je trochu rozdílný a udává maximální výkon elektrického pohonu vozidla. Hybridním vozidlem je například trolejbus, který je napájen trolejovým vedením a zároveň obsahuje i baterii, která je pomocí tohoto vedení nabíjena. Model umožňuje také simulovat chování obvodu trolejového vedení [14].

K simulaci hybridních trakčních vozidel musí být v SUMO implementovány tři základní objekty. Těmito objekty jsou trakční měnič, trolejové vedení a hybridní vozidlo (trolejbus) [9]. Trakčních měničů je obvykle několik, kde každá měnič může poskytovat energii pro více než jeden úsek vedení. Princip vztahů mezi těmito objekty můžeme vidět na následujícím Obr. 2.1.



Obr. 2.1: Princip vztahů elektrické sítě a hybridních trolejbusů

2.1.2.1 Trakční měnič

Základní funkcí trakční rozvodny je převod třífázové soustavy z distribuční sítě na stejnosměrnou soustavu pro napájení trolejových vedení. Současně se v elektrické síti veřejné dopravy města Plzeň používá stejnosměrné napětí o hodnotě 600 V a stejnosměrný proud. Zároveň tyto rozvodny poskytují bezpečné používání této sítě, příkladem může být nadproudová ochrana, protože příliš vysoký proud může způsobit poškození trolejového vedení. S novými moderními systémy by mohla měnič také komunikovat s vozidly k vyvážení spotřeby energie připojených vozidel.

2.1.2.2 Trolejové vedení

Elektrický proud je do trolejbusů veden z trakční rozvodny přes trolejové vedení, přes dvojici trolejových drátů. Trolejové vedení je rozděleno na určité sekce. Každá sekce je napájena trakční rozvodnou a je připojena k sousední úsekům pomocí nevodivých izolátorů. Elektrický proud tedy nemůže protékat z jedné sekce do druhé. V simulátoru je daná sekce ještě ze simulačních důvodů dále rozdělena na menší úseky, na tzv. segmenty. Sekce je tedy rozdělena na segmenty odpovídající jízdním pruhům, nad kterými jsou trolejové dráty vedeny.

2.1.2.3 Hybridní vozidlo

Toto vozidlo využívá sběrač pro čerpání energie z trolejového vedení. Zařízení těchto vozidel musí být schopno zajistit základní strategie řízení napájení, které ovlivňují energetické toky ve vozidle. Tím dosáhneme toho, že čerpání veškeré energie není prováděno přes baterii vozidla. To by mělo negativní vliv na její životnost. Pro definování hybridního elektrického vozidla musíme opět specifikovat parametry vozidla a typ vozidla. Obecné parametry vozidla jsou stejné s parametry elektrického modelu uvedené v *Tab. 2.1*. Níže, v *Tab. 2.2*, jsou uvedené rozšiřující parametry elektrického modelu o parametry pro hybridní vozidlo. Konkrétní hodnoty těchto parametrů budou definovány v praktické části bakalářské práce (str. 25).

Tab. 2.2: Rozšiřující elektrické hybridní parametry vozidla

Parametr	Výchozí hodnota	Popis parametru
overheadWireChargingPower	0 (W)	Konstantní výkon čerpaný z trolejového drátu pro nabíjení baterie
maximumPower	100000 (W)	Maximální výkon elektrického pohonu vozidla

2.2 Síťová data

Bez vytvoření sítě by neproběhla žádná simulace. Pro její vytvoření existují dvě možnosti. První z nich je manuální vytvoření sítě. K tomu slouží síťový nástroj *NETEDIT*. V tomto nástroji můžeme vytvářet a upravovat základní síťové prvky jako jízdní pruhy, křižovatky, ale také i semaforey, autobusové zastávky apod. Tato cesta je vhodná pro vytvoření malé sítě, jelikož její tvorba zabere spoustu času. Pro větší a rozlehlé sítě je mnohem lepší využít druhé možnosti, a to import sítě z různých zdrojů dat. K tomu slouží nástroj *NETCONVERT*. Zdrojem dat může být například OpenStreetMap [12]. Síť, která je importována pomocí tohoto nástroje, může obsahovat určité chyby. Proto je dobré tuto importovanou síť ještě dodatečně upravit v nástroji *NETEDIT*. Proto se tento nástroj stále více vylepšuje, aby se minimalizovali vznikající chyby.

2.3 Simulace

Každé vozidlo v simulaci je definováno identifikátorem (jménem), časem odjezdu a danou trasou v síti. Tyto informace můžeme doplnit více detailním popisem. Lze například určit jízdní pruh, maximální rychlost atd. Ke každému vozidlu může být přiřazen typ vozidla, který popisuje fyzikální vlastnosti jako například hmotnost vozidla. Lze také přiřadit jednu z dostupných tříd emisí znečišťujících látek nebo hluku. Je také možné definovat vzhled vozidla v grafickém uživatelském rozhraní simulace [7]. Základní verze simulátoru má podobu čisté aplikace, která běží v příkazovém řádku. Tato verze může být doplněna grafickou nadstavbou, která přidává grafické uživatelské rozhraní (gui). Pro tuto práci je využita nadstavba simulátoru s grafickým uživatelským rozhraním.

Při simulaci sledujeme objekty simulace (například vozidla, chodce, semaforey) ve vizuálním zobrazení. K tomu slouží aplikace *SUMO-GUI*. Tato aplikace umožňuje sledování simulace při různém nastavení. Je možné nastavit rychlost simulace, zvýraznit různé aspekty či sledovat aktuální data o vozidle, jako například rychlost, nadmořskou výšku a podobně. Simulace poskytuje širokou škálu výstupních souborů. Mezi ně patří trajektorie vozidla, dopravní data shromážděná z modelových detektorů nebo emise, spotřeba energie atd. Tyto výstupní soubory lze vizualizovat pomocí nástrojů SUMO přímo za chodu simulace nebo je lze po skončení simulace exportovat do souborů. Tyto soubory můžeme následně importovat a zpracovat například pomocí softwaru Matlab.

2.4 Doplněk TraCI

TraCI je zkratka pro *Traffic Control Interface*. Tento doplněk umožňuje přístup k simulaci provozu a umožňuje načíst hodnoty simulovaných objektů a měnit jejich hodnoty a chování „on-line“ [15]. Tento doplněk používá architekturu klient/server založenou na protokolu TCP, která poskytuje přístup k SUMO. Simulátor tedy funguje jako server, který je spuštěn pomocí TraCI. SUMO připraví pouze simulaci, ale vše ostatní je ovládáno pomocí externí aplikace. Při použití grafického rozhraní simulace (*SUMO-GUI*) je simulace spuštěna buď stisknutím tlačítka START přímo v simulátoru nebo nastavením příkazu `--start` před samotným zpracováním příkazů TraCI. Příkazy lze rozdělit do několika skupin. Příkazy související s ovládáním simulátoru slouží k řízení připojení k serveru či nastavení simulace. Patří mezi ně příkazy jako `start`, `step`, `close`, `connect`. Další skupinou jsou příkazy pro načítání hodnot. Mezi tyto příkazy patří například `get`. Další příkazy spadají pod skupinu změny stavu. Zde se objevují příkazy `set`. Většina těchto příkazů bude dále podrobněji vysvětlena v praktické části (str. 25) této práce.

V této práci je použita implementace rozhraní TraCI pro software Matlab, tedy doplněk pro Matlab s názvem *TraCI4Matlab*. Tento doplněk umožňuje komunikovat se SUMO ve scénáři klient/server, kde Matlab působí jako klient a SUMO jako server. Je zřejmé, že použití TraCI zpomaluje rychlost simulace oproti klasické simulaci pomocí SUMO, toto zpomalení závisí na mnoha faktorech. Jednoduše řečeno, čím více bude použito volání

funkcí TraCI v každém kroku simulace a následně počet výpočtů v jednom cyklu, tím pomaleji simulace poběží.

PRAKTICKÁ ČÁST

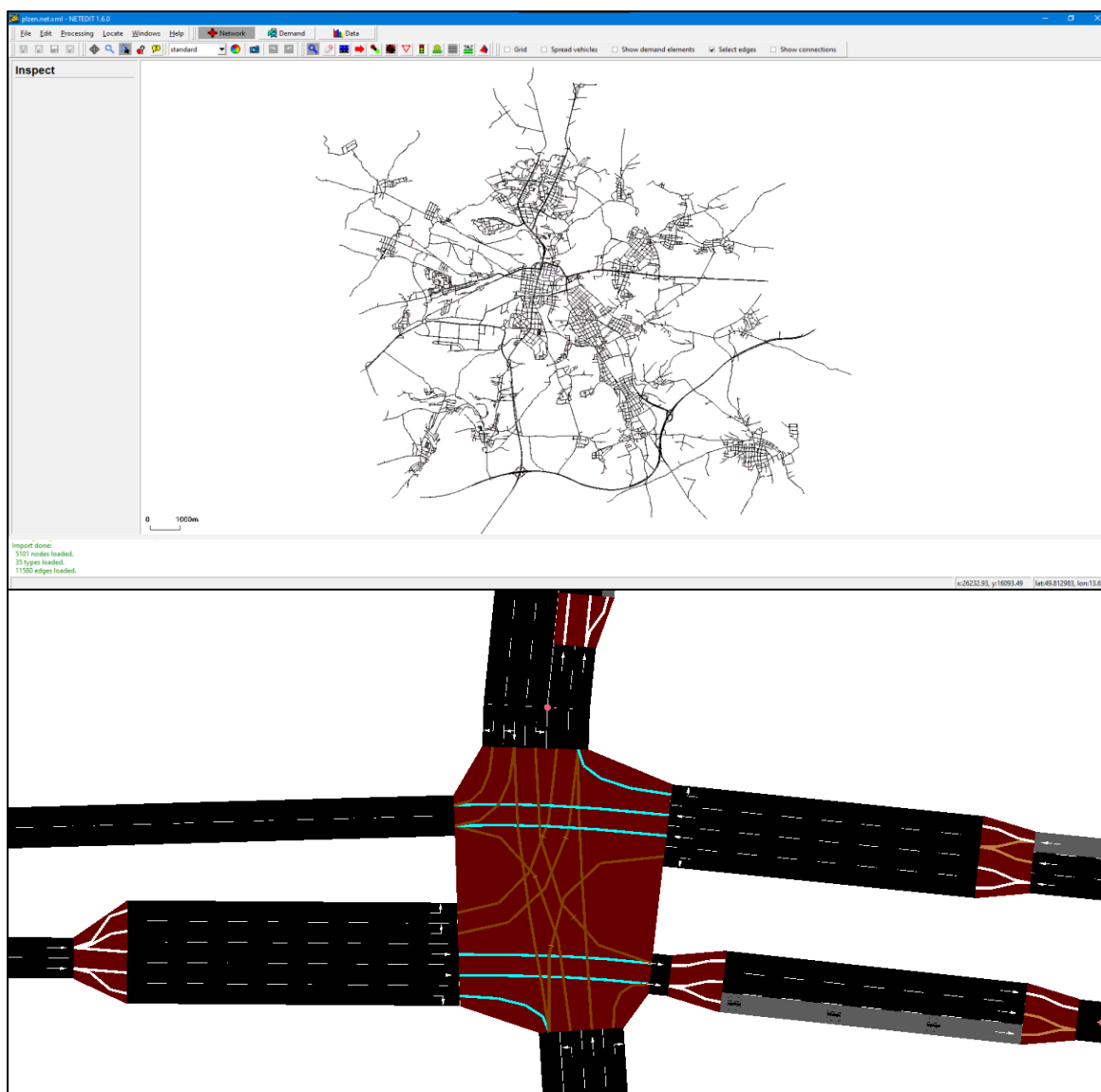
3 Vstupní soubory simulátoru

Jak bylo v předešlých odstavcích zmíněno, k simulaci je potřeba několik vstupních souborů. Ke konkrétní simulaci trolejbusů linky 12 a 13, kterou se zabývá tato práce, je potřeba více vstupních souborů než těch základních. Na všechny tyto soubory se nyní podíváme detailněji a řekneme si, k čemu slouží. Všechny tyto soubory obsahují strukturově uložená data ve formátu *XML* a lze je proto prohlížet jako textové dokumenty. Přípony názvů vstupních souborů jsou zpravidla ve tvaru *typ.xml*, kde *typ* charakterizuje, o jaký soubor jde. Například, zda jde o síťový soubor (*net*) nebo o soubor pro definici vozidel, typů vozidel a tras vozidel (*rou*). Všechny tyto typy souborů naleznete v [17].

3.1 Síťové vstupní soubory

Síťové soubory patří mezi ty důležité. Tyto soubory popisují síť (mapu), na které chceme daný provoz simulovat. Obsahují informace o silnicích, křižovatkách, světelné signalizaci a podobně. Konkrétně síť použitá v této práci obsahuje informace o každé ulici (hraně) jako souboru jízdnicích pruhů včetně polohy, tvaru a rychlostního limitu každého jízdnicího pruhu. Dále tato síť obsahuje informace o logice semaforů, křižovatkách včetně jejich regulace přednosti v jízdě a v neposlední řadě také informace o spojení mezi jízdnicími pruhy v uzlech (křižovatkách).

Pro tuto práci jsou tedy použity soubory *plzen_old.net.xml* a *plzen.net.xml*. Jejich rozdíly budou popsány dále spolu s výsledky simulace. Informace o dané síti jsou v souboru uloženy v konkrétním pořadí. Ačkoliv je možné *XML* soubory číst i upravovat ručně, kvůli značnému rozsahu těchto souborů je pro jejich úpravu vhodné využít nástroj *NETEDIT*, který je součástí distribuce *SUMO*. Náhled grafického rozhraní tohoto nástroje s načtenou sítí Plzně je na *Obr. 3.1* nahoře. Ve spodní části tohoto obrázku je náhled na křižovatku, kde můžeme vidět dané jízdnicí pruhy a jejich spojení.



Obr. 3.1: Ukázka sítě Plzně v nástroji NETEDIT a konkrétní křižovatky

3.2 Soubory pro definici vozidel, typů vozidel a tras vozidel

Dalšími typy vstupních souborů jsou soubory pro definici dopravní zátěže. Lze je definovat různými způsoby jako například ručním generováním, vytvořením pomocí nástroje *NETEDIT*, použitím náhodného generování nebo použitím matic. Každé vozidlo obsahuje informace o typu vozidla, který popisuje fyzikální vlastnosti daného vozidla, dále pak o trase, po které se vozidlo bude pohybovat a samozřejmě také informace o samotném vozidle.

Konkrétně zde byly použity tři tyto soubory. Jeden soubor je použit pro definici okolní dopravy, tedy definici takové dopravy, aby v každém simulačním kroku simulace bylo v síti přibližně 3000 osobních aut. Jedná se o soubor s názvem *cars_009_3000.rou.xml*. Dalším souborem je soubor pro definici simulovaných trolejbusů s názvem *trolleybus1213_flows.rou.xml*. V tomto souboru je definována trasa linek 12 a 13 pro tyto trolejbusy, dále je zde nastaven odjezd trolejbusů s danou periodou a čas zastavení na konkrétních zastávkách pro obě linky. Posledním souborem je soubor s názvem *trolleybus_vType_v0.rou.xml*, který určuje konkrétní vlastnosti trolejbusů jako jejich rozměry, hmotnost, ale i konkrétní parametry pro model hybridního vozidla, které jsou vysvětleny v předchozí kapitole (2) týkající se modelů SUMO. Tyto parametry a jejich definované hodnoty můžeme vidět na Obr. 3.2.

```
<vType id="trolleybus_meanSkodaSolaris" accel="0.6" decel="0.8" length="12" maxSpeed="18.0"
sigma="0.0" minGap="2.5" color="0,1,0" emissionClass="Energy/unknown" vClass="bus" guiShape="bus/trolley">
  <param key="maximumBatteryCapacity" value="46000"/>
  <param key="maximumPower" value="160000"/>
  <param key="overheadWireChargingPower" value="40000"/>
  <param key="vehicleMass" value="13000"/>
  <param key="frontSurfaceArea" value="7.5"/>
  <param key="airDragCoefficient" value="1.1528"/>
  <param key="internalMomentOfInertia" value="397"/> <!-- 0.5*13000*0.061 = 397; default 10-->
  <param key="radialDragCoefficient" value="0.5"/>
  <param key="rollDragCoefficient" value="0.0083"/> <!-- something in between Solaris and Skoda-->
  <param key="constantPowerIntake" value="6000"/>
  <param key="propulsionEfficiency" value="0.85"/>
  <param key="recuperationEfficiency" value="0.85"/>
  <param key="actualBatteryCapacity" value="46000"/>
  <param key="has.elechybrid.device" value="true"/>
</vType>
```

Obr. 3.2: Konkrétní parametry hybridních trolejbusů

3.3 Další vstupní soubory

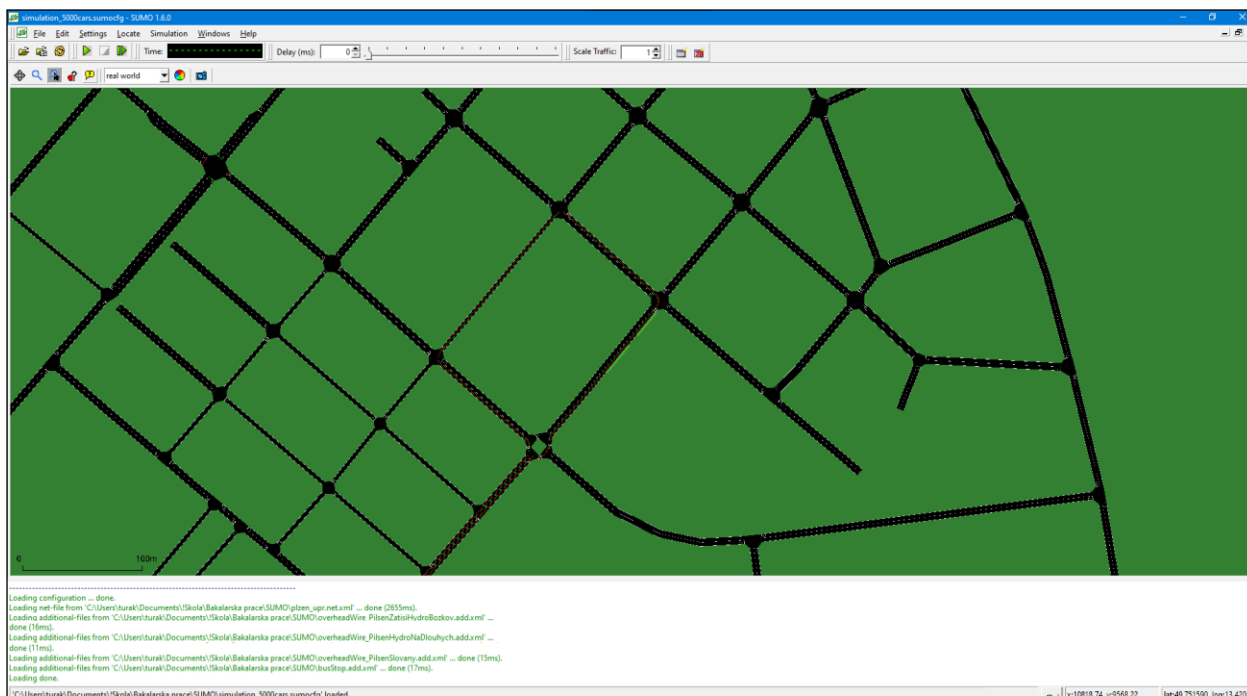
Dalšími vstupními soubory jsou soubory pro definování zastávek a celého zatrolejování sítě. Soubor s definováním zastávek pro obě linky s názvem *busStop.add.xml* určuje pozici, název a délku zastávek. Pro definování trakčních měníren jsou použity tři soubory, *overheadWire_PilsenHydroNaDlouhych.add.xml*, *overheadWire_PilsenSlovany.add.xml* a *overheadWire_PilsenZatysiHydroBozkov.add.xml*. Tyto soubory obsahují celkově 24 trakčních měníren s jejich vlastnostmi. Posledním vstupním souborem je soubor *view.xml*, který pouze nastavuje počáteční pohled při spuštění simulace.

3.4 Konfigurační soubor

Konfigurační soubor je soubor, který nastavuje konkrétní simulaci. Definuje vstupy, tedy všechny zmíněné vstupní soubory, nastavuje simulaci a definuje výstupy. Tímto souborem je *simulation_3000cars.sumocfg*, který po spuštění otevře simulaci v nástroji *SUMO-GUI*.

4 Popis simulace a zpracování výsledků

Jak bylo zmíněno, po načtení souboru *simulation_3000cars.sumocfg* nástrojem *SUMO-GUI* se načtou všechny vstupy do simulátoru. Ukázka tohoto simulačního nástroje je znázorněna na *Obr. 4.1*. Při otevření se ihned načtou všechny vstupy, které jsou uvedeny v konfiguračním souboru simulace. To lze vidět ve spodní části tohoto obrázku. Zároveň vidíme, že v síti nejsou žádná vozidla, ty se objeví až po spuštění simulace.



Obr. 4.1: Ukázka prostředí SUMO-GUI po spuštění

Po načtení všech souborů můžeme tedy spustit simulaci. Simulace se spouští pomocí tlačítka *play* (zelená šipka) v horní části grafického rozhraní. Simulaci také můžeme zastavit pomocí tlačítka *stop* (červený čtverec). Dále můžeme simulaci spustit pouze po jeden simulační krok tlačítkem *single step* (přerušená zelená šipka).

Pokud spustíme simulaci, zobrazí se na dané síti definovaná vozidla a dají se do pohybu. Zároveň se v poli *Time* začne ukazovat aktuální čas simulace. Defaultně je jeden simulační krok nastaven na jednu sekundu, při tomto nastavení tedy aktuální čas simulace zároveň odpovídá i simulačnímu kroku. Vedle zobrazení času (kroku) simulace je pole *Delay (ms)* s posuvným prvkem. Tímto si můžeme nastavit zpoždění v milisekundách mezi kroky

simulace. Pokud si simulaci zastavíme, můžeme se pomocí pravého tlačítka myši a kliknutím na daný objekt dostat k aktuálním informacím.

Konkrétně použitá simulace v této práci je nastavena s defaultním simulačním krokem a s dobou simulace 10800 sekund. Tato doba odpovídá simulaci reálného provozu po dobu třech hodin. Trolejbusy linek 12 i 13 jsou nastaveny v souboru *trolleybus1213_flows.rou.xml* tak, že každých 1080 sekund vyjede nový trolejbus, kde první trolejbusy obou linek vyrazí hned v prvním kroku simulace. Interval pro trolejbusy odpovídá 18 minutám. Tedy každých 18 minut vyjíždí trolejbus z počátku své trasy. Zároveň je zde nastaveno to, že trolejbusy už nevyjíždí po kroku simulace 7100, tedy po necelých dvou hodinách. Tím je ošetřen počet výstupních souborů, kterých by bylo zbytečně mnoho. S tímto nastavením vyjede pro každou linku 7 hybridních trolejbusů.

4.1 Výstupní soubory a jejich zpracování do souboru

Výstupní soubory jsou definovány v konfiguračním souboru s názvem *simulation_3000cars.sumocfg*. Tyto soubory jsou rozděleny do sumárního výstupního souboru, do výstupního souboru pro trakční měničy a do výstupních souborů pro trolejbusy. Všechny tyto soubory mají příponu *.out.xml*. Výstupní soubor *out_summary_3000.out.xml* udává celkové informace o simulaci v každém kroku simulace, například aktuální počet vozidel v síti, počet kolizí, počet aut, které síť už opustili nebo počet aut, které se teleportovali z důvodu dlouhého čekání. Výstupní soubor pro trakční měničy *out_tractionSubstation_3000.out.xml* obsahuje informace o každé měničce jako například které vozidlo je v daném simulačním kroku k této měničce připojeno, jaký odeberá výkon a následně kolik celkové energie dodalo danému vozidlu. Jako poslední jsou výstupy z hybridních trolejbusů. Pokud je v konfiguračním souboru nastavený agregovaný výstup, vytvoří se pouze jeden soubor s informacemi o všech hybridních trolejbusech v simulaci. V opačném případě se vytvoří tolik výstupních souborů, kolik je hybridních trolejbusů v simulaci. Tyto výstupní soubory mají ještě dodatek ke svému názvu podle toho, o jakém konkrétním hybridním trolejbusu jsou v tomto souboru uloženy informace. To je náš případ. Například výstupní soubor *out_trolleybus.elecHybrid_3000.out.xml_trolleybus12_full.0* obsahuje informace o trolejbusu linky 12, který startuje hned při začátku simulace. Výstupní

soubory pro každý trolejbus nesou v sobě informace v každém simulačním kroku například o rychlosti, akceleraci či na jaké silnici se nacházejí, ale i o aktuální kapacitě baterie, aktuální odebírané energie z troleje a podobně.

Abychom mohli výstupní data ze simulace dále zpracovávat, musíme všechny výstupní soubory zpracovat do jednoho souboru, ve kterém budou všechny výstupní data rozříděny. Pro tenhle účel používám skript v Matlabu s názvem *zpracovani_vystupu.m*, který rozdělí všechna výstupní data do matic podle jednotlivých proměnných a poté je uloží do souboru s názvem *data_3000_klasik_old_newVer.mat*. Získaná data lze poté jednodušeji zpracovat opět pomocí Matlabu. Všechna získaná data z jednotlivých trolejbusů jsou uvedena a vysvětlena v *Tab. 4.1* a všechna získaná data z trakčních měření v *Tab. 4.2*. Výstupy z hybridních trolejbusů mají název *results*. Výstupy z trakčních měření mají název *resultsS*. Výstupní veličiny pak mají tvar například pro rychlost trolejbusu *results{x}.speed*, kde *x* znamená daný trolejbus. Třeba pro trolejbus linky 12, který startuje v počátku simulace, je *x* rovno jedné a k jeho rychlosti se dostaneme *results{1}.speed*. Obdobně používáme výstupy pro trakční měření, kde číslo *x* udává, o jakou trakční měřírnu se jedná.

Tab. 4.1: Seznam výstupů z hybridních trolejbusů a jejich význam

Výstupy z hybridních trolejbusů – results	
Název výstupu	Význam výstupu
.id	Název trolejbusu
.distance	Ujetá vzdálenost
.speed	Aktuální rychlost
.energyConsumed	Spotřebovaná energie z troleje bez energie k nabíjení baterie
.index	Indexovací vektor
.overheadWireBool	Stav připojení troleje
.myConsum	Spotřeba energie trolejbusu (trakční + pomocné pohony)
.energyCharged	Dodaná energie do baterie
.circuitVoltage	Napětí na troleji
.current	Proud z troleje
.altitude	Nadmořská výška
.soc	Stav nabití baterie
.alpha	Udává, zda náhradní elektrický obvod trolejové sekce je řešitelný
.powerWanted	Potřebný výkon z troleje
.tractionSubstationId	Název trakční měřírny, na kterou je vozidlo připojeno
.lane	Aktuální název jízdního pruhu, na kterém se vozidlo nachází

Tab. 4.2: Seznam výstupů z trakčních měření a jejich význam

Výstupy z trakčních měření – resultsS	
Název výstupu	Význam výstupu
.id	Název trakční měřírny
.length	Délka trolejového vedení
.totalEnergyCharged	Celková dodaná energie z trakční měřírny
.chargingSteps	Počet dobíjecích kroků
.num_voltage_sources	Celkový počet napěťových zdrojů
.num_claps	Počet propojek na trolejové sekci
.time	Čas
.vehiclesIds	Název vozidel připojené k dané měřírně
.numVehicles	Počet vozidel připojené k dané měřírně
.numVoltageSources	Počet napěťových zdrojů
.ChargingStatus	Prázdná matice – není definováno
.energyCharged	Dodaná energie z trakční měřírny
.current	Proud
.voltage	Napětí zdrojů
.alphaCircuitSolver	Udává, zda náhradní elektrický obvod trolejové sekce je řešitelný
.alphaFlag	Udává důvod, proč je alpha menší než 1

4.2 Zpracování výstupních dat

Výstupní data máme v tomto momentě zpracovaná v souboru. Pomocí skriptu `zpracovani_dat.m` tyto data importujeme, následně je zpracujeme, a nakonec vykreslíme pomocí grafů. Import souboru a následně výpis jednotlivých proměnných v tomto souboru je znázorněno na Obr. 4.2. V souboru jsou obsaženy informace o čtrnácti trolejbusích (7 linky 12 a 7 linky 13) a dvaceti čtyř trakčních měřírkách. Pro vykreslení například závislosti rychlosti na vzdálenosti prvního trolejbusu linky 12 použijeme následující příkaz `plot(results{1}.distance,results{1}.speed)`. Pro zobrazení rychlosti na vzdálenosti trolejbusu linky 13 pak `plot(results{8}.distance,results{8}.speed)`. Obdobně budeme postupovat při vykreslování závislostí z trakčních měření, ačkoliv se k jejich vykreslování hodí více sloupcový diagram.

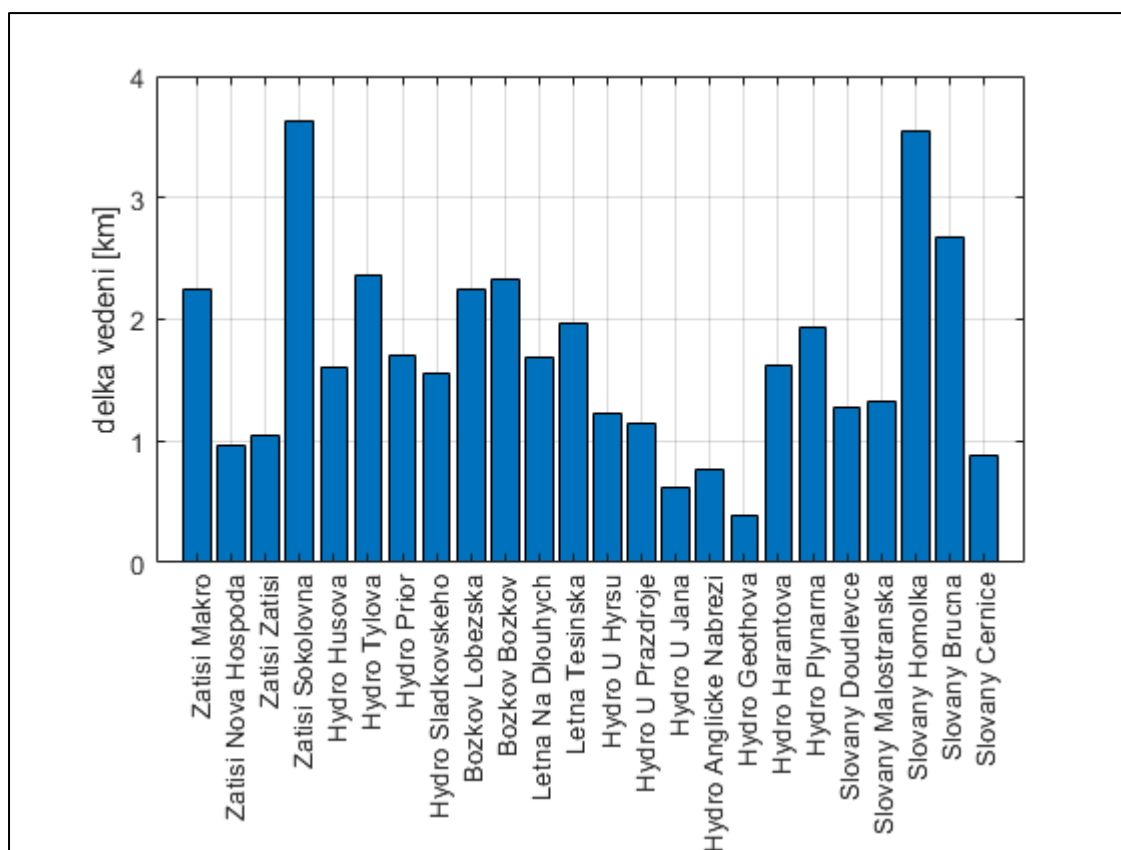
```
1      %% ZPRACOVANI DAT Z TROLEJBUSU (results) A MENIREN (resultsS)
2      clear all
3      clc
4      % nacteni souboru s daty
5      load('data_5000_all_OK')
6
7      % z kolika trolejbusu a stanic mame udaje
8      pocet_trolejbusu = length(results)
9      pocet_stanic = length(resultsS)
10
11     % vypis dat ulozenych v results a resultsS
12     for t = 1:pocet_trolejbusu
13         results{t}
14         t=t+1;
15     end
16
17     for s = 1:pocet_stanic
18         resultsS{s}
19         s=s+1;
20     end
```

Obr. 4.2: Zpracování dat ze souboru

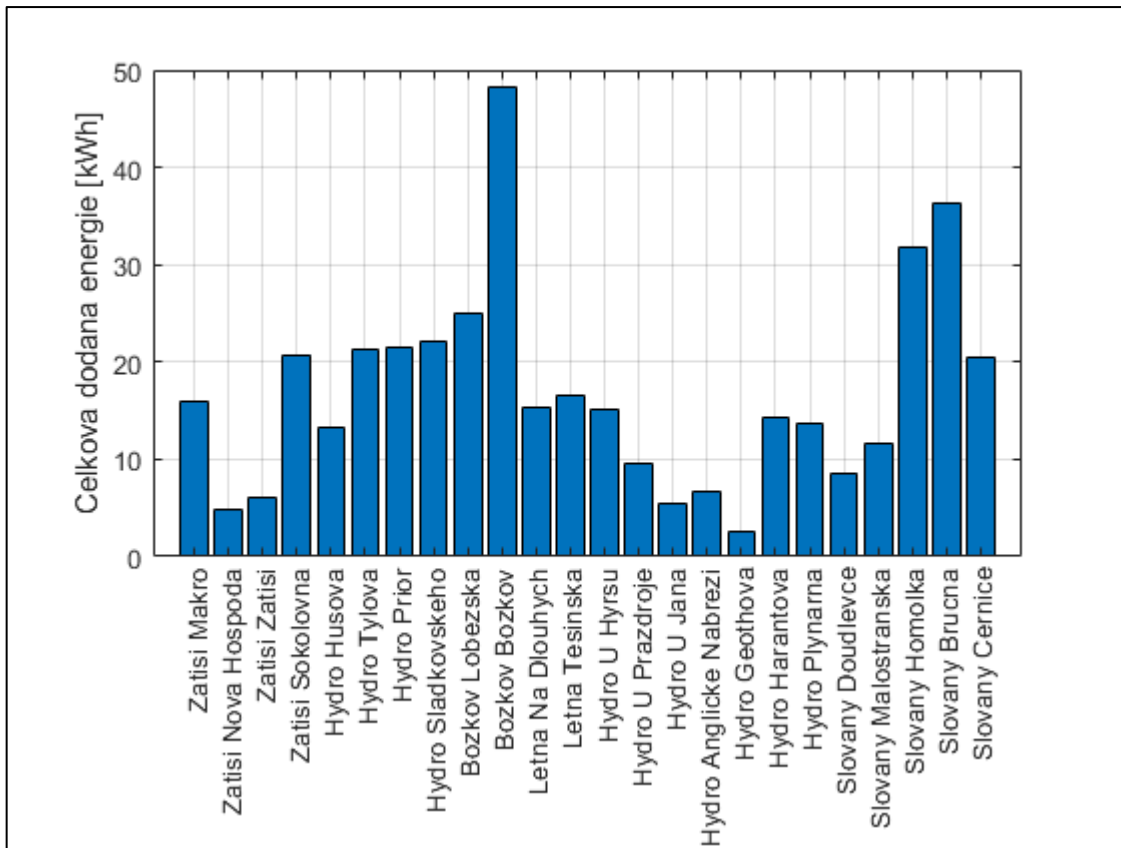
4.2.1 Zpracování výstupních dat z trakčních měření

Podíváme se tedy na konkrétní grafické výstupy ze simulace. Začneme s výstupy z trakčních měření. Simulované trakční měřírny udržují konstantní napětí 600 V, při odběru energie se mění v čase odebíraný (či dodaný) elektrický proud. Trolejové vedení od každé trakční měřírny má rozdílnou délku. To lze vidět na *Obr. 4.3*. S větší délkou trolejového vedení může být na toto vedení připojeno více trolejbusů zároveň, ačkoliv delší vedení jsou obvykle v krajních částech města. Celková dodaná energie z trakčních měření bude záviset na délce trolejového vedení připojené k dané měřírně a také na počtu průjezdů hybridních trolejbusů. Z logického hlediska by měla být dodaná energie z trakčních měření větší v centru města, kde tyto trolejbusy projíždějí častěji než na okraji města Plzně. Pokud se podíváme na *Obr. 4.4*, na kterém je zobrazena celková dodaná energie z jednotlivých trakčních měření, zjistíme, že v centru města Plzně dodaná energie z trakčních měření nedosahuje největších hodnot. To je způsobeno tím, že trolejbusy jedou část své trasy mimo trolejové vedení, tedy odebírají energii z baterie a poté po připojení k troleji odebírají v krajních částech města více energie hlavně pro opětovné nabití baterie. Počet dobíjecích kroků jednotlivých trakčních měření je znázorněn na *Obr. 4.5*.

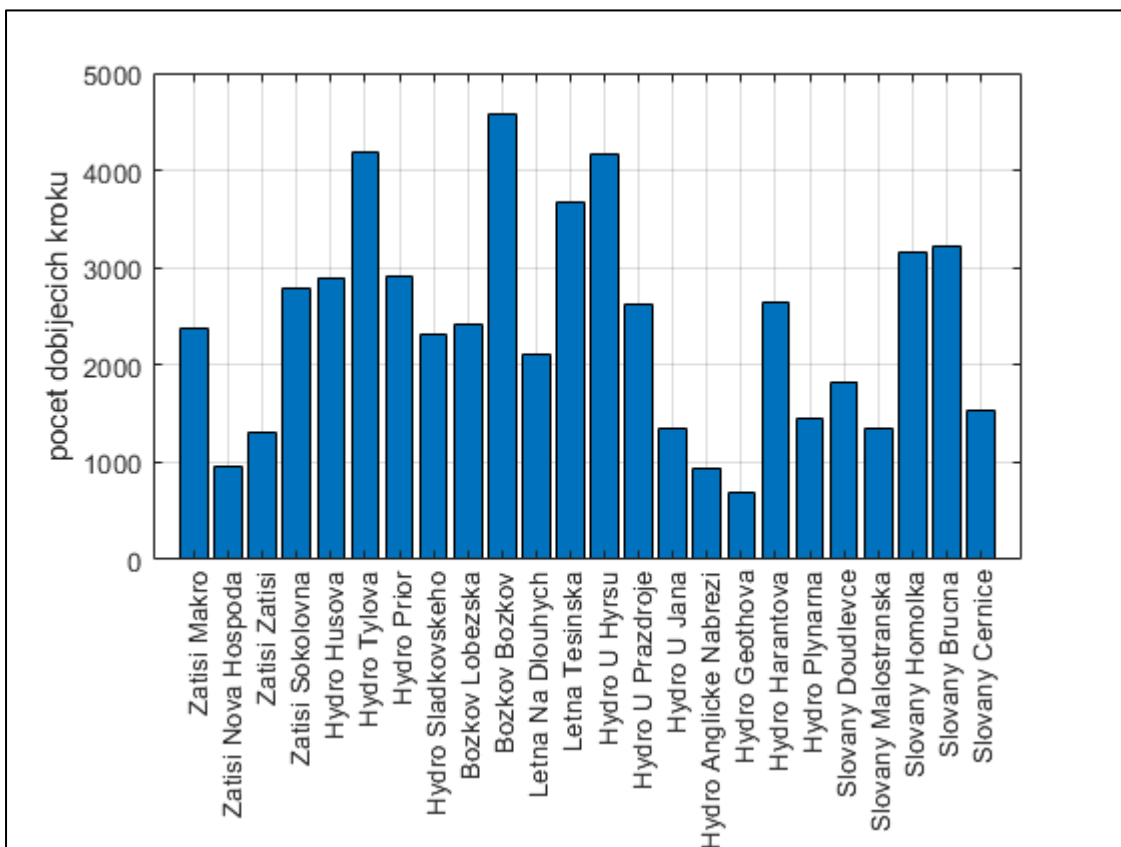
Při porovnání závislostí z obrázků 4.3, 4.4 a 4.5 můžeme vyčíst, že trakční měnírna Božkov, sekce Božkov dodala nejvíce energie ze všech trakčních měření, zároveň nejčastěji dobíjela, ačkoliv nemá nejdelší délku trolejového vedení a neleží ani v centru města. Po detailnějším zkoumání zjistíme, že tato trakční měnírna napájí trolejové vedení, ke kterému jsou připojeny jen trolejbusy linky 12. Tyto trolejbusy dále pokračují směrem na Letkov, kde není zřízeno trolejové vedení. Tedy hybridní trolejbusy odebírají energii z baterie. Po návratu zpět a následném připojení k trolejovému vedení je tedy nutno dobít baterii zpět na plnou kapacitu. Dostáváme se tedy k další závislosti, kdy celková dodaná energie z trakčních měření závisí i na stavu baterie hybridního trolejbusu. Pokud je nabitá na plnou kapacitu, je dodávaná energie pouze pro pohyb vozidla a pomocné pohony, v opačném případě se k tomu ještě přidá energie pro nabíjení baterie.



Obr. 4.3: Znárodnění délky trolejového vedení jednotlivých trakčních měření



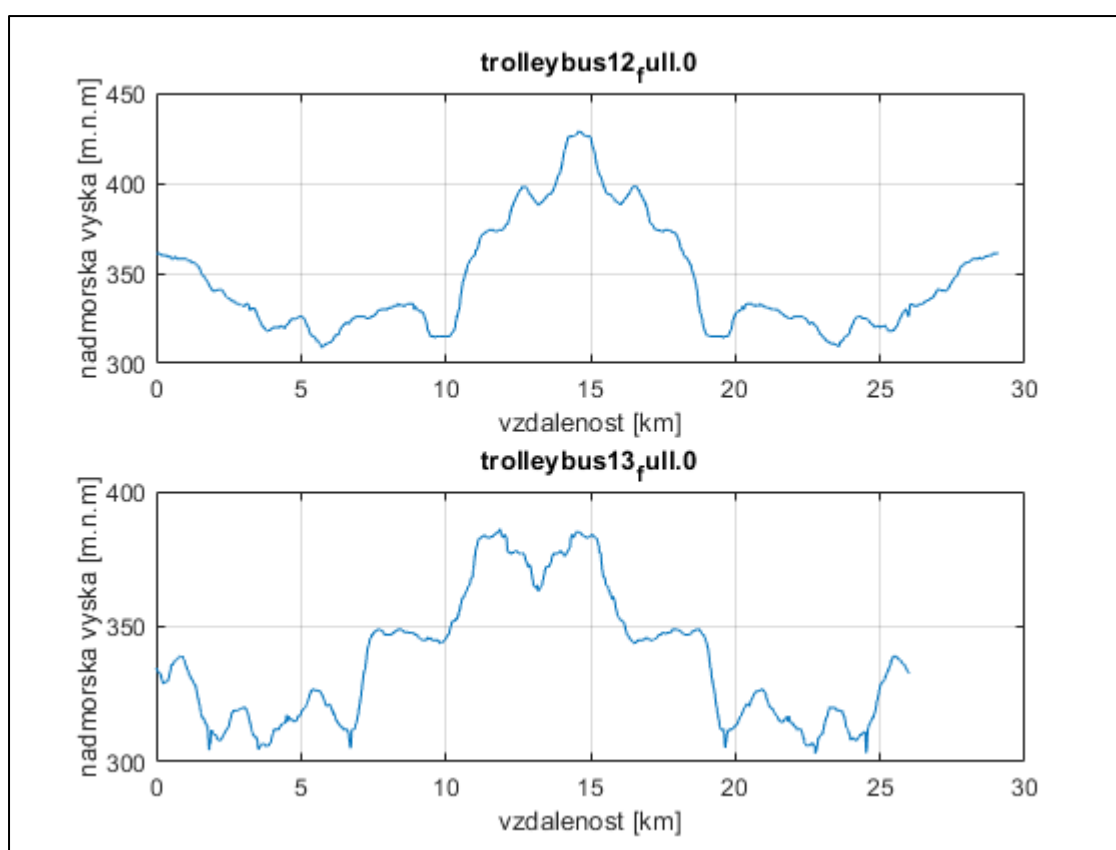
Obr. 4.4: Znárodnění celkové dodané energie jednotlivých trakčních měření



Obr. 4.5: Znárodnění počtu dobijecich kroků jednotlivých trakčních měření

4.2.2 Zpracování výstupních dat z hybridních trolejbusů

Nyní se podíváme na výstupy z hybridních trolejbusů. Začneme sítí, po které se trolejbusy pohybují. Nadmořská výška tras linek 12 a 13 je vykreslena na *Obr. 4.6*. Z tohoto grafu můžeme také odečíst přibližnou délku jednotlivých linek, tedy linka 12 má délku přibližně 29 km a linka 13 přibližně 26 km. Zároveň si můžeme všimnout, že v různých vzdálenostech dochází k neopodstatněným propadům nadmořské výšky, zejména u linky 13. Tyto propady jsou způsobeny importem z reálných mapových podkladů, kde tyto propady nastávají většinou u mostů. Tyto propady jsou tedy skoky nadmořské výšky z mostu do údolí řeky či na kolejové trati pro vlaky. Je tedy nutná oprava této sítě pro vyhlazení těchto propadů. Tato oprava bude znázorněna v další, 5. kapitole práce.



Obr. 4.6: Závislost nadmořské výšky na vzdálenosti trolejbusů linek 12 a 13

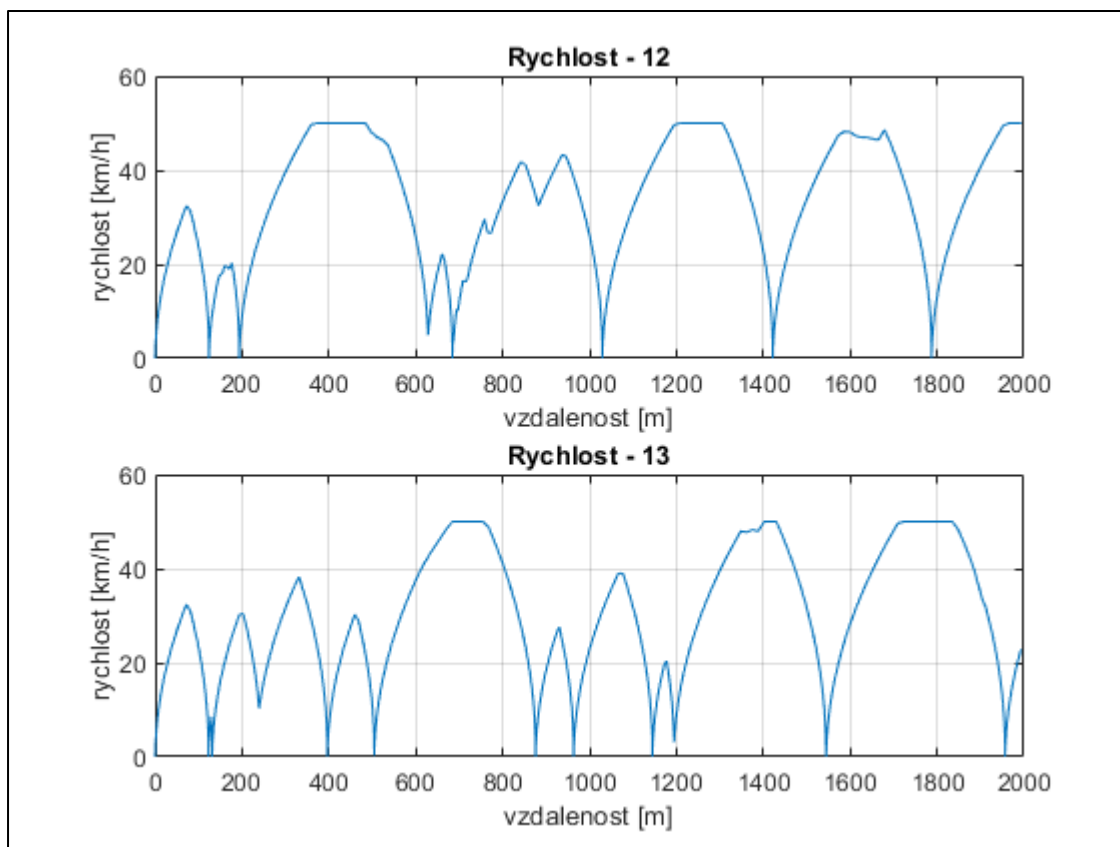
Další důležité výstupy z hybridních trolejbusů jsou veškeré energetické závislosti a také rychlost trolejbusu. Trolejbus je v simulaci nastaven tak, že zrychluje s maximálním nastaveným zrychlením na maximální dovolenou hodnotu rychlosti, která je nastavena na konkrétním jízdním pruhu. Většinou tato hodnota je 13,89 m/s, což se rovná 50 km/hod. Pohyb trolejbusů je omezen křižovatkami, zastávkami a okolní dopravou. Rychlosti

trolejbusů linky 12 a 13 v průběhu dvou kilometrů jsou vidět na *Obr. 4.7*. Tyto průběhy rychlosti se ale neshodují s průběhy rychlosti reálných trolejbusů. V reálném světě neudržují trolejbusy konstantní rychlost (například 50 km/hod), ale lehce zpomalují. To je stav, kdy řidič trolejbusu po dosažení maximální dovolené rychlosti sundá nohu z plynového pedálu a vozidlo poté zpomaluje pomocí jízdních odporů, jako například odporem vzduchu či valivým odporem. Tomuto stavu se říká režim výběhu. Simulace v tomto režimu bude řešena v 5. kapitole této bakalářské práce.

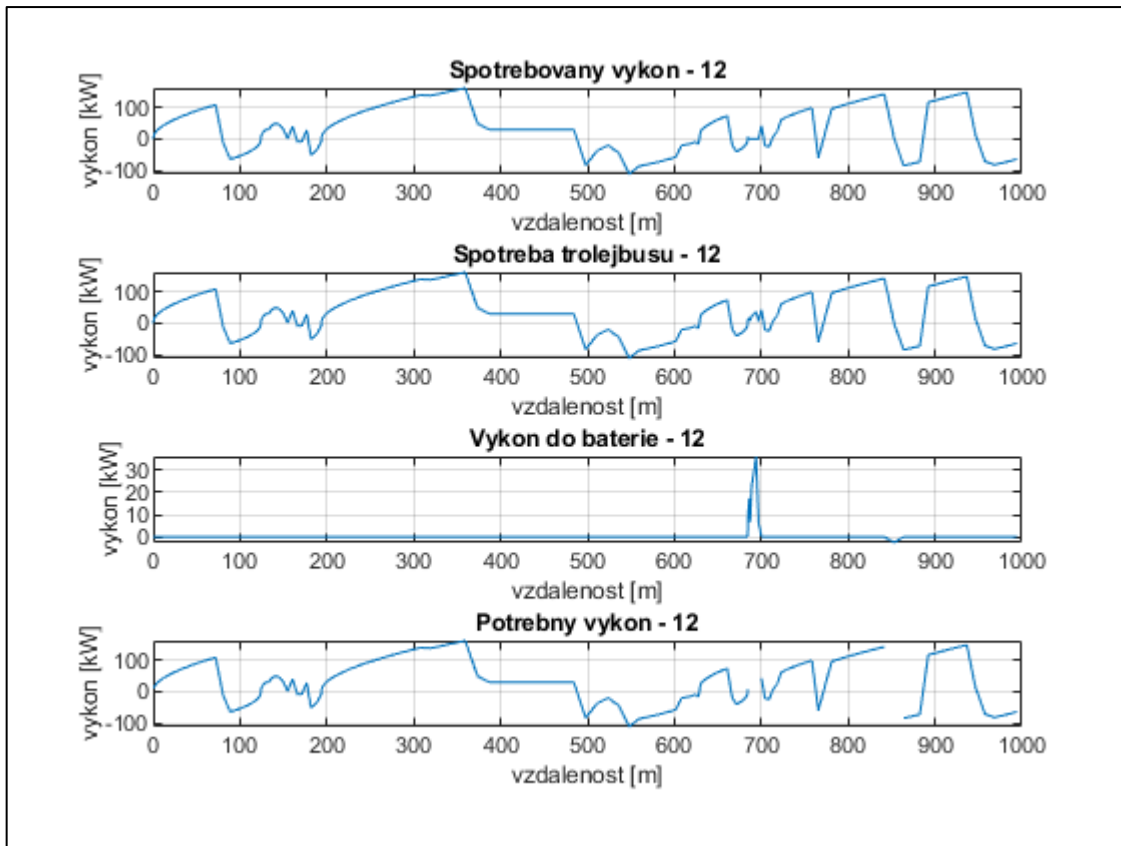
Mezi elektrické výstupy z hybridních trolejbusů patří spotřebovaná energie či výkon z troleje (*energyConsumed*), samotná spotřeba trolejbusu (*myConsum*), výkon dodaný do baterie (*energyCharged*), potřebný výkon z trolejového vedení (*energyWanted*) a stav nabití baterie (*soc*). Z *Obr. 4.8* a z *Obr. 4.9* vidíme, že spotřebovaný výkon z troleje se rovná zároveň spotřebě trolejbusu a zároveň i potřebnému výkonu z troleje. To je ideální případ, který platí většinu času. Najdou se ale místa, kde to tak není. To můžeme vidět na *Obr. 4.8*, kde potřebný výkon je v určité vzdálenosti nespojitý. V tento okamžik nelze energii dostat z trolejového vedení (spotřebovaný výkon z troleje v tomto okamžiku je nulový) a je nutné využít energii z baterie. Co tento stav představuje? Energie z baterie je dodávána trolejbusu, pokud je výstupní parametr *overheadWireBool* roven nule. Tento výstupní parametr indikuje, zda je trolejbus připojen k troleji (rovná se 1) či nikoliv (rovná se 0). Tedy energie z baterie je dodávána trolejbusu v případech, kdy trolejbus jede po trase bez trolejového vedení nebo v případech, kdy po sobě následují dvě rozdílné sekce trolejového vedení, jelikož mezi nimi je vedení izolováno.

Trolejbusy obou linek jedou část své trasy po trase bez trolejového vedení a odebírají energii z baterie. Trolejbusy linky 12 jedou přibližně dvě třetiny trasy připojené k trolejovému vedení a přibližně jednu třetinu trasy odebírají energii z baterie. Trolejbusy linky 13 jedou podstatně větší část své trasy připojené k trolejovému vedení a zbytek trasy (přibližně 4 kilometry) odebírají energii z baterie. To můžeme vidět na *Obr. 4.10* a *Obr. 4.11*. To se projeví na energetickém stavu baterie. Baterie je v simulaci nastavena tak, že na počátku jízdy trolejbusu je baterie nabitá na plnou kapacitu, tedy 46 kWh. Je zřejmé, že kapacita baterie trolejbusu linky 12 klesne více než u linky 13 z důvodu ujeté delší trasy bez trolejového vedení. Z *Obr. 4.12* můžeme vidět, že se kapacita baterie v některých okamžicích dokonce navyšuje. To je způsobeno rekuperací energie, která je vrácena zpět do baterie. Rekuperace může nastat například v okamžiku, kdy trolejbus jede z kopce a brzdí.

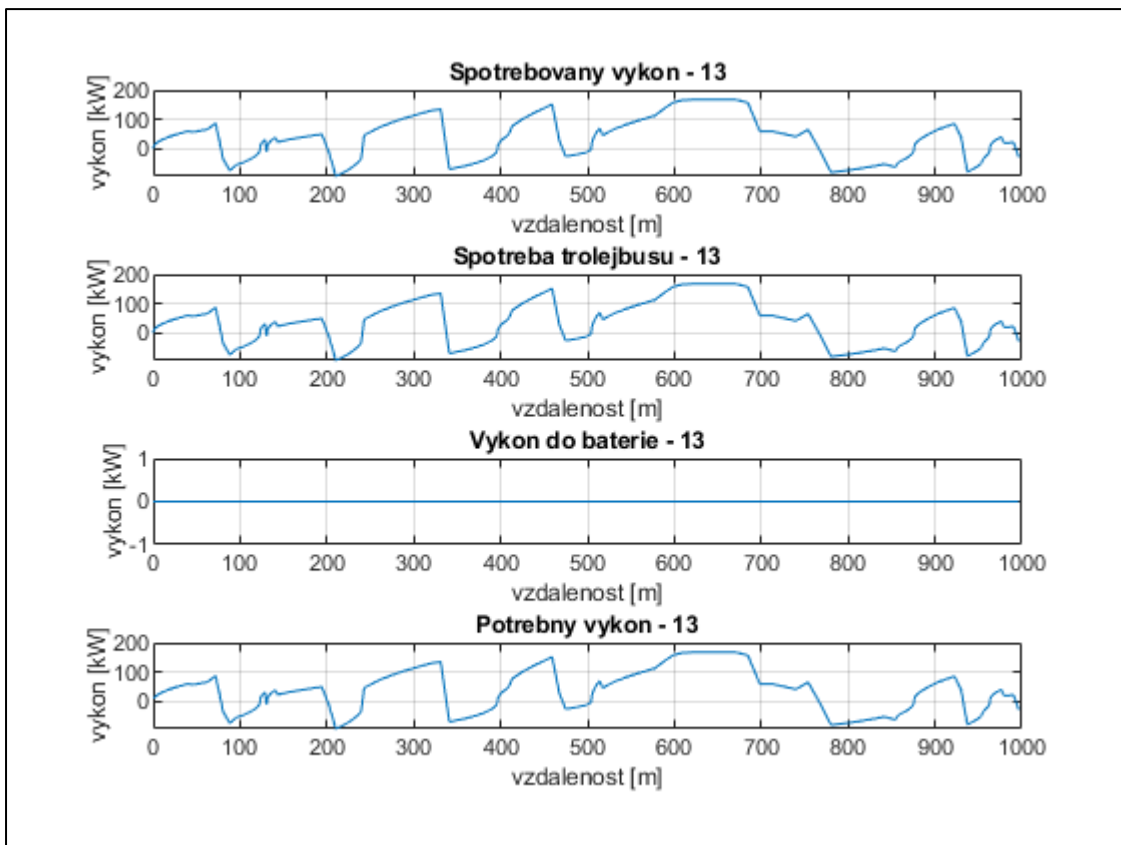
Je tedy otázka, zda se vyplatí pro trolejbusy linky 13 takto velká baterie s kapacitou, kterou moc efektivně nevyužijeme. Současně můžeme z *Obr. 4.12* vyčíst, že při opětovném připojení trolejbusu k trolejovému vedení a následnému dobíjení baterie se baterie nedobije na předešlých 46 kWh. V simulaci je zaimplementována konstanta, která umožňuje dobíjení baterie pouze do 98 % celkové kapacity baterie. Tedy baterie se dobije do hodnoty 45,09 kWh.



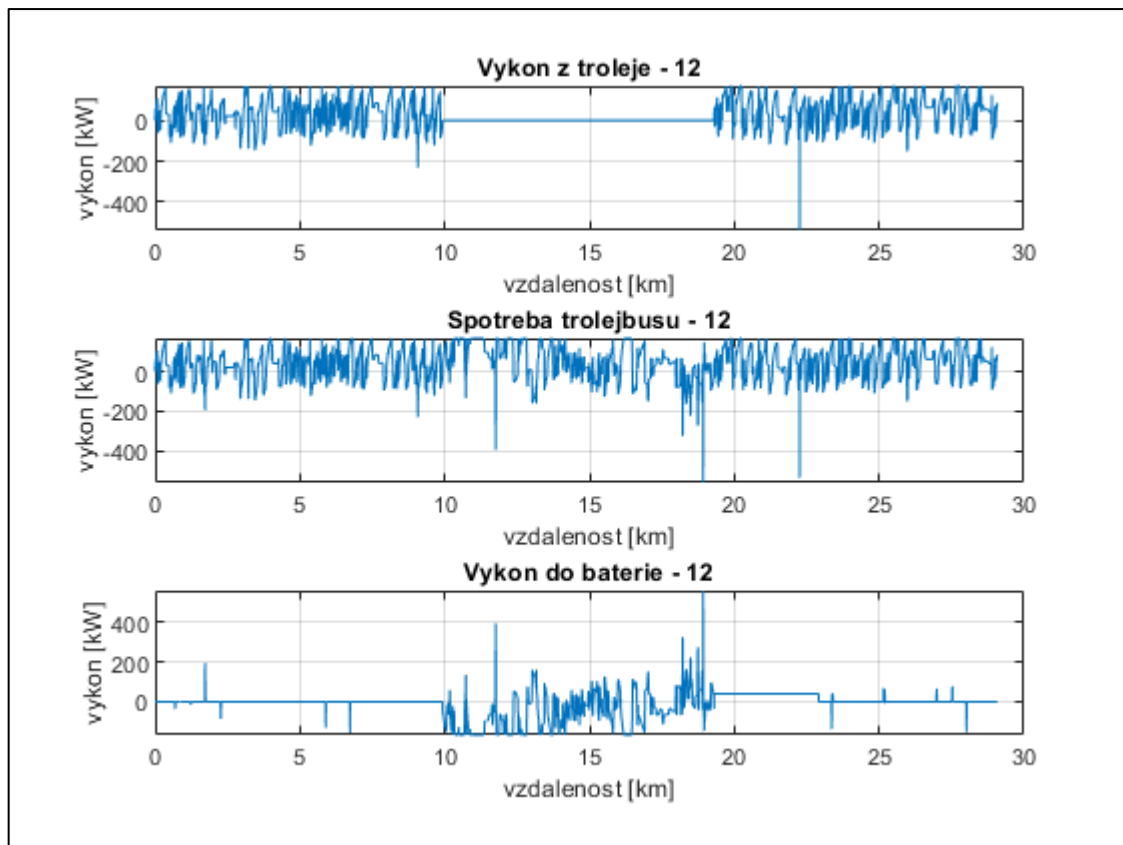
Obr. 4.7: Závislost rychlosti na vzdálenosti pro první 2 km linky 12 a 13



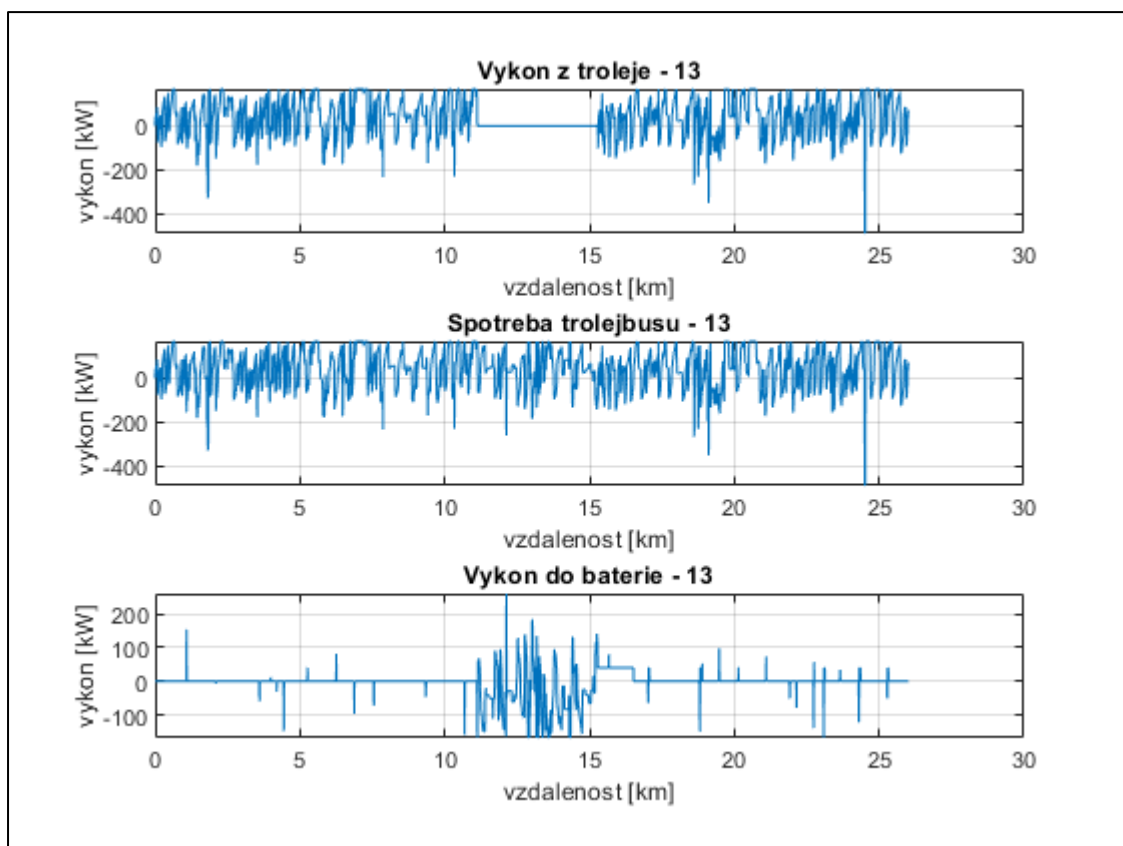
Obr. 4.8: Energetické závislosti trolejbusu linky 12 za 1 km



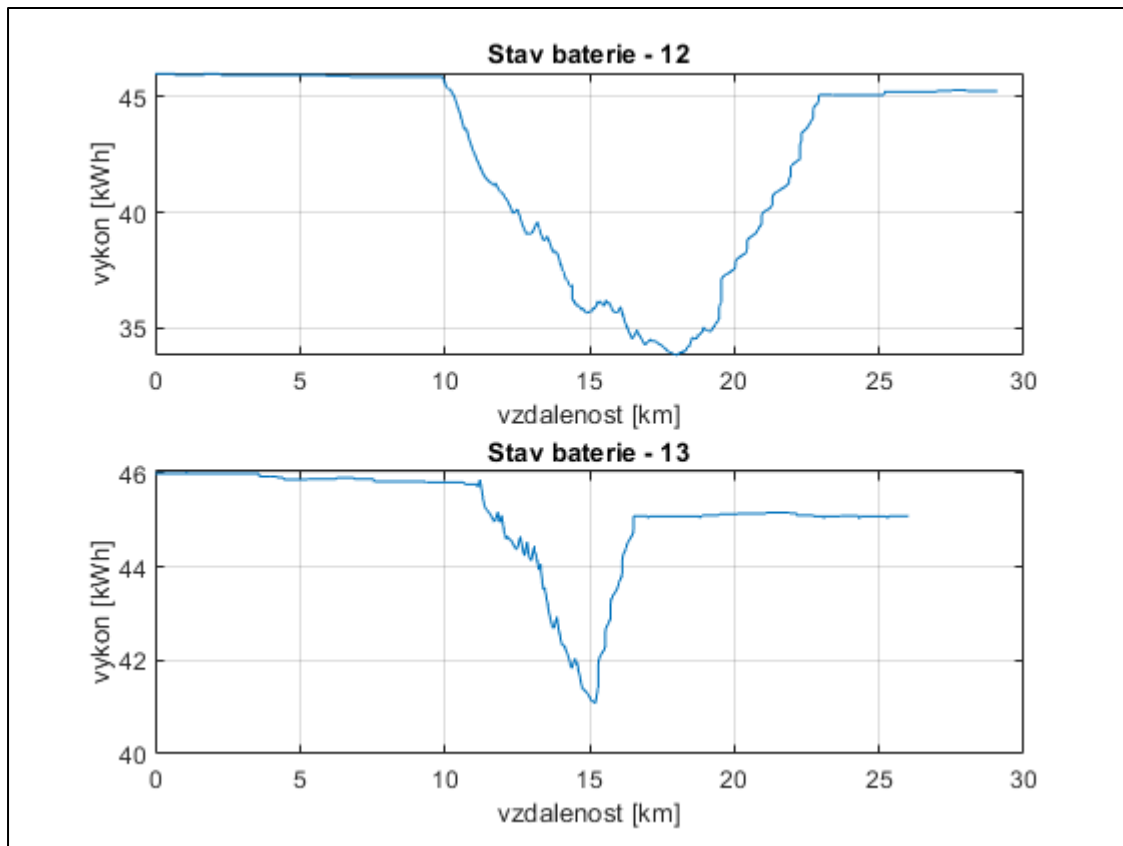
Obr. 4.9: Energetické závislosti trolejbusu linky 13 za 1 km



Obr. 4.10: Energetické závislosti trolejbusu linky 12



Obr. 4.11: Energetické závislosti trolejbusu linky 13



Obr. 4.12: Závislost stavu baterie na vzdálenosti trolejbusů linek 12 a 13

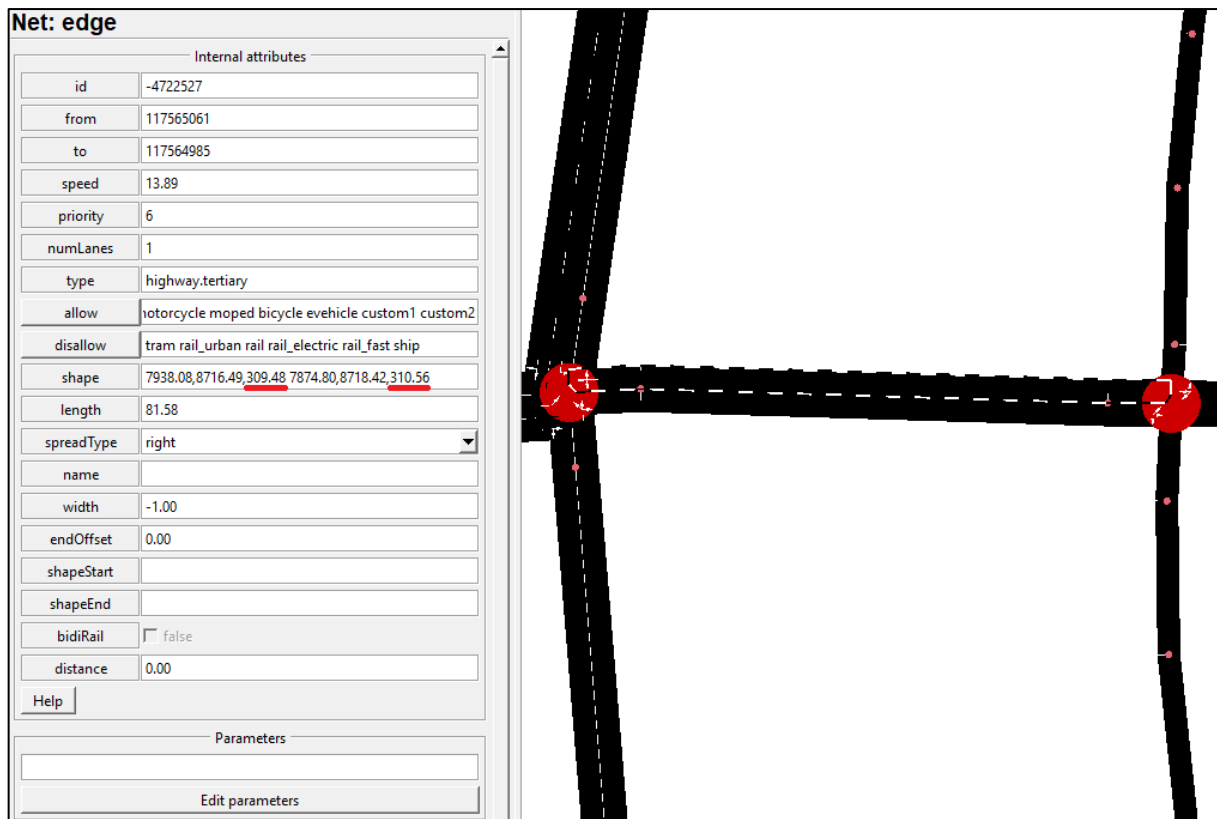
5 Vylepšení simulace

Aby byla simulace přiblížena co nejvíce k reálnému provozu, musí být ještě vylepšena. Nejdříve je třeba upravit výškopisy sítě, aby přibližně odpovídali reálným výškopisům města Plzně. Následně zkusím přiblížit simulaci reálnějšímu chování hybridních trolejbusů v provozu. Jednotlivé výsledky ze simulací (základní či vylepšené) porovnáme s reálnými naměřenými průběhy.

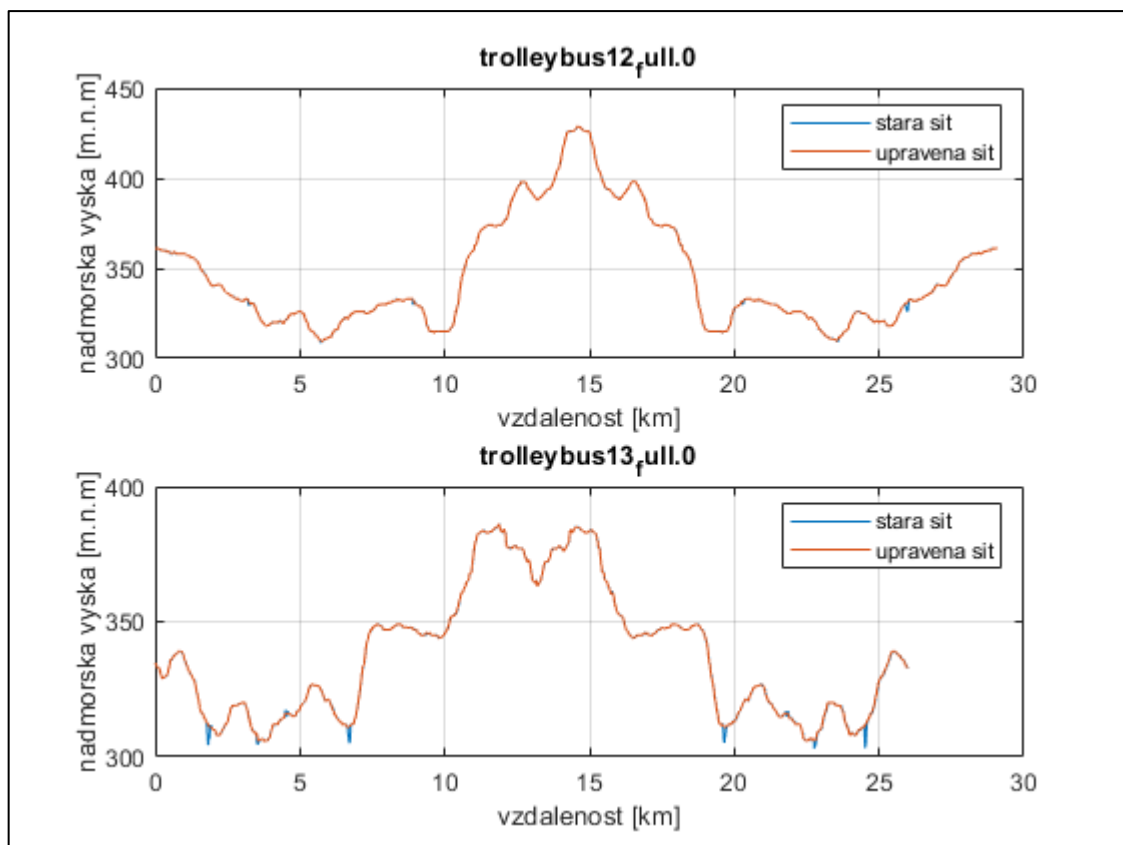
5.1 Oprava sítě

Jak bylo v předešlé části zmíněno, je potřeba upravit síť města Plzně kvůli několika propadům nadmořské výšky (viz. *Obr. 4.6*). Nadmořská výška trasy trolejbusů linky 12 vypadá vcelku v pořádku kromě jednoho menšího propadu okolo třetího a následně dvacátého šestého kilometru. Jde o ten samý propad, jelikož trolejbusy jedou danou trasu linky dvakrát, tedy z počátku linky na konec a zase zpět. Propad nadmořské výšky je způsobený mostem, u kterého nadmořská výška spadne do údolí řeky. To je způsobeno importem sítě z mapových podkladů. Nadmořská výška trasy trolejbusů linky 13 má těchto propadů více. Většinou jde opět o ten samý případ, kde nadmořská výška odpovídá nadmořské výšce údolí řeky, a ne nadmořské výšce daného jízdniho pruhu. K úpravě sítě tedy použijeme nástroj *NETEDIT*. Projdeme trasy jednotlivých linek a najdeme místa s mosty přes řeku či koleje. Následně vybereme daný jízdni pruh a upravíme jeho polohu v souřadnici *z* v kolonce *shape*. Tento úkon je zobrazen na *Obr. 5.1*, kde jsou červeně podtrženy upravované parametry. Dále upravíme i další jízdni pruhy a případně i spojení mezi nimi. Takto budeme postupovat při úpravách ostatních propadů nadmořské výšky.

Po těchto úpravách výškopisů sítě spustíme opět simulaci, ale nyní s novou upravenou sítí. Po dokončení simulace opět zpracujeme data do souboru s názvem *data_3000_vybeh_new_newVer.mat*. Následně data ze souboru zpracujeme a vykreslíme pomocí Matlabu. Výsledek úpravy výškopisů můžeme vidět na *Obr. 5.2*. Stará síť je vykreslena modrou barvou, nová upravená síť je vykreslena oranžovou barvou. Je zřejmé, že jsme se zbavili těch nejhorsích propadů nadmořské výšky.



Obr. 5.1: Náhled na úpravu výškopisů



Obr. 5.2: Výsledek úpravy výškopisů sítě

5.2 Režim výběhu

Režim výběhu je stav, kdy vozidlo (pro nás hybridní trolejbus) zpomaluje z nějaké určité rychlosti díky jízdním odporům. Jízdními odpory trolejbusu jsou odpor vzduchu, valivý odpor kol, odpor sklonu vozovky a odpor při jízdě zatáčkou. Režim výběhu nastane tehdy, když vozidlo dosáhne určité maximální dovolené rychlosti (například 50 km/h) a poté řidič uvolní plynový pedál a vozidlo začne zpomalovat. Přitom řidič nepoužívá brzdy. S tímto režimem se setkáme častěji v praxi oproti režimu, kdy vozidlo udržuje konstantní rychlost. Proto jsem vytvořil skript v Matlabu, který se režim výběhu snaží co nejvíce napodobit a pomocí doplňku TraCI je poté simulován v simulátoru SUMO.

Princip fungování skriptu pro režim výběhu je následující: V každém kroku simulace budeme získávat informace o hybridních trolejbusech v síti. Těmito informacemi budou informace o rychlosti, akceleraci, maximální dovolené rychlosti na daném jízdním pruhu, sklonu jízdního pruhu a informace o energii vozidla. Poté pomocí nějaké podmínky musíme vozidlo dostat do režimu výběhu. Pokud se trolejbus dostane do režimu výběhu, musíme z jeho aktuálních informací vypočítat jeho následné zpomalení. Toto požadované zpomalení získáme pomocí funkce `accel.m`. Zároveň musíme trolejbusy udržet v režimu výběhu po takovou dobu, než klesne jeho aktuální rychlost pod nastavenou mez nebo než nastane jiný požadavek na aktuální rychlost. Těmito požadavky jsou například zvýšení dovolené rychlosti na úseku, požadavek na brždění do zastávky, před křižovatkou anebo kvůli dopravní situaci.

5.2.1 Konkrétní realizace režimu výběhu

Nyní si projdeme skript pro režim výběhu po jednotlivých částech. Jelikož spouštíme simulaci pomocí doplňku TraCI, je vhodné hned na začátku importovat jeho konstanty. Následně musíme nastavit složku, kde se nachází konfigurační soubor, ale i všechny vstupní soubory (viz Vstupní soubory simulátoru). Poté simulaci spustíme a nastavíme proměnné definující začátek a konec simulace. Tato část je znázorněna na *Obr. 5.3*.

Následuje inicializace proměnných. To znamená, že si připravíme místo pro zápis do matice pro každou proměnou během simulace. Poté už se dostaneme na hlavní smyčku programu. Tato smyčka je tvořena cyklem *while* a podmínkou *step < end_step*. Tento cyklus probíhá, dokud je aktuální simulační krok menší než krok nastavený pro konec simulace. Jako další na řadu přichází smyčka, ve které budu zjišťovat informace o trolejbusech linek 12 a 13 nacházející se v daném kroku simulace. Tuto *for* smyčku provádím tak, že v každém simulačním kroku si pomocí příkazu *traci.vehicle.getIDList()* vypíšu aktuální vozidla v síti. Z těchto vozidel následně vyberu jen ty, které mají příslušný název jako například *trolleybus12_full.0*. Pokud se vozidlo s tímto názvem nachází v seznamu vozidel, načtu si od něj všechny potřebné informace. Pokud se vozidlo v seznamu nenachází, tuto část přeskočím a do struktur doplním „NaN“ (*not a number*). Na Obr. 5.4 je zobrazeno zjišťování informací o vozidlech linky 12 pomocí příkazů doplňku TraCI. Obdobně je to uděláno i pro trolejbusy linky 13.

```
%% VYTVORENI REZIMU VYBEHU POMOCI TRACI
clear
close all
clc
import traci.constants

%definice adresare se simulaci
scenarioDir = 'C:\Users\turak\Documents\!Skola\Bakalarska prace\SUMO\';
scenarioPath = [scenarioDir 'simulation_3000cars.sumocfg '];

%nastaveni simulace
disp('run simulation')
traci.start(['sumo-gui -c ' '' scenarioPath '' ' --start --quit-on-end']);
step = 0;
end_step = 10800; %nastaveni konce simulace
```

Obr. 5.3: První část skriptu pro režim výběhu

```

for i = 1:pocet/2
    if step > (i-1)*1080
        trl2 = ['trolleybus12_full.',num2str(i-1)]; %nastavovani nazvu vozidla linky 12
        trl3 = ['trolleybus13_full.',num2str(i-1)]; %nastavovani nazvu vozidla linky 13

        %pokud najdu vozidlo v aktualnim seznamu id vozidel, zpracuji data
        %pomoci traci, jinak tuto cast preskocim (struktury doplnim NaNy)

        %LINKA 12
        if ~isempty(find(strcmp(trl2,vehicle_ids),1,'last'))
            speed(step,i) = traci.vehicle.getSpeed(trl2); %hledani vozidlo linky 12 v seznamu vozidel
            al_speed(step,i) = traci.vehicle.getAllowedSpeed(trl2); %nacitani rychlosti trolejbusu 12
            energy(step,i) = traci.vehicle.getElectricityConsumption(trl2); %nacitani povolene rychlosti 12
            acceleration(step,i) = traci.vehicle.getAcceleration(trl2); %nacitani energie trolejbusu 12
            slope(step,i) = traci.vehicle.getSlope(trl2); %nacitani akcelerace trolejbusu 12
            traci.vehicle.setSpeed(trl2,inf); %nacitani sklonu trasy trolejbusu 12
            v_siti(step,i) = 1; %setreni nastavené rychlosti trolejbusu 12
            %vozidlo linky 12 je v siti
        else
            speed(step,i) = NaN; %pokud vozidlo není v siti,
            energy(step,i) = NaN; %nastavim vsechny hodnoty
            acceleration(step,i) = NaN; %na NaN
            slope(step,i) = NaN;
            rezim_vybehu(step,i) = NaN;
            v_siti(step,i) = 0; %vozidlo linky 12 není v siti
        end
    end
end

```

Obr. 5.4: Zjišťování informací o trolejbech linky 12

Po zjištění informací o všech trolejbech nacházejících se v síti musíme stanovit podmínku pro režim výběhu. Ta je realizována opět pomocí *for* cyklu přes všechny trolejbusy. Podmínka, po které se vozidlo dostane do režimu výběhu má podobu několika složených podmínek. První podmínkou je to, že aktuální čas simulace musí být větší než dva. Tato podmínka je tu, protože v následující podmínce bereme informaci o vozidle v předešlém kroku simulace. Další podmínka porovnává aktuální rychlost trolejbusu s rychlostí trolejbusu v předešlém kroku simulace a to tak, že se tyto rychlosti rovnají. Tato podmínka určuje moment, ve kterém se má dané vozidlo dostat do režimu výběhu, a to v momentě, kdy jede trolejbus konstantní rychlostí. Zároveň se musí trolejbus pohybovat, jinak by se zapínal režim výběhu třeba na zastávce. Poslední podmínkou je to, že trolejbus není v režimu výběhu. Pro shrnutí, podmínka, po které se dané vozidlo dostane do režimu výběhu, zní, pokud je čas simulace větší než dva a zároveň rychlost trolejbusu v aktuálním kroku simulace se rovná rychlosti trolejbusu v předchozím kroku simulace a zároveň tato rychlost není nulová a vozidlo se nenachází v režimu výběhu, nastav pro toto vozidlo režim výběhu. Zároveň spočítám, jak dlouho se vozidlo v režimu výběhu nachází.

Poslední částí programu je samotný režim výběhu. Opět se jedná o *for* cyklus s dvěma částmi, jednou pro linku 12 a druhou pro linku 13. Pokud se některé vozidlo dostane do režimu výběhu, je potřeba spočítat jeho zpomalení. K tomu slouží funkce *accel.m*, která pomocí aktuální rychlosti trolejbusu, sklonu trasy, na které se trolejbus nachází a pomocí

definovaných jízdních odporů vypočte potřebné zpomalení tak, aby byl trakční výkon trolejbusu nulový [16]. Vstupními hodnotami této funkce jsou aktuální rychlost trolejbusu, výkon trolejbusu (který je pro režim výběhu nulový), sklon trasy, po které se pohybuje a také konkrétní parametry modelu spotřeby (viz. Obr. 3.2). Nulový trakční výkon trolejbusu znamená, že trolejbus odebírá energii z troleje (nebo z baterie) pouze pro pomocné pohony jako například klimatizace. Funkce *accel.m* tedy obsahuje výpočet zpomalení vozidla daného pouze jízdními odpory při nulovém trakčním výkonu. Vznikne kubická rovnice, řešením jejíž kořenů poté získáme požadované zpomalení. Následně toto zpomalení přičteme k aktuální rychlosti trolejbusu. Zpomalení přičítáme, jelikož jeho hodnoty jsou záporné a dalo by se vlastně říci, že jde o zrychlení. Tuto upravenou hodnotu rychlosti poté nastavíme danému trolejbusu. Pokud se trolejbus dostane do režimu výběhu, je tento režim defaultně nastavený tak, že vozidlo zpomaluje na 85 % maximální dovolené rychlosti. Pro maximální dovolené rychlosti nad 50 km/h je tato mez upravena tak, že vozidlo zpomaluje na 90 % maximální dovolené rychlosti. Ve výsledku to tedy znamená, že pokud jede vozidlo 50 km/h a následně se dostane do režimu výběhu, klesne jeho rychlost po konci tohoto režimu na 42,5 km/h. U maximální rychlosti vozidla (65 km/h) klesne tato rychlost na 58,5 km/h. Také je zde ošetřeno to, že pokud nastane jiný požadavek na aktuální rychlost, tak trolejbus opustí režim výběhu, aby mohl akcelerovat či zpomalovat. Požadavky na opuštění tohoto režimu jsou zvýšení dovolené rychlosti na úseku, požadavek na brždění do zastávky, před křižovatkou anebo kvůli dopravní situaci. Na Obr. 5.5 je zobrazeno řešení režimu výběhu pro trolejbus linky 12. Pro trolejbus linky 13 je řešení shodné.

```

%režim vyběhu pro trolejbusy linky 12
if režim_vybehu(step,k) == 1
    tt(step,k) = tt(step-1,k)+1;           %pricitani tt pro linku 12
    v = speed(step,k);                   %ziskani aktualni rychlosti
    decel(step,k) = accel(v, 0, slope(step,k)); %vypocetni potrebného zpomalení pomocí fce accel
    traci.vehicle.setSpeed(tr12,v+decel(step,k)); %nastaveni požadované rychlosti

%udrzeni vozidla linky 12 v režimu vybehu
if speed(step,k) > speed(step-tt(step,k)-1,k)*0.85 && speed(step,k) <= 14 %pro 50 km/h
    režim_vybehu(step+1,k) = 1;
elseif speed(step,k) > speed(step-tt(step,k)-1,k)*0.9 && speed(step,k) > 14 %pro 65 km/h
    režim_vybehu(step+1,k) = 1;
else
    režim_vybehu(step+1,k) = 0; %konec režimu vybehu vozidla linky 12
end

```

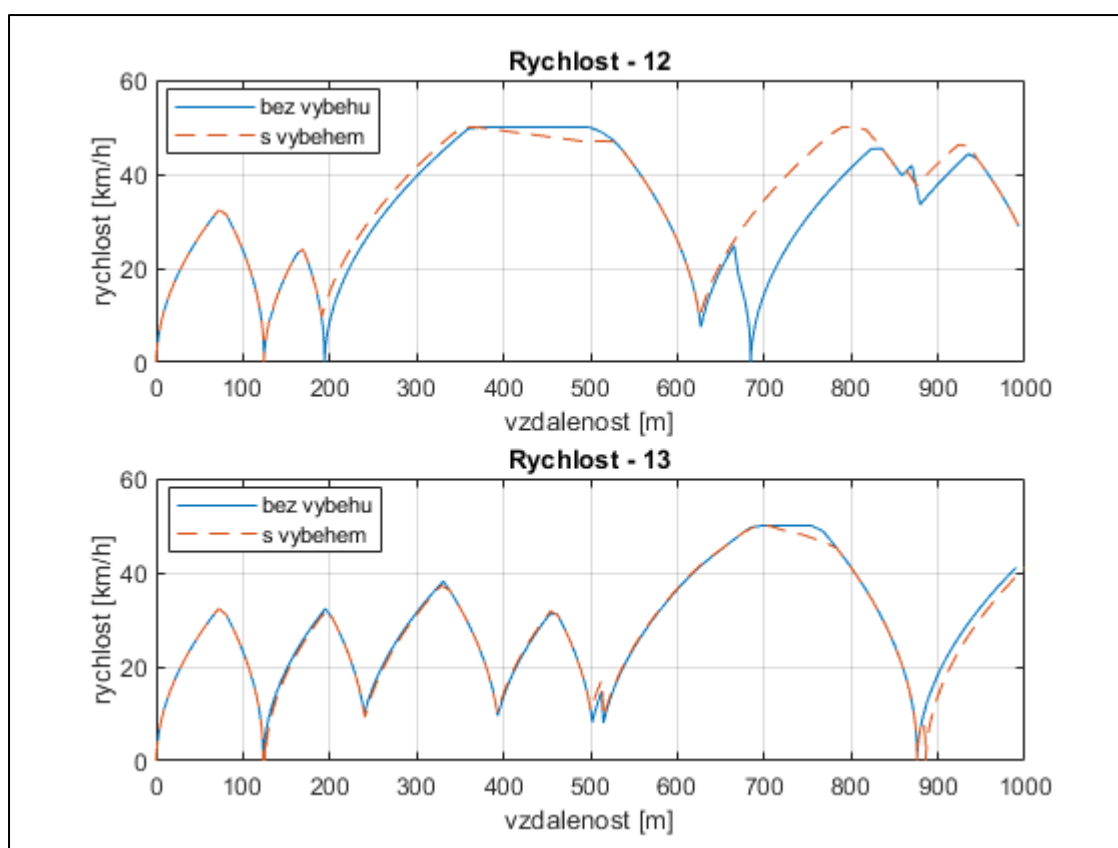
Obr. 5.5: Řešení režimu výběhu pro trolejbusy linky 12

Pokud je aktuální čas simulace větší než čas konce simulace, je potřeba danou simulaci ukončit. K tomu slouží příkaz *traci.close()*. V tomto okamžiku máme tedy nasimulovaná data, takže je opět zpracujeme pomocí skriptu určeného ke zpracování výstupních dat ze

simulátoru. Aby se nám nepřepsaly předchozí výsledky, nastavíme si jiný název pro jednotlivá výstupní data i pro samotný výstupní soubor. Pokud máme výstupní data zpracovaná, vytvoříme další skript pro porovnání výstupních dat z trolejbusů s režimem výběhu a bez něj. Importujeme soubory s nasimulovanými výstupními daty a vykreslíme jejich rozdíly graficky.

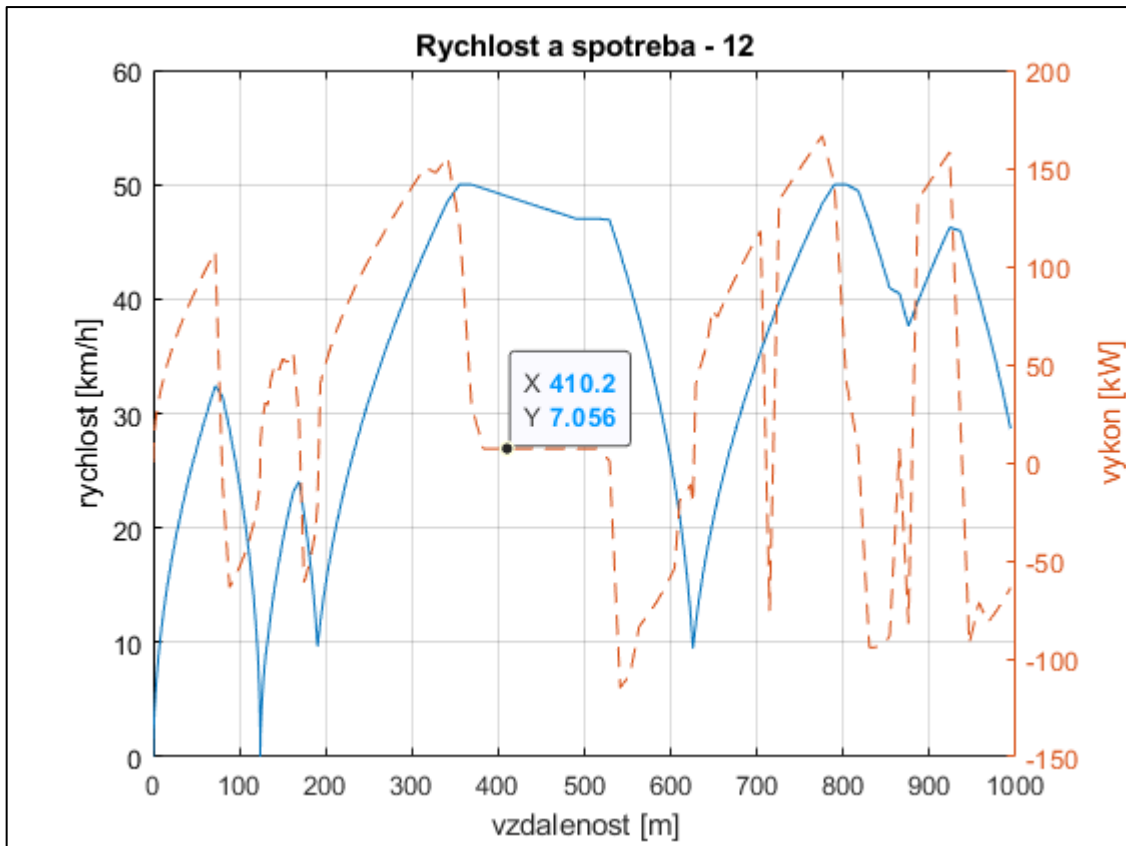
5.2.2 Porovnání výsledků z hybridních trolejbusů s předchozí simulací

Nejdříve se podíváme na rychlost trolejbusů linek 12 a 13 s režimem výběhu a bez něj, který je zobrazen na *Obr. 5.6*. Vidíme, že po dosažení maximální dovolené hodnoty trolejbus s nastaveným režimem výběhu nepokračuje dále konstantní rychlostí, ale zpomaluje. Další rozdíly při porovnání těchto dvou závislostí jsou způsobené okolní dopravou. Trolejbus při režimu výběhu zpomaluje a tím se časově opoždí za trolejbusem bez režimu výběhu. Poté se může stát, že dorazí na světelnou křižovatku o něco později a místo zelené na semaforu dostane červenou a musí tedy zastavit. Může to však být i obráceně. Různé menší odchylky může způsobovat i okolní doprava.

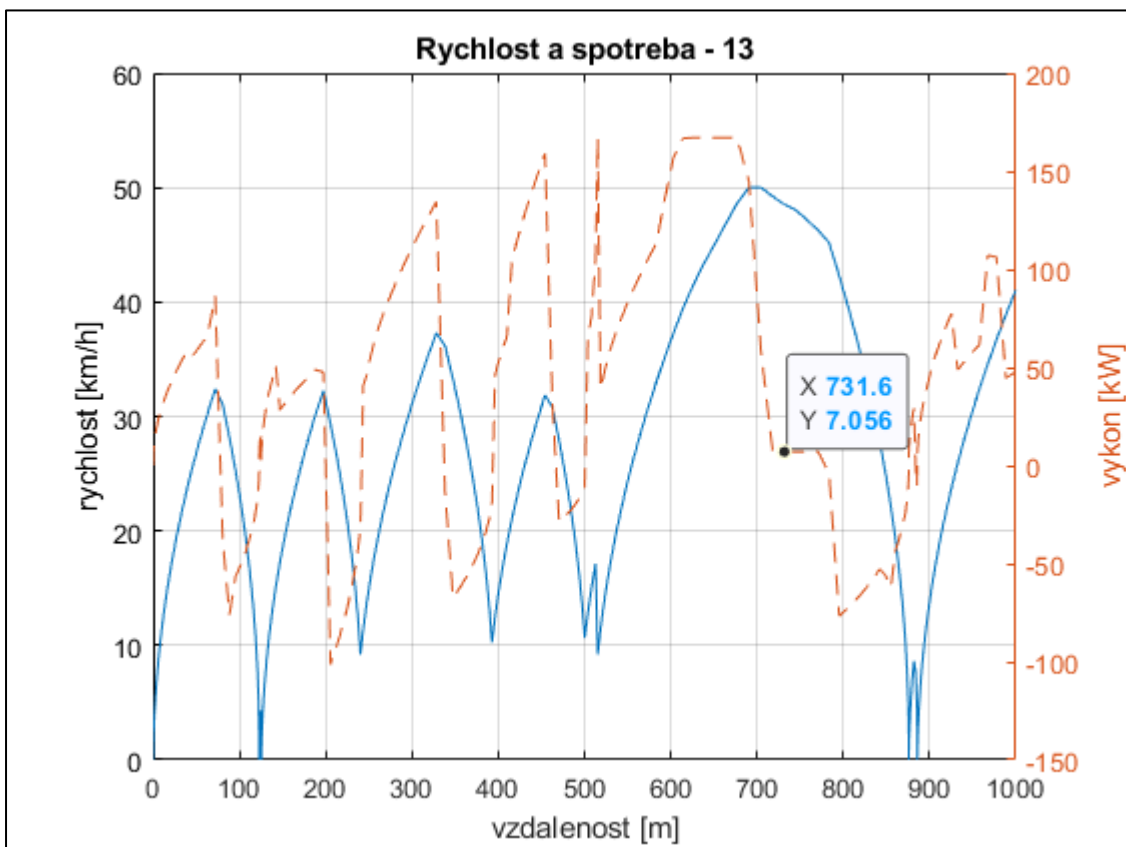


Obr. 5.6: Zobrazení porovnání rychlosti trolejbusu linek 12 a 13 za 1 km

S rychlostí trolejbusu souvisí jeho spotřebovaná energie. Před samotným vykreslením si spotřebu energie ve Wh během jednoho simulačního kroku přepočteme na spotřebovaný výkon trolejbusu v kW. Na *Obr. 5.7* je zobrazen spotřebovaný výkon trolejbusu linky 12 v závislosti na rychlosti trolejbusu, to celé za jeden kilometr. Na dalším *Obr. 5.8* je zobrazeno to samé pro trolejbus linky 13. Na obou obrázcích je zvýrazněna hodnota spotřebovaného výkonu o hodnotě 7,056 kW. Co to znamená? Jde o hodnotu výkonu pouze pomocných pohonů vozidla. Tato hodnota vychází z nastavení trolejbusů v souboru *trolleybus_vType_v0.rou.xml* (viz. *Obr. 3.2*). Je zde definovaný konstantní výkon, který právě trolejbus používá k pohonu pomocných pohonů. Tato hodnota je ale pouze 6 kW. Účinnost těchto pomocných pohonů je předpokládána na 85 %. Pokud tedy konstantní výkon pomocných pohonů vydělíme účinností pohonů, dostaneme právě hodnotu 7,056 kW. Jde tedy o příkon, který je dodáván trolejovým vedením pro provoz pomocných pohonů. Tento stav nastává právě při režimu výběhu. To lze právě vidět na *Obr. 5.7* a na *Obr. 5.8*. Pokud se vozidlo dostane do režimu výběhu a začne zpomalovat pomocí jízdních odporů, tak spotřebovává energii pouze pro pomocné pohony. Trolejbus bez režimu výběhu by udržoval konstantní rychlost a jeho spotřebovaná energie (výkon) z troleje by byl vyšší, jelikož by musel část energie používat k udržení konstantní rychlosti. Také můžeme z obrázků vyčíst, že pokud trolejbus brzdí, tak klesá jeho spotřeba energie (spotřebovaný výkon). To je dáno rekuperací energie, při které si trolejbus část energie vyrobí sám a z troleje poté odebírá zbylou část energie.



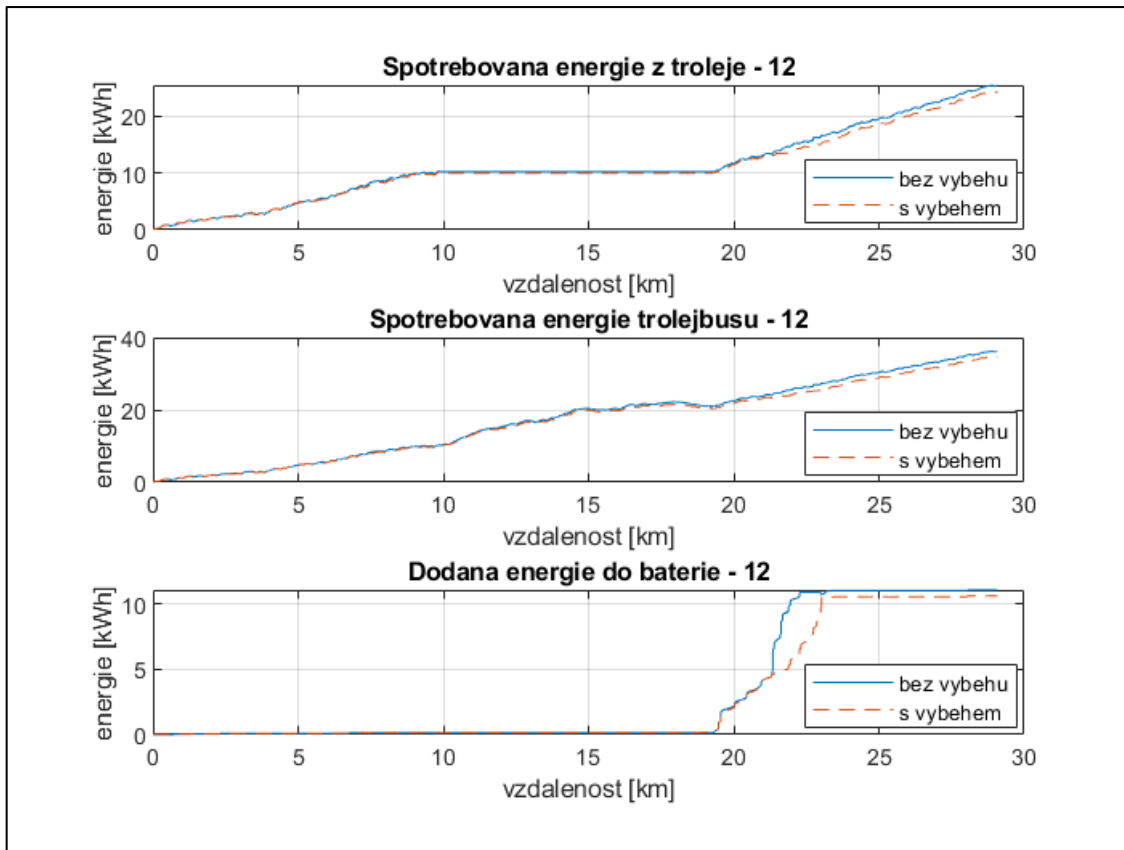
Obr. 5.7: Zobrazení závislosti spotřebovaného výkonu na rychlosti trolejbusu linky 12



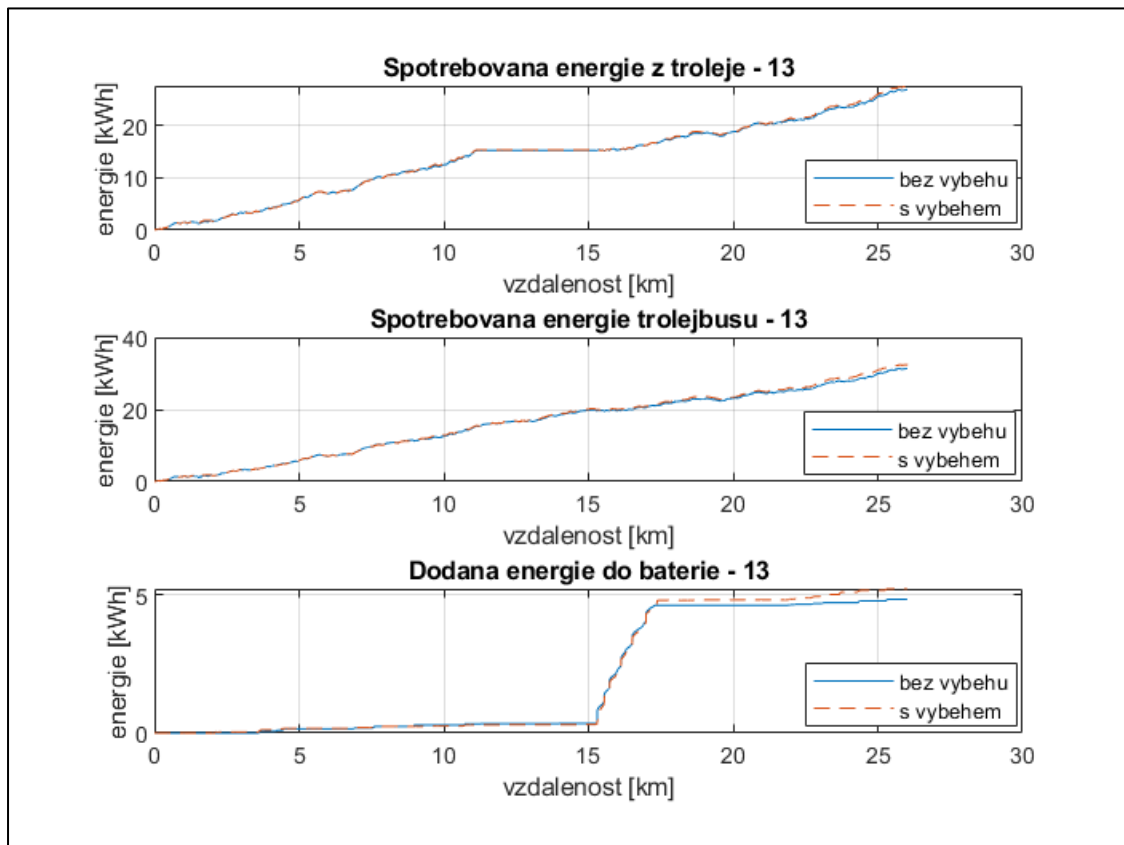
Obr. 5.8: Zobrazení závislosti spotřebovaného výkonu na rychlosti trolejbusu linky 13

Pokud se podíváme na celkovou spotřebu energie z troleje, spotřebu energie hybridního trolejbusu či na energii dodanou z trolejového vedení do baterie, zjistíme opět rozdíly při použití režimu výběhu a bez něj. Na *Obr. 5.9* je zobrazena spotřeba energie z troleje pro trolejbus linky 12, samotná spotřeba trolejbusu linky 12 a také energie, která byla dodána pomocí trolejového vedení do baterie. Na *Obr. 5.10* je zobrazeno to samé pro linku 13. Na těchto charakteristikách je zobrazení kumulace spotřebované energie, tedy energie je s rostoucí vzdáleností nasčítávána.

Po detailnější prohlídce spotřeby energie trolejbusu linky 12 (*Obr. 5.9*), vidíme, že při použití režimu výběhu se spotřebovaná energie odebíraná z trolejového vedení snížila. Je zřejmé, že přibližně mezi desátým a devatenáctým kilometrem je spotřebovaná kumulativní energie z troleje konstantní. To znamená, že se energie pro pohon trolejbusu neodebírání z trolejového vedení, ale z baterie vozidla. Tedy hybridní trolejbusy nejsou v této části připojené k trolejovému vedení a pohybují se pomocí energie z baterie. Po opětovném připojení trolejbusu k trolejovému vedení, ke kterému dochází přibližně na devatenáctém kilometru, je opět odebírána energie z troleje. Kromě odebírání energie pro pohon je odebírána energie i pro nabíjení baterie, která byla v předchozích kilometrech vybíjena. Proto má křivka od tohoto bodu větší nárůst. Energii, kterou dodáváme baterii vidíme na třetím grafu. Z této charakteristiky lze vyčíst, že baterii dodáme energii zhruba 11 kWh. Zároveň tato energie byla trolejbusu dodána při jízdě mimo trolejového vedení mezi desátým a devatenáctým kilometrem. Také vidíme, že tato dodaná energie je menší při použití režimu výběhu. Spotřebovaná energie trolejbusu vyjde tedy také menší při použití režimu výběhu. A proč je spotřeba energie ve všech třech případech menší při použití režimu výběhu? Nejpravděpodobněji je to způsobené zpomalováním trolejbusu v tomto režimu, kde odebírání méně energie (jak z troleje, tak i poté z baterie) než kdyby udržoval konstantní rychlost. Pro udržení rychlosti trolejbus odebírání energie jak pro pomocné pohony, tak i pro samotný pohon. Naopak při zpomalování v režimu výběhu trolejbus odebírání energii pouze pro pomocné pohony a zpomaluje pomocí jízdních odporů. To se uplatní především při jízdě na baterii, tedy trolejbus nespotřebuje tolik energie z baterie. Navíc ještě na spotřebovanou energii může mít vliv dodávání energie do baterie při brzdění na nulovou rychlost (zastavení vozidla). V tomto stavu vozidlo s režimem výběhu zpomaluje z menší rychlosti (42,5 km/h) než vozidlo bez režimu výběhu (50 km/h) a dodá méně energie do baterie.



Obr. 5.9: Zobrazení spotřeby energie trolejbusu linky 12

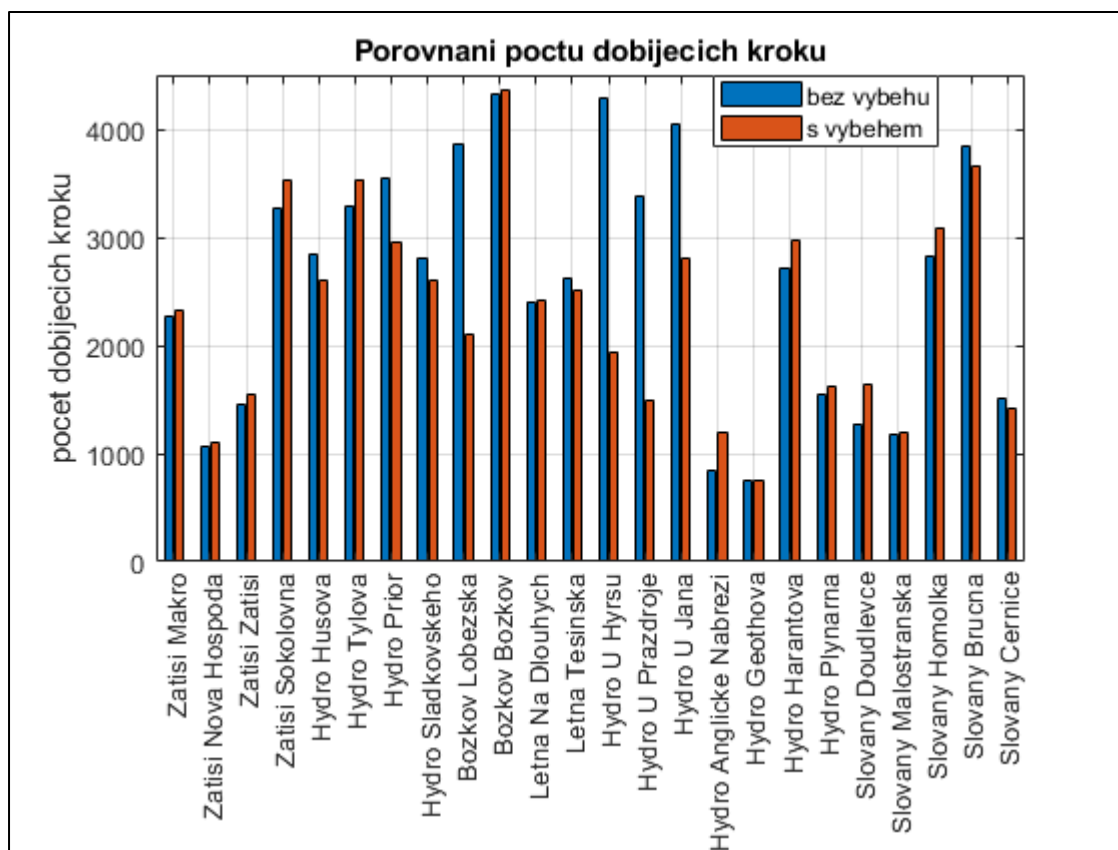


Obr. 5.10: Zobrazení spotřeby energie trolejbusu linky 13

Při pohledu na *Obr. 5.10* zjistíme, že spotřebovaná energie trolejbusu linky 13 nevychází podle předpokladu. Spotřebovaná energie trolejbusu, spotřebovaná energie z troleje i dodaná energie do baterie při použití režimu výběhu je vyšší než bez něj. Proč tomu tak je? Je to nejvíce kvůli spotřebě energie, kterou trolejbus odebírá při jízdě na baterii, kdy není připojen k trolejovému vedení. Pokud se podíváme na výškový profil linky 13 na *Obr. 5.2*, zjistíme, že při jízdě na baterii mezi jedenáctým a patnáctým kilometrem jede hybridní trolejbus většinou do kopce. Spotřeba energie z baterie při jízdě do kopce je vyšší než ušetřená energie vlivem režimu výběhu. Po opětovném připojení vozidla ke trolejovému vedení odebíráme energii pro pohon, ale i pro dobíjení baterie. Proto je i spotřebovaná energie z troleje v režimu výběhu větší. Spotřeba energie může záviset ještě na dalších vlastnostech či parametrech, příkladem může být okolní doprava.

5.2.3 Porovnání výsledků z trakčních měření s předchozí simulací

Průjezdy hybridních trolejbusů s režimem výběhu ovlivnili samozřejmě i výstupy a výsledky z trakčních měření. Parametry trolejového vedení jako jeho délka zůstávají ovšem nezměněny, mění se elektrické veličiny jako například dodaný výkon. Nejdříve se podíváme na změny v počtu dobíjecích kroků jednotlivých trakčních měření. Porovnání počtu dobíjecích kroků s režimem výběhu a bez něj je zobrazen na *Obr. 5.11*. Z tohoto obrázku můžeme vyzorovat, že se počet dobíjecích kroků (cyklů) u téměř všech trakčních měření změnil při použití režimu výběhu. U většiny trakčních měření se počet dobíjecích kroků zvýšil, ale pouze o malý počet (například trakční měřna Zátíší, segment Makro). Naopak u ostatních trakčních měření se počet dobíjecích kroků zmenšil, ale o významný počet (například trakční měřna Hydro, segment U Hyršů). Pokud bychom sečetli jednotlivé sloupce pro oba režimy, vyšlo by nám, že celkový počet dobíjecích kroků bez režimu výběhu je 61928 a pro režim výběhu 55352. To znamená, že režim výběhu v celkovém důsledku snížil počet dobíjecích kroků.

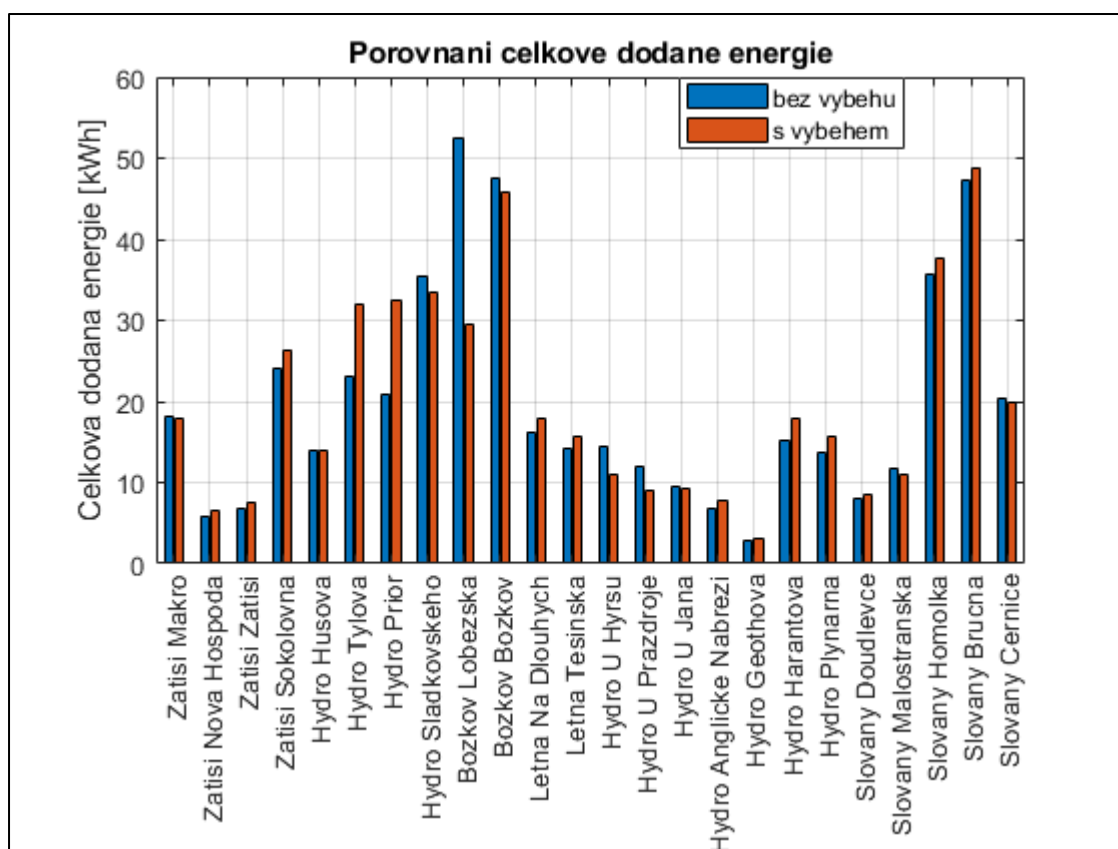


Obr. 5.11: Porovnání počtu dobíjecích kroků jednotlivých trakčních měření

5.2.4 Porovnání výsledků z trakčních měření s předchozí simulací

Na Obr. 5.12 je znázorněno porovnání celkové dodané energie z jednotlivých trakčních měření do trolejového vedení pro potřeby hybridních trolejbusů. Opět vidíme, že ve většině případů se dodaná energie z trakčních měření zvýšila při použití režimu výběhu. Ve většině je toto zvýšení dodané energie jen o několik kWh, ale občas je zvýšení výraznější (trakční měřenína Hydro, segment Prior). V opačných případech, kdy se dodaná energie z trakčních měření snížila při režimu výběhu, je tento pokles většinou zase nevýrazný. Jediný výrazný pokles dodané energie vznikl na trakční měření Božkov, segmentu Lobežská. V celkovém důsledku vyšla celková dodaná energie z trakčních měření bez režimu výběhu přibližně 476 kWh a s režimem výběhu přibližně 478,5 kWh. To znamená, že při použití režimu výběhu se celková dodaná energie z trakčních měření zvýšila, ačkoliv celkový počet dobíjecích kroků se snížil. To je zřejmě způsobeno tím, že hybridní trolejbusy při použití režimu výběhu odebírají energii z troleje méně často, ale s větším výkonem. Dalším důvodem může být to, že hybridní trolejbusy bez režimu výběhu zpomalují z větší rychlosti

(50 km/h) oproti s režimem výběhu (42,5 km/h) a více energie zrekuperují a následně ji využijí nebo ji vrátí zpět do trolejového vedení.

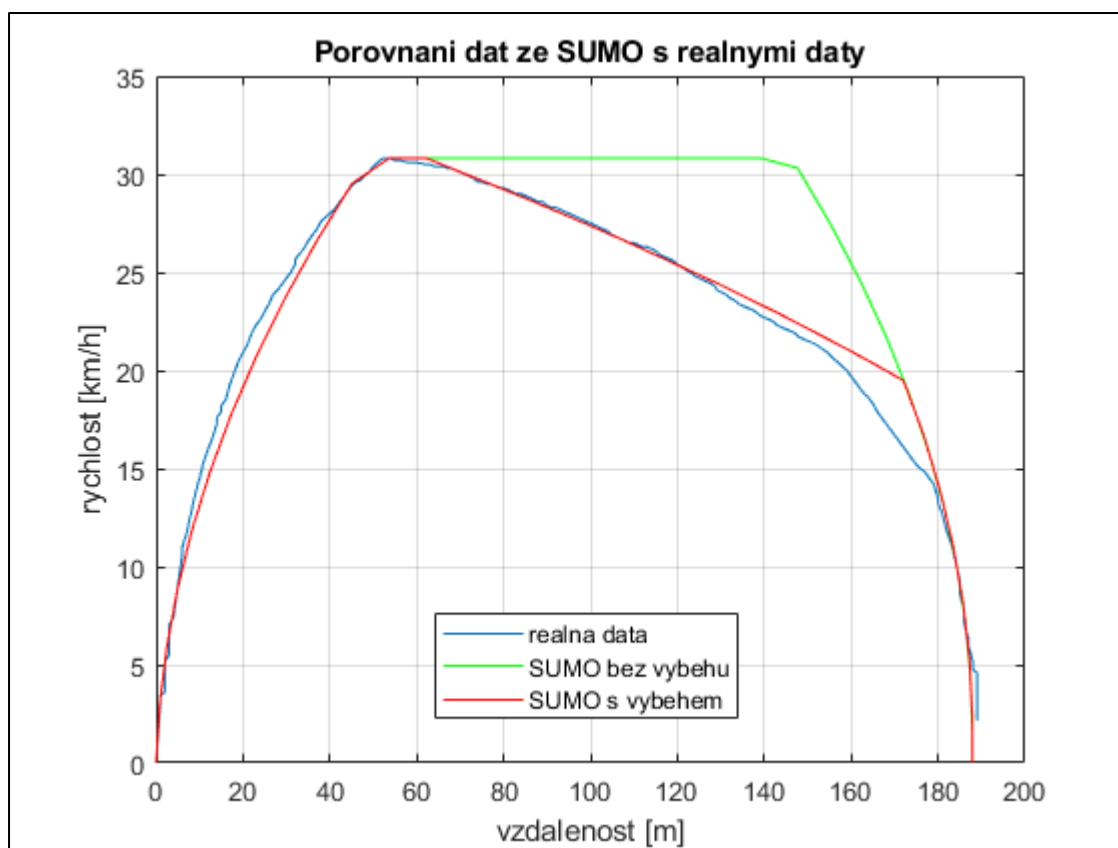


Obr. 5.12: Porovnání celkové dodané energie z jednotlivých trakčních měření

5.3 Porovnání výsledků s reálnými daty

Výsledky ze simulace jak s režimem výběhu, tak i bez něj porovnáme s reálnými daty. Reálná data jsou získána přímo od Plzeňských městských dopravních podniků (PMDP) z reálných průjezdů trolejbusů po dané lince. Těchto dat je velké množství, proto jsem vybral k porovnání pouze vybraný úsek, který je velice názorný. Jde o úsek, kde trolejbus zrychluje z nulové (malé) hodnoty rychlosti na požadovanou rychlost a následně po nějaké ujeté dráze brzdí opět k nulové hodnotě. K tomu jsem vytvořil novou síť tak, aby odpovídala danému úseku. Na této síti jsem opět simuloval trolejbus bez režimu výběhu a následně s režimem výběhu. Na Obr. 5.13 je tedy zobrazeno porovnání těchto třech průběhů. Při porovnání průběhů vidíme, že blíže k reálným datům z trolejbusů má simulace s využitím režimu výběhu. Tedy i v realitě se vozidla pohybují tak, že neudržují konstantní rychlost, ale

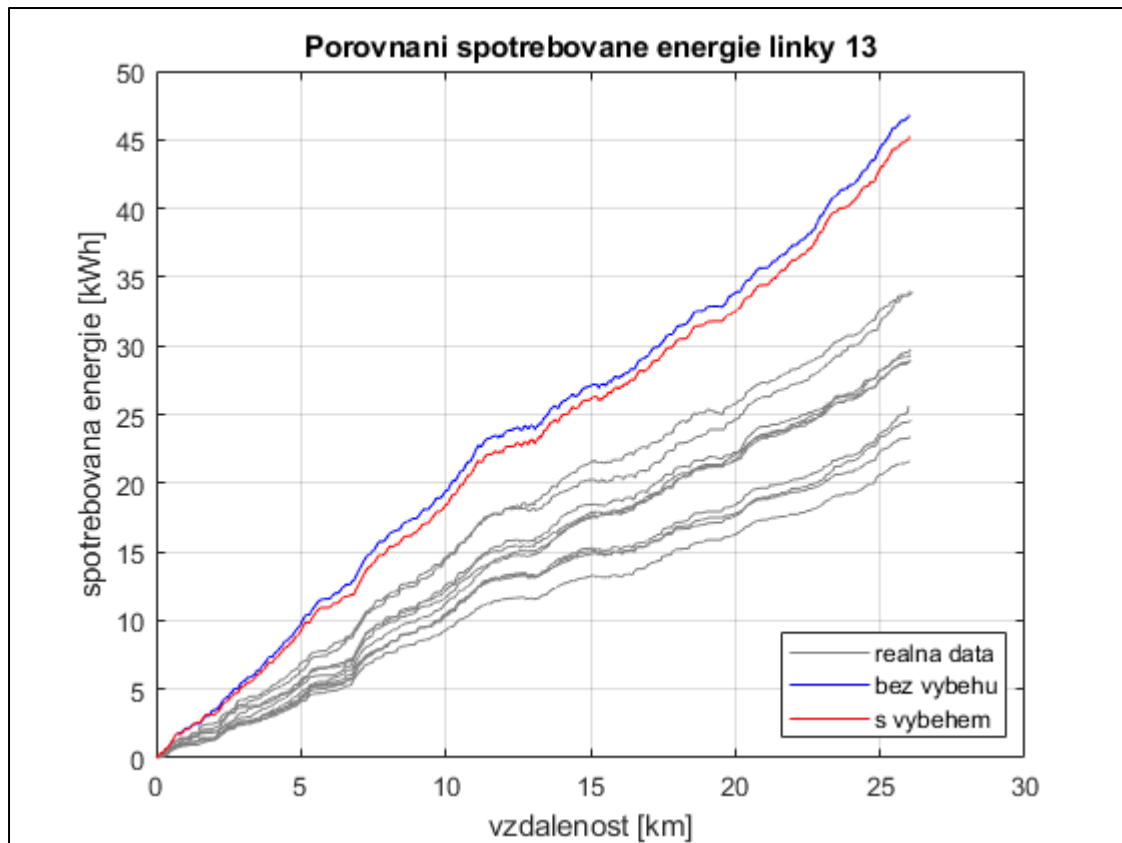
zpomalují pomocí jízdních odporů. Při porovnání průběhu simulace pomocí režimu výběhu s reálným průběhem vidíme, že odchylky nejsou příliš velké. To znamená, že simulace zahrnující režim výběhu je tedy docela přesná oproti simulaci bez režimu výběhu.



Obr. 5.13: Porovnání rychlosti jednotlivých simulací v SUMO s reálnými daty

Na Obr. 5.14 je znázorněno porovnání spotřebované energie hybridního trolejbusu s režimem výběhu a bez použití režimu výběhu s reálnými naměřenými daty na lince 13. Můžeme vidět, že spotřebovaná energie získaná simulacemi je vyšší, než je spotřebovaná energie reálného hybridního trolejbusu linky 13. To je způsobeno nepřesným nastavením parametrů simulace. Parametry, které by se mohly upravovat a doladovat tak, aby se simulace přiblížila reálným hodnotám, jsou uvedeny na Obr. 3.2. Konkrétněji bychom upravovali parametry zrychlení (*accel*) či zpomalení (*decel*), dále pak hmotnost vozidla (*vehicleMass*), čelní plochu vozidla (*frontSurfaceArea*), následně bychom mohli upravit hodnoty jízdních odporů (*airDragCoefficient*, *radialDragCoefficient* a *rollDragCoefficient*) a v neposlední řadě i účinnost pohonu (*propulsionEfficiency*) a účinnost rekuperace (*recuperationEfficiency*). Dále má na tuto spotřebu energie vliv okolní doprava. V simulaci používám okolní dopravu s 3000 vozidly, kdy po uplynutí zhruba desetiny času simulace je síť (město Plzeň) v některých místech velmi ucpaná. Tedy okolní doprava v simulaci zcela

neodpovídá dopravě v reálu. Okolní doprava může tedy způsobovat to, že trolejbus projede část trasy v simulaci o něco déle než v reálu. Tedy odebírá z troleje energii o něco déle. Okolní doprava může také hybridní trolejbus zpomalovat a on poté musí častěji akcelarovat, což je energeticky nevýhodné.



Obr. 5.14: Porovnání spotřebované energie trolejbusu linky 13 s reálnými daty

Závěr

Teoretická část bakalářské práce je rozdělena na dvě kapitoly. První z nich se zabývá elektrickými vozidly, jejich rozdělením a jejich vlastnostmi. Následně je vysvětleno, proč mají elektrická či hybridní vozidla smysl uplatnění v městské hromadné dopravě. Před jejich nasazením do provozu je vhodné provést simulaci daného provozu. K tomu může sloužit simulátory dopravy. Na konci první kapitoly práce je tedy uvedeno rozdělení simulátorů dopravy a následně vybrán vhodný simulátor, kterým je simulační nástroj SUMO. V druhé kapitole této práce je vysvětleno fungování tohoto simulačního nástroje. Také jsou zde uvedeny jeho elektrické modely, které jsou velmi důležité pro simulaci hybridního trolejbusu. Popsány jsou i jednotlivé nástroje, které jsou součástí SUMA. V závěru druhé kapitoly popisují princip ovládání simulátoru dopravy pomocí doplňku TraCI, který je implementován do softwaru Matlab.

Praktická část bakalářské práce je následně rozdělena do třech kapitol. Nejdříve se zabývám vstupními soubory, které jsou použity pro simulaci dopravy. Práce je zaměřena na simulaci dopravy ve městě Plzeň týkající se hybridních trolejbusů linek 12 a 13. Ty jsou napájeny pomocí trolejového vedení z trakčních měníren. Ve vstupních souborech je popsáno veškeré nastavení jak simulace, tak i samotných hybridních trolejbusů či okolní dopravy. Jsou zde uvedeny i konkrétní parametry hybridních trolejbusů. Ve čtvrté kapitole práce popisují samotnou simulaci v simulátoru a její výstupní soubory. Tyto výstupní soubory jsou následně zpracovány pomocí Matlabu. Následně jsou vykresleny výsledky simulace z hybridních trolejbusů, i z trakčních měníren. V této základní simulaci udržuje hybridní trolejbus konstantní maximální dovolenou rychlost. Toto chování neodpovídá rychlostnímu profilu reálného hybridního trolejbusu. Proto je základní simulace vylepšena, aby se její výsledky přiblížily více k reálným průběhům.

V páté kapitole bakalářské práce se zabývám vylepšením simulace. Nejprve je upravena síť města, konkrétněji její výškový profil, jelikož z vykreslení nadmořské výšky linek 12 a 13 byly patrné výškové propady. Ty byly způsobeny importem sítě. Tyto propady jsem odstranil pomocí nástroje *NETEDIT*, který je součástí balíčku simulátoru SUMO. Následně jsem vyvinul skript pro tzv. režim výběhu, který přibližuje chování hybridních trolejbusů k reálnému provozu. Hybridní trolejbusy v režimu výběhu při dosažení maximální dovolené rychlosti začnou zpomalovat pomocí jízdnic odporů. Tento režim je realizován

pomocí softwaru Matlab a následně je pomocí doplňku TraCI simulován v SUMO. Výsledky ze simulace využívající režim výběhu jsou následně porovnány s výsledky základní simulace. Následně jsou výsledky simulace jak s režimem výběhu, tak i bez něj porovnány s reálnými naměřenými daty, které jsou poskytnuty od Plzeňských městských dopravních podniků (PMDP).

Seznam literatury a informačních zdrojů

- [1] Mahmud, K., Town, G. E. *A review of computer tools for modeling electric vehicle energy requirements and their impact on power distribution networks* [online]. 2016 [cit. 26.05.2021]. Dostupné pomocí ScienceDirect. DOI: 10.1016/j.apenergy.2016.03.100.
- [2] *Simulation software MATSim* [online]. 2021 [cit. 26.05.2021]. Dostupné z: <https://www.matsim.org>
- [3] *Simulation software SUMO* [online]. 2021 [cit. 26.05.2021]. Dostupné z: <https://sumo.dlr.de/docs/index.html>
- [4] *Simulation software TRANSIM* [online]. 2021 [cit. 26.05.2021]. Dostupné z: <https://www.transim.com>
- [5] *Simulation software SimTraffic* [online]. 2021 [cit. 26.05.2021]. Dostupné z: <https://www.trafficware.com/>
- [6] ŠKODA. Druhy elektromobilů – znáte je všechny? [online]. 2019 [cit. 26.05.2021]. Dostupné z: Druhy elektromobilů – znáte je všechny? - ŠKODA Storyboard (skoda-storyboard.com)
- [7] Kharrazi, S., Almén, M., Frisk, E., Nielsen. L. *Extending Behavioral Models to Generate Mission-Based Driving Cycles for Data-Driven Vehicle Development* [online]. 2019 [cit. 26.05.2021]. Dostupné pomocí IEEE Xplore. DOI: 10.1109/TVT.2018.2887031.
- [8] Sagaama, I., Kchiche, A., Trojet, W., Kamoun. F. *Proposal of More Accurate Energy Model of Electric Vehicle For SUMO* [online]. 2018 [cit. 26.05.2021]. Dostupné pomocí IEEE Xplore. DOI: 10.1109/IWCMC.2018.8450371.
- [9] Ševčík, J., Příkryl, J. *A Vehicle Device Tailored for Hybrid Trolleybuses and Overhead Wires Implementation in SUMO* [online]. 2019 [cit. 26.05.2021]. Dostupné z: https://www.researchgate.net/publication/335301819_A_Vehicle_Device_Tailored_for_Hybrid_Trolleybuses_and_Overhead_Wires_Implementation_in_SUMO
- [10] Silva, D., Perrotta, D., Seyfried, A., Sánta, C., Ventresque, A., Chraïbi, M., Dalaff, C., Ebdndt, R., Erdmann, J., Gurczik, G., Soares, G., Tapani, A., Zaksek, T., Samal, M., Krajzewicz, D., Krumnow, M., Schneider, E., Li, W., Pesch, D., Wenger, A. *1st SUMO User Conference 2013* [online]. 2013 [cit. 26.05.2021] Dostupné z: https://sumo.dlr.de/2013/SUMO2013_15-17May%202013_Berlin-Adlershof.pdf
- [11] Behrisch, M., Bieker-Walz, L., Erdmann, J., Krajzewicz, D. *SUMO – Simulation of Urban MObility: An Overview* [online]. 2011 [cit. 26.05.2021]. Dostupné z: https://www.researchgate.net/publication/225022282_SUMO_-_Simulation_of_Urban_MObility_An_Overview
- [12] Alvarez Lopez, P., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E. *Microscopic Traffic Simulation using SUMO* [online]. 2018 [cit. 26.05.2021]. Dostupné z: <https://elib.dlr.de/127994/1/08569938.pdf>
- [13] *Electric* [online]. 2021 [cit. 26.05.2021]. Dostupné z: <https://sumo.dlr.de/docs/Models/Electric.html>

- [14] *ElectricHybrid* [online]. 2021 [cit. 26.05.2021]. Dostupné z:
<https://sumo.dlr.de/docs/Models/ElectricHybrid.html>
- [15] *TraCI* [online]. 2021 [cit. 26.05.2021]. Dostupné z:
<https://sumo.dlr.de/docs/TraCI.html>
- [16] *HelpersEnergy* [online]. 2021 [cit. 26.05.2021]. Dostupné z:
<https://github.com/eclipse/sumo/blob/master/src/Utils/emissions/HelpersEnergy.cpp>
- [17] *File Extensions* [online]. 2021 [cit. 26.05.2021]. Dostupné z
https://sumo.dlr.de/docs/Other/File_Extensions.html
- [18] RICE. *Elektrotechnické technologie s vysokým podílem vestavěné inteligence Studie proveditelnosti* [online]. Plzeň: Západočeská univerzita v Plzni, 2018 [cit. 26.05.2021].

Seznam obrázků

OBR. 2.1: PRINCIP VZTAHŮ ELEKTRICKÉ SÍTĚ A HYBRIDNÍCH TROLEJBUSŮ	20
OBR. 3.1: UKÁZKA SÍTĚ PLZNĚ V NÁSTROJI NETEDIT A KONKRÉTNÍ KŘÍŽOVATKY	27
OBR. 3.2: KONKRÉTNÍ PARAMETRY HYBRIDNÍCH TROLEJBUSŮ	28
OBR. 4.1: UKÁZKA PROSTŘEDÍ SUMO-GUI PO SPUŠTĚNÍ.....	30
OBR. 4.2: ZPRACOVÁNÍ DAT ZE SOUBORU	34
OBR. 4.3: ZNÁZORNĚNÍ DÉLKY TROLEJOVÉHO VEDENÍ JEDNOTLIVÝCH TRAKČNÍCH MĚNÍREN	35
OBR. 4.4: ZNÁZORNĚNÍ CELKOVÉ DODANÉ ENERGIE JEDNOTLIVÝCH TRAKČNÍCH MĚNÍREN	36
OBR. 4.5: ZNÁZORNĚNÍ POČTU DOBÍJECÍCH KROKŮ JEDNOTLIVÝCH TRAKČNÍCH MĚNÍREN	36
OBR. 4.6: ZÁVISLOST NADMOŘSKÉ VÝŠKY NA VZDÁLENOSTI TROLEJBUSŮ LINEK 12 A 13	37
OBR. 4.7: ZÁVISLOST RYCHLOSTI NA VZDÁLENOSTI PRO PRVNÍ 2 KM LINKY 12 A 13	39
OBR. 4.8: ENERGETICKÉ ZÁVISLOSTI TROLEJBUSU LINKY 12 ZA 1 KM.....	40
OBR. 4.9: ENERGETICKÉ ZÁVISLOSTI TROLEJBUSU LINKY 13 ZA 1 KM.....	40
OBR. 4.10: ENERGETICKÉ ZÁVISLOSTI TROLEJBUSU LINKY 12	41
OBR. 4.11: ENERGETICKÉ ZÁVISLOSTI TROLEJBUSU LINKY 13	41
OBR. 4.12: ZÁVISLOST STAVU BATERIE NA VZDÁLENOSTI TROLEJBUSŮ LINEK 12 A 13	42
OBR. 5.1: NÁHLED NA ÚPRAVU VÝŠKOPISŮ.....	44
OBR. 5.2: VÝSLEDEK ÚPRAVY VÝŠKOPISŮ SÍTĚ	44
OBR. 5.3: PRVNÍ ČÁST SKRIPTU PRO REŽIM VÝBĚHU	46
OBR. 5.4: ZJIŠŤOVÁNÍ INFORMACÍ O TROLEJBUSECH LINKY 12.....	47
OBR. 5.5: ŘEŠENÍ REŽIMU VÝBĚHU PRO TROLEJBUSY LINKY 12	48
OBR. 5.6: ZOBRAZENÍ POROVNÁNÍ RYCHLOSTI TROLEJBUSU LINEK 12 A 13 ZA 1 KM	49
OBR. 5.7: ZOBRAZENÍ ZÁVISLOSTI SPOTŘEBOVANÉHO VÝKONU NA RYCHLOSTI TROLEJBUSU LINKY 12	51
OBR. 5.8: ZOBRAZENÍ ZÁVISLOSTI SPOTŘEBOVANÉHO VÝKONU NA RYCHLOSTI TROLEJBUSU LINKY 13	51
OBR. 5.9: ZOBRAZENÍ SPOTŘEBY ENERGIE TROLEJBUSU LINKY 12	53
OBR. 5.10: ZOBRAZENÍ SPOTŘEBY ENERGIE TROLEJBUSU LINKY 13	53
OBR. 5.11: POROVNÁNÍ POČTU DOBÍJECÍCH KROKŮ JEDNOTLIVÝCH TRAKČNÍCH MĚNÍREN	55
OBR. 5.12: POROVNÁNÍ CELKOVÉ DODANÉ ENERGIE Z JEDNOTLIVÝCH TRAKČNÍCH MĚNÍREN	56
OBR. 5.13: POROVNÁNÍ RYCHLOSTI JEDNOTLIVÝCH SIMULACÍ V SUMO S REÁLNÝMI DATY	57
OBR. 5.14: POROVNÁNÍ SPOTŘEBOVANÉ ENERGIE TROLEJBUSU LINKY 13 S REÁLNÝMI DATY	58

Seznam tabulek

TAB. 2.1: ELEKTRICKÉ PARAMETRY VOZIDLA	18
TAB. 2.2: ROZŠÍŘUJÍCÍ ELEKTRICKÉ HYBRIDNÍ PARAMETRY VOZIDLA	22
TAB. 4.1: SEZNAM VÝSTUPŮ Z HYBRIDNÍCH TROLEJBUSŮ A JEJICH VÝZNAM	32
TAB. 4.2: SEZNAM VÝSTUPŮ Z TRAKČNÍCH MĚNÍREN A JEJICH VÝZNAM	33

Přílohy

Příloha A – Skript pro simulaci bez režimu výběhu pomocí TraCI

```
%% SIMULACE BEZ REZIMU VYBEHU
clear
close all
clc
import traci.constants

%definice adresare se simulaci
scenarioDir = 'C:\Users\turak\Documents\!Skola\Bakalarska prace\SUMO\';
scenarioPath = [scenarioDir 'simulation_3000cars_old.sumocfg '];

%nastaveni simulace
disp('run simulation')
traci.start(['sumo-gui -c ' "' scenarioPath "' ' --start --quit-on-
end']);
step = 0;
end_step = 10800; %nastaveni konce simulace

%% ZACATEK PROGRAMU
while step < end_step
    step = step + 1;
    traci.simulationStep();
end

traci.close(); %KONEC SIMULACE
disp('simulation completed')

%% CTENI DAT ZE SIMULACE
disp('waiting...'); pause(5);
disp('read data using zpracovani_vystupu')
zpracovani_vystupu;
disp('done');
```

Příloha B – Skript pro simulaci režimu výběhu pomocí TraCI

```
%% VYTVORENI REZIMU VYBEHU POMOCI TRACI
clear
close all
clc
import traci.constants

%definice adresare se simulaci
scenarioDir = 'C:\Users\turak\Documents\!Skola\Bakalarska prace\SUMO\';
scenarioPath = [scenarioDir 'simulation_3000cars.sumocfg '];

%nastaveni simulace
disp('run simulation')
traci.start(['sumo-gui -c ' '' scenarioPath '' ' --start --quit-on-
end']);
step = 0;
end_step = 10800;      %nastaveni konce simulace

%inicializace promennych
if end_step < 7100
    pocet = 2*(fix(end_step/1080)+1);
else
    pocet = 2*(fix(7100/1080)+1);
end
tt = zeros(end_step,pocet);
speed = zeros(end_step,pocet);
al_speed = zeros(end_step,pocet);
energy = zeros(end_step,pocet);
acceleration = zeros(end_step,pocet);
slope = zeros(end_step,pocet);
rezim_vybehu = zeros(end_step,pocet);
decel = zeros(step,pocet);
v_siti = zeros(step,pocet);

%% ZACATEK PROGRAMU
while step < end_step
    step = step + 1;
    traci.simulationStep();
    vehicle_ids = traci.vehicle.getIDList();    %nacistani ID vsech vozidel

    for i = 1:pocet/2
        if step > (i-1)*1080
            tr12 = ['trolleybus12_full.',num2str(i-1)];
            tr13 = ['trolleybus13_full.',num2str(i-1)];

            %LINKA 12
            if ~isempty(find(strcmp(tr12,vehicle_ids),1,'last'))
                speed(step,i) = traci.vehicle.getSpeed(tr12);
                al_speed(step,i) = traci.vehicle.getAllowedSpeed(tr12);
                acceleration(step,i) = traci.vehicle.getAcceleration(tr12);
                slope(step,i) = traci.vehicle.getSlope(tr12);
                traci.vehicle.setSpeed(tr12,inf);
                v_siti(step,i) = 1;
            else
                speed(step,i) = NaN;
                energy(step,i) = NaN;
                acceleration(step,i) = NaN;
                slope(step,i) = NaN;
                rezim_vybehu(step,i) = NaN;
                v_siti(step,i) = 0;
            end
        end
    end
end
```

```
end

%LINKA 13
if ~isempty(find(strcmp(tr13,vehicle_ids),1,'last'))
    speed(step,i+pocet/2) = traci.vehicle.getSpeed(tr13);
    al_speed(step,i+pocet/2) =
traci.vehicle.getAllowedSpeed(tr13);
    acceleration(step,i+pocet/2) =
traci.vehicle.getAcceleration(tr13);
    slope(step,i+pocet/2) = traci.vehicle.getSlope(tr13);
    traci.vehicle.setSpeed(tr13,inf);
    v_siti(step,i+pocet/2) = 1;
else
    speed(step,i+pocet/2) = NaN;
    energy(step,i+pocet/2) = NaN;
    acceleration(step,i+pocet/2) = NaN;
    slope(step,i+pocet/2) = NaN;
    v_siti(step,i+pocet/2) = 0;
end
end
end

%podminka po ktere se dostanu do rezimu vybehu
for j = 1:pocet
    if step > 2 && speed(step,j) == speed(step-1,j) && speed(step,j)
~= 0 && rezim_vybehu(step,j) == 0
        rezim_vybehu(step,j) = 1;
        tt(step,j) = 0;
    end
end

%% REZIM VYBEHU
for k = 1:pocet/2
    if step > (k-1)*1080
        tr12 = ['trolleybus12_full.',num2str(k-1)];
        tr13 = ['trolleybus13_full.',num2str(k-1)];
    end

    %rezim vybehu pro trolejbusy linky 12
    if rezim_vybehu(step,k) == 1
        tt(step,k) = tt(step-1,k)+1;
        v = speed(step,k);
        decel(step,k) = accel(v, 0, slope(step,k));
        traci.vehicle.setSpeed(tr12,v+decel(step,k));

        %udrzeni vozidla linky 12 v rezimu vybehu
        if speed(step,k) > speed(step-tt(step,k)-1,k)*0.85 &&
speed(step,k) <= 14 %pro 50 km/h
            rezim_vybehu(step+1,k) = 1;
        elseif speed(step,k) > speed(step-tt(step,k)-1,k)*0.9 &&
speed(step,k) > 14 %pro 65 km/h
            rezim_vybehu(step+1,k) = 1;
        else
            rezim_vybehu(step+1,k) = 0;
        end
    end

    %rezim vybehu pro trolejbusy linky 13
    if rezim_vybehu(step,k+pocet/2) == 1
        tt(step,k+pocet/2) = tt(step-1,k+pocet/2)+1;
```

```
v = speed(step, k+pocet/2);
decel(step, k+pocet/2) = accel(v, 0, slope(step, k+pocet/2));
traci.vehicle.setSpeed(tr13, v+decel(step, k+pocet/2));

%udrzeni vozidla linky 13 v rezimu vybehu
if speed(step, k+pocet/2) > speed(step-tt(step, k+pocet/2)-
1, k+pocet/2)*0.85 && speed(step, k+pocet/2) <= 14 %pro 50 km/h
    rezim_vybehu(step+1, k+pocet/2) = 1;
elseif speed(step, k+pocet/2) > speed(step-tt(step, k+pocet/2)-
1, k+pocet/2)*0.9 && speed(step, k+pocet/2) > 14 %pro 65 km/h
    rezim_vybehu(step+1, k+pocet/2) = 1;
else
    rezim_vybehu(step+1, k+pocet/2) = 0;
end
end
end
end %od cyklu while - konec programu
traci.close(); %KONEC SIMULACE
disp('simulation completed')

%% CTENI DAT ZE SIMULACE
disp('waiting...'); pause(5);
disp('read data using zpracovani_vystupu_vybeh')
zpracovani_vystupu;
disp('done');
```

Příloha C – Skript pro výpočet požadovaného zpomalení pro režim výběhu

```

function a = accel(lastV, P, slope)

%parameters
eff = 0.85;
mass = 13000;
rollDrag = 0.0083;
TS = 1;
m_rot = 397;
S_front = 7.5;
airDrag = 1.1528;
g = 9.80665

if P > 0
    Prest = P * 3600 * eff;
else
    Prest = P * 3600 / eff;
end

% calculate power drop due to a potential energy difference
Prest = Prest - (mass * g * sin(deg2rad(slope)) * lastV);
const1 = mass * g * sin(deg2rad(slope)) * TS;

% Power loss through Roll resistance [W]
Prest = Prest - (rollDrag * g * mass * lastV);
const1 = const1 + (rollDrag * g * mass * TS);

% Constant loads are omitted. We assume P as the max limit for the main
traction drive. Constant loads are often covered by an auxiliary drive
% Power loss through constant loads (e.g. A/C) [W]
% Prest -= param->find(SUMO_ATTR_CONSTANTPOWERINTAKE)->second / TS;

% kinetic energy difference of vehicle
const1 = const1 + (0.5 * mass * (2 * lastV));
const2 = 0.5 * mass * (TS);

% add rotational energy diff of internal rotating elements
const1 = const1 + (0.5 * m_rot * (2 * lastV));
const2 = const2 + (0.5 * m_rot * (TS));

% Energy loss through Air resistance [Ws]
Prest = Prest - (0.5 * 1.2041 * S_front * airDrag * (lastV * lastV *
lastV));
const1 = const1 + (0.5 * 1.2041 * S_front * airDrag * (3 * lastV * lastV
* TS));
const2 = const2 + (0.5 * 1.2041 * S_front * airDrag * (3 * lastV * TS *
TS));
const3 = 0.5 * 1.2041 * S_front * airDrag * (TS * TS * TS);

% Prest = const1*a + const2*a^2 + const3*a^3
% solve cubic equation in a

a = roots([const3 const2 const1 -Prest]);

im_a = imag(a);
a = a(im_a == 0);

a = max(a);

```