

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

MIKROSKOPICKÁ DOPRAVNÍ SIMULACE S GUI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

TIBOR FÜLÖP

BRNO 2012



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

MIKROSKOPICKÁ DOPRAVNÍ SIMULACE S GUI

MICROSCOPIC TRAFFIC SIMULATION WITH GUI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

VEDOUCÍ PRÁCE
SUPERVISOR

TIBOR FÜLÖP

Ing. PAVOL KORČEK

BRNO 2012

Abstrakt

V současnosti se silniční doprava výrazně rozvíjí a narůstá počet vozidel na komunikacích. Cílem práce je navrhnout mikroskopický simulátor silniční dopravy s grafickým uživatelským prostředím, se kterým bude možné experimentovat. Simulátor umožňuje pomocí uživatelského rozhraní konstrukci silniční topologie s cestami s více pruhy a křižovatkami. Podporuje neoznačené křižovatky, křižovatky s hlavní cestou a semaforey. Během simulace silničního provozu generuje a graficky zobrazuje makroskopické data. Je založen na celulární automatu, model se inspirovuje Nagel-Schreckenbergovým dopravním modelem. V práci se nacházejí výsledky simulací základních dopravních situací.

Abstract

Nowadays the road traffic is more complex and number of vehicles increases. The goal of this thesis is to develop microscopic traffic simulator with graphical user interface which it is possible to experiment with. The simulator can be used to construction road topology with multiple lane roads and crossroads. It supports creating uncontrolled crossroads, crossroads with main road or crossroad with traffic lights. It is based on cellular automaton and model is inspired by Nagel-Schreckenberg traffic model. There are results of simulating simple traffic simulations in the thesis.

Klíčová slova

simulace, doprava, mikroskopický model, celulární automat.

Keywords

simulation, traffic, microscopic model, cellular automaton.

Citace

Tibor Fülöp: Mikroskopická dopravní simulace s GUI, bakalářská práce, Brno, FIT VUT v Brně, 2012

Mikroskopická dopravní simulace s GUI

Prohlášení

Vyhlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením Ing. Pavola Korčeka. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Tibor Fülöp
15. května 2012

Poděkování

Chcem poďakovať môjmu vedúcemu bakalárskej práce Ing. Pavlovi Korčekovi za pomoc a rady pri bakalárskej práci a odborné materiály.

© Tibor Fülöp, 2012.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	2
2 Simulačné modely dopravy	4
2.1 Rozdelenie modelov podľa úrovne detailu	5
2.2 Model celulárneho automatu	5
3 Použitie celulárneho automatu v simulácií dopravy	7
3.1 Nagel-Schreckenbergov model	7
3.1.1 Možné vylepšenia Nagel-Schreckenbergovho modelu	8
3.2 Viacpruhové cesty	9
3.3 Križovatky	9
4 Návrh a implementácia simulátora	11
4.1 Trieda <i>Cell</i>	11
4.2 Trieda <i>Car</i>	13
4.3 Trieda <i>Crossroad</i>	13
4.4 Trieda <i>Generator</i>	14
4.5 Trieda <i>World</i>	15
4.6 Trieda <i>Renderer</i>	16
4.7 Činnosť simulátora	16
4.8 Implementácia križovatiek	17
4.8.1 Prechádzanie ciest križovatky	18
4.8.2 Vyhodnotenie pravidiel pre prechod cez križovatku	19
4.9 Spôsob ukladania topológie do pamäte	19
4.10 Vykresľovanie	19
4.10.1 Prevod z interných súradníc na súradnice plátna a naopak	20
4.10.2 Zoom	20
4.11 Generovanie makroskopických štatistík	21
4.12 Algoritmus uskutočnenia kroku auta	22
5 Použitie simulátora	25
6 Záver	47
A Obsah CD	49
B Manual a obrázky z aplikácie	50

Kapitola 1

Úvod

Nevyhnutnou požiadavkou na činnosť hospodárstva je prítomnosť dopravnej infraštruktúry. Doprava je nutná na prevoz surovín medzi priemyselnými podnikmi, tovaru z podnikov do skladov alebo obchodov. Ľudia ju potrebujú okrem iného na dochádzku do práce a školy. Kvalita infraštruktúry je jedným z najdôležitejších faktorov, ktoré ovplyvňujú rýchlosť ekonomického rozvoja regiónu. Infraštruktúra musí byť dostatočne robustná na to, aby sa tovar mohol prepraviť z ktoréhokoľvek miesta na iné miesto rýchlo, plynulo a bezpečne. Preto v krajinách na celom svete vznikajú štátne inštitúcie zodpovedné za prevádzku, kontrolu a budovanie rôznych druhov dopravnej infraštruktúry. Rozširovanie dopravnej siete si vyžaduje investovať nemalé finančné prostriedky. Preto by mala prebiehať efektívne.

Základnou zložkou dopravnej infraštruktúry je cestná sieť. Vzhľadom na jej výrazné rozšírenie a zvýšené využívanie v posledných desaťročiach vznikli pravidlá na jej používanie. Ich základné princípy sú vo väčšine krajín sveta rovnaké. Vďaka nim je premávka regulovaná tak, aby nedochádzalo ku kolíznym situáciám a bola zabezpečená jej plynulosť. Plynulosť premávky závisí okrem iného aj na šírke cesty a počte pruhov, maximálnej povolenej rýchlosti na úseku, tvaru križovatiek, kvality povrchu vozovky, úrovne prevýšenia, prevládajúcim type vozidiel. Pre štatistické účely sa v súčasnosti uskutočňujú sčítania dopravy. Počet vozidiel prejdejších za jednotku času je základným údajom pre odvodenie úrovne plynulosti na danom úseku a aj od ktorého sa odvíja budovanie novej alebo rozširovanie existujúcej infraštruktúry, tak ako aj nastavenie pravidiel cestnej premávky. Vznikajú aj počítačové dopravné simulátory, ktorých cieľom je na základe vonkajších údajov (počet vozidiel, tvar križovatiek) uskutočniť simuláciu premávky. Výstupom sú údaje, ktoré môžu slúžiť na optimalizáciu návrhov budovania infraštruktúry alebo na sledovanie dopadov neobvyklej situácie na ceste.

Súčasťou tejto práce je návrh dopravného simulátora, ktorý je orientovaný na vyššie spomenuté úlohy. Jednoduché grafické užívateľské rozhranie umožňuje návrh cestnej siete s križovatkami. Na jednotlivé úseky sa nastavuje zvolená maximálna rýchlosť a počet pruhov v oboch smeroch. V rámci križovatky je možné určiť hlavné cesty alebo svetelné signály s periódou zmeny signálov. Jednotlivé vozidlá sú modelované ako entity bez informácie o ich celi. Náhodne sa rozhodujú o smere na nasledujúcej križovatke podľa pravdepodobností, aké križovatka má nastavené na výber cieľovej cesty z križovatky. Vozidlá vznikajú v generátoroch, ktoré sa pripájajú na cesty, na ktoré majú generovať vozidlá. Existujú mnohé riešenia dopravných simulácií. Časť z nich sú komerčné a drahé riešenia, ktoré neumožňujú nastavovanie všetkých parametrov. Cieľom tejto práce je simulátor, s ktorým bude možné experimentovať.

Kapitola 1 zhrňa možnosti simulácie dopravy, ich teoretické východiská a zhrnie niektoré

všeobecné údaje cestnej premávky. Kapitola 2 sa bude podrobnejšie zaoberať simuláciou založenou na celulárnom automate, ktorý využíva implementovaný simulátor. Kapitola 3 bude zameraná na podrobný popis simulátora a na jeho základné princípy. Súčasťou kapitoly bude aj popis objektového modelu simulátora. V kapitole 4 bude v praxi na niektorých situáciach cestnej premávky znázornená činnosť simulátora.

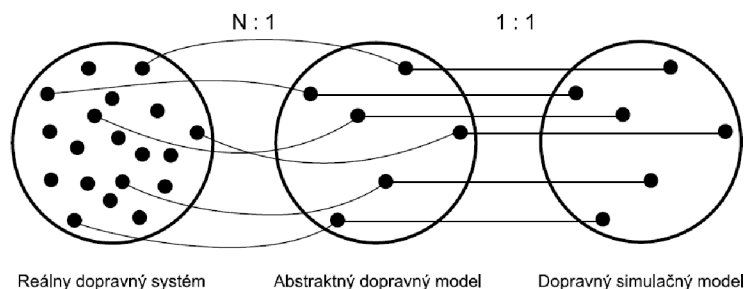
Kapitola 2

Simulačné modely dopravy

Pojem systém v oblasti simulácií definuje súbor elementárnych častí, ktoré majú medzi sebou určité väzby [8]. Dopravu možno chápať ako systém, ktorého prvkami sú cesty, križovatky, vozidlá s vodičmi a podobne. Tieto prvky sa navzájom ovplyvňujú. Ak sa vozidlo blíži ku križovatke alebo ostrej zákrute, vozidlo spomalí. Vozidlá sa ovplyvňujú sami medzi sebou. Ak pomalé vozidlo dobieha niektoré rýchlejšie, podľa podmienok prispôsobí rýchlosť k pomalému vozidlu alebo ho predbehne. Jazda vodiča je ovplyvnená veľkým počtom faktorov ako aj vlastnosťami samotného vodiča. Dokonca aj vozidlá vplývajú na vozovku cesty. Ak je cesta zaťažovaná príliš veľkým počtom hlavne ťažších vozidiel, dochádza k prasklinám na vozovke a celkovo sa jej povrch deformuje. Samotní vodiči majú rôzne vlastnosti, ktoré ovplyvňujú ich štýl jazdy. Agresívnejší vodiči robia riskantnejšie zmeny pruhov a predchádzania, rýchlejšie akcelerujú a brzdia. Vodiči majú rozličnú rýchlosť reakcie na prekážky alebo iné nepredvídané udalosti.

Štatistiky cestnej premávky v Českej republike v podobe informácií o diaľniciach a cestách, o priebehu intenzít dopravy alebo o nehodovosti je možné nájsť na stránkach Ředitelství silnic a dálnic ČR v [12].

Pri modelovaní abstraktného dopravného modelu je potrebné uvažovať systém s rôznymi typmi ciest, križovatiek a celkovo cestnej siete, tak ako aj s vodičmi, ich chovaním a vlastnosťami ich vozidiel. Dôležité je zahrnúť do systému pravidlá cestnej premávky a rôzne signalizačné zariadenia. Počasie a parametre ciest tiež nezanedbateľnou mierou ovplyvňujú systém. Zo systému sa vypúšťajú tie prvky, ktoré majú na dopravu minimálny až žiadny vplyv. Získa sa tým abstraktný dopravný model. Vzhľadom na komplikovanosť systému je možné vypustiť aj tie prvky, ktoré vplývajú na dopravný systém v menšej, prípustnej miere. Niektoré prvky sú za niektorých okolností dôležité, za iných nemusia mať žiadny vplyv. Uskutočňujú sa abstrakcie s rôznou mierou. Pre reálny dopravný systém je tak možné vytvoriť viacero abstraktných modelov a abstraktný model môže odpovedať viacerým systémom, preto medzi reálnym systémom a abstraktným modelom je homomorfný vzťah. Z abstraktného modelu sa vytvára simulačný dopravný model, ktorým môže byť úsek cesty, križovatka alebo aj celá cestná sieť mesta. Medzi abstraktným a simulačným modelom je izomorfný vzťah.



Obrázek 2.1: Znáozornenie vzťahu reálneho systému s abstraktným a simulačným modelom

2.1 Rozdelenie modelov podľa úrovne detailu

Úroveň detailu v dopravných systémoch popisuje, ako podrobne je systém modelovaný. Modely sa najčastejšie delia na mikroskopické a makroskopické [10].

Mikroskopické modely sa zameriavajú na modelovanie správania jednotlivých entít (vozidlá/vodiči). Modelujú vzťahy a interakcie medzi entitami. Ak niektoré vozidlo zahájí manéver (zmena pruhu, predbiehanie), musí to robiť s ohľadom na iné entity.

Makroskopické modely sú na vyššej úrovni abstrakcie ako mikroskopické modely. Vozidlá ako samostatné entity sa nemodelujú. Makroskopické modely sa definujú pomocou troch parametrov: priepustnosť, hustota a priemerná rýchlosť. Medzi nimi platí vzťah: [9]

$$q = ku \quad (2.1.1)$$

kde q je priepustnosť, k je hustota a u je rýchlosť. Podľa týchto parametrov sa dá určiť charakter cestnej premávky. Priepustnosť vyjadruje počet vozidiel prejdejších cez dané miesto za určitý čas, hustota počet vozidiel na určitom úseku cesty. Priemerná rýchlosť sa vypočítava prechodmi vozidiel cez určité miesto počas daného času.

Validácia mikroskopického modelu sa môže uskutočniť umiestnením lokalizačných prijímačov GPS do reálnych vozidiel v sledovanom úseku a porovnať ich trajektórie s modelom. Ďalšou možnosťou je validácia porovnávaním makroskopických charakteristík vozidiel. V reálnom systéme zber makroskopických dát umožňujú rôzne druhy senzorov a kamier. V takom prípade musí mikroskopický model umožňovať generovanie údajov týchto parametrov, čiže simulovať spomínané zaznamenávacie zariadenia.

Existujú rôzne mikroskopické modely simulátorov. Často je to model založený na celulárnom automate (sekcia 2.2) alebo model postavený na základe diferenciálnych rovníc. Ide o spojitý model, v ktorom správanie vodičov je závislý na viacerých faktoroch, ako vlastná rýchlosť, vzdialenosť od najbližšieho vozidla, vzdialenosť k vedúcemu vozidlu alebo aj rýchlosť vedúceho vozidla. Je možné nastavovať akceleráciu vozidla či pohodlné brzdné spomalenie, čím sa dajú modelovať rôzne typy správania vodičov [2].

2.2 Model celulárneho automatu

Celulárny automat sa využíva na modelovanie procesov v mnohých oblastiach. Je založený na rozdelení priestoru modelovaného systému na sústavu buniek s rovnakými veľkosťami. Každá bunka reprezentuje určitú drobnú časť priestoru systému. Celulárny automat môže popisovať jednorozmerný či viacrozmerý priestor. Ľahko sa implementuje na počítačových systémoch, kde sa môže definovať ako pole alebo matica. Každý prvok vektoru popisuje

jednu bunku. Hodnota predstavuje stav bunky a môže sa v čase meniť. Obecné stav bunky v čase závisí na stave samotnom a stavoch buniek v okolí: [8]

$$s(t + 1) = f(s(t), N_s(t)) \quad (2.2.1)$$

V implementovanej simulácii dopravy má bunka v danom čase buď stav obsadená, kedy ňou práve prechádza niektoré vozidlo, alebo stav voľná, kedy sa v jej priestore nenachádza žiadne vozidlo. Základnou variantou celulórneho automatu pre modelovanie dopravy je Nagel-Schreckenbergov model[6], ktorý bude popísaný v nasledujúcej kapitole.

Kapitola 3

Použitie celulárneho automatu v simulácií dopravy

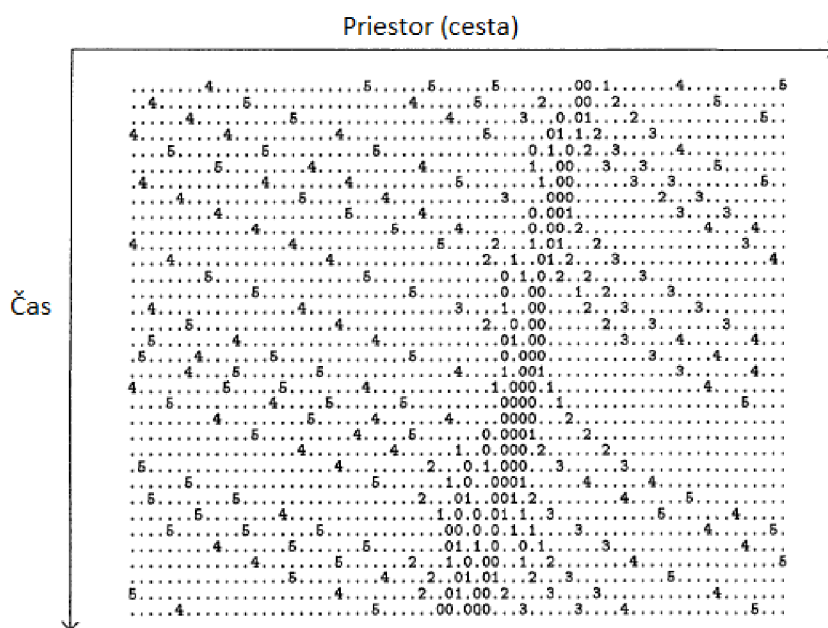
3.1 Nagel-Schreckenbergov model

Model je založený na jednorozmernom poli, pričom každá bunka môže byť buď obsadená alebo voľná. Každé vozidlo má definovanú rýchlosť medzi hodnotou 0 a v_{max} . Po každom kroku sa na všetky vozidlá paralelne aplikujú nasledovné pravidlá: [6]

- **Akcelerácia:** ak rýchlosť vozidla v je nižšia ako v_{max} a vzdialenosť k nasledujúcemu vozidlu je väčšia ako $v + 1$, potom sa zvýši o jednu jednotku: $v := v + 1$.
- **Spomalenie:** ak vozidlo v bunke i má pred sebou vozidlo v bunke $i + j$, pričom $j \leq v$, potom jeho rýchlosť sa zníži: $v := j - 1$.
- **Náhodnosť:** rýchlosť každého pohybujúceho vozidla sa s pravdepodobnosťou p zníži o jednu jednotku: $v := v - 1$. Toto pravidlo je dôležité, keďže zachytáva ľudský faktor vedenia vozidla.
- **Pohyb vozidla:** každé vozidlo sa posunie o v buniek.

Systém založený na Nagel-Schreckenbergovom modeli simuluje cestnú premávku na jednom pruhu. Existujú jeho dva druhy:

- **Otvorený:** vozidlá vystupujúce zo systému sa doňho naspäť nevracajú. Namiesto toho sa musia nové vozidlá generovať.
- **Uzavretý:** vozidlá vystupujúce zo systému pokračujú na bunku tvoriacu začiatok systému. Pred simuláciou sa uskutoční náhodné rozmiestnenie vozidiel.



Obrázek 3.1: Simulácia s Nagel-Schreckenbergovým modelom. Čísla značia aktuálnu rýchlosť vozidiel. Toto je prípad uzavretého typu[6].

3.1.1 Možné vylepšenia Nagel-Schreckenbergovho modelu

Existuje niekoľko možností, ako model ešte vylepšiť. Pravidlo o spomalení má nevýhodu v tom, že vozidlo môže počas jedného kroku upraviť rýchlosť z pomerne vysokej hodnoty, dokonca aj z v_{max} , na 0. Tento prípad sa stáva vtedy, ak sa vozidlo pohybujúce sa vysokou rýchlosťou blíži k stojacej kolóne vozidiel. Riešenie tohto neželaného efektu je úprava pravidla o spomalení tým, že by sa vozidlo pozeralo pred seba o viac ako v buniek, napríklad o $2v$ buniek. Je možné aj ovplyvniť rýchlosť spomaľovania podľa toho, či nasledujúce vozidlo spomaľuje alebo zrýchľuje [7].

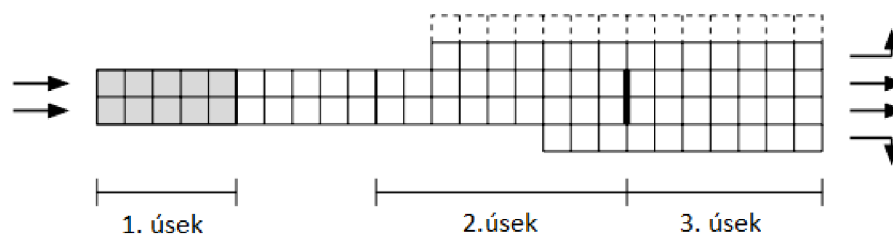
Pravidlo náhodnosti vyvoláva tiež neželaný jav v tom, že vozidlo pohybujúce sa najmenšou možnou rýchlosťou (rýchlosťou jednej bunky za krok) môže až zastaviť. Riešením je aplikácia tohto pravidla len na vozidlá pohybujúce sa rýchlosťou aspoň dve bunky za krok.

Základný model predpokladá, že bunka je taká veľká ako veľkosť vozidla. Bežná veľkosť priemerného vozidla je 7.5 metra. Najnižšia možná rýchlosť vozidla môže byť v takom prípade 7.5 metra za sekundu. Vozidlo môže mať iba rýchlosti deliteľné touto rýchlosťou. Ak by sa veľkosť bunky znížila, bola by k dispozícii širšia škála rýchlostí, ktoré by mohli vozidlá nadobúdať. Takisto existujú vozidlá s rôznou dĺžkou. S jednotnou dĺžkou vozidla 7.5 metra sa v kvalitnom simulátore nedá vystačiť. Preto je potrebné v simulátore zahrnúť možnosť nastavenia ľubovoľnej dĺžky vozidla. Rôzne vozidlá by tak zabrali rôzny počet buniek. Ich dĺžka by bola deliteľná dĺžkou jednej bunky, aby vždy bola bunka buď úplne obsadená alebo voľná.

Vyvinuli sa aj rôzne modifikácie Nagel-Schreckenbergovho modelu, ktoré zohľadňujú ešte viac faktorov ako základný model. Medzi ne patria VDR (velocity-dependent randomization) a BL (brake-light) modifikácie [4]. Tieto modifikácie zohľadňujú rýchlosti iných vozidiel alebo aj reakcie na brzdenie nasledujúcich vozidiel.

3.2 Viacpruhové cesty

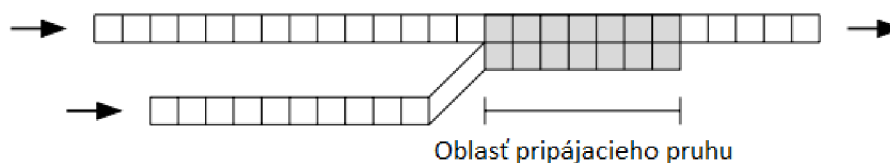
Prechod medzi pruhmi na viacpruhových cestách (viacpruhových v jednom smere) je potrebný pri dodržiavaní požadovanej trasy hlavne pred križovatkami. Preto sa viacpruhové cesty z hľadiska správania vozidiel delia na tri úseky (obrázok 3.2) [3]. Prvý úsek sa nachádza ešte ďaleko pred križovatkou a vozidlá nie sú obmedzované pri zmene pruhu, to znamená že vozidlá môžu napríklad predchádzať iné pomalšie vozidlá. Druhý úsek sa nachádza už nie veľmi ďaleko ale ani ešte bezprostredne pred križovatkou. V ňom vozidlá môžu zmeniť pruh jedine za účelom zachovania požadovanej trasy, keďže pred križovatkou sú pruhy vyhradené na jazdu pre rôzne smery od križovatky. Pred niektorými križovatkami sa cesta rozširuje o ďalšie pruhy za účelom zvýšenia priepustnosti premávky. Miesta, kde vznikajú tieto nové pruhy, musia byť súčasťou práve druhého úseku, aby vozidlá mohli tieto pruhy využiť. V treťom úseku už vozidlo nemá možnosť zmeniť pruh. Do tohto úseku patrí posledná časť cesty pred križovatkou, kde je medzi pruhmi neprerušovaná čiara.



Obrázok 3.2: Viacpruhová cesta pred križovatkou [3].

Pre druhý úsek je typické, že vozidlá menia pruhy viac riskantne a to obzvlášť pred hranicou s tretím úsekom. Môže nastať situácia, keď na hranici druhého a tretieho úseku uviaznu dve vozidlá, pričom obidva z nich sa chcú dostať na pruh druhého vozidla. Tento problém je možné vyriešiť napríklad výmenou vozidiel.

Pre viacpruhové cesty sú taktiež typické spojenie dvoch ciest formou nájazdu. Prechod z nájazdu na hlavný pruh vozidlo uskutoční len pri dostatočnej vzdialenosti od iných vozidiel. V prípade potreby vozidlo na konci nájazdu zastaví. Tieto spojenia sa vyskytujú hlavne pri diaľničných križovatkách.



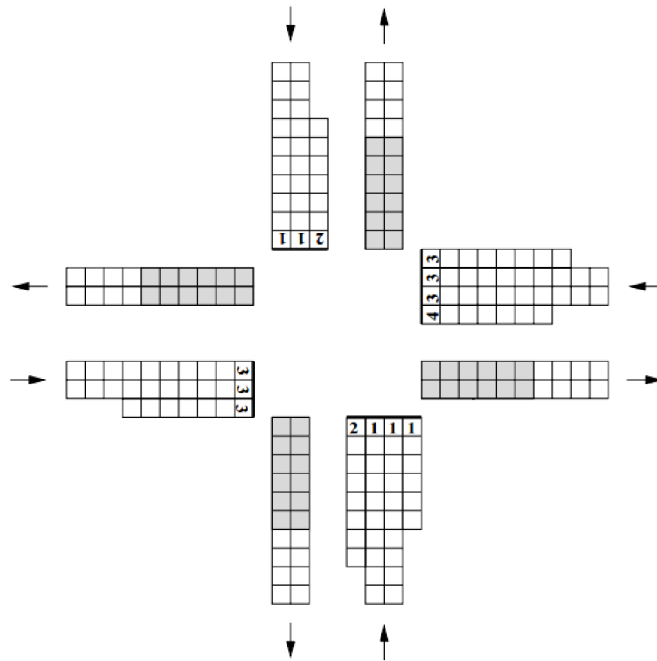
Obrázok 3.3: Nájazd [3].

3.3 Križovatky

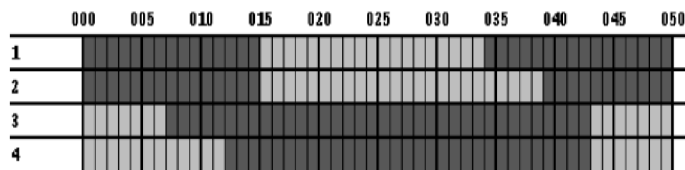
Pre simuláciu križovatky je potrebné asociovať vstupné pruhy jednej cesty s výstupnými pruhmi druhej cesty. Na križovatkou sa môžu nastaviť priority pre jednotlivé prechody re-

spektíve definovať hlavné cesty.

Zvláštnou kategóriou križovatiek sú križovatky so semafori. Na simuláciu semaforov sa používa tzv. matica prepínania. Aby bolo možné ju použiť, určia sa skupiny prechodov cez križovatku, ktoré sa riadia svetlami napojenými na rovnaký signál. Signál môže reprezentovať svietenie zeleného alebo červeného svetla (oranžové svetlo sa uvažuje ako zelené). Každý riadok matice predstavuje pre skupinu prechodov priebeh tohto signálu v čase počas jednej periódy prepínania svetiel. V križovatke 'X' môžu skupiny prechodov tvoriť napríklad všetky prechody vychádzajúce z dvoch ciest umiestnené navzájom oproti. Ak by križovatka mala dodatočný svetelný signál (v tvare šípky), tak by sa daný prechod zodpovedajúci odbočke doľava alebo doprava priradil do novej skupiny prechodov a mal by tak osobitný riadok v matici prepínania [3].



Obrázek 3.4: Križovatka s pruhmi asociovanými svetelným signálom 1-4 [3].

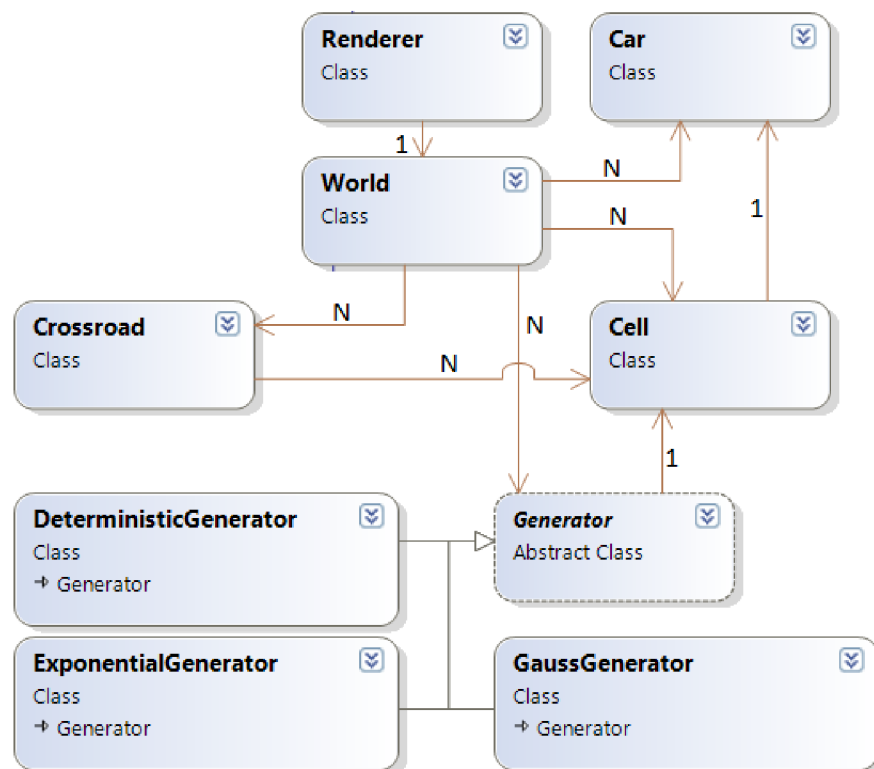


Obrázek 3.5: Matica prepínania s periódou 50 sekúnd. Svetlé obdĺžniky predstavujú zelený signál, tmavé červený signál [3].

Kapitola 4

Návrh a implementácia simulátora

Simulátor je napísaný v objektovo-orientovanom jazyku C#. Užívateľské rozhranie je navrhnuté nadstavbou Windows Presentation Framework [1]. Projekt bol vyvíjaný v prostredí Visual Studio 2010. V nasledujúcom obrázku je objektový model navrhnutého simulátora s podstatnými triedami a ich atribútmi a metódami.



Obrázek 4.1: Objektový model simulátora.

4.1 Trieda *Cell*

Reprezentuje bunku celulárneho automatu. Dôležité atribúty tejto triedy sú:

- *AllowedSpeed*: maximálna povolená rýchlosť v bunke.
- *Car*: referencia na vozidlo, ktoré obsadzuje bunku.
- *Location*: poloha bunky.
- *DirectionVector*: orientácia bunky vyjadrená normalizovaným vektorom.
- *CellOnTheLeft*: referencia na bunku vľavo z pohľadu v smere orientácie bunky.
- *CellOnTheRight*: referencia na bunku vpravo z pohľadu v smere orientácie bunky.
- *FollowingCell*: referencia na nasledujúcu bunku.
- *PreviousCell*: referencia na predchádzajúcu bunku.
- *IsMainRoad*: príznak, či bunka je súčasťou hlavnej cesty v najbližšej križovatke v smere orientácie.
- *NearestCrossroadDirection*: referencia na inštanciu triedy *Crossroad*, na najbližšiu križovatku v smere orientácie bunky.
- *nextCrossroadDirections*: množina normalizovaných vektorov vyjadrujúcich orientáciu v smere od križovatky tých ciest, na ktoré sa v najbližšej križovatke bude možné dostať z pruhu, na ktorom sa nachádza bunka.
- *carSpeeds*: zoznam rýchlostí vozidiel, ktoré v minulosti prešli cez túto bunku.
- *timesOfCarCrossing*: zoznam časov posledných vozidiel, ktoré prešli v minulosti touto bunkou.

Referencie na okolité bunky sa využívajú pri presúvaní vozidla alebo pri hľadaní vozidiel v okolitom priestore, s ktorými by hrozila kolízia. Zoznamy *carSpeeds* a *timesOfCarCrossing* sa využívajú na výpočty makroskopických vlastností premávky cez túto bunku, čo bude popísané v sekcii 4.11.

Trieda *Cell* obsahuje metódy:

- *Free* a *Seize*: uvoľnenie/obsadenie bunky.
- *AddCrossroadDirection*: pridanie orientácie cesty do zoznamu *nextCrossroadDirections*. Táto funkcia sa volá po dokončení konfigurácie mapovania pruhov najbližšej križovatky.
- *ContainsDirection*: metóda vracia príznak true/false, ak zoznam *nextCrossroadDirections* obsahuje daný smer.
- *GetAverageSpeed* a *GetFlow*: vypočíta na bunke priemernú rýchlosť vozidiel a prietok. Využíva pritom zoznamy *CarSpeed* a *timesOfCarCrossing*.
- *RecordCrossingCar*: zaznamená čas prechodu vozidla cez túto bunku a jeho rýchlosť, ktoré sa uložia do príslušných vyššie spomenutých zoznamov.

4.2 Trieda *Car*

Predstavuje vozidlo, ktoré je samostatnou entitou. Má nasledujúce atribúty:

- *Color*: farba, ktorou sa vozidlo vykresľuje na plochu.
- *CurrentSpeed*: aktuálna rýchlosť vozidla.
- *PreviousSpeed*: rýchlosť vozidla v predchádzajúcom simulačnom kroku.
- *Length*: dĺžka vozidla v počte buniek, ktoré obsadzuje.
- *MaxSpeed*: maximálna rýchlosť vozidla.
- *nextCrossroadDirection*: vektor orientácie cesty, ktorú vozidlo použije pri prechádzaní ďalšej križovatky. Ak už nie je pred vozidlom žiadna križovatka, vektor je nastavený ako nulový.
- *id*: identifikačné číslo vozidla.

Hlavnou metódou je *Move*, ktorá sa pre každé vozidlo zavolá počas každého simulačného kroku. Táto metóda obsahuje celý proces pohybu vozidla vrátane prechodu cez križovatku prípadne spomalenia kvôli udeleniu prednosti inému vozidlu alebo zmene pruhu (sekcia 4.12)

4.3 Trieda *Crossroad*

Reprezentuje križovatku ciest. Jej atribúty sú nasledovné:

- *Location*: poloha križovatky (dvojrozmerný bod).
- *directions*: zoznam ciest pripojených na križovatku. Je to slovník, ktorého kľúčom je vektor orientácie niektorej najbližšej bunky križovatky, ktorou vozidlá opúšťajú križovatku (v prípade jednosmernej cesty, ktorej smer jazdy je v smere do križovatky, je to opačný vektor niektorej bunky). Tento vektor je identifikátorom pripojenej cesty. Hodnotou je pár krajných buniek (ak je to jednopruhovú cestu, obidve bunky sú referenciou na tú istú bunku).
- *laneMappings*: obsahuje prepojenia vstupných a výstupných pruhov medzi jednotlivými cestami v podobe slovníka. Kľúčom je vektor vstupnej cesty, hodnotou je ďalší slovník s kľúčom ako vektor výstupnej cesty a hodnotou zoznam párov buniek (vstupná-výstupná).
- *probabilities*: obsahuje pravdepodobnosti prechodov z jednotlivých ciest na ostatné cesty pripojené na križovatku. Kľúčom je vektor cesty a hodnotou zoznam ostatných vektorov ciest s hodnotou pravdepodobnosti, ktorým je celé číslo.
- *mainroads*: zoznam hlavných ciest.
- *semaphoreSignals*: definícia svetelných signálov. Je v podobe slovníku, kde kľúčom je identifikátor svetelného signálu a hodnotou zoznam párov vektorov ciest (vstupná-výstupná), ktoré sú asociované k svetelnému signálu.

- *semaphoreIntervals*: definícia intervalov prepínania svetelných signálov, kde kľúčom je identifikátor svetelného signálu a hodnotou zoznam intervalov s príznakom, či je signál červený alebo zelený.
- *semaphoreTimers*: slovník, kde sú v asociácií identifikačné čísla svetelných signálov s referenciami na časovače, ktoré odpočítavajú časy na prepínanie signálov.
- *currentPartOfPeriodOfSignalSwitching*: slovník, ktorý ku každému svetelnému signálu udržiava poradie intervalu, z ktorého sa má určiť aktuálny svetelný signál.

Obsahuje niekoľko podstatných metód:

- *AddRoad*: pridá novú cestu do križovatky doplnením vektoru novej cesty do zoznamu *Directions*. Ak je na križovatku pripojených už 3 alebo viac ciest, vyvolá sa dialógové okno na konfiguráciu križovatky. Ak užívateľ uskutoční konfiguráciu križovatky, prejdú sa všetky pripojené cesty až po iné križovatky alebo ich konce a nastaví sa v bunkách s orientáciou smerom ku križovatke položky *nextCrossroadDirections*, *IsMainRoad* a do *nearestCrossroad* sa nastaví referencia na nakonfigurovanú križovatku.
- *SetCrossroadInfo*: uskutoční nastavenie atribútov buniek týkajúcich sa údajov o najbližšej križovatke. Táto metóda sa volá v metóde *AddRoad* a taktiež sa volá v triede *World* v metóde *ConnectWithOtherRoad*.
- *GetNextDirection*: získa ďalší vektor cesty v smere hodinových ručičiek od vektoru daného v parametri. Táto metóda sa využíva pri určovaní predností v jazde.

Trieda *Crossroad* obsahuje ďalšie metódy, ktoré slúžia na získavanie informácií zo zoznamov obsahujúcich konfiguráciu križovatky.

4.4 Trieda *Generator*

Trieda *Generator* je abstraktná. Obsahuje vlastný časovač, ktorým generuje lístky na umiestnenie nových vozidiel na pripojenú cestu. Jeden lístok umožňuje počas simulačného kroku na začiatok cesty vygenerovať jedno auto. Časovač po každom uplynulom tiku zvýši počet lístkov o 1 a nastaví nový interval budenia, ktorý vypočíta abstraktná metóda *GetNextInterval*.

V simulátore sú vytvorené tri druhy generátorov, ktoré dedia z triedy *Generator*: deterministický, exponenciálny a Gaussov (normálny) generátor.

Deterministický generátor obsahuje atribút *Interval*, ktorý sa nastaví pomocou konštruktora. Telo metódy *GetNextInterval* spočíva len vo vrátení tohto intervalu.

Generovanie v exponenciálnom generátore závisí od atribútu A . Distribučnou funkciou normovaného exponenciálneho rozloženia je $1 - e^{-x/A}$. Metóda *GetNextInterval* vypočíta nový interval pomocou inverznej funkcie k distribučnej funkcii: $t = -A * \log(1.0 - \text{random}())$, kde *random()* vracia náhodné číslo v intervale $< 0, 1$) [8].

Generovanie v Gaussovom (normálnom) generátore závisí od atribútov μ a σ . Metóda *GetNextInterval* využíva Marsagliovu polárnu metódu generovania čísel normálnej distribúcie [11].

4.5 Trieda *World*

Je základnou triedou simulátora. Obsahuje nasledovné atribúty:

- *carsNewPositions* a *carsOldPositions* sú slovníky s pozíciami všetkých vozidiel v simulátore. Hodnotou sú najprednejšie bunky, ktoré vozidlo obsadzuje.
- *carsIdCounter*: čítač, podľa ktorého sa priradujú identifikačné čísla novým vozidlám.
- *carsToRemove*: zoznam vozidiel, ktoré v predchádzajúcom simulačnom kroku opustili systém. Tento zoznam sa udržuje, aby sa vozidlá ešte do ďalšieho simulačného kroku vykresľovali.
- *cells*: zoznam všetkých okrajových buniek ciest v systéme. Tento zoznam je ešte rozdelený podľa pozícií buniek v systéme.
- *generators*: zoznam všetkých generátorov.
- *crossroadLocations*: zoznam všetkých križovatiek.
- *simulationStepTimer* a *statisticsTimer*: hlavný časovač simulácie a časovač zaznamenávania makroskopických štatistík.

Definuje aj nasledovné konštanty:

- *CellLength*: dĺžka bunky.
- *LaneWidth*: šírka pruhu/bunky.
- *SimulationStepTime*: dĺžka simulačného kroku.

V simulátore je nastavená dĺžka bunky ako 0.5 metra a dĺžka simulačného kroku 0.36 sekundy. Prechod cez jednu bunku počas jedného simulačného kroku zodpovedá rýchlosti 5 km/h. Vozidlá môžu obsahovať rýchlosti, ktoré sú násobkom 5 km/h.

Metódami triedy *World* sú:

- *BuildRoad* - zabezpečuje postavenie cesty. Pred jej konštrukciou sa pre oba konce novej cesty kontroluje blízkosť iných koncov ciest alebo križovatiek. Ak sa na jednom z koncov nachádza iný koniec cesty alebo križovatka, nastane spojenie s touto cestou respektíve križovatkou.

Spojenie ciest prebehne prostým posunutím stredu konštruovaného konca cesty na stred konca existujúcej cesty. Pokiaľ spojením ciest vznikne ostrá zákruta, prejaví sa nevýhoda tohto riešenia. Vozidlá pri prechode z jednej cesty na druhú vykonajú úskok, obzvlášť pri širších cestách na krajoch. Na druhej strane je toto riešenie implementačne jednoduché. Rovnaký princíp sa uplatňuje pri spájaní ciest ku križovatke, kedy križovatku tvorí len bod, na ktorý sa pripájajú stredy ciest.

- *CellForwards* a *CellBackwards*: statické metódy, ktoré k danej bunke vrátia jej nasledujúcu alebo predchádzajúcu bunku. Metódy využívajú vzájomné referencie buniek a zohľadňujú pritom aj koniec pruhu a v prípade metódy *CellForwards* aj prechod cez križovatku.
- *ConnectToCrossroad* a *ConnectWithOtherRoad* sú metódami volanými v metóde *BuildRoad* v prípade pripájania cesty na inú cestu alebo križovatku

- *StartTimers*, *StopTimers* slúžia na zapnutie/vypnutie časovačov a tým aj simulácie. *ResetCars* simuláciu reštartuje.
- *SimulateStep*: vykoná jeden simulačný krok

4.6 Trieda *Renderer*

Slúži na vykresľovanie topológie a vozidiel na obrazovku. Obsahuje atribúty:

- *Scale*: úroveň priblíženia v logických obrazových bodoch na meter. Približovaním/odďaľovaním sa tento atribút mení, je naviazaný na grafický element typu *Slider*.
- *viewMidpoint*: poloha stredu zobrazovacieho okna. Na začiatku po spustení programu je to bod [0,0]. Posúvaním zobrazenia sa mení hodnota tohto atribútu.
- *World*: obsahuje referenciu na triedu *World*.
- *renderingTimer*: časovač vykreslenia vozidiel.
- *elementToRenderOn*: referencia na grafický element, na ktorom prebieha vykresľovanie. Využíva sa pre explicitné zneplatnenie jeho obsahu po zmene cestnej topológie alebo po každom tiku časovača pre vykreslenie vozidiel.

Metódy triedy *Renderer* sú:

- *Render*: vykresľuje vozidlá. Po každom tiku časovača dôjde k zneplatneniu obsahu grafického elementu. Element je typu *CanvasWorld*, ktorý definuje vlastnú metódu *OnRender* s volaním metódy *Render*.
- *RedrawTopology*: prekreslí topológiu na plátne
- *RedrawCanvas*: metóda dostupná zvonku triedy, ktorá prekreslí topológiu a zároveň zneplatní obsah plátna
- *DrawRoad*: vykreslenie cesty
- *DrawRoadPreview*: vykreslenie náhľadu cesty. Náhľad sa vykresľuje po prvom kliknutí myši na plátno v móde konštrukcie cesty.
- *ChangeViewMidpoint*: posunutie pohľadu na plátne o vektor daný v parametri

4.7 Činnosť simulátora

Simulátor si uchováva stav automatu v slovníkoch *carsNewPositions* a *carsOldPositions*, ktorých kľúčom je samotná inštancia triedy *Car* a hodnotou je inštancia bunky (trieda *Cell*), ktorú auto obsadzuje ako prvú spredu. Počas simulačného kroku obsahuje jedno z nich predchádzajúci stav, kým do druhého sa ukladá nový stav. V ďalšom kroku sa ich referencie vymenia. Po zahájení simulácie sa aktivuje hlavný časovač simulácie. Časovač po každom tiku zahájí vykonávanie ďalšieho kroku simulácie. Každé auto v topológii zahájí pohyb o určitú vzdialenosť dopredu, podľa toho, akú má aktuálnu rýchlosť. Aktivuje sa aj časovač vykresľovania a všetky časovače generátorov. Časovače sú typu *DispatcherTimer* [1]. Tieto časovače bežia na UI vlákne a pokiaľ je UI zaneprázdnené iným časovačom,

zaradí sa do fronty informácia o obslúžení ďalšieho časovača. Nie je tak potrebné riešiť synchronizáciu dát. Teoreticky môže dojsť k zablokovaniu UI, ak sa fronta nebude stíhať obsluhovať. Pri testovaní menších topológií k tejto situácii ale nedochádzalo.

Činnosť simulátora je znázornená v nasledujúcom pseudokóde:

```
1: pom:=carsOldPositions
2: carsOldPositions:=carsNewPositions
3: carsNewPositions:=pom
4:
5: for all car in carsOutsideTheWorld do
6:     remove car from carsOldPositions
7:     remove car from carsNewPositions
8: end for
9: clear carsToRemove
10:
11: for all generator in generators do
12:     if generator has ticket then
13:         if adding new car on generator's connected cell successful then
14:             pop ticket from generator
15:         end if
16:     end if
17: end for
18:
19: for all [car - cell] in carsOldPositions do
20:     newCell:=car.Move(cell)
21:     carsNewPositions[car]:=newCell
22:     if newCell is null then
23:         add car to carsToRemove
24:     end if
25: end for
```

Na začiatku prebehne výmena referencií asociatívnych polí stavu automatu. Následne prebehne vymazanie aut zo stavu automatu, ktoré v predchádzajúcom simulačnom kroku opustili jednu z končiacich ciest. Ak niektoré z generátorov majú viac ako jeden lístok na vygenerovanie vozidla, prebehne pokus o umiestnenie vozidla na bunku, ktorá je pripojená na generátor. Ak je cesta voľná, odoberie sa jeden lístok z generátora. Nakoniec pre každé vozidlo sa zavolá metóda pre vykonanie jeho pohybu. Návratová hodnota ako bunka novej pozície auta sa nastaví do nového stavu automatu. V prípade, že vozidlo natrafilo na koniec cesty, je návratovou hodnotou odkaz, ktorý nikam neukazuje. V takom prípade sa vozidlo zaradí do zoznamu vozidiel na vymazanie a vozidlo sa odoberie z automatu.

4.8 Implementácia križovatiek

Všetky informácie o križovatke sa nachádzajú v inštancií triedy *Crossroad*. Každá bunka má referenciu na križovátku, ktorá sa nachádza v jej smere. Vozidlo, ktoré vstúpi na cestu z generátora alebo z inej križovatky, využije túto referenciu na získanie smeru, ktorý má zvoliť v tejto križovatke. Tým má auto počas celej cesty vždy len informáciu o smere z najbližšej križovatky. Ak bunka nemá referenciu nastavenú na inštanciu triedy *Crossroad*, nebude nasledovať už žiadna križovatka a vozidlá na tejto ceste opustia systém. Pri približovaní sa

ku križovatke musí vozidlo spomaliť, ak to vyplýva z pravidiel premávky. Vyhodnocuje, či aktuálna cesta vozidla bude označená ako hlavná, či cesta cieľového smeru je hlavná, rešpektíve či na križovatke neplatí pravidlo pravej ruky alebo nie sú aktívne svetelné signály. Pokiaľ sa vozidlo blíži k neoznačenej križovatke, riadi sa týmito pravidlami:

1. Vozidlá na cestách, ktoré sa nachádzajú v oblúku z aktuálnej cesty v smere hodinových ručičiek k cieľovej ceste (vrátane), môže ignorovať. Tieto vozidlá mu v prípade skríženia trasy musia dať prednosť podľa pravidla pravej ruky.
2. Vozidlá na ostatných cestách môže ignorovať iba vtedy, ak ich cieľovou cestou je niektorá z ciest v oblúku z aktuálnej cesty proti smeru hodinových ručičiek k cieľovej ceste. S týmito vozidlami môže prejsť vozidlo cez križovatku súčasne.
3. Ostatné vozidlá majú prednosť, preto prebehne vyhodnotenie, či vozidlo má spomaliť.

V prípade, že dôjde k uviaznutiu (tri alebo viac vozidiel si navzájom dávajú prednosť), náhodne sa zvolí vozidlo, ktoré prejde cez križovatku ako prvé.

Pokiaľ najbližšia križovatka má definované hlavné cesty, blížiaco vozidlo sa riadi týmito pravidlami:

1. Ak vozidlo prichádza po hlavnej ceste a bude pokračovať taktiež po hlavnej ceste, môže všetky iné vozidlá z iných ciest ignorovať.
2. Ak sa druhá hlavná cesta nenachádza v oblúku medzi aktuálnou cestou v smere hodinových ručičiek k cieľovej ceste, vozidlo môže plynule prejsť cez križovatku. Inak sa vozidlo riadi pravidlami o neoznačenej križovatke 2 a 3 pre vozidlá z druhej hlavnej cesty. Týmto pravidlom sa vozidlo riadi vtedy, ak prichádza po hlavnej ceste a cieľová cesta je vedľajšia.
3. Ak vozidlo prichádza z vedľajšej cesty, riadi sa pravidlami pre neoznačenú križovatku. Navyiac ak sa nachádza hlavná cesta v oblúku z aktuálnej cesty v smere hodinových ručičiek k cieľovej ceste, v prípade skríženia trasy s vozidlom z tejto cesty mu musí dať prednosť.

Pokiaľ sa križovatka riadi svetelnými signálmi, platia pre vozidlo nasledovné pravidlá:

1. Ak vozidlo má červený svetelný signál, nemôže prejsť cez križovatku.
2. Ak vozidlo má zelený svetelný signál, vozidlo sa riadi pravidlami pre neoznačenú križovatku. Prednosť podľa týchto pravidiel dáva iba tým vozidlám, ktoré majú taktiež zelený svetelný signál.

4.8.1 Prechádzanie ciest križovatky

Uvedené pravidlá v predchádzajúcej sekcii vyžadujú procedúru, ktorá k danej ceste vracia vektor najbližšej cesty v smere/proti smeru hodinových ručičiek. Táto procedúra sa nachádza v triede *Crossroad* ako metóda *GetNextDirectionToRight*, ktorá vracia vektor najbližšej cesty proti smeru hodinových ručičiek. Najprv sa vytvorí zoznam všetkých ciest, ktoré sa nachádzajú vľavo a vpravo od osi idúcej cez vektor aktuálnej cesty. Využije sa výpočet determinantu vektoru aktuálnej cesty a vektoru inej cesty v križovatke. Ak je determinant menší ako nula, cesta sa nachádza na pravej strane, inak na ľavej strane.

Ak nie je zoznam ciest na pravej strane prázdny, najbližšou cestou v oblúku proti smeru hodinových ručičiek je vektor, ktorého koncový bod sa nachádza najbližšie ku koncovému bodu vektoru aktuálnej cesty. Ak sa cesty nachádzajú len na ľavej strane, najbližším vektorom je vektor s najvzdialenejším koncovým bodom od koncového bodu aktuálneho vektoru.

4.8.2 Vyhodnotenie pravidiel pre prechod cez križovatku

Funkcia *CheckCrossroadForCollisions* vyhodnotí, či má vozidlo cez križovatku voľný prechod alebo musí dať prednosť inému vozidlu. Riadi sa pravidlami popísanými v sekcii 4.8. Pokiaľ hrozí kolízia s niektorým iným vozidlom, ktorému podľa pravidiel má dať vozidlo prednosť, podľa odhadu časov ich príchodov ku križovatke sa vyhodnotí, či je potrebné spomaliť. Ak platí vzťah:

$$|(n/v - n_i/v_i)| < p/t \quad (4.8.1)$$

(kde n a n_i sú vzdialenosti vozidla a iného vozidla od križovatky, v a v_i ich rýchlosti, t dĺžka simulačného kroku, parameter p je rozdiel časov príchodov v sekundách vozidiel na križovatku, v simulátore nastavený na 5.0), potom hrozí zrážka a vozidlo musí spomaliť. Parameter p sa môže voliť aj náhodne na odlíšenie rôznej agresivity vodičov.

4.9 Spôsob ukladania topológie do pamäte

Bunky ciest sú medzi sebou pripojené prostredníctvom referencií na bunku vpredu, vzadu, vľavo a vpravo. Po vybudovaní cesty v stave konštrukcie sa vytvorí požadovaný počet buniek. Okrajové bunky (ako pár, bunka najviac vpravo a bunka najviac vľavo) sa ukladajú v triede *World* do zoznamov buniek. Zoznamy sú uložené v asociatívnom poli, pričom kľúčom je bod (typ *Point*). Súradnice bodov použitých ako kľúče sú vždy deliteľné číslom 100 a tak každý prvok asociatívneho poľa obsahuje bunky z oblasti veľkej 100 x 100 metrov. Takéto uloženie je výhodné pri vykresľovaní topológie, kedy sa k vykresleniu pripravujú len tie bunky, ktoré sú v zornom poli plátna.

Generátory a križovatky sa ukladajú do zoznamu, keďže ich počet je výrazne menší ako počet buniek. Do zoznamu sa ukladajú aj páry buniek na krajoch ciest. Pri konštrukcii nových ciest sa kontrolujú zoznamy križovatiek a okrajových buniek a ak sú v blízkosti k jednému z krajov konštruovanej cesty/križovatky, cesta sa k nim pripojí.

4.10 Vykresľovanie

O vykresľovanie cestnej topológie a vozidiel v nej sa stará trieda *Renderer*, ktorá obsahuje referenciu na triedu *World*. Hlavnou časťou okna aplikácie je plátno, v ktorom sa graficky znázorňuje simulácia. Aplikácia sa môže nachádzať v jednom z dvoch stavov. Buď je v stave konštrukcie cestnej topológie alebo je v stave simulácie. V stave konštrukcie topológie sa budujú cesty, križovatky a generátory a definujú sa pre ne parametre. V stave simulácie je budovanie topológie zablokované. V momente zahájenia simulácie sa spúšťa hlavný časovač simulácie, časovače generátorov a časovač vykresľovania.

Počas stavu konštrukcie topológie sa pri jej každej zmene prekreslí obsah. Podľa toho, aká úroveň priblíženia je zvolená, a ako je pohľad na topológiu posunutý, sa vykreslí tá časť topológie, ktorá sa nachádza v zornom poli plátna. Vykreslená časť topológie sa nastaví ako pozadie pre plátno. Počas zmeny úrovne priblíženia alebo po posunutí sa celá topológia

prekreslí. Počas simulácie sa vykresľujú iba vozidlá v podobe obdĺžnikov. Trieda *Renderer* obsahuje referenciu na plátno. Po každom tiku časovača sa označí plocha plátna za neplatnú. Počas jej prekreslenia použije triedu *Renderer* na prekreslenie vozidiel. Podobne ako pri prekreslení topológie sa aj pri kreslení áut zohľadňuje úroveň priblíženia a posunutie pohľadu.

Časovač vykresľovania je pre príjemnejší vizuálny efekt simulácie nastavený na nižší interval ako hlavný časovač simulátora. Je známa poloha každého vozidla z predchádzajúceho a aktuálneho simulačného kroku zo slovníkov *carsNewPositions* a *carOldPositions*. Poloha vozidla sa pri každom tiku časovača vykresľovania interpoluje podľa uplynulého času od posledného tiku hlavného časovača simulátora. Vozidlá sa preto po plátne pohybujú plynule.

4.10.1 Prevod z interných súradníc na súradnice plátna a naopak

Aplikácia umožňuje približovanie/oddďalovanie a posúvanie pohľadu. Z tohto dôvodu sa počas vykresľovania transformujú interné súradnice na súradnice plátna. Trieda *Renderer* si udržiava informáciu o úrovni priblíženia a posunutí pohľadu. Úroveň priblíženia má uloženú v jednotke zobrazovacích bodov na meter. Posunutie pohľadu je vyjadrené v bode interných súradníc, ktorý sa nachádza v strede plátna. Výpočet interných súradníc na súradnice plátna sa riadi týmto vzťahom:

$$[x_p, y_p] = ([x_i, y_i] - [x_s, y_s]) * z + [X_p/2, Y_p/2], \quad (4.10.1)$$

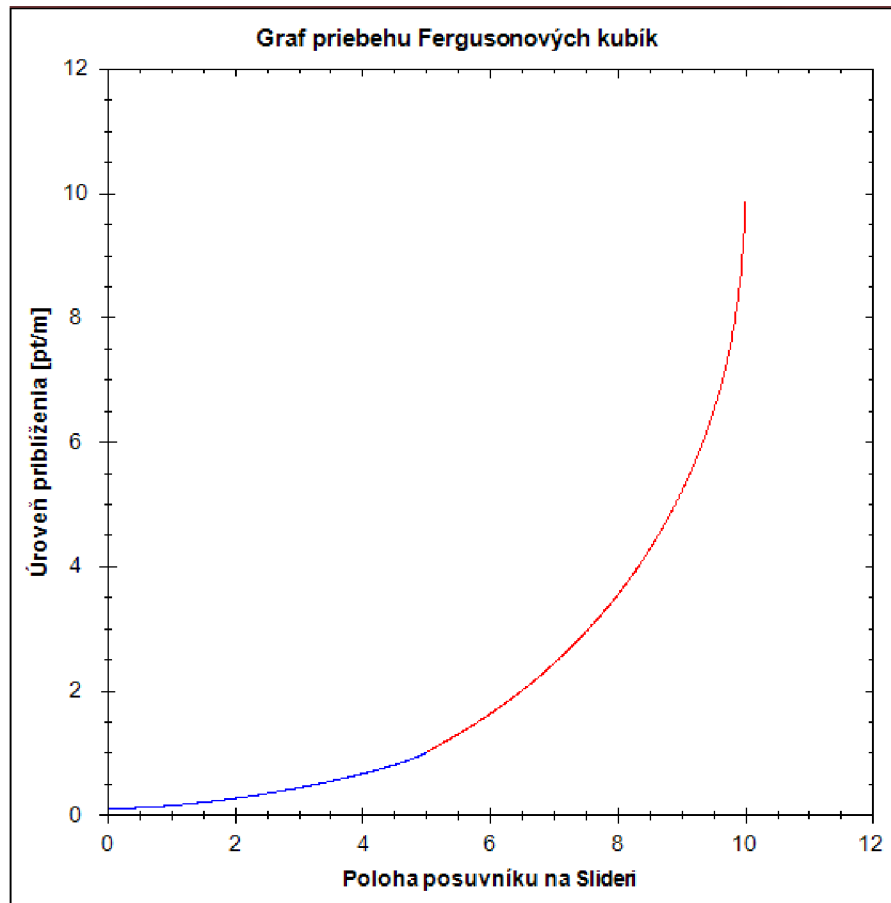
vzťah na opačný výpočet zo súradníc plátna na interné súradnice je nasledovný:

$$[x_i, y_i] = ([x_p, y_p] - [X_p/2, Y_p/2])/z + [x_s, y_s], \quad (4.10.2)$$

kde $[x_i, y_i]$ je bod v interných súradniciach, $[x_p, y_p]$ je bod v súradniciach plátna, $[x_s, y_s]$ je posunutie pohľadu, z je úroveň priblíženia, $[X_p, Y_p]$ je veľkosť plátna.

4.10.2 Zoom

Zoom sa nastavuje pomocou grafického užívateľského elementu typu *Slider* [1], ktorý umožňuje pomocou posuvníku nastaviť číselnú hodnotu v rozmedzí hodnôt 0-10. Je naviazaný na atribút triedy *Renderer Scale*, ktorého hodnota je v jednotke zobrazovacieho bodu na meter (pt/m). Maximálne priblíženie je 10 pt/m, minimálne je 0.1 pt/m, stredné priblíženie 1 pt/m. Pre realistické približovanie/oddďalovanie je potrebné zaviesť nelineárne mapovanie medzi nastavenou hodnotou z grafického elementu typu *Slider* na hodnotu priblíženia úrovne *Scale*. Na tento účel sa využili dve Fergusonové kubiky [5], jedna pre prvú polovicu rozsahu (0.1 - 1 pt/m), druhá pre druhú polovicu (1 pt/m - 10 pt/m). Prvá kubika začína v bode [0; 0,1] s riadiacim vektorom [6;0] a končí v bode [5; 1] s riadiacim vektorom [1,2; 0,8]. Druhá kubika začína v rovnakom bode, kde končí prvá kubika a takisto má aj rovnaký riadiaci vektor pre zachovanie spojitosti C1. Končí v bode [10, 10] s riadiacim vektorom [0, 20]. Na nasledovnom obrázku je graf priebehu týchto kriviek.



Obrázek 4.2: Priebeh Fergusonových kubík.

Prevod hodnoty posuvníku na Slideri podľa priebehu Fergusonových kubík zabezpečuje trieda *ZoomConverter* s využitím Hermitových polynómov [5] metódou rozdeľovania intervalov na polovice, pričom výsledná hodnota je prevedená s presnosťou 0.001.

4.11 Generovanie makroskopických štatistík

Pre každú bunku sa priebežne zbierajú štatistiky o počte prejdených vozidiel a ich rýchlosti, ktorou bunku prešli. Pomocou týchto údajov je možné vypočítať údaje makroskopických štatistík parametrami prietok, hustota a priemerná rýchlosť. Simulátor počíta prietok v jednotke počtu vozidiel za minútu, hustotu v jednotke počtu vozidiel za kilometer a priemernú rýchlosť v jednotke kilometrov za hodinu. V každej bunke sa preto ukladajú časy a rýchlosti všetkých prejdených vozidiel. Pre úsporu v obsadzovaní miesta v pamäti prebehne pri každom pridávaní nového záznamu kontrola, či neobsahuje záznamy o vozidlách staršie ako jedna minúta. Také sú zo zoznamov vymazané. Metóda triedy *Cell GetFlow* vráti počet áut za poslednú minútu, metóda *GetSpeed* vráti ich priemernú rýchlosť. Na výpočet hustoty je potrebné vybrať až dve bunky a spočítať počet vozidiel medzi nimi. Pomocou vzťahu 2.1.1 sa takisto dá dopočítať z vypočítaných hodnôt prietoku a priemernej rýchlosti.

Aplikácia počas simulácie držaním kurzoru myši nad cestou po chvíli zobrazí tieto štatistické hodnoty v uvedených jednotkách ako vyskakovacie okno.

4.12 Algoritmus uskutočnenia kroku auta

Pri každom tiku hlavného časovača simulácie sa vykoná simulačný krok. Pre všetky vozidlá sa zavolá funkcia *Move*, ktorou prejde vozidlo ďalšie bunky automatu. Algoritmus pre pohyb auta je nasledovný:

- Uloží sa súčasná rýchlosť vozidla do atribútu *PreviousSpeed*. Hodnota predchádzajúcej rýchlosti vozidla používajú vozidlá za týmto vozidlom na určenie intenzity akcelerácie.
- Ak vozidlo neobsahuje informáciu o trase cez ďalšiu križovatku (po nasadení na cestu z generátora alebo opustení križovatky), použije referenciu objektu triedy *Crossroad* (*CrossroadRef*), ktorý vozidlu vygeneruje informáciu o trase cez najbližšiu križovatku. Táto informácia je v podobe vektoru smeru cieľovej cesty z bodu križovatky. Ak je referencia nulová, nebude nasledovať už žiadna križovatka a vektor smeru sa nastaví ako nulový vektor.
- Začne sa prechádzanie buniek pred vozidlom. Prechádza sa $CurrentSpeed * 5 + 5$ buniek, ktorými vozidlo získa prehľad o dopravnej situácii pred sebou. Môže byť predčasne ukončené, ak sa narazí na koniec cesty alebo na iné nasledujúce vozidlo. Pre každú bunku, ktorá sa skontroluje pri prehľadávaní, sa použije nasledovný algoritmus vyhodnotenia:
 - Ak je bunka súčasťou viacpruhovej cesty, uloží sa informácia, či pruh zľava sa nezlučuje s pruhom bunky a či sa na ňom nenachádza vozidlo idúce s rýchlosťou nižšou alebo rovnou 2 buniek za simulačný krok. Táto informácia sa môže neskôr použiť, aby vozidlo prípadne spomalilo a podľa pravidla zipsu dala prednosť vozidlu na ľavom pruhu.
 - Ak má bunka iný smerový vektor ako predchádzajúca bunka (zákruta, odbočka križovatky), získa sa informácia o maximálnej rýchlosti cez túto zákrutu. V simulátore sa táto rýchlosť určuje podľa tabuľky:

Uhol zákruty/odbočenia do [°]	Maximálna rýchlosť [km/h]
10	-
25	70
50	40
75	30
100	20
125	15
150	10
180	5
 - Ak vozidlo má pred sebou križovatku, do ktorej vstupuje viacero pruhov, prebehne kontrola, či vozidlo používa správny pruh. Kontrola prebehne na aktuálnej bunke. Použije pritom zoznam vektorov smerov ciest, ktoré sú dosiahnuteľné z aktuálnej bunky. Ak nie je požadovaná cesta v zozname, prejdú sa priečnym prechádzaním bunky cesty. Uloží sa informácia, akým smerom má vozidlo zmeniť pruh a koľko pruhov má daným smerom zmeniť. Pre každý pruh, ktorý má vozidlo prejsť na/prekrižiť sa skontrolujú možnosti kolízie s inými vozidlami. Ak

ide vozidlo rýchlejšie ako 1 bunku za simulačný krok a vozidlo v inom pruhu prechádza hneď vedľa alebo platí vzťah:

$$n/v_s < 2 + (v_s/4) - v \quad (4.12.1)$$

(n je vzdialenosť vozidla v inom pruhu od bunky vedľa prednej bunky aktuálneho vozidla, v_s rýchlosť vozidla na boku, v rýchlosť vozidla), vozidlo nemôže zmeniť pruh, inak pruh zmeniť môže. Táto informácia sa uloží a použije neskôr buď na zníženie rýchlosti vozidla alebo na jeho prechod na bunku vľavo/vpravo. Ak ide vozidlo rýchlosťou 1 bunky za simulačný krok alebo stojí, vozidlo zmení pruh s pravdepodobnosťou p_1 , v simulátore $p_1 = 0.4$, respektíve uloží sa informácia, že vozidlo môže zmeniť pruh, inak pruh zmeniť nemôže.

- Ak bunka je poslednou v pruhu, vozidlo sa nachádza v pruhu, ktorý čoskoro končí. Vyhodnotenie o možnosti zmeny pruhu prebieha rovnako, ako v predchádzajúcom bode.
- Ak bunka je poslednou pred križovatkou, uloží sa informácia o tom, že vozidlo sa blíži ku križovatke.
- Ak sa vozidlo blíži k fronte stojacich vozidiel a samo má rýchlosť vyššiu ako dve bunky za simulačný krok, vyhodnotí, či nemôže použiť iný pruh s kratšou frontou. Toto pravidlo je užitočné pri príjazde auta ku križovatke, kedy v prípade možnosti použitia viacerých pruhov pre dosiahnutie cieľovej cesty použije ten, ktorý má najkratšiu frontu. Zmenu pruhu uskutoční iba vtedy, ak nehrozí kolízia s iným vozidlom. Inak sa vozidlo tohto úmyslu pre tento simulačný krok vzdá a pokračuje ďalej po rovnakom pruhu.
- Ak je vozidlo k inému vozidlu vpredu vzdialené menej ako $2 * CurrentSpeed$, spomalí podľa vzťahu

$$CurrentSpeed = (n + acc)/p_2 \quad (4.12.2)$$

(n je vzdialenosť v počte buniek k vozidlu, acc je rozdiel medzi aktuálnou a predchádzajúcou rýchlosťou, $acc = CurrentSpeed - PreviousSpeed$. Ak $acc > 0$, potom $p_2 = 2$, inak $p_2 = 3$) [7].

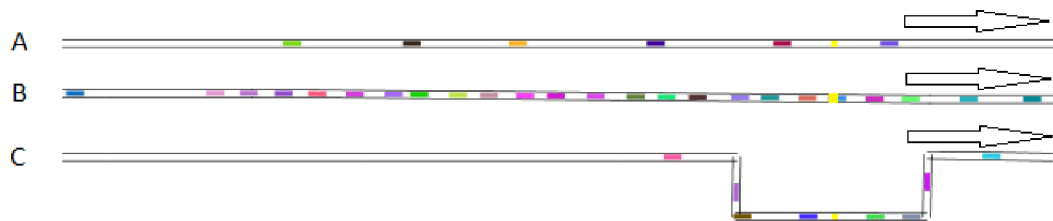
- Ak sa počas prechádzania buniek vyššie ukázalo, že sa vozidlo blíži ku križovatke, zavolá sa funkcia *CheckCrossroadForCollisions* pre detekciu kolízií na križovatke. Postupuje sa podľa pravidiel uvedených v sekcii 4.8.
- Ak vozidlo ide nižšou alebo rovnou rýchlosťou ako 6 buniek za simulačný krok, zvýšenie rýchlosti prebehne každý krok. Inak zvýšenie rýchlosti prebieha len každý druhý simulačný krok. Vozidlo nezrýchľuje, ak musí spomaliť kvôli zákrute, križovatke, kde musí dať prednosť, alebo kvôli nutnej zmene pruhu, ktorú zatiaľ nemôže uskutočniť. Intenzita spomalenia vozidla závisí od vzdialenosti ku križovatke alebo konci pruhu. Vozidlo takisto spomalí s pravdepodobnosťou p_3 vtedy, ak sa nachádza v ľavom pruhu auto s rýchlosťou nižšou alebo rovnou 2 buniek za simulačný krok. V simulátore je p_3 je 0.75.
- Ak má vozidlo voľnú cestu, s pravdepodobnosťou p_4 spomalí ($p_4 = 0.2$) [6].

- Ak je vozidlo v pohybe, jeho súčasné bunky sa pohybom dozadu označia ako voľné. Vozidlo sa posunie o príslušný počet buniek podľa jeho rýchlosti. Ak vozidlo má a môže zmeniť pruh, uskutoční sa aj pohyb doprava/dolava. Opäť pohybom dozadu od novej polohy sa obsadia bunky novej pozície vozidlom. Funkcia *Move* končí a vracia referenciu na novú bunku.

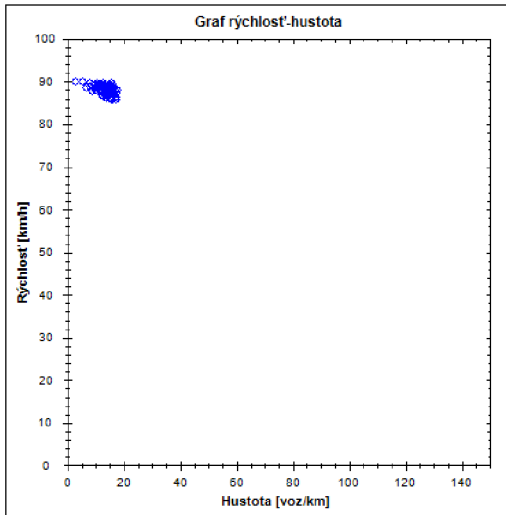
Kapitola 5

Použitie simulátora

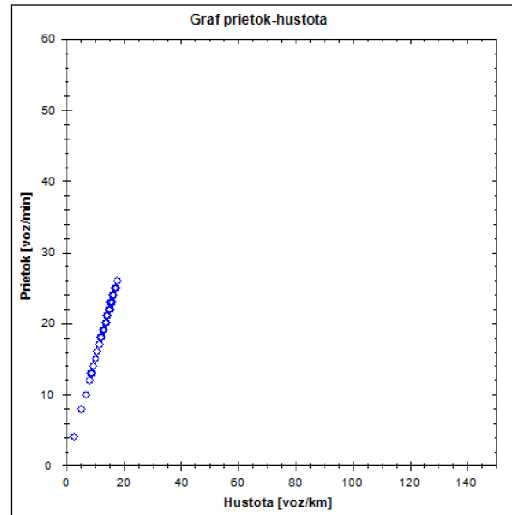
V tejto kapitole sú demonštrované možnosti simulátora v podobe ukážok a grafov makroskopických parametrov rýchlosti, prietoku a hustoty, ktoré sú výstupom simulátora. Tieto grafy sa môžu použiť na validáciu pomocou matematických makroskopických modelov [9]. Prvá ukážka znázorňuje tri jednoduché jednosmerné cesty A, B a C. Cesta A je rovnou cestou bez obmedzení s maximálnou povolenou rýchlosťou 90 km/h. Cesta B má pred jej koncom obmedzenie (napr. v dôsledku práce na ceste) a zníženú povolenú rýchlosť na 30 km/h. Cesta C má pred koncom úseku ostré zákruty. Môžu predstavovať menšiu výluku cesty napríklad vtedy, ak sa na celej šírke cesty robia práce a výluka prechádza cez niektorú bočnú cestu. Na všetky tri cesty sú pripojené exponenciálne generátory s parametrom $A = 3s$



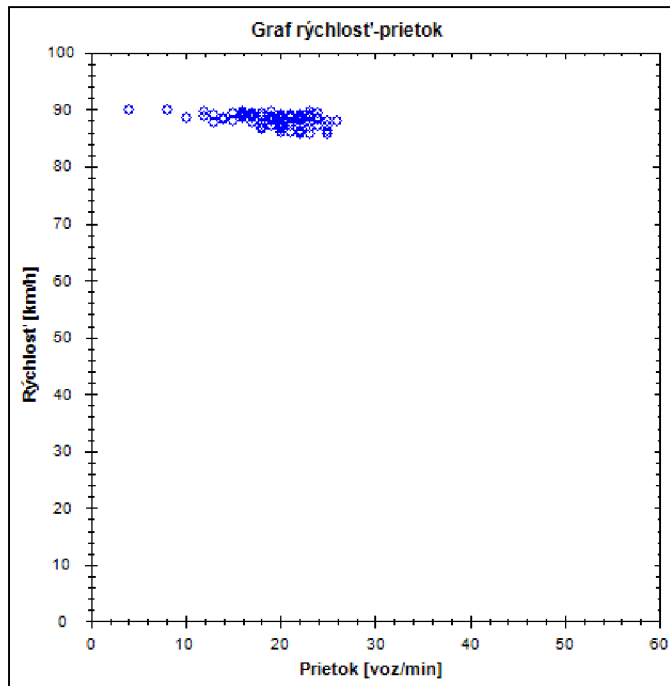
Obrázek 5.1: Jednosmerné cesty.



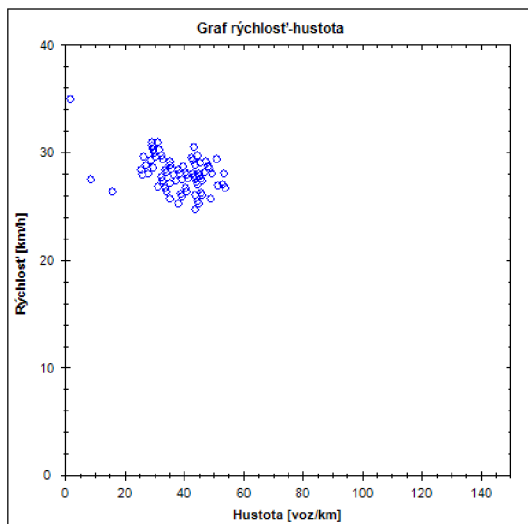
Obrázek 5.3: Graf rýchlost'-hustota na ceste bez obmedzení.



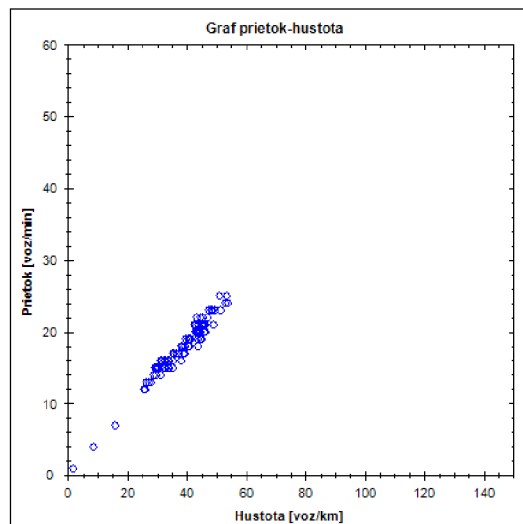
Obrázek 5.4: Graf prietok'-hustota na ceste bez obmedzení.



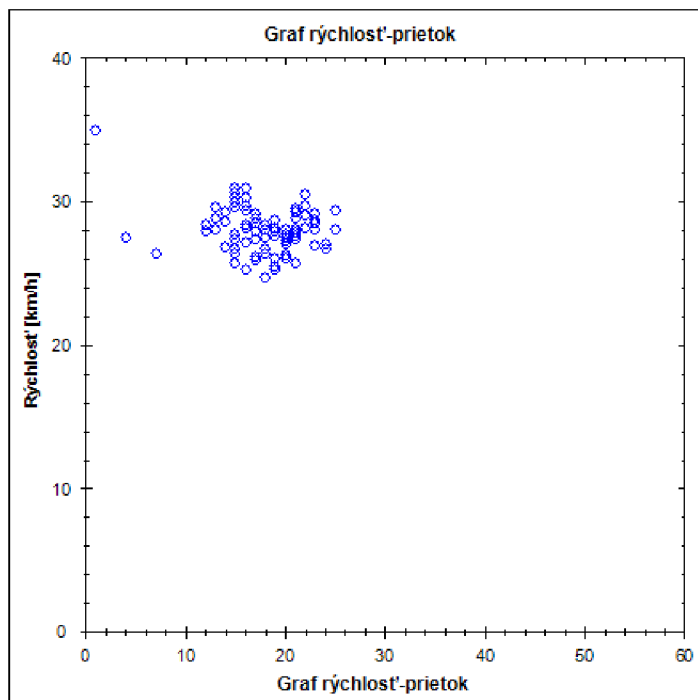
Obrázek 5.2: Graf rýchlost'-prietok na ceste bez obmedzení.



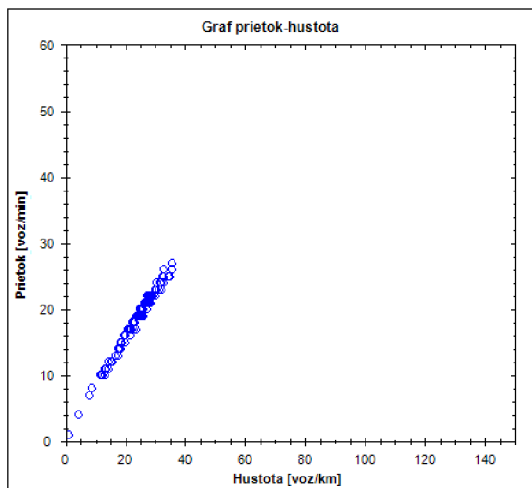
Obrázek 5.6: Graf rýchlosť-hustota na ceste s obmedzením rýchlosti 30 km/h.



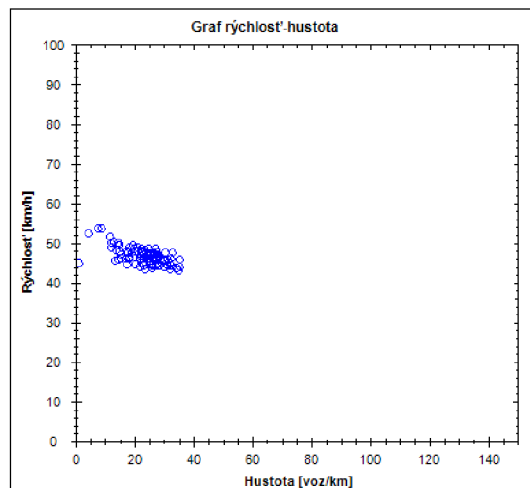
Obrázek 5.7: Graf prietok-hustota na ceste s obmedzením rýchlosti 30 km/h.



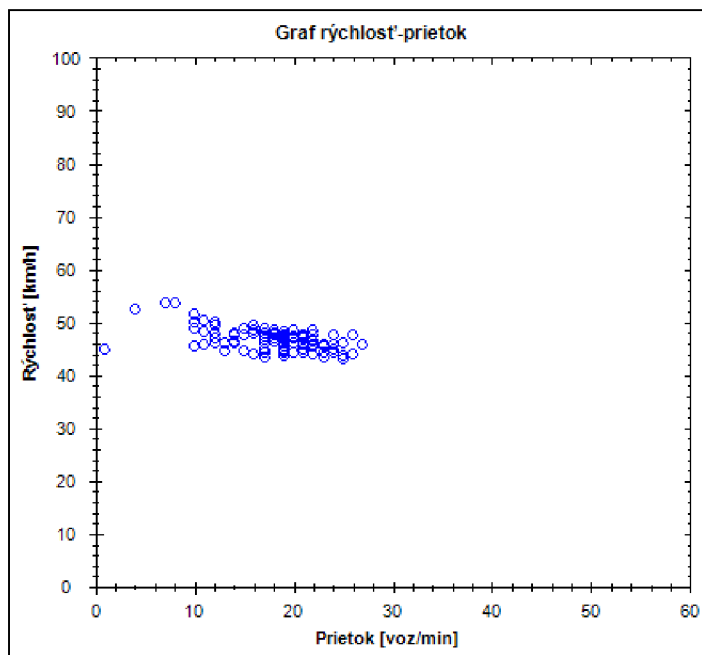
Obrázek 5.5: Graf rýchlosť-prietok na ceste s obmedzením rýchlosti 30 km/h.



Obrázek 5.9: Graf prietok-hustota na ceste so zákrutami. Sledovaná bunka je v oblasti zákrut.



Obrázek 5.10: Graf rýchlosť-hustota na ceste so zákrutami. Sledovaná bunka je v oblasti zákrut.

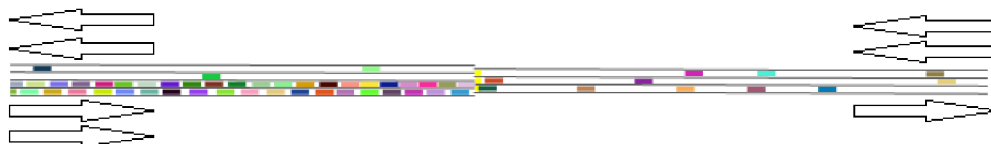


Obrázek 5.8: Graf rýchlosť-prietok na ceste so zákrutami. Sledovaná bunka je v oblasti zákrut.

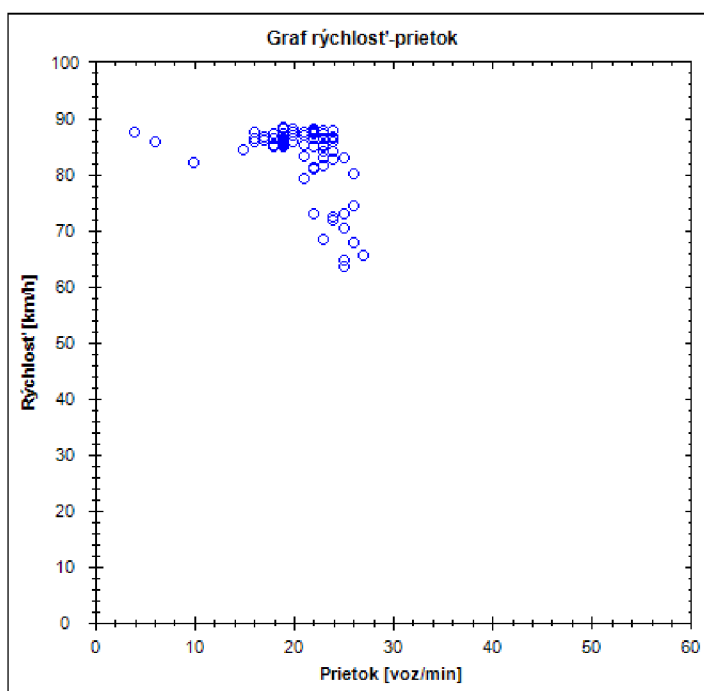
Na grafe závislosti rýchlosti na prietoku je vidieť, že na cestách so zníženou rýchlosťou alebo ostrými zákrutami nie je znížený prietok oproti úseku bez obmedzení. Na všetkých troch cestách sa prietok pohybuje v rozmedzí 10-30 vozidiel za minútu. Priemerná rýchlosť na ceste s výlukou približne v polovici výluky je 50 km/h. Graf závislosti rýchlosti na hustote premávky ukazuje rozdielnosť hustoty premávky na všetkých troch cestách. Najnižšia hustota je na ceste bez obmedzení, kedy nepresahuje hodnotu 20 vozidiel na kilometer. V prípade cesty s výlukou sa hustota blíži niekedy k hodnote 40 vozidiel na kilometer. Najvyššia hustota je na ceste s obmedzením, kde na sledovanej bunke presahuje hodnotu

50 vozidiel na kilometer. Na grafoch závislosti prietoku a hustoty je badateľná ich lineárna závislosť. Pomyselná priamka závislosti je najstrmejšia na grafe cesty bez obmedzenia a najmenej strmá na grafe cesty s obmedzenou rýchlosťou.

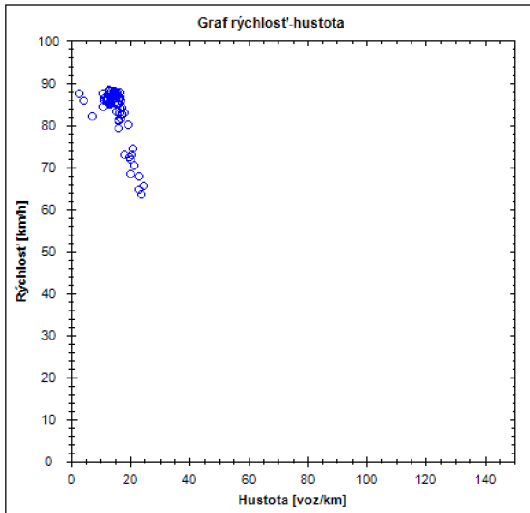
Nasleduje ukážka obojsmernej štvorpruhovej cesty. V jednom smere nie je premávka ničím obmedzovaná. V druhom smere sa v polovici úseku zužuje cesta z dvoch pruhov na jeden napríklad v dôsledku opráv na jednom pruhu alebo prekážke. Na celej ceste je povolená rýchlosť 90 km/h. Na oba konce cesty je pripojený exponenciálny generátor s parametrom $A = 1.5s$. Merania boli uskutočnené na mieste, kde sa cesta zužuje.



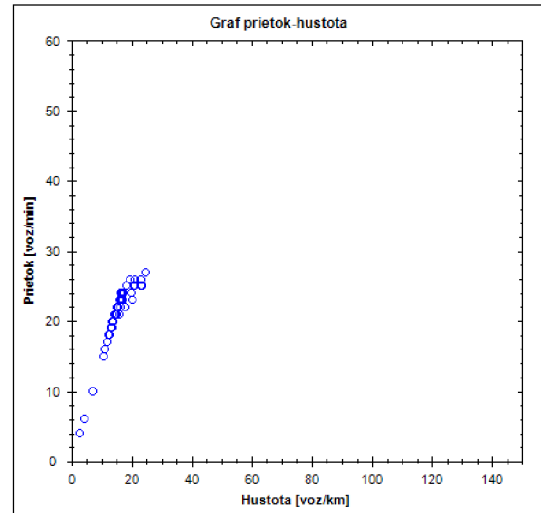
Obrázek 5.11: Štvorpruhová cesta s prekážkou na jednej strane.



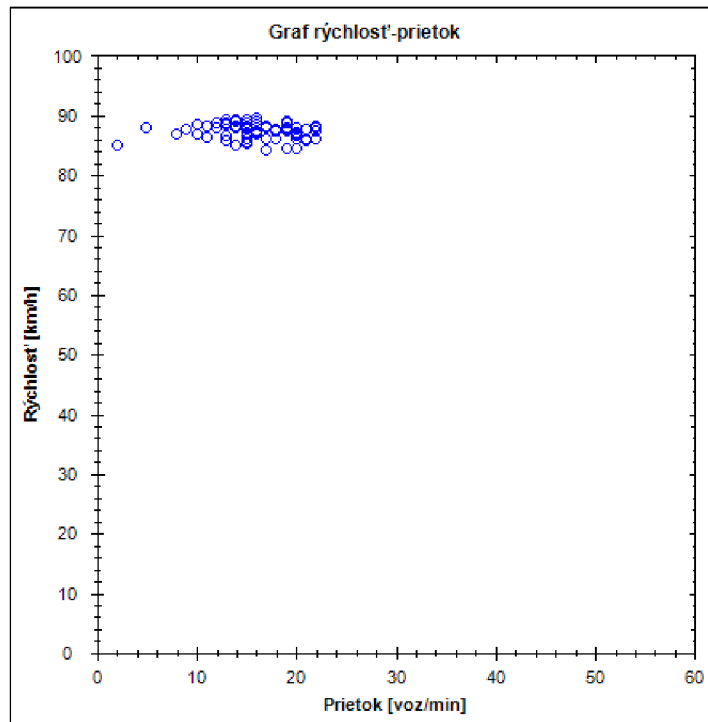
Obrázek 5.12: Graf rýchlosť-prietok na vrchnom pruhu.



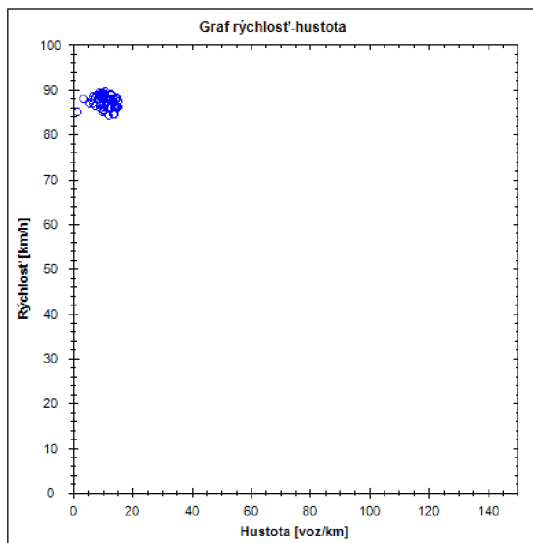
Obrázek 5.13: Graf rýchlosť-hustota na vrchnom pruhu.



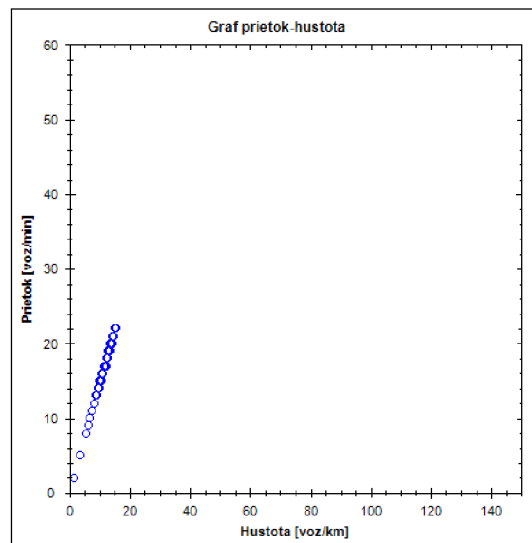
Obrázek 5.14: Graf prietok-hustota na vrchnom pruhu.



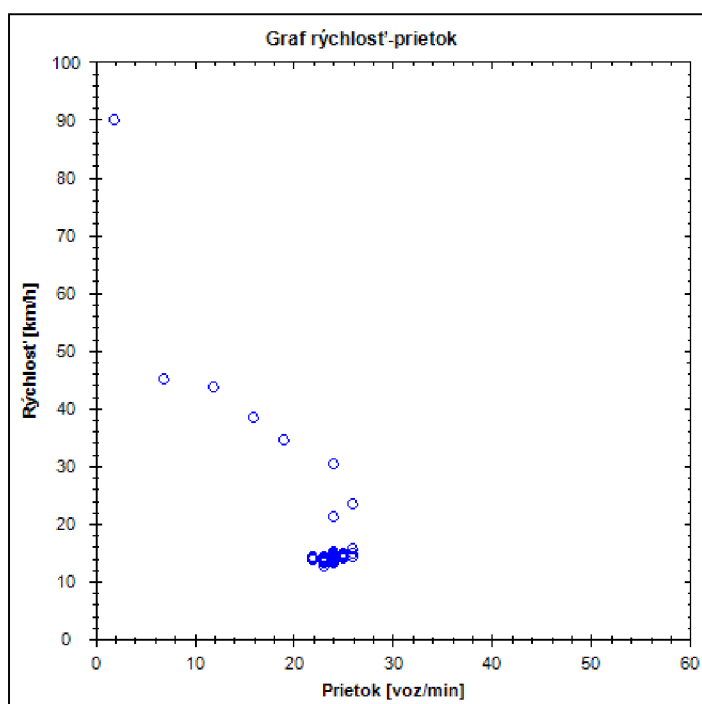
Obrázek 5.15: Graf rýchlosť-prietok na strednom pruhu (v mieste zúženej cesty).



Obrázek 5.16: Graf rýchlosť-hustota na strednom pruhu (v mieste zúženej cesty).

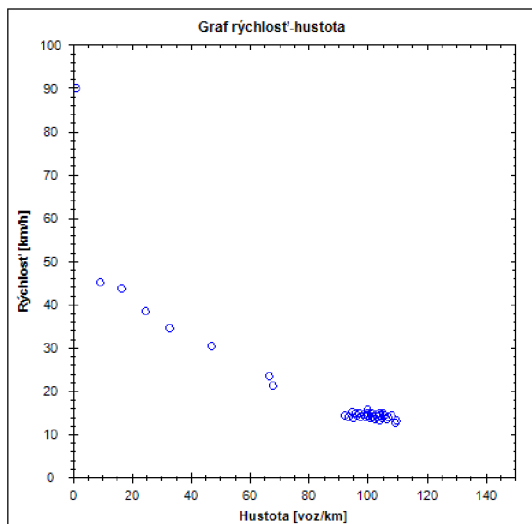


Obrázek 5.17: Graf prietok-hustota na strednom pruhu (v mieste zúženej cesty).

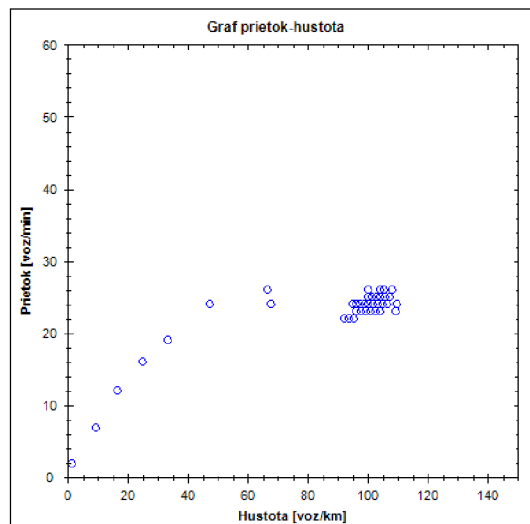


Obrázek 5.18: Graf rýchlosť-prietok na spodnom pruhu (v mieste zúženej cesty).

Na pruhoch neobmedzenej strany cesty sa prietok pohybuje na 15-25 vozidlách za minútu v krajnom a 10-22 v strednom. Spolu tak prietok v tomto smere dosahuje okolo 25-47 vozidiel za minútu, čo zodpovedá nastaveniu pripojeného generátora. V smere, kde sa dva pruhy zlučujú, je prietok na zlúčenom pruhu 20-28 vozidiel za minútu. Tento prietok je prinízky na to, aby obslúžil premávku z generátora. Preto sa vytvorila kolóna. Priemerná rýchlosť na mieste zlúčenia sa ustálila na 15 km/h. Taktiež priemerná rýchlosť v krajnom



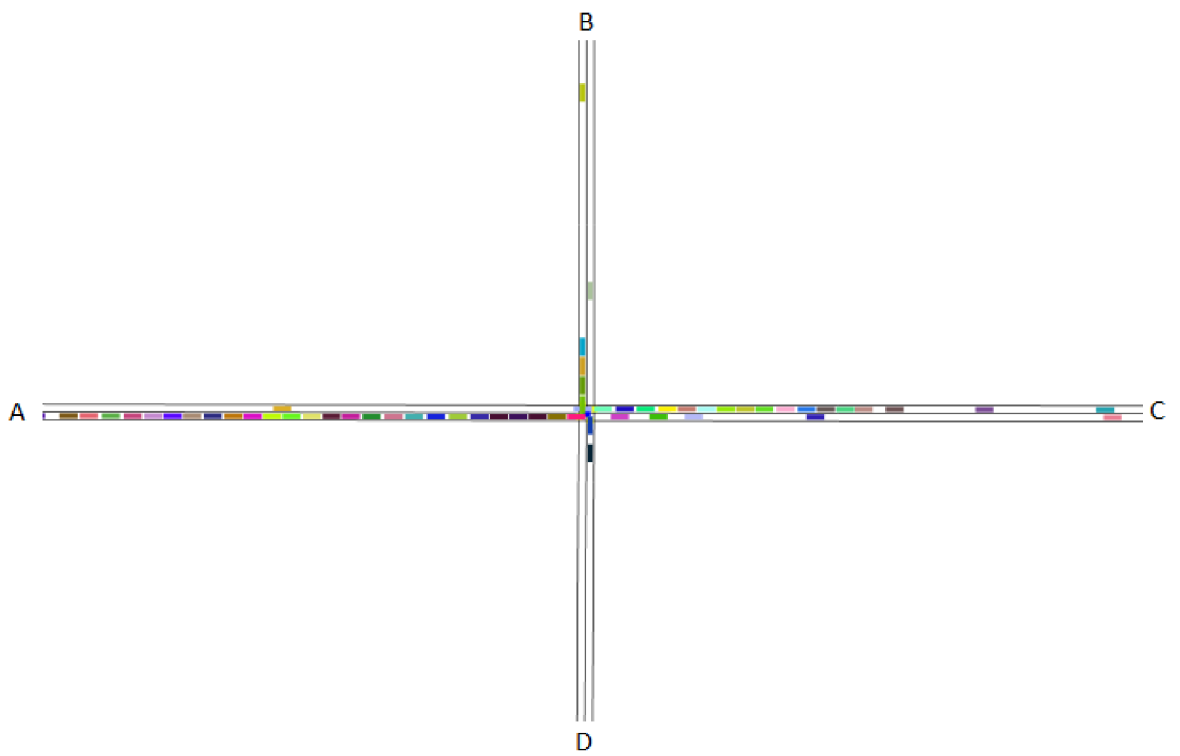
Obrázek 5.19: Graf rýchlosť-hustota na spodnom pruhu (v mieste zúženej cesty).



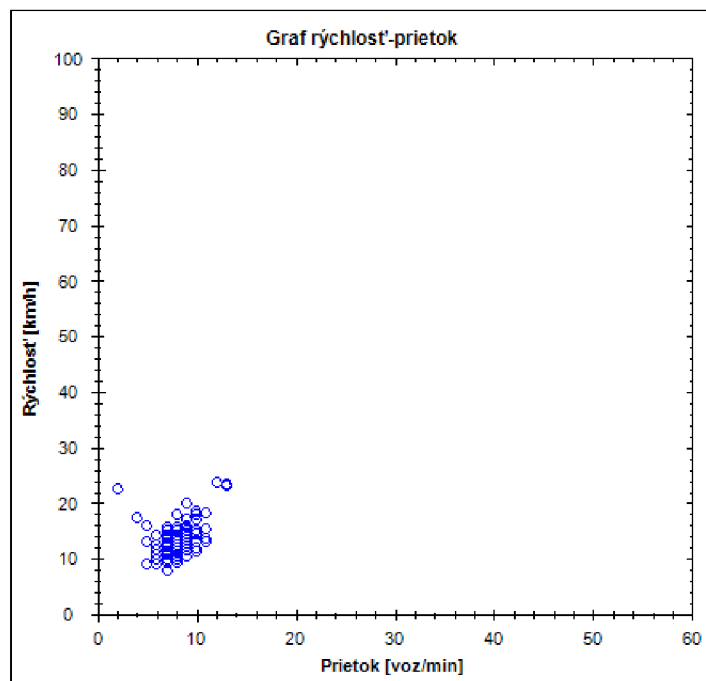
Obrázek 5.20: Graf prietok-hustota na spodnom pruhu (v mieste zúženej cesty).

pruhu opačného smeru dosahovala len 70 km/h, avšak väčšinou sa udržiavala na 90 km/h, teda premávka bola na opačnom smere plynulá. Hustota premávky na dvoch pruhoch neobmedzeného smeru sa držala väčšinou pod 20 vozidlami na kilometer na každom pruhu. Na opačnej strane na zlúčenom pruhu sa ustálila až na 90-110 vozidlách na kilometer. Takáto vysoká hustota naznačuje dopravný kolaps. V grafoch závislosti prietoku od hustoty je opäť badateľná lineárna závislosť až na graf pre zlúčený pruh, kde v dôsledku spomalenia premávky a následného vytvárania sa kolóny tvoria namerané body krivku. Keďže kolóna sa začína tvoriť skoro po začiatku simulácie, najviac nameraných bodov je sústredených na konci krivky.

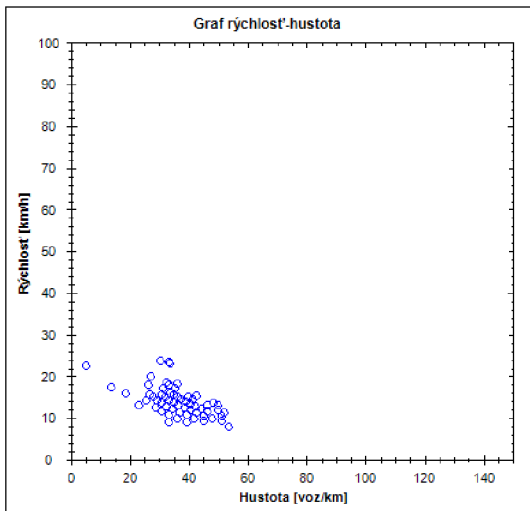
Na ďalšej ukážke sa nachádza križovatka tvaru 'X'. Na cesty A a C sa generujú vozidlá na základe exponenciálneho rozloženia s parametrom $A = 3s$. Na cesty B a D s parametrom $A = 6s$. Križovatka je neoznačená a vozidlá sa riadia pravidlom pravej ruky. Vozidlá idúce po ceste A/C budú s pravdepodobnosťou 60% pokračovať rovno a s pravdepodobnosťami po 20% odbočia doľava alebo doprava. Vozidlá idúce po ceste B/D budú s pravdepodobnosťou 20% pokračovať rovno a s pravdepodobnosťami po 40% odbočia doľava alebo doprava. Cesta A-C môže predstavovať významnú štátnu cestu a cesta B-D lokálnu cestu. Sledujú sa bunky ciest blízko hranice s križovatkou.



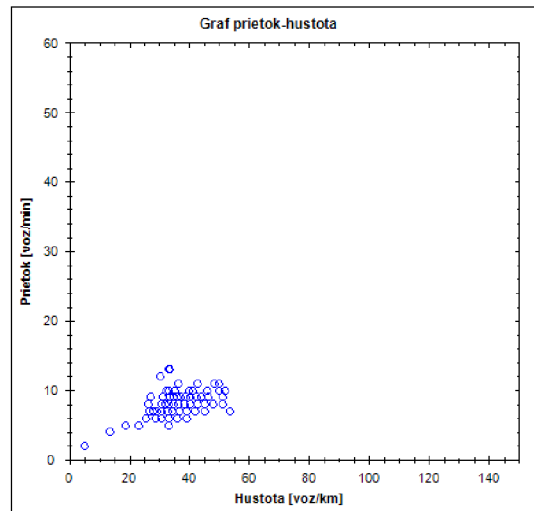
Obrázek 5.21: Neoznačená križovatka.



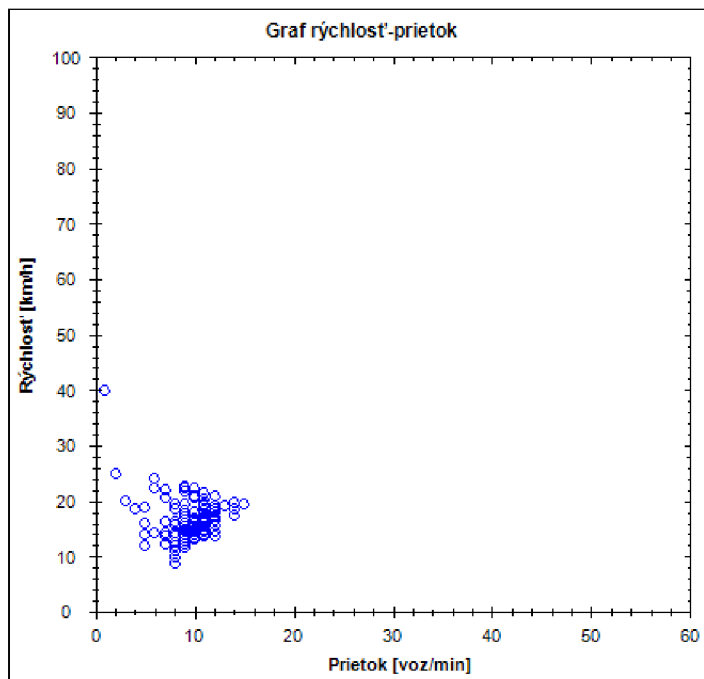
Obrázek 5.22: Graf rýchlosť-prietok na ceste A.



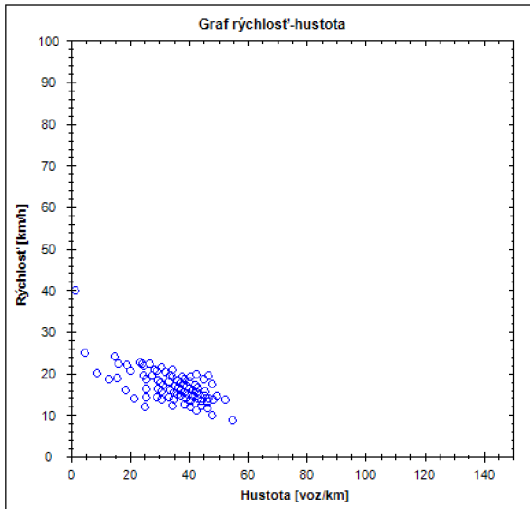
Obrázek 5.23: Graf rýchlosť-hustota na ceste A.



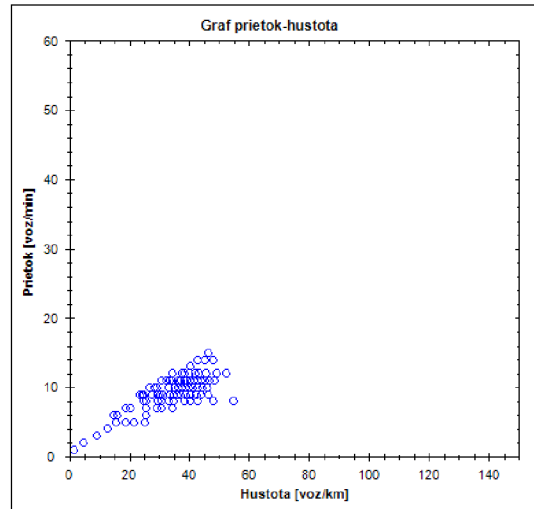
Obrázek 5.24: Graf prietok-hustota na ceste A.



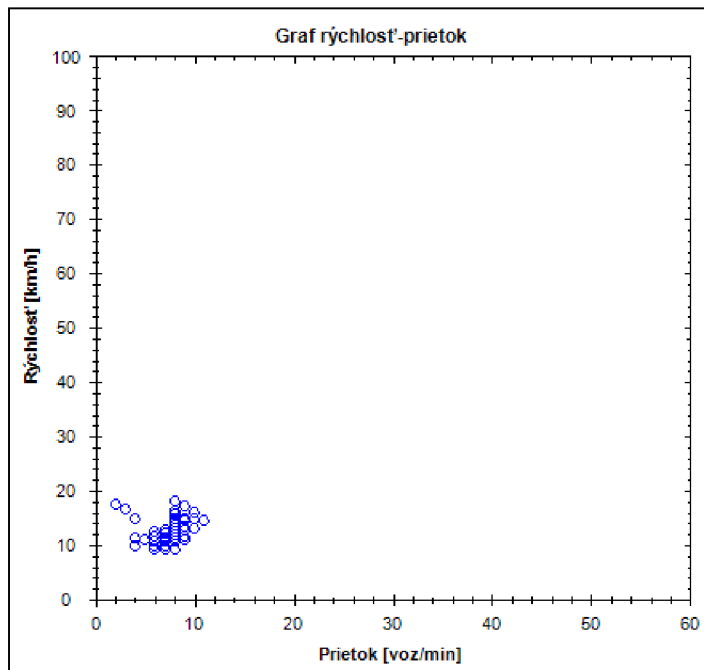
Obrázek 5.25: Graf rýchlosť-prietok na ceste B.



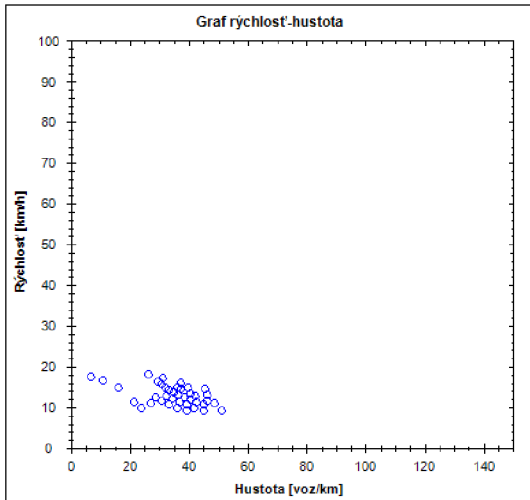
Obrázek 5.26: Graf rýchlosť-hustota na ceste B.



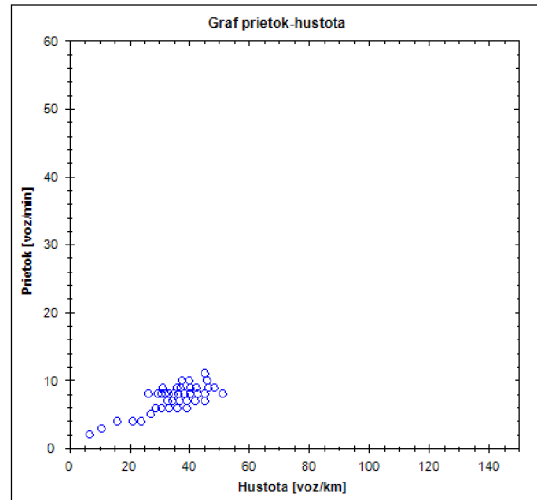
Obrázek 5.27: Graf prietok-hustota na ceste B.



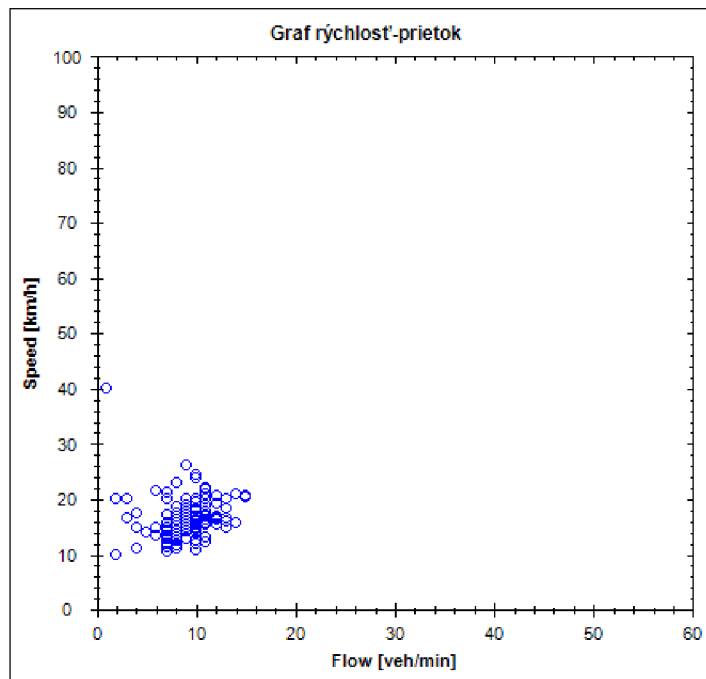
Obrázek 5.28: Graf rýchlosť-prietok na ceste C.



Obrázek 5.29: Graf rýchlosť-hustota na ceste C.



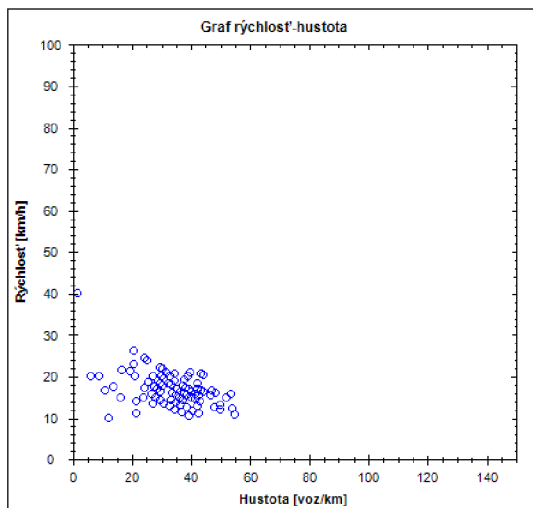
Obrázek 5.30: Graf prietok-hustota na ceste C.



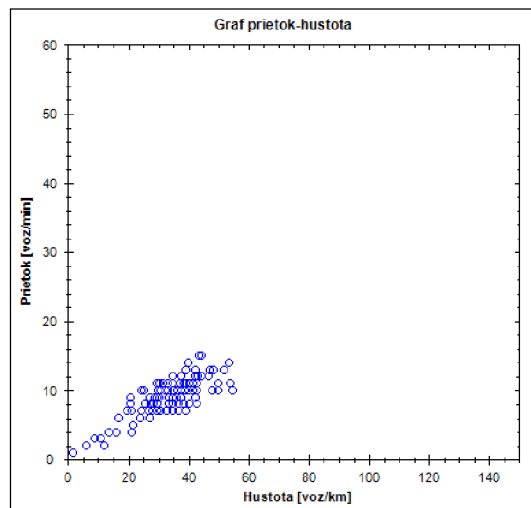
Obrázek 5.31: Graf rýchlosť-prietok na ceste D.

Generátory vytvárajú príliš hustú premávku, aby ju križovatka dokázala obslúžiť. Keďže všetky autá majú v križovatke rovnakú prioritu, vytvoria sa kolóny hlavne na cestách A a C, keďže z nich prichádza viac vozidiel. Priepustnosť je na každej ceste nižšia ako 15 vozidiel za minútu, čo je málo na obsluhu premávky z významnej cesty. Všetky vozidlá prechádzajú križovatkou pomaly, väčšinou nie viac ako 30 km/h.

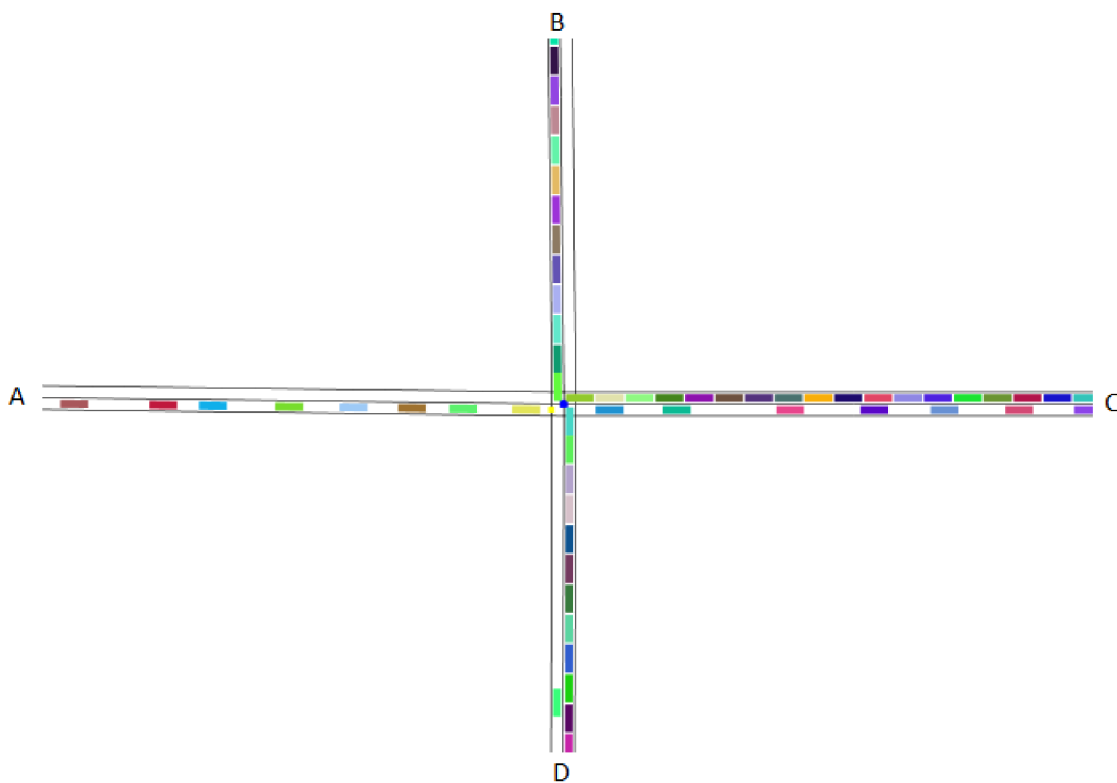
V ďalšej ukážke je opäť tá istá križovatka s rozdielom, že sa významná cesta nastaví ako hlavná.



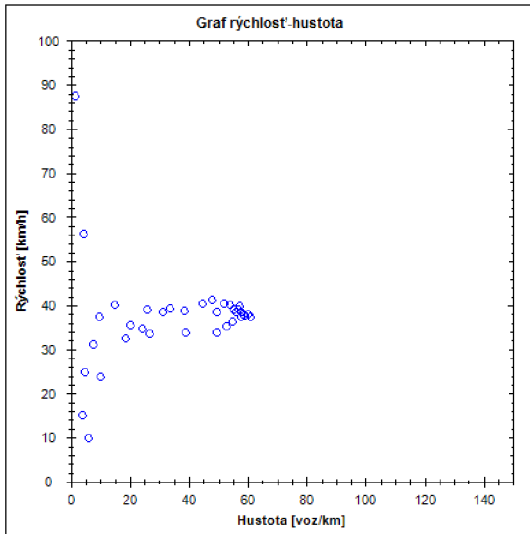
Obrázek 5.32: Graf rýchlosť-hustota na ceste D.



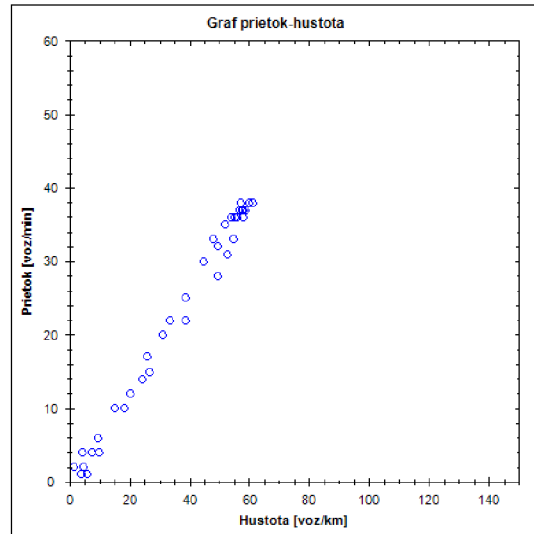
Obrázek 5.33: Graf prietok-hustota na ceste D.



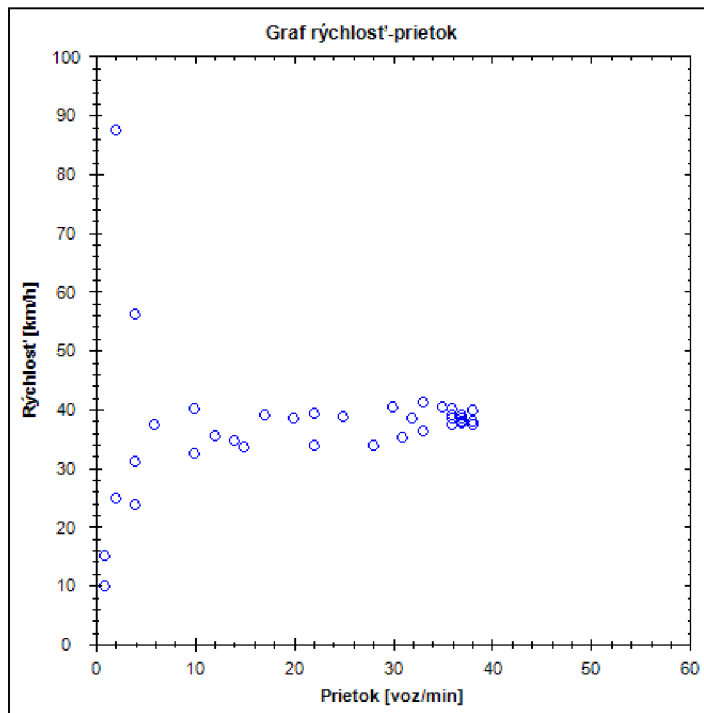
Obrázek 5.34: Križovatka s hlavnou cestou.



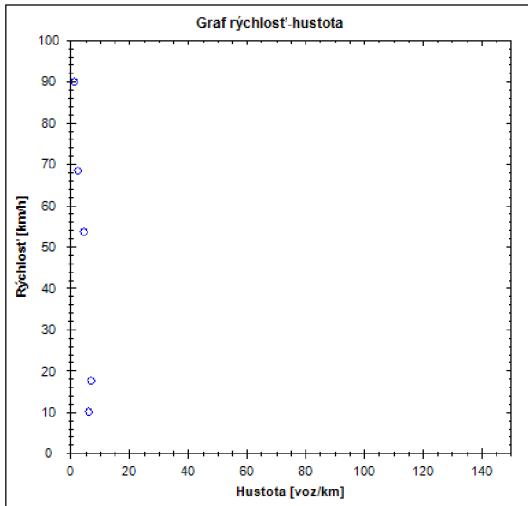
Obrázek 5.36: Graf rýchlosť-hustota na ceste A.



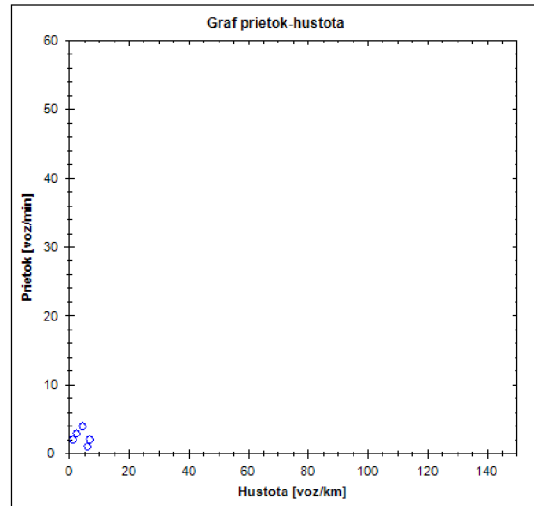
Obrázek 5.37: Graf prietok-hustota na ceste A.



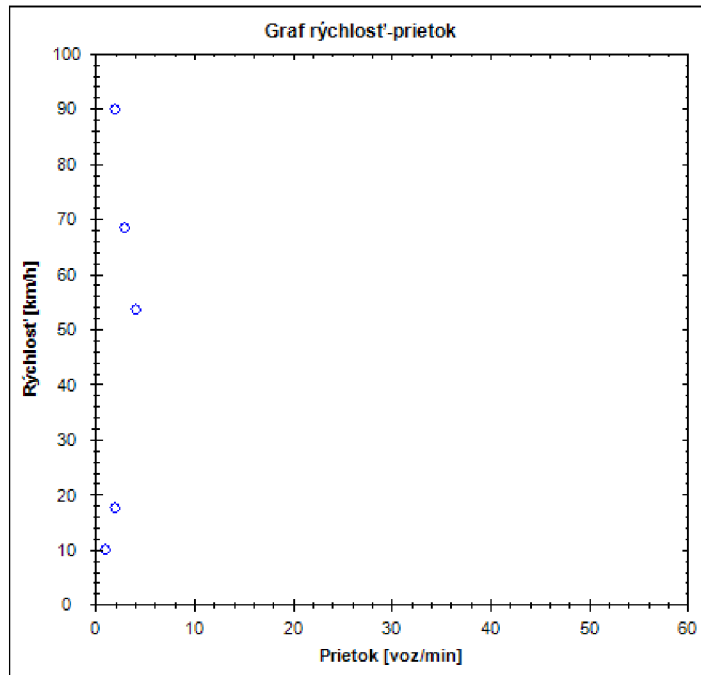
Obrázek 5.35: Graf rýchlosť-prietok na ceste A.



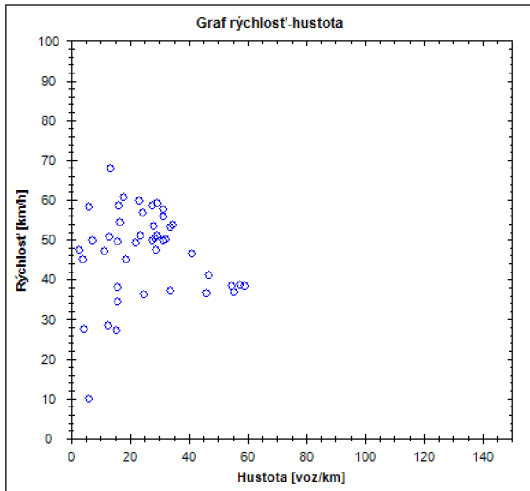
Obrázek 5.39: Graf rýchlosť-hustota na ceste B.



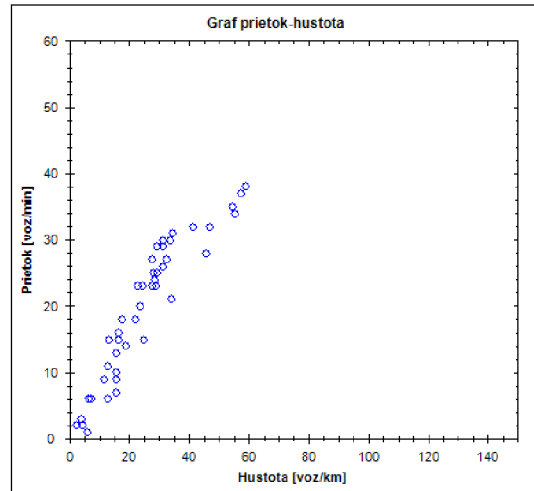
Obrázek 5.40: Graf prietok-hustota na ceste B.



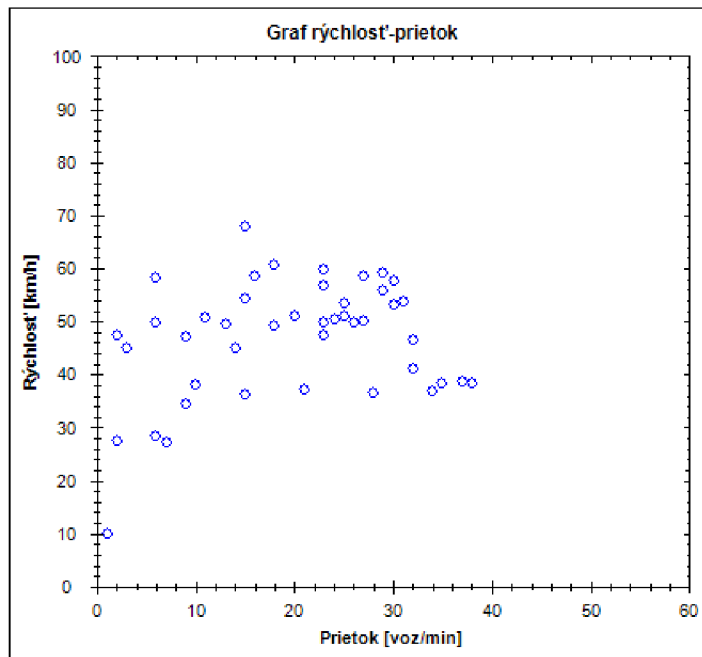
Obrázek 5.38: Graf rýchlosť-prietok na ceste B.



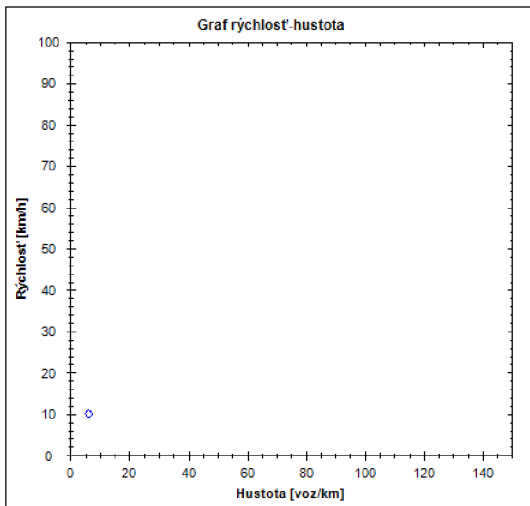
Obrázek 5.42: Graf rýchlosť-hustota na ceste C.



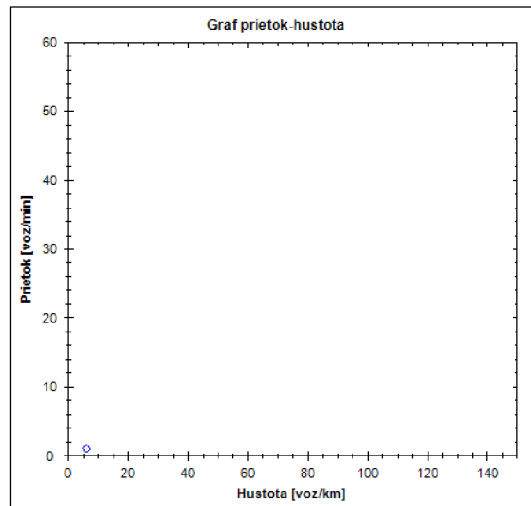
Obrázek 5.43: Graf prietok-hustota na ceste C.



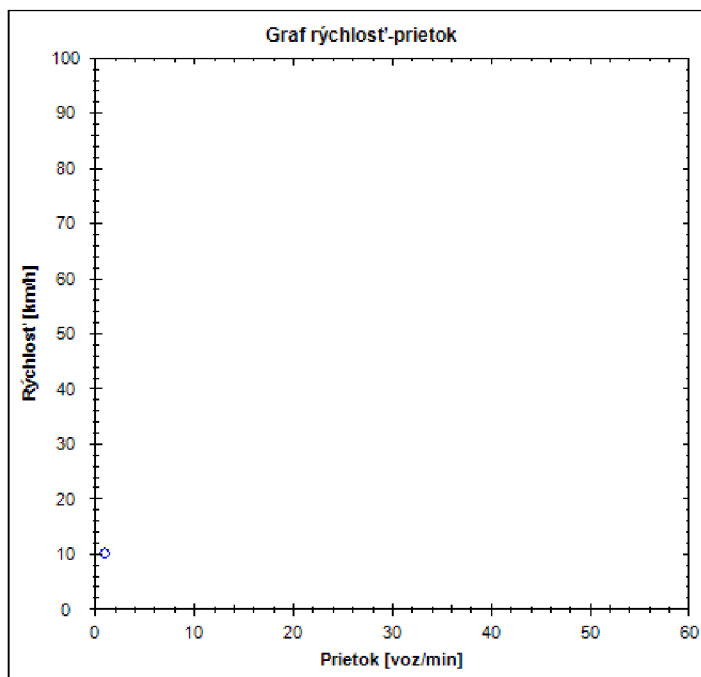
Obrázek 5.41: Graf rýchlosť-prietok na ceste C.



Obrázek 5.45: Graf rýchlosť-hustota na ceste D.



Obrázek 5.46: Graf prietok-hustota na ceste D.

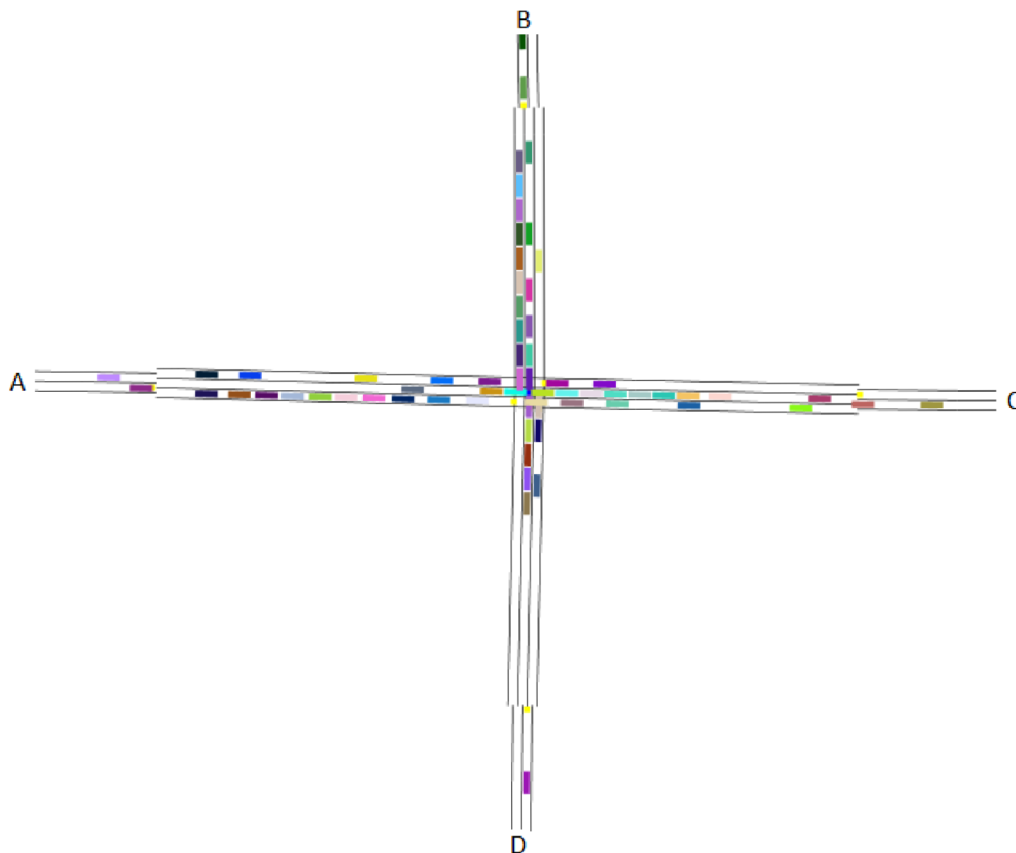


Obrázek 5.44: Graf rýchlosť-prietok na ceste D.

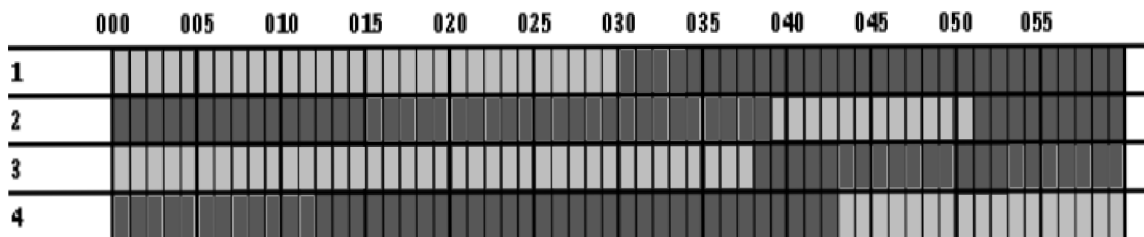
Na križovatke sa opäť vytvorili kolóny. Premávku spomaľovali predovšetkým vozidlá, ktoré odbočovali z hlavnej cesty vľavo. Museli často čakať dlhšiu dobu na chvíľu, kedy sa medzi protiidúcimi vozidlami vytvorila medzera. Týmto aj blokovali vozidlá z vedľajších ciest, ktoré nemali možnosť na prechod cez križovatku. Preto sú aj grafy pre vedľajšie cesty prázdne, keďže sa cez križovatku dostalo minimálne množstvo vozidiel z hlavnej cesty, a to len v krátkom čase počas začiatku simulácie. Namerané body v grafoch pre hlavné cesty sú oveľa viac rozptýlené ako v grafoch v prvých dvoch ukážkach, čo spôsobili hlavne odbočujúce vozidlá.

Činnosť semaforov je demonštrovaná na ďalšej ukážke. Nasadia sa na križovatku z pred-

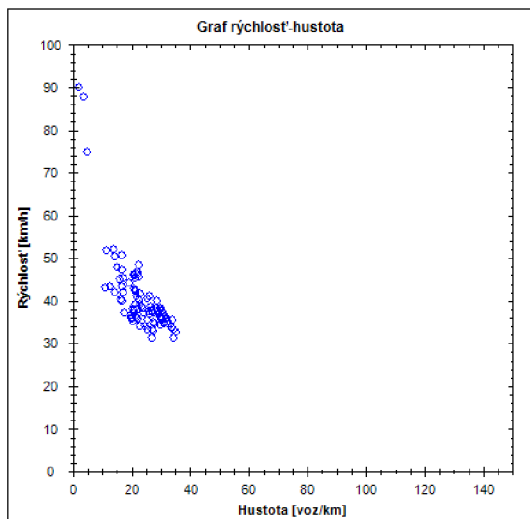
chádzajúcich ukážok. Nastavia sa tak, aby prechod A-C a B-D boli napojené na osobitný svetelný signál. Signál cesty A-C (Id 1) má pre zelené svetlo pridelenú väčšiu časť periódy prepínania signálov ako signál cesty B-D (Id 2). Vytvorí sa aj osobitné odbočovacie pruhy pre vozidlá odbočujúce vľavo. Aby sa vyhlo tvorby kolóny na týchto pruhoch pre odbočenie vľavo, zavedú sa ďalšie dva osobitné signály (Id 3, 4). Tie zabezpečia priestor pre vozidlá odbočujúce vľavo. Matica prepínania je na obrázku 5.48.



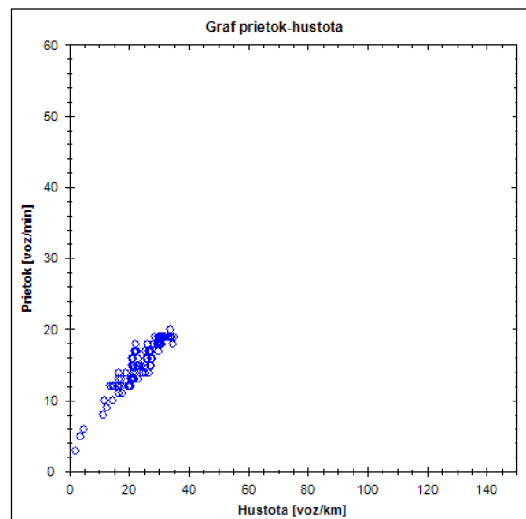
Obrázek 5.47: Križovatka so semaformi a osobitným odbočovacím pruhom.



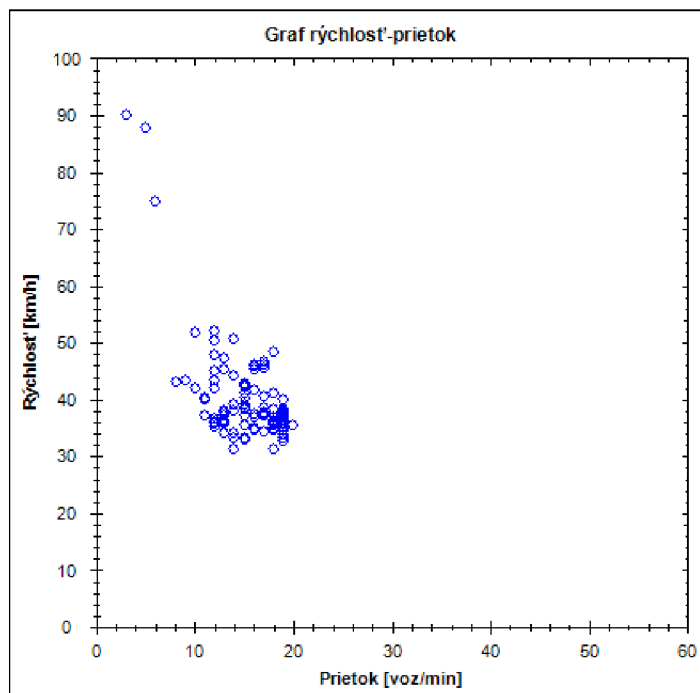
Obrázek 5.48: Matica prepínania semaforov. Signál s Id 1: Prechody AC, AD, CA, CB; Signál s Id 2: Prechody BD, BC, DB, DA; Signál s Id 3: Prechody AB, CD; Signál s Id 4: Prechody BA, DC.



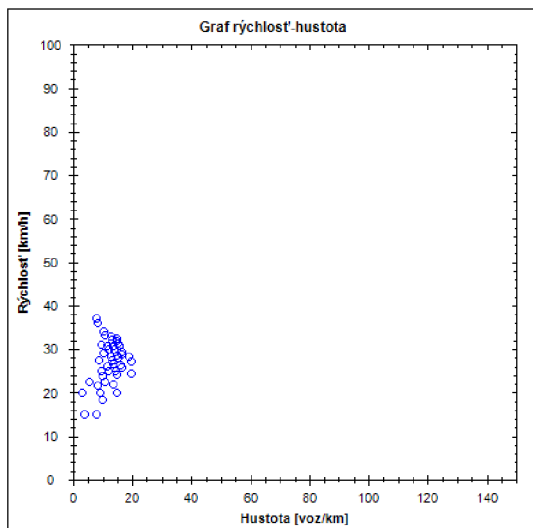
Obrázek 5.50: Graf rýchlosť-hustota na ceste A (pruh rovno/doprava).



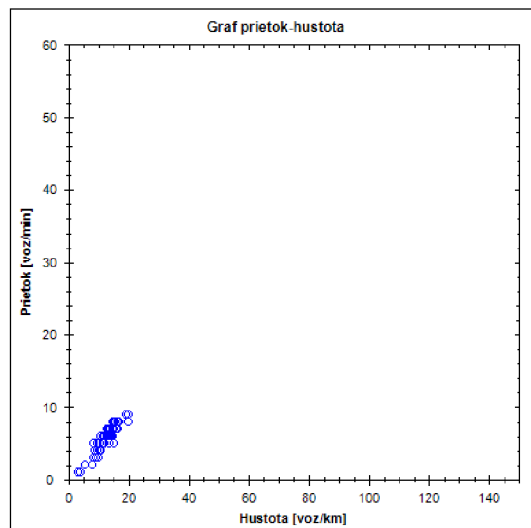
Obrázek 5.51: Graf prietok-hustota na ceste A (pruh rovno/doprava).



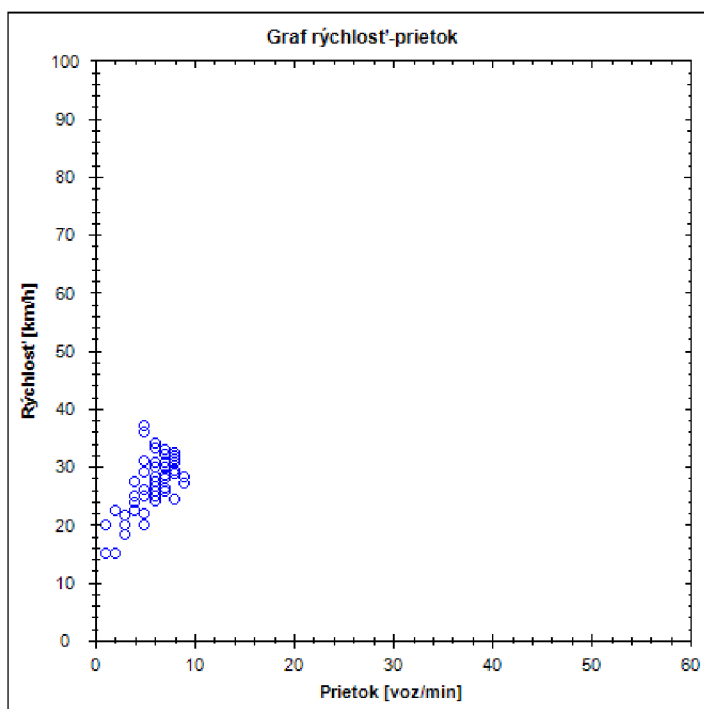
Obrázek 5.49: Graf rýchlosť-prietok na ceste A (pruh rovno/doprava).



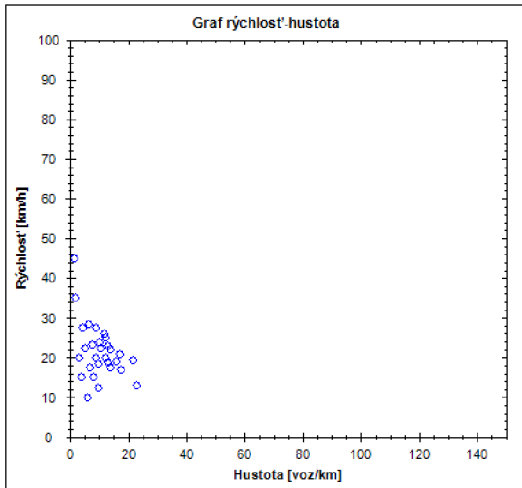
Obrázek 5.53: Graf rýchlosť-hustota na ceste B (pruh rovno/doprava).



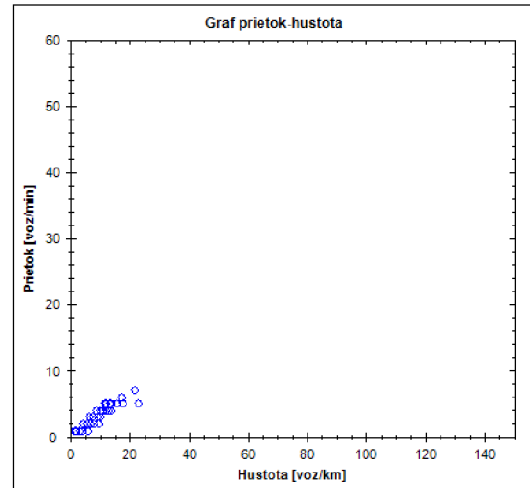
Obrázek 5.54: Graf prietok-hustota na ceste B (pruh rovno/doprava).



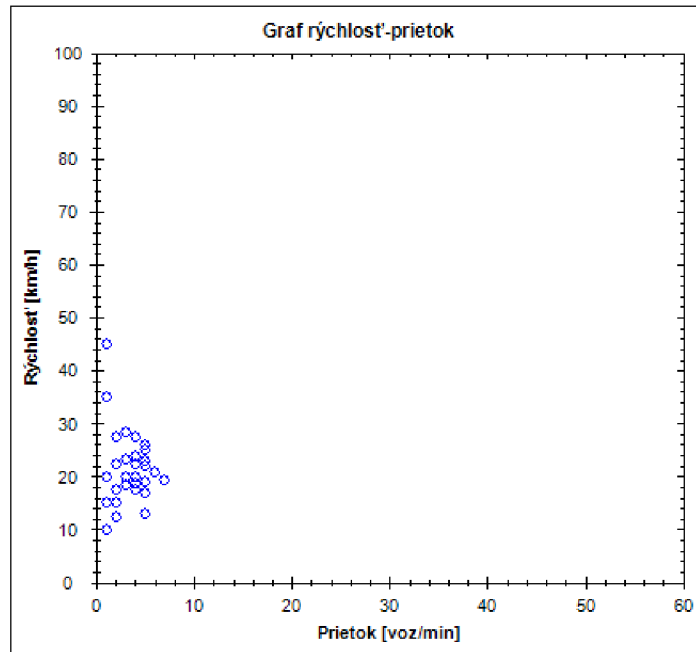
Obrázek 5.52: Graf rýchlosť-prietok na ceste B (pruh rovno/doprava).



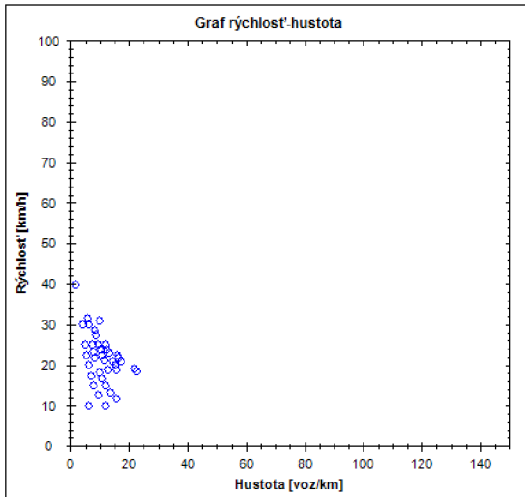
Obrázek 5.56: Graf rýchlosť-hustota na ceste C (pruh doľava).



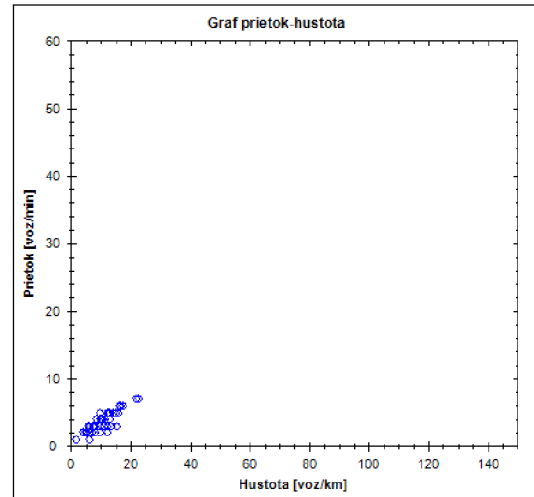
Obrázek 5.57: Graf prietok-hustota na ceste C (pruh doľava).



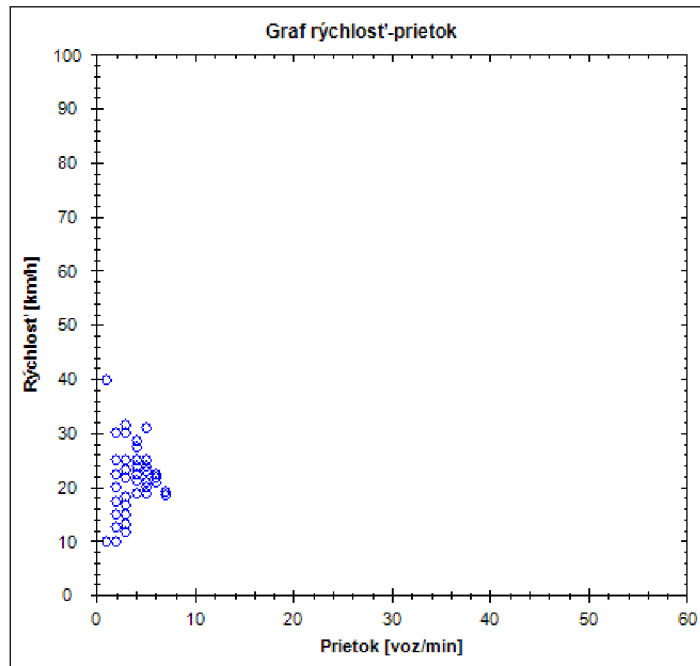
Obrázek 5.55: Graf rýchlosť-prietok na ceste C (pruh doľava).



Obrázek 5.59: Graf rýchlosť-hustota na ceste D (pruh doľava).



Obrázek 5.60: Graf prietok-hustota na ceste D (pruh doľava).



Obrázek 5.58: Graf rýchlosť-prietok na ceste D (pruh doľava).

Križovatka už dokázala obslúžiť vozidlá bez väčšieho zdržania. Počas simulácie sa tvorili len drobnejšie kolóny typické pre svetelnú križovatku.

Kapitola 6

Záver

V práci boli popísané základné princípy, na ktorých stojí implementovaný simulátor. Bol úspešne predvedený na niekoľkých cestných topológiách. Umožňuje simulovať bežné dopravné situácie a získavať údaje o premávke. Pri vývoji simulátora bolo potrebné riešiť niekoľko parciálnych problémov napríklad transformáciu súradníc z plátna na interné súradnice a naopak, realistické približovanie/oddľavovanie, navrhnutie križovatiek, určovaní predností v jazdách, spôsob vykresľovania topológie a vozidiel, generovanie nových vozidiel a pohyb vozidiel po ceste, ich prechody cez križovatky alebo zmeny jazdných pruhov. Nagel-Schreckenbergov model bol upravený spôsobmi, ktoré boli uvedené v teoretických východiskách na začiatku práce. Objektový model simulátora bol navrhnutý tak, aby bol čo najjednoduchšie rozšíriteľný pre budúce experimenty. Celulárny automat sa ukázal ako vhodný nástroj pre simuláciu cestnej premávky.

Práca mi priniesla skúsenosti s vývojom projektu väčšieho rozsahu založenom na objektivej orientácii. Získal som prax v tvorbe užívateľského prostredia pomocou nadstavby Windows Presentation Foundation. Osvojil som si vykresľovanie dvojrozmernej scény a posúvaní kamery. Získal som prehľad v základoch teórie o dopravnej premávke a možnostiach popisu jej vlastností pomocou makroskopických veličín.

Oblasť simulovania cestnej premávky je všeobecne komplikovaná oblasť, ktorá si vyžaduje nemalé prostriedky na vývoj. Jej výsledky ale môžu pomáhať v predikcii a optimalizácii riadenia cestnej premávky, čo sa môže vyplatiť obzvlášť v dobe stále zvyšujúcej sa intenzity premávky.

Súčasťou práce nebola validácia so skutočným svetom, existuje preto veľký priestor na optimalizáciu simulátora a zahrnutie ďalších faktorov, ktoré môžu ovplyvniť pohyb vozidiel. Potenciál na vylepšenia má aj grafické užívateľské rozhranie, ktoré obsahuje iba základné prvky na konštrukciu topológie, tak ako aj okno konfigurácie križovatky. Vykresľovanie je možné vylepšiť zobrazovaním jazdných pruhov, smeroviek vozidiel, aktuálnych svetelných signálov na križovatkách.

Literatura

- [1] MSDN Library. <http://msdn.microsoft.com/en-us/library/>.
- [2] Antal, A.; Bača, P.; Bakiča, M.; aj.: Analýza: Modelovanie a simulácia cestnej dopravy. <http://labs2.fiiit.stuba.sk/TeamProject/2008/team17is-si/web/dokumentacia.html>, 2008-12-10 [cit. 2012-01-26].
- [3] Esser, J.; Schreckenberg, M.: Microscopic simulation of urban traffic based on cellular automata. http://www.uni-due.de/imperia/md/content/ptt/paper/1997_ca_olsim.pdf, 1997-06-02 [cit. 2012-01-29].
- [4] Jablonskyte, J.: Simulations of vehicular traffic by cellular automata. <http://intranet.imet.gr/Portals/0/UsefulDocuments/documents/01725.pdf>, 2010-07-11 [cit. 2012-05-10].
- [5] Kršek, P.; Španěl, M.: Základy počítačové grafiky: Křivky v počítačové grafice. 2012 [cit. 2012-05-10].
- [6] Nagel, K.; Schreckenberg, M.: A cellular automaton model for freeway traffic. <http://hal.archives-ouvertes.fr/docs/00/24/66/97/PDF/ajp-jp1v2p2221.pdf>, 1992-09-10 [cit. 2012-01-26].
- [7] P. Korcek, L. S.; Fucik, O.: Evolutionary Approach to Calibration of Cellular Automaton Based Traffic Simulation Models. Nepublikováno, 2012-04-11 [cit. 2012-05-10].
- [8] Peringer, P.: Modelování a simulace IMS: Studijní opora. 2006 [cit. 2012-05-10].
- [9] Rakha, H.; Crowther, B.: A Comparison of the Greenshields, Pipes, and Van Aerde Car-Following and Traffic Stream Models. <http://www.filebox.vt.edu/users/hrakha/Publications/Car%20Following%20Comparison%20-%20TRB%2002-2143.pdf>, 2003-01-02 [cit. 2012-01-26].
- [10] S.P. Hoogendoorn, P. B.: State-of-the-art of Vehicular Traffic Flow Modeling, in Proc. of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, vol.215. 2001, 283-303.
- [11] Thomas, D. B.; Luk, W.; W.Leong, P. H. .; aj.: Gaussian random number generators. <http://dl.acm.org/citation.cfm?doid=1287620.1287622>, 2007 [cit. 2012-05-10].
- [12] Ředitelství silnic a dálnic ČR: Silnice a dálnice v České republice 2011. <http://www.rsd.cz/>, 2012 [cit. 2012-05-10].

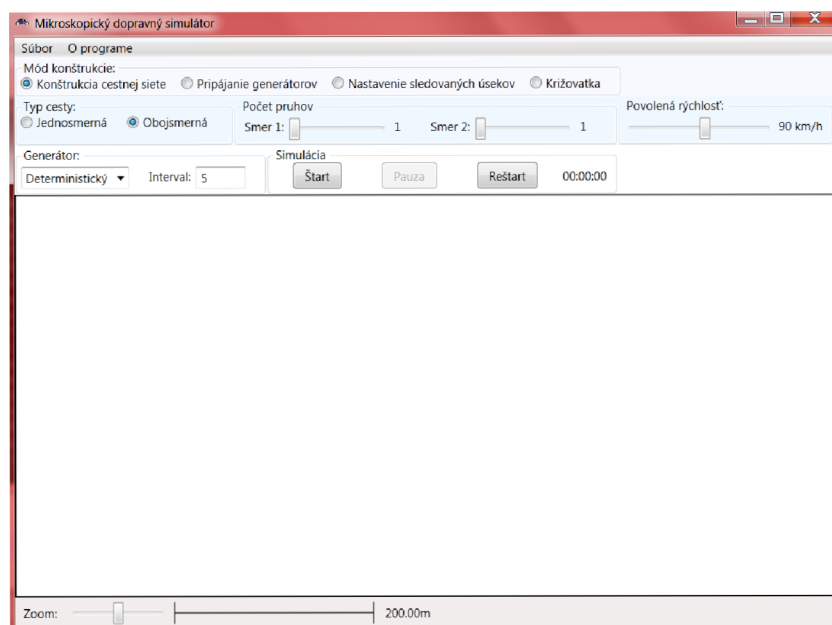
Příloha A

Obsah CD

<code>/bin/</code>	Spúšťací binárny súbor aj s podpornou .dll knižnicou na vykresľovanie grafov
<code>/src/</code>	Zdrojové súbory spolu s projektovými súbormi pre prostredie Visual Studio 2010
<code>/topologies/</code>	Súbory s uloženými topológiami na načítanie do programu
<code>/tex/</code>	Zdrojové texty tejto práce
<code>/bp_xfulop00.pdf</code>	PDF súbor s touto prácou
<code>/README.txt</code>	Popis obsahu DVD a návod na spustenie aplikácie

Příloha B

Manual a obrázky z aplikácie



Obrázek B.1: Aplikácia po spustení.

Možnosti v menu Súbor:

- Nový: zastavenie simulácie a vyčistenie plochy.
- Uložiť: uloženie topológie do súboru.
- Načítať: načítanie topológie zo súboru.
- Jazyk: zmena jazyka.
- Ukončiť: uzatvorenie aplikácie

Konštrukcia cesty:

- Mód konštrukcie nastaviť na *Konštrukcia cestnej siete*.

- Nastavenie typu cesty, počtu pruhov v oboch smeroch a povolenej rýchlosti pomocou prvkov na ovládacom paneli.
- Kliknutie LTM na plátno (ohraničené čiernou čiarou). Bude sa vykresľovať náhľad novej cesty zelenou farbou.
- Kliknutie LTM na plátno. Cesta sa postaví a nadobudne čiernu farbu.
- Ak sa niektorý koniec nachádza blízko iného konca cesty, spoja sa. Pokiaľ sa cesty krížia, vznikne most.

Konštrukcia generátora:

- Mód konštrukcie nastaviť na *Pripájanie generátorov*.
- Nastavenie parametrov generátora v ovládacom paneli.
- Kliknutie LTM na plochu. Generátor sa automaticky pripojí na najbližší koniec cesty. Objaví sa ako zelená bodka.

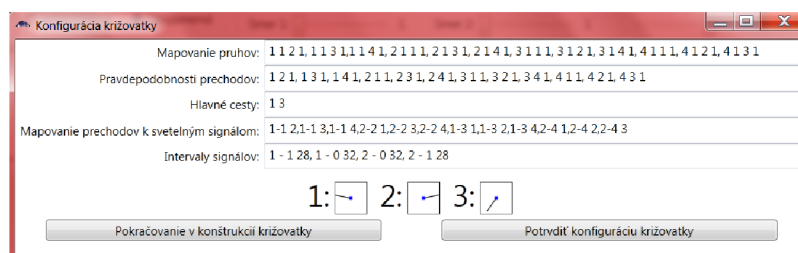
Posúvanie plochy:

- Držaním PTM a posúvaním myši sa presúva pohľad.
- Pohybom kolečka myši alebo posuvníku v spodnom paneli sa približuje/odďaľuje pohľad.

Konštrukcia križovatky:

Poznámka: Najprv je potrebné umiestniť križovatku a až potom sa k nej pripájajú cesty.

- Mód konštrukcie nastaviť na *Križovatka*.
- Kliknutie LTM sa umiestni križovatka. Je potrebné ju umiestniť mimo ktorejkoľvek cesty. Objaví sa ako modrá bodka.
- Konštrukciou cesty, ktorej jeden koniec sa umiestni blízko križovatky, sa pripojí nová cesta na križovatku.
- Ak sa na križovatku pripojí tretia cesta, objaví sa okno konfigurácie križovatky.



Obrázek B.2: Okno konfigurácie križovatky.

- Ak bolo plánované pripojiť na križovatku ďalšie cesty, kliknúť na tlačítko *Pokračovať v konštrukcii križovatky*. Po pripojení ďalšej cesty sa opäť objaví rovnaké okno. Teoreticky je možné pripojiť neobmedzené množstvo ciest.

- Po pripojení poslednej cesty konfigurácia križovatky pomocou zobrazeného okna. Po dokončení konfigurácie kliknúť na tlačítko *Potvrdiť konfiguráciu križovatky*.

Konfigurácia križovatky:

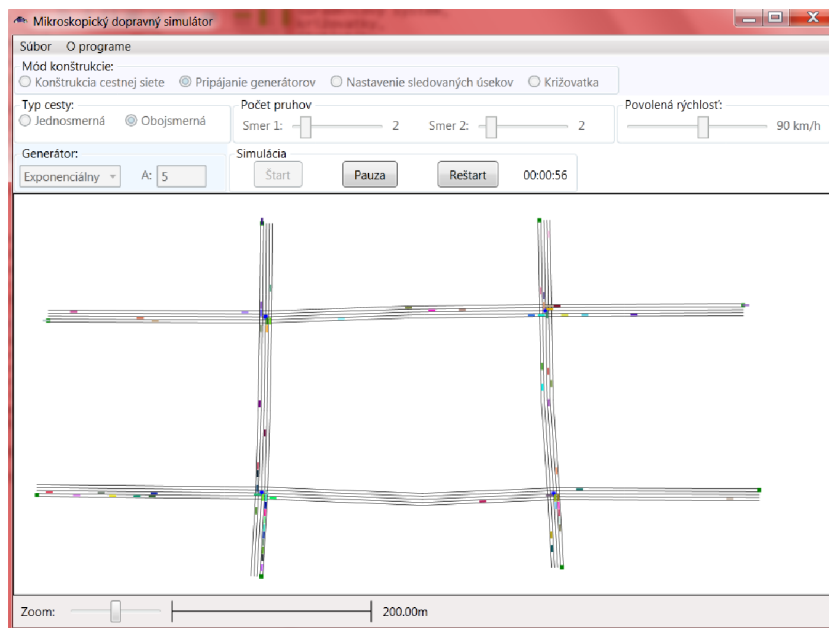
- Okno konfigurácie križovatky má už hneď po otvorení vyplnenú konfiguráciu pre svetelnú križovatku bez odbočovacích pruhov pre štyri dvojpruhové cesty.
- V dolnej časti sú znázornené jednotlivé cesty s identifikačným číslom, ktoré sú pripojené na križovatku.
- *Mapovanie pruhov*: textový reťazec skladajúci sa zo štvoric čísel $[c_1, p_1, c_2, p_2]$. Jedna štvorica priradí vstupnému pruhu jednej cesty výstupný pruh inej cesty. c_1 - identifikačné číslo vstupnej cesty, p_1 - poradie pruhu z pohľadu ku križovatke, c_2 - identifikačné číslo výstupnej cesty, p_2 - poradie pruhu z pohľadu od križovatky.
- *Pravdepodobnosti prechodov*: textový reťazec skladajúci sa z trojíc čísel $[c_1, c_2, p]$. Jedna trojica priradí vstupnej ceste c_1 pravdepodobnosť p prechodu na výstupnú cestu c_2 . Pravdepodobnosť sa zadáva ako celé číslo. Čím vyššie číslo, tým je väčšia pravdepodobnosť voľby tohto prechodu.
- *Hlavné cesty*: textový reťazec skladajúci sa z čísla c_1 a c_2 , identifikačných čísel hlavných ciest.
- *Svetelné signály*: trojice čísel $[s, c_1, c_2]$, s je identifikačné číslo signálu, c_1 identifikátor vstupnej cesty a c_2 identifikátor výstupnej cesty. Prechod $c_1 - c_2$ sa priradí signálu s .
- *Intervaly signálov*: trojice čísel $[s, rg, t]$, s je identifikačné číslo signálu, rg príznak červeného (0) alebo zeleného (1) signálu, t čas trvania tohto svetelného signálu s červeným/zeleným svetlom. Ak je trojíc čísel so signálom s viac (prakticky treba aspoň dve), striedajú sa v poradí, v akom boli napísané v textovom reťazci. Po dokončení posledného intervalu sa pokračuje opäť prvým intervalom.

Nastavenie grafického zobrazovania makroskopických veličín:

Poznámka: Pre každú bunku sa priebežne počítajú makroskopické veličiny. Držaním kurzoru myši počas simulácie nad cestou sa po chvíli objaví vyskakovacie okno s hodnotami veličín pre najbližšiu bunku ku kurzoru.

- Mód konštrukcie nastaví na *Nastavenie sledovaných úsekov*.
- Kliknúť LTM na cestu.
- Najbližšia bunka sa zobrazí žltou bodkou. Po spustení simulácie sa otvorí okno s grafmi makroskopických veličín. Grafy sa postupne počas simulácie dopĺňajú o aktuálne hodnoty.

Predchádzajúcimi postupmi je možné konštruovať aj zložitejšie topológie.



Obrázek B.3: Simulácia so zložitejšou topológiou.