

Úvod

V dnešní době je na dálnicích velká snaha o dynamické řízení dopravy obzvláště v místech s uzavírkami jízdních pruhů či při vysoké hustotě provozu. Uzavření jízdního pruhu může nastat prakticky kdykoliv a to plánovaně (např. při pravidelné údržbě) a nebo neplánovaně (dopravní nehoda). Prvním úkolem pro nás je, abychom se v tuto chvíli pokusili dopravu řídit, jak nejlépe to jde. Ani experti se ovšem neshodují, co znamená slovo nejlépe z předchozí věty. Může to být např. konkrétní hodnota intenzity či rychlosti, dojezdový čas či další parametry dopravního proudu.

Pokud získáme surová data z daného úseku, např. intenzita vztažená na minutu pro každý pruh, a následně je vyhodnotíme, dostaneme přesné informace o úseku. Poté můžeme také tyto data vzít a pokusit se nasimulovat danou oblast. Otázkou zůstává jak. Ideálně tak, aby simulace byla blízka k již získaným informacím o úseku. V této práci se zaměříme na mikroskopické modely, podle kterých budeme simulace provádět. To znamená, že budeme simulovat každé jednotlivé vozidlo.

Po simulacích vyhodnotíme jednotlivé modely a zjistíme, které modely odpovídají reálným situacím. Předpoklad je takový, že některé modely budou přesnější např. při vysokých hustotách, některé zase naopak.

Ve chvíli, kdy získáme tento přehled modelů, můžeme již přesněji analyzovat danou situaci a do budoucna budeme vědět, že daný model simuluje tento stav nejlépe.

Hlavní cíl práce tedy spočívá v simulaci na základě reálných dat pomocí různých mikroskopických modelů a výsledky z těchto simulací posléze porovnat s reálnými daty. Tím pádem se posléze naskytá další možnost využití při řízení dané oblasti či predikování např. při plánované uzavírce.

Cíle práce

Cíle práce lze formulovat následovně:

- Seznámení se se základními typy mikroskopických modelů (hlavně jejich matematickým popisem) v dopravě pro použití na dálnicích
- Seznámit se s kalibrací těchto modelů
- Seznámit se simulačním softwarem SUMO, možnosti jeho využití

Tato témata jsou náplní první, teoretické části práce. V druhé, praktické, části byly cíle práce následující.

- implementace všech mikroskopických modelů z teoretické části do simulačního softwaru SUMO
- kalibrace mikroskopických modelů z teoretické části
- nasimulování reálných dat, případně vytvoření vlastních scénářů pro mikroskopické modely
- Makroskopický popis výsledků simulace.
- Mikroskopický popis výsledků simulace.

1. Modely předcházející IDM

Jednoduše řečeno „Car Following“ modely vycházejí z předpokladu, že pokud n -té vozidlo následuje $n - 1$ vozidlo na homogenní dálnici, trajektorie n -tého vozidla bude stejná jako vozidla před ním ($n - 1$) až na posuny v čase a místě. (Vozidlo bude ve stejném místě jako předcházející v jiný čas a ve stejný čas bude na jiném místě.) Všechny další dispozice jednotlivých modelů v sobě tento předpoklad obsahují.

1.1 Newellův model

V teorii dopravního proudu je Newellův model jeden z modelů, který popisuje chování řidiče dle vozidla před ním. Řidič se snaží držet za vozidlem jedoucím před ním v konstantní vzdálenosti. Tento model vznikl v roce 1961 a postupně se vyvíjel. Vychází z něho později Intelligent Driver Model (IDM) a z tohoto důvodu je zde zmíněn.

V článku [11] je přesně popsán nejen model, ale i jeho verifikace a porovnání s ostatními modely. Pro nás je důležitý popis modelu.

1.1.1 Popis modelu

Jestliže n -té vozidlo jede za $n - 1$ vozidlem (které jede za $n - 2$ vozidlem atd.), cíl každého Car following modelu je zjistit závislost trajektorie n -tého vozidla $x_n(t)$, jeho pozici v čase t na $n - 1$ vozidle, viz. obrázek 1.1. (To také znamená, že není žádná náhodná spojitost mezi těmito vozidly.) Jestliže se $n - 1$ vozidlo pohybuje konstantní rychlostí v ,

$$x_{n-1} = x_n + vt,$$

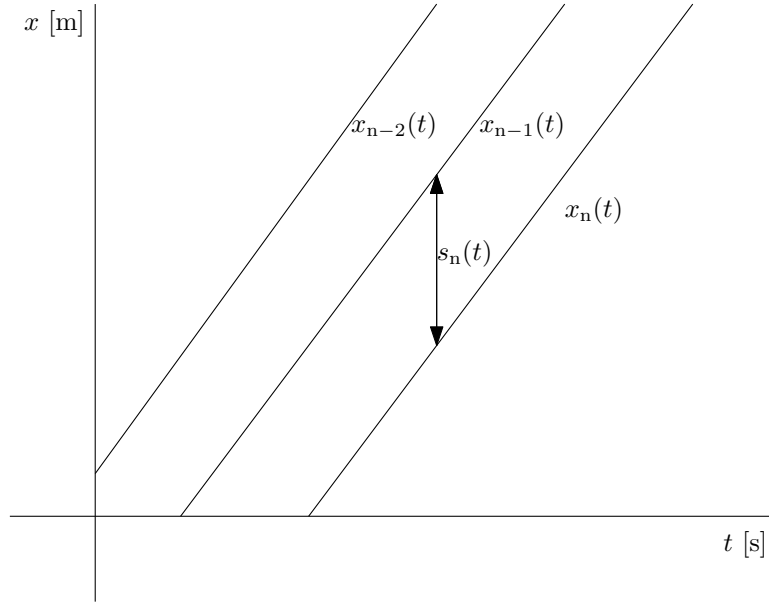
n -té vozidlo také pojede průměrnou rychlostí v . Pokud by n -té vozidlo zrychlovalo, tak by došlo ke kolizi s $n - 1$ vozidlem a naopak pokud by zpomalovalo, tak by se n -té vozidlo neustále vzdalovalo. Totéž platí pro všechna vozidla jedoucí za n -tým. Tento model se nezabývá zjištěním hodnoty rychlosti v , předpokládá se, že je určena buďto na základě omezení rychlosti či možnostech vozidla.

Vzdálenost $s_n = x_{n-1}(t) - x_n(t)$ mezi vozidly n a $n - 1$ se může měnit v čase. Pokud je dálnice homogenní, tato vzdálenost zůstane konstantní okolo hodnoty s_n . Tato hodnota se mění na základě typu vozidla a také závisí na rychlosti v .

Předpokládejme, že existuje nějaký empirický vztah mezi rychlostí v a vzdáleností mezi vozidly s_n . Pokud rychlost v roste, je logické, že řidiči chtějí dosáhnout většího rozestupu mezi vozidly. Tato závislost mezi v a s_n je znázorněna na obrázku 1.2. Každý řidič má svoji preferovanou rychlost V_n . Jestliže tato rychlost je u $n - 1$ vozidla vyšší než u n -tého $v > V_n$, znamená to, že n -té vozidlo pojede svoji preferovanou rychlostí (na obrázku 1.2 znázorněna čárkovanou čarou) a $n - 1$ vozidlo mu ujede. Hodnota rychlosti v nemůže být záporná a vzdálenost mezi vozidly při nulové rychlosti by měla být níže než polopřímka lineární závislosti, viz černá tečka na obrázku 1.2 při rychlosti $v = 0$.

Nyní předpokládejme, že $n - 1$ se vozidlo nějakou dobu t pohybuje konstantní rychlostí v^1 a potom náhle změní rychlost na hodnotu v' . Trajektorie vozidel n a $n - 1$ mohou poté vypadat jako na obrázku 1.3. Z obrázku lze také vypočítat jak časovou τ_n , tak prostorovou d_n mezeru

¹hodnota rychlosti osciluje okolo hodnoty v



Obrázek 1.1: Trajektorie vozidel s konstantní rychlostí, zdroj: [11]

mezi vozidly n a $n - 1$. Z čárkovaného obdelníku poté dostáváme vztah vzdálenost mezi vozidly před změnou rychlosti s_n a po změně rychlosti s'_n ,

$$s_n = d_n + v\tau_n, \quad s'_n = d_n + v'\tau_n.$$

Z toho vyplývá, že pokud leží v a v' na polopřímce z obrázku 1.2, sklon této přímky je právě τ_n a hodnota s_n při rychlosti $v = 0$ je d_n .

Ze vztahu mezi v a s_n , jak je zobrazeno na obrázku 1.2, plyne nezávislost mezer d_n a τ_n na rychlostech v , resp. v' . Pokud se tedy změni rychlost z hodnoty v' na v'' , d_n a τ_n zůstanou při této změně rychlosti stejné. Lineární trajektorie vozidla $x_n(t)$ potom bude jednoduše posun v čase τ_n a místě d_n ,

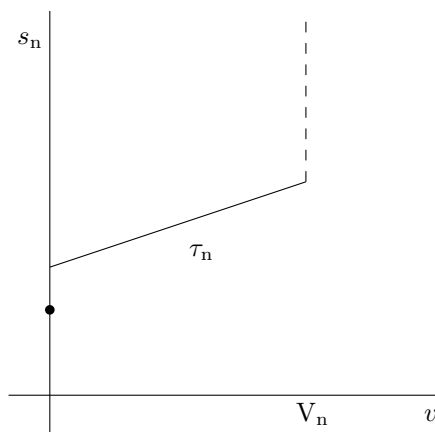
$$x_n(t + \tau_n) = x_{n-1}(t) - d_n. \quad (1.1)$$

U Newellova modelu platí, že n -té vozidlo bude (přibližně) kopírovat trajektorii $n-1$ vozidla dle vztahu (1.1) při vhodných hodnotách d_n a τ_n . Tím, jak se přesně dokáže n -té vozidlo dodržovat vztah (1.1), se Newellův model nezabývá, pouze předpokládá, že řidič je schopen se tímto vztahem řídit. To není tak těžké, protože pokud se změni rychlost vozidla $n - 1$, n -tý řidič nemusí zareagovat okamžitě, ale může počkat dokud se mezera s_n nezvýší (neklesne) na hodnotu, která odpovídá nové rychlosti vozidla $n-1$ (viz obrázek 1.2).

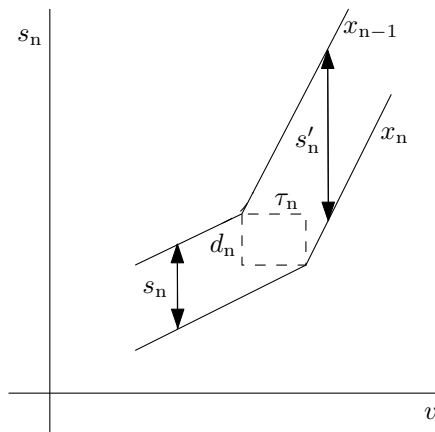
Při pozorování se došlo k závěru, že každý řidič nezkontroluje každého jiného řidiče na komunikaci, ale pouze jen vhodnou „makroskopickou“ část. Hodnoty τ_n a d_n jsou přirozené, to vychází ze vztahu (1.1), kde postupným iterováním dostaneme

$$x_n(t + \tau_n + \tau_{n-1} + \dots + \tau_1) = x_0(t) - d_n - d_{n-1} - \dots - d_1. \quad (1.2)$$

Hodnoty τ_n a d_n se značně liší mezi jednotlivými vozidly v závislosti na typu řidiče. První skupina řidičů raději jede blíže vozidlu před sebou ($n - 1$) až na minimální bezpečnou vzdálenost,



Obrázek 1.2: Vztah mezi rychlostí v a vzdáleností mezi vozidly s_n , zdroj: [11]



Obrázek 1.3: Lineární aproximace při změně rychlosti vozidel, zdroj: [11]

zatímco druhá skupina naopak preferuje delší vzdálenost pro klidnou reakci na nenadálou událost. Je rozumné, aby se hodnoty τ_n a d_n lišily, a to tak, že každému jednotlivému vozidlu budou hodnoty vygenerovány z nějakého pravděpodobnostního rozdělení, jehož variační koeficient se bude blížit jedné. Při generování rychlostí vozidel by naopak měl být variační koeficient zanedbatelný.

Položíme-li

$$\bar{\tau} = \frac{1}{n} \sum_{k=1}^n \tau_k, \quad \bar{d} = \frac{1}{n} \sum_{k=1}^n d_k, \quad (1.3)$$

kde $\bar{\tau}$ a \bar{d} jsou průměrné posuny v čase resp. místě, potom průměrnou vlnovou rychlost spočteme jako podíl $\bar{d}/\bar{\tau}$.

Tento model se dá ovšem popsat nejen mikroskopicky (rychlost, mezera mezi vozidly), ale i makroskopicky (hustota k , intenzita q). Stacionární stav nastane ve chvíli, kdy se všechna vozidla budou pohybovat konstantní rychlostí s rozdílnými posuny (časovými i prostorovými).

Jestliže

$$s_n = d_n + v\tau_n,$$

a rychlost vozidel je konstantní, tak platí

$$\bar{s} = \bar{d} + v\bar{\tau}.$$

Hustotu k lze potom vypočítat jako převrácenou hodnotu průměrné mezery mezi vozidly $k = 1/\bar{s}$ a rychlost jako podíl intenzity a hustoty $v = q/k$. Tudíž platí

$$q = \frac{1}{\bar{\tau}} - \frac{\bar{d}}{\bar{\tau}}k, \quad (1.4)$$

za předpokladu, že rychlost v je menší než preferovaná rychlost jakéhokoliv vozidla V_k .

Rovnice (1.4) propojuje Newellův model s klasickými makroskopickými modely. Problém nastává ve chvíli, kdy průměrná rychlost v je vyšší, než některá s preferovaných rychlostí jednotlivých vozidel. Vozidla se v tomto modelu nemohou předjíždět a proto vznikne kongesce za vozidlem s malou preferovanou rychlostí. V následujících řádcích předpokládáme, že aktuální rychlost vozidla bude menší než preferovaná rychlost V_n .

Uvažujme, že se vozidlo n pohybuje přesně podle rovnice (1.1) a vozidlo za ním $(n-1)$ jede plynule za ním.

Rovnici (1.1) můžeme přepsat do tvaru

$$x_n(t + \tau_n) = x_n(t) + \tau_n v_n(t + T_n). \quad (1.5)$$

Rovnice (1.5) lze pro rovnoměrný pohyb zjednodušit na tvar

$$x_n(t + \tau_n) = x_n(t) + \tau_n v_n(t).$$

Tvar rovnice (1.5) plyne z matematické analýzy², přičemž hodnota T_n se nachází někde mezi nulou a τ_n . Pokud je funkce hladká bude přibližně

$$T_n = \frac{\tau_n}{2}. \quad (1.6)$$

Rovnici (1.5) lze přepsat na přibližný tvar

$$x_n(t + \tau_n) = x_n(t) + \tau_n v_n(t) + \tau_n T_n a_n(t), \quad (1.7)$$

který uvažuje zrychlení vozidla a_n ³. Kombinací rovnic (1.7) a (1.1) dostáváme vztah pro rychlost

$$v_n(t + \tau_n) = \frac{1}{\tau_n} [x_{n-1}(t) - x_n(t)] - \frac{d_n}{\tau_n}. \quad (1.8)$$

Po zderivování rovnice (1.8) dostáváme vztah pro zrychlení vozidla

$$a_n(t + \tau_n) = \frac{1}{\tau_n} [v_{n-1}(t) - v_n(t)]. \quad (1.9)$$

²věta o střední hodnotě

³Předpokládejme vozidlo v klidu, které se rozjíždí. Na začátku zvolíme $\tau_n = 5s, t = 0s, T_n = \tau_n/2 = 2,5s$, potom platí

$$x_n(5) = x_n(0) + 5v_n(0 + 2,5).$$

Pro výpočet dráhy bereme hodnotu rychlosti v čase 2,5s. Nemůžeme vzít hodnotu v nule, vozidlo by v tomto případě stálo na místě. Přibližná hodnota bude tedy opravdu v polovině hodnoty τ_n .

Z rovnic (1.8) a (1.9) je vidět zřejmá závislost na vozidle, které jede před aktuálním, členem $v_{n-1}(t)$, resp. $x_{n-1}(t)$. Řidič volí svoji rychlost na základě odstupu od předchozího vozidla, resp. zrychlení na základě rychlostí. **Konečný vztah pro Newellův model je tedy**

$$a_n(t + T_n) = \frac{1}{\tau_n} [v_{n-1}(t) - v_n(t)] - \frac{d_n}{\tau_n} - v_n(t). \quad (1.10)$$

1.2 Gippsův model

Další model, kterým se zde budeme zabývat, se jmenuje Gippsův. Tento model byl poprvé popsán v roce 1981 a odvolává se i na zmíněný Newellův model. I v dnešní době se tento model využívá pro simulaci dopravy. Hlavním zdrojem pro tuto podkapitolu je článek P.G. Gippse[4]. Většina modelů, které vznikly před Gippsovým, jsou různé variace na rovnici

$$a_n(t + T_n) = l_n \frac{[v_{n-1}(t) - v_n(t)]^k}{[x_{n-1}(t) - x_n(t)]^m}, \quad (1.11)$$

kde jednotlivé proměnné mají stejný význam jako při popisu Newellova modelu 1.1. Proměnné l_n , k a m jsou parametry, které je nutno empiricky odhadnout. Tyto modely se nazývají Modely General Motors a jsou blíže rozebrány v [17].

Ačkoliv podávají tyto modely dobré výsledky v mnoha situacích, je žádoucí, aby během přepočtů poloh, rychlostí a zrychlení nějakým způsobem byl zahrnut reakční čas řidiče T_n . To si vyžaduje značné množství historických dat, pokud model bude použit v simulačním programu. Navíc, parametry l_n , k a m nemají přímou souvislost s vlastnostmi vozidla či řidiče.

Gipps se proto rozhodl, že vytvoří model, který bude splňovat následující podmínky:

- model by měl napodobovat chování opravdového provozu,
- parametry modelu by měly odpovídat vlastnostem řidiče a vozidla, které jsou jasné; to znamená, že pro nastavení parametrů není potřeba model kalibrovat,
- model by se měl chovat správně i v případě, že doba mezi přepočítáváním polohy a rychlosti je stejná, jako reakční čas řidiče.

1.2.1 Popis modelu

Následující model je odvozen z omezení jak řidiče, tak i vozidla a pomocí těchto omezení se dopočítávají bezpečná rychlost podle předcházejícího vozidla. Předpokládá se přitom, že řidič volí svoji rychlost tak, aby zajistil bezpečné zastavení nebo aby dokázal náhle rychle zastavit při nenadálé události.

První podmínka, kterou aplikujeme na vozidlo n , souvisí s rychlostí vozidla. Ta nepřekročí jeho preferovanou rychlost V_n a jeho zrychlení na volné komunikaci⁴ se zvyšuje stejně tak jako rychlost dokud se také zvyšuje kroutící moment motoru vozidla a poté začne klesat na nulovou hodnotu a té nabude ve chvíli, kdy se vozidlo bude pohybovat svoji preferovanou rychlostí V_n

$$v_n(t + T_n) = v_n(t) + 2,5a_n\tau \left(\frac{1 - v_n(t)}{V_n} \right) \sqrt{\left(0,025 + \frac{v_n(t)}{V_n} \right)}. \quad (1.12)$$

Koeficienty v nerovnici 1.12 byly stanoveny na základě z měření na hlavních dopravních komunikacích při průměrném provozu. Využití nerovnice 1.12 pro tento model je považované za

⁴tj. komunikace, na které se nevyskytuje žádné další vozidlo

přijatelné až do chvíle, kdy se vozidlo přiblíží vozidlu jedoucím před ním. Od té chvíle důraz na tuto podmínku klesá, naopak největší důležitost je ve chvíli, kdy vozidlo nemá před sebou žádné vozidlo či vozidlo před ním se nachází velmi daleko.

Další omezení, které je nutno zmínit, se týká brždění. Jestliže $n - 1$ (první) vozidlo zahájí brždění v čase t , zpomalí a následně zastaví v místě $x_n^* - 1$, které dostáváme rovnicí

$$x_{n-1}^* = x_{n-1}(t) - \frac{v_{n-1}(t)^2}{2b_{n-1}}, \quad (1.13)$$

kde se člen b_{n-1} nazývá nejvyšší kritická decelace, kterou je řidič vozidla $n - 1$ schopen vykonat. Její hodnota je vždy záporná.

Vozidlo n jedoucí za ním nebude reagovat na toto zpomalení ihned, ale až v čase $t + T_n$ a tudíž zastaví až v místě x_n^* dané rovnicí

$$x_n^* = x_n(t) + [v_n(t) + v_n(t + T_n)] \frac{T_n}{2} - \frac{v_n(t + T_n)^2}{2b_n} \quad (1.14)$$

A proto pro bezpečnost musí řidič vozidla n zajistit, že bude splněna podmínka

$$x_{n-1}^* - s_{n-1} < x_n^*, \quad (1.15)$$

kde s_{n-1} je délka vozidla $n - 1$ sečtena s bezpečnou vzdáleností následujícího (n) vozidla. Nicméně pokud tato podmínka neplatí, vozidlo nemá žádný prostor pro případnou řidičovu chybu. Z toho důvodu obsahuje tento model ještě další podmínku; bezpečný odstup s další časovou rezervou, θ . Řidič začne reagovat na vozidlo před sebou až v čase $t + T_n$. Od této chvíle tedy máme reakční čas T_n a bezpečný reakční čas $T_n + \theta$, který budeme ve výpočtech dále využívat. S využitím rovnic (1.13), (1.14) a (1.15) můžeme tuto podmínku přepsat do rovnice

$$x_{n-1}(t) - \frac{v_{n-1}(t)^2}{2b_{n-1}} - s_{n-1} \geq x_n(t) + [v_n(t) + v_n(t + T_n)] \frac{T_n}{2} + v_n(t + T_n)\theta - \frac{v_n(t + T_n)^2}{2b_n} \quad (1.16)$$

Bez parametru θ by řidič vozidla byl nucen brzdit až ve chvíli, kdy opravdu musí, to znamená na maximální výkon brzd. Do té chvíle by se stále pohyboval svoji preferovanou rychlostí. Parametr θ slouží k dřívějšímu a přitom ne tak prudkému brždění.

Při skutečném pozorování dopravy jsme schopni změřit všechny parametry vozidla n v rovnici (1.16) kromě členu b_{n-1} . Tento člen nahradíme odhadnutou hodnotou \hat{b} , upravíme tuto rovnici a dostáváme

$$-\frac{v_n(t + T_n)^2}{2b_n} + v_n(t + T_n) \left(\frac{T_n}{2} + \theta \right) - [x_{n-1}(t) - s_{n-1} - x_n(t)] + v_n(t) \frac{T_n}{2} + \frac{v_{n-1}(t)^2}{2\hat{b}} \leq 0. \quad (1.17)$$

Relativní hodnoty T_n a θ jsou důležité v určování chování vozidel. Stejně jako v případě Newellova modelu 1.1, (1.6), pokud je hodnota θ rovna právě jedné polovině T_n , vozidlo jedoucí bezpečnou rychlostí bude schopno udržovat tento stav nekonečně dlouhou dobu. Díky tomu můžeme rovnici (1.17) přepsat do tvaru

$$-\frac{v_n(t + T_n)^2}{2b_n} + v_n(t + T_n)T_n - [x_{n-1}(t) - s_{n-1} - x_n(t)] + v_n(t) \frac{T_n}{2} + \frac{v_{n-1}(t)^2}{2\hat{b}} \leq 0. \quad (1.18)$$

Z toho plyne

$$v_n(t + T_n) \leq b_n T_n + \sqrt{\left(b_n^2 T_n^2 - b_n \left(2[x_{n-1}(t) - s_{n-1} - x_n(t)] - v_n(t)T_n - \frac{v_{n-1}(t)^2}{2\hat{b}} \right) \right)}. \quad (1.19)$$

Z nerovnice (1.18) vychází, že bezpečné rychlosti v_n budou ležet mezi dvěma kořeny této nerovnice, ale pokud bude mít spodní kořen zápornou hodnotu můžeme ho ignorovat, protože nás zajímají pouze kladné hodnoty rychlostí. Podcenění plynulého brždění řidiče vozidla n nastane ve chvíli, kdy

$$v_n(t + T_n) < v_n(t) + b_n T_n.$$

V tuto chvíli musí vozidlo začít brzdit rychleji, než by sám řidič chtěl. Řidič tedy volí rychlost na základě preferované decelarace, ale může brzdit i rychleji pokud by to bylo nutné.

Nerovnice (1.12) pro vozidlo n na volné komunikaci a (1.19) pro vozidlo n před kterým se nachází vozidlo $n - 1$ představují dvě hlavní podmínky pro rychlost v_n v čase $t + T_n$ a jestliže řidič přizpůsobí rychlost parametrům vozidla a tak, aby jízda byla bezpečná, dostáváme **konečný vztah pro Gippsův model**:

$$v_n(t+T_n) = \min \left(v_n(t) + 2, 5a_n\tau \left(\frac{1 - v_n(t)}{V_n} \right) \sqrt{\left(0,025 + \frac{v_n(t)}{V_n} \right)}, \right. \\ \left. b_n T_n + \sqrt{\left(b_n^2 T_n^2 - b_n \left(2[x_{n-1}(t) - s_{n-1} - x_n(t)] - v_n(t)T_n - \frac{v_{n-1}(t)^2}{2\hat{b}} \right) \right)} \right) \quad (1.20)$$

Pokud platí pro téměř všechna vozidla nižší hodnota v rovnici (1.20), vychází se z nerovnice (1.19), vozidla jedou blízko sebe, z čehož vyplývá vysoká hustota provozu. Pokud platí opak, tz. nerovnice (1.12), dopravní proud je volný.

1.3 Kraussův model

Pro simulaci v této práci budeme využívat software SUMO⁵ a součástí tohoto programu je předpřipravený Kraussův model vytvořený v rámci disertační práce stejnojmenného autora. V popisu tohoto modelu budeme vycházet z této práce[8].

Tento model se zaměřuje primárně na všeobecné vlastnosti dopravního proudu, zkoumání chování řidiče zde není primární. Myšlenka autora je taková, že existují obecné vlastnosti dopravního proudu, ze kterých vyplývá chování jednotlivých účastníků silničního provozu a není to chování jednotlivce, které hlavně ovlivňuje vlastnosti dopravního proudu.

1.3.1 Obecný přístup

Pokud se dopravní proud modeluje mikroskopicky, musí být brány v úvahu dva typy pohybu vozidla. Prvním typem pohybu je vozidlo jedoucí po vozovce osamoceno, zatímco ve druhém případě vozidlo interaguje s ostatními vozidly. Na základě těchto pohybů předpokládáme následující podmínky. První podmínka má přímo souvislost s rychlostí vozidla. Ta je omezena nějakou maximální rychlostí v_{\max} ,

$$v \leq v_{\max}. \quad (1.21)$$

Za maximální rychlost může být například zvolena preferovaná rychlost vozidla.

Hlavní důvod, proč mezi sebou vozidla interagují, je fakt, že řidiči nemají v úmyslu srážet se s ostatními vozidly. Další podmínkou tedy bude tzv. „nekoliznost“ systému. Budeme předpokládat,

⁵Simulation of Urban MObility

že řidič vozidla zvolí tedy takovou rychlost vozidla, která nebude vyšší než maximální bezpečná rychlost v_{safe} ,

$$v \leq v_{\text{safe}}. \quad (1.22)$$

Z maximální bezpečné rychlosti poté plyne interakce mezi vozidly. Její stanovení bude uvedeno dále.

Je možné formulovat modely na základě podmínek (1.21) a (1.22). Nicméně je vcelku vhodné předpokládat také omezující podmínky pro zrychlení,

$$\begin{aligned} -b &\leq \frac{dv}{dt} \leq a, \\ a, b &> 0. \end{aligned} \quad (1.23)$$

Později bude ukázáno, že podmínka (1.23) je nezbytnou složkou, pokud chceme modelovat dopravní proud správně.

Předchozí podmínky můžeme shrnout do následující nerovnice

$$v(t + \Delta t) \leq \min(v_{\text{max}}, v_{\text{safe}}, v(t) + a\Delta t), \quad (1.24)$$

kde se maximální bezpečná rychlost v_{safe} musí splňovat podmínku

$$v(t + \Delta t) \geq v(t) - b\Delta t. \quad (1.25)$$

Celkové povědomí o tom, jak se jednotlivá vozidla pohybují a interagují mezi sebou závisí na výpočtu maximální bezpečné rychlosti v_{safe} . Jakmile je stanovena hodnota v_{safe} , nerovnice (1.24) představuje aktualizované schéma pro simulace dopravního proudu, pokud jsou splněny všechny podmínky zmíněné výše.

1.3.2 Interakce mezi vozidly

Pro vytvoření sofistikovaného dopravního modelu je nutné ukázat, jak vozidla mezi sebou interagují. Podívejme se na dvě vozidla na komunikaci, první vozidlo se nachází v bodě x_1 a jede rychlostí v_1 , vozidlo, které ho následuje je v bodě x_f a pohybuje se rychlostí v_f . Jestliže délka prvního vozidla bude l , tak mezera mezi vozidly se vypočte vztahem

$$g = x_1 - x_f - l. \quad (1.26)$$

Jak již bylo zmíněno, hlavní důvod, proč mezi sebou vozidla interagují, je fakt, že řidiči nemají v úmyslu srážet se s ostatními vozidly. Z toho plyne, že mezera mezi vozidly musí být nezáporná. Narozdíl od jiných modelovaných přístupů, zde nezačneme s předpokladem jak může být zrychlení vozidla vypočteno z vozidla jedoucí před ním, protože nekoliznost stejně není splněna automaticky a někdy je velice složité ji dokázat.

Místo toho začneme zavedením toho, že se v spojitých modelech spolu interagující vozidla nesrazí tehdy, pokud je mezera mezi vozidly g větší než nějaká preferovaná mezera mezi vozidly g_{des} a splňuje tuto dynamickou nerovnici

$$\frac{dg}{dt} \geq \frac{g_{\text{des}} - g}{\tau_{\text{des}}}. \quad (1.27)$$

Preferovaný bezpečný časový odstup mezi vozidly τ_{des} a preferovaná mezera g_{des} by mohly funkcemi odstupů mezi vozidly či jejich rychlostmi. Fakt, že v tomto modelu nemohou nastat

kolize je zřejmá, protože pokud položíme g rovné nule časová derivace bude vždy nezáporná a preferovaná mezera g_{des} také.

Předpokládejme případ dvou vozidel jedoucích rychlost v_1 resp. v_f s odstupem g . Druhé vozidlo jede svoji bezpečnou rychlostí a řidič tohoto vozidla může zastavit za jakýkoliv okolností tak, aby nedošlo ke kolizi s prvním vozidlem. To znamená, že pokud řidič má reakční čas τ a brzdná vzdálenost vozidel jedoucích rychlostí v je dána funkcí $f(v)$, je situace bezpečná právě tehdy když

$$f(v_f) + v_f\tau \leq f(v_1) + g. \quad (1.28)$$

Hodnota funkce $f(v)$ nemusí být nutně minimální možná brzdá vzdálenost, ale může to být jiná funkce závisající na způsobu řízení řidiče a pohodlné jízdy. To samé platí pro reakční čas τ .

Úprava nerovnice 1.28 vede k bezpečnostnímu pravidlu pro rychlost druhého vozidla. Nicméně my toto pravidlo neodvozujeme ze dvou důvodů. První z nich je ten, že po úpravě bychom potřebovali znát přesný předpis funkce $f(v)$, který jak uvidíme není možné zjistit. Zadruhé je zbytečné a výpočetně náročné používat typ řízení, které potřebuje předvídat nenádalou událost a vypočítávat decelaci až k úplnému zastavení každý jednotlivý krok. Proto zavedeme Taylorův rozvoj brzdě vzdálenosti $f(v)$ okolo průměrné rychlosti obou vozidel,

$$\bar{v} = \frac{v_1 + v_f}{2}, \quad (1.29)$$

který budeme v dalším odvození dále používat. Sudé členy tohoto rozvoje se vyruší a zanedbáním členů vyšších řádů dostáváme

$$f'(\bar{v})v_f + v_f\tau \leq f'(\bar{v})v_1 + g. \quad (1.30)$$

Smysl derivace $f'(\bar{v})$ vysvětlíme jednoduše. Pokud se podíváme na decelaci vozidla b z rychlosti v na nulu (nemusí být nutně konstantní), tedy $\dot{v} = -b(v)$, kdy $b(v) > 0$, obrdžíme vztah

$$f'(v) = -\frac{d}{dv} \int_v^0 \frac{v'}{b(v')} dv' = \frac{v}{b(v)} \quad (1.31)$$

Nyní můžeme přepsat podmínku bezpečnosti do tvaru

$$v_1 - v_f \geq \frac{v_1\tau - g}{\frac{\bar{v}}{b(\bar{v})} + \tau}, \quad (1.32)$$

kde

$$v_1 - v_f = \frac{dg}{dt}. \quad (1.33)$$

Nerovnice (1.32) se v podstatě rovná nerovnici (1.27), kde preferovaný odstup $g_{\text{des}} = v_1\tau$ a preferovaný bezpečný časový odstup mezi vozidly $\tau_{\text{des}} = \tau_b + \tau$, kde $\tau_b = \bar{v}/b(\bar{v})$ je stanoveno na základě decelací, které řidič využije.

1.3.3 Diskretizace modelu

Nerovnice v části 1.3.1 je potřeba převést na diskrétní rovnice pro dráhy, rychlosti a zrychlení pro možnost simulace na počítači.

Vhodná cesta pro vytvoření rovnic dynamického systému z bezpečnostní podmínky (1.27) je převést spojitou rychlost v_f ve výrazu $\dot{g}(t) = v_1(t) - v_f(t)$ jako rychlost diskretní s diferencí či integračním krokem $t + \Delta t$. Potom rovnici (1.27) přepíšeme do tvaru

$$v_f(t + \Delta t) \leq v_1(t) + \frac{g(t) - g_{\text{des}}}{\tau_{\text{des}}}. \quad (1.34)$$

Rovnice pro polohu vozidla se potom na stejném principu převede na

$$x(t + \Delta t) = x(t) + v(t + \Delta t)\Delta t. \quad (1.35)$$

Víme, že při $\Delta t \rightarrow 0$ a $g_{\text{des}} \geq 0$ rovnice (1.34) a nerovnice (1.35) garantují bezpečnost. Pro konečný čas Δt musí být tato garance opět dokázána.

Mezera $g(t)$ mezi dvěma vozidly je diskretizována vztahem

$$g(t + \Delta t) = g(t) + \Delta t(v_1(t + \Delta t) - v_f(t + \Delta t)). \quad (1.36)$$

Přidáním rovnice (1.34) dostáváme

$$\xi(t + \Delta t) = \xi(t) + \left(1 - \frac{\Delta t}{\tau_b + \tau}\right) + \Delta t \frac{g_{\text{des}} - v_1(t)\Delta t}{\tau_b + \tau}, \quad (1.37)$$

kde

$$\xi(t) = g(t) - v_1(t)\Delta t.$$

Bezpečnost ($g \geq 0$) je tedy garantována tehdy, když platí:

$$\begin{aligned} \xi(t = 0) &\geq 0, \\ \Delta t &\leq \tau, \\ g_{\text{des}} &\geq v_1\Delta t. \end{aligned} \quad (1.38)$$

To je výsledek, který se ovšem dal předpokládat. Záleží pouze na tom, zda je i po provedení iteračního kroku stále splněna podmínka bezpečnosti, tj. kontrolovat jestli je náš iterační krok menší než reakční čas řidiče τ .

1.3.4 Rovnice modelu

Předpokládáme tedy, nezávisle na náhodném kolísání, že každé vozidlo se pohybuje nejvyšší rychlostí na základě podmínek, které jsou stanoveny níže. **Kraussův model tedy stanovíme následujícími rovnicemi:**

$$\begin{aligned} v_{\text{safe}}(t) &= v_1(t) + \frac{g(t) - g_{\text{des}}}{\tau_b + \tau}, \\ v_{\text{des}} &= \min[v_{\text{max}}, v(t) + a(t)\Delta t, v_{\text{safe}}(t)], \\ v(t + \Delta t) &= \max[0, v_{\text{des}} - \eta]. \end{aligned} \quad (1.39)$$

Preferovaná mezera g_{des} může být zvolena různě. Většinou se volí $g_{\text{des}} = v_1\tau$, kde τ je reakční čas řidiče. Hodnota τ_b je definována jako podíl \bar{v}/b , kde \bar{v} plyne ze vztahu (1.29). Nově zde také zavádíme proměnnou η , která je náhodnou výchylnou od běžného řízení. Volba integračního kroku Δt a preferované mezery g_{des} je závislá na podmínkách (1.38).

2. Intelligent Driver Model

V této kapitole budeme primárně vycházet z práce [17], ve které je Intelligent Driver Model (IDM) do značné míry rozebrán. Využijeme hlavně podkapitolu 5.1 z [17], kterou doplníme o poznatky, které jsou uplatněny z kapitoly 1.

IDM model patří mezi novější. První článek o tomto model vyšel v roce 2000,

2.1 Matematický popis

Pro výpočet zrychlení v IDM je použit vztah

$$\dot{v}_n = a_{\text{free}} + a_{\text{int}}, \quad (2.1)$$

kde se aktuální zrychlení účastníka vypočítá jako součet zrychlení při volném proudu a_{free} a zrychlení, ve kterém je zahrnut i účastník jedoucí před ním a_{int} . Zde je vidět určitá paralela s Gippsovým modelem (kapitola 1.2, kde se ovšem vybírá pouze jedna z těchto dvou hodnot. Navíc i rovnice se od IDM odlišují. Pro volný proud je tedy zrychlení $a_{\text{int}} = 0$.

Zrychlení při volném proudu je definováno vztahem

$$a_{\text{free}} = a \left[1 - \left(\frac{v_n}{v_{n\text{free}}} \right)^\delta \right], \quad (2.2)$$

kde a je maximální zrychlení, δ je exponent zrychlení, v_n je aktuální rychlost vozidla a $v_{n\text{free}}$ je rychlost, které chce vozidlo dosáhnout.

Pokud v rovnici (2.2) bude zlomek v_n/v_{free} roven jedné, auto se již dostalo na svou preferovanou rychlost v_{free} a celkové zrychlení a_{free} bude rovno nule. Naopak pokud bude v_n rovno nule, znamená to, že a_{free} bude přímo rovno a , a auto pojede svým maximálním zrychlením. Pokud v_n bude menší než v_{free} , pak platí že, čím vyšší hodnotu bude mít nastavenou parametr δ , tím se dosáhne vyšší hodnota a_{free} . Pokud v_n bude větší než v_{free} , bude platit pravý opak.

Auto jedoucí při volném proudu je ale nereálná představa, proto se do rovnice (2.1) přidává interakční člen, který přímo závisí na parametrech vozidla jedoucí před naším účastníkem. Tento člen je roven

$$a_{\text{int}} = -a \left(\frac{s_n^*}{s_n} \right)^2, \quad (2.3)$$

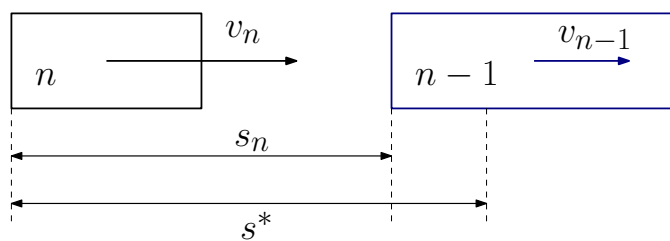
kde a je maximální zrychlení, $s_n = x_{n-1} - x_n$ je rozdíl poloh vozidel a s_n^*

$$s_n^* = s_0 + v_n T + \frac{v_n(v_n - v_{n-1})}{2\sqrt{ab}} \quad (2.4)$$

je preferovaný odstup. s_0 je součet minimální bezpečné vzdálenosti mezi vozidly a délkou následujícího vozidla, T je bezpečnostní časový odstup mezi vozidly a b je konstanta brzdění. Situace je schématicky znázorněna na obrázku 2.1.

Člen úměrný $v_n - v_{n-1}$ je kvadratický v proměnné v_n a obsahuje součin $v_n v_{n-1}$. Tyto nelineární členy popisují interakci, v tomto případě dvou vozidel. Pokud tedy bude vzdálenost s_n mezi auty vysoká, potom zlomek s_n^*/s_n bude mít velmi malou hodnotu a interakční člen a_{int} lze zanedbat, takže vozidlo se pohybuje jako ve volném proudu.

Pokud rychlost auta v_n bude vyšší než rychlost auta v_{n-1} , pak interakční člen s_n^* bude kladný a požadovaný odstup se zvětší, takže zrychlení auta \dot{v}_n se musí snížit.



Obrázek 2.1: Proměnné jednotlivých aut

Literatura

- [1] Barceló J, *Fundamentals of traffic simulation*, proceedings, International Series in Operations Research and management science, Springer, 2010
- [2] Barrow J, *Nové teorie všeho*, Dokořán, Praha, 2008
- [3] Brdička M, Samek L, Sopko B, *Mechanika kontinua*, Academia, Praha, 2011
- [4] Gipps, P. G. (1981). A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological*, 15(2), 105-111.
- [5] Haberman R, *Mathematical models: Mechanical vibrations, population dynamics and traffic flow*, Society for Industrial and Applied Mathematics, Philadelphia, 1988
- [6] Helbing D, Herrmann H J, Schreckenberg M, Wolf D E, *Microscopic Simulation of Congested Traffic* v knížce *Traffic and Granular Flow*, Springer, Berlin, 2000
- [7] Horák J, Krlín L, Raidl A, *Deterministický chaos a jeho fyzikální aplikace*, Academia 2003
- [8] KRAUSS, Stefan. *Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics*. Köln, 1998. PhD. Thesis. Universität Köln.
- [9] Xiaoliang Ma, *A Neural-Fuzzy Framework for Modeling Car-following Behavior* [online], dostupné z http://www.ctr.kth.se/publications/ctr2006_08.pdf
- [10] May A D, *Traffic flow fundamentals*, Prentice Hall, 1989
- [11] Newell, G. F. (2002). A simplified car-following theory: a lower order model. *Transportation Research Part B: Methodological*, 36(3), 195-205.
- [12] Příkryl P, *Numerické metody matematické analýzy*, SNTL, Praha, 1988
- [13] Scholtz M, *Classical mechanics and deterministic chaos* [online], dostupné z <http://www.fd.cvut.cz/personal/scholma1/>
- [14] Scholtz M, Vaniš M, Veselý P, Matějka P, *Applied mathematics on Faculty of Transportation Sciences*, vyjde ve sborníku k výročí Fakulty dopravní ČVUT
- [15] Treiber M, Hennecke A, Helbing D, *Congested traffic states in empirical observations and microscopic simulations*, Physical Review E, **62** (2), pp. 1805–1824, 2000
- [16] Treiber M, *Microsimulation of road traffic flow* [online], dostupné z <http://www.traffic-simulation.de/>
- [17] VANIŠ, Miroslav. *Matematické modelování vybraných problémů v dopravě v jazyce Java*. Praha, 2013. Bakalářská práce. FD ČVUT.
- [18] Vitásek E, *Numerické metody*, SNTL, Praha, 1987

1. Příloha A: Kompilace simulačního softwaru SUMO

1.1 SUMO

SUMO je volně šiřitelný dopravní simulátor, který vznikl v roce 2001. SUMO umožňuje modelování intermodálních dopravních systémů. Ty mohou obsahovat dopravní prostředky, městskou hromadnou dopravu či chodce. Dále v SUMO existují nástroje pro vyhledávání cesty, visualizace, import či export map, počítání emisí a další. Nejdůležitější vlastností pro nás je možnost implementování vlastního dopravního modelu. SUMO samo o sobě již obsahuje některé modely - Kraussův model 1.3, Intelligent Driver Model 2 a Wiedermanův model.

1.1.1 SUMO download

Na stránkách projektu SUMO¹ se stáhne aktuální verze SUMO na pravé straně. Uvažujeme pouze případ dopravního simulátoru SUMO pod Windows, v době vzniku byla na světě verze SUMO 0.22. Ideální je varianta stáhnutí tří zkomprimovaných souborů. Pro potřeby kompilace stačí pouze soubor SUMO 0.22 sources (sumo-src-0.22.0, 14 MB). V souboru SUMO 0.22 manual, tutorial and examples (sumo-doc-0.22.0.zip, 35 MB) se nachází podrobná dokumentace ke všem naprogramovaným součástem SUMO a SUMO 0.22 for Windows (sumo-winbin-0.22.0.zip, 33 MB) obsahuje již zkompilevané SUMO, které je možné rovnou použít. V tomto souboru se ve složce *docs/userdoc* nachází uživatelská příručka, která obsahuje i návod na zkompilevání SUMO pod Windows. Tento návod ovšem v některých částech není dostatečně přesný.

1.2 Visual Studio

Pro zkompilevání SUMO je zapotřebí mít nainstalované Visual Studio. To je možné získat přímo na stránkách Visual Studia² a po zaregistrování je možno získat Express edici Visual Studia. Autor pracuje s Visual Studio 2010 Pro. Kompilace SUMO je tedy vyzkoušena na této verzi. **Pro správné fungování Visual Studia je nutné mít nainstalován Microsoft .NET Framework verze 4 ne vyšší!**

1.3 Ostatní závislosti

SUMO ke svému zkompilevání potřebuje další knihovny. V první řadě je to **Xerces-C** (xerces-c-3.1.1-x86-windows-vc-10.0.zip, 23 MB)³, které už je samo o sobě zkompilevané. Podstatné je stáhnout verzi x86 a ne 64bitovou, aby s ní byly schopni další součásti pracovat.

Další závislostí jsou **knihovny FOX**, kterou si budeme muset ve Visual Studiu zkompilevat sami. V návodu u SUMO je uvedeno, že je vyzkoušena pouze verze 1.6.36⁴. Přesně pro tuto jedinou verzi se mi povedlo provést kompilaci v pořádku.

¹<http://www.dlr.de/ts/sumo/en>

²<http://www.visualstudio.com>

³<http://mirror.hosting90.cz/apache//xerces/c/3/binaries/xerces-c-3.1.1-x86-windows-vc-10.0.zip>

⁴<http://ftp.fox-toolkit.org/pub/fox-1.6.36.zip>

Poslední částí pro zdárnou kompilaci jsou FWTools. Verze doporučená SUMO ovšem nefunguje a ani nikde na internetu se jí nepodařilo nalézt. Funkční je ovšem verze 2.4.7⁵.

Důrazně se doporučuje ani jednu z těchto částí neinstalovat do klasického adresáře Program Files, může to způsobit problémy.

Nutné je také mít nainstalovaný Python (pro něj neplatí předchozí odstavec), ovšem pozor rozhodně ne nejnovější verzi⁶, nýbrž verzi 2.7⁷. U Pythonu není nutné žádné další nastavení.

1.3.1 Xerces-C

Nejprve rozbalíme obsah archivu a posléze překopírujeme ze složky *bin* obě dll knihovny⁸ do složky *Windows/System32*. Přes veškerou snahu se nepodařilo XERCES naimportovat všem projektům. Tím pádem se musí celý obsah složky *include* překopírovat do složky, kde je nainstalované Visual Studio do složky *vc/include*⁹. Výsledkem tedy bude překopírovaná složka *Xercesc* ze složky *Xerces/include/* do složky */vc/include*. Pro kompilaci a následné spuštění zkompilevaných programů je nutno vytvořit systémovou proměnnou XERCES¹⁰, jejíž hodnota bude nastavena na kořenový adresář Xerces. Poté vyhledáme v systémových proměnných proměnnou PATH a nakonec hodnoty této proměnné přidáme %XERCES%/bin/.

1.3.2 FOX

FOX klasicky rozbalíme z archivu a poté musíme Visual Studiem zkompilevat. Otevřeme v rozbaleném archivu složku *windows/vcpp* a v ní otevřeme Visual Studiem soubor *win32.dsw*. Dle verze Visual Studia se může objevit hlášení o nutnosti konvertování. Potvrdíme pro všechny (možno zaškrtnout) a necháme Visual Studio načíst FOX. Po načtení musíme vybrat možnost kompilace **Release** a zkompileujeme **pouze** projekt foxdll. Pak provedeme to samé s tím rozdílem, že vybereme možnost **Debug**. Při kompilaci můžou vzniknout chyby, ty nás ovšem netrápí, protože to důležité by se mělo vytvořit. V kořenovém adresáři se vytvoří složka *lib* ve které se nacházejí zkompilevané knihovny FOXDLL-1.6.dll a FOXDLLD-1.6.dll, které také překopírujeme do složky *Windows/System32*. Do systémových proměnných je nutné přidat proměnnou z názvem FOX16 s cestou do kořenového adresáře FOX¹¹ a následně na konec hodnoty systémové proměnné PATH přidat %FOX16%/lib.

1.3.3 FWTools

FWTools klasicky nainstalujeme (ideálně ne do složky Program Files). Vytvoříme stejně jako pro FOX systémovou proměnnou s názvem PROJ_GDAL, která bude opět odkazovat do kořenového adresáře FWTools. Poté vyhledáme v systémových proměnných proměnnou PATH a nakonec hodnoty této proměnné přidáme %PROJ_GDAL%/bin.

1.3.4 SUMO

V kořenovém adresáři SUMO ve složce */build/msvc10* se nachází soubor s názvem *prj.sln*. Ten otevřeme Visual Studiem. Pokud je vše nastaveno správně, je možno spustit kompilaci

⁵<http://home.gdal.org/fwtools/FWTools247.exe>

⁶3 a výše

⁷<https://www.python.org/ftp/python/2.7.8/python-2.7.8.msi>, 16 MB

⁸xerces-c_3-1.dll a xerces-c_3_1D

⁹např. c:/Program Files/Microsoft Visual Studio 10.0/VC/include/

¹⁰Start - Počítač - Vlastnosti systému - Upřesnit nastavení systému (nabídka na pravé straně) - Proměnné prostředí... - Systémové proměnné - Nová...

¹¹např. C:/fox-1.6.36/

SUMO, to znamená zkompilovat všechno (všech 54 projektů). Pokud kompilujeme v režimu Release, dostaneme výstup, který se nachází v kořenovém adresáři ve složce *bin*. Tento výstup je v `sumo-winbin-0.22.0.zip`.