



Riešenie  
Projekt z predmetu SIN  
Zadanie: Dopravní telematika

8. prosince 2015

Autor: Martin Maga (xmagam00), Vojtěch Meca (xmecav00) - vedúci tímu,  
Fakulta Informačních Technologií  
Vysoké Učení Technické v Brně

# Obsah

<b>1</b>	<b>Riešenie</b>	<b>2</b>
1.1	Zadání . . . . .	2
<b>2</b>	<b>Návrh a cieľ riešenia</b>	<b>2</b>
<b>3</b>	<b>Implementácia</b>	<b>2</b>
<b>4</b>	<b>Štatistiky križovatky</b>	<b>3</b>
<b>5</b>	<b>Agenti</b>	<b>3</b>
<b>6</b>	<b>Rozšírenia</b>	<b>4</b>
<b>7</b>	<b>Záver</b>	<b>5</b>

# 1 Riešenie

## 1.1 Zadání

Zadanie spočíva v tvorbe modelu križovatky v nami zvolenom simulačnom frameworku. Cieľom je model križovatky, ktorý bude riešiť problematiku dopravnej telekinematiky tj možné problémy a riešenie, ktoré by mohlo riešiť problémy z praxe.

## 2 Návrh a cieľ riešenia

Cieľom projektu je automatizácia križovatky. Uvažujeme jednoduchú križovatku so štyrmi smermi SEVER, JUH, VÝCHOD A ZÁPAD. Činnosť križovatky nei je závislá na nejakej časovej perióde tj. križovatka je prevádzka cez deň aj v noci. Počas simulácia sbierame štatistiky príchodu aut, ktoré sú náhodné generované v každom smere, rovnako aj stav semaforom a počet aut pred jednotlivými semaformi, tj vo frontách.

## 3 Implementácia

Projekt je implementovaný v Jave s využitím simulačného frameworku JADE, čo je platforma založená na jazyku Java.

Jadro projektu tvorí model križovatky v tvare X a križovatka obsahuje 2 semaforey, pričom autá prichádzajú v každom smere.

Beh modelu je založený na náhodnom generovaní aut v každom smere, pričom pred každým semaforom sa tvorí istá fronta, ktorá má svoju maximálnu dĺžku. Semaforey sú riadené stochastickým procesom, pričom každý semafor má informáciu o tom koľko aut aktuálne pred ním čaká. Tieto informácie sa používajú k vyhodnoteniu situácie a spusteniu smerom, v ktorom je väčší počet aut.

V projekte je striktne dodržaná agentná architektúra, ktorá znamená, že jednotliví agenti sú vzájomne nezávislí a komunikujú prostredníctvom zvolených správ.

CreatorAgent predstavuje hlavného agenta, ktorý obsluhuje aj ostatných agentov. Tento agent je zodpovedný za spustenie behu celej aplikácie a generovanie aut pre každý semafor (agenta). Zároveň pri generovaní posiela informácie o nagentovaných autách jednotlivým agentom. Títo agenti (RouteAgent) spracovávajú prijaté správy o pridání nových aut a posielajú informácie späť hlavnému agentovi. Hlavný agent (CreatorAgent) podľa prijatých informácií od jednotlivých agentov rozhoduje, ktorý semafor bude mať zelenú a ktorý červenú. Zároveň prebieha logovanie na úrovni konzole, kde sú zobrazované stavové a súhrné informácie o autách a aktuálnom stave semaforov.

## 4 Štatistiky križovatky

Hlavným cieľom aplikácie je okrem funkčnosti je poskytnúť aktuálnu situáciu o stave frontách pre jednotlivé semafore (agentov). Do konzole sa pravidelne zobrazujú informácie nasledovného formátu:

*nazov agenta – stav Semaforu – smer – 1.auto – veľkosť fronty – smer – 1.auto – veľkosť fronty*  
(1)

Príklad:

*route1 – GREEN – N – 2 – 3 – S – 4 – 5*  
(2)

Vyššie uvedený príklad odpovedá výstupu fronty pre 1. agenta obsluhujúci semafor zo severu a juhu križovatky. Pričom vieme povedať, že 1. auto na križovatke zo severu má ID 2 a na semafore čakajú celkovo 3 autá a z má 1. auto na semafore ID 4 a na semafore čaká celkovo 5 aut. Všetky tieto správy sú spracované zároveň hlavným agentom (CreatorAgent), ktorý tieto správy rozparsuje a rozhodne činnosti križovatky, tj v ktorých smeroch budú pustené autá.

## 5 Agenti

- CreatorAgent - Hlavný agent, ktorý sa stará o spustenie celého modelu a zároveň generovanie aut a zároveň prijíma správy od agentov na semaforochoch a rozhoduje, ktoré semafore budú pustené
- RouteAgent - Agent, ktorý prijíma správy od Creator agenta zaraďuje autá do jednotlivých interných front a stavu front sa posielajú CreatorAgent-ovi

## 6 Rozšírenia

Aplikácia obsahuje len konzolový výstup, ktorý je trochu obmedzujúci. Preto ako možné rozšírenie môže aplikácia obsahovať GUI, ktoré bude zbierať štatistické výstupy a vykreslovať GUI. Teda bude obsahovať semaforey, autá pred semaformi.

Ďalšie možné rozšírenie môže predstavovať striedanie dňa a noci, ktorý by mohlo byť realizované ďalším agentom, ktorý bude informovať ostatných agentov. Toto bude mať za následok, že počas dňa budú semaforey pustené a počas noci budú vypnuté.

Rovnako by náš model mohol obsahovať viac príjazdových pruhov pred semaformi, ktoré by bolo následne obsluhované agentom na semafore.

Rovnako by semafor, v prípade, že by aplikácia bola vybavená GUI zobrazovať počet aut, ktoré križovatkou prejdú a spôsob rozhodovania.

Rovnako by užívateľ mohol sám v nejakom formáte vložiť vlastné pravidlá, ktoré by mohli ovplyvňovať celkové ovládanie križovatky.

## 7 Záver

Poradilo sa nám vytvoriť správny a korektný model simulujúci križovatku. Modal obsahuje niekoľko semaforov, pričom pred každým semaforom sa tvorí fronta áut, pričom autá do tejto fronty sú náhodne generované a následne sú autá zaradované do front. Vizualizácia prebieha zapisovaním správ do konzole. Možné vylepšenie predstavuje tvorba GUI pre lepšiu vizualizáciu stavu križovatky.

V projekte sme dodržali agentnú architektúru. Agenti nie sú na sebe závislí a nemajú pevné väzby a komunikácia prebieha prostredníctvom správ.