

Řízení pohybu stanice v simulačním prostředí OPNET Modeler podle mapového podkladu

Map-based mobility control system for wireless stations in OPNET Modeler

Peter Jakúbek, Jiří Hošek

xjakub00@stud.feec.vutbr.cz, hosek@feec.vutbr.cz

Fakulta elektrotechniky a komunikačních technologií VUT v Brně.

Abstrakt: Článek se zabývá problematikou řízení pohybu mobilní stanice v simulačním prostředí OPNET Modeler. V článku jsou shrnuty a popsány stávající možnosti řízení pohybu bezdrátové mobilní stanice. Pro zpracování mapového podkladu je použit nástroj MATLAB. Zpracování podkladu provádí vytvořená funkce pro generování trajektorie. Výstup funkce je následně použitý pro řízení pohybu bezdrátové stanice v OPNET Modeleru. Pro zautomatizování procesu řízení pohybu stanice je vytvořeno speciální propojení MATLABu s prostředím OPNET Modeler pomocí knihovny MATLAB Engine. Tato knihovna umožňuje volání MATLAB funkcí z jazyka C/C++, na kterém je založený simulační nástroj OPNET Modeler.

Abstract: This paper introduces the map-based mobility control system for mobile stations created in the simulation environment OPNET Modeler. The current possibilities for mobility control in OPNET Modeler are described in the paper. The MATLAB tool is used for the map source processing. Both simulation environments were linked together by the MATLAB Engine library which allows external programs to access MATLAB functionalities. This library uses the C/C++ functions for the data exchange between two communication systems. The proper function of system designed was evaluated by a simulation.

Řízení pohybu stanice v simulačním prostředí OPNET Modeler podle mapového podkladu

Peter Jakúbek¹, Jiří Hošek²

Fakulta elektrotechniky a komunikačních technologií, VUT v Brně
Email: xjakub00@stud.feec.vutbr.cz¹, hosek@feec.vutbr.cz²

Abstrakt – Článek se zabývá problematikou řízení pohybu mobilní stanice v simulačním prostředí OPNET Modeler. V článku jsou shrnuty a popsány stávající možnosti řízení pohybu bezdrátové mobilní stanice. Pro zpracování mapového podkladu je použit nástroj MATLAB. Zpracování podkladu provádí vytvořená funkce pro generování trajektorie. Výstup funkce je následně použitý pro řízení pohybu bezdrátové stanice v OPNET Modeleru. Pro zautomatizování procesu řízení pohybu stanice je vytvořeno speciální propojení MATLABu s prostředím OPNET Modeler pomocí knihovny MATLAB Engine. Tato knihovna umožňuje volání MATLAB funkcí z jazyka C/C++, na kterém je založený simulační nástroj OPNET Modeler.

1 Úvod

Bezdrátové počítačové sítě se v dnešní době stávají součástí života běžných uživatelů. Jedním z typů bezdrátových počítačových sítí jsou MANET (Mobile Ad-hoc Networks) sítě. Neboť se jedná o komunikační systém tvořený mobilními stanicemi, jedním z důležitých faktorů při simulaci těchto sítí je možnost řízení pohybu mobilních stanic. Bezdrátová komunikace umožňuje při zachování funkcionality stanic a podsítí měnit jejich polohu založenou na předdefinovaných trajektoriích, orbitách a náhodně vybraných cestách.

Možnost řízení pohybu stanice umožňuje efektivnější predikci a plánování síťových zdrojů pro jednotlivé stanice při optimalizaci handoveru v MANET sítích. Článek je zaměřený na řízení pohybu stanice pomocí přímé manipulace pozice mobilní stanice.

2 Možnosti řízení pohybu stanice v simulačním prostředí OPNET Modeler

Trajektorie je důležitý parametr při implementování mobilních rádiově orientovaných technologií. Řízení pohybu stanic je klíčovým pro simulaci komunikačních sítí. Při bezdrátové komunikaci se mění poloha stanic, která je založena na předem definovaných trajektoriích, anebo náhodně vybraných cestách. Poloha mobilní stanice je automaticky aktualizovaná během simulace a tím je tedy modelován souvislý pohyb.

V OM (OPNET Modeler) je možné modelovat pohyb definováním trajektorií a orbit, anebo přímou manipulací s hodnotami pozic [1]. Při simulování může být pro každý jednotlivý případ mobilní stanice použitý jenom jeden z těchto

mechanizmů. Pokud není definovaná trajektorie pohybu, je možné přímo měnit pozici mobilní stanice pomocí nějakého procesu. Stanice musí začít její pohyb v rámci svojí podsítě a pak se může pohybovat do jiné podsítě.

V OM je několik možností pro řízení pohybu stanice. Patří sem:

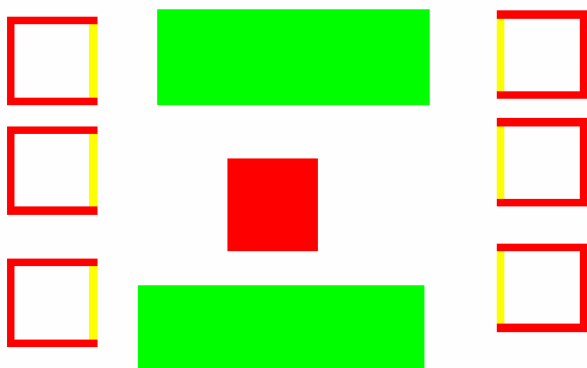
- **Konfigurace náhodné mobility** – jedná se o implementaci řízení pohybu stanice na základě konfigurace náhodné mobility pomocí objektu „Mobility config“. Pomocí tohoto objektu je možné vytvářet profily, které mají nastavené různé parametry náhodné mobility a následně je přiřadit mobilním stanicím.
- **Konfigurace trajektorie** – doplnění možnosti řízení pohybu stanice pomocí náhodné mobility. Konfigurace náhodné mobility je doplněná o možnost řízení z externího souboru obsahujícího trajektorii pohybu. Nejedná se tedy o náhodnou trajektorii definovanou uživatelem. Při tomto typu řízení se jedná o spojitý pohyb stanice. Pohyb je řízený simulačním jádrem OM. Externí soubory určené pro řízení pohybu stanice je možné vytvořit ručně, anebo je použít pro uložení vygenerované náhodné trasy pro pozdější použití.
- **Přímá manipulace pozice stanice** – přímé manipulování pozice stanice pomocí procesu. Jedná se o alternativu řízení pohybu stanice podle souboru s trajektorií. Přímá manipulace pozice stanice poskytuje nespojitý pohyb stanice. O řízení se stará uživatelem vytvořený proces, který používá diskretní hodnoty trajektorie.

3 Mapový podklad v MATLABu

Mapový podklad je zpracovaný do takové podoby, aby byl použitelný pro řízení pohybu stanice v prostředí OM. Byla vytvořena funkce v nástroji MATLAB, která zpracuje bitmapový obrázek jako mapový podklad [2, 3]. Vzorová podoba vstupního obrázku je uvedena na obrázku 1.

Funkce detekuje objekty nacházející se v mapovém podkladu. Následně je objektům přiřazena pravděpodobnost vstupu na jejich pole. Funkce rozeznává 4 barvy, kterým přiřazuje předem definované pravděpodobnosti vstupu. Na obrázku 1 jsou červenou barvou znázorněny stěny budov, které jsou neprůchodné. Žlutá barva reprezentuje vchody do budov

s nastavenou pravděpodobností vstupu. Zelené plochy reprezentují travnaté plochy anebo parky s malou pravděpodobností vstupu. Plocha s bílou barvou reprezentuje komunikaci, která má maximální pravděpodobnost vstupu. Na základě zpracovaného vstupního obrázku a vstupních proměnných funkce se vygeneruje trajektorie, která se uloží do výstupního souboru s vhodným formátem hlavičky.

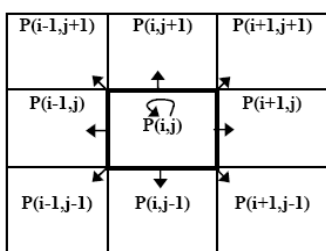


Obrázek 1: Vstupní obrázek

3.1 Popis pohybu po mapě

Funkce má implementované dvě možnosti pohybu. První možnost je volný pohyb, který je omezený pouze vstupní mapou. Další možností je použití vyhledávání hotspotů [4].

Pohyb stanice je ve vytvořené MATLAB funkci implementovaný pomocí Markovovi mřížky, viz. obrázek 2 [4]. Jedná se o matici pravděpodobností přechodů mezi konečným počtem stavů.



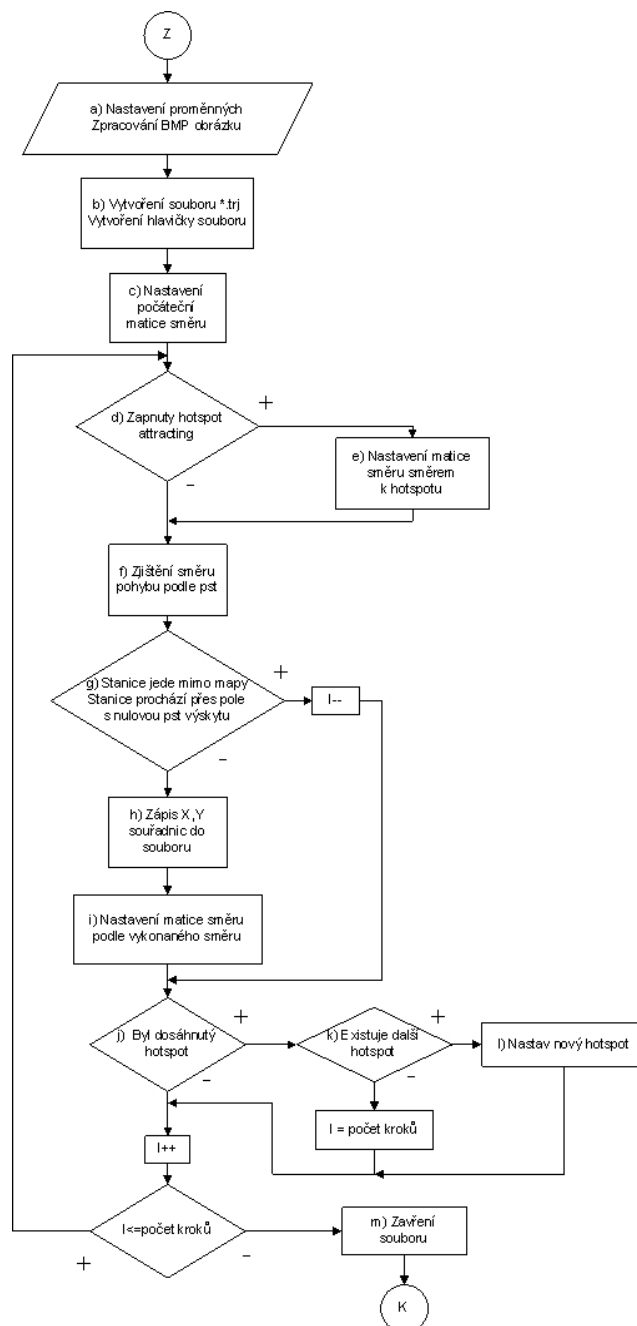
Obrázek 2: Markovova mřížka

Každé pole má svoji pravděpodobnost výskytu nazývanou váha. V prvním kroku se určí směr na základě pravděpodobnostních políček. Následně se vykoná pohyb v určeném směru a nastaví se aktuální mřížka na novou. Nová mřížka bude mít přizpůsobené váhy políček tak, aby byl v případě volného pohybu upřednostňovaný pohyb ve směru předcházejícího pohybu. V případě použití vyhledávání hotspotů bude upřednostněný pohyb ve směru k vyhledávanému hotspotu. Pro každý z devíti směrů je nadefinována vlastní mřížka s váhami. Matice s nastavenými váhami pro směr doprava je definována podle vztahu (1).

$$\text{mat_R} = \begin{bmatrix} 0 & 0,03 & 0,03 \\ 0 & 0,01 & 0,79 \\ 0 & 0,03 & 0,03 \end{bmatrix}; \quad (1)$$

Z hodnot pravděpodobností je vidět preferovaný směr doprava. Cyklus pohybu se opakuje, dokud se nevykoná definovaný počet kroků. V případě použití vyhledávání hotspotů se cyklus ukončí po dosažení posledního nastaveného hotspotu.

Na obrázku 3 je zobrazený vývojový diagram funkce v MATLABu pro generování souborů s trajektorií.



Obrázek 3: Vývojový diagram funkce pro generování souborů s trajektorií

3.2 Vstupy a výstupy funkce

Funkce zpracovává vstupní obrázek a generuje trajektorii podle kritérií, kterými jsou vstupní proměnné funkce. Vstupní a výstupní proměnné jsou uvedené v tabulce 1 a 2. Funkce je nadefinována následovně:

```
function [x_coord,y_coord,pocet] = zapis_variable_func(Xp,Yp,Pkrok,X0,Y0,Hpoc,Rpol,Hatr,Id);
```

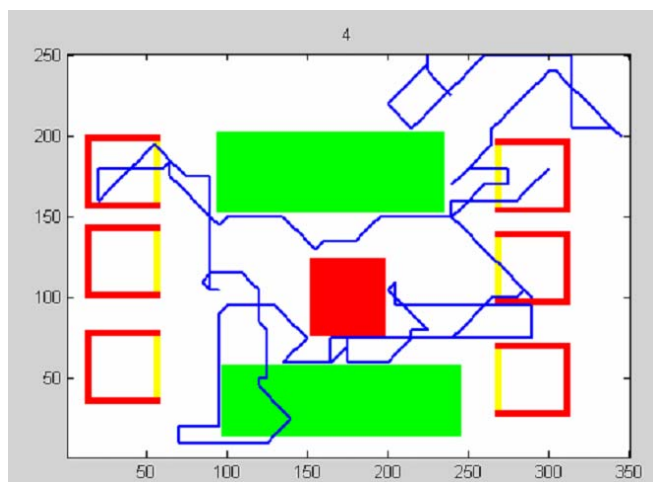
Tabulka 1: Vstupní proměnné funkce

Proměnná	Popis
Xp	počáteční X souřadnice
Yp	počáteční Y souřadnice
Pkrok	maximální počet kroků pro generování souřadnic
X0	vektor X souřadnic hotspotů
Y0	vektor Y souřadnic hotspotů
Hpoc	počet hotspotů
Rpol	poloměr dosáhnutí hotspotu
Hatr	atribut zapnutí vyhledávání hotspotů
Id	identifikační číslo stanice, které slouží pro identifikaci výstupních obrázků

Tabulka 2: Výstupní proměnné funkce

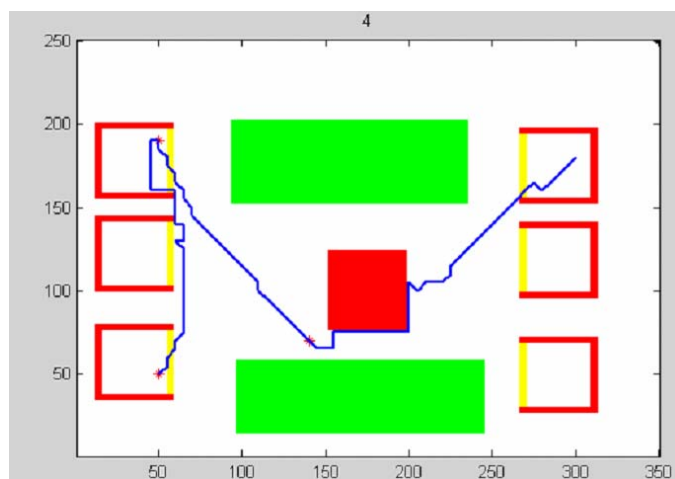
Proměnná	Popis
x_coord	vektor s hodnotami vygenerovaných X souřadnic
y_coord	vektor s hodnotami vygenerovaných Y souřadnic
pocet	počet vygenerovaných hodnot X a Y souřadnic

Tyto výstupní proměnné mají význam pro přímé řízení pozice stanice, kde se jejich hodnoty předávají z MATLABu do OM. Výstupem funkce je soubor s trajektorií použitelný pro další zpracování. Pro grafické zobrazení trajektorie slouží výstupní obrázek, který se skládá z vstupního obrázku a vygenerované trajektorie. Na obrázku 4 je zobrazený výstupní obrázek s vygenerovanou náhodnou trasou bez použití funkce vyhledávání hotspotů. Vygenerovaná trasa je znázorněna na obrázku a uložena v souboru s příponou „.trj“.



Obrázek 4: Výstupní obrázek bez Hotspotů

Na obrázku 5 je zobrazený výstupní obrázek při zapnutí funkce vyhledávání hotspotů. Startovní pozice se nachází vpravo nahoře. První hotspot se nachází uprostřed dole. Po jeho dosáhnutí se nastaví jako aktivní další hotspot, který se nachází vlevo nahoře. Poslední hotspot se nachází vlevo dole. Hotspoty jsou dosaženy pouze v případě, že se mezi nimi a aktuálními souřadnicemi nenachází slepá cesta. To se může stát v případě, když stanice vstoupí na pozici, která je z více než jedné strany ohraničena neprůchozími objekty a zároveň se další hotspot nachází za těmito objekty.



Obrázek 5: Výstupní obrázek s použitím hotspotů

4 MATLAB Engine

Pro volání funkcí nástroje MATLAB z externích programů napsaných v jazyce C/C++ slouží knihovna MATLAB Engine. Pro použití knihovny Engine musí být nainstalovaný kompletní software MATLAB, není totiž možné spouštět MATLAB Engine funkce pouze s použitím MATLAB kompilátoru. Engine funkce jsou nezávislé na programech v jazyce C/C++, které odděleně komunikují s MATLAB procesem přes komunikační rozhraní. MATLAB poskytuje knihovnu funkcí, které umožňují spustit a ukončit MATLAB proces, zaslat data do MATLABu a zpět a zaslat příkazy pro zpracování v MATLABu.

4.1 MATLAB Engine knihovna

V tabulce 3 jsou uvedené funkce knihovny MATLAB Engine pro použití v jazyce C/C++.

Tabulka 3: Funkce knihovny MATLAB Engine

Funkce	Účel
engOpen	spuštění MATLAB Engine
engClose	zastavení MATLAB Engine
engGetVariable	získání hodnot z rozhraní Engine
engPutVariable	zapsání hodnot na rozhraní Engine
engEvalString	vykonání MATLAB příkazu
engOutputBuffer	vytvoření bufferu pro uložení

	textového výstupu z MATLABu
engOpenSingleUse	spuštění MATLAB engine sekce pro jednoduché nesdílené použití
engGetVisible	zjištění viditelnosti sekce MATLAB Engine
engSetVisible	nastavení viditelnosti sekce MATLAB Engine

Příklady použití Engine příkazů v C/C++ programu:

- Spuštění MATLAB Engine:

```
ep = engOpen(NULL);
```

- Vložení hodnoty proměnné „XP“ z jazyka C++ do proměnné „A“ v MATLABu:

```
engPutVariable(ep, "A", XP);
```

- Vykonání příkazu v MATLABu, konkrétní vykonání funkce s názvem „zapis_variable_func“ s jejími vstupními a výstupními hodnotami:

```
engEvalString(ep, "[X_COORD, Y_COORD, P]=zapis_variable_func(A, B, C, D, E, F, G)");
```

- Získání hodnoty proměnné „X_COORD“ z MATLABu a vložení do proměnné „XS“ v jazyce C++:

```
XS = engGetVariable(ep, "X_COORD");
```

5 Projekt s automatizovaným procesem

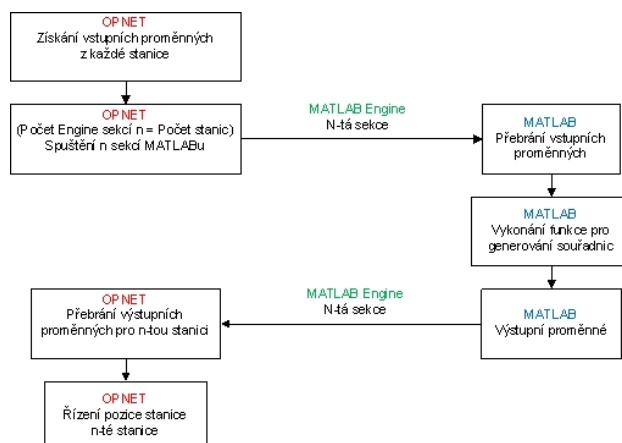
Pozici stanice je možné řídit z externího zdroje souřadnic pomocí přiřazení vytvořené trajektorie stanici, anebo přímou manipulací pozice stanice.

Způsob řízení pozice mobilní stanice pomocí externích souborů neumožňuje celý proces řízení zautomatizovat tak, aby nebyl potřebný zásah uživatele od vytvoření souboru s trajektorií po samotnou simulaci. Automatizace selhává v bodě, kde je potřeba trajektorii přiřadit stanici během simulace. To však současná verze OM nepodporuje podle odborné odpovědi z podpory OM. Trajektorii je možné přiřadit před simulací a následně projekt simulovat. Celý proces je však možné zautomatizovat s využitím přímé manipulace pozice stanice.

5.1 Model automatizovaného procesu

V projektu s automatizovaným procesem se jedná o proces předávání hodnot mezi OM a MATLABem během simulace bez zásahu uživatele. MATLABu jsou během simulace předány vstupní hodnoty z proměnných OM ve formátu jazyka C a následně se vykoná zavolání externí funkce pro vygenerování trajektorie. Po vykonání funkce se předávají výstupní hodnoty z

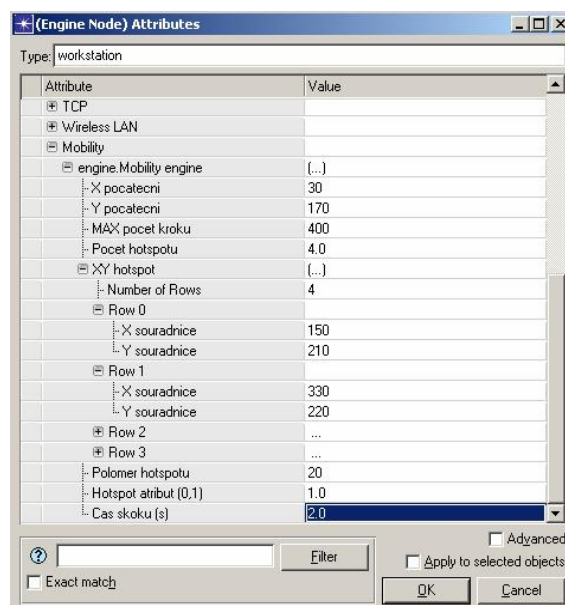
MATLABu zpět do OM. Na základě výstupních hodnot se přímou manipulací řídí pozice stanice. Na obrázku 6 je zobrazený diagram automatizovaného procesu.



Obrázek 6: OPNET MATLAB spolupráce

5.2 Atributy stanice

Před spuštěním simulace je třeba v OM definovat vstupní proměnné. Potřebná skupina vstupních atributů byla přidána do základní sady atributů každé mobilní stanice v prostředí OM [5]. Do vytvořených atributů může uživatel jednoduše vkládat potřebné hodnoty. Tento způsob byl zvolen proto, neboť je grafické rozhraní uživatelsky přívětivější oproti editování samotného zdrojového kódu. Především se tak i možnému chybnému editování zdrojového kódu do nefunkční podoby. Dané hodnoty se z grafického rozhraní vyčítají do zdrojového kódu, kde se dále zpracovávají. V projektu byla vytvořena skupina atributů s názvem „Mobility engine“. Tato skupina obsahuje vstupní proměnné, které jsou předávány vytvořenou funkcí v MATLABu. Na obrázku 7 je zobrazené menu vytvořených vstupních atributů stanice.



Obrázek 7: Vstupní atributy stanice

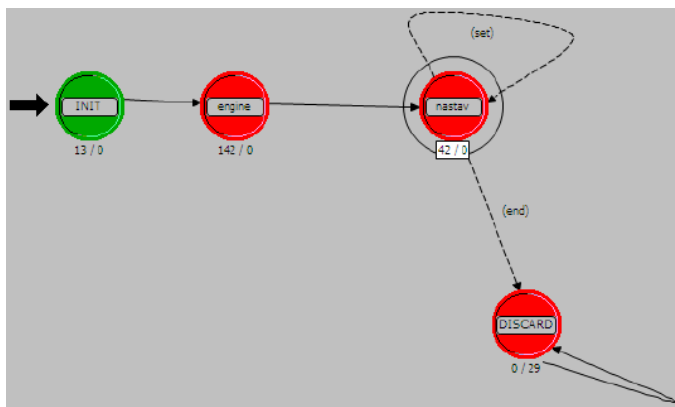
Atributy vytvořené skupiny a jejich význam je zobrazený v tabulce 4.

Tabulka 4: Atributy skupiny „Mobility engine“

Atribut	Význam
X pocatecni	počáteční X souřadnice
Y pocatecni	počáteční Y souřadnice
MAX pocet kroku	maximální počet kroků pro generování souřadnic
Pocet hotspotu	počet použitých hotspotů
XY hotspot	společné pole souřadnic pro hotspoty
Number of Rows	celkový počet hotspotů
X souradnice	X souřadnice pro daný hotspot
Y souradnice	Y souřadnice pro daný hotspot
Polomer hotspotu	vzdálenost od hotspotu pro jeho dosáhnutí
Hotspot atribut	atribut zapnutí vyhledávání hotspotů, 1 pro zapnutí, 0 pro vypnutí
Cas skoku	čas mezi skokem stanice z pozice A na pozici B

5.3 Vlastní procesy pro řízení pohybu

Na obrázku 8 je zobrazený konečný stavový automat procesoru s názvem „engine“. Byly v něm vytvořené dva nové stavy „engine“ a „nastav“. V stavu „engine“ se vyčítají proměnné z atributů stanice, které se následně zpracují. Provede se volání MATLAB funkce a zpracují se výstupní proměnné. Stav „nastav“ obsahuje kód, který vykonává samotné nastavování atributů pozice stanice. Nastavování trvá, dokud se nevyčerpají všechny souřadnice.



Obrázek 8: Konečný stavový automat procesoru ENGINE

6 Závěr

Byla přezkoumána problematika řízení pohybu mobilní stanice v OM. Pohyb stanice může být náhodný anebo řízený vstupními daty. Dále může být pohyb spojitý anebo nespojitý. O řízení pohybu se podle typu řízení stará jádro OM nebo uživatelem vytvořený proces k tomu určený.

V nástroji MATLAB byla vytvořena funkce pro zpracování mapového podkladu. Mapový podklad je v podobě vstupní

mapy ve formátu „*.bmp“. Nástroj MATLAB byl propojen s OM pomocí knihovny MATLAB Engine. Tato knihovna umožňuje propojení rozdílných programovacích jazyků a systémů. Obsahuje definované funkce, které zapouzdřují příkazy MATLABu do jazyka C/C++, na němž je založený OM.

Byl vytvořený automatizovaný model řízení pohybu bezdrátové stanice v OM. Pro samotné řízení byl vytvořený vlastní proces, který přímo řídí pohyb stanice.

7 Poděkování

Tento článek vznikl na základě podpory poskytnuté Grantovou agenturou České republiky (projekt č. GA102/09/1130) a Ministerstvem školství, mládeže a tělovýchovy České republiky (projekt č. MSM0021630513).

8 Literatura

- [1] OPNET Technologies, OPNET Modeler Product Documentation Release 16.0, 2010.
- [2] MOORE, H.: MATLAB for Engineers. New York: Prentice Hall, 2008, ISBN: 978-0136044222.
- [3] MATLAB, MATLAB Help version 7.5, R2007b.
- [4] AIT ALI, Kathina; MUSTAPHA LALAM, Mustap; MOALIC Laurent; BAALA, Oumaya. 2010 Ninth International Conference on Networks. *V-MBMM: Vehicular Mask-based Mobility Mode*.
- [5] O. Zeman. *Implementace simulačního modelu zjednodušené databáze diffserv-mib*. Diplomová práce, Brno: FEKT VUT v Brne, 2008. 55 s.