

# Simulace evakuačního procesu

Michal Špondr

27. května 2010

## 1 Úvod a popis simulovaného modelu

Tento projekt vychází ze článku *Simulation of evacuation processes using a biotics-inspired cellular automaton model for pedestrian dynamics* autorů Ansgar Kirchner a Andreas Schadschneider z Institut für Theoretische Physik. Universität zu Köln, vydaný 22. 3. 2002. Dále budu odkazovat na tuto literaturu použitím slova *článek*.

Dynamika pohybu vozidel patří k častým tématům fyzikálních modelů. Dynamika pohybu chodců již tak často zpracovávaná není, jelikož ji nelze popsat tak snadno jako dynamiku pohybu vozidel – na vozovce jsou dána pravidla, která vozidla dodržují, zatímco pro chůzi žádná taková pravidla nejsou. Byly však pozorovány některé zajímavé jevy, které se u dynamiky chodců vyskytují, například shlukování, tvoření front, oscilace u tzv. úzkých hrdel u skupin jdoucích proti sobě, či vzory pohybu v křižnicích se cestách.

Model se inspirovat procesem chemotaxe<sup>1</sup>. Podobně jako například mravenci zanechávají stopy pro navigaci ostatních jedinců, i chodci nechávají podobnou cestu. Jde však o jakousi virtuální trasu, která existuje jen v mysli chodce. V reálném světě to odpovídá například následování stejné cesty, jakou má chodec jdoucí vpředu.

Tento model použijeme pro simulaci evakuace lidí z místnosti. Při této situaci může dojít k panickým situacím, které jsou na simulaci velmi složité, ale pokusíme se tyto stavy zohlednit také.

Metoda ověřování vlivu každého chodce s každým je náročná na prostředky (složitost  $O(N^2)$ ). Zde použitý model ověřuje pouze své nejbližší okolí, má tedy složitost  $O(N)$ .

## 2 Koncepce modelu

Model (místnosti) je rozdělen na buňky, které mohou být buď prázdné nebo obsazené právě jedním chodcem. Každý chodec se může s určitou pravděpodob-

---

<sup>1</sup>Chemotaxe je pohyb buňky ve směru chemického gradientu. Například některé bílé krvinky putují do míst zánětu, v nichž vznikají specifické chemické látky (mediátory). – zdroj Wikipedia

ností přesunout čtyřmi směry na sousední buňku (nahoru, dolů, vlevo, vpravo; nikoliv diagonálně); jde o Von-Neumannův typ okolí (slidy SNT.pdf, str. 129). Pravděpodobnost přechodu je vypočtena podle interakce dvou *polí podlahy* – statické pole (S) a dynamické pole (D).

## 2.1 Statické pole podlahy

Statické pole se v čase nemění, není ovlivňováno chodci. Určuje oblasti místnosti, které jsou nějakým způsobem atraktivní, například únikové východy nebo okna. Zde statické pole popisuje nejkratší cestu k východovým dveřím.

Pro výpočet hodnot statického pole byla zvolena metrika použitá ve článku. Hodnota statického pole v místě  $(i, j)$  se vypočítá:<sup>2</sup>

$$S_{ij} = \max_{(i_{T_s}, j_{T_s})} \{ \max_{(i_l, j_l)} \{ \sqrt{(i_T - i_l)^2 + (j_{T_s} - j_l)^2} \} - \sqrt{(i_{T_s} - i)^2 + (j_{T_s} - j)^2} \}.$$

Síla statického pole je dána nejkratší vzdáleností k východu. Největší vzdálenost ze všech buněk od východu  $(i_{T_s}, j_{T_s})$  je  $\max_{(i_l, j_l)} \{ \sqrt{(i_T - i_l)^2 + (j_{T_s} - j_l)^2} \}$ , kde  $(i_l, j_l)$  jde přes všechny buňky mřížky (*lattice*). Tato hodnota díky normalizaci roste: čím blíže je k východu, a je nulová v nejvzdálenějším bodě od východu. Příklady statických polí jsou v podčásti 3.2.

## 2.2 Dynamické pole podlahy

Dynamické pole je virtuální cesta vytvářená chodci. Slouží k modelování atraktivity interakce mezi částicemi. Jakmile částice  $(i, j)$  přeskóčí na vedlejší buňku, zvýší se hodnota původní buňky o 1:  $D_{ij} \rightarrow D_{ij} + 1$ .

Dynamické pole D má nezáporné hodnoty, lze jej přirovnat k poli bosonů<sup>3</sup>. Chodci při svém přesunu „upouštějí“ bosony a tvoří tak virtuální cestu. Dynamické pole je nezávislé na čase, parametry, které jej ovlivňují, jsou *rozptyl*  $\alpha \in [0, 1]$  a *slábnutí*  $\delta \in [0, 1]$ . V každém kroku simulace každý boson zeslábně s pravděpodobností  $\delta$  a rozptýlí se do některého ze sousedních buněk s pravděpodobností  $\alpha$ .

## 2.3 Krok modelu

Pravidla pro aktualizaci jednoho kroku modelu jsou následující:

1. Dynamické pole D projde procesem zeslábnutí nebo rozptylu, jak bylo popsáno v předchozí podčásti.
2. Pro každého chodce se vypočte pravděpodobnost přechodu  $p_{ij}$  na neobsazenou buňku  $(i, j)$ . Hodnoty polí podlah D a S jsou váženy koeficienty

<sup>2</sup>V článku se vyskytuje hodnota  $\min_{(i_{T_s}, j_{T_s})}$ , předpokládám, že jde jen o chybu v zápisu; dále v článku i v popisu modelu je hodnota počítána jako  $\max_{(i_{T_s}, j_{T_s})}$ .

<sup>3</sup>Zjednodušený model popisující pole fermionů byl popsán v části 1 v souvislosti s chemotaxí

citlivosti  $k_S \in [0, \infty]$  a  $k_D \in [0, \infty]$ . Vzorec tedy je:

$$p_{ij} = \frac{\exp k_D D_{ij} \exp k_S S_{ij} (1 - n_{ij}) \xi_{ij}}{\sum_{i,j} \exp k_D D_{ij} \exp k_S S_{ij} (1 - n_{ij}) \xi_{ij}},$$

kde  $n_{ij} = 0, 1$  značí obsazenost buňky;  $\xi_{ij} = 0$  pro zakázané buňky (zdi), jinak 1. Suma ve jmenovateli značí normalizaci hodnot.

3. Každý chodec si vybere koncovou buňku podle vypočtených pravděpodobností. Vyřeší se případné konflikty, pokud chce více chodců vstoupit na jednu buňku. Pokud se může pohybovat, přesune se na danou buňku.
4. Všechny pohybující se částice zvýší hodnotu  $D$  v daném místě.

Všechny změny se provádí zároveň, aktualizace je paralelní.

Dle článku model ukazuje, že jednoduchá pravidla a dvě pole podlah dostačují k dosažení bohatosti složitého jevu. Tento způsob výpočtu je vhodný pro simulace davů v reálném čase například ve veřejných budovách. Například pro větší obeznámenost chodců s tvarem místnosti a umístěním východů (a tedy zvýšení povědomí o statickém poli, což značí zvednutí konstanty  $k_S$ ) by bylo možné dávat na zdi nebo podlahu směrová znamení, pokud by simulace ukázala, že chodci se zbytečně shlukují na nějakém místě.

### 3 Architektura a implementace simulátoru

Celulární automat (slidy SNT.pdf, str. 127) byl naprogramován v programovacím jazyce C++, důvodem je kromě přenositelnosti hlavně kompromis mezi jednoduchostí vývoje a rychlostí výsledného programu.

Výstup programu je prováděn do textového terminálu v podobě jednoduchého textového obrázku; tato volba byla zvolena proto, že je zde podstatná samotná simulace a výstup dostačuje textový. Znak **@** odpovídá chodci, **X** představuje neprůchodnou překážku (zeď). Pro přehlednější zobrazení byla použita knihovna *ncurses*<sup>4</sup>, která umožňuje snadné překreslování terminálu.

#### 3.1 Spuštění programu

Program lze sestavit příkazem `make`, `make clean` vymaže vytvořené soubory včetně spustitelného.

Aplikace po spuštění (název spustitelného souboru je `evacuation`) zobrazí okno, pod kterým je vypisován počet zbývajících chodců v místnosti a uběhlý čas (čas je zde bezrozměrná veličina, která odpovídá počtu kroků automatu. Program se zastaví, jakmile v místnosti není žádný chodec. Ukončení a návrat do terminálu

<sup>4</sup><http://www.gnu.org/software/ncurses/>

se provede stiskem libovolné klávesy.

Program má několik parametrů, které umožňují nastavit chod simulace.

- -a přirozené číslo – nastavuje šířku generovaného pole
- -h přirozené číslo – nastavuje výšku generovaného pole
- -p přirozené číslo – počet chodců v místnosti
- -d reálné číslo 0.0-1.0 – rozptyl udávající, s jakou pravděpodobností se hodnota buňky dynamického pole rozšíří na nějakou z okolních buněk
- -e reálné číslo 0.0-1.0 – slábnutí udávající, s jakou pravděpodobností se sníží hodnota buňky dynamického pole o 1
- -s reálné číslo – hodnota konstanty  $k_S$  statické vrstvy, která udává, jak velký význam tato vrstva má
- -k reálné číslo – hodnota konstanty  $k_D$  dynamické vrstvy, která udává, jak velký význam tato vrstva má
- -r přirozené číslo – sůl generující tvorbu náhodných čísel, implicitně není zadáno a číslo je generováno náhodně
- -t přirozené číslo – čas v mikrosekundách udávající dobu trvání zobrazení jednoho kroku simulace
- -a číslo 0, 1, 2, 3, 4 – číslo určující typ místnosti, které jsou uvedeny v podčásti 3.2
- -u – nastavením tohoto příznaku se lze posouvat v simulaci stiskem libovolné klávesy

## 3.2 Použité typy místností

Pro demonstraci simulace bylo vytvořeno 5 různých typů místností, které lze v programu zvolit volbou -a a číslem 0 až 4. Ukažme si následující místnosti.

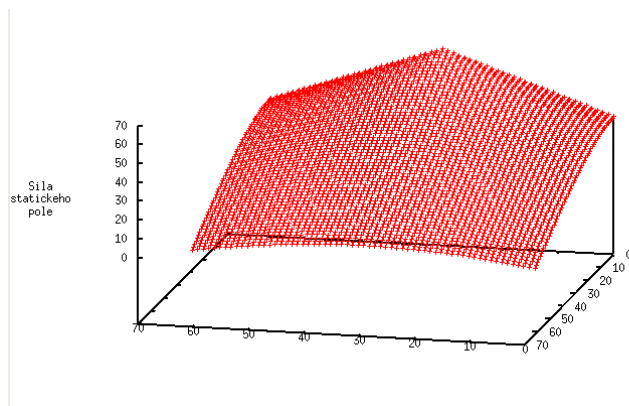
### 3.2.1 Místnost 0 – jedny dveře uprostřed v horní části místnosti

Takto vypadá statické pole místnosti. Můžeme vidět, že nejvyšší hodnota je u východu.

```

XXXXXXXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXXXXXXX
X                                     X
X                                     X
X                                     X
X                                     X
X                                     X
XXXXXXXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXXXXXXX

```



Obrázek 1: Statické pole místnosti 0

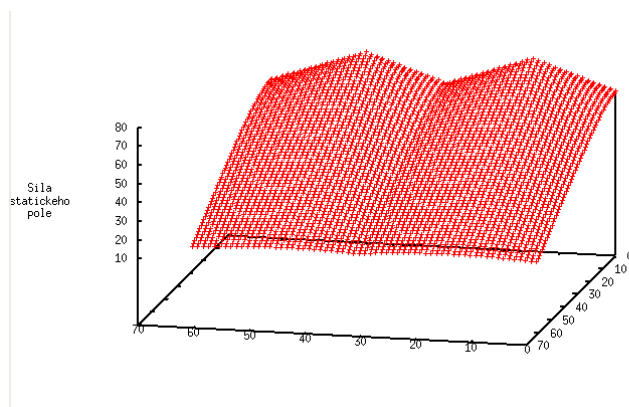
### 3.2.2 Místnost 1 – dvoje dveře v horní části místnosti

Při simulaci se chodci rovnoměrně hromadí okolo obou východů.

```

XXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXX
X                                                    X
X                                                    X
X                                                    X
X                                                    X
XXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXX

```



Obrázek 2: Statické pole místnosti 1

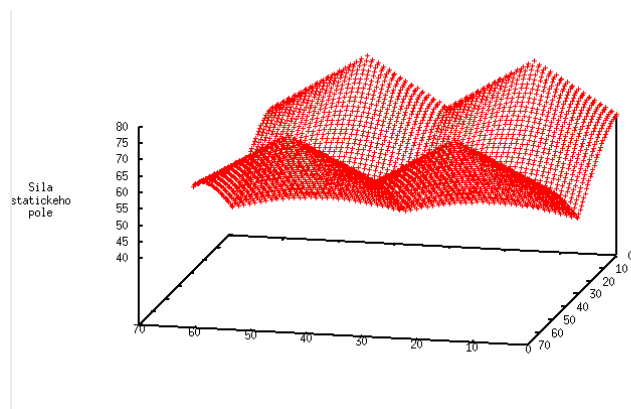
### 3.2.3 Místnost 2 – čtyři dveře, dvoje v horní části a dvoje v dolní části místnosti

Zde se chodci rovnoměrně rozmístí mezi 4 východy.

```

XXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXX
X                                                    X
X                                                    X
X                                                    X
X                                                    X
XXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXX

```



Obrázek 3: Statické pole místnosti 2

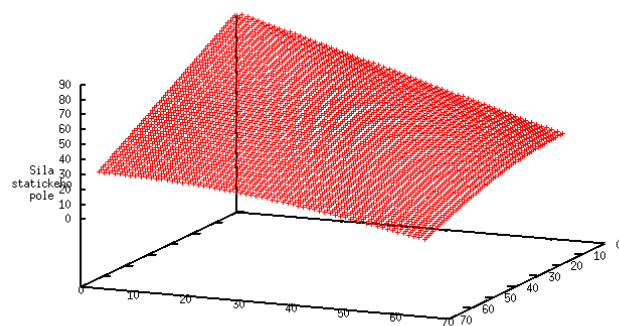
### 3.2.4 Místnost 3 – dvoje dveře v levé horní části místnosti

Chodci vyplní celou levou stěnu místnosti. Není-li místnost dostatečně vysoká, nemá dav takový kulovitý tvar, jako okolo dveří uprostřed stěny.

```

X-XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
-                                                    X
X                                                    X
X                                                    X
X                                                    X
XXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXX

```



Obrázek 4: Statické pole místnosti 3

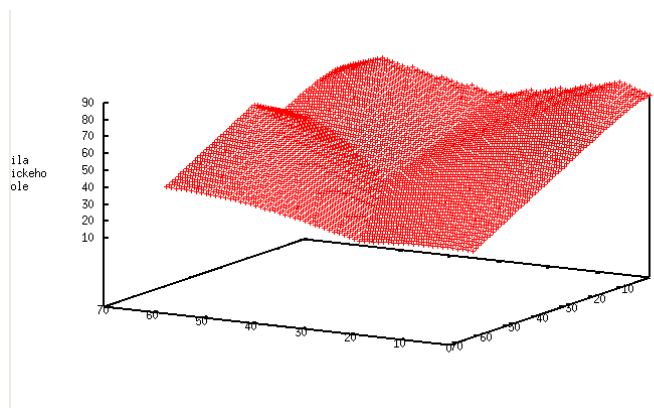
### 3.2.5 Místnost 4 – 9 dveří v horní části místnosti a jedny dveře v pravé části místnosti

Zde se dav obvykle rozdělí do 2 částí. Jedna odchází dveřmi vlevo nahoře, druhá dveřmi v pravé stěně. Zde je viditelný vliv dynamické vrstvy – chodci se snaží opustit místnost dveřmi, kterými je opouští ostatní, i když mají blíže nějaké volné dveře.

```

XXXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX
X                                           X
X                                           X
X                                           -
X                                           X
X                                           X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```



Obrázek 5: Statické pole místnosti 4

### 3.3 Vlivy parametrů $k_S$ a $k_D$

Experimenty ukazují, že pro  $k_S \rightarrow 0$  k odchodu lidí z místnosti nedojde. Jejich pohyb je chaotický a náhodný, neexistuje žádné podvědomí o vzhledu místnosti. Chodci pak dospějí do nějakého bodu, z kterého už se nedostanou, hodnoty dynamické vrstvy budou příliš vysoké. Tato hodnota  $k_S$  neodpovídá realitě, člověk většinou nějaký přehled o svém okolí má. Pro hodnotu 1 došlo k velmi pomalému úbytku chodců, zřejmě jde o nevhodně nastavené implicitní hodnoty (dynamická vrstva příliš rychle zeslábne, hodnoty se málo šíří...) Se zvyšujícími se hodnotami  $k_S$  rychlost narůstá, pro  $k_S \rightarrow \infty$  získává chodec absolutní znalost místnosti a míří nejkratší možnou cestou k východu.  $k_S$  a statické pole jsou hlavními parametry ovlivňující cestu chodců.

$k_D$  a dynamické pole se chovají spíše jako dodatečný šum, ovlivňující  $k_S$  a statické pole, v podstatě dobu vyprázdnění místnosti prodlužuje. Nicméně je důležitý pro odraz chování lidí. Pro hodnoty  $k_D \rightarrow 0$  je vliv dynamického pole nepatrný a chodci odcházejí díky hodnotám statického pole. Se zvyšující se hodnotou  $k_D$  roste chaotický pohyb obzvláště v místech, která mají nízkou hodnotu  $k_S$  (= statické pole zde nemá téměř žádný vliv). Zde se chodci pohybují stádně, následují ostatní a může se stát, že dospějí k nějakému lokálnímu extrému, z kterého už se k východu nedostanou.

### 3.4 Vlivy parametrů $\alpha$ a $\delta$

Máme-li konstantní hodnoty  $k_S$  a  $k_D$ , můžeme porovnávat vliv hodnot rozptylu  $\alpha$  a slábnutí  $\delta$ , které ovlivňují hodnoty dynamického pole D.

$\alpha \rightarrow 0$  způsobuje, že D není rušeno, se zvyšující se hodnotou  $\alpha$  dochází ke zvyšování evakuační doby. Rozptyl tedy znamená jen dodatečný šum, vysoká hodnota není výhodná.

$\delta \rightarrow 0$  způsobuje, že hodnota D přetrvává delší dobu a jak bylo řečeno v podčásti 3.3, dynamická vrstva je zde spíše nežádoucí šum. Čím je hodnota  $\delta$  vyšší, tím více způsobuje slábnutí dynamické vrstvy a zrychlení doby evakuace.

### 3.5 Náměty k dalšímu vylepšení

Simulátor by bylo možné vylepšit v řadě směrů. Například přidáním větší interaktivity (vytváření vlastních místností za chodu, přidávání chodců, změna parametrů za běhu atd.) Také by se hodil nějaký generátor jiných místností, než jsou obdélníkové.

Dále si myslím, že by bylo třeba více promyslet výpočet dynamické vrstvy, který dle mého názoru neodpovídá realitě (chodci se dostávají do lokálních extrémů, z kterých se nedostanou ven). Úprava by mohla spočívat v pečlivějším výběru koeficientu  $k_D$  (hodnota by se mohla například dynamicky měnit), případně v jiném výpočtu hodnot dynamického pole, který by dokázal tyto lokální extrémy vyrušit.

Také jsem vypořizoval, že se v nakupených skupinách okolo východů chodci netlačí. Tedy že dojde k uvolnění místa před chodcem směrem k východu, ale on i přesto zvolí méně výhodný krok například vzad. V reálu lidé v davu krok vzad nedělají. Možnou změnou by byla výše zmíněná dynamická změna  $k_S$  a  $k_D$ , kdy v okamžiku, kdy chodec východ vidí, by se  $k_D$  snížilo k 0 a  $k_S$  zvýšilo.

Aplikace dále naráží na limity dané velikostí rozsahu proměnných. Výpočet pravděpodobnosti používá exponenciální funkci a vezmeme-li maximální velikost typu double, tj. 1.79769e+308, zjišťujeme, že nejvyšší hodnota funkce  $\exp()$  může být zhruba 709. Je-li místnost velká, může se stát, že tato hodnota bude překročena, tím dojde k přetečení této proměnné a ke špatnému výpočtu. Příkladem může být simulace `./evacuation -a 3 -w 141` a `./evacuation -a 3 -w 142` (a vyšší hodnoty parametry `-w`) – zde vidíme, že chodci z místnosti neodejdou, protože při výpočtu dojde k přetečení a pravděpodobnost přechodu na pole s dveřmi je 0%.



## 4 Shrnutí a závěr

Vytvořil jsem simulátor, který simuluje model popsany ve článku *Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics*. Model demonstruje evakuaci lidí z uzavřené místnosti, bere v potaz jak samotný tvar místnosti a pozici východů (statická vlastnost), tak chování lidí v davu, jako např. stádové chování, panika, zácpy, ... (dynamická vlastnost). Simulátor je napsán v programovacím jazyce C++ a používá knihovnu ncurses. Má výstup do textové konzole a nastavuje se parametry v příkazové řádce, má 5 přednastavených typů místností.

Na základě experimentů se domnívám, že práce s dynamickým pole ne zcela odpovídá reálnému chování, je třeba hledat vhodné koeficienty  $k_D$ ,  $\alpha$  a  $\delta$  pro konkrétní místnost. V přeplněných místnostech se může během shlukování u vchodu stát, že velmi naroste hodnota dynamické vrstvy na určité buňce a po uvolnění okolních buněk nemá chodec tendenci se přesouvat jinam a uvázne na nějakou dobu na jednom místě, než hodnota  $D$  na dané buňce dostatečně zeslábně. Teoreticky by tento stav mohl odpovídat například nějakému psychickému zkolabování jedince, nicméně o této možnosti jsem v článku nic nenašel a proto to považuji spíše za chybu algoritmu. Ve článku samotném autoři zmiňují, že tento model představuje pouze jednoduchý scénář. V reálném světě má každý jednotlivec vlastní  $k_S$  a  $k_D$  a řešení nahrává spíše použití agentních systémů.

Ponecháme-li  $k_D$  velmi malé a zvýšíme-li  $k_S$ , chodci se pěkně hromadí okolo východů, k žádnému uvážnutí či zablokování jednotlivců nedochází. Vychází to tedy „pěkně“, v situacích, kde chodci místnost výborně znají, by mohlo k takové ideální evakuaci dojít.

Simulaci by šlo v praxi použít například pro ověření, kudy se budou chodci pohybovat a na místa, kde by docházelo k lokálním extrémům, použít nějaké prvky (značky, šipky, nápisy), které by zvýšily hodnotu statického pole a chodce tak nasměrovaly a vyvedly z tohoto lokálního extrému ven.