**Akademia Górniczo-Hutnicza**
**im. Stanisława Staszica w Krakowie**

Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki


# Symulacja ruchu samochodów na I Obwodnicy Krakowa

**Oskar Kocjan**
**Maciej Koch**

# Spis treści

# 1 Wprowadzenie

## 1.1 Opis problemu

W wielu miastach codziennie możemy spotkać się z dużym ruchem na drogach publicznych co niestety w przypadku większych miejscowości skutkuje korkami, częstymi wypadkami, wstrzymaniami komunikacji miejskiej. Innymi powodami wyżej wymienionych sytuacji mogą być również planowane przez urzędy miast plany rozbudowy infrastruktury czy też samych dróg. Przeprowadzanie skutecznych i realistycznych symulacji pozwala na lepsze rozwiązanie tych problemów. W naszym projekcie zajmiemy się I obwodnicą Krakowa. W tym celu posłużymy się modelem Nagela-Schreckenberga - klasyczny sposób oparty na automatach komórkowych , który służy do opisu ruchu samochodów. Uwzględnimy specyfikę pojazdów oraz ich ruch w oparciu o automaty komórkowe. Określimy sąsiedztwo (w ujęciu Moore'a), przyśpieszenie, hamowanie, sytuacje wyprzedzania oraz zmiany pasa ruchu.

## 1.2 Analiza modelu

Automaty komórkowe ( ang. cellular automata) mogą być prezentowane w postaci czwórki $A = (\alpha, S, N, f)$ :

$\alpha$ - definiuje regularną, uporządkowaną siatkę złożoną z jednakowych komórek

$S$ - skończony zbiór stanów, jaki może przyjąć komórka

$N$ - zbiór sąsiadów

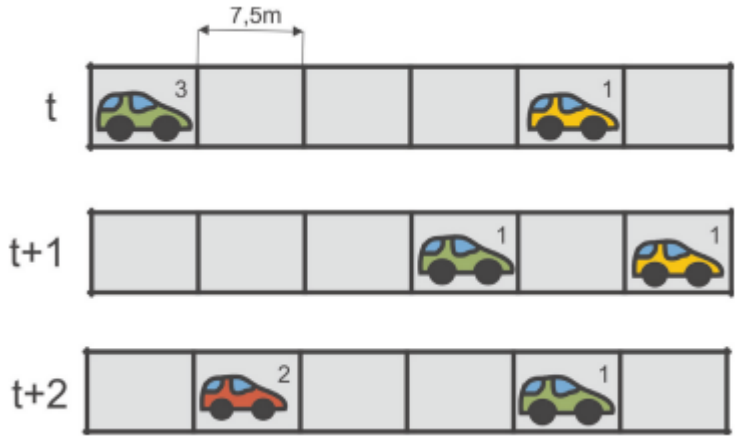
$f$ - funkcja przejścia która definiuje ewolucje automatu w kolejnych jednostkach czasu

W modelu Nagel–Schreckenberga droga jest podzielona na komórki. Każda z nich może być w dwóch stanach tj. albo pełna (czyli zawiera jeden samochód) lub pusta (nie zawiera wcale). Każdy pojazd ma przypisaną prędkość od 0 do $V_{max}$ ustalonego gdzie ta wartość jest liczbą całkowitą. Informuje ona o liczbie komórek , o które przesunie się pojazd w kolejnym kroku.Jest to model symulacyjny autostrady opracowany do badania warunków spontanicznego powstawania korków, jednak idealnie nada się on również do opisu ruchu miejskiego.

## 1.3 Reguły poruszania pojazdów

W każdym dyskretnym kroku dla wszystkich pojazdów są wykonywane równolegle kolejno następujące reguły poruszania się:

- przyśpieszenie $v(t+1) \rightarrow min(v(t)+1, v_{max})$, gdzie $v(t)$ to prędkość aktualna pojazdu
- hamowanie $v(t+1) \rightarrow min(v(t), g(t)-1)$, gdzie $g(t)$ jest liczbą pustych komórek między pojazdami
- element losowy, prawdopodobieństwo $p$, że zajdzie $v(t+1) \rightarrow max(v(t)-1)$, jeżeli $v(t) \geqslant 1$
- ruch (zmiana położenia w czasie) $x(t+1) = x(t) + v(t)$




Rysunek 1: Ruch modelu Nagela-Schreckenberga na pasie ruchu kolejnych chwilach czasowych. W lewym górnym rogu każdej komórki można zauważyć aktualną prędkość pojazdu podaną w komórkach na chwilę czasową

**Sygnalizacja świetlna**

W naszym projekcie chcemy zmodyfikować model poprzez dodanie dodatkowych parametrów w postaci sygnalizacji świetlnej co wpłynie na element ruchu:

jeżeli: $s(t) = 1$ i $x_i(t) < x_s$, to $v_i(t) \leftarrow min(v_i(t), x_s - x_i(t) - 1)$
gdzie:

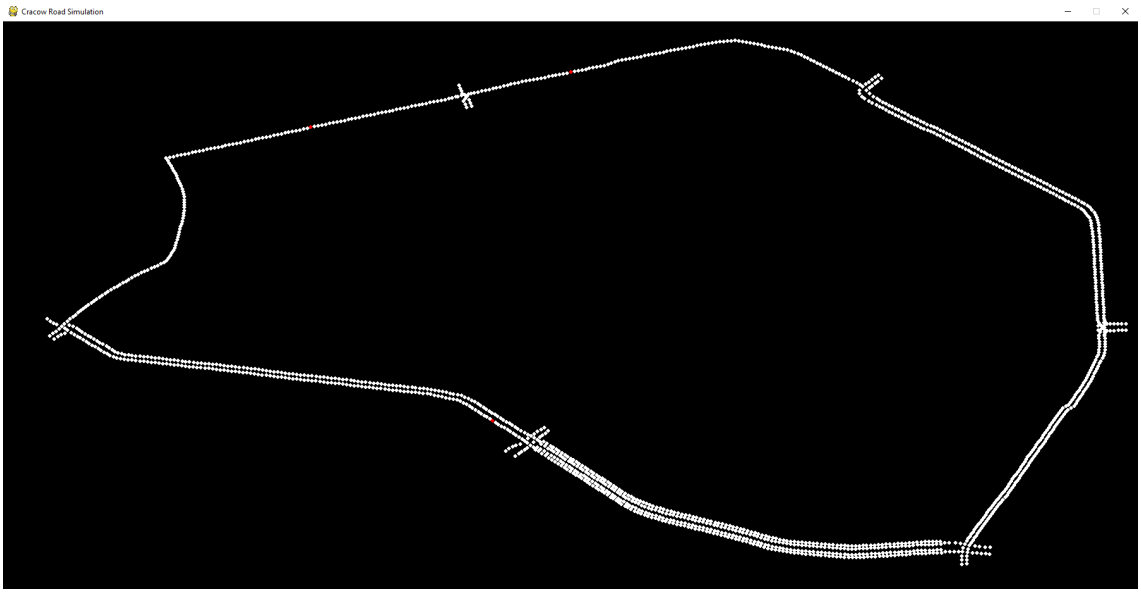
$s(t)$ - oznacza sygnał wyświetlany dla pojazdów w chwili $t$
($s(t) = 1$, gdy wyświetlany jest sygnał czerwony)

$x_s$ - jest numerem komórki, która znajduje się na modelowanym wlocie skrzyżowania, bezpośrednio za linia warunkowego zatrzymania

# 2 Plan pracy

## 2.1 Przygotowanie siatki obwodnicy

Początek pracy rozpoczęliśmy od przygotowania siatki naszej obwodnicy. W tym celu wykorzystaliśmy oprogramowanie JOSM, do którego zaimportowaliśmy mapę w formacie *mapa.osm* , a następnie nanieśliśmy punkty otrzymując w formacie *mapa.json* zbiór koordynatów. Kolejnym krokiem była odpowiednia ich konwersja w Pythonie oraz wstępna wizualizacja poprzez wykorzystanie biblioteki Pygame.



## 2.2 Stworzenie logiki modelu

## 2.3 Tworzenie wizualizacji symulacji

## 2.4 Testowanie i poprawianie błędów

## 2.5 Analiza wyników końcowych