

# Symulacja ruchu drogowego na IV obwodnicy Krakowa

Szymon Gałuszka, Michał Worsowicz, Maciej Nalepa

10 czerwca 2020

## Część I

# Omówienie projektu

## 1 Definicja problemu

Nasz cel to symulacja ruchu drogowego na IV obwodnicy Krakowa. Konieczna jest definicja tras oraz sposobu poruszania się po nich. Obwodnica tworzy zamkniętą pętlę (uwzględniając odcinek w budowie) i dzieli się na ruch zgodny i przeciwny do wskazówek zegara.

Ruch drogowy powinien uwzględniać pojazdy osobowe, ciężarowe i transport publiczny. Ciężkie pojazdy obowiązują inne ograniczenia prędkości oraz zajmują więcej przestrzeni na jezdni. Napływ ruchu powinien odbywać się przez wjazdy na obwodnicę, które wpuszczają samochody z określoną częstotliwością, która może zależeć od pory dnia.

Symulacja może obejmować wydarzenia losowe, takie jak:

- blokada pasa ruchu,
- zamknięcie zjazdu,
- nagłe hamowanie.

## 2 Obszar symulacji

Symulowany przez nas obszar to IV obwodnica Krakowa, znana także jako obwodnica autostradowa Krakowa, ponieważ większość jej odcinka stanowi autostrada A4.

Na obwodnicy nie ma sygnalizacji świetlnej, skrzyżowania znajdują się najczęściej pod wiaduktem i najpierw należy zjechać z drogi szybkiego ruchu. Trasa ma zróżnicowane ograniczenia prędkości oraz różną ilość pasów ruchu.

Fragment ten na zachodnich obrzeżach miasta jest dwupasmowy, a na południowych - trójpasmowy. W najbliższym czasie przewidywana jest budowa odcinka północnego, który uwzględniliśmy w naszej symulacji, odtwarzając i dodając ten fragment do projektu. Poniżej przedstawiony jest wykaz węzłów drogowych, które obejmuje symulacja (zgodnie z kierunkiem ruchu wskazówek zegara):

- Kraków Nowa Huta,
- Kraków Przewóz,
- Kraków Bieżanów,
- Kraków Wieliczka,
- Kraków Łagiewniki,
- Kraków Południe,
- Kraków Skawina,
- Kraków Tyniec,
- Kraków Bielany,
- Kraków Balice (lotnisko),
- Balice I,
- Modlniczka,
- Modlnica,
- Kraków Północ – dla uproszczenia przyjęto taką nazwę planowanego węzła (wariant I)

### 3 Algorytm

Zastosowany przez nas algorytm ruchu drogowego to zmodyfikowany model Nagela-Schreckenberga, będący teoretycznym modelem mikroskopowym o charakterze dyskretnym, w którym obie jednokierunkowe jezdnie są podzielone na komórki, które zawierają informację na temat ilości pasów oraz ilości pojazdów w danej chwili na jezdni. Dzięki takiemu uproszczeniu, nie jest konieczna symulacja zmiany pasu ruchu, ponieważ nie jest on określony wprost.

Każda z komórek może być interpretowana na jeden ze sposobów:

- komórka pusta – fragment pustej drogi
- komórka używana – fragment zawierający pojazd na nieokreślonym pasie
- komórka pełna – fragment drogi z pojazdem na wszystkich pasach ruchu

Każdy samochód  $n$  posiada swoją prędkość  $v(n)$  o wartości liczby naturalnej, nie większej od prędkości maksymalnej  $v_{max}$  – symbolizuje ona maksymalną prędkość osiąganą przez samochód.

Czas  $t$  w modelu Nagela-Schreckenberga przyjmuje wartość dyskretną o stałym kroku, gdzie każdy etap można podzielić na cztery następujące po sobie czynności:

1. Przyspieszanie:

Jeżeli  $v(n) < v_{max}$  to prędkość samochodu może ulec zwiększeniu o zadaną jednostkę, nie przekraczając prędkości maksymalnej.

2. Zwalnianie:

Jeżeli odległość  $d(n)$  samochodu  $n$  od samochodu znajdującego się przed nim jest mniejsza od prędkości  $v(n)$  tego samochodu to prędkość samochodu ulega zmniejszeniu. Jako że odległość mierzona jest w liczbie komórek, a prędkość w liczbie komórek na jednostkę czasu, to prędkość ostateczna może wynosić maksymalnie  $d(n)$  na jednostkę czasu, a minimalnie zero.

3. Losowość:

Czynność ta symbolizuje wszelkie przypadki losowe z jakimi kierowca może spotkać się na drodze. Dla każdego samochodu  $n$ , gdzie  $v(n) > 0$ , prędkość zostaje zmniejszona o jedną jednostkę, z pewnym zadanyim prawdopodobieństwem  $p$ .

4. Ruch samochodu:

W ostatnim kroku każdy z samochodów zostaje przesunięty do przodu o odpowiednią ilość komórek, wynikającą z jego prędkości.

## Część II

# Metodologia

## 4 Struktury danych

## 5 Implementacja

## 6 Usprawnienie

## Część III

# Podsumowanie

## 7 Wyniki

Czas wykonania 10 minutowej symulacji wynosi około 2 minut przy domyślnych ustawieniach (procesor i5-8th Gen 4GHz).

Wszystko jest gites i działa!

## 8 Od autorów

Nasz framework umożliwia w obecnym stanie analizę interesującego parametru: średniego czasu jaki samochód spędza na obwodnicy. Postanowiliśmy zostawić to jako propozycję do samodzielnego eksperymentu z tą symulacją.

## 9 Dalsza praca

Projekt nie posiada innych pojazdów, niż domyślny osobowy. Zatem pierwszym usprawnieniem jest wprowadzenie pojazdów, które:

- obowiązują inne ograniczenia prędkości
- zajmują więcej, niż jedną komórkę

Obsługa całej platformy może zostać zawarta w wygodnej dla użytkownika oprawie graficznej, np. `kivy`. Na chwilę obecną najbardziej interaktywny interfejs uzyskujemy poprzez `jupyter notebook`, który ma duże ograniczenia wydajności.

Zapis jednego stanu mapy to ramka danych o rozmiarze ilości komórek na mapie. Oznacza to, że pojedyncza sekunda symulacji generuje nawet 30000 wpisów do ramki. W celu uniknięcia problemu z zapelnianiem pamięci stworzono specjalną klasę `LastCellStat` przechowującą określoną ilość stanów wstecz. Lepszym rozwiązaniem może być zapis logów bezpośrednio do pliku.

## Literatura

- [1] Wiki, *Kraków Obwodnica IV*, [wikipedia.org](https://wikipedia.org)
- [2] OpenStreet, *mapa*, [openstreetmap.org](https://openstreetmap.org)
- [3] K. Nagel, M. Schreckenberg, *Two lane traffic simulations using cellular automata*, [arxiv.org](https://arxiv.org)