

# 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

<b>1</b>	<b>Wprowadzenie</b>	<b>2</b>
1.1	Opis problemu . . . . .	2
1.2	Cel naszej symulacji . . . . .	2
1.3	Podstawowy model Na-Sch . . . . .	2
1.4	I Obwodnica Krakowa . . . . .	3
<b>2</b>	<b>Analiza modelu oraz modyfikacje podstawowego modelu Na-Sch</b>	<b>4</b>
2.1	Automaty komórkowe . . . . .	4
2.2	Reguły poruszania pojazdów . . . . .	4
<b>3</b>	<b>Wybór technologii</b>	<b>6</b>
<b>4</b>	<b>Symulacja zjawiska - implementacja</b>	<b>6</b>
4.1	Przygotowanie siatki obwodnicy . . . . .	6
4.2	Stworzenie logiki modelu . . . . .	8
4.3	Opis klasy Point i Car . . . . .	9
4.4	Opis zasady działania algorytmu . . . . .	10
4.5	Dane wejściowe i wyjściowe . . . . .	11
4.5.1	Dane wejściowe . . . . .	11
4.5.2	Dane wyjściowe . . . . .	11
<b>5</b>	<b>Seria symulacji i analiza wyników</b>	<b>12</b>
5.1	Statystyki . . . . .	13
5.2	Wykresy . . . . .	15
<b>6</b>	<b>Wnioski</b>	<b>17</b>
<b>7</b>	<b>Literatura</b>	<b>18</b>

# 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.

## 1.2 Cel naszej symulacji

W naszym projekcie zajmiemy się I obwodnicą Krakowa. Głównym celem będzie przeprowadzenie takiej symulacji, aby w jak najbardziej realistyczny sposób zasymulować dynamikę ruchu miejskiego. Przedstawić ruch samochodów jednoosobowych oraz ciężarowych. Odzwierciedlimy faktyczny układ dróg z uwzględnieniem najważniejszych skrzyżowań, dróg jednokierunkowych, wielopasmowych oraz węzłów z sygnalizacją świetlną. 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), przyspieszenie, hamowanie, losową zmianę prędkości, regułę prawej ręki.

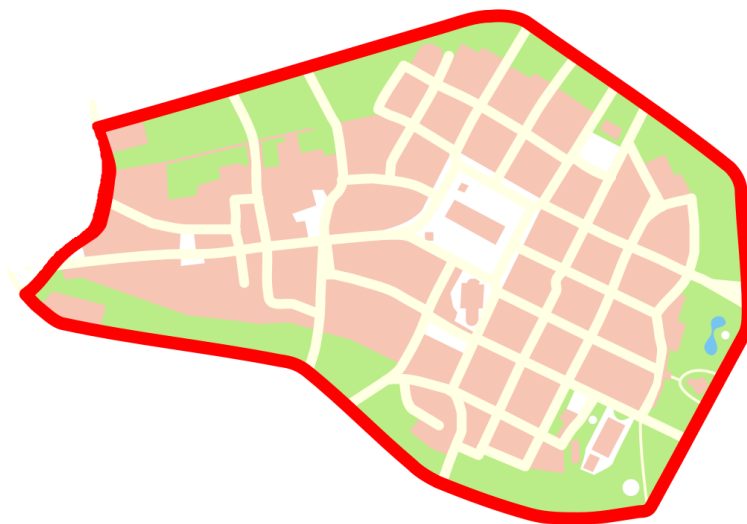
## 1.3 Podstawowy model Na-Sch

W 1992 r. Nagel i Schreckenberg przedstawili model - automat komórkowy opisujący ruch pojazdów. Założenia modelu to:

- podział ulicy na komórki długości 7.5m
- droga jest jednokierunkowa jedno pasowa
- komórka może być zajęta przez jeden pojazd lub wolna
- każdej komórce przypisana jest wartość prędkości
- ruch pojazdów w tym modelu opisują 4 pojęcia:
  1. przyspieszanie - pojazd zwiększa swoją prędkość
  2. hamowanie - zmniejszenie szybkości w komórce poprzedzającej znajduje się pojazd
  3. hamowanie - na drodze bywają zdarzenia losowe np. wtargnięcie pieszego na jezdnię
  4. przesunięcie - o wykonaniu powyższych czynności następuje przesunięcie wszystkich pojazdów do następnej komórki i zwiększenie zmiennej czasu

## 1.4 I Obwodnica Krakowa

Mianem pierwszej obwodnicy określa się w Krakowie ciąg ulic otaczających Stare Miasto wzdłuż plant. Ruch na obszarze wewnątrz pierwszej obwodnicy jest wysoce ograniczony



W skład obwodnicy wchodzi niżej przedstawiony wykaz ulic (zgodnie z ruchem wskazówek zegara) :

- ul. św. Idziego (ograniczenie ruchu w jednym kierunku)
- ul. Podzamcze (ograniczenie ruchu w jednym kierunku)
- ul. F. Straszewskiego (ruch jednokierunkowy na odcinku ulicy)
- ul. Podwale (ograniczenie ruchu na odcinku ulicy i ruch jednokierunkowy na całości)
- ul. J. Dunajewskiego (ograniczenie ruchu i ruch jednokierunkowy)
- ul. Basztowa (ograniczenie ruchu na odcinku i ruch jednokierunkowy na całej ulicy)
- ul. Westerplatte
- ul. św. Gertrudy

## 2 Analiza modelu oraz modyfikacje podstawowego modelu Na-Sch

### 2.1 Automaty komórkowe

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

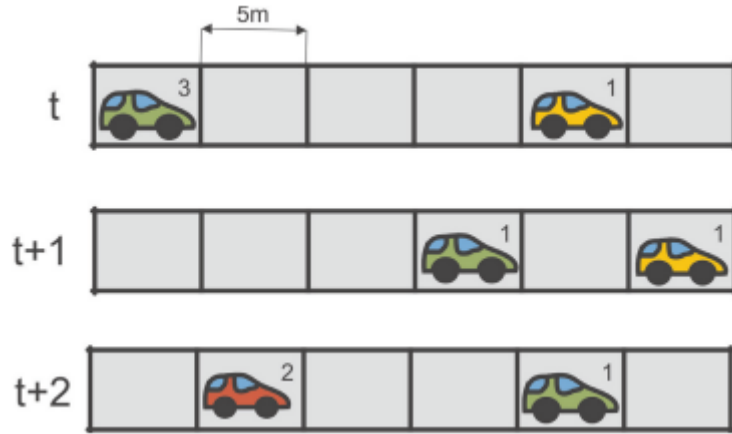
$f$  - funkcja przejścia która definiuje ewolucje automatu w kolejnych jednostkach czasu

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. W naszym projekcie ustaliliśmy trzy rodzaje prędkości, gdyż zakładamy, iż samochody nie przekraczają pewnej granicy tym bardziej na I obwodnicy, gdzie ruch jest i tak z powodów oczywistych utrudniony. Dodatkowo w naszym przypadku, komórki mają długość 5m.

### 2.2 Reguły poruszania pojazdów

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

- przyspieszenie  $v(t+1) \rightarrow \min(v(t) + 1, v_{max})$ , 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  $v(t) \geq 1$
- 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 linią warunkowego zatrzymania

## Reguła prawej ręki

Kolejną modyfikacją podstawowego modelu będzie dodanie reguły prawej ręki, która będzie działać w następujący sposób dla danego skrzyżowania:

jeżeli  $(x(t) = p \in A) \wedge (\forall z \in B)$ , jeżeli  $\exists z_n = 1 \Rightarrow v(t) = 0$   
gdzie:

$A$  - oznacza zbiór punktów (komórek) linii warunkowego zatrzymania na skrzyżowaniu w przypadku świateł, bądź ustąpienia pierwszeństwa.

$B$  - oznacza zbiór punktów (komórka) *otoczenia* skrzyżowania, co w praktyce oznacza punkty skrzyżowania oraz punkty odcinka pasa ruchu na którym poruszają się samochody, którym być może będzie należało ustąpić pierwszeństwa. Każdy element tego zbioru może przyjmować wartość 0 lub 1 co oznacza kolejno iż punkt jest wolny lub zajęty.

### 3 Wybór technologii

Do stworzenia naszego projektu, wykorzystaliśmy następujące technologie:

- Środowisko PyCharm
- OpenStreetMap
- Oprogramowanie JOSM
- Github

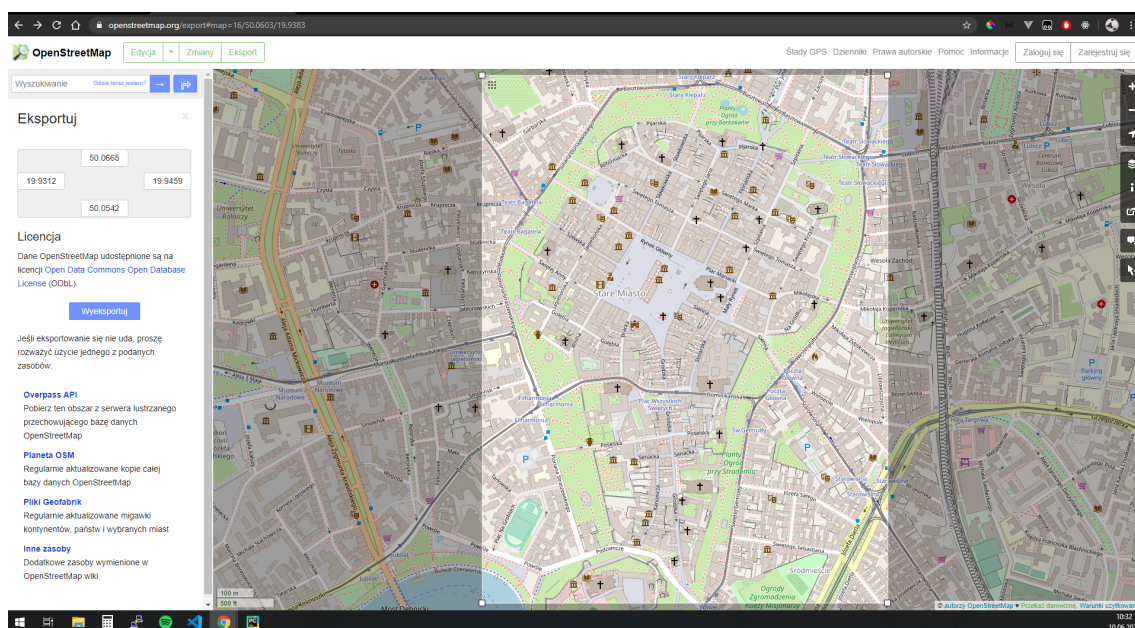
Zdecydowaliśmy się na wykorzystanie Pythona, gdyż jest prostym językiem, idealnie nadającym się do wykonywania różnych nieskomplikowanych matematycznie operacji. Istnieje wiele przydatnych bibliotek, które ułatwiają pracę. W naszym projekcie do samej wizualizacji użyliśmy PyGame, a do implementacji wszelkich zachowań oraz warunków ruchu wykorzystaliśmy m. in. os, json, threading, math, czy też time.

Dzięki stronie OpenStreetMap mogliśmy uzyskać rzeczywiste współrzędne interesujących nas punktów, a przy użyciu JOSM mogliśmy je odpowiednio edytować i eksportować do przyjaznego formatu *.json*. Całość była regularnie aktualizowana na zdalnym repozytorium na Githubi'e.

## 4 Symulacja zjawiska - implementacja

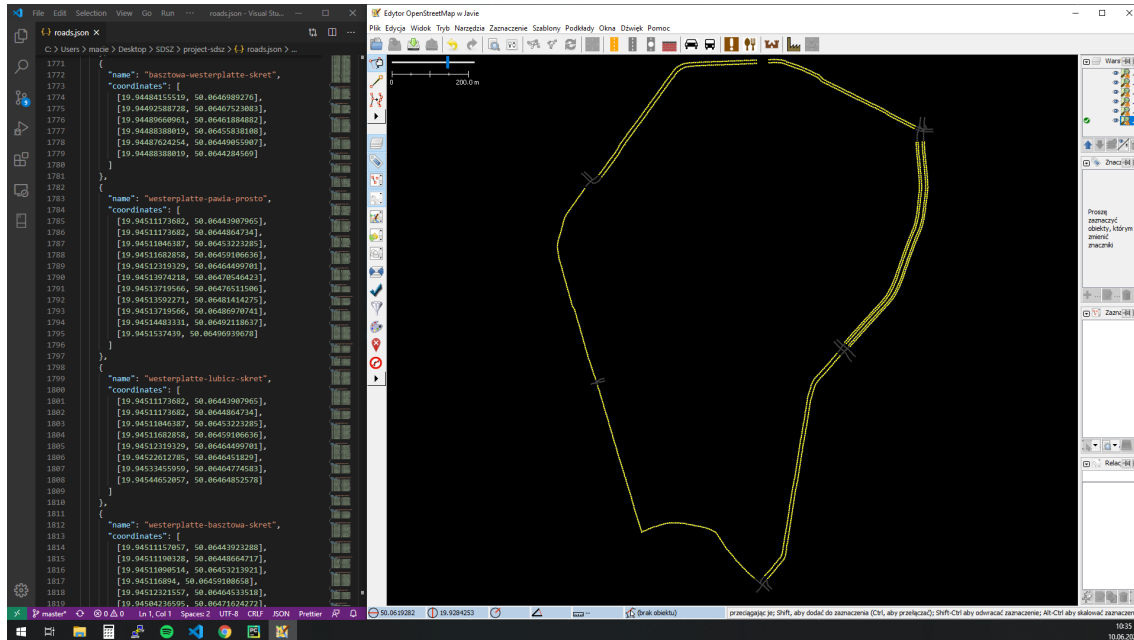
### 4.1 Przygotowanie siatki obwodnicy

Początek pracy rozpoczęliśmy od znalezienia i pobrania mapy w formacie *mapa.osm* ze strony OpenStreetMap.



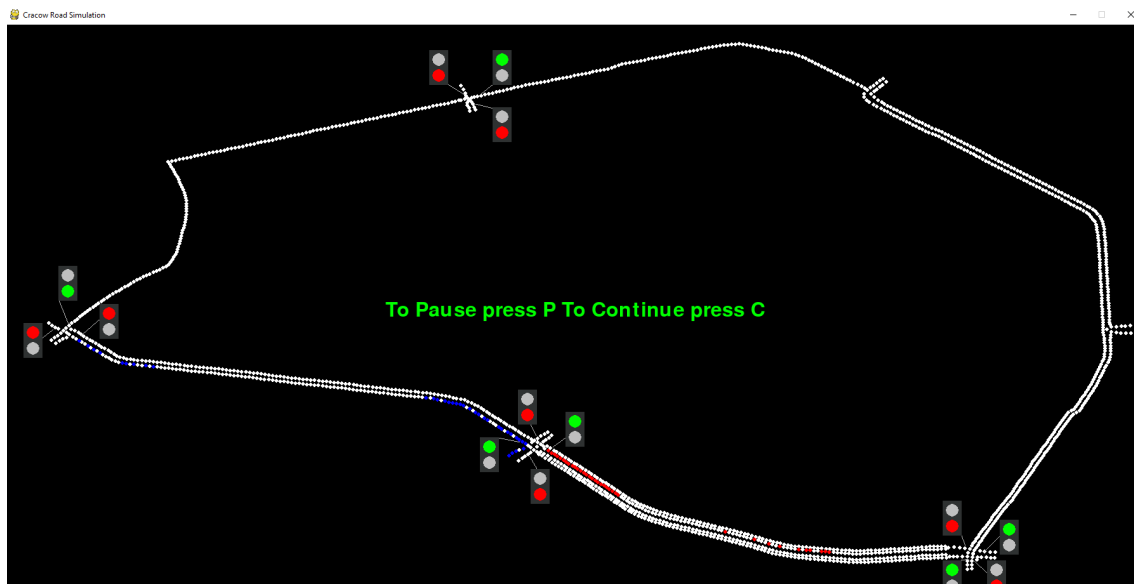
Rysunek 2: Strona internetowa OpenStreetMap

Następnie wykorzystaliśmy oprogramowanie JOSM, do którego zaimportowaliśmy mapę w formacie *mapa.osm* , a następnie nanieśliśmy punkty otrzymując w formacie *mapa.json* zbiór koordynatów. Współrzędne dzięki oprogramowaniu zostały ustawione w taki sposób, że każda komórka jest oddalona od siebie o 5m , gdyż tak założyliśmy w naszym modelu.



Rysunek 3: Docelowym plik *roads.js* oraz program JOSM

Po odpowiedniej konwersji punktów w pythonie oraz wykorzystaniu PyGame wizualizacja modelu wyglądała tak:

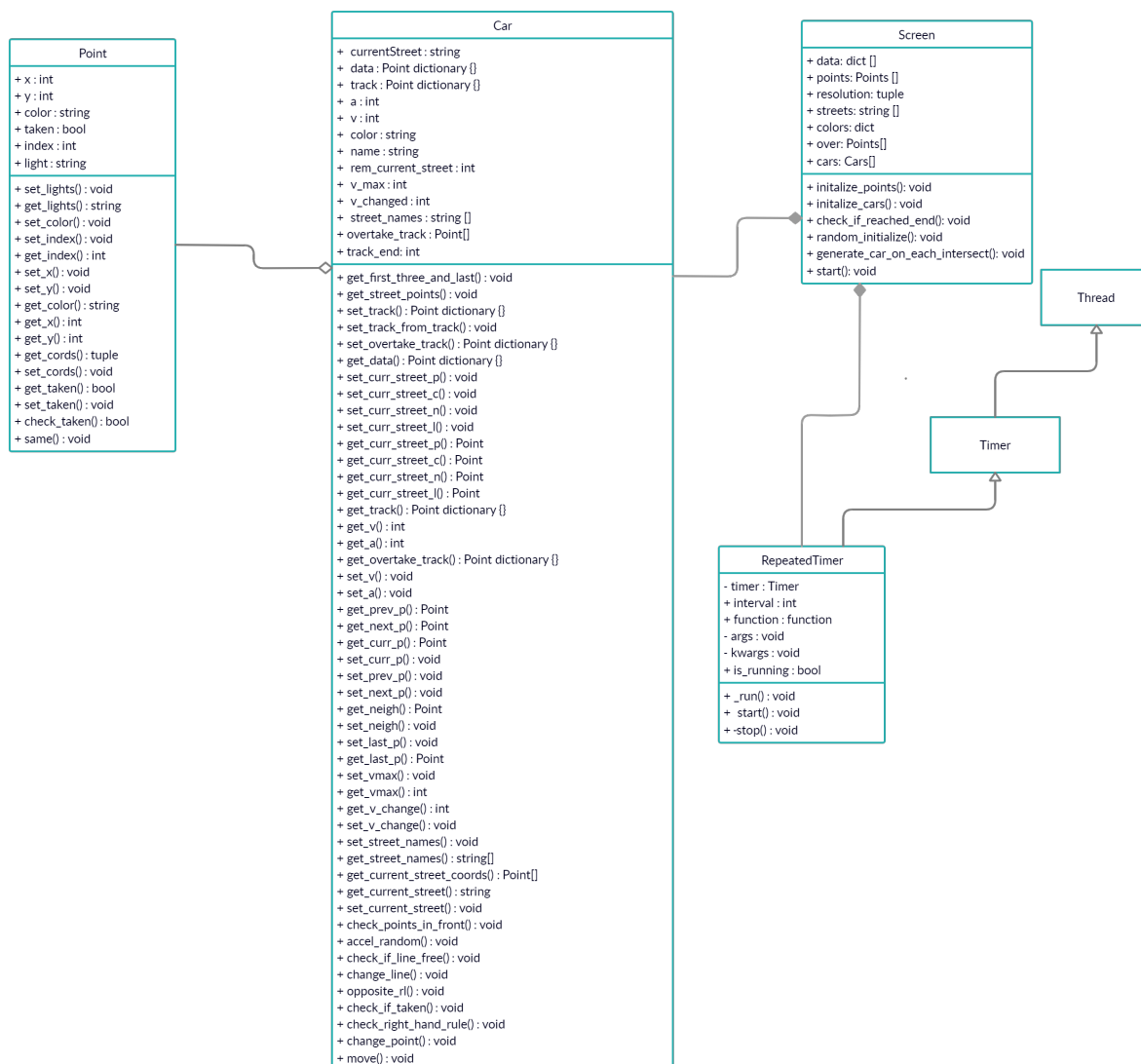


Rysunek 4: Wstępna wizualizacja modelu



## 4.2 Stworzenie logiki modelu

Poniżej został przedstawiony diagram UML klas naszego projektu, zasadniczo jak wi-  
dać najwięcej się dzieje w obiektach klasy Car, gdzie zostały zaimplementowane wszystkie  
warunki dotyczące ruchu, klasa Point posłużyła do pracy z punktami i stworzeniem siatki  
z komórek, klasa Screen odpowiada jedynie za wizualizację, a jako ostatnia użyta kla-  
sa RepeatedTimer odpowiada za liczenie czasu i m. in. obsługę działania świateł. Czas  
niejako jest liczony w osobnym wątku, tak aby nie ingerować w sam ruch pojazdów na  
obwodnicy.



Rysunek 5: Diagram UML klas

## 4.3 Opis klas Point i Car

Pokrótkie opiszemy najważniejsze pola i metody klas Point i Car.

### Klasa Point

#### Pola:

- x i y - skonwertowane współrzędne
- color - kolor punktu
- taken - stan punktu 0 (wolne) oraz 1 (zajęte)
- index - unikatowy indeks punktu
- light - jeżeli punkt jest wlotem na skrzyżowanie otrzymuje kolor świateł

#### Metody:

- set\_color() - ustawianie na stan 'red' bądź 'green'
- set\_taken() - ustawianie stanu punktu
- get\_cords() - pobieranie koordynatów w postaci krotki

### Klasa Car

#### Pola:

- prev\_p, curr\_p, next\_p - obiekty Point opisujące otoczenia i miejsce pobytu auta
- a, v, v\_max - kolejno przyspieszenie, prędkość chwilowa oraz prędkość max
- track, streets - zbiór Point'ów po których będzie jechał pojazd oraz jego wykaz ulic
- current\_street - ulica na której obecnie jest pojazd
- v\_changed - zmiana prędkości
- overtake\_track - zmiana trasy podczas ewentualnego wyprzedzania
- track\_end - flaga informująca o dojechaniu do celu <https://www.overleaf.com/project/5e821ea7208f>

#### Metody:

- check\_points\_in\_front() - metoda sprawdzająca tyle punktów przed przed sobą o ile w kolejnej iteracji miałyby się przemieścić pojazd, jeśli wykryje inny zwalnia zgodnie z przyjętym modelem.
- accel\_random() - metoda odpowiedzialna za tzw. element losowy czyli prawdopodobieństwo na hamowanie bądź przyspieszenie

- `check_if_line_free()` - metoda sprawdzająca czy pas ruchu obok jest wolny na ewentualne wyprzedzanie
- `change_lane()` - metoda odpowiedzialna za zmianę pasa
- `opposite_rl()` - metoda zwracająca równoległy przeciwny pas podczas wyprzedzania
- `check_right_hand_rule()` - metoda symulująca regułę prawej ręki
- `check_taken()` - metoda pomocnicza do ww sprawdzająca możliwość skrętu w lewo
- `change_point()` - metoda odpowiadająca za przesuwanie pojazdu
- `move()` - metoda skupiająca wszystkie warunki ruchu i ostatecznie przesuwająca pojazd

#### 4.4 Opis zasady działania algorytmu

- każdy pojazd zajmuje dokładnie jedną jedną komórkę o długości 5m (samochody jednoosobowe)
- każdy pojazd losuje trasę jaką pokona (wszystkie kombinacje tras zostały wygenerowane w osobnym pliku)
- każdy pojazd losuje prędkość maksymalną, z którą będzie się poruszał z przedziału (40 km/h - 1 komórka, 50 km/h - 2 komórki, 60 km/h - 3 komórki ) gdyż zakładamy iż rzeczywisty ruch na obwodnicy nie przekracza pewnej granicy, zważywszy na ograniczenia.
- w momencie rozpoczęcia symulacji wszystkie samochody rozpoczynają swój ruch i poruszają się do przodu o zadaną ilość krutek w przeliczeniu na odpowiednią prędkość. W czasie trwania ruchu zachodzi sprawdzanie wszelkich warunków oraz wykonywane są zdarzenia losowe zgodnie z przyjętym modelem.
- w czasie trwania ruchu każdy pojazd o ile ma możliwość to sprawdza i ewentualnie wyprzedza pojazd przed nim. Następnie z ustalonym prawdopodobieństwem zmienia ponownie pas na prawy o ile ma na to warunki.
- każdy pojazd przy wjeździe na skrzyżowanie sprawdza czy istnieją na nim światła i jeśli są to odpowiednio się zachowuje. W naszym modelu zakładamy dwa rodzaje światel czerwone i zielone. Zostały pominięte z strzałki warunkowe oraz światło żółte z powodów komplikacji implementacji
- kiedy pojazd dotrze do skrzyżowania docelowego zjeżdża z niego i zostaje usunięty ze zbioru pojazdów znajdujących się na obwodnicy. W przypadku mniejszej ilości pojazdów zostają losowo generowane kolejne samochody z pewnym zmiennym prawdopodobieństwem tak aby nie zaszła sytuacja, iż obwodnica jest pusta bądź praktycznie cała zapełniona.

## 4.5 Dane wejściowe i wyjściowe

### 4.5.1 Dane wejściowe

Przed rozpoczęciem symulacji możemy ustalić pewne dane wejściowe takie jak:

- początkowa liczba pojazdów na obwodnicy
- cykl sygnalizacji świetlnej każdego ze skrzyżowań
- czas trwania symulacji
- inflow - ilość pojazdów wjeżdżających na na każde ze skrzyżowań w odstępie czasu

### 4.5.2 Dane wyjściowe

W trakcie trwania symulacji otrzymujemy na bieżąco pewne informacje:

- prędkość średnia na obwodnicy  $V_{sr}$
- średnią prędkość maksymalną  $V_{sr_{max}}$
- różnica procentowa pomiędzy  $\frac{V_{sr_{max}}}{V_{sr}}$
- ilość pojazdów na obwodnicy
- czas trwania czy też numer iteracji
- outflow każdego ze skrzyżowań

Dzięki tym danym jesteśmy w stanie przeprowadzić badanie i obserwować wyniki co umożliwi nam dalsze formułowanie wniosków odnośnie ruchu na obwodnicy.

## 5 Seria symulacji i analiza wyników

Nasze badania są uwarunkowane o następujących danych wejściowych:

- czas trwania jednej symulacji dla pewnego zestawu danych wynosi 5min
- kolejnym parametrem jest inflow na każdym skrzyżowaniu, dobraliśmy tutaj wartości 2, 5 oraz 10 samochodów dla dwóch różnych czasów w jaki odbywają owe wjazdy tj. co 7 sekund i co 14 sekund. Co przekłada się na to, że w co każdą jednostkę czasu jest generowane na każdym wlocie 10, 25 lub 50 pojazdów.
- następną daną wejściową jest początkowa ilość samochodów generowana na losowych skrzyżowaniach obwodnicy

Numer symulacji	Liczba aut wjeżdżających na obwodnicę co 7 sekund	Liczba aut inicjowanych na losowych wlotach w momencie startu symulacji
1	10	100
2	25	100
3	50	100
4	10	500
5	25	500
6	50	500
7	10	1000
8	25	1000
9	50	1000

Numer symulacji	Liczba auto wjeżdżających na obwodnicę co 14 sekund	Liczba aut inicjowanych na losowych wlotach w momencie startu symulacji
10	10	100
11	25	100
12	50	100
13	10	500
14	25	500
15	50	500
16	10	1000
17	25	1000
18	50	1000

## 5.1 Statystyki

Poniżej zostały przedstawione tabele, ze statystykami prędkości średnich na początku ruchu oraz na końcu. Jak można zauważyć dla wszystkich naszych prób końcowe prędkości oscylują pomiędzy 25km/h - 30km/h. Może to być spowodowane faktem, iż w naszym modelu przyjęliśmy trzy rodzaje prędkości jakie samochód losuje tj. 40km/h, 50km/h i 60km/h. Na obwodnicy istnieje wiele dróg jednokierunkowych, oraz stosunkowo co nie wielkie odcinki co długości pojawiają się światła. Te uwarunkowania mogą być powodem takich wyników, gdyż samochody jadące szybciej nie mają zbyt częstej możliwości wyprzedzenia pojazdów jadących przed nimi więc teoretycznie wystarczy jeden pojazd jadący najwolniej, a automatycznie cała reszta jadąca za nim z  $V_{max}$  większym, mogłaby to zrobić, ale nie zawsze ma na to możliwość. Dodatkowo częste występowanie sygnalizacji oznacz i samochody przez kilkanaście sekunda posiadają zerową prędkość co również zaniża średnią.

Numer symulacji	Początkowa liczba pojazdów	Końcowa liczba pojazdów	Średnia prędkość początkowa	Średnia prędkość końcowa
1	100	178	35.51 km/h	26.46 km/h
2	100	263	39.48 km/h	27.56 km/h
3	100	396	37.62 km/h	26.36 km/h
4	300	179	28.75 km/h	25.69 km/h
5	300	348	28.84 km/h	24.56 km/h
6	300	484	28.71 km/h	25.64km/h
7	500	265	30.08 km/h	22.64 km/h
8	500	401	27.44 km/h	26.63 km/h
9	500	511	29.04 km/h	25.69km/h

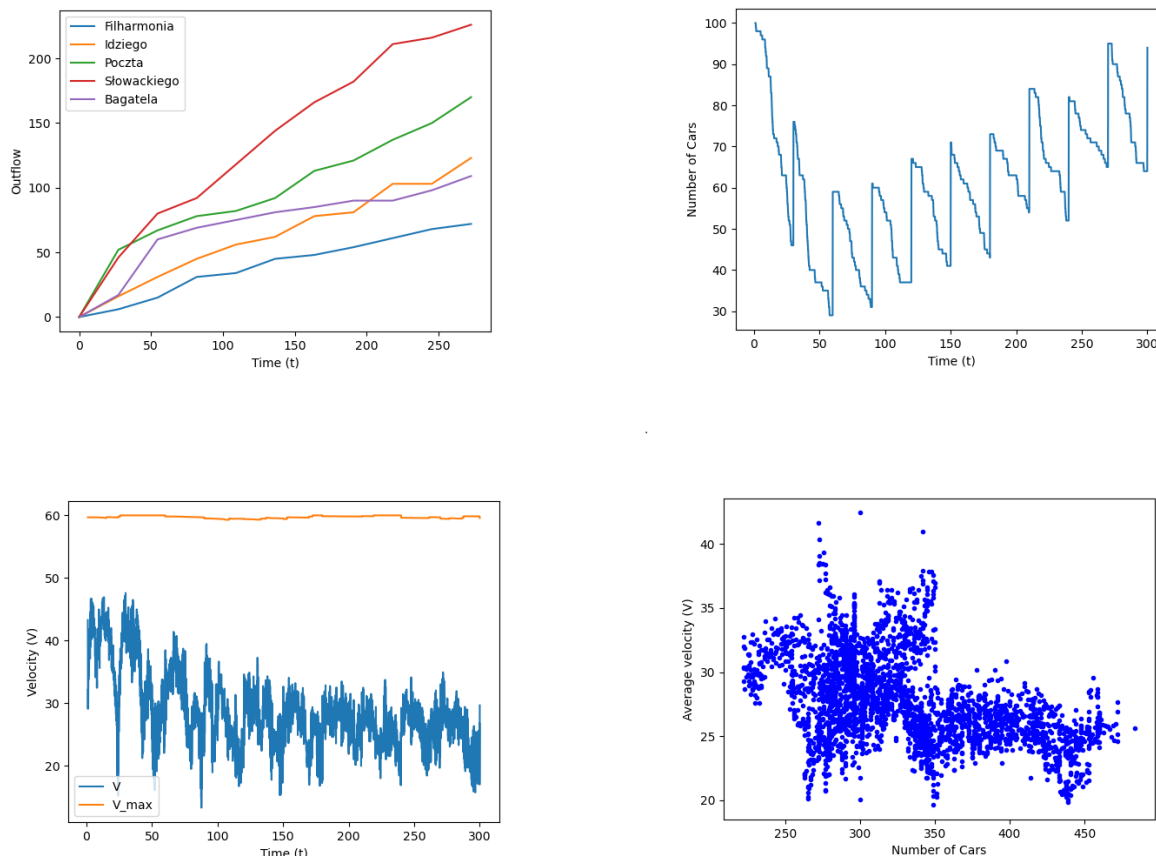
Numer symulacji	Początkowa liczba pojazdów	Końcowa liczba pojazdów	Średnia prędkość początkowa	Średnia prędkość końcowa
10	100	178	37.14 km/h	29.68 km/h
11	100	263	38.88 km/h	25.00 km/h
12	100	396	41.51 km/h	28.60 km/h
13	300	179	28.98 km/h	28.00 km/h
14	300	348	31.20 km/h	24.81 km/h
15	300	484	27.85 km/h	27.81 km/h
16	500	265	27.18 km/h	22.73 km/h
17	500	401	28.39 km/h	22.01 km/h
18	500	511	28.61 km/h	24.92 km/h

W poniższej tabeli zostały porównane wyniki *outflow'ów* minimalnych oraz maksymalnych, spośród wszystkich skrzyżowań. Można zauważyć, iż występuje duża rozbieżność, gdyż minimalne wartości są często zerem, a maksymalne w kilkunastu przypadkach wynoszą paręset pojazdów. Może to oznaczać, że pośród całego zbioru rozjazdów istnieje, takie które jest mniej oblegane, od pozostałych, ale również istnieje takie, które te wyniki zawyża. W dalszej części po analizie wykresów będziemy mogli zbadać lepiej.

Numer symulacji	Minimalny outflow spośród wszystkich skrzyżowań	Maksymalny outflow spośród wszystkich skrzyżowań
1	0	73
2	0	114
3	0	268
4	0	531
5	0	460
6	0	377
7	1	864
8	0	243
9	0	625
10	0	52
11	3	81
12	5	125
13	0	378
14	4	293
15	7	265
16	2	684
17	2	1486
18	0	1035

## 5.2 Wykresy

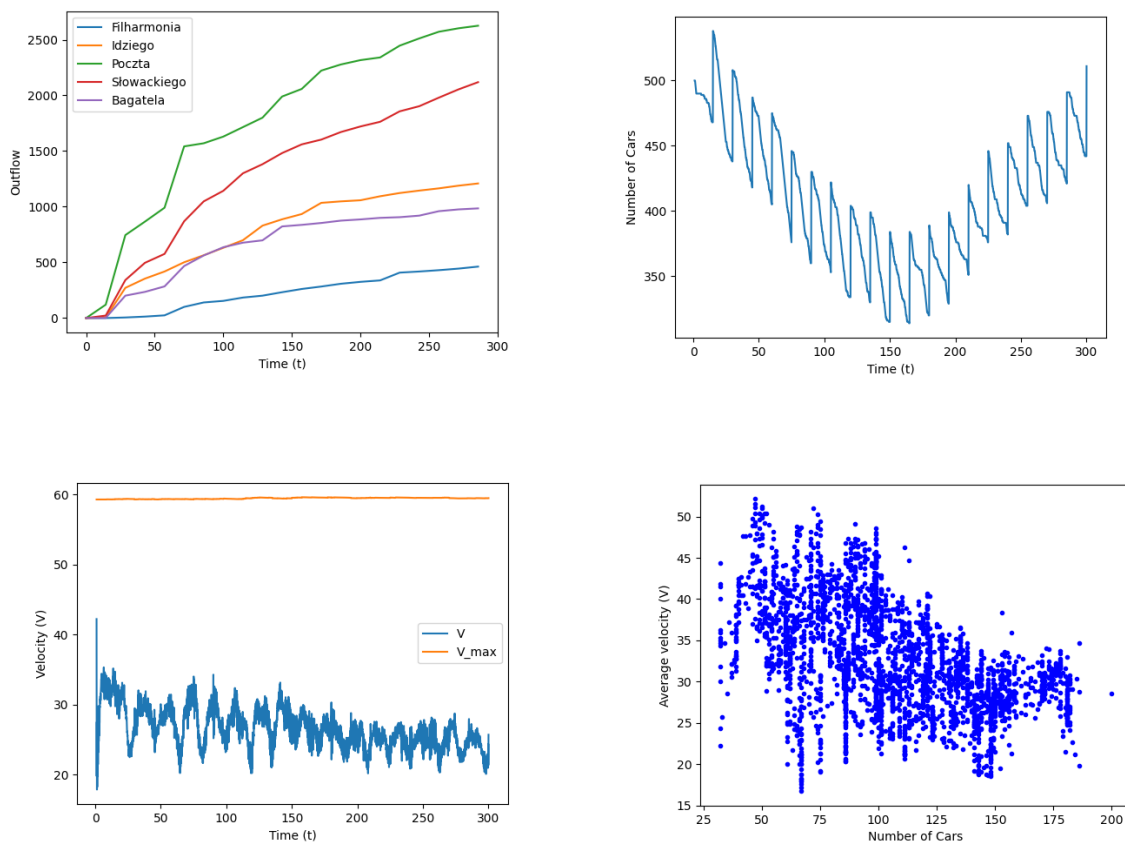
W dokumentacji umieszczono tylko kilka przykładowych wykresów dla wybranych zestawów danych wejściowych. Cały przegląd wykresów znajduje się w katalogu docs w repozytorium.



Rysunek 6: Wykresy zależności kolejnych danych zmieniających się w czasie tj. kolejno outflow, ilość pojazdów, prędkości i prędkości średniej. Inflow równy 3 co 30 sekund, początkowo 100 pojazdów, czas symulacji 5 min.

Powyższe wykresy dostarczają więcej informacji o ruchu. Po pierwszym wykresie można zauważyć iż dwa skrzyżowania tj. Słowackiego i na Poczcie są oblegane dużo częściej niż np. na Filharmonii. Kolejny wykres zachowuje pewnego rodzaju oscylacje jednak pnie się stopniowo do góry. Widoczne wzrosty i spadki sugerują, iż przy takich danych wejściowych, pewna część samochodów wjeżdża na obwodnicę, ale kilkadziesiąt sekund później następuje rozładowanie i ich zjazd, jednak w czasie i tak na obwodnicy liczba pojazdów wciąż wzrasta. Kolejny wykres również wydaje się być zgodny z realiami, gdyż im więcej pojazdów na drodze tym większy ruch, korki przez co prędkość, średnia spada. Widoczne oscylacje mogą sugerować zatrzymywania na skrzyżowaniach bądź sygnalizacji świetlnej.





Rysunek 7: Wykresy zależności kolejnych danych zmieniających się w czasie tj. kolejno outflow, ilość pojazdów, prędkości i prędkości średniej. Inflow równy 7 co 15 sekund, początkowo 500 pojazdów, czas symulacji 5 min.

W tym zestawie były dobrane inne dane wejściowe. Można zauważyć jak poprzednio z pierwszego wykresu iż ponownie najbardziej oblegane Skrzyżowania to Słowackiego czy też Poczta. Kolejny wykres ma zauważalne zagłębienie, które przedstawia spadek ilości samochodów na obwodnicy w pierwszych 3 min ruchu, jednak później następuje ich wzrost. Mógł to być element losowy gdyż pojazdy mogły w danym odstępie czasu na mniej obleganych skrzyżowaniach. Kolejne wykresy bardzo podobny do poprzednich.

## 6 Wnioski

W wyżej przedstawionych wynikach, można sugerować, iż przygotowana przez nas symulacja została przygotowana poprawnie. Wyniki wykresów oraz statystyk wydają się być zgodne z rzeczywistością, gdyż np. outflow na skrzyżowaniach Słowackiego i Pocztowej są zawsze na każdym z wykresów największe, a w rzeczywistości również, są to węzły na których jest duży ruch, a z kolei skrzyżowanie przy Filharmonii posiada mniejsze wartości tego współczynnika gdyż, z jednej strony jest możliwe tylko wyjazd z tego skrzyżowania bo druga strona posiada tylko drogę jednokierunkową, co zmniejsza liczbę pojazdów opuszczających w tym miejscu obwodnicę. Kolejnym argumentem jest zachowywanie się samochodów na drodze, gdyż im więcej pojazdów tym mniejszą prędkość średnią i vice versa. Mimo wszystko nasz program nie zawiera wielu równie ważnych założeń, które zdecydowanie mogłyby usprawnić działanie i sprawić symulację bardziej realną. Mowa tu o liniach tramwajowych, które de facto oplatają całą obwodnicę, czy też o samochodach towarowych (dłuższych niż przeciętna osobówka). Kolejnym problemem było rozeznanie, jak traktować, niektóre drogi, gdyż np. Podzamcze teoretycznie na Google Maps, czy też OpenStreetMap jest przejezdna, a z drugiej strony nie w Internecie oraz na StreetView po znakach można zauważyć, że jest dostępna dla służb miejskich, ale mimo to było tam widać jadące samochody.

## 7 Literatura

- (1) *Problematyka modelowania ruchu miejskiego z wykorzystaniem automatów komórkowych*, Jasroław Wąs, Rafał Bieliński, Bartłomiej Gajewski, Patryk Orzechowski
- (2) *Zeszyty naukowe Politechniki Śląskiej*, Marcin Bernaś, Barthmiej Płaczek
- (3) [https://pl.wikipedia.org/wiki/Obwodnice\\_Krakowa](https://pl.wikipedia.org/wiki/Obwodnice_Krakowa)
- (4) <https://www.openstreetmap.org/>
- (5) <https://josm.openstreetmap.de/>
- (6) <https://www.google.pl/maps/place/Kraków>
- (7) [https://en.wikipedia.org/wiki/Nagel-Schreckenberg\\_model](https://en.wikipedia.org/wiki/Nagel-Schreckenberg_model)