



**AGH**

**AKADEMIA GÓRNICZO-HUTNICZA  
IM. STANISŁAWA STASZICA  
W KRAKOWIE**

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

Praca dyplomowa inżynierska

Optymalizacja trasy z wykorzystaniem algorytmu przybliżonego

Autor: Rafał Siniewicz

Kierunek studiów: Automatyka i automatyka

Opiekun: dr inż. Joanna Kwiecień

Kraków, 2019/2020r.

## **Spis treści**

### **1. Wstęp**

- 1.1. Cel pracy
- 1.2. Zakres pracy
- 1.3. Wstęp teoretyczny

### **2. Opis projektu**

- 2.1. Użyte technologie
- 2.2. Opis aplikacji

### **3. Dokumentacja i działanie aplikacji**

### **4. Testy**

- 4.1. Sprawdzenie poprawności wczytywanych danych
- 4.2. Działanie dla różnych danych i sprawdzenie optymalności otrzymanego rozwiązania

### **5. Podsumowanie**

# 1. Wstęp

Transport od zawsze pełnił bardzo ważną rolę w życiu społeczeństw. W przeciągu wieków zmieniały się środki transportu, jego metody, a także szybkość przemieszczania ludzi oraz towarów. Za każdym jednak razem wyzwaniem stanowiło jak najoptymalniejsze przeprowadzenie procesu transportu. Szczególnie w dzisiejszych czasach zaplanowanie tras w taki sposób, aby zminimalizować czas ich pokonania- co wiąże się z mniejszymi kosztami oraz emisjami spalin- jest bardzo cenne. Również zwyklesze sytuacje, jak np. zaplanowanie dojazdu do pracy (komunikacją miejską lub samochodem) wymagają od wielu osób skorzystania z odpowiednich narzędzi (świadczą o tym chociażby popularność google maps czy aplikacji jakdojade) przeszukujących wiele możliwych tras i wybierających tą najlepszą. Widać zatem jak dużą rolę odgrywa w życiu niemalże każdego dobranie odpowiedniej trasy dla danej podróży. Problem optymalizacji trasy jest bardzo powszechny i wielokrotnie studiowany, zwłaszcza w badaniach operacyjnych, teorii grafów czy logistyce i dziedzinach pokrewnych. W teorii najczęściej dąży się do uzyskania jak najkrótszej trasy, choć problem można rozwinąć o wiele różnych czynników, które wpływają na przebieg trasy, np. jakość nawierzchni czy określona pojemność pojazdów.

## 1.1. Cel pracy

Praca ma na celu stworzenie aplikacji komputerowej wraz z interfejsem graficznym służącej do obliczania optymalnej trasy na podstawie odpowiednich danych. Optymalna trasa zostanie obliczona przy pomocy odpowiednich algorytmów opisanych w poniższym wstępie teoretycznym. Program będzie również wyświetlał przebiegi tras w formie graficznej wraz z zaznaczeniem poszczególnych punktów na mapie. Całość ma stanowić ujednolicony i łatwy w użyciu system, w którym można ustalać parametry, takie jak, np. ilość pojazdów i ich pojemności, współrzędne geograficzne punktów, a także uzyskać informacje o długości trasy, jej przebiegu i innych statystykach dotyczących rozwiązywanego problemu.

## 1.2. Zakres pracy

W pracy zagadnienie optymalizacji trasy rozważono dla problemu marszrutyzacji, czyli VRP (Vehicle Routing Problem) oraz CVRP (Capacity Vehicle Routing Problem). Projekt skupia się głównie na dwóch czynnikach decydujących o optymalności: długości trasy oraz zapełnieniu pojazdów ładunkiem.

Przy rozwiązywaniu problemu marszrutyzacji korzystano z algorytmów przybliżonych, a konkretniej algorytmu genetycznego, przede wszystkim ze względu na złożoność problemu, gdyż przy problemach bardziej złożonych dobrze sprawdzają się algorytmy przybliżone. Algorytm genetyczny wybrano również ze względu na jego uniwersalność, możliwość uzyskania stosunkowo dobrych wyników w krótkim czasie oraz prostą koncepcję podejścia do rozwiązania.

### 1.3. Wstęp teoretyczny

Na podstawie [1] oraz [2]:

VRP (Vehicle Routing Problem)- problem kombinatoryczny bazujący na zbiorach krawędzi i wierzchołków grafu  $G(V,E)$ . Jest on rozszerzeniem problemu komiwojażera czy problemu chińskiego listonosza. Jest to problem z kategorii NP- trudnych. Notacja:

- $V$  jest zbiorem punktów do odwiedzenia i punktu początkowego,
- $E$  jest zbiorem krawędzi łączących punkty  $V$ ,
- $d$  jest wektorem pojemności towarów w punktach,
- $k$  jest liczbą tras,
- $C$  to pojemność pojazdu,

Ogólnie dla problemu CVRP rozwiązanie składa się z:

- zbioru  $\{R_1, \dots, R_k\}$  z  $V$  takiego, że:  $\sum_{j \in R_i} d_j \leq C, \quad 1 \leq i \leq k$
- permutacji  $\theta_i$  z  $R_i \cup \{0\}$  określającej kolejność odwiedzonych punktów w trasie  $i$

Istnieją różne warianty problemu marszrutyzacji, na przykład:

- VRP with Time Windows- czyli uwzględnienie okien czasowych odbioru/ wysłania towaru
- Split Delivery VRP- czyli możliwość obsługi jednego klienta przez kilka pojazdów
- Vehicle Routing Problem with Multiple Trips (VRPMT)- czyli możliwość pokonania więcej niż jednej trasy przez jeden pojazd
- Capacitated Vehicle Routing Problem: CVRP- pojazdy mają ograniczoną pojemność
- inne

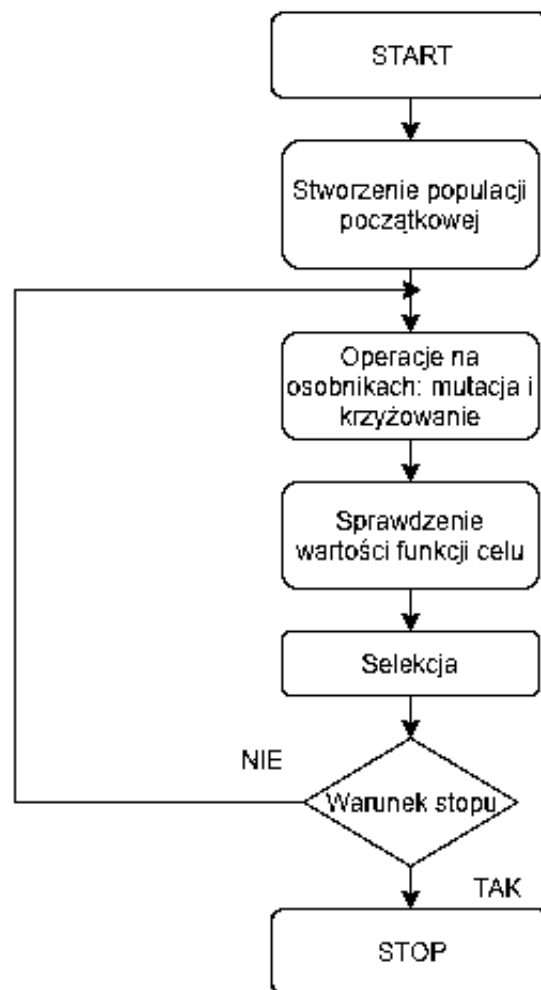
Powyższy problem można również rozwinąć o wiele innych czynników, jak np. kolejność odwiedzenia miejsc czy opcjonalnego odwiedzenia niektórych punktów lub funkcja kosztu może rozpatrywać różne parametry, np. czas wykonania zleceń czy ilość przewiezionego ładunku. Widać zatem, że problem marszrutyzacji jest zagadnieniem bardzo rozległym, a przy tym elastycznym, tzn. można go dostosować do wielu różnych problemów.

Na podstawie [3]:

Termin algorytm genetyczny po raz pierwszy wprowadził John Holland, bazując na koncepcji Darwina- teorii ewolucji. Jest to metaheurystyka (metoda poszukiwania rozwiązań, która nie gwarantuje znalezienia optymalnego rozwiązania) bazująca na zjawisku naturalnej selekcji (teorii ewolucji gatunków), a także dziedziczności. Należy do szerszej grupy- algorytmów ewolucyjnych. Algorytm ten polega na znajdowaniu rozwiązania naśladując zjawiska występujące w środowisku naturalnym, takie jak: mutacje, krzyżowania gatunków, a także selekcja.

Algorytm genetyczny w krokach:

1. Utworzenie początkowej populacji
2. Przeprowadzanie mutacji i krzyżowania
3. Sprawdzenie funkcji celu dla osobników
4. Wybór najlepszych osobników poprzez selekcję
5. Powtarzanie kroków 2-4 aż do spełnienia warunku stopu
6. Koniec algorytmu i wybór najlepszego osobnika



Rysunek1. Schemat algorytmu genetycznego

## 2. Opis projektu

### 2.1. Użyte technologie

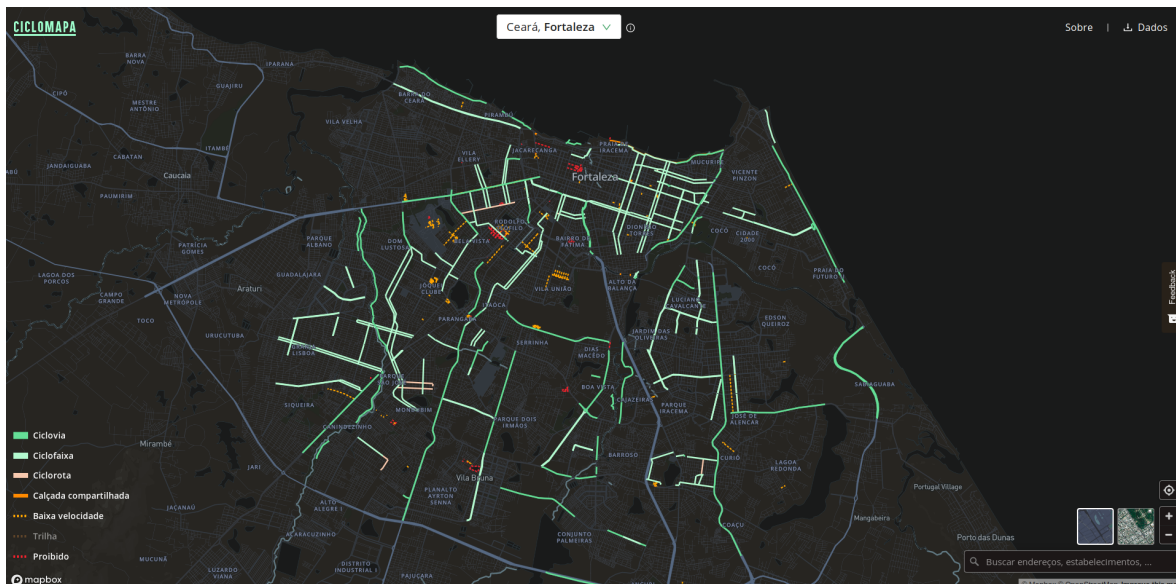
Na podstawie [4]:

**Python**- jest językiem programowania ogólnego przeznaczenia. Jest to język wysokiego poziomu. Może być skutecznie wykorzystywany przy budowie w zasadzie każdego programu. Nie potrzebuje bezpośredniego dostępu do sprzętu komputerowego. Python nie jest optymalny dla programów, które mają duże ograniczenia niezawodności lub są zbudowane i utrzymywane przez wiele osób lub przez długi czas. Ma on jednak kilka zalet w porównaniu z innymi językami programowania, m.in.:

- jest łatwy w nauce
- posiada bardzo dużą liczbę darmowych i powszechnie dostępnych bibliotek, zapewniających rozszerzoną funkcjonalność

Na podstawie [5]:

**OSM (OpenStreetMap)**- jest to darmowa, edytowalna mapa całego świata, tworzona przez wolontariuszy, wydawana na licencji open-content. Licencja OpenStreetMap pozwala na bezpłatny dostęp do obrazów i podstawowych danych map. Projekt ma na celu promowanie nowych zastosowań tych danych. W pracy OSM zostało wykorzystane do zwizualizowania przebiegów tras i pokazania ich na rzeczywistej mapie. Przykład obrazu uzyskanego dzięki OSM [6]:



Rysunek2. Przykładowy screen z osm

Biblioteki:

## Źródła

- [1] Dantzig, George Bernard; Ramser, John Hubert (October 1959). "The Truck Dispatching Problem", <https://andresjaquep.files.wordpress.com/2008/10/2627477-clasico-dantzig.pdf>
- [2] T. Ralphs, J. Hartman and M. Galati. "Capacitated Vehicle Routing and Some Related Problems". Some CVRP Slides. Rutgers University. 2001
- [3] D. E. Goldberg: Algorytmy genetyczne i ich zastosowania. Warszawa: WNT, 1998. (pol.)
- [4] Guttag, John V. (12.08.2016). Introduction to Computation and Programming Using Python: With Application to Understanding Data. MIT Press. ISBN 978-0-262-52962-4.
- [5] [https://wiki.openstreetmap.org/wiki/About\\_OpenStreetMap](https://wiki.openstreetmap.org/wiki/About_OpenStreetMap)
- [6] [https://wiki.openstreetmap.org/wiki/File:Screenshot\\_2019-10-23\\_CicloMapa.png](https://wiki.openstreetmap.org/wiki/File:Screenshot_2019-10-23_CicloMapa.png)