

Matematyka w projektowaniu sieci i systemów

Projekt optymalizatora kosztów przejazdu pomiędzy dwoma punktami

Autorzy:

Jakub Bobko

Patryk Dzióbek

Anna Gancarczyk

Dariusz Grabowski

Marcin Rys

1. Przedstawienie problemu i podział pracy

Celem projektu jest optymalizacja kosztów transportu pomiędzy dwoma punktami, by spełniała podane wymagania. Obliczanie najtańszej ścieżki jest uzależnione od fizycznego położenia węzłów. Metryka kosztu drogi jest więc zależna od odległości fizycznej między punktem początkowym i końcowym.

W projekcie model w języku GMPL wykonał Jakub Bobko oraz Dariusz Grabowski. Algorytm heurystyczny w języku C++ przygotowany został przez Annę Gancarczyk. Frontend został przygotowany przez Marcina Rysa. Za część backendową odpowiadał Patryk Dzióbek, a za dane wejściowe Dariusz Grabowski.

2. Opis rozwiązania GMPL

Model GMPL opisuje minimalizacyjny problem liniowy, którego celem jest znalezienie najtańszej ścieżki łączącej wybrane miasta oraz dobór odpowiedniej floty pojazdów. Sformułowanie jakie zostało zastosowane to Link-Path, czyli ścieżka-łącze. Charakterystyczną cechą takiego podejścia jest występowanie tzw. delty, czyli macierzy określającej, która ścieżka używa konkretnego łącza. Aby wyznaczenie delty przebiegło szybko i bezbłędnie, został wykorzystany parser służący do wyciągania ścieżek z danych. Zaimplementowany model matematyczny do rozwiązania naszego problemu, został połączony ze skryptem generującym dane ze ścieżkami.

Po wyborze punktu początkowego i docelowego generowany jest plik z danymi, w którym zostały zdefiniowane dostępne ścieżki, rozmiary pojazdów z kosztami ich wykorzystania, ilość ładunku do przetransportowania, kilometry na każdej ścieżce oraz koszt przejazdu autostradą. Zakładamy, że jedno łącze nie ma limitu jednostek które jest w stanie przenieść. Lokalizacja węzłów jest określona a priori, dzięki temu nie ma potrzeby liczenia odległości pomiędzy miastami.

W celu uproszczenia modelu założyliśmy jedną, niezmienną cenę benzyny jak i stałą wartość spalania paliwa na 100 km. Istnieje możliwość wyboru różnych pojazdów o różnych pojemnościach jednostek:

Rodzaj pojazdu	Pojemność pojazdu	Koszt pojazdu
Mały	2	2
Średni	10	8
Duży	32	25

Funkcja celu przedstawia się następująco:

$$\begin{aligned} \text{minimize } z: & \sum_{e \in E, d \in D, p \in P} (KSI[e] * LAM[e, d, p] * x[d, p] * 4.95 * 10 / 100 + \\ & KSA[e] * LAM[e, d, p] * x[d, p]) + \\ & \sum_{e \in E, k \in K} g[k] * f[e, k] \end{aligned}$$

gdzie:

KSI - opłata za autostradę

LAM - macierz delta

KSA - liczba kilometrów na trasie

x - łączy

g - koszt wykorzystania danego pojazdu

f - modularność pojazdów

4,95 - przyjęta cena benzyny

10/100 - przyjęte średnie spalanie na 100 km

Minimalizowana jest suma kosztów poszczególnych łączy, koszt wyboru pojazdu oraz ewentualne wykorzystanie autostrady.

Ograniczenia:

$$\sum_{p \in P} x[d, p] \leq \sum_{e \in E, k \in K} m[k] * f[e, k]$$

przesyłane zapotrzebowanie musi być mniejsze, bądź równe pojemności pojazdów,

$$h[d] == \sum_{p \in P} x[d, p]$$

zapotrzebowanie, które wychodzi z miasta A musi być równe zapotrzebowaniu, które wchodzi do miasta B.

3. Opis rozwiązania heurystycznego

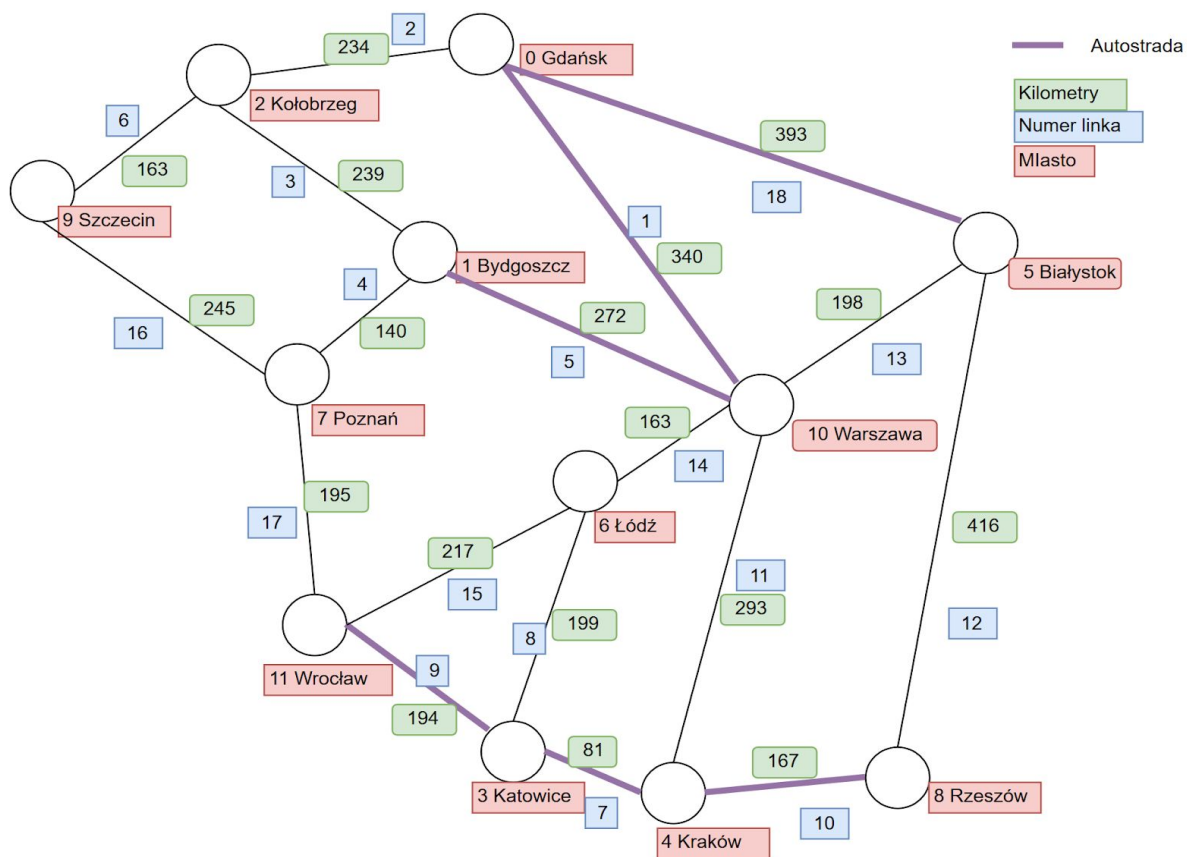
Heurystyka jest metodą, która pozwala algorytmowi na znalezienie jak najlepszych rozwiązań. Algorytm ma za zadanie zoptymalizować podaną topologię pod kątem realizacji wszystkich wymaganych ograniczeń. Przykładowo - chcemy przemieścić się z punktu A do punktu B, ale chcemy aby koszt tej trasy był jak najmniejszy. Podajemy algorytmowi wszystkie możliwe ścieżki dojścia z punktu A do punktu B, a on wylicza, która ścieżka będzie miała najniższy koszt, czyli będzie optymalna.

Podchodząc do heurystyki musimy wybrać rozwiązania, które będą spełniały wszystkie warunki zadania (w miarę możliwości jak najlepsze). Najlepiej, aby weryfikacja założeń była napisana funkcją celu, która sprawdza, czy zestaw zmiennych decyzyjnych spełnia wszystkie ograniczenia. Funkcja ta wiedząc w jakim stanie próbujemy szukać rozwiązania przeskoczy do innego i sprawdzi czy jest lepsze od poprzedniego, jeśli tak to wybierze to lepsze rozwiązanie.

Niestety nie udało nam się wykonać rozwiązania heurystycznego, ale też nasz problem nie jest na tyle skomplikowany, aby było ono konieczne do wykonania. To co udało nam się zrobić zostanie przesłane wraz z resztą kodu.

4. Wyniki

W celu łatwiejszego analizowania wyników narysowaliśmy graficznie topologię zawierającą wszystkie potrzebne informacje takie jak: autostrady, odległość pomiędzy poszczególnymi miastami, nazwa miasta i numer identyfikacyjny linka.



Scenariusz 1. Podróż z Białegostoku do Wrocławia z ładunkiem 16 jednostek

Trasa 1: Białystok - Gdańsk - Kołobrzeg - Bydgoszcz - Poznań - Wrocław

Trasa 2: Białystok - Rzeszów - Kraków - Katowice - Wrocław

Trasa 3: Białystok - Gdańsk - Kołobrzeg - Szczecin - Poznań - Wrocław

Trasa 4: Białystok - Gdańsk - Warszawa - Kraków - Katowice - Wrocław

W wyniku optymalizacji uzyskaliśmy wynik:

Koszt	35230
Liczba małych pojazdów	3
Liczba średnich pojazdów	1
Liczba dużych pojazdów	0
Nadmiarowa pojemność	0
Optymalna trasa	1

Scenariusz 2. Podróż z Kołobrzegu do Krakowa z ładunkiem 27 jednostek

Trasa 1: Kołobrzeg - Gdańsk - Białystok - Rzeszów - Kraków

Trasa 2: Kołobrzeg - Bydgoszcz - Warszawa - Kraków

Trasa 3: Kołobrzeg - Gdańsk - Warszawa - Kraków

Trasa 4: Kołobrzeg - Gdańsk - Białystok - Warszawa - Kraków

Trasa 5: Kołobrzeg - Szczecin - Poznań - Bydgoszcz - Warszawa - Kraków

Trasa 6: Kołobrzeg - Bydgoszcz - Poznań - Wrocław - Katowice - Kraków

Trasa 7: Kołobrzeg - Bydgoszcz - Warszawa - Łódź - Katowice - Kraków

W wyniku optymalizacji uzyskaliśmy wynik:

Koszt	48732
Liczba małych pojazdów	0
Liczba średnich pojazdów	3
Liczba dużych pojazdów	0
Nadmiarowa pojemność	3
Optymalna trasa	2

5. Wnioski

Przeprowadzanie analizy rozwiązania pozwala na sformułowanie następujących wniosków:

- stworzenie modelu rozwiązania w języku GMPL pozwala obniżyć koszt optymalizacji topologii drogowej,
- należy pamiętać, że tworzenie modelu zmusza nas do zastosowania pewnych uproszczeń, które mogą mieć wpływ na wynik końcowy optymalizacji.