Mateusz Chechliński

Michał Padzik

Zastosowanie Google Maps API w rozwiązywaniu DVRP

Projekt

# Cel

Celem projektu jest przygotowanie aplikacji, który posłuży zademonstrowaniu możliwości wykorzystania usługi map firmy Google w rozwiązywaniu problemu DVRP. Aplikacja udostępni interface webowy, który pozwoli internautom zapoznać się z zagadnieniem DVRP oraz samodzielnie przetestować jego działanie.

# Opis problemu

Problem marszrutyzacji to problem decyzyjny polegający na wyznaczeniu optymalnych tras przewozowych dla pewnej ściśle określonej liczby środków transportu, której zadaniem jest obsłużenie zbioru klientów znajdujących się w różnych punktach przy zachowaniu ograniczeń. Kryterium optymalizacji jest całkowity koszt transportu (wyrażony odległościowo, cenowo lub czasowo).

Dynamic Vehicle Routing Problem, czyli w skrócie DVRP, to dynamiczna odmiana klasycznego problemu marszrutyzacji, cechująca się tym, że w momencie rozpoczęcia obliczeń nie są znani wszyscy klienci. Wraz z upływem dnia, klienci składają nowe zamówienia, które również muszą zostać obsłużone przy możliwie najniższym koszcie. Podobnie jak klasyczny problem marszrutyzacji, DVRP zalicza się do klasy problemów NP-trudnych.

# Architektura aplikacji

Aplikacja będzie składać się z trzech podstawowych modułów: Interface’u użytkownika, WebService’u oraz silnika optymalizującego.

## Interface użytkownika

Użytkownik będzie mógł korzystać z aplikacji za pomocą przeglądarki WWW. Dane wprowadzane będą ręcznie wpisując z klawiatury kolejne lokalizacje zamówień i miejsce magazynu, lub poprzez wyklikanie tych miejsc na mapie googla.

Do pewnego momentu po rozpoczęciu obliczeń użytkownik będzie mógł dodawać nowe zamówienia, oraz aktualizować obliczenia uwzględniając nowe miejsca.

Obliczana na bieżąco trasa będzie wyrysowywana automatycznie na mapie i uaktualniania w czasie działania algorytmu. Ponadto użytkownik będzie miał możliwość przejrzenia historii obliczania trasy.

## WebService

WebService będzie warstwą pośredniczącą pomiędzy silnikiem optymalizującym, a interfacem użytkownika. Będzie odpowiedzialny za przekazywanie do silnika informacji o nowych zamówieniach, zmianach konfiguracji czy żądaniach rozpoczęcia optymalizacji, użytkownikowi zaś przekaże zoptymalizowane trasy przejazdu pojazdów.

## Silnik optymalizujący

Silnik optymalizujący stanowi rdzeń aplikacji. Jest on implementacją rozwiązania zaproponowanego przez zespół badaczy z Lille i Malagi pod kierunkiem M. R. Khouadija pod nazwą Multi-Environment Multi-Swarm Optimizer (MEMSO). Służy on wydajnej i relatywnie szybkiej optymalizacji problemu DVRP.

# Google Maps

Usługa map Google będzie wykorzystywana w aplikacji w dwojaki sposób. Po pierwsze, mapa będzie stanowiła kluczowy element interface’u użytkownika. To tam będą prezentowane wszystkie zamówienia złożone przez wirtualnych klientów oraz trasy dla pojazdów powstałe w wyniku optymalizacji.

Po drugie, silnik optymalizujący w swoich obliczeniach będzie posługiwał się funkcją kosztu opartą na czasie i odległości pomiędzy poszczególnymi klientami pobieranymi z Google Maps API. Aby nie przekroczyć limitów zapytań, silnik posługiwał się będzie grafem rzadkim. Długość trasy pomiędzy zamówieniami niepołączonymi krawędzią w grafie będzie wyznaczana albo jako najkrótsza ścieżka pomiędzy danymi wierzchołkami grafu, albo jako odległość w linii prostej pomnożona przez pewien ustalony współczynnik.

# Dane wejściowe

Aplikacja pozwala użytkownikowi zdefiniować lokalizację i godziny otwarcia bazy, a także liczbę i pojemność pojazdów, przy czym cała flota jest homogeniczna. Oprócz tego użytkownik może przed rozpoczęciem pracy optymalizatora, a także w jej trakcie dodawać nowe zamówienia od klientów. Każde zamówienie składa się z lokalizacji oraz rozmiaru wyrażonego w umownych jednostkach (tych samych, co pojemność pojazdu) i staje się dostępne w momencie dodania – nie definiuje się okien czasowych. Po upływie połowy symulowanego dnia, blokowana jest możliwość dodawania nowych zamówień.

Alternatywnie użytkownik będzie mógł wczytać uprzednio wprowadzone i zapisane dane. Wczytane dane zostaną dołączone do już wprowadzonych.

Jako lokalizację rozumie się współrzędne geograficzne wskazane przez: kliknięcie na mapie, wpis w benchmarku lub geokodowanie adresu.

Użytkownik może również wybrać z dostępnych opcji czas trwania symulacji, czyli jak długi ma być wirtualny dzień. W zależności od wybranej opcji, dzień będzie podzielony na więcej lub mniej przedziałów, co wynika z konieczności zarezerwowania odpowiednio dużej ilości czasu na wykonanie obliczeń. W trakcie tworzenia aplikacji ustalone zostaną eksperymentalnie proporcje długości czasu symulacji i ilości przedziałów czasowych, na które wirtualny dzień zostanie podzielony.

# Funkcjonalność

Aby rozpocząć pracę, użytkownik musi zdefiniować parametry bazy i floty pojazdów (jej rozmiar musi wynosić minimum 1). Nie jest wymagane definiowanie zamówień przed rozpoczęciem pracy silnika optymalizującego – można zrobić to w trakcie, symulując scenariusz, w którym nie ma zamówień z dnia poprzedniego, w ciągu dnia pojawiają się za to nowe zamówienia.

Silnik optymalizujący dzieli dzień na ustaloną liczbę przedziałów i w każdym przedziale czasowym uruchamiana jest osobna optymalizacja, dla wszystkich zamówień znanych w chwili rozpoczęcia danej tury optymalizacji.

Każde rozwiązanie cząstkowe, tj. będące rezultatem pojedynczej optymalizacji dla danego przedziału czasowego, jest prezentowane na mapie wraz z hipotetycznym położeniem pojazdów w danej chwili – specyfika problemu DVRP zakłada, że pojazdy mogą już być w drodze, podczas gdy dalsza część ich trasy jest ciągle optymalizowana.