

AKADEMIA GÓRNICZO - HUTNICZA

WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,  
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ

MATEMATYCZNE METODY WSPOMAGANIA DECYZJI

---

# Projektowanie tras biegowych w mieście zrealizowane z wykorzystaniem algorytmu Taboo Search

---

*Projekt zrealizowali:*

Żaneta BŁASZCZUK

Jakub PORĘBSKI

*Prowadzący:*

Dr inż. Piotr KADŁUCZKA

12 stycznia 2015



# Spis treści

|          |                                      |          |
|----------|--------------------------------------|----------|
| <b>1</b> | <b>Abstract</b>                      | <b>2</b> |
| <b>2</b> | <b>Opis problemu</b>                 | <b>2</b> |
| <b>3</b> | <b>Parametry modelu</b>              | <b>2</b> |
| 3.1      | Uproszczenia i uogólnienia . . . . . | 2        |
| 3.2      | Parametry i zależności . . . . .     | 2        |
| 3.3      | Postać rozwiązania . . . . .         | 2        |
| 3.4      | Funkcja celu . . . . .               | 2        |
| 3.5      | Ograniczenia . . . . .               | 3        |
| <b>4</b> | <b>Implementacja algorytmu</b>       | <b>3</b> |
| 4.1      | Postać danych . . . . .              | 3        |
| 4.2      | Struktury danych . . . . .           | 4        |
| 4.2.1    | Klasa graph . . . . .                | 4        |
| 4.3      | Interface użytkownika . . . . .      | 4        |
| <b>5</b> | <b>Testowanie algorytmu</b>          | <b>5</b> |
| 5.1      | Test 1 . . . . .                     | 5        |
| <b>6</b> | <b>Analiza efektywności</b>          | <b>5</b> |
| <b>7</b> | <b>Podsumowanie</b>                  | <b>5</b> |

# 1 Abstract

Sprawozdanie piszemy w formie bezosobowej: wykonano, zrobiono, obliczono

## 2 Opis problemu

Podczas planowania trasy biegowej często nie chcemy dobiec do danego punktu najkrótszą możliwą drogą, lecz drogą o określonej długości. Jednocześnie jeżeli teren po którym biegamy jest pagórkowaty możemy unikać zmian wysokości, bądź wręcz przeciwnie ich oczekiwać. Równocześnie wiedząc o szkodliwości biegania po betonie możemy chcieć zminimalizować ilość dróg betonowych na naszej trasie.

## 3 Parametry modelu

Celem algorytmu jest znalezienie optymalnej trasy o długości najbardziej zbliżonej do zadanej przez użytkownika z uwzględnieniem parametru wysokości oraz ilości betonu na trasie biegu.

### 3.1 Uproszczenia i uogólnienia

W trakcie implementacji algorytmu przyjęto założenie, że badany graf jest grafem planarnym, który dobrze opisuje układ dróg w mieście. Algorytm przybliża długości dróg do linii prostych, nie uwzględniając zakrętów na trasie. Równocześnie przyjęto, że atrakcyjność danej trasy zależy wyłącznie od różnicy wysokości na trasie oraz od ilości dróg asfaltowych na trasie.

### 3.2 Parametry i zależności

Głównymi parametrami trasy jest mapa krawędzi, po których może przebiegać trasa. Zawiera ona jednocześnie informacje o wysokości w każdym węźle grafu oraz o rodzaju nawierzchni.

Do parametrów algorytmu zaliczamy również zadaną przez użytkownika odległość oraz preferencje własne dotyczące trasy.

### 3.3 Postać rozwiązania

Rozwiązaniem naszego problemu jest uporządkowana lista krawędzi grafu, która osiąga minimalną wartość funkcji celu.

### 3.4 Funkcja celu

Minimalizowano funkcję celu problemu poszukiwania optymalnej trasy biegowej, która wygląda następująco:

$$f(d, h, b) = w_1 \left| \bar{d} - \sum_{i=1}^n d_i \right| + w_2 \sum_{i=1}^n |\bar{h} - |h_i|| + w_3 \left| \bar{b} - \frac{\sum_{i=1}^n b_i}{n} \right| \quad (1)$$

$d_i > 0$  – długość  $i$ -tej krawędzi  
 $h_i$  – różnica wysokości na  $i$ -tej krawędzi  
 $b_i$  – rodzaj nawierzchni na  $i$ -tej krawędzi (0 – nawierzchnia dobra, 1 – nawierzchnia betonowa)  
 $\bar{d}$  – zadana wartość drogi  
 $\bar{h}$  – zadana wartość różnicy wysokości na każdej krawędzi  
 $\bar{b}$  – zadana ilość dróg o nawierzchni betonowej w %  
 $w_1, w_2, w_3$  – wagi dobierane przez użytkownika

### 3.5 Ograniczenia

- a) Biegacz nie przebiega dwukrotnie po tej samej drodze.
- b) Maksymalna długość trasy biegu wynosi 1000000 jednostek (metrów). Jest to ograniczenie obliczeniowe algorytmu
- c)

## 4 Implementacja algorytmu

Algorytm Taboo Search

### 4.1 Postać danych

Dane w trakcie wykonywania algorytmu są przechowywane w kilku macierzach przyległości. Program w celu rozpoczęcia pracy potrzebuje 7 plików wejściowych z kolejnymi strukturami danych:

#### 1. Współrzędne wierzchołków

- rozszerzenie: .xy
- Pierwsza linia:  $h \times w$  – wysokość i szerokość grafu
- Kolejne linie: wartości współrzędnych  $X \ Y$  każdego punktu

#### 2. Krawędzie

- rozszerzenie: .kr
- Pierwsza linia:  $n$  – liczba krawędzi
- Kolejne linie:  $X1, Y1, X2, Y2$

#### 3. Macierz przyległości

Macierz kwadratowa przeskalowanych odległości między wierzchołkami

- rozszerzenie: .txt
- Pierwsza linia:  $h \times w$  – liczba wierzchołków
- Kolejne linie: wartości kolejnych komórek, 0 oznacza nieskończoność

#### 4. Lista wysokości

- rozszerzenie: .wys
- Pierwsza linia: min max - minimalna i maksymalna wysokość
- Kolejne linie: wysokości kolejnych wierzchołków

#### 5. Macierz wysokości

- rozszerzenie: .mwys
- Pierwsza linia:  $h \times w$  – liczba wierzchołków
- Kolejne linie: jak w macierzy przyległości. Zawiera różnicę między dwoma łączonymi wierzchołkami

#### 6. Lista betonowości

- rozszerzenie: .bet
- Pierwsza linia:  $h \times w$  – liczba krawędzi
- Kolejne linie: wartości, 0 oznacza brak betonu, 1 to beton (bardzo nie atrakcyjne)

#### 7. Macierz wysokości

- rozszerzenie: .mbet
- Pierwsza linia:  $h \times w$  – liczba wierzchołków
- Kolejne linie: jak w macierzy przyległości. Symetryczna, nieujemna

## 4.2 Struktury danych

### 4.2.1 Klasa graph

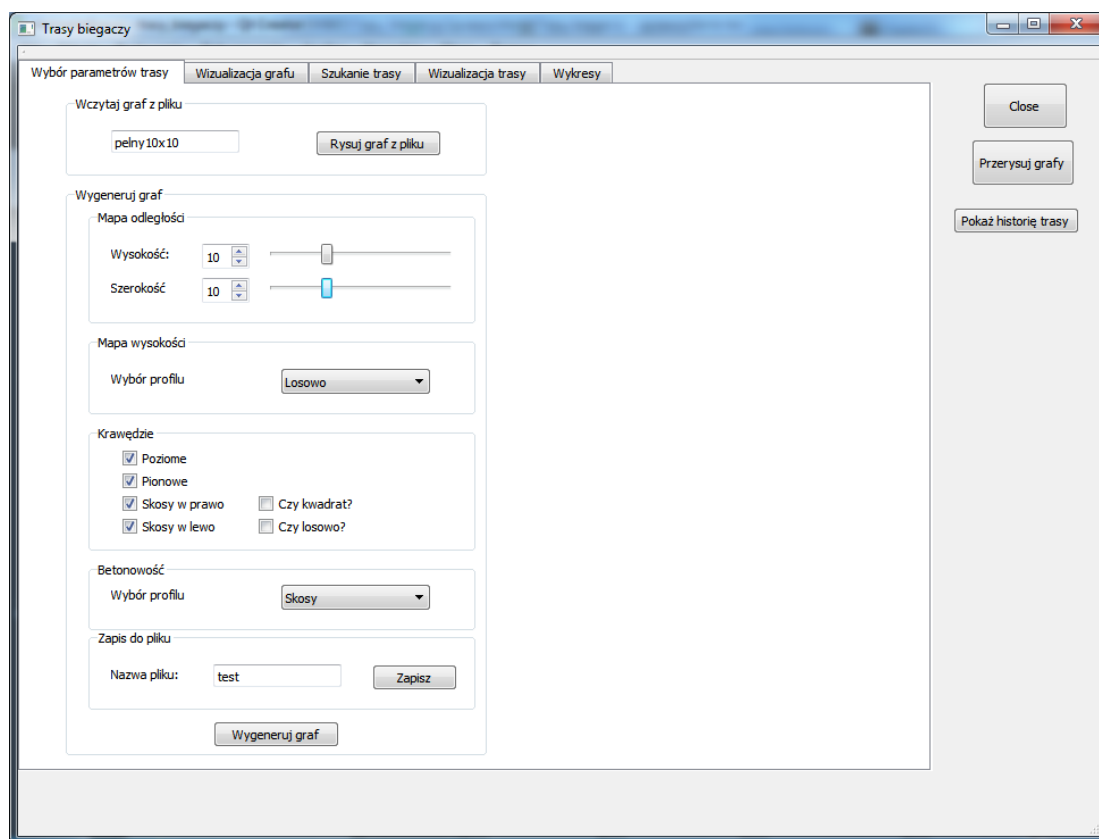
Klasa graph opisuje w pełni wszystkie połączenia i wierzchołki w badanym grafie.

## 4.3 Interface użytkownika

Podczas pracy programu użytkownik ma pełną kontrolę nad przebiegiem algorytmu. W pierwszym oknie możliwe jest wczytanie znanej mapy z pliku lub wygenerowanie nowej mapy zgodnie z przyjętymi parametrami.

Następnie po wygenerowaniu lub wczytaniu mapy możemy dokonać jej podglądu. Drogi pokryte betonem zaznaczono linią przerywaną, a ścieżki bardzo atrakcyjne dla biegacza linią ciągłą. Kolory wierzchołków oznaczają względną wysokość danego wierzchołka. Wierzchołki zielone są niżej, a wierzchołki czerwone są wyżej na mapie.

Kolejną zakładką jest wybór docelowych opcji trasy. Możemy tu wybrać jaka ma być długość trasy, ile na trasie ma być krawędzi betonowych oraz jak bardzo pochyłą chcemy naszą trasę. Następnie po kliknięciu na przycisk „Wygeneruj najlepszą trasę!” program generuje trasę, którą możemy zobaczyć na zakładce 4. W zakładce „Wizualizacja trasy” możemy zobaczyć na czerwono zaznaczoną trasę optymalną obliczoną przez algorytm. W odcieniach szarości zaznaczone są krawędzie, po których algorytm „chodził”. Im ciemniejsza krawędź tym częściej algorytm znajdował się na niej.



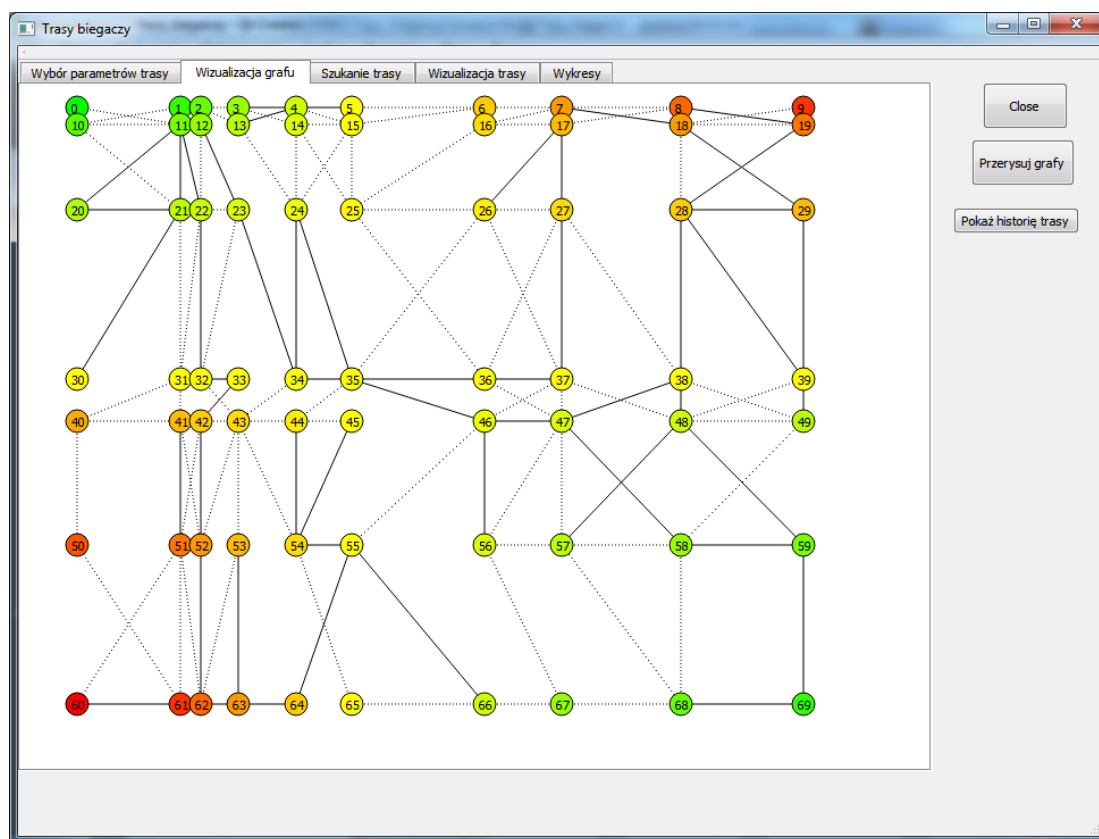
Rysunek 1: Zakładka pozwalająca na edycję mapy.

## 5 Testowanie algorytmu

### 5.1 Test 1

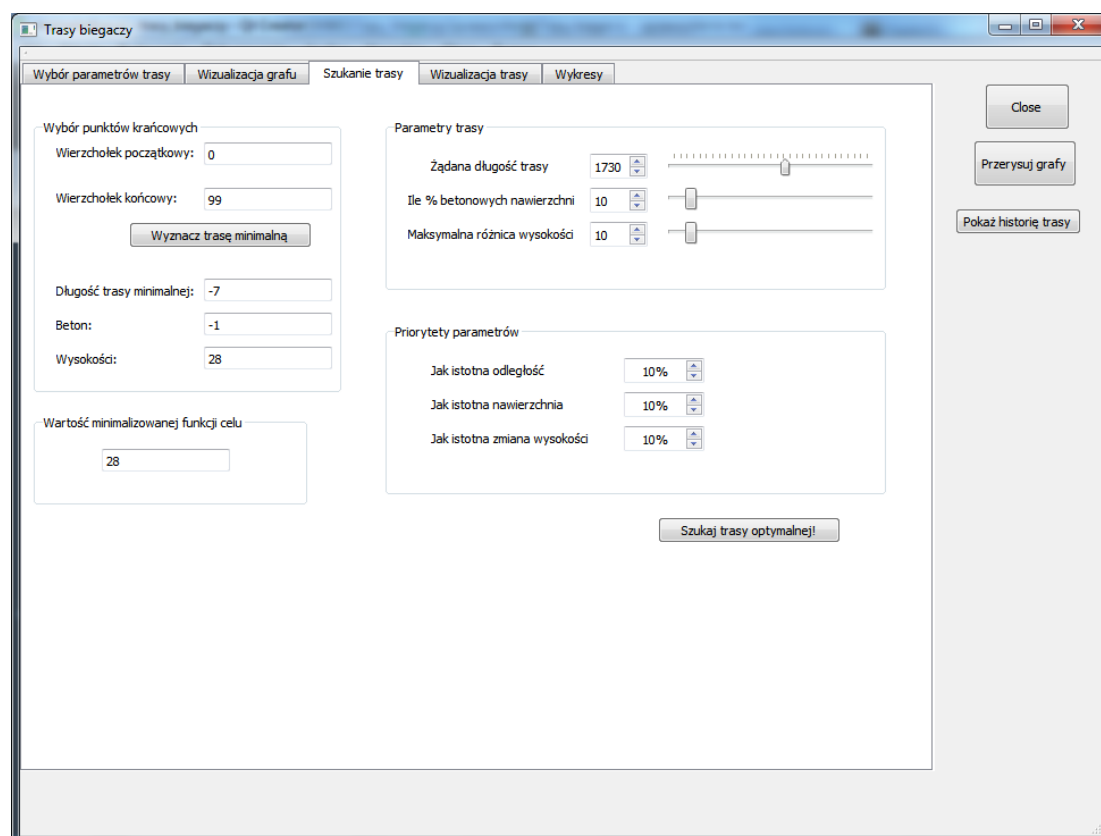
## 6 Analiza efektywności

## 7 Podsumowanie



Rysunek 2: Zakładka pozwalająca podgląd mapy.

Rysunek 3: Zakładka pozwalająca na wybór parametrów do algorytmu.



Rysunek 4: Zakładka „Wizualizacja trasy” wraz z wyświetlonym przykładowym rozwiązaniem i historią przebiegu.