**Programowanie Efektywnych Algorytmów  
Zadanie projektowe 3**

„Implementacja i analiza efektywności algorytmu genetycznego dla problemu Komiwojażera”

Kacper Chrostowski 259120

**Problem Komiwojażera**

Problem optymalizacyjny polegający na znalezieniu w n-wierzchołkowym pełnym grafie takiego cyklu Hamiltona(doliczając do tego powrót do wierzchołka startowego), w którym suma długości ścieżek między wierzchołkami jest jak najmniejsza. Algorytm genetyczny jest heurystyką symulującą faktyczne sposoby ewolucji w środowiskach naturalnych. Kolejno przeprowadzana jest krzyżowanie, mutacja i selekcja nowej populacji.

W omawianym rozwiązaniu zastosowano krzyżowanie OrderCrossover (OX) oraz dwa rodzaje mutacji: Scramble oraz Invert.

**Przebieg algorytmu:**

1)Wygeneruj startową populację składającą się z losowych permutacji rozwiązania tsp

2)Wylosuj dwa chromosomy należące do populacji i porównaj ich długość, zapamiętaj ten w którym trasa jest krótsza jako pierwszego rodzica

3)Powtórz krok 2 i zapamiętaj wynik jako drugiego rodzica

4)Korzystając z krzyżowania wygeneruj parę potomstwa dla wybranych rodziców i zapisz je w wektorze

5)Wróć do punktu 2 i powtarzaj tak długo aż ilość wygenerowanego potomstwa będzie wynosiła rozmiar\_startowej\_populacji \* współczynnik\_krzyżowania.

6)Dla każdego potomka sprawdź prawdopodobieństwo wystąpienia mutacji (określane przez współczynnik mutacji). Jeżeli prawdopodobieństwo na to wskaże, wykonaj odpowiednią mutację na potomku.

7)Połącz ze sobą struktury przechowujących rodziców i dzieci, posortuj populację rosnąco, a następnie usuń nadmiar chromosomów w taki sposób aby struktura miała rozmiar wskazany przez rozmiar startowej populacji

8)Wróć do punktu 2 i powtarzaj dopóki nie minie wskazany czas

9)Najlepsze znalezione rozwiązanie znajduje się w populacji na pierwszym miejscu.

**Opis klas:**

*Menu*: Jest odpowiedzialna za nawigację po programie. Przyjmuje instrukcje od użytkownika, a następnie uruchamia odpowiednie metody.

*Timer*: Klasa odpowiedzialna za pomiar czasu. Zawiera metody uruchamiające pomiar czasu oraz pobierającą aktualny czas.

*Graph*: Klasa, po której dziedziczy klasa Genetic. Służy ona do obsługi danych, na których operuje algorytm (wczytanie i wyświetlanie danych) oraz zawiera metodę obliczającą długość podanej trasy.

*Genetic:* Klasa, która przeprowadza algorytm genetyczny. Zawiera główną metodę , w której wykonuje się główne zadanie oraz metody będące niezbędne do przeprowadzenia algorytmu czyli operator krzyżowania OrderCrossover oraz dwa rodzaje mutacji Scramble oraz Inverse.

**Plan eksperymentu**

Dla każdego algorytmu przeprowadzono 10 testów. Każdy test trwał 120 sekund oraz co 10 sekund pobierano najlepsze znalezione rozwiązanie. W ten sposób będzie można przeanalizować jak z upływem czasu poprawia się rozwiązanie.

Dla każdego zestawu danych testy powtórzono dla dwóch rodzajów mutacji, oraz dla populacji początkowych liczących 1000, 10000 i 100000 osobników.

Testy przeprowadzone zostały na grafach zapisanych w plikach *br17*, *ft53* oraz *ftv170*. Najlepsze otrzymane wyniki będą porównywane z najlepszymi rozwiązaniami tych problemów rozwiązanych przy pomocy algorytmu TabuSearch z poprzedniego zadania.

**Wyniki eksperymentu**

**Wnioski**

Tabu Search

Algorytm działa poprawnie. Jego wadą natomiast jest bardzo szybkie i łatwe wpadanie w lokalne minimum przez brak dywersyfikacji i alternatywnych ruchów. Przez wspomniane wady w bardzo wczesnym etapie znajdujemy minimum lokalne, w którym trwamy przez większość czasu badania. Dla małych problemów jesteśmy w stanie znaleźć rozwiązania o małym błędzie oraz wraz z czasem trwania badania widać poprawę natomiast jest ona dużo mniejsza w porównaniu z drugim algorytmem. Zmienianie wartości kary nakładnej w liście tabu nie przynosi odnotowalnych zmian. Wraz z przebiegiem algorytmu widoczny jest pojedyncza poprawa. Widać dzięki temu, że algorytm po trafieniu w lokalne minimum nie jest w stanie z niego wyjść.

**Źródła:**

<http://staff.iiar.pwr.wroc.pl/antoni.sterna/>  
<https://www.ii.uni.wroc.pl/~prz/2011lato/ah/opracowania/met_podz_ogr.opr.pdf>