**Koncepcja aplikacji i algorytmu genetycznego dla optymalizacji świateł drogowych**

**1. Funkcjonalność aplikacji – stanowisko badawcze**

**Konfiguracja algorytmu** Użytkownik może dostosować parametry algorytmu za pomocą interfejsu graficznego (GUI):

* Rozmiar populacji
* Liczba generacji
* Procent elityzmu
* Prawdopodobieństwo mutacji
* Typ crossover (np. BLX, liniowy)
* Współczynnik alpha dla operacji crossover
* Liczba cykli w symulacji

Dane dotyczące parametrów algorytmu i konfiguracji sygnalizacji świetlnej są generowane dynamicznie przez algorytm symulacji, który definiuje kolejkę pojazdów oraz parametry skrzyżowań.

Po zakończeniu optymalizacji aplikacja:

* Wyświetla najlepszą konfigurację świateł dla każdego skrzyżowania.
* Prezentuje wykres najlepszych wartości fitness w kolejnych generacjach.
* Umożliwia uruchomienie symulacji graficznej dla optymalnego rozwiązania.

Interfejs został zaprojektowany w technologii Tkinter z wykorzystaniem pasków postępu (progress bar) oraz panelu do prezentacji wykresów (matplotlib). Wyniki są dostępne w osobnym oknie dialogowym.

Wykresy są generowane za pomocą biblioteki Matplotlib i przedstawiają postęp optymalizacji (wartość fitness w zależności od generacji).

**2. Prezentacja własności rozwiązania - problemu**

Problem dotyczy optymalizacji czasów świateł na skrzyżowaniach w celu zminimalizowania całkowitego czasu oczekiwania pojazdów. Każde skrzyżowanie posiada:

* Kolejki pojazdów w każdym kierunku.
* Czasy świecenia się zielonego światła dla każdego kierunku.
* Cykl sygnalizacji.

Rozwiązanie problemu wymaga znalezienia optymalnego parametru (czasu), które minimalizuje łączny czas oczekiwania pojazdów.

**3. Prezentacja własności algorytmu – przebieg, mechanizmy, elementy**

**Przebieg algorytmu:**

1. Inicjalizacja populacji genów reprezentujących konfiguracje sygnalizacji świetlnej.
2. Iteracyjne generowanie nowych populacji poprzez:
   * Selekcję rodziców na podstawie ich wartości fitness.
   * Operacje crossover (BLX, liniowy).
   * Mutację (zmiana czasów świateł lub kolejności sygnalizacji).
3. Zapis najlepszych rozwiązań (elityzm).

**Mechanizmy:**

* Funkcja fitness ocenia efektywność konfiguracji na podstawie wyników symulacji.
* Algorytm BLX i liniowy służą do krzyżowania genomów rodziców w celu generowania nowych rozwiązań.
* Mutacje losowo zmieniają parametry, co pozwala uniknąć lokalnych minimów.

**Elementy:**

* Genotyp: czasy świateł oraz kolejność sygnalizacji dla 4 skrzyżowań.
* Populacja: zestaw genomów testowanych w każdej generacji.

**4. Eksperymenty obliczeniowe**

**Instancje testowe**

* Sposób generacji: Parametry początkowe (liczba pojazdów, ich miejsca początkowe i docelowe) są możliwe do zmieniania w kodzie.
* Liczba instancji: 10 testów dla każdej konfiguracji algorytmu.
* Rozmiar: 4 skrzyżowania, każde z czterema kierunkami.

**Metodyka badań**

* Środowisko testowe: Komputer z procesorem Intel i5 i 8GB RAM.
* Statystyka badań:
  + Liczba powtórzeń: 10 iteracji dla różnych kombinacji parametrów.

**Problemy badawcze**

* Wpływ parametrów algorytmu (rozmiar populacji, liczba generacji, prawdopodobieństwo mutacji) na wyniki.
* Wpływ typu crossover (BLX vs liniowy).

**Zmienne wynikowe**

* Łączny czas oczekiwania pojazdów w symulacji.

**Wykresy**

* Wpływ parametrów na łączny czas oczekiwania.