

IMO – LAB 4 Rozszerzenia lokalnego przeszukiwania

Jan Bróździak 141142

Opis zadania

Celem niniejszego zadania jest implementacja i analiza trzech metod rozszerzających lokalne przeszukiwanie, które są często stosowane w problemach optymalizacji kombinatorycznej. W szczególności, zajmujemy się metodami Multiple Start Local Search (MSLS) oraz dwiema wersjami Iterated Local Search (ILS), gdzie różnicę stanowi sposób perturbacji rozwiązania. W ramach tego eksperymentu będziemy przeprowadzać testy na instancjach problemów kroa200 i krob200, a wyniki zostaną porównane pod względem jakości rozwiązań oraz efektywności czasowej.

Opis zaimplementowanych algorytmów w pseudokodzie

MSLS

```
Function MSLS(cities, local_search_function, iterations)
    Set best_solution to None
    Set best_score to infinity

    For iteration from 1 to iterations
        Generate random initial cycles (cycle1, cycle2) from cities

        Perform local search starting from (cycle1, cycle2)
        (cycle1, cycle2, score) = local_search_function(cycle1,
cycle2, cities, generate_combined_moves_list_2)

        If score < best_score Then
            Set best_score to score
            Set best_solution to (cycle1, cycle2)
        End If
    End For

    Return best_solution, best_score
End Function
```

ILS1

```
Function ILS1(cities, local_search_function, time_limit,
perturbation_strength)
    Set start_time to current time

    Generate random initial cycles (cycle1, cycle2) from cities
    Perform initial local search
    (cycle1, cycle2, best_score) = local_search_function(cycle1,
cycle2, cities, generate_combined_moves_list_2)

    While current time - start_time < time_limit
        Create a copy of the current best solution (new_cycle1,
new_cycle2)

        For perturbation_strength times
            Select random indices in new_cycle1 and new_cycle2
            Swap vertices between cycles (new_cycle1, new_cycle2)
        End For

        Perform local search on the perturbed solution
        (new_cycle1, new_cycle2, new_score) =
local_search_function(new_cycle1, new_cycle2, cities,
generate_combined_moves_list_2)

        If new_score < best_score Then
            Set cycle1, cycle2 to new_cycle1, new_cycle2
            Set best_score to new_score
        End If
    End While

    Return cycle1, cycle2, best_score
End Function
```

ILS2

```
Function ILS2(cities, local_search_function, time_limit,
destroy_fraction)
    Set start_time to current time

    Generate random initial cycles (cycle1, cycle2) from cities
    Perform initial local search
    (cycle1, cycle2, best_score) = local_search_function(cycle1,
cycle2, cities, generate_combined_moves_list_2)

    While current time - start_time < time_limit
        Create a copy of the current best solution (new_cycle1,
new_cycle2)

        Calculate destroy_count as a fraction of the total number of
vertices
        Select random vertices to remove from the cycles

        For each vertex in vertices_to_remove
            Remove vertex from the corresponding cycle
        End For

        Use a repair heuristic to reinsert the removed vertices into
new_cycle1 and new_cycle2

        Perform local search on the repaired solution
        (new_cycle1, new_cycle2, new_score) =
local_search_function(new_cycle1, new_cycle2, cities,
generate_combined_moves_list_2)

        If new_score < best_score Then
            Set cycle1, cycle2 to new_cycle1, new_cycle2
            Set best_score to new_score
        End If
    End While

    Return cycle1, cycle2, best_score
End Function
```

Przebieg i parametry eksperymentu

Lokalne przeszukiwanie: Steep + Krawędzie

Ilość iteracji dla MSLS: 100

Czas uruchomień dla ILSx: 473 sek. **Uwaga:** Ponieważ obliczenia trwały długo, równolegle ustawiono czas na podstawie pierwszego przebiegu MSLS. Faktyczny średni czas MSLS to 522s. Oznacza to że wyniki ILSx mogły być potencjalnie lepsze.

Ilość uruchomień ILSx: 10

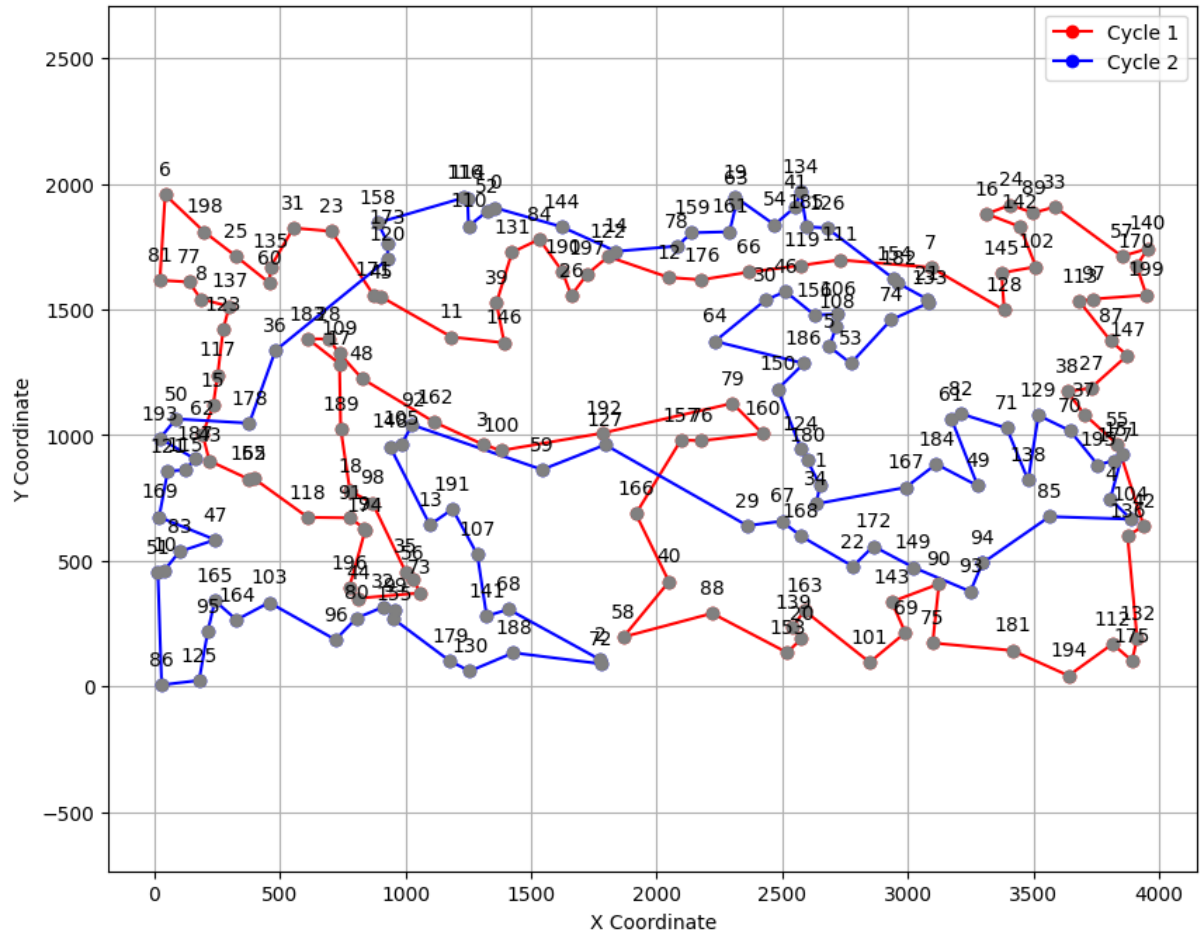
Perturbacja w ILS1: 5 wierzchołków

Perturbacja w ILS2: 30% wierzchołków

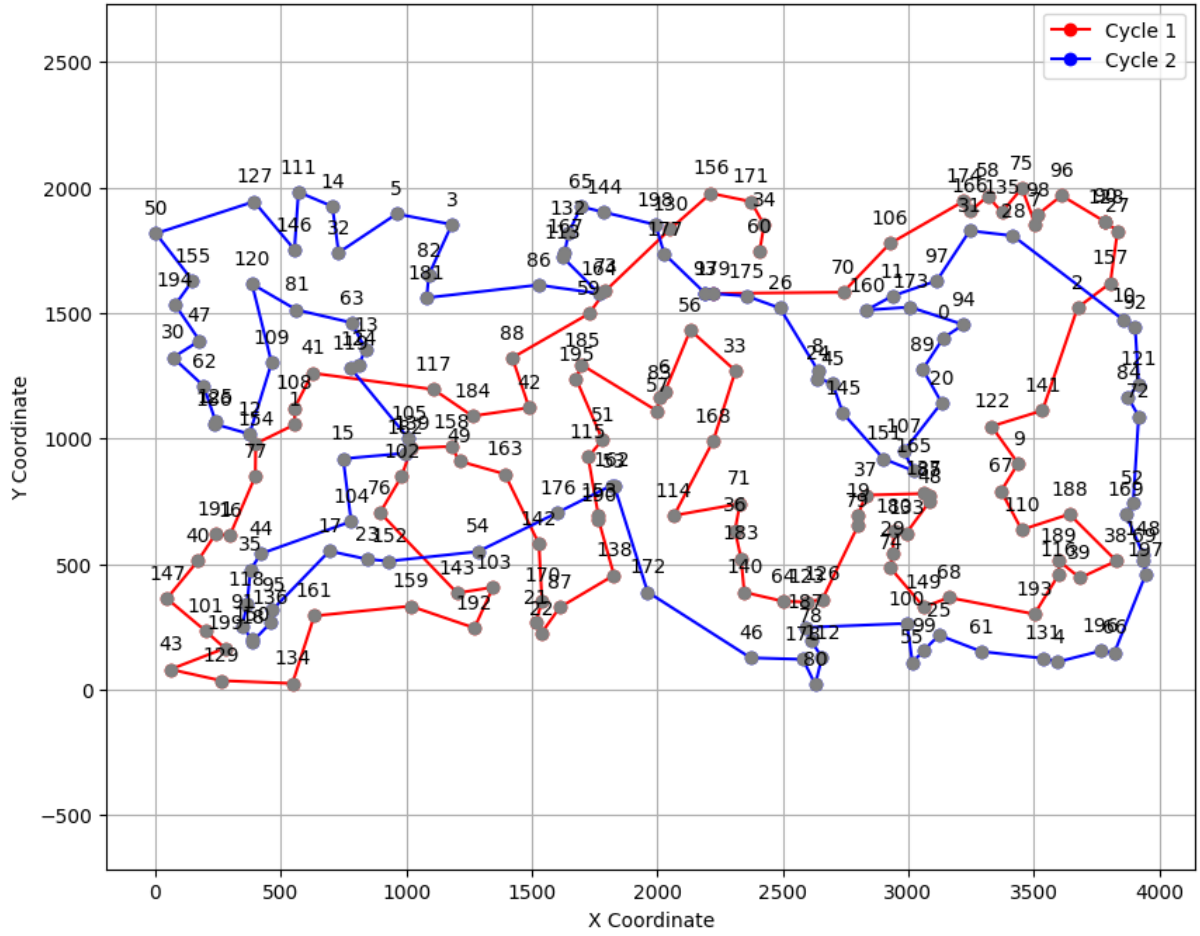
Wyniki eksperymentu:

Metoda	Instancja	Min	Max	Avg
MSLS	KRO200A	35341	36510	35945
ILS1	KRO200A	33345	36223	34409
ILS2	KRO200A	31913	34204	32782
MSLS	KRO200B	35638	36853	36161
ILS1	KRO200B	34447	36264	35438
ILS2	KRO200B	32920	34841	33752

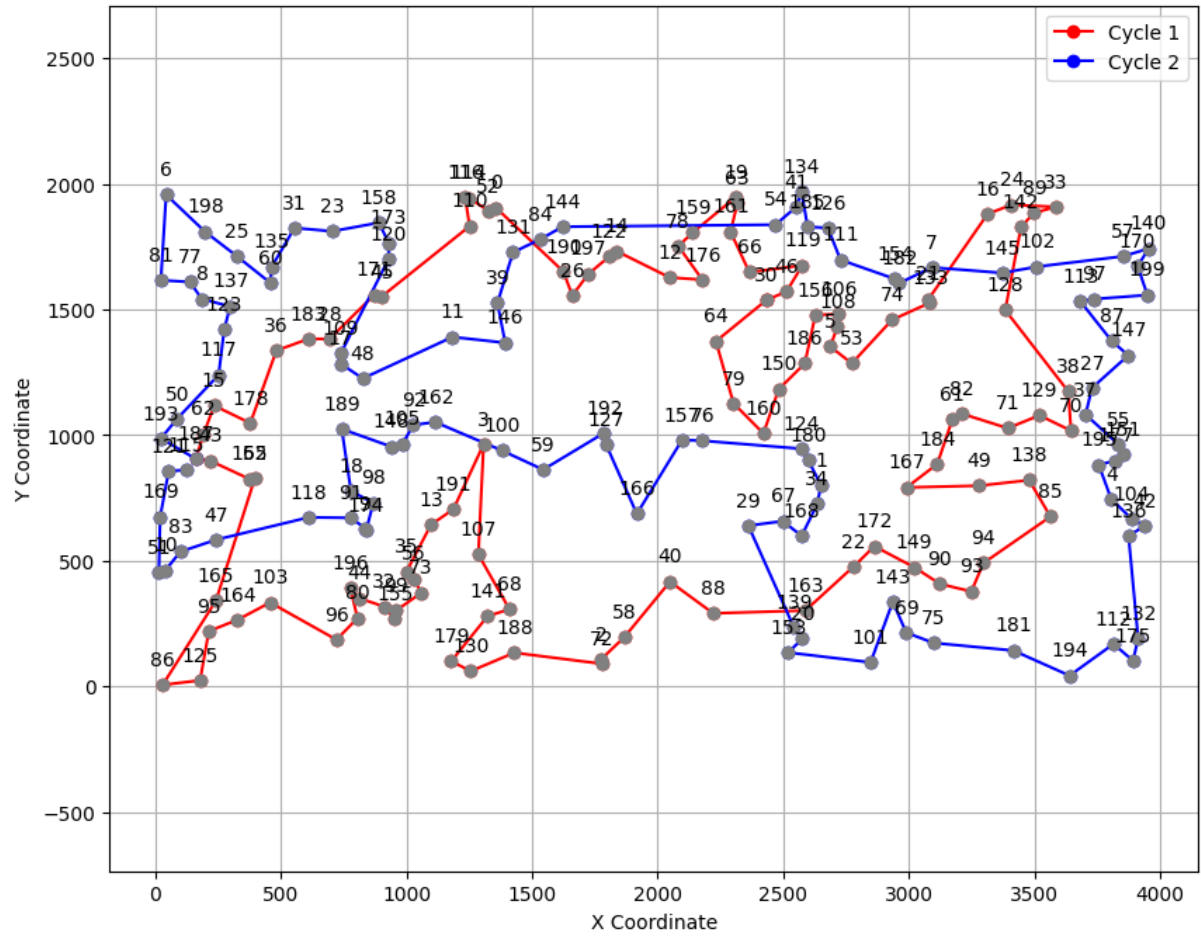
MSLS Solution for kroA200



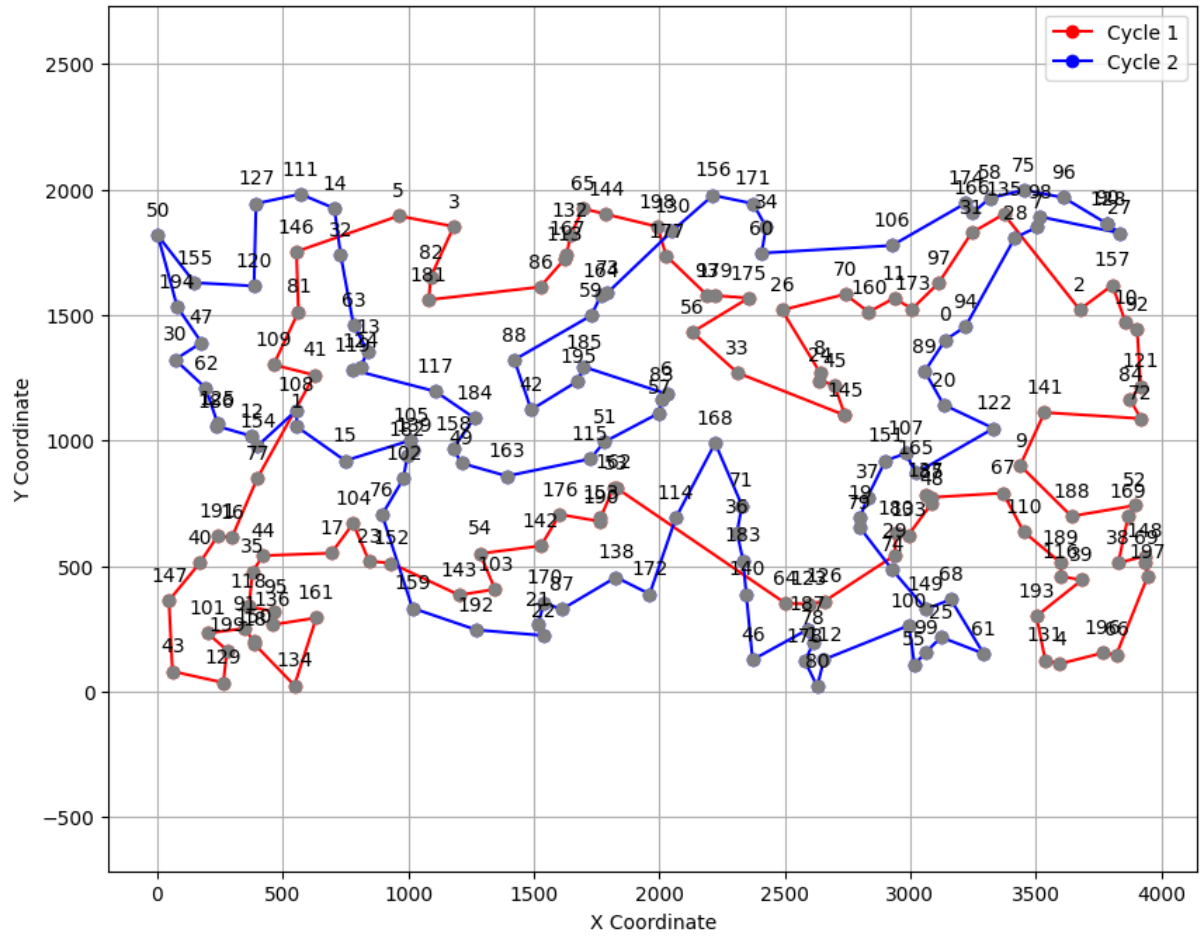
MLS Solution for kroB200



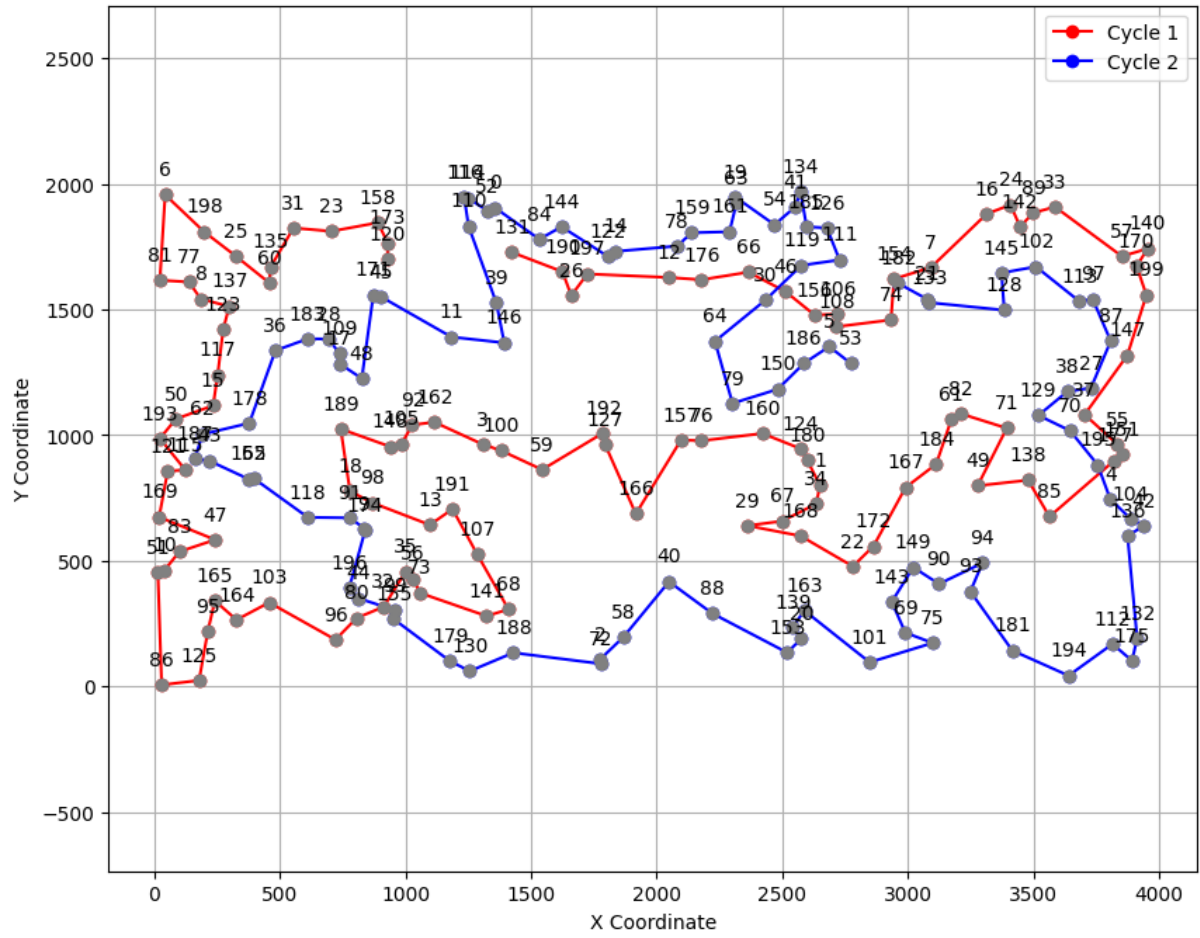
ILS1 Solution for kroA200

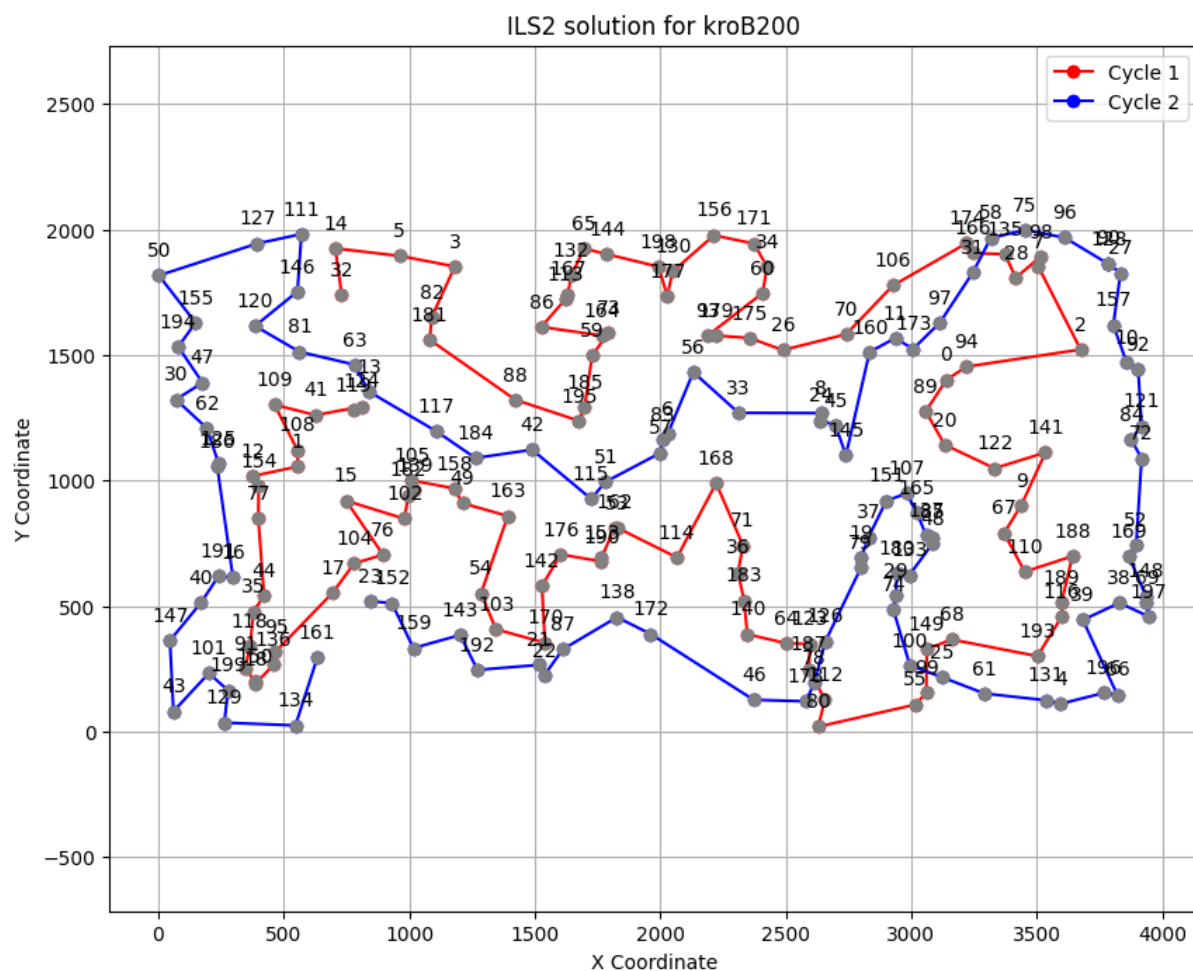


ILS1 solution for kroB200



ILS2 solution for kroA200





Kod programu jest dostępny pod linkiem: <https://github.com/Evarios/IMO>