

Hybrydowy algorytm ewolucyjny

Autorzy: Dominik Maćkowiak 151915, Szymon Szymankiewicz 151821

Opis zadania

Celem zadania jest rozwiązanie zmodyfikowanego problemu komiwojażera poprzez znalezienie dwóch rozłącznych cykli, obejmujących wszystkie wierzchołki. Każdy cykl powinien zawierać połowę wierzchołków (lub o jeden więcej w przypadku nieparzystej liczby).

Dane są dwie instancje ("kroA200" oraz "kroB200") z biblioteki TSPLib, które zawierają informacje o położeniu konkretnych wierzchołków. Celem zadania jest utworzenie z każdej instancji dwóch rozłącznych cykli, tak aby ich łączna długość była jak najmniejsza.

Zaimplementowano i porównano hybrydowy algorytm ewolucyjny (HAE) z metodami MSLS, ILS i LNS. HAE oparty jest na populacji elitarniej i strategii steady-state. Potomstwo tworzone jest przez usunięcie z jednego rodzica krawędzi niewystępujących u drugiego, a następnie naprawiane heurystycznie. Testowano wersję z i bez lokalnego przeszukiwania po rekombinacji. Każdą metodę uruchamiano 10 razy, przy tych samych parametrach co wcześniej.

Algorytm

Funkcje pomocnicze:

Recombine

Na wejściu otrzymujemy rozwiązanie1 i rozwiązanie2

Tworzymy listę krawędzi z rozwiązania2 (vertexA->vertexB)

Z rozwiązania1 usuwamy z cyklu1 i cyklu2 krawędzie, które znajdują się na liście krawędzi rozwiązania2

Zapisujemy, które krawędzie usunęliśmy

Uruchamiamy algorytm **Repair** żeby uzupełnić brakujące krawędzie

Repair

Przywracamy usunięte wierzchołki do cykli w kolejności malejącego żalu

1. Oblicz żal dla wstawienia wierzchołka do cyklu 1 i 2
2. Wybierz cykl z większym żalem (lub mniejszym wzrostem długości w przypadku remisu)
3. Wstaw wierzchołek do tego cyklu

Zamknij cykle, jeśli nie są zamknięte

HAE

Wygeneruj początkową populację (algorytm Steepest, z 20 losowych cykliów)

Powtarzaj aż do osiągnięcia limitu czasu (średni czas MSLS dla aktualnej instancji):

- Wybierz dwóch losowych rodziców z populacji
- Zastosuj **rekombinację** na rodzicach, żeby otrzymać potomka
- (jeśli wariant z LS) Zastosuj algorytm Steepest na potomku
- Jeżeli potomek jest lepszy niż najgorszy rodzic w populacji, to usuń najgorszego rodzica i dodaj potomka do populacji

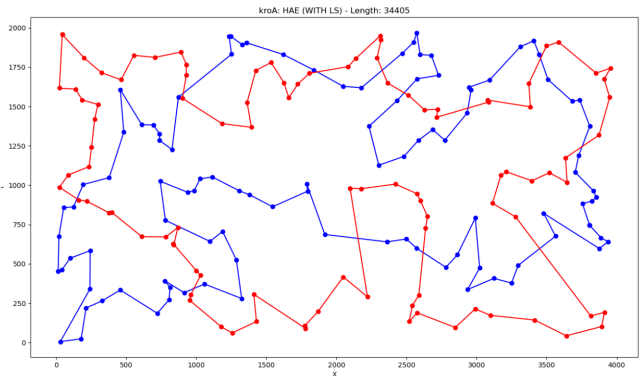
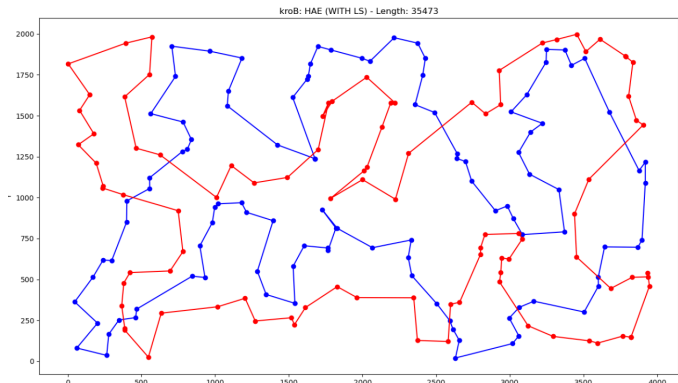
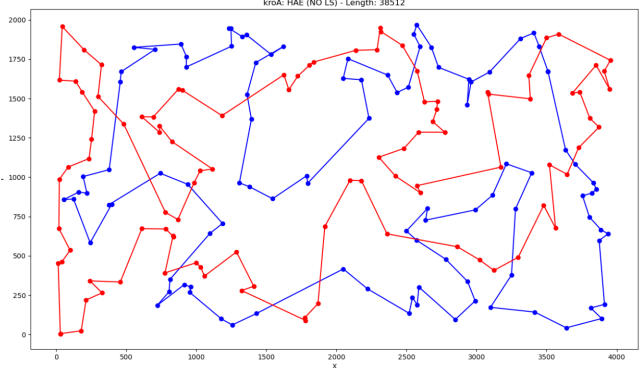
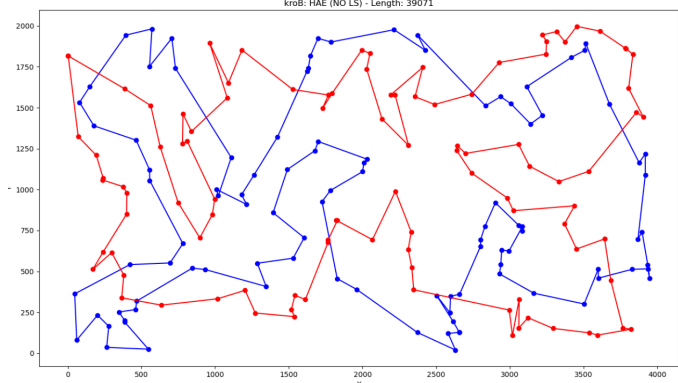
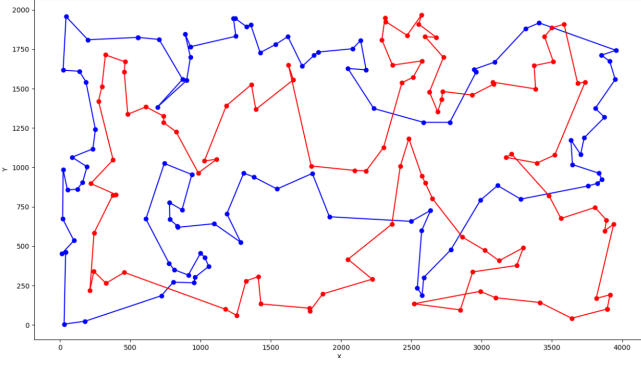
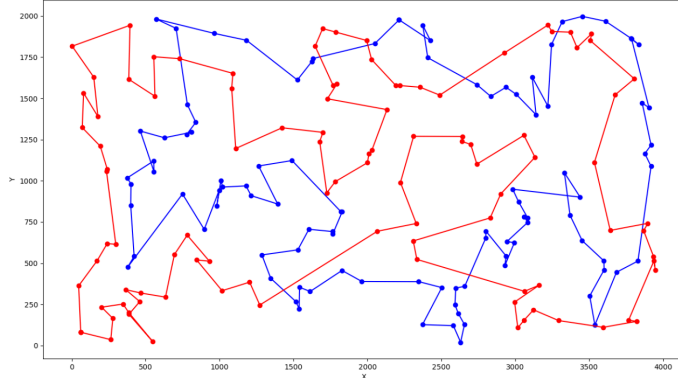
Zwróć najlepszego rodzica z populacji

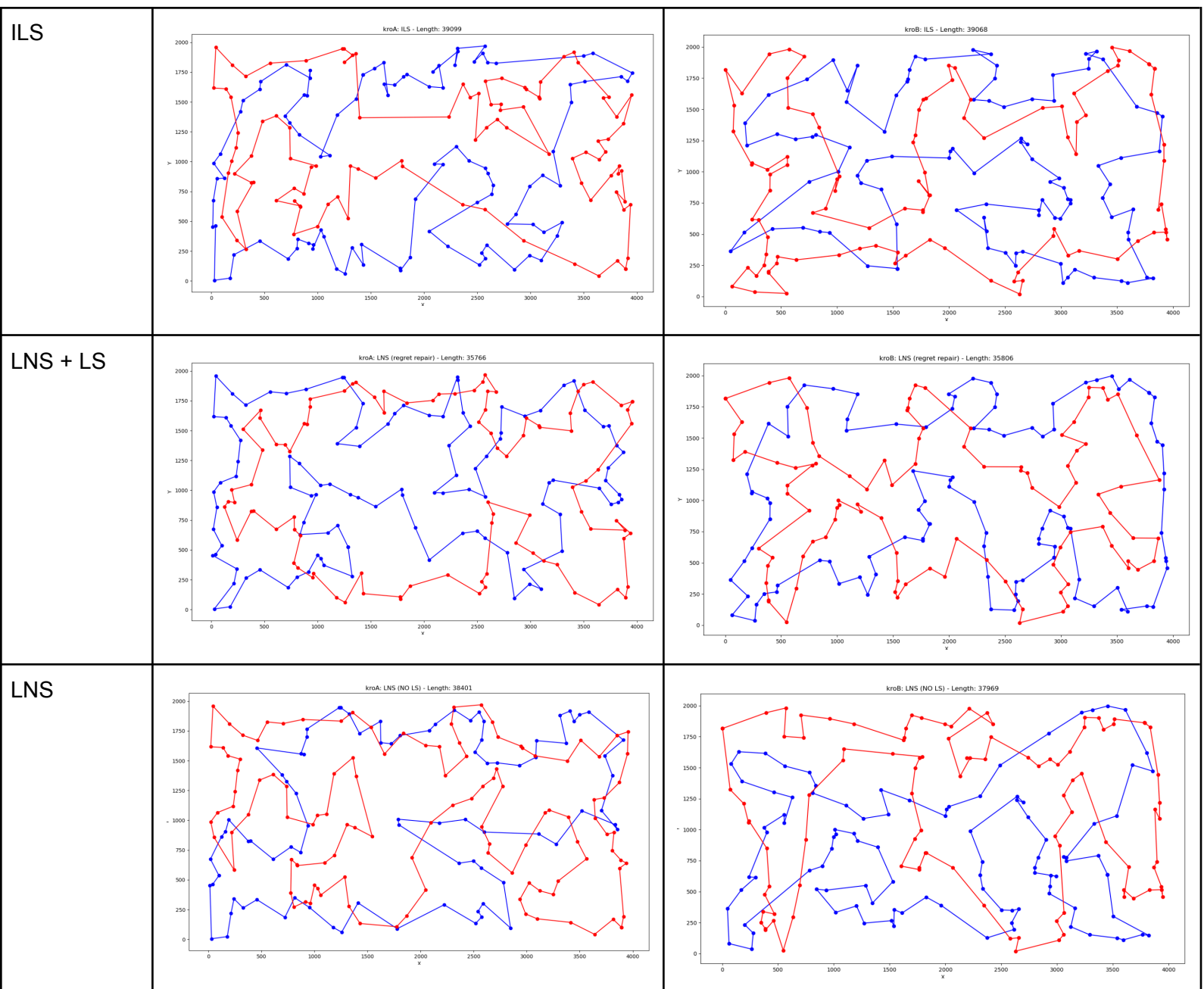
Wyniki eksperymentu obliczeniowego

Przeprowadziliśmy eksperyment, który polegał na zmierzeniu długości dwóch utworzonych cykli. Dla każdej instancji przeprowadziliśmy 10 testów, gdzie każdy działał przez 1300 sekund (średni czas MSLS dla 200 uruchomień). Obliczyliśmy średnią długość z tych testów, oraz wybraliśmy odcinek najdłuższy i najkrótszy.

	kroA200				kroB200			
	średnia	min	max	iteracje (śred.)	średnia	min	max	iteracje (śred.)
HAE + LS	32496.3	31405	33422	1650	32669.6	32473	33022	2486
HAE	36433.6	35512	37178	84436	36561.6	36071	36816	84590
MSLS	35955	34460	37262	200	36040.8	34948	37231	200
ILS	34401.7	33929	36542	3652	35629.3	34523	37098	3960
LNS + LS	33342.6	32766	34240	1012	33711.6	32806	34706	1166
LNS	36005.6	34435	36425	140734	35009.3	34199	35189	140184

Wizualizacja dla najlepszych rozwiązań

	kroA200	kroB200
HAE + LS	 <p>kroA: HAE (WITH LS) - Length: 34405</p>	 <p>kroB: HAE (WITH LS) - Length: 35473</p>
HAE	 <p>kroA: HAE (NO LS) - Length: 38512</p>	 <p>kroB: HAE (NO LS) - Length: 39071</p>
MSLS	 <p>kroA: MSLS (200 starts) - Length: 38274</p>	 <p>kroB: MSLS (200 starts) - Length: 38986</p>



Wnioski

Kod programu: <https://github.com/szymon240/imo-lab5>