

Inteligentne metody optymalizacji

Laboratorium nr 1

24 stycznia 2024

Prowadzący: prof. dr hab. inż. Andrzej Jaskiewicz

Autorzy: **Krzysztof Martin** 141275
Bartosz Paliński 145224

Zajęcia poniedziałkowe, 11:45.

Oświadczam/y, że niniejsze sprawozdanie zostało przygotowane wyłącznie przez powyższych autora/ów, a wszystkie elementy pochodzące z innych źródeł zostały odpowiednio zaznaczone i są cytowane w bibliografii.

1 Opis zadania

Celem zajęć laboratoryjnych było zaimplementowanie oraz przetestowanie heurystyk konstrukcyjnych dla zmodyfikowanego problemu komiwojażera. Modyfikacja polegała na stworzeniu dwóch rozłącznych zamkniętych cykli (ścieżek), każdy zawierający 50 % wierzchołków (jeżeli liczba wierzchołków w instancji nie jest parzysta, to pierwsza ścieżka zawiera jeden wierzchołek więcej), minimalizując łączną długość obu ścieżek. Efektywność algorytmów była oceniana na podstawie instancji kroaA100 i kroB100 z biblioteki TSPLib. W ramach zadania zaimplementowano 3 algorytmy:

- algorytm zachłanny inspirowany metodą najbliższego sąsiada,
- algorytm zachłanny inspirowany metodą rozbudowy cyklu,
- algorytm typu regret heuristics (z żalem).

2 Algorytmy

2.1 Heurystyka najbliższego sąsiada

- 1: Wybierz losowy wierzchołek startowy pierwszego cyklu
- 2: Wybierz najdalszy wierzchołek względem startowego pierwszego cyklu, który będzie startowy, dla drugiego cyklu
- 3: **Dopóki** istnieją nieodwiedzone wierzchołki:
 - 4: Dla pierwszego cyklu wybierz parę wierzchołków - jeden z cyklu, drugi nieodwiedzony, między którymi odległość jest najmniejsza
 - 5: Wstaw drugi wierzchołek po pierwszym
 - 6: Oznacz ten wierzchołek jako odwiedzony
 - 7: Dla drugiego cyklu wybierz parę wierzchołków - jeden z cyklu, drugi nieodwiedzony, między którymi odległość jest najmniejsza
 - 8: Wstaw drugi wierzchołek po pierwszym
 - 9: Oznacz ten wierzchołek jako odwiedzony
- 10: Zwróć ścieżki oraz sumę odległości.

2.2 Metoda rozbudowy cyklu

- 1: Wybierz losowy wierzchołek startowy pierwszego cyklu
- 2: Wybierz najbliższy wierzchołek i stwórz z tych dwóch wierzchołków cykl
- 3: Wybierz najdalszy wierzchołek względem startowego pierwszego cyklu, który będzie startowy, dla drugiego cyklu
- 4: Wybierz najbliższy wierzchołek i stwórz z tych dwóch wierzchołków cykl
- 5: **Dopóki** istnieją nieodwiedzone wierzchołki:
 - 6: Wstaw do pierwszego cyklu w najlepsze możliwe miejsce wierzchołek powodujący najmniejszy wzrost długości tego cyklu
 - 7: Oznacz ten wierzchołek jako odwiedzony
 - 8: Wstaw do drugiego cyklu w najlepsze możliwe miejsce wierzchołek powodujący najmniejszy wzrost długości tego cyklu
 - 9: Oznacz ten wierzchołek jako odwiedzony
- 10: Zwróć ścieżki oraz sumę odległości.

2.3 Heurystyka 2-żal

- 1: Wybierz losowy wierzchołek startowy pierwszego cyklu
- 2: Wybierz najbliższy wierzchołek i stwórz z tych dwóch wierzchołków cykl
- 3: Wybierz najdalszy wierzchołek względem startowego pierwszego cyklu, który będzie startowy, dla drugiego cyklu
- 4: Wybierz najbliższy wierzchołek i stwórz z tych dwóch wierzchołków cykl
- 5: **Dopóki** istnieją nieodwiedzone wierzchołki:
- 6: Dla każdego wierzchołka oblicz listę kosztów wstawienia go do cyklu
- 7: Oblicz 2-żal jako różnicę między najmniejszą oraz drugą najmniejszą wartością w liście oraz pomniejsz go o $1/3$ kosztu wstawienia
- 8: Wybierz wierzchołek z najmniejszym 2-żalem
- 9: Oznacz ten wierzchołek jako odwiedzony
- 10: Ponów operacje **6,7,8,9** dla 2 cyklu
- 11: Zwróć ścieżki oraz sumę odległości.

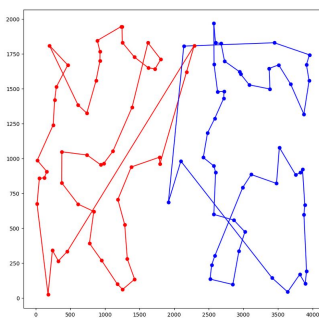
3 Wyniki eksperymentu obliczeniowego

Każdy algorytm został przetestowany 100 razy dla każdej instancji, za każdym razem wybierając inny wierzchołek startowy cyklu. Średnie, najlepsze oraz najgorsze wyniki są przedstawione w poniższej tabeli.

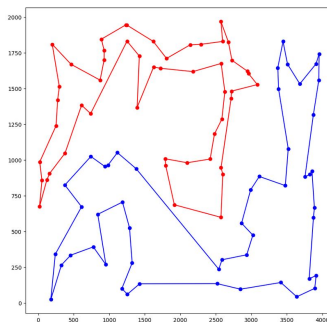
	Instancja kroA100	Instancja kroB100
Nearest neighbor	32506 (27168 - 35345)	32283 (26287 - 40214)
Greedy cycle	28717 (26268 - 30223)	28501 (27689 - 30152)
2-regret cycle	27587 (22499 - 32374)	28242 (24691 - 30467)

4 Wizualizacje najlepszych rozwiązań

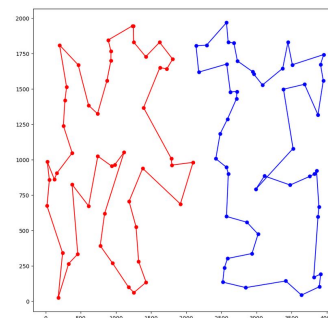
4.1 kroaA100



Rysunek 1: nearest neighbor

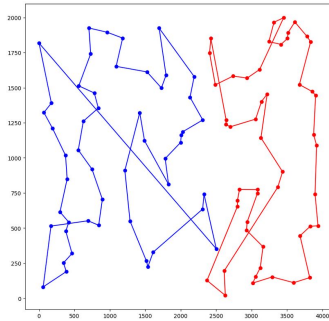


Rysunek 2: greedy cycle

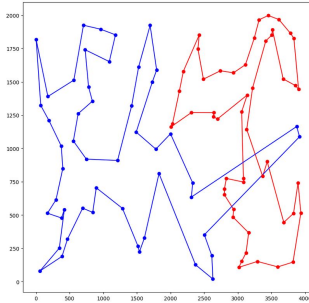


Rysunek 3: 2-regret

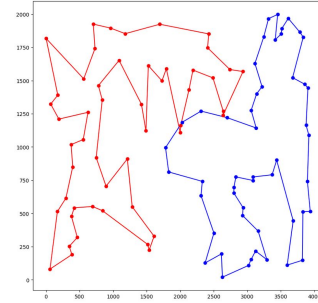
4.2 kroaB100



Rysunek 4: nearest neighbor



Rysunek 5: greedy cycle



Rysunek 6: 2-regret

5 Wnioski

Najlepsze wyniki, zarówno dla średniej, minimum oraz maksimum zmierzono dla heurystyki 2-żal. Najlepszy wynik dla tej heurystyki był aż o 14% lepszy dla algorytmu rozbudowy cyklu, oraz o 17% od algorytmu najbliższego sąsiada. Algorytm najbliższego sąsiada osiąga gorsze wyniki niż algorytm rozbudowy cyklu oraz heurystyki 2-żal. Wynika to z faktu, że algorytm najbliższego sąsiada, wyczerpując wierzchołki w danej lokalizacji przestrzennej, ponosi duży koszt, aby przeskoczyć w inny obszar płaszczyzny. Również łączenie pierwszego z ostatnim wierzchołkiem jest kosztowne dla tego algorytmu.

Dla wszystkich algorytmów zachłanych ważnym elementem jest wybór wierzchołka początkowego. Dla heurystyki najbliższego sąsiada dla zbioru kroB100, sumaryczna długość cykli dla najgorszego wierzchołka początkowego była aż o 52% większa niż dla najlepszego wierzchołka początkowego. Dla heurystyk greedy cycle zaobserwowano znacznie mniejszą różnicę między wartością minimalną oraz maksymalną, wynoszącą od 8% do 15%.

6 Kod programu

Kod źródłowy dostępny on-line: <https://github.com/barosz0/IMO/tree/main/lab1vscod>