

Algorymy metaheurystyczne

Mateusz Chęciński, Mateusz Tofil

16 maja 2022

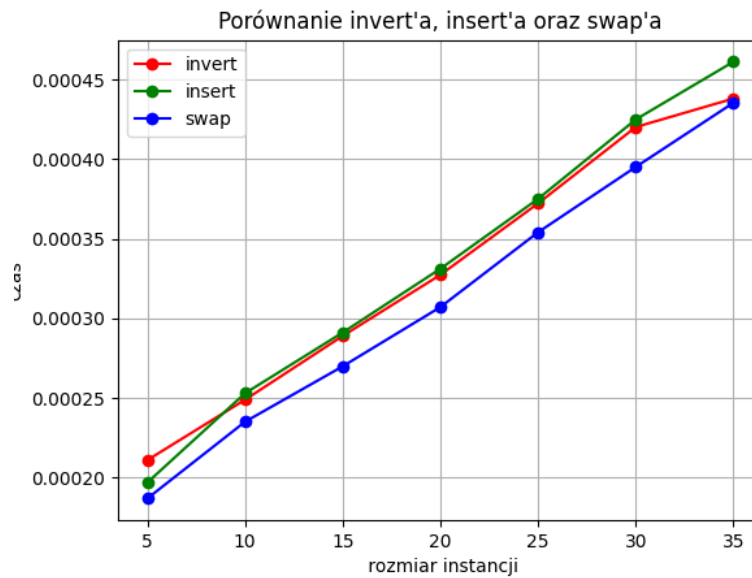
1 Wprowadzenie

Celem tej listy było zapoznanie się z algorytmem przeszukiwania z zabronieniami (z ang. Tabu search).

Język programowania: Python 3.10

2 Porównanie otoczeń: insert, invert, swap

W celu zbadania jakie otoczenie jest najlepsze z powyższych 3, przeprowadziliśmy eksperymenty wywołując metode ze zmienionym parametrem początkowym. Wszystkie eksperymenty były przeprowadzone dla tej samej instancji wraz z tym samym rozwiązaniem początkowym.



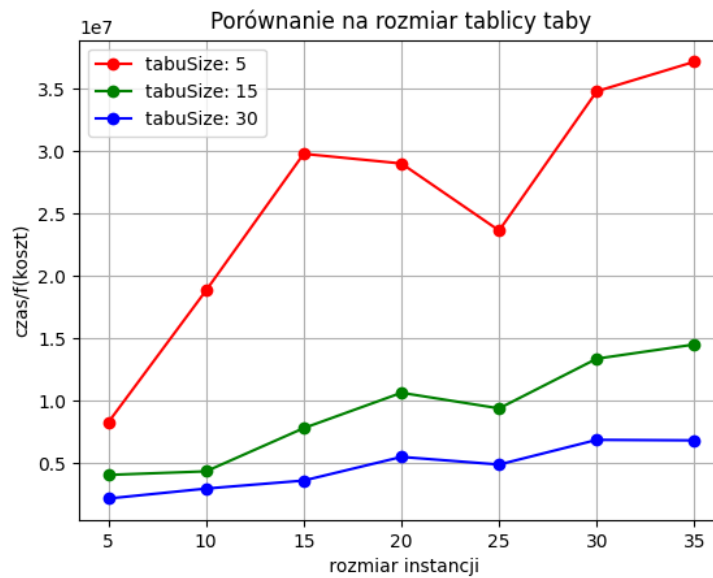
Rysunek 1: Porównanie otoczeń

Jak możemy zauważyć, dla wszystkich przeprowadzonych przez nas instancji, otoczenie *swap* okazało się być najefektywniejszym. Pozostałe otoczenia, też nie są złe. Wszystkie otoczenia działają w czasie $\mathcal{O}(n)$ i różnią się tylko stałą, najmniejszą stałą posiada *swap*.

3 Czy długość listy Tabu ma znaczenie?

Chcemy upewnić się, czy zwiększając długość listy Tabu otrzymamy lepszy wynik, tj cykl o najmniejszym koszcie w chwili kończenia algorytmu. Doskonale

wiemy, że jakbyśmy zwiększyli sam rozmiar listy tabu to algorytm wykonywał by się znacznie dłużej. Dlatego zdecydowaliśmy się, że podzielimy wartość (koszt) cyklu na zakończeniu algorytmu przez czas w jakim otrzymaliśmy wynik.



Rysunek 2: Porównanie otoczeń

Wyszło, tak jak zakładaliśmy, czyli mimo, że algorytm dłużej pracował, (im większa liczba tabu, tym dłużej działa), daje lepsze wyniki. Zatem warto jest poczekać odpowiednio dłużej aby otrzymać lepszy wynik.