

Grzegorz Ważny 242507

SPRAWOZDANIE

Sztuczna Inteligencja i inżynieria wiedzy

Laboratorium 1

Implementacja i badanie Algorytmów Ewolucyjnych

Prowadzący przedmiot: dr inż. Jacek Gruber

1. Zakres ćwiczenia

W całym ćwiczeniu korzystałem z takich narzędzi jak Python oraz Visual Paradigm. Moim zadaniem było stworzenie algorytmu generycznego oraz przetestowanie, które współczynniki dadzą najlepsze efekty. Wyniki, które osiągnąłem na podstawie tego algorytmu generycznego porównałem z wynikami algorytmów losowych oraz zachłannych. W przypadku algorytmu zachłannego wyniki były w każdej próbie takie same, więc postanowiłem pominąć 10-krotne testowanie. Stworzyłem wykresy dla poniższych danych oraz przedstawiłem tabelę z wynikami. Algorytmy, z których korzystałem:

- Selekcja turniejowa (Tournament Select)
- Operator krzyżowania OX (Ordered Crossover)
- Mutacja zamiana (Swap)

Do testowania wykorzystałem następujące zbiory danych:

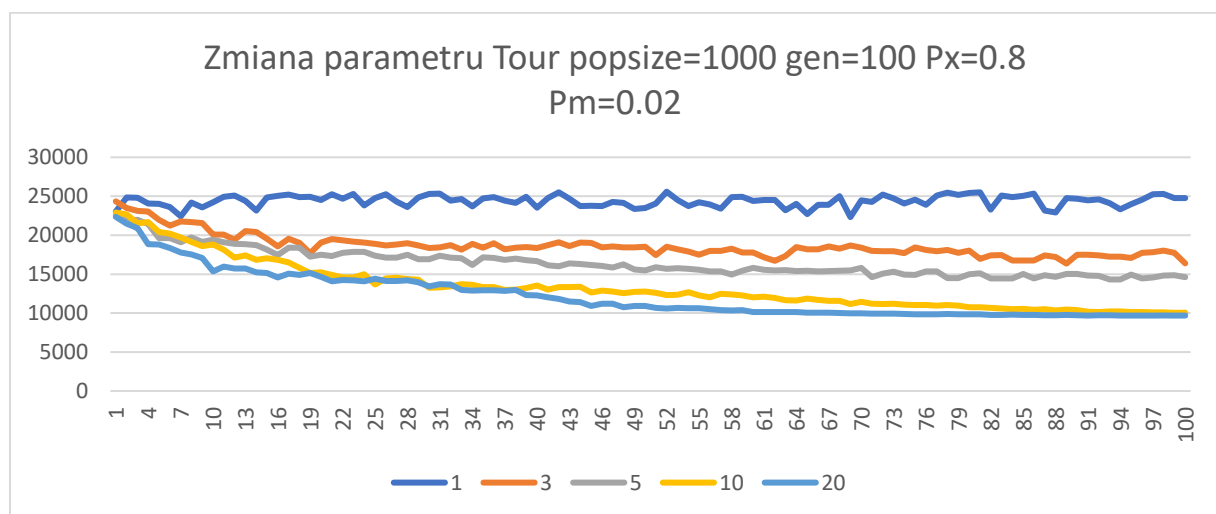
- berlin11_modified.tsp
- berlin52.tsp
- kroA100.tsp
- kroA150.tsp
- kroA200.tsp
- fl417.tsp

Parametry takie jak Gen, Pop_size, Px, Pc, Tour wyprowadziłem na podstawie testów.

2. Wyprowadzenie najbardziej optymalnego Tour

Do testowania wykorzystałem berlin52 o parametrach popsize=1000 gen=100 Px=0.8 Pm=0.02

Testowanie wartości tour: 1,3,5,10,20

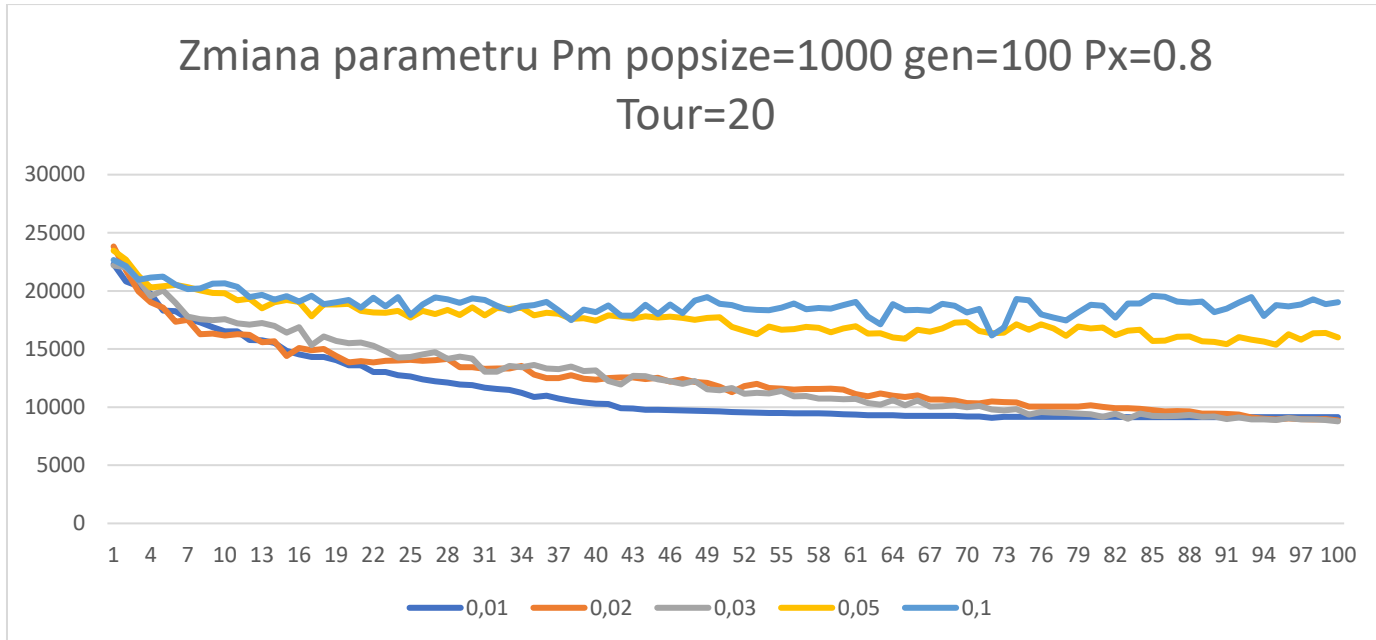


Najbardziej optymalne okazały się być wartości 10 i 20. Najlepsze dla Tour = 20

3. Wyprowadzenie najbardziej optymalnego parametru Pm

Do testowania wykorzystałem berlin52 o parametrach popsize=1000 gen=100 Px=0.8 Tour=20

Testowanie wartości Pm: 0.01, 0.02, 0.03, 0.05, 0.10

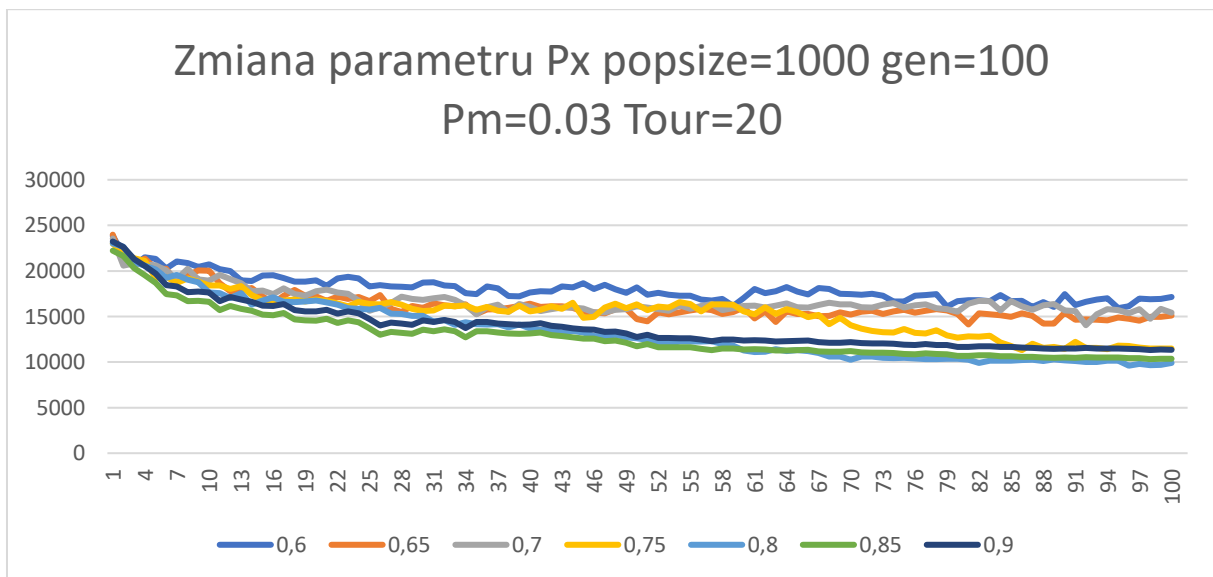


Najbardziej optymalne okazały się być wartości 0.02 i 0.03, przy czym ostatecznie najlepszy wynik osiągnąłem przy parametrze równym 0.03

4. Wyprowadzenie najbardziej optymalnego parametru Px

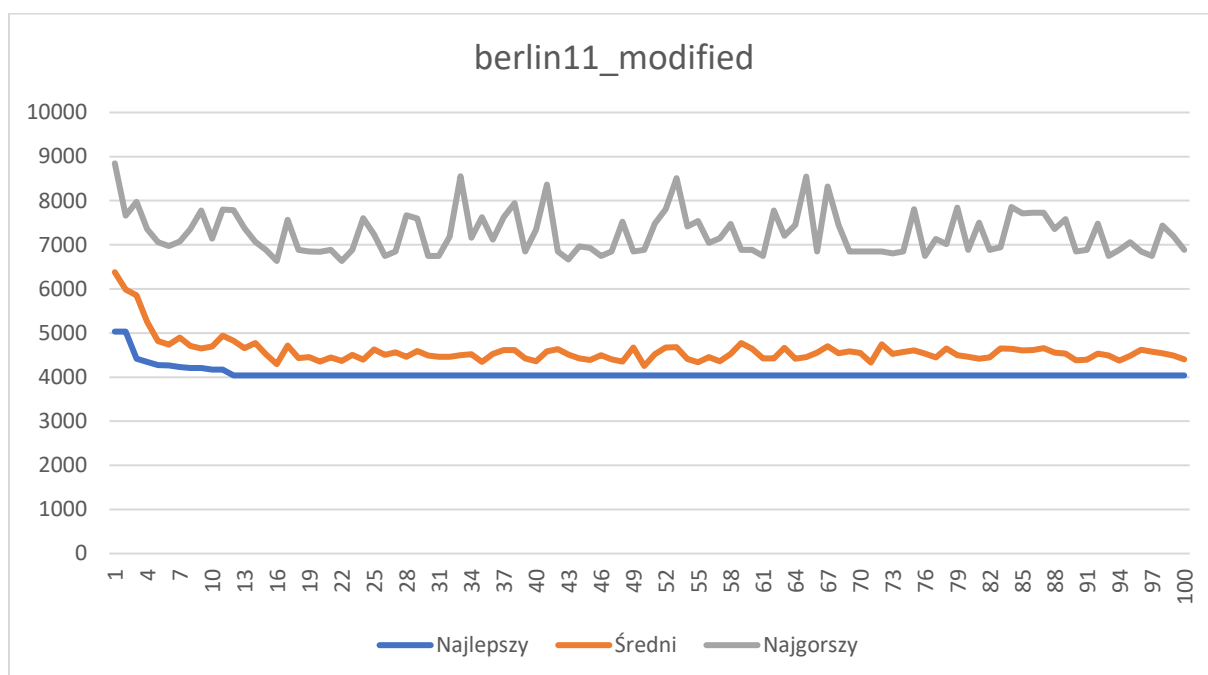
Do testowania wykorzystałem berlin52 o parametrach popsize=1000 gen=100 Pm=0.03 Tour=20

Testowanie wartości Px: 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9



Najbardziej optymalne okazały się być wartości 0.8 i 0.85, przy czym ostatecznie najlepszy wynik osiągnąłem przy parametrze Px równym 0.8

5. Wyniki dla berlin11_modified

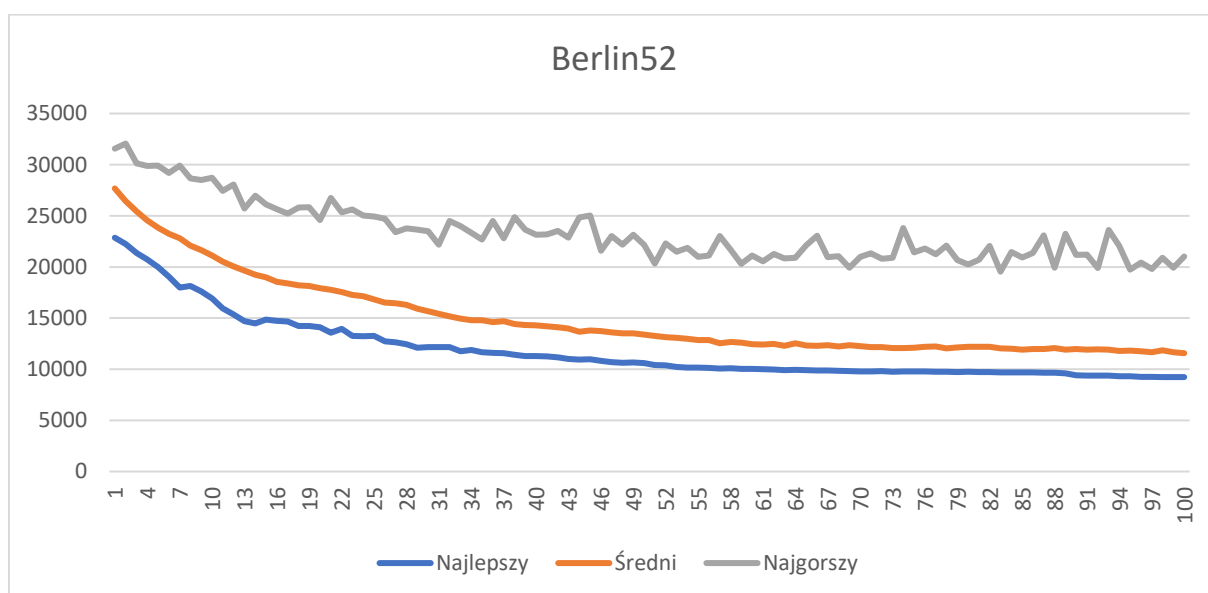


Parametry: popsize=1000 gen=100 Px=0.8 Pm=0.03 Tour=20

Wynik optymalny w 12 pokoleniu

Najlepszy wynik: 4038,438 Średni wynik: 4399,688 Najgorszy wynik: 6885,092

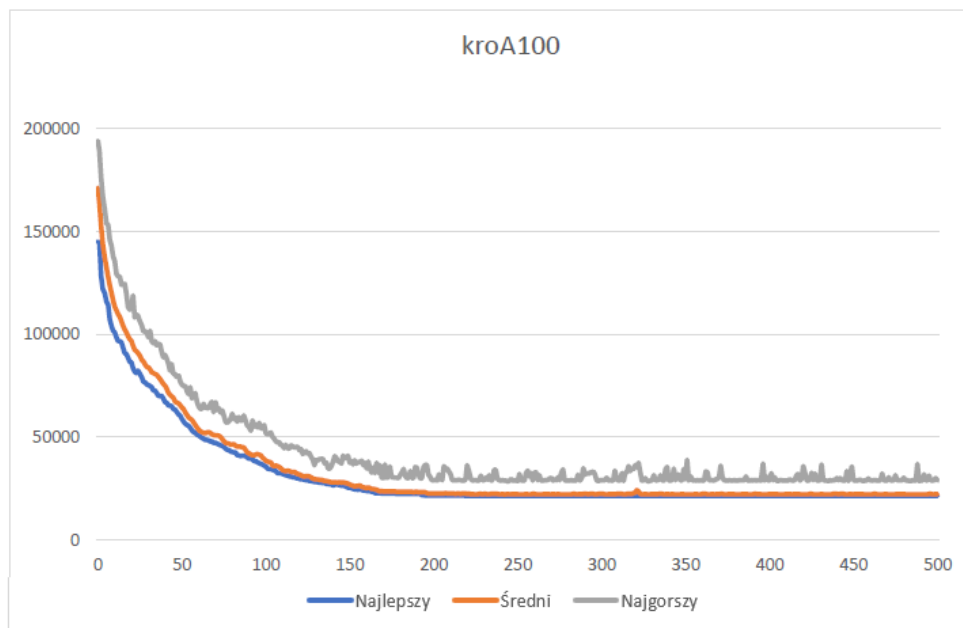
6. Wyniki dla berlin52



Parametry: popsize=1000 gen=100 Px=0.8 Pm=0.03 Tour=20

Najlepszy wynik: 8230,36 Średni wynik: 11565,59 Najgorszy wynik: 21013,4

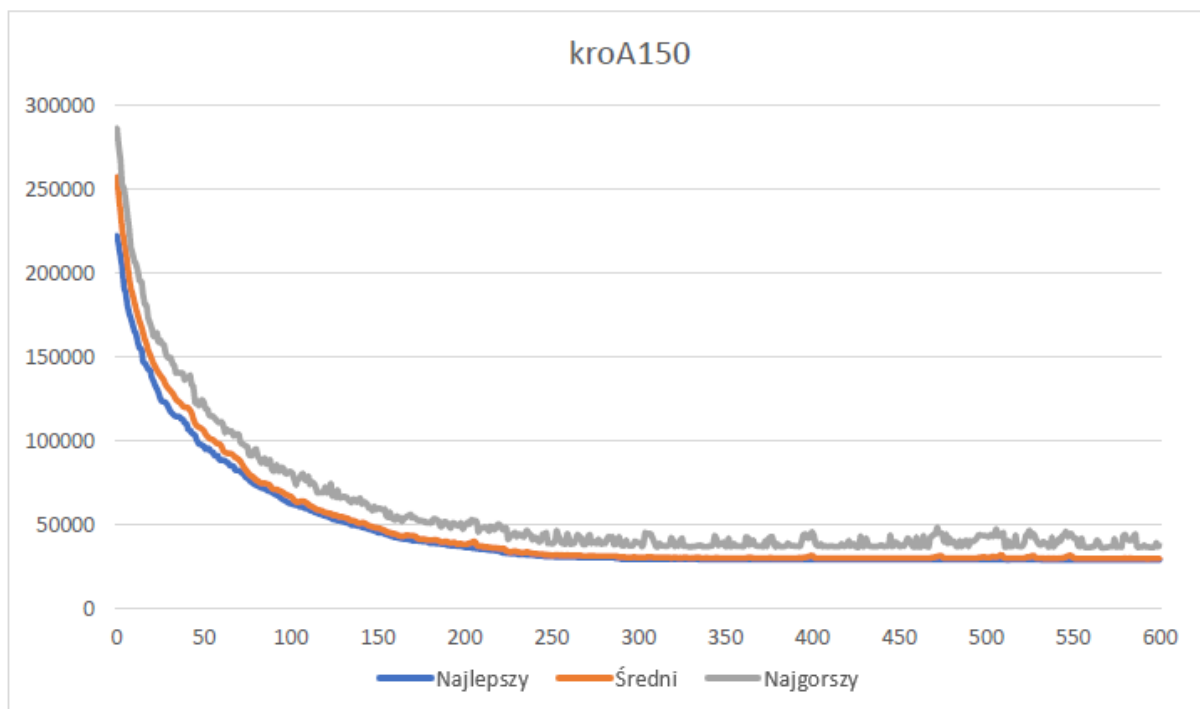
7. Wyniki dla kroA100



Parametry: popsize=1000 gen=500 Px=0.8 Pm=0.03 Tour=20

Najlepszy wynik: 21313,21 Średni wynik: 25532,213 Najgorszy wynik: 35325,243

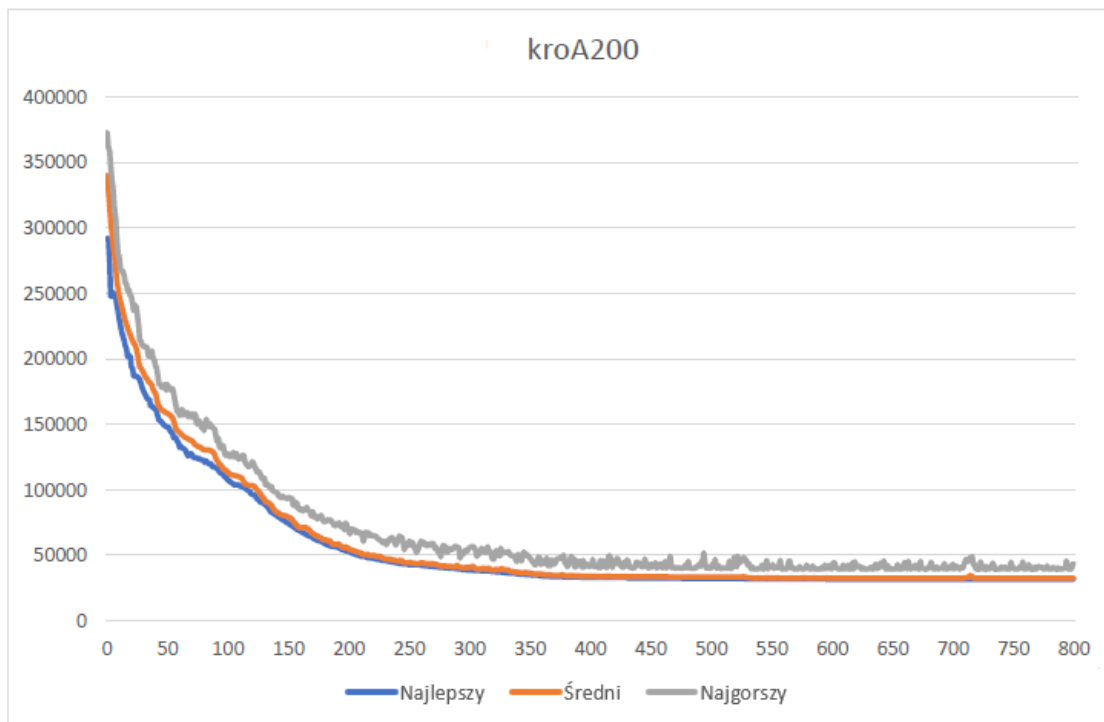
8. Wyniki dla kroA150



Parametry: popsize=1300 gen=600 Px=0.8 Pm=0.03 Tour=20

Najlepszy wynik: 28943,232 Średni wynik: 29635,324 Najgorszy wynik: 41345,334

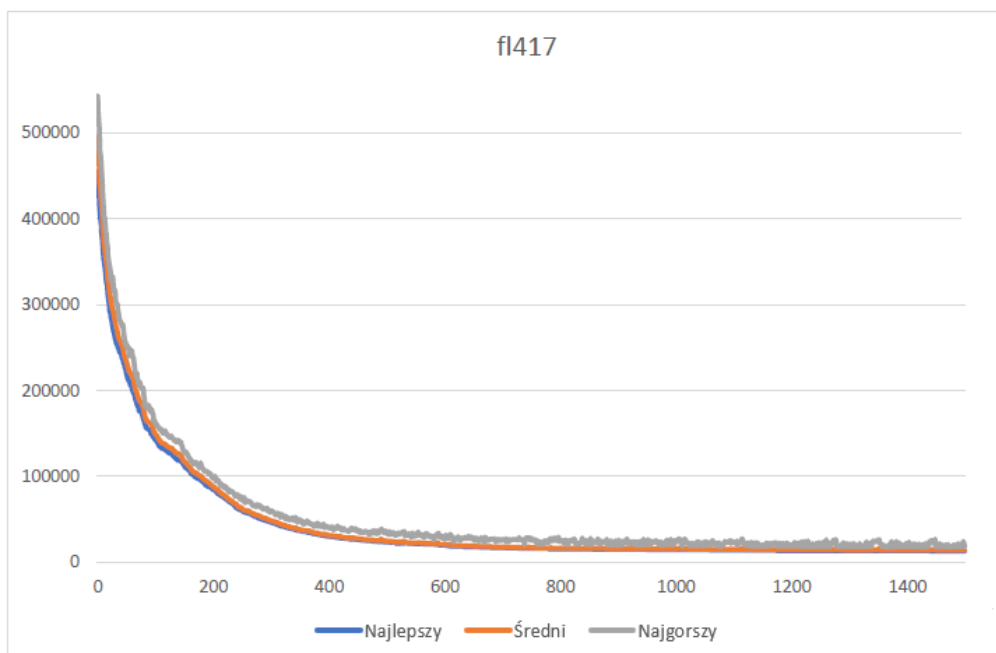
9. Wyniki dla kroA200



Parametry: popsize=1500 gen=800 Px=0.8 Pm=0.03 Tour=20

Najlepszy wynik: 31344,232 Średni wynik: 32394,42 Najgorszy wynik: 47567,325

10. Wyniki dla fl417



Parametry: popsize=1500 gen=1500 Px=0.8 Pm=0.03 Tour=20

Najlepszy wynik: 12934,432 Średni wynik: 14242,865 Najgorszy wynik: 26423,532

11. Porównanie wyników z algorytmami zachłannym i losowym

Instancja	Optymalny	Zachłanny	Losowy				Ewolucyjny			
			najlepsze	średnie	najgorsze	std	najlepsze	średnie	najgorsze	std
berlin11	4038	4543	4374	7230	8532	928	4038	4123	4244	223
berlin52	7542	898	23432	29433	312423	1375	8230	11543	12243	582
kroA100	21282	26856	93453	184543	194324	7932	21313	25764	26342	732
kroA150	26524	33609	173324	243243	268432	8454	28943	28967	31432	485
kroA200	29368	35798	289852	322343	363432	8543	31344	32575	34543	612
fl417	11861	15191	4756324	493453	519433	14643	12934	13976	14547	563

12. Wnioski

Algorytm generyczny daje, przy odpowiednich współczynnikach, wyniki znacznie bliższe optymalnym niż. Wymaga niestety dużej mocy obliczeniowej i w przypadku działania mojego programu w środowisku Python dla dużych plików danych i populacji oraz liczby genów program liczył kilka godzin (fl417). Odpowiedni wybór parametrów ma bardzo duże znaczenie, co postarałem się udowodnić w punktach 2,3,4. Wyniki są dokładniejsze z każdym kolejnym pokoleniem i w przypadku większych pokoleń wyniki również są lepsze.