

LAB2 - STRATEGIE EWOLUCYJNE

Dane początkowe

Do wygenerowania startowej populacji skorzystałem z funkcji **generate_population()**, za jej pomocą utworzyłem populację które będzie wykorzystywana do wszystkich testów jako baza, by zminimalizować wpływ losowości na wynik zadania.

```
population = [  
    [-24, -28, -67, -5, 28, -18, -32, -11, 32, -62],  
    [-73, 31, 60, -23, 76, -100, -46, 38, 99, -52],  
    [-50, -3, -58, 11, -92, -32, -41, -19, 98, -100],  
    [-2, -83, 86, 85, -34, -75, 52, 44, -11, -16],  
    [6, 21, -16, -17, 52, 47, 0, -19, -60, 49],  
    [86, -97, -58, 3, 27, 5, -79, -97, -30, -61],  
    [-93, 86, -5, -71, 76, -57, -93, -52, 15, 4],  
    [75, -8, -69, -16, -39, 56, -100, -40, -32, -4],  
    [-19, 60, 88, 32, 61, -16, 21, -59, -17, -93],  
    [37, 52, 12, 43, -100, -21, 18, -70, -77, -50],  
    [-65, 45, -12, -69, 28, -87, -89, -73, -90, 24],  
    [-14, 81, -35, -20, 92, -98, 24, -35, 94, 66],  
    [70, -69, 93, -99, 53, 12, -64, 2, 10, -50],  
    [-41, 10, -44, 97, -16, -30, 72, 52, -72, 16],  
    [-48, 68, 0, -35, 53, -65, -54, -54, 35, 6],  
    [-40, -15, -26, 60, -74, 20, 76, 97, -64, -39],  
    [10, -47, -52, -1, 66, 77, 4, -34, -60, 45],  
    [71, -40, 78, -73, 6, -42, -92, 29, -70, -69],  
    [-95, 9, -23, -6, -36, -16, -7, 13, -67, -21],  
    [78, 68, 73, -50, -62, -28, -62, 93, -51, -6]  
]
```

Pozostałe parametry wykorzystywane w testach:

D - wymiar, przyjąłem zgodne z treścią zadania równe 10,

τ - parametr tau, zgodne z treścią zadania, $1/\sqrt{10}$,

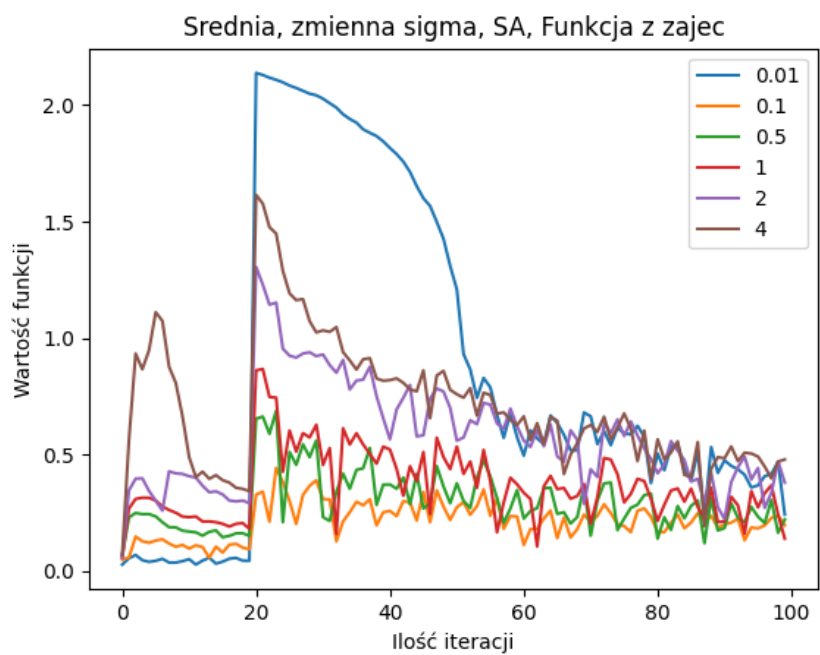
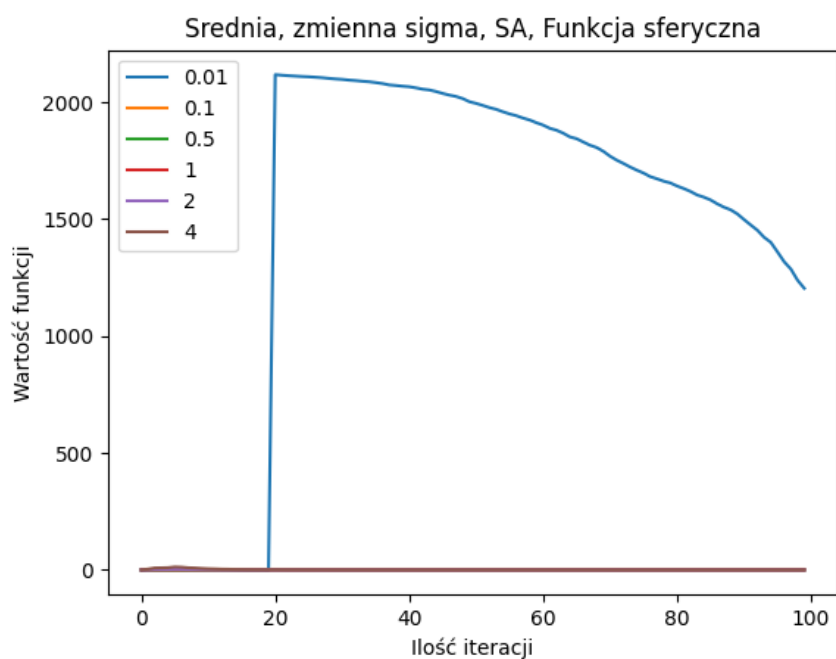
σ - zależna od testów, ustalana przeze mnie

Λ - ilość potomków, zależne od testów, ustalane przeze mnie,

μ - wielkość populacji, ustaliłem ją na stałą wartość równą 20

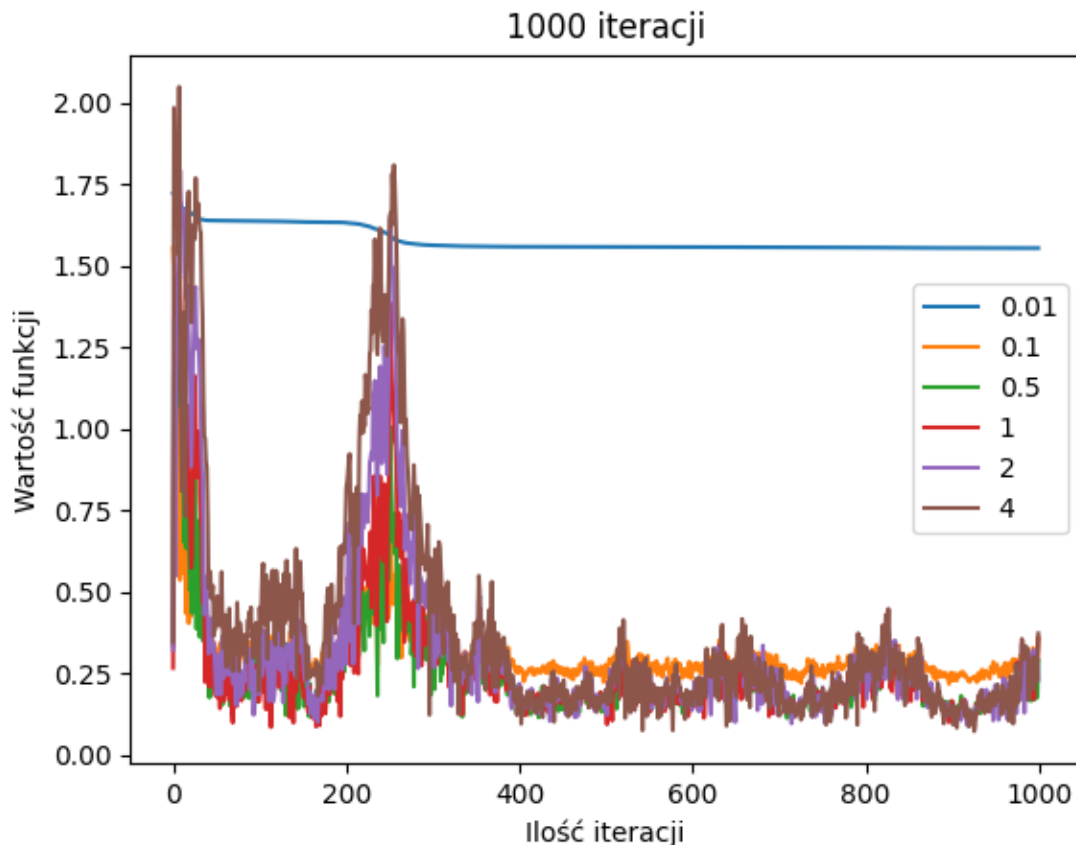
Dodatkowe parametry

Ilość ziaren, z których korzystałem ustaliłem na jedno. Przy testach które opierały się na uśrednionych wartościach z pięciu ziaren otrzymywałem “szpilki”, które wyraźnie kłóciły się z wynikami uzyskiwanymi z pojedynczego testu.



Kod który wykorzystywałem do obliczenia średniej z wyników znajduje się w funkcjach **experiment_average** oraz **experiment_best**. Zostały jednak zakomentowane żeby nie zaburzać wyników. Testy więc wykonywane były na pojedynczym ziarnie stałym między testami.

Ilość iteracji w przeprowadzanych testach została przeze mnie ustalona na 100. Zauważyłem że każda funkcja wyraźnie się spłaszcza gdy dąży 1000 iteracji, oraz występuje wyraźne zaburzenie wartości funkcji celu w okolicach 200-300 iteracji, jednak dla zachowania ogólnej czytelności wykresów zdecydowałem się na jedynie 100 iteracji.



Opis planowanych eksperymentów

Planuję przeprowadzić ogólnie 8 testów, podzielone na dwie grupy. Testów które swój wynik uzależniają od zmienności σ oraz testów uzależnionych od zmian Λ . Każda grupa będzie składała się z testów dla kombinacji różnych funkcji celu oraz metod. Każdy test odbywał się będzie dla zestawu sześciu funkcji zależnych od różnych σ lub Λ .

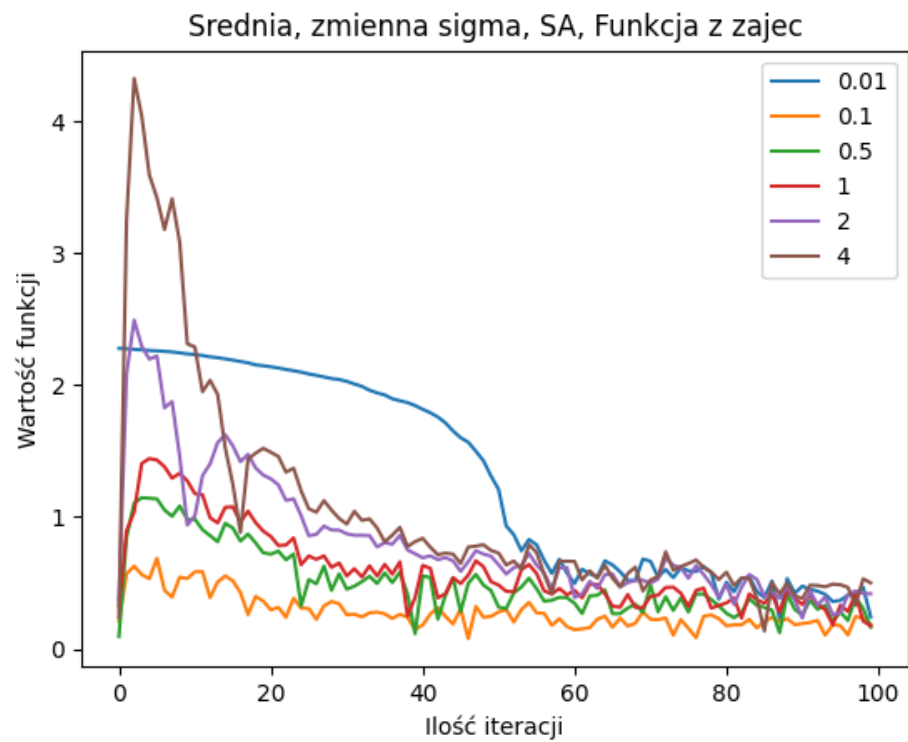
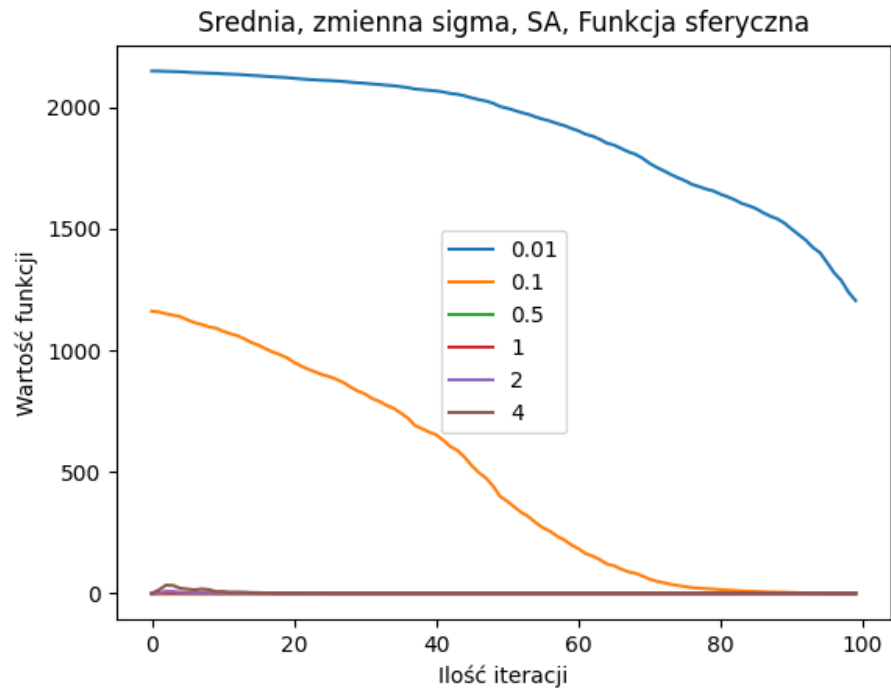
Wartości σ to [0.01, 0.1, 0.5, 1, 5, 10],

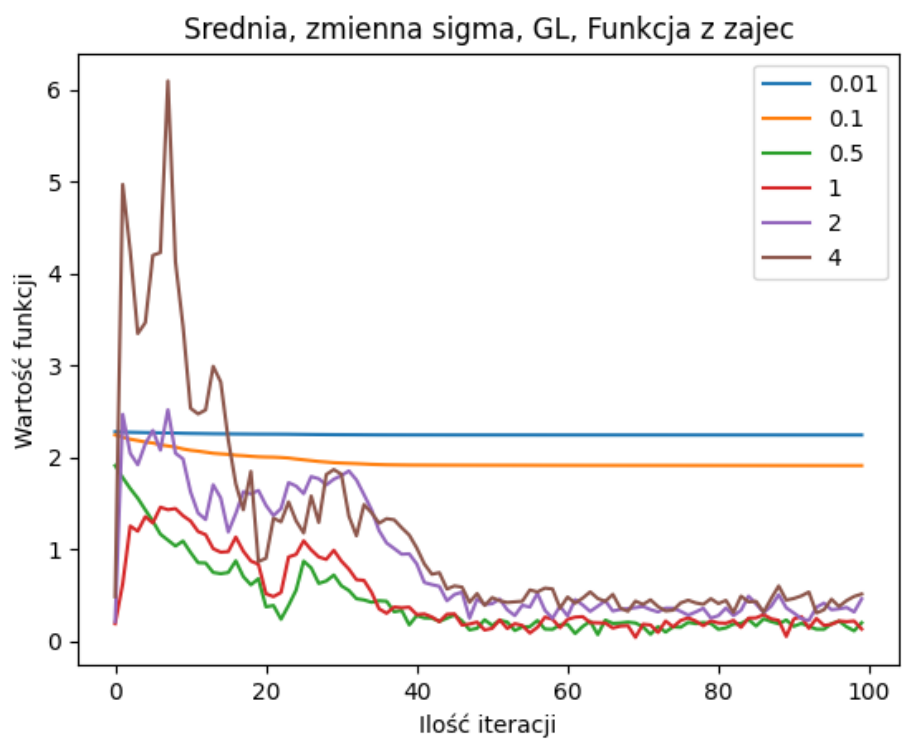
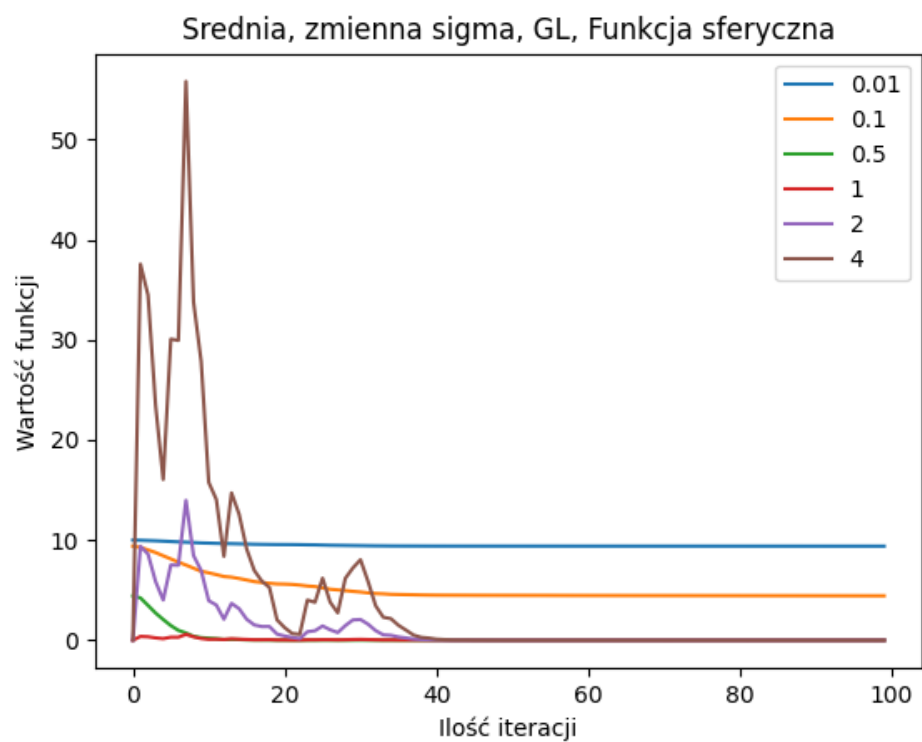
Wartości Λ to [20, 25, 30, 35, 40, 45]

Wykresy przedstawiają wartość funkcji obliczoną dla punktu reprezentującego średnią wartość dla każdej generacji osobników.

Przebieg eksperymentów numerycznych.

Eksperymenty rozpocząłem od badania wpływu zasięgu mutacji na działanie poszczególnych metod w zależności od wybranych funkcji celu.

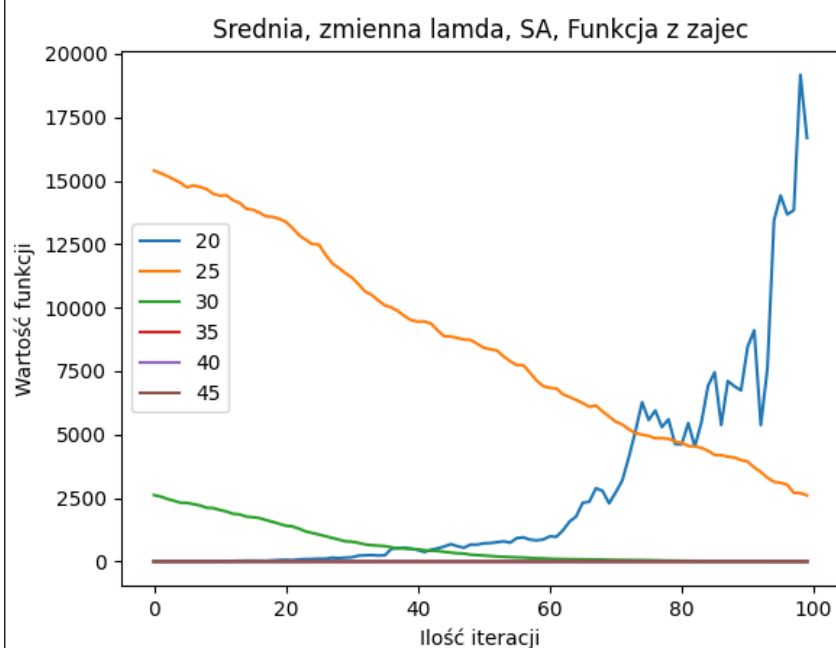
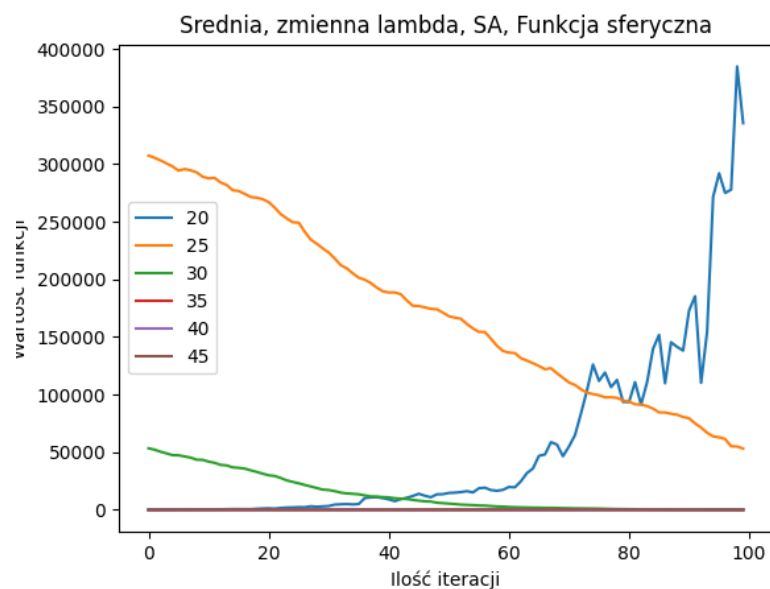




Zauważyłem drastycznie różne wyniki dla kombinacji metody samoadaptacji oraz funkcji sferycznej. Jako jedyny z czterech eksperymentów osiągnął wartość dla punktu średniego równą około 2000 gdy pozostałe oscylowały w okolicach 10.

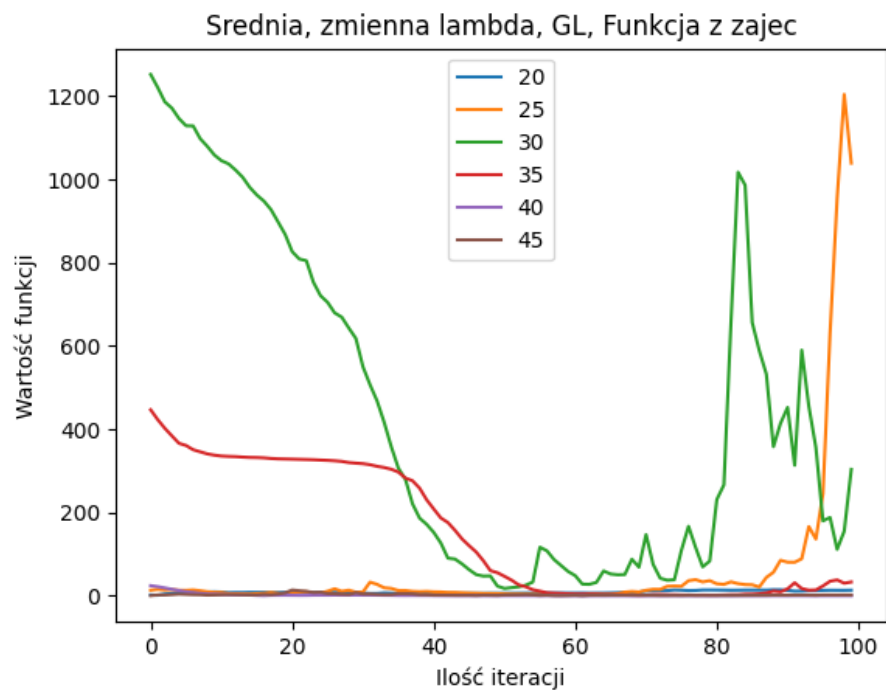
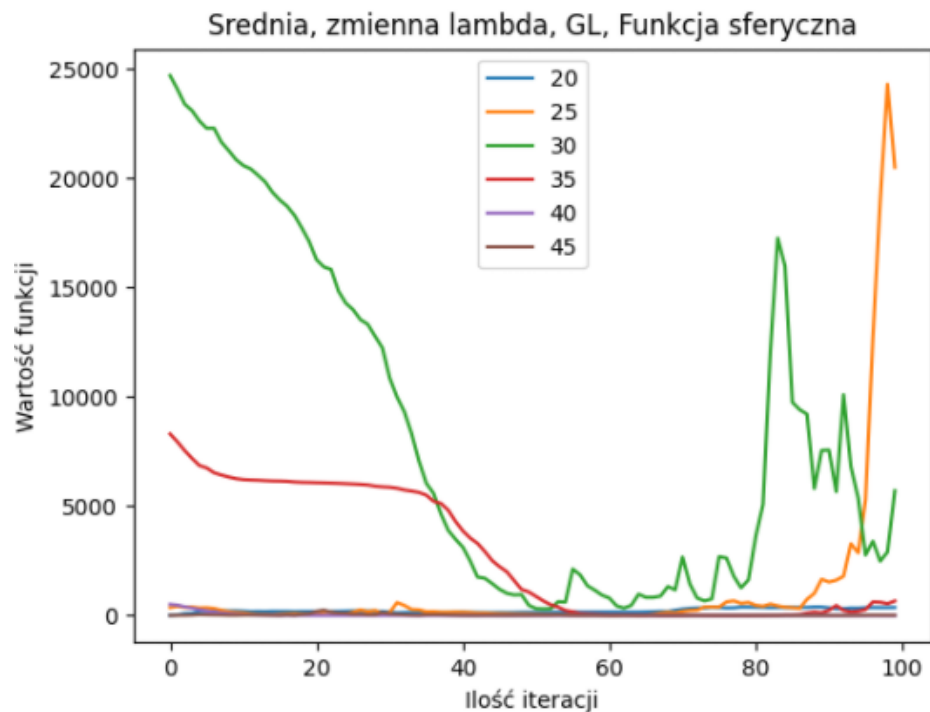
Zauważyć również można że kształt funkcji jest najbardziej spłaszczony dla sigmy w okolicach większych od 0.1 i mniejszych od 1. Jednak trzeba zwrócić uwagę że dla wartości 0.1 oraz mniejszej, przy metodzie GL funkcja spłaszcza się bardzo daleko od minimum.

Kolejne przeprowadziłem eksperymenty dla zmiennej lambdy. Pośród tych testów zauważyć można pewną zależność, mianowicie proporcje obu wykresów, niezależnie od funkcji celu są podobne.



Mimo że można zauważyć sporą różnicę w wartości samej funkcji, niezaprzeczalnie podobieństwo obu wykresów jest wyraźnie i łatwe do spostrzeżenia.

Podobny efekt można zauważyć dla drugiej metody, Logarytmiczno-Gaussowskiej, również i tutaj, zauważalne są wyraźne różnice w wartości funkcji w danych punktach. Mimo wszystko sam kształt i proporcje funkcji niezależnie od wybranej funkcji celu jest podobny do siebie przy obu metodach.



Wnioski

Po dokładniejszej inspekcji można zauważyć pewien zestaw podobieństwa dla każdego testu. Zarówno przy badaniu wpływu siły mutacji jak przy wielkości populacji zauważyć można że w tym przypadku sama funkcja celu ma wpływ na dokładną wartość funkcji, jednak samo zachowanie się algorytmu oraz proces w jakim będzie ono szukało minimum jest bardzo zbliżony, niezależnie od funkcji celu.

Ponad wnioski wyjątkowe dla tego zadania można zauważyć bardziej oczywiste rzeczy. Między innymi wielkość populacji wpływa korzystnie na szybkość z jaką poszukiwane jest minimum. Duża ilość osobników łatwiej dopasowuje się podczas przeszukiwań minimum.

Natomiast wpływ siły mutacji jest zależny od wykorzystanej metody, względnie duża wartość sigmy działa korzystnie na przeszukiwanie dzięki samoadaptacji, osobniki lepiej się regulują i szybciej znajdują dobry wynik, przez metodzie gaussowskiej efekt może być odwrotny ponieważ osobniki po mutacji mogą minąć minimum zamiast do niego się zbliżać co będzie efektem nieporządanym,