

Dokumentacja zadania laboratoryjnego -
badanie algorytmu $(1+1)$ -ES

Krzysztof Wyrzykowski, nr indeksu 331455

7 listopada 2024

0.1 Opis działania algorytmu

Algorytm (1+1)-ES to algorytm ewolucyjny służący do rozwiązywania problemów optymalizacyjnych. Działa na populacji składającej się z jednego osobnika. Wykorzystuje on mechanizm mutacji z samoczynną adaptacją jej zasięgu.

Algorytm rozpoczyna działanie w punkcie początkowym x_0 . W każdej iteracji do punktu x_0 dodawany jest losowy wektor z rozkładu normalnego przeskalowany o parametr σ .

$$x_{k+1} = x_k + \sigma \mathcal{N}(0, 1)$$

Następnie sprawdzamy czy wartość funkcji w punkcie x_k jest mniejsza (dla problemu minimalizacji), niż najmniejsza dotąd znaleziona wartość. Jeśli tak to mamy sukces i zapamiętujemy punkt x_k oraz wartość w tym punkcie. Zadaniem algorytmu jest utrzymanie współczynnika sukcesów na poziomie $1/5$, bo ta wartość pozwala na uzyskanie największego średniego spadku wartości najmniejszej. Co n iteracji sprawdzamy jaką część z nich zakończyła się sukcesem. Jeśli wartość ta jest większa niż $1/5$ to znaczy, że musimy zwiększyć wartość parametru σ . Wartość zwiększamy poprzez pomnożenie jej przez $1,22$, jest to liczba dobrana eksperymentalnie na podstawie badań nad funkcją kwadratową. Jeśli mniej niż $1/5$ iteracji zakończyło się sukcesem to znaczy, że musimy zmniejszyć wartość parametru σ . Robimy to poprzez pomnożenie jego wartości przez 0.82 . Warunkiem stopu tego algorytmu jest maksymalna liczba iteracji.

0.2 Opis planowanych eksperymentów numerycznych

0.2.1 Cel przeprowadzenia eksperymentów

Celem pierwszego etapu eksperymentów było zbadanie zachowania algorytmu dla różnych wartości początkowego zasięgu mutacji σ . W kolejnym etapie celem było zbadanie wpływu parametrów sterujących regułą mutacji. Po znalezieniu wartości dla których funkcja znajdowała najlepsze rozwiązania w zadanej liczbie iteracji przeprowadziłem porównanie algorytmu 1+1-ES z algorytmem gradientu prostego.

0.2.2 Parametry eksperymentów

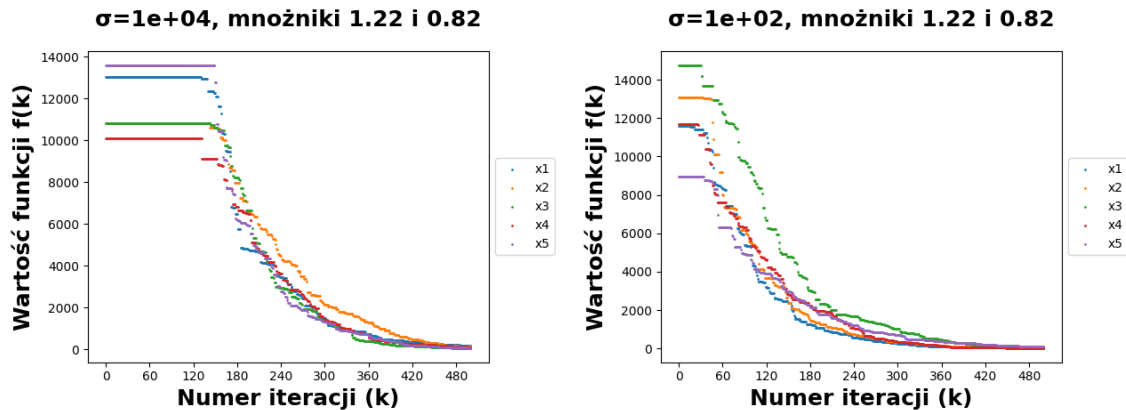
Do testów wykorzystałem trzy funkcje - funkcję kwadratową oraz funkcje f3 i f7 z benchmarku CEC2017. Jako zbiór punktów początkowych przyjąłem zbiór $[-100, 100]^{10} \in R^{10}$. Przyjąłem maksymalną liczbę iteracji równą 500 oraz wartość parametru n równą 5. Każdy pomiar został przeprowadzony dla pięciu losowo wybranych punktów początkowych. W celu skompensowania stochastyczności badanego algorytmu, jego działanie zostało powtórzone dziesięciokrotnie dla każdego punktu początkowego, a prezentowane na wykresach wartości to średnia arytmetyczna wartości w badanym punkcie dla k -tej iteracji.

0.3 Etap 1. - parametr σ

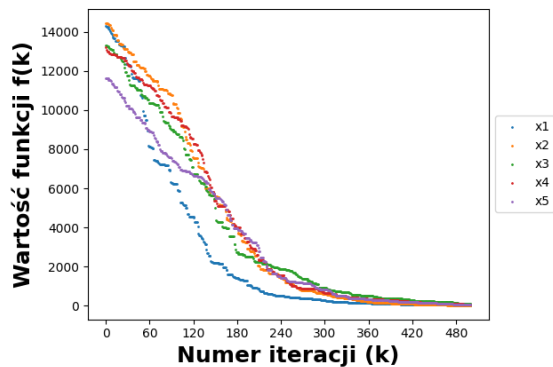
0.3.1 Przebieg badań

Dla każdej funkcji sprawdziłem działanie algorytmu dla parametrów σ równych: $1e4$, $1e2$, $1e0$, $1e-2$, $1e-4$. Na tej podstawie przygotowałem wykresy zbieżności, które pokazują jaka była wartość badanej funkcji w punkcie x_k dla k -tej iteracji.

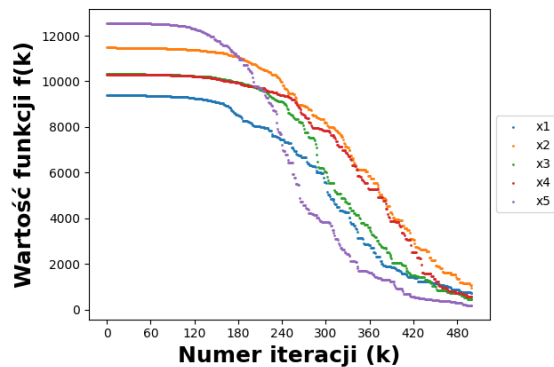
Funkcja kwadratowa



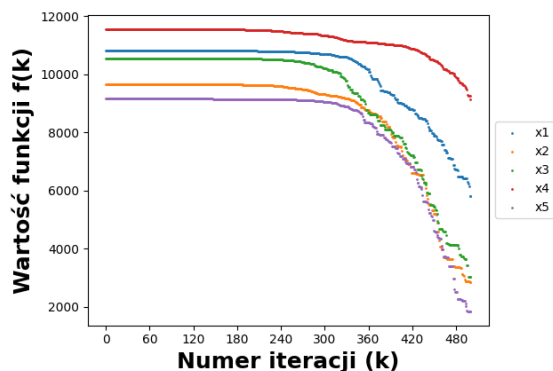
$\sigma=1e+00$, mnożniki 1.22 i 0.82



$\sigma=1e-02$, mnożniki 1.22 i 0.82



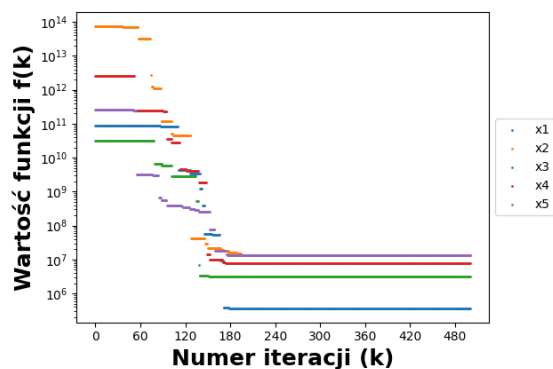
$\sigma=1e-04$, mnożniki 1.22 i 0.82



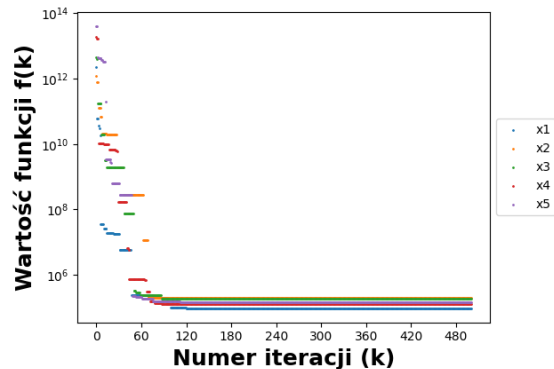
Z wykresów wynika, że pomimo różnych początkowych wartości zasięgu mutacji. Algorytm w prawie każdym przypadku zdołał dotrzeć do minimum globalnego funkcji. Dla wartości σ równej $1e-4$, zadana liczba iteracji okazała się niewystarczająca. Przy zbyt wysokiej wartości σ , algorytm na początku trudności ze znalezieniem lepszych rozwiązań, ale gdy wartość się dostosuje algorytm znajduje optimum. Dla zadanej funkcji algorytm najszybciej zbliża się do optimum przy σ równym $1e2$.

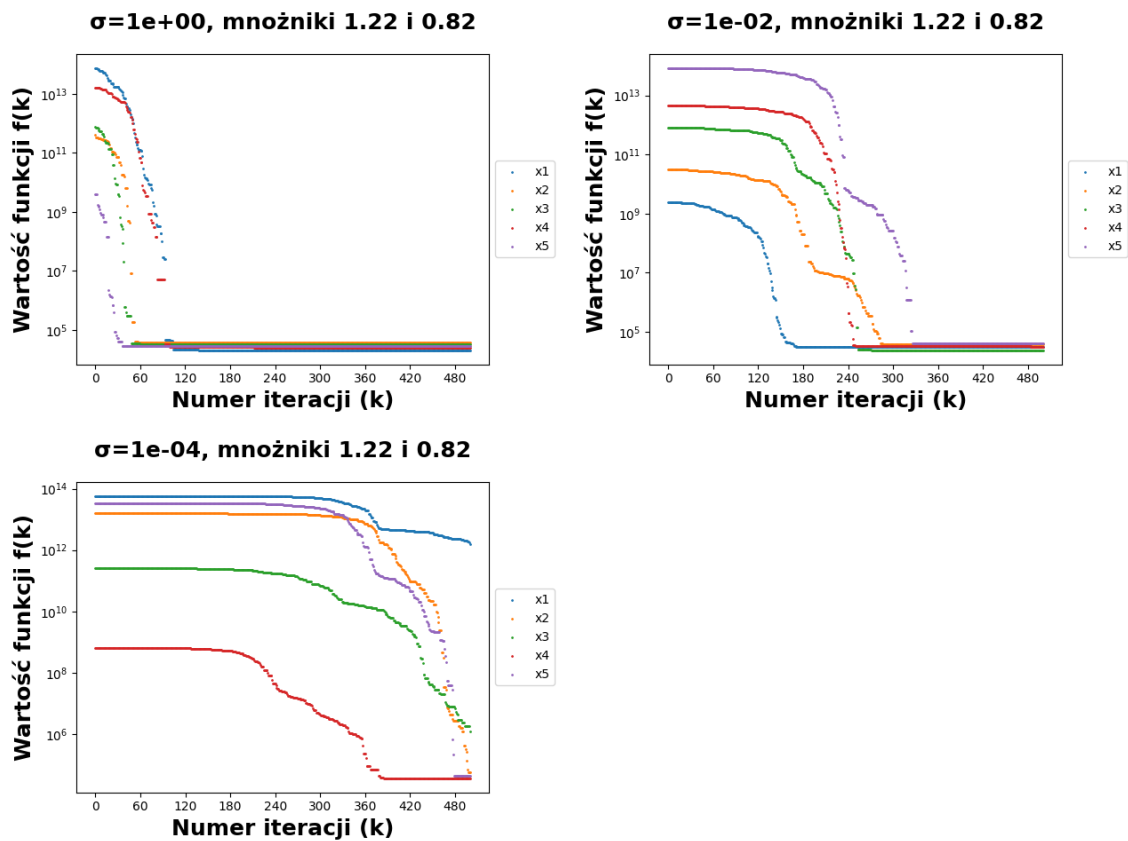
Funkcja f3

$\sigma=1e+04$, mnożniki 1.22 i 0.82



$\sigma=1e+02$, mnożniki 1.22 i 0.82

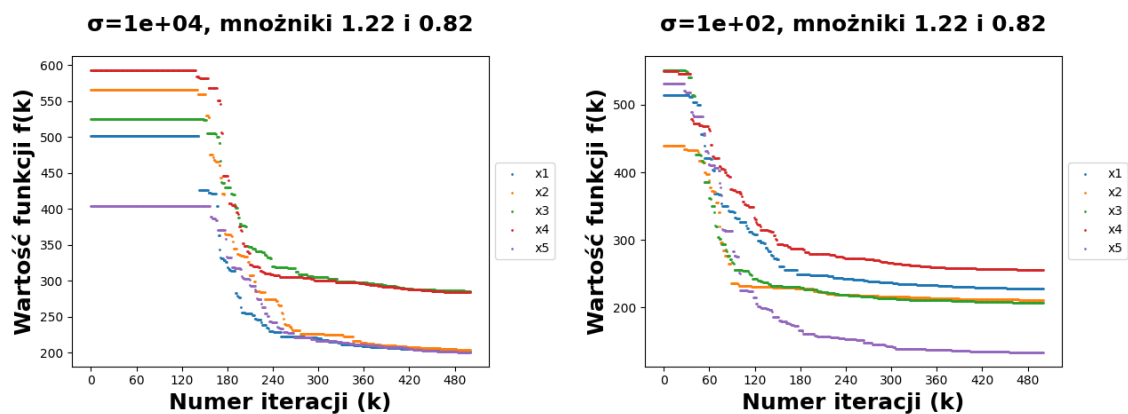


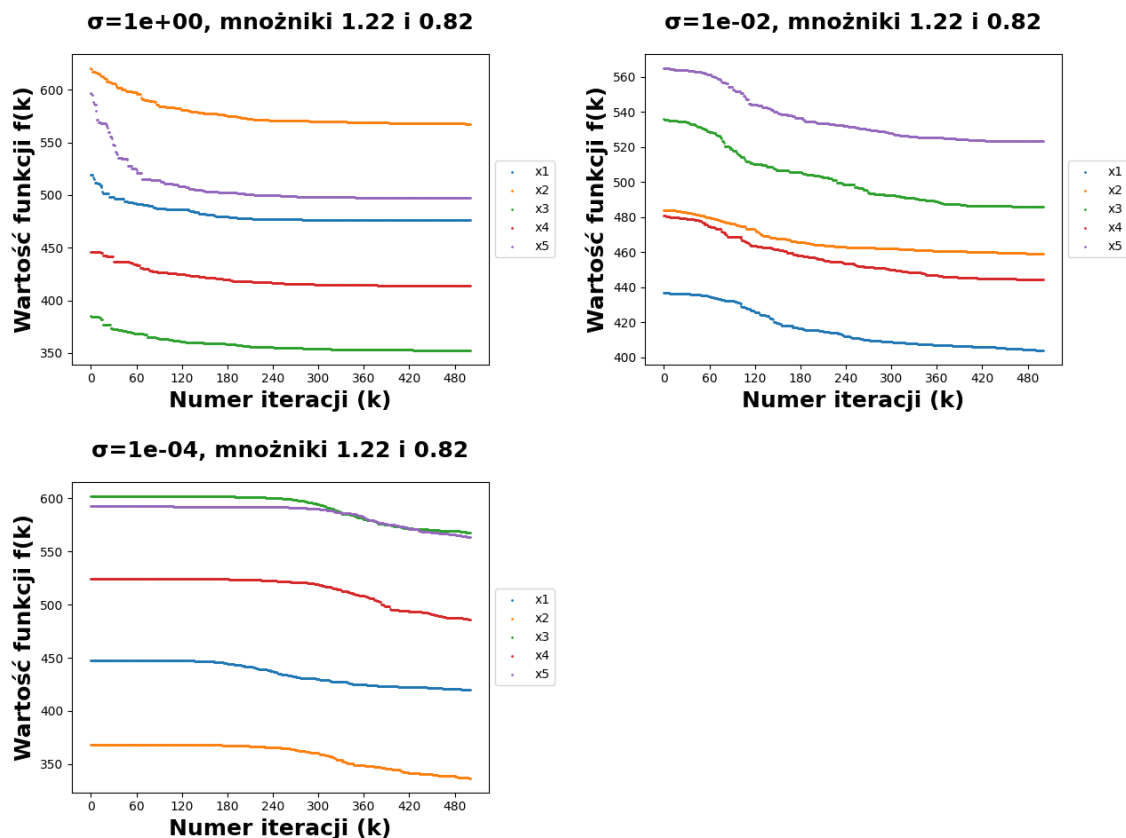


Rysunek 1: Przebiegi zmian wartości dla funkcji f3

Duża wartość parametru σ powoduje, że funkcja przez większość iteracji nie znajduje lepszych rozwiązań. Dla najmniejszej wartości algorytm powoli zaczyna dążyć do rozwiązania, ale potrzebowałby większej liczby iteracji. Dla pozostałych wartości otrzymujemy podobne wyniki, różniące się w początkowej fazie.

Funkcja f7





Rysunek 2: Przebiegi zmian wartości dla funkcji f7

W tym przypadku małe wartości początkowe okazują się być niewystarczające, aby algorytm był w stanie szybko znajdować znacząco lepsze rozwiązania. Zdecydowanie lepsze okazały się być wartości $1e2$ i $1e4$.

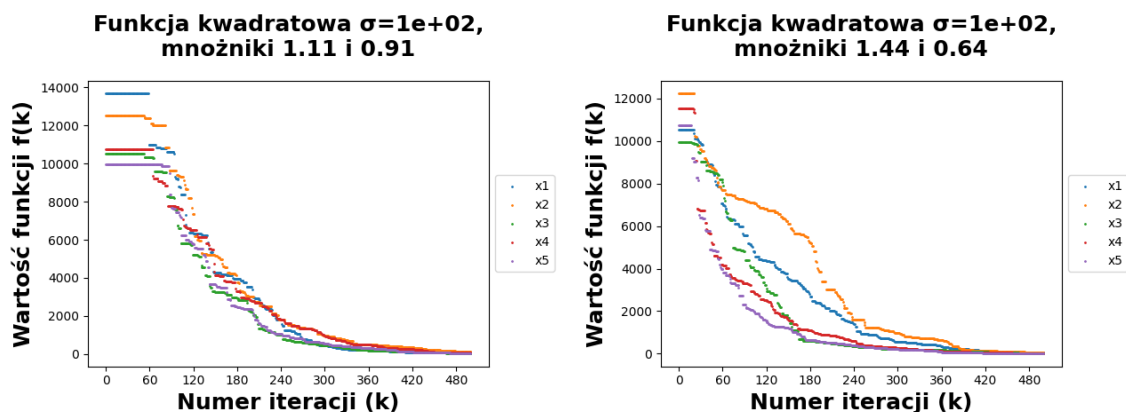
Wnioski

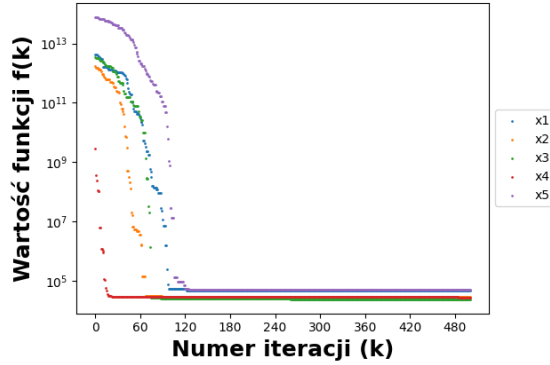
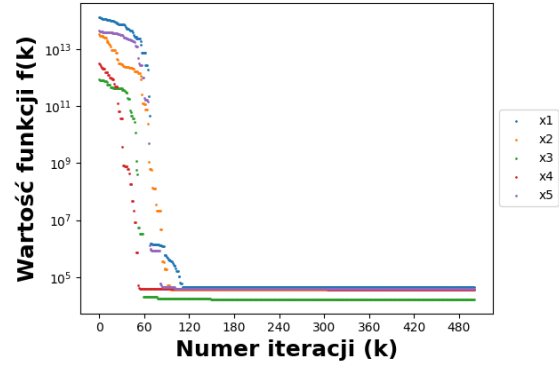
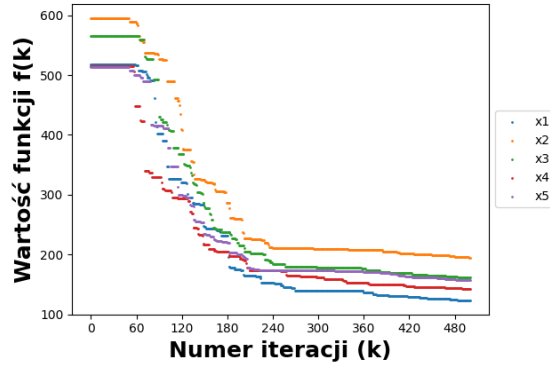
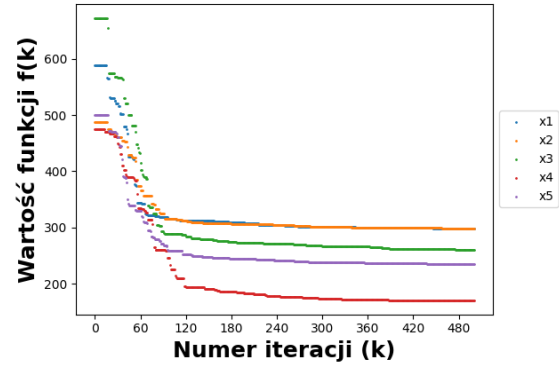
Dobranie odpowiedniej wartości parametru σ zmniejsza minimalną liczbę iteracji koniecznych do uzyskania satysfakcjonującego wyniku. Niemniej jednak zdolność adaptacyjna algorytmu jest na tyle wysoka, że potrafi on dostosować zasięg mutacji, aby znaleźć rozwiązanie w większości sytuacji. Można zauważyć że dla $\sigma=100$ algorytm poradził sobie z każdą zadaną funkcją.

0.4 Etap 2. - parametry sterujące zasięgiem mutacji

0.4.1 Przebieg badań

Dla wartości σ , które dawały najlepsze rozwiązania dla poszczególnych funkcji przeprowadziłem badania dwukrotnie zwiększając oraz zmniejszając wartości odpowiadające za sterowanie zasięgiem mutacji.



Funkcja f3 $\sigma=1e+00$, mnożniki 1.11 i 0.91Funkcja f3 $\sigma=1e+00$, mnożniki 1.44 i 0.64Funkcja f7 $\sigma=1e+02$, mnożniki 1.11 i 0.91Funkcja f7 $\sigma=1e+02$, mnożniki 1.44 i 0.64

Wnioski

Z wykresów wynika, że zmiany w tych parametrach nie mają wpływu na charakter funkcji przebiegu zmian wartości. Większe wartości parametrów powodują większe skoki wartości funkcji, jednakże różnice są niezauważalne. Oznacza to że wartości 1.22 oraz 0.82 można uznać za uniwersalne.

0.5 Etap 3. - porównanie algorytmów 1+1-ES oraz SGD

W celu porównania obu algorytmów przeprowadziłem obliczenia na 100 losowo wybranych punktach początkowych, a następnie zliczyłem punkty dla których algorytmy uzyskały lepsze wyniki. Dobrałem parametry σ i α na podstawie badań przeprowadzonych we wcześniejszych etapach oraz podczas badań nad algorytmem SGD.

Funkcja	Lepsze wyniki 1+1-ES	Lepsze wyniki SGD	Średnia różnica	p-wartość
kwadratowa	0	100	2,09 e-4	3,90 e-18
f3	100	0	2,15 e+8	3,90 e-18
f7	100	0	1,60 e+3	3,90 e-18

Tabela 1: Porównanie algorytmów 1+1-ES oraz SGD - liczba wymiarów 10

Funkcja	Lepsze wyniki 1+1-ES	Lepsze wyniki SGD	Średnia różnica	p-wartość
kwadratowa	0	100	3,28 e+1	3,90 e-18
f3	100	0	6,48 e+8	3,90 e-18
f7	100	0	4,32 e+3	3,90 e-18

Tabela 2: Porównanie algorytmów 1+1-ES oraz SGD - liczba wymiarów 30

Wnioski

Z przeprowadzonych badań wynika, że algorytm SGD znalazł lepsze rozwiązanie dla funkcji kwadratowej, ale dla funkcji f3 oraz f7 lepszy był algorytm 1+1-ES. Różnica w średnich wartościach dla funkcji kwadratowej był rzędu $1e-4$ dla 10 wymiarów. Jest to wartość pomijalnie mała i można uznać, że dla tak prostej funkcji algorytmy zachowują się podobnie. Natomiast różnice dla pozostałych funkcji są znaczące, co pokazuje, że dla funkcji dla o bardziej złożonym kształcie lepszym działaniem wykazuje się 1+1-ES. Powtarzalność wyników została potwierdzona testem Wilcoxona, dla którego p-wartości były rzędu $1e-18$. Wymiar zadania nie miał wpływu na uzyskane wyniki.