

Bartosz Durys 229869 229869@edu.p.lodz.pl
Szymon Klewicki 229911 229911@edu.p.lodz.pl

Zadanie 1 - PSO i DE

1. Cel

Celem zadania było porównanie skuteczności działania algorytmów PSO i DE w zależności od zastosowanych parametrów dla funkcji wielu zmiennych.

2. Opis algorytmów

Zaimplementowaliśmy w języku Python algorytmy PSO (Particle Swarm Optimization) oraz DE (Differential Evolution). Wykorzystaliśmy je do przeprowadzenia wielu eksperymentów, w których optymalizowaliśmy funkcje wielu zmiennych. Z nich wybraliśmy najciekawsze i zaprezentowaliśmy je w sprawozdaniu.

W obu algorytmach występują wspólne parametry. Są to:

- Rozmiar populacji.
- Wymiar przestrzeni.
- Cel algorytmu - liczba iteracji, dokładność.
- Maksymalna liczba iteracji - warunek stopu.
- Maksymalna liczba stagnacji - warunek stopu.
- Liczba powtórzeń symulacji o danej konfiguracji.
- Parametry równania:
 - Słownik z równaniem jako funkcja anonimowa (lambda).
 - Minimalna wartość argumentów funkcji.
 - Maksymalna wartość argumentów funkcji.
 - Pożądana dokładność.
 - Poszukiwana wartość.

Algorytm PSO przebiega w następujących etapach:

1. Stworzenie roju cząstek.
2. Obliczenie początkowego przystosowania cząstek
3. Aktualizacja prędkości i pozycji cząstek.
4. Aktualizacja przystosowania cząstek.

5. Sprawdzanie warunków stopu algorytmu.
Podajemy poniższe parametry dla tego algorytmu:
 - Waga inercji.
 - Współczynnik poznawczy.
 - Współczynnik społeczny.

Algorytm DE przebiega w następujących etapach:

1. Stworzenie populacji.
2. Zastosowanie wybranego wariantu operacji mutacji (rand lub best).
3. Zastosowanie operacji krzyżowania.
4. Selekcja na podstawie wartości dopasowania.
5. Sprawdzanie warunków stopu algorytmu.

Podajemy poniższe parametry dla tego algorytmu:

- Czynniki skalujące.
- Prawdopodobieństwo krzyżowania.
- Wariant mutacji: "rand" lub "best".

3. Metodyka

Badaliśmy dwie funkcje celu:

1. **Sphere.**
 $f_1 = \sum_{i=1}^n x_i^2$
Zakres argumentów: $x_i \in (-100, 100)$
Wymiary przestrzeni: 20, 30
Szukane minimum: $f_1 = 0$
Dokładność: 0.0001
2. **Rastrigin.**
 $f_2 = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$
Zakres argumentów: $x_i \in (-5.12, 5.12)$
Wymiary przestrzeni: 20, 30
Szukane minimum: $f_2 = 0$
Dokładność: 30

Warianty zatrzymania algorytmu:

1. Liczba iteracji: 200.
2. Osiągnięcie dokładności.

Testowane rozmiary populacji:

1. 10.
2. 100.

Dodatkowe wartości parametrów:

- Liczba powtórzeń każdej symulacji: 10.
- Poziom stagnacji do zatrzymania algorytmu: 200.

- Maksymalna liczba iteracji do zatrzymania algorytmu: 10000.

Poniżej przedstawimy wybrane wartości parametrów dla poszczególnych algorytmów.

Tabela 1. Parametry dla algorytmu PS.

ID	Waga inercji	Współczynnik kognitywny	Współczynnik społeczny
PS1	0.9	1.9	0.1
PS2	0.9	0.3	0.1

Tabela 2. Parametry dla algorytmu DE.

ID	Parametr skalujący F	Prawdopodobieństwo krzyżowania	Wariant
DE1	0.3	0.1	rand
DE2	0.21	0.22	best

4. Wyniki

W tej sekcji przedstawimy wyniki naszych eksperymentów. Będą one dotyczyć poniższych kombinacji parametrów.

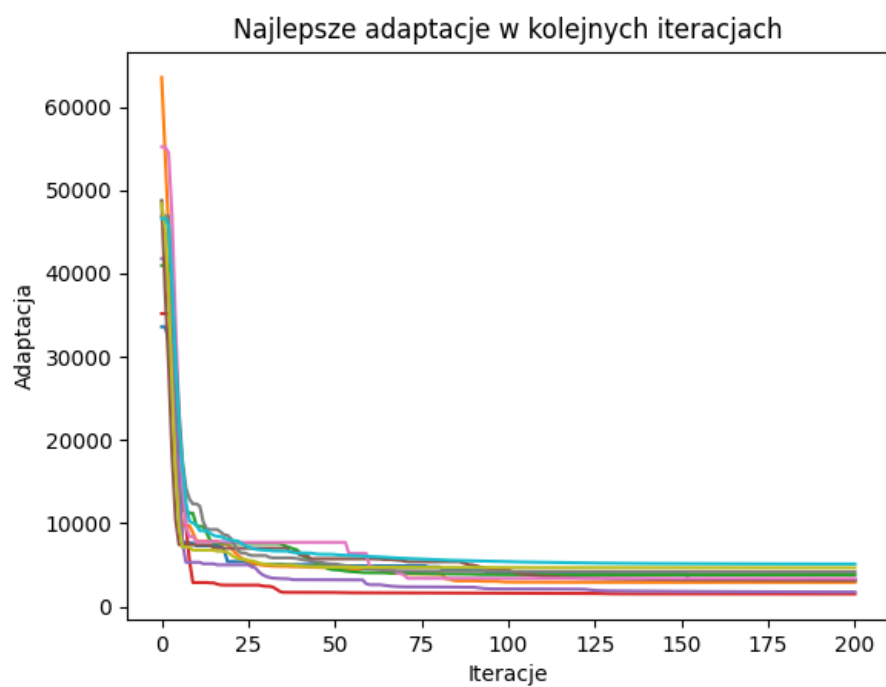
Tabela 3. Przedstawione eksperymenty.

ID2	ID	Funkcja	Warunek stopu	Rozmiar populacji	Wymiarowość
E1	PS1	Sphere	200	10	20
E2	PS2	Rastrigin	dokładność	100	30
E3	DE1	Sphere	200	10	20
E4	DE2	Rastrigin	dokładność	100	30

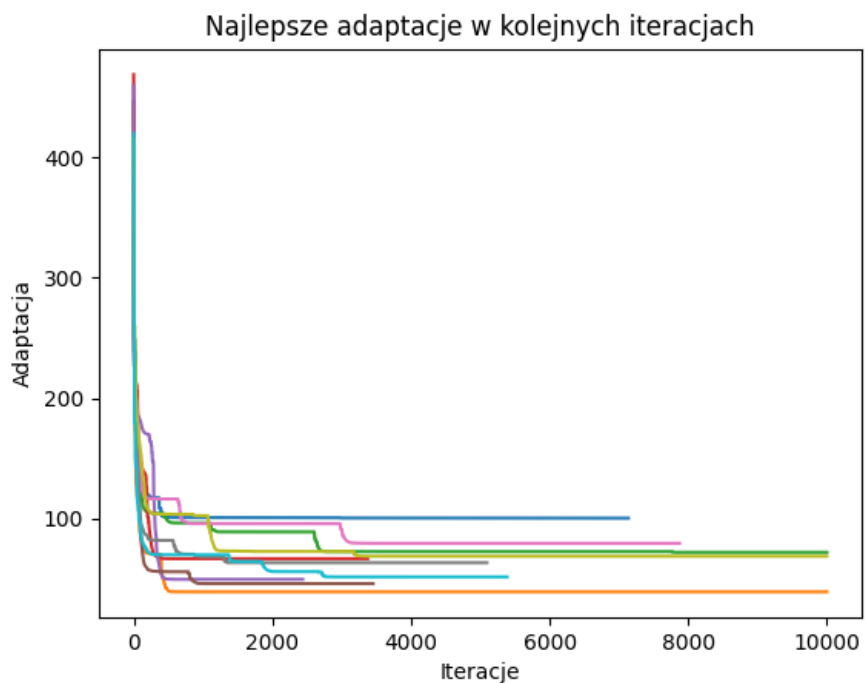
Wyniki przedstawimy w postaci tabeli oraz wykresów.

Tabela 4. Wyniki eksperymentów.

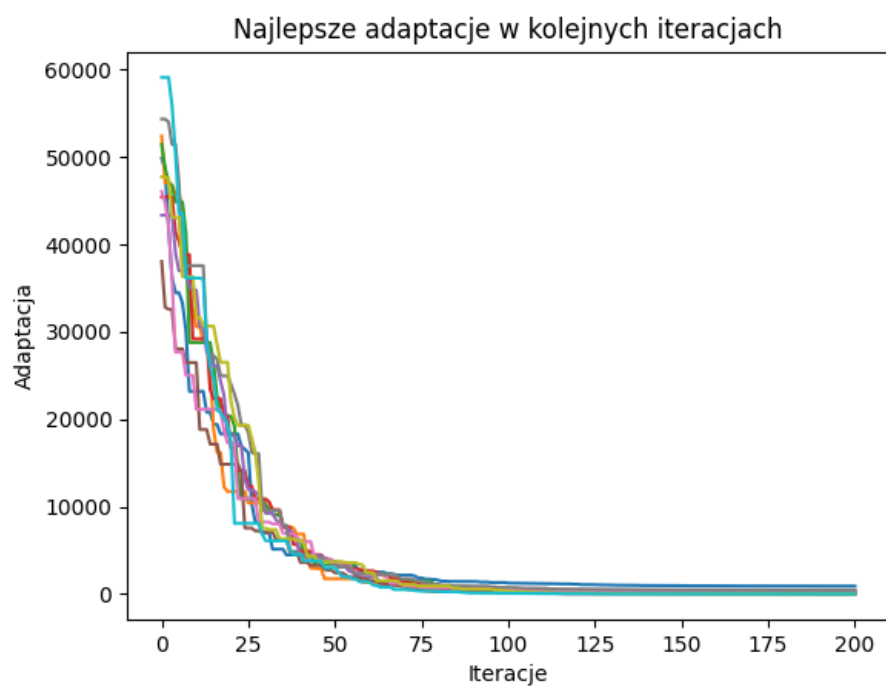
ID2	Średnia liczba iteracji	Średnie rozwiązanie	Odchylenie standardowe
E1	200	3354.69	1151.35
E2	6476	63.58	18.14
E3	200	239.97	281.95
E4	677	28.81	1.56



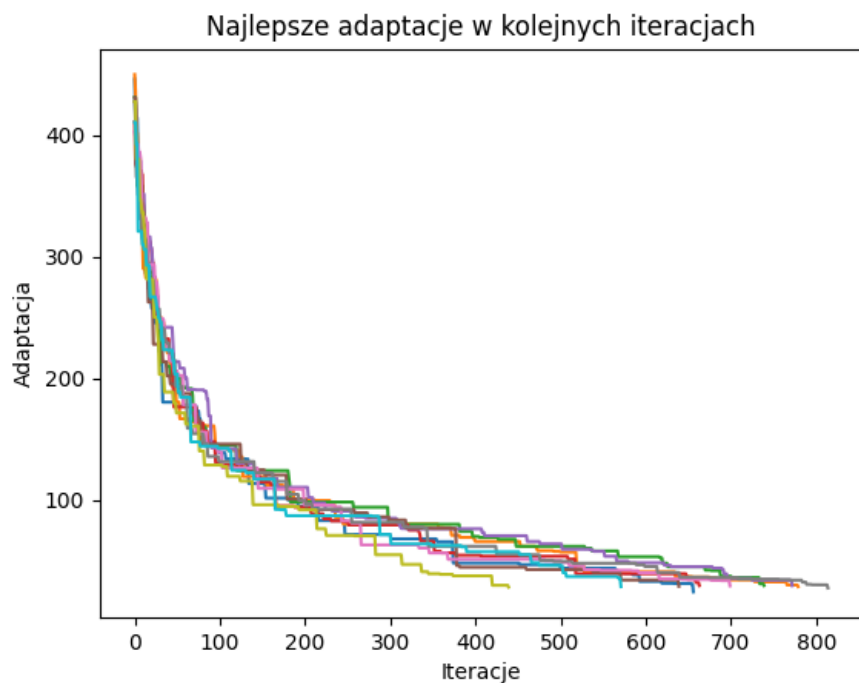
1. Wykres wartości najlepszych adaptacji w kolejnych iteracjach dla eksperymentu E1.



2. Wykres wartości najlepszych adaptacji w kolejnych iteracjach dla eksperymentu E2.



3. Wykres wartości najlepszych adaptacji w kolejnych iteracjach dla eksperymentu E3.



4. Wykres wartości najlepszych adaptacji w kolejnych iteracjach dla eksperymentu E4.

5. Analiza

Porównując parami eksperymenty E1 i E3 oraz E2 i E4 widzimy, że zgodnie z tabelą nr 4 możemy zauważyć lepsze wyniki dla algorytmu DE. Wartości średniego znalezionej rozwiązania oraz odchylenia standardowego są znacząco mniejsze. Do tego algorytm PS w eksperymencie E2 nie doszedł do pożądanej dokładności - skończył wcześniej przez osiągnięcie stagnacji lub maksymalnej liczby iteracji.

Na wykresach zainteresowała nas różnica w działaniu algorytmów PS i DE. Pierwszy szybciej znajduje mniejsze wartości, ale potem ma trudności w znalezieniu lepszych kandydatów. Drugi za to dąży do minima bardziej krokowo.

Podczas eksperymentów zmierzaliśmy się też z trudnością w algorytmie PS. Ma on znacznie większą przestrzeń wartości parametrów, którą należy przeszukać, żeby znaleźć dobrą kombinację argumentów w przeciwieństwie do algorytmu DE.

6. Wnioski

Po wykonaniu implementacji możemy stwierdzić, że algorytm DE jest skuteczniejszy w zastosowanych funkcjach jeżeli chcemy otrzymać dokładne wyniki. Dobór wartości parametrów jest również w tym algorytmie łatwiejszy w przeciwieństwie do algorytmu PS.