Sprawozdanie z Projektu 5

Oliwia Wiatrowska, Karolina Starzec, Jakub Wawrzyczek, Jakub Świerczyński

Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z problematyką optymalizacji wielokryterialnej i wyznaczenie rozwiązań minimalnych w sensie Pareto.

Wykonanie

Kod funkcji

Metoda Powella:

```
import numpy as np

def powell_method(func, x0, epsilon, Nmax, ud1, ud2):
    try:
        n = len(x0)
        d = np.eye(n)
        X = np.array(x0)

        for _ in range(Nmax):
            p = np.copy(X)

            for i in range(n):
                a = np.column_stack((p, d[i]))
                section = expansion(func, 0, 1, 1.2, Nmax, ud1, a)
                h = golden(func, section[0], section[1], epsilon, Nmax, ud1, a)
                p += h * d[i]

            if np.linalg.norm(X - p) < epsilon:
                return p

            d[:, :-1] = d[:, 1:]
            d[:, -1] = p - X

            a = np.column_stack((p, d[:, -1]))
            section = expansion(func, 0, 1, 1.2, Nmax, ud1, a)
            h = golden(func, section[0], section[1], epsilon, Nmax, ud1, a)
            X = p + h * d[:, -1]

        raise ValueError("Przekroczono limit wywołań funkcji Powell")
    except Exception as e:
        print(f"Error in Powell method: {e}")
```

Testowa funkcja celu

Funkcja: $$f_1(x_1,x_2) = a((x_1-2)^2 + (x_2-2)^2) \qquad f_2(x_1,x_2) = {}^1/_a\,((x_1+2)^2 + (x_2+2)^2)$$

```
def test_function(x, ud1):
    a = ud1[0]
    f1 = a * ((x[0] - 2)**2 + (x[1] - 2)**2)
    f2 = (1 / a) * ((x[0] + 2)**2 + (x[1] + 2)**2)
    return np.array([f1, f2])
```

Optymalizacja:

```
results = []
for w in np.linspace(0, 1, 101):
    x0 = np.random.uniform(-10, 10, 2)
    for a in [1, 10, 100]:
        ud1 = [a]
        result = powell_method(test_function, x0, epsilon=1e-6, Nmax=100, ud1=ud1, ud2=None)
        results.append(result)
```

Problem rzeczywisty

```
def real_function(x):
    rho = 7800
    P = 1000
    E = 207e9

    l, d = x
    y1 = rho * l * np.pi * (d**2 / 4)
    y2 = (64 * P * l**3) / (3 * E * np.pi * d**4)
    y3 = (32 * P * l) / (np.pi * d**3)

    return np.array([y1, y2, y3])
```

Kod optymalizacyjny:

```
for w in np.linspace(0, 1, 101):
    x0 = [np.random.uniform(200, 1000) / 1000, np.random.uniform(10, 50) / 1000]
    result = powell_method(real_function, x0, epsilon=1e-6, Nmax=100, ud1=[w], ud2=None)
    print(result)
```

Omówienie wyników

Wyniki przeprowadzonej optymalizacji pokazują wyraźne zależności między analizowanymi parametrami a funkcją celu. Na podstawie uzyskanych danych można zauważyć, że długość i średnica belki znacząco wpływają na wartości masy, ugięcia i naprężeń maksymalnych. Przede wszystkim:

1. **Zależność masy od parametrów geometrycznych:** Masa belki, obliczana na podstawie długości i średnicy, pokazuje liniowy wzrost wraz ze zwiększaniem tych parametrów. Jest to zgodne z teoretycznymi oczekiwaniami i potwierdza poprawność implementacji funkcji celu.
2. **Ugięcie belki:** Wzrost długości belki prowadzi do znaczącego wzrostu ugięcia, podczas gdy zwiększenie średnicy zmniejsza ten efekt. Wynika to z właściwości geometrycznych i materiałowych analizowanego układu, co znajduje odzwierciedlenie w wynikach.
3. **Naprężenia maksymalne:** Naprężenia osiągają większe wartości dla mniejszych średnic, co pokazuje, że optymalizacja zmierza do znalezienia kompromisu między wytrzymałością a ograniczeniami geometrycznymi i materiałowymi.

Wyniki optymalizacji wielokryterialnej dla różnych wartości wag wskazują na możliwość manipulowania priorytetami funkcji celu. Na przykład, dla wysokich wag przy ugięciu belki, algorytm zmierza do minimalizacji tego parametru kosztem masy. Z kolei dla wag skoncentrowanych na masie optymalizacja prowadzi do zwiększenia masy przy jednoczesnym ograniczeniu ugięcia.

Wizualizacja wyników w postaci wykresów Pareto pokazała, że rozwiązania optymalne w sensie Pareto układają się wzdłuż przewidywanej krzywej zależności między masą a ugięciem. Większa masa prowadzi do mniejszych wartości ugięcia, co jest intuicyjne i spójne z teorią mechaniki materiałów.

Histogram liczby wywołań funkcji w trakcie optymalizacji wskazuje na zmienność trudności problemów optymalizacyjnych w zależności od początkowych parametrów i wartości wag. Algorytm Powella dobrze radzi sobie z większością przypadków, chociaż w niektórych sytuacjach wymaga większej liczby iteracji, aby osiągnąć zbieżność.

Wnioski

Metoda Powella, pomimo swojej pozornej prostoty, ukazuje olbrzymi potencjał w zastosowaniach optymalizacyjnych, szczególnie w kontekście problemów wielokryterialnych. Analiza wyników dowodzi, że potrafi ona efektywnie identyfikować rozwiązania w sensie Pareto, co czyni ją narzędziem nie tylko efektywnym, ale i wszechstronnym. Zdolność tej metody do adaptacji do różnych funkcji celu oraz jej skuteczność w znajdowaniu rozwiązań minimalnych podkreślają jej znaczenie w obszarach takich jak inżynieria czy analiza systemów.

Analiza danych wynikowych wskazuje na istotne zależności między rozważanymi parametrami. Wartości długości i średnicy belki pokazują stabilność wyników optymalizacji, co odzwierciedla skuteczność algorytmu w eksploracji przestrzeni rozwiązań. Masa belki, zależna od jej parametrów geometrycznych, pokazuje znaczącą odwrotną korelację z ugięciem, co jest zgodne z teorią mechaniki materiałów. Zwiększenie masy prowadzi do zmniejszenia ugięcia, co potwierdza, że metoda Powella skutecznie odnajduje rozwiązania zgodne z intuicją inżynierską.

Zmienność liczby wywołań funkcji w procesie optymalizacji wskazuje na złożoność przestrzeni poszukiwań i wpływ parametrów początkowych na trudność zbieżności. Jednakże większość wyników koncentruje się w pobliżu średnich wartości, co świadczy o spójności i niezawodności metody.

To, co wyróżnia metodę Powella, to jej transformacyjny potencjał. Dzięki niej możliwe jest nie tylko zrozumienie istniejących zależności w badanych systemach, ale również projektowanie i rozwijanie nowych technologii o niespotykanej dotąd wydajności i precyzji. Wyniki tego projektu stanowią mocny argument za dalszym rozwijaniem tej metody, która z powodzeniem może znaleźć zastosowanie w projektowaniu przyszłościowych rozwiązań technologicznych, takich jak nanomateriały, inteligentne systemy czy innowacyjne konstrukcje inżynieryjne. Metoda ta, z odpowiednią modyfikacją, może stać się narzędziem przełomowym w optymalizacji najbardziej złożonych problemów współczesnego świata.