

Akademia WIT w Warszawie



Metaheurystyki

Projekt MEH

Temat nr 17

Zagadnienie plecakowe

Autor: Adrian Gwiazdowski

Nr albumu: 22219

Warszawa 2025

1. Cel ćwiczenia

Celem projektu było opracowanie rozwiązania dla klasycznego problemu plecakowego. W problemie tym, dla danego zbioru n przedmiotów, z których każdy charakteryzuje się określoną wagą oraz wartością, należy wybrać taki podzbiór tych przedmiotów, aby spełnione zostały dwa kluczowe warunki:

1. Suma wag wybranych przedmiotów nie może przekroczyć określonego limitu wagowego plecaka.
2. Suma wartości wybranych przedmiotów powinna być możliwie jak największa, czyli maksymalizujemy całkowitą wartość ładunku.

Matematyczny opis:

Dany jest zbiór n przedmiotów, gdzie każdy przedmiot j ma:

- wagę w_j
- wartość c_j

oraz plecak o maksymalnej pojemności W .

Celem jest wybranie podzbioru przedmiotów, gdzie

$$x_j = \{1 - \text{jeśli przedmiot jest wybrany}, 0 - \text{jeśli przedmiot nie został wybrany}\}$$

tak, aby:

$$\sum_{j=1}^N c_j x_j$$

przy ograniczeniu:

$$\sum_{j=1}^N w_j x_j \leq W$$

2. Zastosowany algorytm

Do rozwiązania problemu zastosowano algorytm ewolucyjny. Algorytm ewolucyjny to metaheurystyka inspirowana procesem naturalnej ewolucji i doboru naturalnego, stosowana do rozwiązywania problemów optymalizacyjnych. W kontekście problemu plecakowego operuje na populacji potencjalnych rozwiązań i iteracyjnie udoskonala je przez kolejne pokolenia, działa według następujących kroków:

- **Generowanie populacji początkowej:** n losowych przedmiotów z różnymi wagami i wartościami w formie binarnej (0 – nie wybrano, 1 – wybrano).
- **Funkcja celu:** Oblicza sumę wartości wybranych przedmiotów. Jeżeli suma wag przekracza dopuszczalną wartość, dopasowanie jest zerowane.

- **Selekcja:** Spośród n losowych osobników wybierany jest ten z najwyższym dopasowaniem.
- **Krzyżowanie:** Dziecko powstaje z fragmentów dwóch rodziców, rozdzielonych w losowym punkcie.
- **Mutacja:** Każdy gen ma pewne prawdopodobieństwo odwrócenia.
- **Ewolucja populacji:** Stała liczba pokoleń. Algorytm wykonuje n iteracji. W każdym pokoleniu tworzone są nowe osobniki przez selekcję, krzyżowanie i mutację. Zachowywane jest najlepsze dotąd znalezione rozwiązanie.

3. Parametry

Dla realizacji eksperymentu przyjęto następujące założenia:

- **Liczba przedmiotów:** 100
- **Rozmiar populacji:** 150
- **Wagi i wartości:** generowane losowo w zakresie:
 - waga [1, 30]
 - wartość [10, 100]
- **Pojemność plecaka:** 500
- **Liczba generacji:** 500
- **Krzyżowanie:** Jednopunktowe
- **Prawdopodobieństwo mutacji:** 0.0125
- **Rozmiar turnieju:** 5

4. Przykładowe dane wyjściowe

Po zakończeniu działania algorytmu wypisywane są:

- Całkowita wartość najlepszego rozwiązania.
- Całkowita waga plecaka.
- Indeksy wybranych przedmiotów.

Dodatkowo generowany jest wykres przedstawiający postęp najlepszej sumy wartości w czasie, co pozwala ocenić efektywność algorytmu.

Generacja 0 - Najlepsza wartość: 0

Generacja 50 - Najlepsza wartość: 1962

Generacja 100 - Najlepsza wartość: 2958

Generacja 150 - Najlepsza wartość: 3244

Generacja 200 - Najlepsza wartość: 3244

Generacja 250 - Najlepsza wartość: 3244

Generacja 300 - Najlepsza wartość: 3244

Generacja 350 - Najlepsza wartość: 3244

Generacja 400 - Najlepsza wartość: 3244

Generacja 450 - Najlepsza wartość: 3244

Generacja 500 - Najlepsza wartość: 3244

Najlepsze rozwiązanie:

Całkowita wartość: 3244

Całkowita waga: 500

Wybrane przedmioty: [2, 4, 8, 10, 12, 15, 16, 17, 18, 19, 22, 25, 26, 29, 31, 32, 35, 37, 42, 45, 49, 50, 51, 53, 54, 55, 56, 57, 61, 64, 66, 67, 68, 73, 75, 76, 77, 78, 79, 80, 81, 84, 87, 88, 90, 93, 97, 98, 99, 100]

Wynik końcowy pokazuje rozwiązanie bardzo bliskie optimum – plecak został zapełniony do 500 jednostek wagi (z maks 500), osiągając wysoką wartość całkowitą (3244). Lista wybranych przedmiotów sugeruje, że algorytm znalazł dobrze zbilansowany zbiór o wysokiej wartości przy zachowaniu dopuszczalnej wagi.

5. Analiza doboru parametrów algorytmu ewolucyjnego

W celu zwiększenia sprawności algorytmu ewolucyjnego w rozwiązywaniu problemu plecakowego, przeanalizowano wpływ następujących parametrów:

1. Prawdopodobieństwo mutacji

Testowana wartość	Wynik rozwiązania	Czas działania
0.001 (0.1%)	0 - program utknął	0 - program utknął
0.005 (0.5%)	3129	6.48
0.01 (1%)	3236	6.39
0.0125 (1.25%)	3242	6.56
0.02 (1.5%)	3238	6.36
0.02 (2%)	3221	6.42
0.1 (10%)	2701	6.03

Wnioski:

- Zbyt niska mutacja powoduje utknięcie programu.
- Zbyt wysoka mutacja niszczy strukturę dobrych osobników, co destabilizuje populację.
- Optymalna wartość: 1%–2% (dla obecnego kodu wartość 0.0125 działa najlepiej).

2. Rozmiar turnieju

Testowana wartość	Wynik rozwiązania	Czas działania
1	2219	3.20

2	3136	4.10
3	3229	4.65
5	3242	6.51
7	3223	8.26
10	3236	10.77
20	3236	17.97

Wnioski:

- Mniejszy rozmiar turnieju (1–3) obejmuje szerszy zakres rozwiązań, ale może wybierać słabych osobników.
- Duży turniej (≥ 10) zwiększa ryzyko utraty różnorodności, co może prowadzić do stagnacji.
- Optymalna wartość: 4–6 – pozwala zachować balans.

3. Rozmiar populacji

Testowana wartość	Wynik rozwiązania	Czas działania
50	3163	2.15
100 - domyślna	3193	4.51- domyślna
150	3244	6.46
200	3235	9.00
300	3233	13.14
500	3241	20.92

Wnioski:

- Zbyt mała populacja (np. 50) prowadzi do szybkiego czasu działania programu, ale o niższej jakości.
- Większe populacje (np. 200) zwiększają różnorodność genetyczną, co umożliwia znalezienie lepszego rozwiązania, ale kosztem czasu działania.
- Optymalna wartość: 100–150 dla problemu 100-przedmiotowego.

6. Podsumowanie

Algorytm ewolucyjny dobrze radzi sobie z rozwiązywaniem problemu plecakowego, oferuje wysoką jakość rozwiązań przy zadowalającym czasie działania. Jego skuteczność w dużej mierze zależy od odpowiedniego doboru parametrów. Poprawnie skonfigurowany, stanowi wydajne narzędzie do rozwiązywania złożonych problemów optymalizacyjnych, szczególnie tam, gdzie tradycyjne metody zawodzą. Dodatkowo, łatwość modyfikacji i adaptacji algorytmu czyni go uniwersalnym i praktycznym podejściem w wielu dziedzinach nauki.