

Sprawozdanie z ćwiczenia laboratoryjnego  
„Problem plecakowy”

Karolina Morawska

25 maja 2014

## Spis treści

1	Przybliżenie pojęcia problemu plecakowego	3
1.1	Opis implementacji . . . . .	3
2	Wnioski	4
	Bibliografia	5

## 1 Przybliżenie pojęcia problemu plecakowego

Problem plecakowy (ang. knapsack problem) jest przykładem zagadnienia optymalizacyjnego, które może być rozwiązywane za pomocą wielu technik algorytmicznych (oczywiście z różną efektywnością). Niemniej jednak jest to doskonałe „modelowe” zadanie, dzięki któremu jesteśmy w stanie w przybliżony sposób zaprezentować sposoby działania kilku grup algorytmów. O trudności tego problemu niech świadczy fakt, że nie istnieje (przynajmniej nie został jeszcze odkryty) algorytm rozwiązujący dyskretny problem plecakowy, którego pesymistyczna złożoność obliczeniowa byłaby lepsza niż wykładnicza  $O(2^n)$ . Nie powinno nas to dziwić, gdyż problem plecakowy jest głównym przykładem algorytmu tzw. NP-trudnego.

Główna zasada tego problemu brzmi: mamy pewien zbiór mniej lub bardziej cennych przedmiotów scharakteryzowanych za pomocą dwóch parametrów:

- Wartość (cena) przedmiotu  $i$ -tego  $c$
- Masa przedmiotu  $i$ -tego  $w$

Oprócz tych obiektów mamy pewien zbiorczy obiekt, zwyczajowo nazywany plecakiem, który z kolei posiada jeden parametr – całkowitą pojemność  $W$  (a ściślej rzecz biorąc powinna być to wytrzymałość na obciążenie). Zadanie jakie stoi przed osobą, która dokonuje załadunku przedmiotów do plecaka (może to być złodziej w muzeum sztuki albo turysta wyruszający na długą wyprawę), jest następujące:

**Należy dokonać takiego wyboru przedmiotów do plecaka, by ich sumaryczna wartość była możliwie maksymalna, zachowując przy tym ograniczenia dotyczące pojemności ( $W$ ) plecaka.**

### 1.1 Opis implementacji

Występuje kilka sposobów implementacji tego problemu:

- Algorytm aproksymacyjny W wersji zachłannej algorytm aproksymacyjny sortuje elementy w kolejności malejącej porównując stosunek wartości do wagi elementu  $h_j = \frac{c_j}{w_j}$ . Następnie wstawia je kolejno zaczynając od przedmiotu o największym ilorazie do plecaka. Jeśli jakiś element nie mieści się w plecaku to jest omijany, a algorytm przechodzi do następnego. W algorytmie wybierany jest maksymalny wynik z tak obliczonego upakowania plecaka oraz plecaka z elementem o największej wartości. Jeśli  $k$  jest maksymalną wartością przedmiotów w optymalnie upakowanym plecaku, algorytm zachłanny osiąga wyniki nie gorsze niż  $k/2$ . Złożoność obliczeniowa algorytmu zależy od sortowania ( $\Theta(n \cdot \log n)$ ). Jest to metoda która nie zawsze zwraca najlepszy sposób spakowanie plecaka.

- Przy użyciu programowania dynamicznego -wybrany przeze mnie. Problem plecakowy może być rozwiązany w czasie pseudowielomianowym przy użyciu programowania dynamicznego. Rozwiązanie niżej dotyczy przypadku w którym można użyć wielokrotnie każdego elementu:

Niech  $w_1, \dots, w_n$  będzie wagą elementów oraz  $c_1, \dots, c_n$  wartościami. Algorytm ma zmaksymalizować sumę wartości elementów, przy zachowaniu sumy ich wagi mniejszej bądź równej  $W$ . Niech  $A(i)$  będzie największą możliwą wartością, która może być otrzymana przy założeniu wagi mniejszej bądź równej  $i$ .  $A(W)$  jest rozwiązaniem problemu.

Rozwiązanie dla pustego plecaka jest równe zero. Obliczenie wyników kolejno dla  $A(0)$ ,  $A(1)$ ... aż do  $A(W)$  pozwala obliczyć wynik. Ponieważ obliczenie  $A(i)$  wymaga sprawdzenia  $n$  elementów, a wartości  $A(i)$  do obliczenia jest  $W$ , złożoność obliczeniowa programu wynosi  $\Theta(nW)$ .

Powyższa złożoność nie neguje faktu, że problem plecakowy jest NP zupełny, ponieważ  $W$ , w przeciwieństwie do  $n$ , nie jest proporcjonalne do rozmiaru danych wejściowych dla problemu. Rozmiar wejścia jest proporcjonalny do ilości bitów w liczbie  $W$ , nie do wartości  $W$ .

## 2 Wnioski

Różnice i wyjaśnienie dlaczego wybrałam właśnie sposób implemetacji programowaniem dynamicznym chciałabym przedstawić na przykładzie, by lepiej zrozumieć problem plecakowy. Dyskretny problem plecakowy:

Złodziej rabujący sklep znalazł  $n$  przedmiotów;  $i$ -ty przedmiot ma wartość  $c_i$  złotych i waży  $w_i$  kilogramów, gdzie  $c_i$  i  $w_i$  są nieujemnymi liczbami całkowitymi. Dąży on do zabrania jak najwartościowszego łupu, ale do swojego plecaka nie może wziąć więcej niż  $W$  kilogramów. Złodziej nie może dzielić przedmiotów (zabrać do plecaka tylko część wybranego przedmiotu) ani wielokrotność przedmiotu. Interesuje nas odpowiedź na pytanie: Jakie przedmioty z puli  $n$  wybranych przedmiotów może zabrać złodziej, przy wymienionych wyżej ograniczeniach. Dyskretny problem plecakowy nie może być rozwiązany metodą zachłanną.

Czy dyskretny problem plecakowy ma własność wyboru zachłannego?

Odpowiedź : NIE

Kontrprzykład

Dane:

$n=3$ . Zbiór wybranych przedmiotów składa się z następujących elementów:

- Przedmiot 1 waga 10 kg i wartość 60 zł
- Przedmiot 2 waga 20 kg i wartość 100 zł
- Przedmiot 3 waga 30 kg i wartość 120 zł

Plecak ma maksymalną pojemność 50 kg. Kryterium wyboru przy zastosowaniu metody zachłannej jest cena jednostkowa, czyli cena 1 kg przedmiotu. Według tego kryterium najwyższą cenę jednostkową ma :

- Przedmiot 1(6 zł/kg),
- Przedmiot 2(5 zł/kg),
- Przedmiot 3(4 zł/kg).

i to właśnie Przedmiot 1 zostałby wybrany jako pierwszy. Następny wybrany byłby Przedmiot 2. Do plecaka nie trafiłby Przedmiot 3, ponieważ wszystkie trzy przedmioty mają za dużą wagę łączną (60 kg). Rozwiązanie polegające na wyborze Przedmiotu 1 i Przedmiotu 2 nie jest optymalne. Rozwiązaniem optymalnym jest natomiast wybór Przedmiotu 2 i Przedmiotu 3 (łączna waga wynosi 50 kg, łączna wartość 220 zł).

Czy dyskretny problem plecakowy ma własność optymalnej podstruktury?

Odpowiedź : TAK.

Dyskretny problem plecakowy wykazuje cechę optymalnej podstruktury. Rozważmy najwartościowszy ładunek plecaka o masie nie większej niż  $W$ . Jeżeli usuniemy z tego ładunku przedmiot  $j$  o wadze  $w_j$ , to pozostający ładunek jest najwartościowszym zbiorem przedmiotów o wadze nie przekraczającej  $W - w_j$ , jakie złodziej może wybrać z  $n - 1$  oryginalnych przedmiotów z wyjątkiem  $j$ .

Okazuje się, że dyskretny problem plecakowy można rozwiązać stosując technikę programowania dynamicznego.

## Bibliografia

- [1] [http : //pl.wikipedia.org/wiki/Problem\\_plecakowy](http://pl.wikipedia.org/wiki/Problem_plecakowy),
- [2] [http : //www.cs.put.poznan.pl/arybarczyk/ProgrDynamiczne.pdf](http://www.cs.put.poznan.pl/arybarczyk/ProgrDynamiczne.pdf),
- [3] [http : //iair.mchtr.pw.edu.pl/bputz/aisd\\_cpp/lekcja2/segment5/main.htm](http://iair.mchtr.pw.edu.pl/bputz/aisd_cpp/lekcja2/segment5/main.htm)