

AiSD — zadanie piąte

Gliwiński Jarosław Marek
Kruczyński Konrad Marek
Grupa dziekańska I5

5 czerwca 2008

1 Wstęp

W zadaniu piątym zajmowaliśmy się jednym z klasycznych problemów informatyki, tj. dyskretnym problemem plecakowym. Problem ten należy do klasy problemów NP-zupełnych. W rozważanej postaci danymi wejściowymi dla zadania był zestaw parametrów opisujących grupę astronautów (dla każdego podano jego wagę i iloraz inteligencji – IQ) oraz masę, jaką może przewieźć statek kosmiczny. Zadaniem programu jest wybranie optymalnej podgrupy astronautów, tj. takich, którzy mogą zostać zabrani przez statek i mają największy spośród wszystkich podgrup sumaryczny IQ.

Najprostszym (tzw. brute force) rozwiązaniem tego problemu, które przychodzi od razu na myśl, jest obliczenie sumarycznego IQ dla wszystkich możliwych wyborów podgrupy i wybranie zeń największego, pomijając oczywiście takie zestawy astronautów, które nie mieszczą się na statku.

W przedstawianym programie zaimplementowano wyżej wymienioną metodę oraz znacznie efektywniejszą – tzw. programowanie dynamiczne. Wykorzystujemy następującą właściwość: optymalny wybór astronautów dla statku o danej pojemności zawiera w sobie rozwiązania dla statków o pojemności mniejszej. Można zatem podać rekurencyjną zależność na najwyższe możliwe sumaryczne IQ. Wówczas, ponieważ pojemność statku zmienia się w sposób dyskretny, możemy rozpocząć wyliczanie równania rekurencyjnego od dołu, tj. od najniższych wartości i zachowywać je, co umożliwi wykorzystanie przy wyliczeniach dla wartości wyższych. Metoda ta jest znacznie efektywniejsza, o czym mowa będzie w dalszej części sprawozdania.

2 Pomiary

2.1 Generowanie danych wejściowych

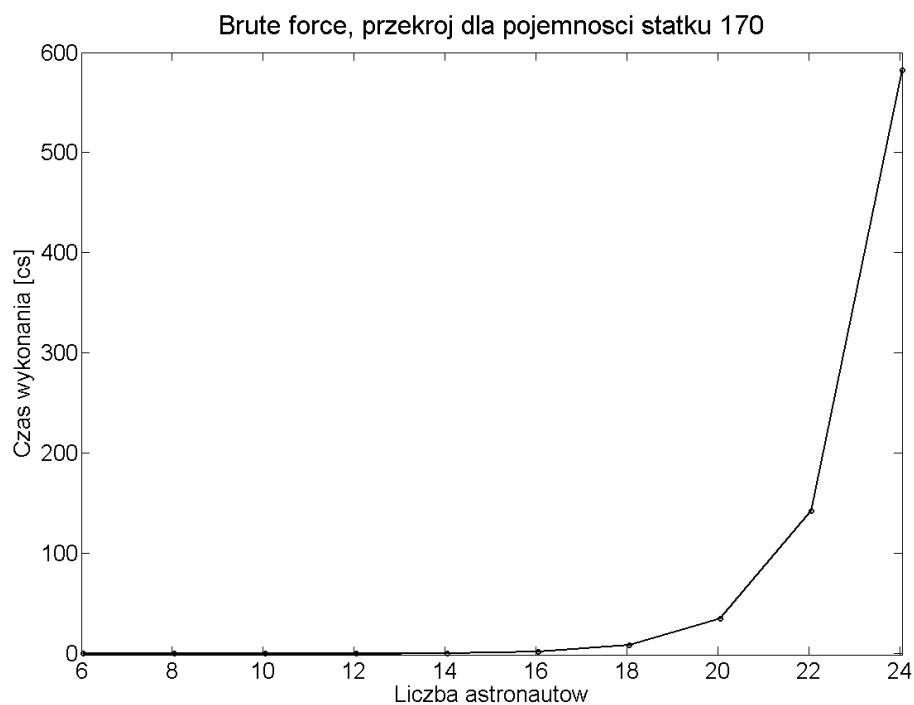
Parametrami wejściowymi dla procedury generującej są ilość astronautów oraz pojemność statku. Ta druga jest oczywiście bezpośrednio przepisywana, generowanie parametrów astronautów odbywa się w następujący sposób: zarówno ich waga, jak i IQ dobierane jest w sposób losowy. Rozkład zmiennych tych jest normalny o parametrach odpowiednio

$\mu = 75, \sigma = 18$ oraz $\mu = 100, \sigma = 16$. Należy jeszcze dodać, że oba rozkłady są ograniczone, tj. wartości odpowiednio spoza $[70, 200]$ oraz $[45, 150]$ są odrzucane.

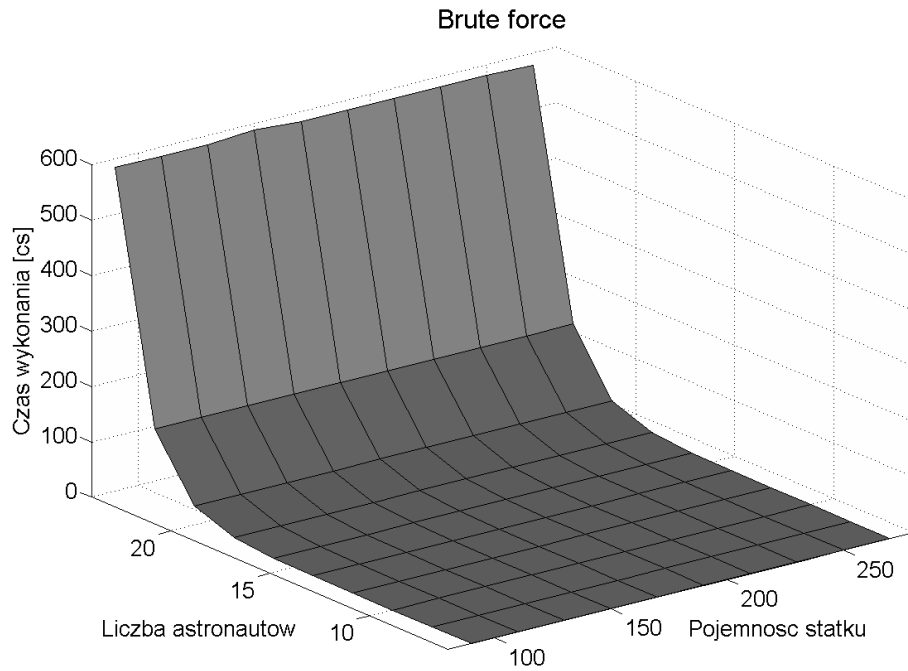
Tak wygenerowany zbiór danych zapisywany jest do pliku zgodnie z poleceniem zadania oraz oczywiście przekazywany dalej algorytmom PD i BF. Wyniki obu również zapisywane są w pliku, zgodnie z poleceniem.

2.2 Algorytm brute force

Poniższy wykres przedstawia czas wykonania zadania przez algorytm brute force w zależności od ilości danych (ilości astronautów).



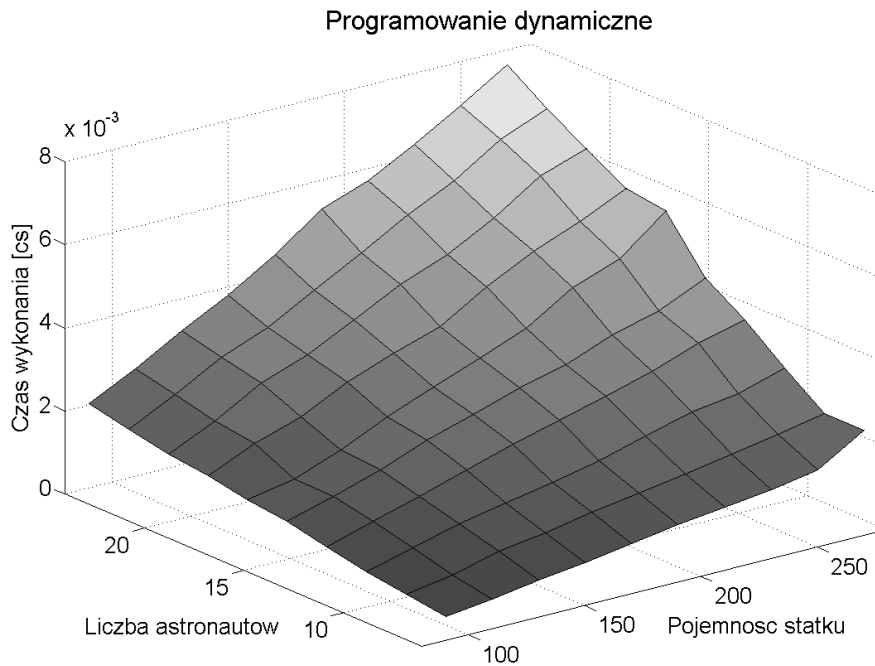
Dla porównania z wykresem, który zamieszczony będzie później dla algorytmu programowania dynamicznego poniżej znajduje się wykres czasu w funkcji dwóch zmiennych – poprzednio używanej ilości astronautów oraz wielkości statku.



Jak widać, czas ten nie zależy od pojemności statku, co jest zgodne z naturą algorytmu. Niezależnie od pojemności zawsze dokonujemy obliczeń dla wszystkich możliwych wyborów ze zbioru n astronautów. Wyborów tego typu mamy 2^n , zatem i złożoność wynosi $O(2^n)$, ma zatem charakter wykładniczy, co zgadza się z wykresem.

2.3 Algorytm programowania dynamicznego

Tym razem na początku umieszczony zostaje wykres trójwymiarowy, obrazujący zależność od dwóch zmiennych.



Jak widać, zależność ta jest liniowa z uwagi na każdą zmienną. Warto zauważyć, że zgadza się to z przypuszczeniami opartymi o analizę algorytmu. Ponieważ wykonujemy dokonujemy wyliczenia dla każdej wartości pojemności statku od 1 do k (gdzie k oznacza zadaną pojemność), potrzeba na to k iteracji. Z kolei w każdej z nich badamy, przy dołożeniu którego astronauty maksymalizujemy sumaryczny IQ, a zatem wymaga to n iteracji. Wobec tego ostateczna złożoność powinna być postaci $O(nk)$ i z taką właśnie mamy do czynienia na wykresie.

Odchylenia wyników liczone po całości pomiarów prezentują się następująco:

Dane	Odchylenie względne [%]
Brute Force	$2,4 \pm 0,4$
Programowanie dynamiczne	$6,19 \pm 1,12$

Dodatkowo poniżej znajdują się inne ilustracje przekrojowe obrazujące wydajność algorytmów w reprezentacji na płaszczyźnie - wykresy te są oczywiście w pełni zgodne z zamieszczonymi wcześniej trójwymiarowymi, jako że są skonstruowane na podstawie tych samych zbiorów danych.

