

## **PROBLEM SUMY PODZBIORU**

Osoby realizujące:

Rafał Malin,  
Albert Podraza,  
Nikodem Tokarski.

# TEORIA

W problemie sumy podzbioru jest NP-zupełnym problemem arytmetycznym. Mamy dany zbiór skończony  $S \subset \mathbb{N}$  i wartość docelową  $t \in \mathbb{N}$ . Pytamy, czy istnieje podzbiór  $S' \subset S$ , którego suma elementów wynosi  $t$ . Jeśli na przykład  $S = \{1, 10, 15, 87, 13, 156, 21\}$ , a  $t = 26$ , to podzbiór  $S' = \{1, 10, 15\}$  jest rozwiązaniem problemu. Należy pamiętać, że w kodowaniu standardowym, wejściowe liczby całkowite są kodowane jako ciągi binarne. Mając na uwadze to założenie, można pokazać, że szybki algorytm dla tego problemu najprawdopodobniej nie istnieje.

## OPIS PRZYJĘTEJ METODY

### 1. Opis osobnika

Biorąc pod uwagę fakt, że w algorytmach genetycznych bierze udział bardzo dużo osobników staraliśmy się by reprezentacja pojedynczego chromosomu była jak najprostsza. Pojedynczy osobnik reprezentuje tablica wypełniona pseudolosowo wartościami 0 lub 1.

### 2. Losowanie populacji

W tym miejscu ustalamy jak wielka będzie tworzona populacja. Jeśli populacja będzie zawierała zbyt mało osobników to algorytm może zatrzymać się w jakimś lokalnym minimum (gdy jakieś niezłe, lecz nie najlepsze rozwiązanie zdominuje całą populację). Z drugiej strony zbyt duża liczebność populacji zmniejsza szybkość działania algorytmu. By zapewnić jak największą różnorodność każdy osobnik tworzony jest losowo. W trakcie losowania populacji, liczby z podzbioru o rozmiarze równym zadanej długości chromosomu i będące na miejscu jedynki zostaną wybrane, a te będące na miejscu zera nie zostaną uwzględnione. Dla przykładu chromosom  $[1001110]$  reprezentuje liczby  $\{12, 98, 14, 54\}$  z podzbioru  $\{12, 87, 35, 98, 14, 54, 67\}$ .

### 3. Ocena osobników

W tym kroku badamy jak dobry jest poszczególny osobnik. Dobroć ocenia funkcja oceny dla każdego chromosomu. Im suma wszystkich liczb wybranych dla osobnika bliższa zera, tym wyższą ocenę (fitness value) otrzymuje osobnik.

### 4. Selekcja

Krok najważniejszy dla całej genetyki. Tworzymy nową populację na podstawie już istniejącej. W zależności od wartości funkcji oceny obliczanej w poprzednim kroku dany osobnik ma większe lub mniejsze szanse na dopuszczenie do

krzyżowania. Szansę poszczególnych osobników obliczamy na zasadzie koła ruletki, która polega na n-krotnym losowaniu osobników ze starej populacji, gdzie osobniki mają różne prawdopodobieństwa wylosowania. Prawdopodobieństwo obliczamy ze wzoru:

$$probability = 1 - \frac{fitness\_value}{sum\_of\_all\_fitness\_values}$$

## 5. Krzyżowanie

Zadaniem kroku krzyżowania jest wymiana "materiału genetycznego" pomiędzy dwoma rozwiązaniami w populacji. W wyniku krzyżowania na podstawie dwóch rozwiązań (rodzice) tworzone są dwa nowe osobniki (dzieci). Po wykonaniu krzyżowania dzieci zastępują w populacji rodziców. Mając binarną reprezentację nasze krzyżowanie polega na podziale dwóch chromosomów na dwie części i z nich tworzymy dzieci. Pierwsze dziecko składa się z początkowej części pierwszego rodzica i końcówki drugiego natomiast drugie dziecko odwrotnie - początek drugiego rodzica i koniec pierwszego. Dla przykładu mając chromosomu rodziców: [10011101] i [10111000] dzieci będą wyglądały następująco: [10011000] i [10111101].

## 6. Mutacja

Mutacja zapewnia dodawanie do populacji nowych osobników. W odróżnieniu od krzyżowania modyfikowany jest tylko jeden osobnik. Osobniki do mutacji wybierane są losowo. Mutacja polega na zanegowaniu pojedynczego bitu w chromosomie. Bit do zanegowania wybierany jest drogą losowania.

## 7. Powrót do oceny osobników(2) lub wyjście z pętli

Za warunek wyjścia z pętli przyjmujemy znalezienie podzbioru sumującego się do zera lub brak poprawy wyniku przez zadaną ilość iteracji w argumencie tSearchTolerance.