

# Problem Mobilnego Złodzieja

## podjęcie wieloobiektowe

### - Sprawozdanie 2

#### I. OPIS PROBLEMU

Założenie klasycznego problemu mobilnego złodzieja jest złożeniem dwóch problemów – komiwojażera i problemu plecakowego. Otrzymujemy na wstępie listę miast wraz z ich współrzędnymi, a także listę przedmiotów, wśród których każdy charakteryzuje się swoim id, profitem, wagą oraz miastem, w którym się znajduje. Naszym zadaniem jest napisać metaheurystykę, która ustali możliwie najlepszą drogę oraz sposób zapakowania plecaka. Zgodnie z zaproponowanym problemem w każdym mieście może znajdować się wiele przedmiotów, a każdy przedmiot może znajdować się w wielu miastach. Ze względu na otrzymane dane, w których każdy przedmiot ma przypisany unikalny numer id przyjmujemy, że jeden przedmiot występuje w każdym mieście tylko raz.

Podjęcie wieloobiektowe zakłada, że wraz ze wzrostem wagi plecaka tracimy prędkość a także im dłużej niesiemy przedmiot, tym mniejszą ma on wartość. Aspekt drugi (zmniejszanie wartości przedmiotu) zostaje pominięty, ze względu na brak niezbędnych danych.

Do rozwiązania wyżej opisanego problemu NP-trudnego użyto algorytmu NSGA-II.

#### II. REPREZENTACJA OSOBNIKA

Na każde rozwiązanie (osobnika) składają się dwa elementy. Jeden z nich, droga, przeszukiwany jest przy pomocy algorytmu genetycznego. Drugi natomiast konstruowany jest przy pomocy algorytmu zachłannego. Droga to tablica której kolejne elementy są indeksami miast odwiedzanych w kolejności. Plan pakowania plecaka jest tablicą, w której kolejne elementy odpowiadają indeksowi danego przedmiotu. Ponieważ każdy element występuje tylko w jednym mieście, nie trzeba w rozwiązaniu zawierać informacji o tym, z którego miasta pobieramy przedmiot. Używamy zatem zapisu zero-jedynkowego, określając jedynie czy przedmiot bierzemy, czy też nie.

#### III. OPIS ZASTOSOWANYCH HEURYSTYK

##### 1. Inicjalizacja

Początkowa populacja generowana jest w sposób losowy. Zgodnie z parametrem wielkości populacji tworzymy zadaną liczbę dróg i do każdej z nich przypisujemy plan pakowania plecaka przy użyciu algorytmu zachłannego.

##### 2. Greedy

Zaproponowany algorytm zachłanny przypisuje każdemu z elementów stosunek  $\frac{p}{w \cdot t}$ , gdzie  $p$  to profit za dany przedmiot,  $w$  to waga tego przedmiotu, a  $t$  to czas, jaki zajęłoby nam przeniesienie tego tylko przedmiotu (nie zabieramy nic innego), z miasta, w którym się on znajduje do miasta końcowego. Tak określone przedmioty sortujemy wg wyliczonego stosunku i, zaczynając od największego stosunku, wybieramy te, które się mieszczą do plecaka. Stosunek ten różni się więc dla każdej drogi, ale algorytm jest deterministyczny.

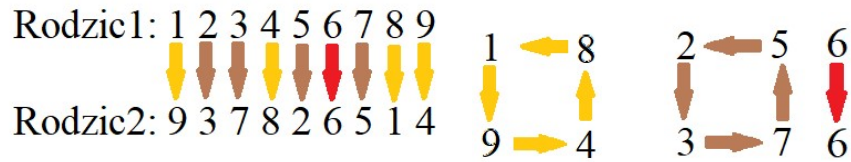
##### 3. Selekcja

Jako selekcja zaimplementowany jest turniej, który przyjmuje za parametr liczbę losowanych osobników. Kolejne osobniki losujemy i porównujemy z najlepszym dotychczas

wylosowanym. Do porównania używamy numeru frontu, w którym znajduje się dany osobnik. Osobniki z tego samego frontu porównujemy przy pomocy crowding distance.

#### 4. Krzyżowanie

Zaimplementowane krzyżowanie to tzw. cycle-crossover. Polega ono na szukaniu kolejnych cykli genów obojga rodziców w sposób następujący:



Następnie naprzemiennie albo przepisujemy liczby z danego cyklu w niezmienionej kolejności, albo podmieniamy każdą liczbę z cyklu na tę poprzedzającą ją.

Dziecko1: 1 5 2 4 7 6 3 8 9

Dziecko2: 9 2 3 8 5 6 7 1 4

W wyniku tego powstaje dwoje dzieci, gdzie wyszarzone pola, to liczby poddane cyklicznej przemianie.

Modyfikacją danej metody jest zastrzeżenie, że jeśli cykl jest jednoelementowy nie jest on uwzględniany w liczeniu naprzemiennym cykli.

Np. bez tej modyfikacji, dla poniższego cyklu dzieci byłyby niezmienione.



#### 5. Mutacja

Dla każdego kolejnego genu wykonywana jest z zadaniem prawdopodobieństwem zamiana z losowym elementem danego ciągu.

#### 6. Pareto front

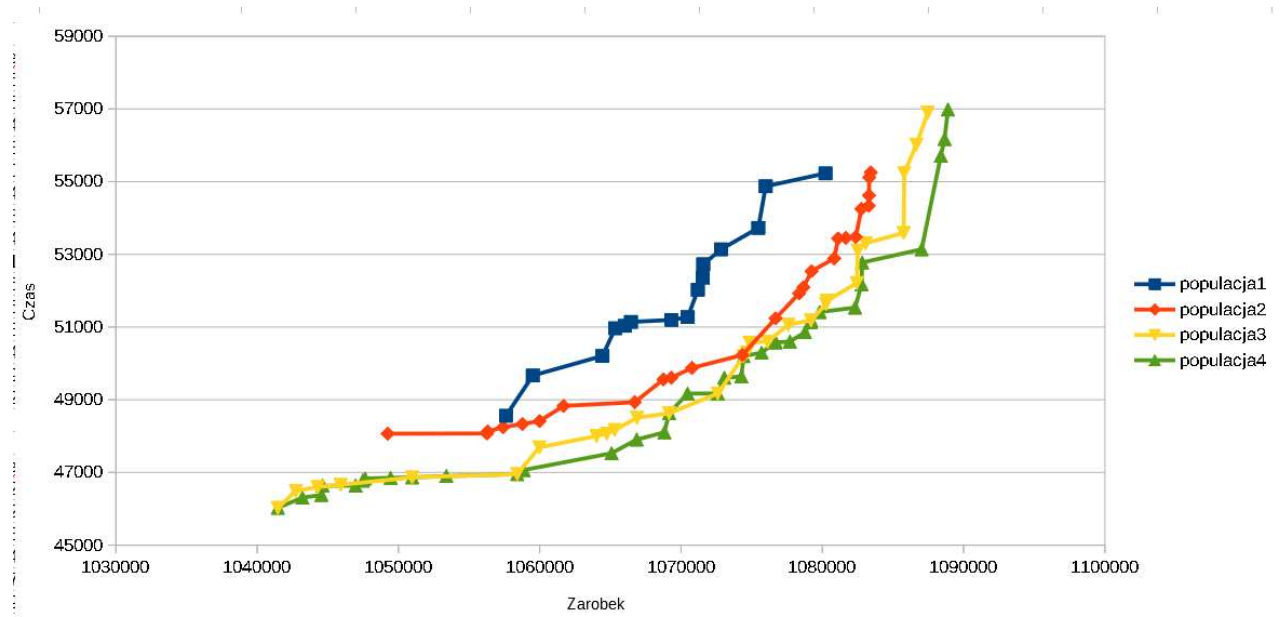
Pareto front działa na zasadzie analogicznej do insert – kolejne elementy z populacji przyrównujemy z kolejnymi elementami zaczynając od najlepszego frontu i w zależności od wyniku podejmując decyzje co do modyfikacji istniejących frontów. Ze względu na pewne specyficzne przypadki istnieje również pomocnicza metoda naprawcza.

#### 7. Crowding distance

Zgodnie z założeniami jest liczony jako pole prostokąta o wierzchołkach równych dwóm sąsiednim elementom. Im większe pole, tym bardziej promowany osobnik.

#### IV. WYNIKI

Poniższe wykresy prezentują dwa uruchomienia algorytmu dla problemu hard\_3



Można więc zauważyć, że kolejne fronty nie tylko mają rozwiązania bliższe frontowi pareto, ale również obejmują szerszy zakres, więc wyniki są bardziej różnorodne.