

SPD - Lab nr 6 - Sprawozdanie

Marcin Wiśniewski (226524) Piotr Kida (226237)

18 czerwca 2018

Spis treści

1	Założenia	2
2	Budowa algorytmu w pseudokodzie	3
3	Wnioski	4

1 Założenia

Celem ćwiczenia było utworzenie implementacji algorytmu poszukiwań z zabronieniami, tzw. Tabu Search oraz rozwiązanie za jego pomocą przykładowego problemu przepływowego.

Algorytm Tabu Search polega na przeszukiwaniu przestrzeni wszystkich możliwych rozwiązań z wykorzystaniem specyficznej sekwencji ruchów. W sekwencji tej zawierają się tzw. ruchy Tabu - czyli ruchy, których wykonanie jest absolutnie niedozwolone. Dodatkowo tworzona jest lista Tabu, określana w literaturze akronimem TL, w której przechowywane są dane dotyczące rozwiązań, które zostały już sprawdzone.

Dzięki zastosowaniu TL zapewniona zostaje pewność, że algorytm będzie unikał oscylowania wokół optimum lokalnego.

2 Budowa algorytmu w pseudokodzie

wygeneruj losowo rozwiązanie początkowe $x_0 \in X$

$x_{opt} \leftarrow x_0$

repeat

 znajdź w $N(x_0)$ takie x , dla którego $m_{val}(x_0, x)$ jest największa

$x_0 \leftarrow x$

 if $f(x_0) > f(x_{opt})$ then

$x_{opt} \leftarrow x_0$

until warunek zakończenia

3 Wnioski

Metaheurestyka Tabu Search jest metodą wszechstronną. Jej podstawowa wersja, której implementacja jest przedmiotem tej rozprawy, może zostać rozbudowana o wiele funkcjonalności, tj.: wyznaczanie kadencji czy zastosowanie kryterium aspiracji.

Kadencja określa liczbę iteracji, przez którą element znajduje się na liście tabu, taka zmiana wpływałaby bezpośrednio na definicję sąsiedztwa.

```
wybierz lub wylosuj punkt startowy  $x_0 \in X$ 
 $x_{opt} \leftarrow x_0$ 
tabu_list  $\leftarrow \emptyset$ 
repeat
    znajdź  $x \in N'(x_0)$ , dla którego  $m_{val}(x_0, x)$  jest największa
     $x_0 \leftarrow x$ 
    if  $f(x_0) > f(x_{opt})$  then
         $x_{opt} \leftarrow x_0$ 
    zweryfikuj tabu_list - dodaj nowe elementy
     $\forall element \in tabu\_list$  do
        -- kadencjai
        if kadencjai = 0 then
            usuń element(atrybuti, kadencjai) z tabu_list
until warunek zakończenia
```

Racjonalność zastosowania nowego rozwiązania, tj.: dodania tzw. kryterium aspiracji, wynika z pewnego podstawowego problemu występującego w podstawowej wersji implementacji algorytmu Tabu Search - zastosowanie TL może prowadzić do zakazywania niektórych bardzo satysfakcjonujących ruchów. Dzieje się tak ze względu na wynik wartości obliczonej funkcji celu. Dochodzimy w tym momencie do fundamentalnego wniosku: rygorystyczne trzymanie się zasad funkcjonowania metaheurestyki Tabu Search może prowadzić do pomijania rozwiązań optymalnych.

W celu zażegnania tego problemu możliwym jest utworzenie wyżej wymienionego kryterium aspiracji. Kryterium to mówi, że jeżeli w otoczeniu rozwiązania znaleziono rozwiązanie lepsze, znajdujące się na TL, to to właśnie rozwiązanie będzie wzięte jako rozwiązanie kolejne.

Wprowadzenie w/w kryterium spowoduje kolejną bezpośrednią zmianę definicji sąsiedztwa algorytmu.

```
wyberz lub wylosuj punkt startowy  $x_0 \in X$ 
 $x_{opt} \leftarrow x_0$ 
tabu_list  $\leftarrow \emptyset$ 
repeat
    znajdź  $x \in N''(x_0)$ , dla którego  $m_{val}(x_0, x)$  jest największa
     $x_0 \leftarrow x$ 
    if  $f(x_0) > f(x_{opt})$  then
         $x_{opt} \leftarrow x_0$ 
    zweryfikuj tabu_list
     $\forall element \in tabu\_list$  do
        -- kadencjai
        if kadencjai = 0 then
            usuń element(atrybuti, kadencjai) z tabu_list
until warunek zakończenia
```

Metaheurestyka Tabu Search jest potężnym narzędziem optymalizacyjnym. Dodatkowym czynnikiem jasno potwierdzającym powyższą tezę jest fakt, że metodę tę można łączyć z innymi heurystykami.