Metody obliczeniowe w nauce i technice

Sprawozdanie z Laboratorium 5 Symulowane wyżarzanie | Poniedziałek 12:50

Marek Fudaliński

Zadanie 1. Wygeneruj 2-wymiarowy zbiór punktów na płaszczyźnie euklidesowej o losowych współrzędnych. Korzystając z metody symulowanego wyżarzania wyznacz jak najlepszą trasę. Porównaj czas wykonania programu z algorytmem dokładnym.

Napisany program losuje na płaszczyźnie 2d 9 punktów.

Następnie mierzy czas znalezienia optymalnej ścieżki przy pomocy algorytmu wyżarzania.
Wypisuje czas oraz znaleziona drogę.

Następnie mierząc czas dla tego samego zbioru punktów szukamy w najkrótszej ścieżki w sposób klasyczny.
Wypisuje czas oraz znaleziona drogę.

Przykładowe wynik przedstawiają poniższe screeny.

```
PS C:\Users\fudal\Desktop\Informatyka agh\ROK III\Mownit\lab5\zad1> python main.py
#########skrocony######
[(id:7), (id:8), (id:2), (id:0), (id:5), (id:1), (id:6), (id:4), (id:3)]
2.949621884917195
czas 0.113760
#########calosciowy######
[(id:6), (id:4), (id:5), (id:1), (id:7), (id:2), (id:8), (id:0), (id:3)]
2.784749286036866
czas 2.855474
```

```
PS C:\Users\fudal\Desktop\Informatyka agh\ROK III\Mownit\lab5\zad1> python main.py
#########skrocony######
[(id:4), (id:7), (id:0), (id:3), (id:2), (id:1), (id:5), (id:8), (id:6)]
2.5240093559009487
czas 0.104626
#########calosciowy######
[(id:2), (id:7), (id:0), (id:5), (id:4), (id:1), (id:3), (id:8), (id:6)]
2.0493359181284982
czas 2.832494
```

```
PS C:\Users\fudal\Desktop\Informatyka agh\ROK III\Mownit\lab5\zad1> python main.py
#########skrocony######
[(id:3), (id:0), (id:1), (id:7), (id:2), (id:5), (id:4), (id:6), (id:8)]
1.9506036562344358
czas 0.109354
#########calosciowy######
[(id:7), (id:4), (id:0), (id:1), (id:6), (id:8), (id:5), (id:3), (id:2)]
1.935851317862756
czas 2.929295
```

```
PS C:\Users\fudal\Desktop\Informatyka agh\ROK III\Mownit\lab5\zad1> python main.py
#########skrocony######
[(id:4), (id:3), (id:5), (id:7), (id:2), (id:6), (id:0), (id:1), (id:8)]
1.7093908363282866
czas 0.109473
#########calosciowy######
[(id:4), (id:8), (id:1), (id:7), (id:2), (id:6), (id:0), (id:5), (id:3)]
1.6159894330872662
czas 2.858181
```

Zadanie 2. Symulacja własnej fizyki. Dana jest siatka punktów w 2D (dla ambitnych: 3D) - reprezentacja np. w postaci mapy bitowej. Każdy punkt może być wypełniony lub nie. Zaproponuj funkcję energii (np. w oparciu o grawitację dodatkowo sterowaną kolorami - pełna dowolność - mile widziane rozwiązania kreatywne), a następnie dokonaj jej minimalizacji z wykorzystaniem SA. Przedstaw wizualizacje.

Optymalizowana wartość:

Suma sił przyciągania ma być jak największa.

Pewne kolory przyciągają się mocniej inne słabiej(współczynnik) dodatkowo każdy z brył ma swoja masę. Wzór siły wyraża się przez:

$$F = \frac{G(c1, c2) \ * \ m1 \ * \ m2}{d}$$

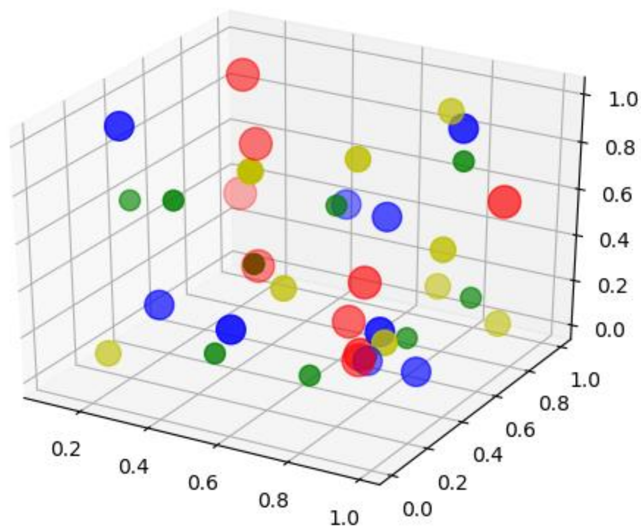
Gdzie:

Funkcja G - wylicza współczynnik oddziaływania między dwoma kolorami.
c1,c2 – kolory 2 punktów
m1,m2 – masa 2 punktów
d – odległość między dwoma punktami

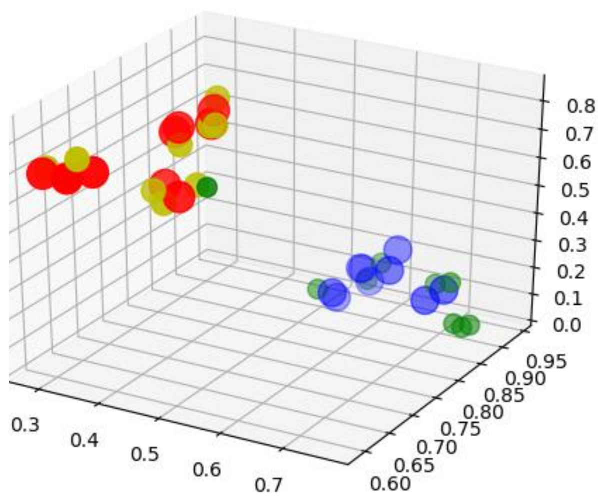
Do programu dostarczamy losowych punktów na mapie 3d.
Następnie program za pomocą wyżarzania optymalizuje otrzymany układ.

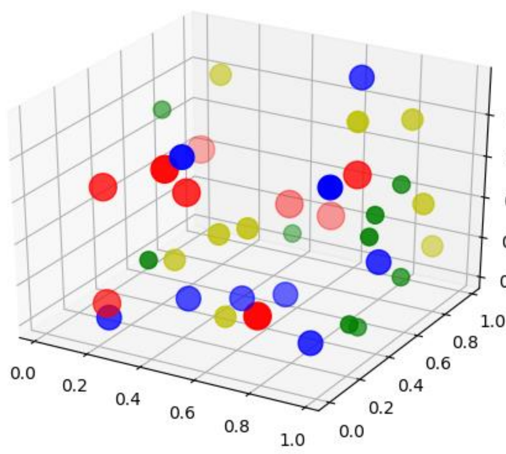
Przykłady działania na poniższych rysunkach:

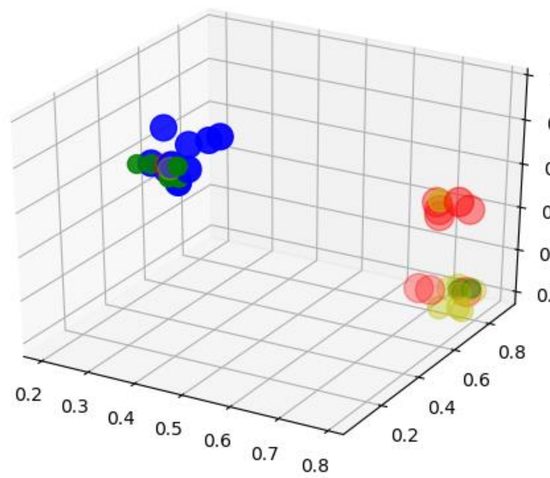
Przed wykonaniem:

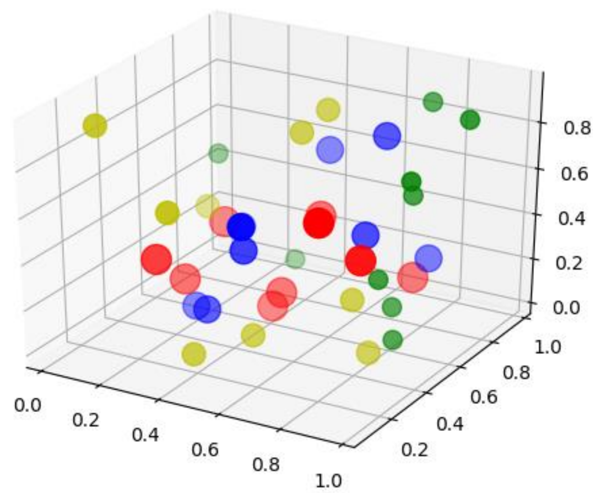


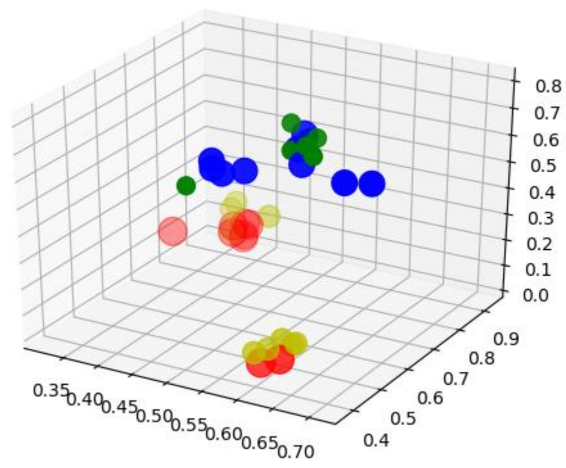
Po wykonaniu programu:




0.0 0.2 0.4 0.6 0.8 1.0
0.0 0.2 0.4 0.6 0.8 1.0
0.0 0.2 0.4 0.6 0.8


0.2 0.3 0.4 0.5 0.6 0.7 0.8
0.2 0.4 0.6 0.8
0.0 0.2 0.4 0.6 0.8 1.0


0.0 0.2 0.4 0.6 0.8 1.0
0.2 0.4 0.6 0.8 1.0
0.0 0.2 0.4 0.6 0.8


0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70
0.4 0.5 0.6 0.7 0.8 0.9
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8

Zadanie 3. Zaproponuj metodę rozwiązywania sudoku metodą symulowanego wyżarzania.

Program działa następująco.

Otrzymuje pewien zbiór punktów z sudoku (początkowy stan) które w programie są oznaczone jako statyczne(nie można ich przemieszczać).
Pozostałe puste pola są inicjalizowane jako 0.

```
+-+-+-+-+-+-+-+-+-+
|0|1|0|6|0|4|3|0|7|
+-+-+-+-+-+-+-+-+-+
|3|5|6|0|0|0|0|0|0|
+-+-+-+-+-+-+-+-+-+
|0|0|0|0|5|3|6|9|0|
+-+-+-+-+-+-+-+-+-+
|0|8|3|2|6|0|4|0|9|
+-+-+-+-+-+-+-+-+-+
|0|0|0|0|0|0|0|0|0|
+-+-+-+-+-+-+-+-+-+
|4|0|5|0|7|8|2|6|0|
+-+-+-+-+-+-+-+-+-+
|0|4|2|5|3|0|0|0|0|
+-+-+-+-+-+-+-+-+-+
|0|0|0|0|0|0|7|2|4|
+-+-+-+-+-+-+-+-+-+
|7|0|9|4|0|2|0|8|0|
+-+-+-+-+-+-+-+-+-+
```

Wypelnia wiersz w ten sposób żeby znajdowalo się w nim po jednym wystapieniu z kazdej cyfry (wstawia je w miejsce 0)

```
+-+-+-+-+-+-+-+-+-+
|2|1|5|6|8|4|3|9|7|
+-+-+-+-+-+-+-+-+-+
|3|5|6|1|2|4|7|8|9|
+-+-+-+-+-+-+-+-+-+
|1|2|4|7|5|3|6|9|8|
+-+-+-+-+-+-+-+-+-+
|1|8|3|2|6|5|4|7|9|
+-+-+-+-+-+-+-+-+-+
|1|2|3|4|5|6|7|8|9|
+-+-+-+-+-+-+-+-+-+
|4|1|5|3|7|8|2|6|9|
+-+-+-+-+-+-+-+-+-+
|1|4|2|5|3|6|7|8|9|
+-+-+-+-+-+-+-+-+-+
|1|3|5|6|8|9|7|2|4|
+-+-+-+-+-+-+-+-+-+
|7|1|9|4|3|2|5|8|6|
+-+-+-+-+-+-+-+-+-+
```

Następnie wyliczamy współczynnik poprawności układu:
1) Im niższy tym rozwiązanie bliższe poprawnemu
2) Jego wartość jest dla każdego punktu w którym to jego częsciowa wartość jest równa sumie liczby powtórzeń danej cyfry na w kolumnie i wierszu której występuje
3) W powyższym przypadku jest on równy

```
WYNIK
96
```

Następnie metodą wyżarzania próbujemy rozwiązać sudoku.

```
+-+-+-+-+-+-+-+-+-+
|9|1|8|6|2|4|3|5|7|
+-+-+-+-+-+-+-+-+-+
|3|5|6|7|9|1|8|4|2|
+-+-+-+-+-+-+-+-+-+
|2|7|4|1|5|3|6|9|8|
+-+-+-+-+-+-+-+-+-+
|1|8|3|2|6|5|4|7|9|
+-+-+-+-+-+-+-+-+-+
|6|2|7|8|4|9|1|3|5|
+-+-+-+-+-+-+-+-+-+
|4|3|5|9|7|8|2|6|1|
+-+-+-+-+-+-+-+-+-+
|8|4|2|5|3|7|9|1|6|
+-+-+-+-+-+-+-+-+-+
|5|9|1|3|8|6|7|2|4|
+-+-+-+-+-+-+-+-+-+
|7|6|9|4|1|2|5|8|3|
+-+-+-+-+-+-+-+-+-+
0
```

W tym wypadku doszliśmy do poprawnego rozwiązania jednak zdarza się ze algorytm prowadzi do lokalnego minimum z którego nie chce wyjść.