

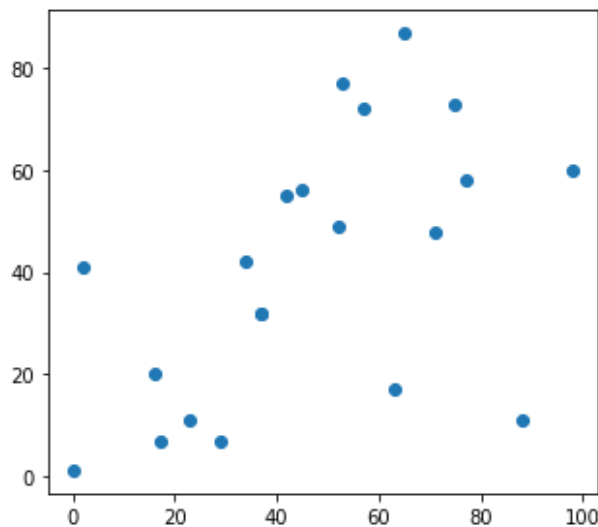
# Optymalizacja Kombinatoryczna Travelling Salesperson Problem

Jan Metzler 148137  
Mateusz Oleszek 144608  
Kamil Szostak 145308

Do realizacji zadania wykorzystaliśmy algorytm mrówkowy.

## 1. Inicjalizacja:

Do zobrazowania poszczególnych kroków algorytmu stworzyliśmy graf z 20 losowych punktów znajdujących się pomiędzy współrzędnymi 0 i 100.



## 2. Opis Algorytmu:

Rozwiązanie jest implementacją algorytmu mrówkowego.

Mrówki poruszając się w poszukiwaniu jedzenia zostawiają za sobą szlak złożony z feromonów.

Zostawione na szlaku feromony wyparowują w określonym tempie. Im droga jest krótsza, tym więcej feromonów na niej zostanie. Mrówki wybierają swoją trasę na podstawie odległości do następnego celu oraz właśnie stężenia feromonów.

Na początku mrówki poruszają z dużą losowością, jednak te które znajdą krótsze trasy będą zostawiały na nich więcej feromonów. Dzięki temu w następnych iteracjach inne mrówki będą chętniej podążały tymi trasami lub ich odcinkami, wciąż mając szansę na znalezienie jeszcze lepszych cykli.

Ten mechanizm pozwala na eksplorację różnych możliwych rozwiązań, jednocześnie promując te najlepsze, dzięki czemu wraz z kolejnymi iteracjami rozwiązanie przybliża się stopniowo do optymalnego.

### 3. Pseudokod:

Dla  $t=1$  do *ilość iteracji*

    Dla  $m=1$  do *ilość mrówek*

        Dopóki mrówka nie skończy cyklu

            Mrówka wybiera następne miasto  $j$  do odwiedzenia

            z prawdopodobieństwem  $p_{ij}$  wyliczonym za pomocą wzoru (1)

        Zapisać dystans całego cyklu mrówki,

        podmienić z dotychczasowym najkrótszym, jeśli jest lepszy od niego.

    Zaktualizować poziomy feromonów na ścieżkach za pomocą równań (2)

Koniec

(1). Prawdopodobieństwo wybrania miasta przez mrówkę

$$p_{ij}^m = \frac{[\tau_{ij}]^\alpha * [\eta_{ij}]^\beta}{\sum_{s \in \text{możliwe}} [\tau_{is}]^\alpha * [\eta_{is}]^\beta}$$

$i$ -obecne miasto,  $j$ -potencjalne miasto docelowe,  $\tau_{ij}$ -intensywność feromonów na ścieżce,  $\eta_{ij}$ -widoczność ścieżki, czyli odwrotność odległości,  $\alpha$  i  $\beta$ - współczynniki regulujące co ma preferować mrówka,  $s$ -miasta w których mrówka jeszcze nie była

Jeśli mrówka odwiedziła już miasto  $j$  to prawdopodobieństwo wynosi 0.

Wszystkie prawdopodobieństwa sumują się do 1.

(2). Aktualizacja poziomów feromonów

$$\tau_{ij}(t+1) = \rho * \tau_{ij}(t) + \Delta \tau_{ij}$$
$$\Delta \tau_{ij} = \sum_{m=1}^{\text{ilość mrówek}} \Delta \tau_{ij}^m$$

$\rho$  – parametr zanikania feromonów ustalony między wartościami  $[0,1]$

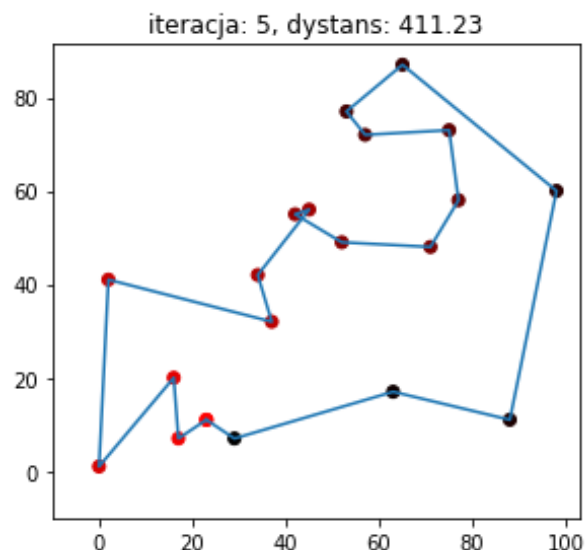
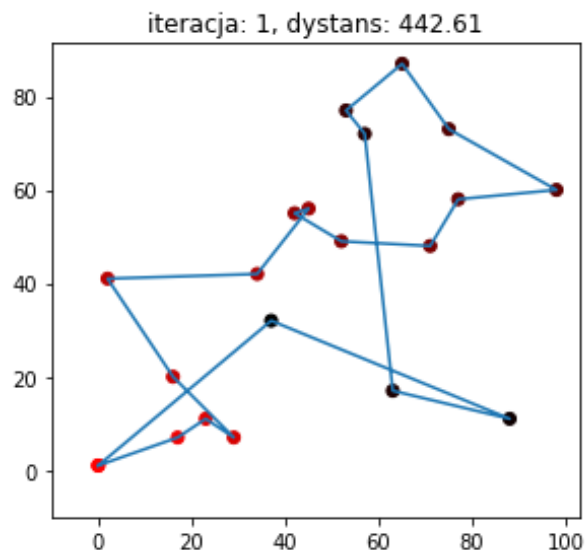
Nowa wartość feromonów na ścieżce to poprzednia wartość zmniejszona przez parametr zanikania plus suma feromonów które naniósł mrówki w obecnej iteracji.

$$\Delta \tau_{ij}^m = \begin{cases} \frac{Q}{L_m} & \text{jeśli mrówka } m \text{ przeszła przez krawędź } (i,j) \\ 0 & \text{w pozostałych przypadkach} \end{cases}$$

$Q$ -ustalona wartość wydzielania feromonów,  $L_m$ -długość całego cyklu który mrówka przeszła w obecnej iteracji

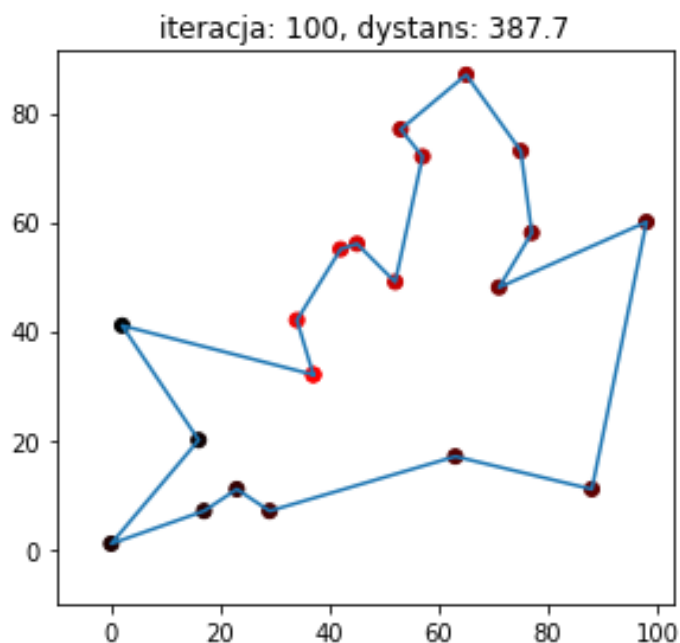
#### 4. Przykład obrazujący działanie:

Na trasę zostało wypuszczonych 20 mrówek startujących w losowych miastach. Parametry zostały ustawione następująco:  $\alpha = 1$ ,  $\beta = 4$ ,  $\rho = 0.6$ . Limit iteracji został ustawiony na 100.



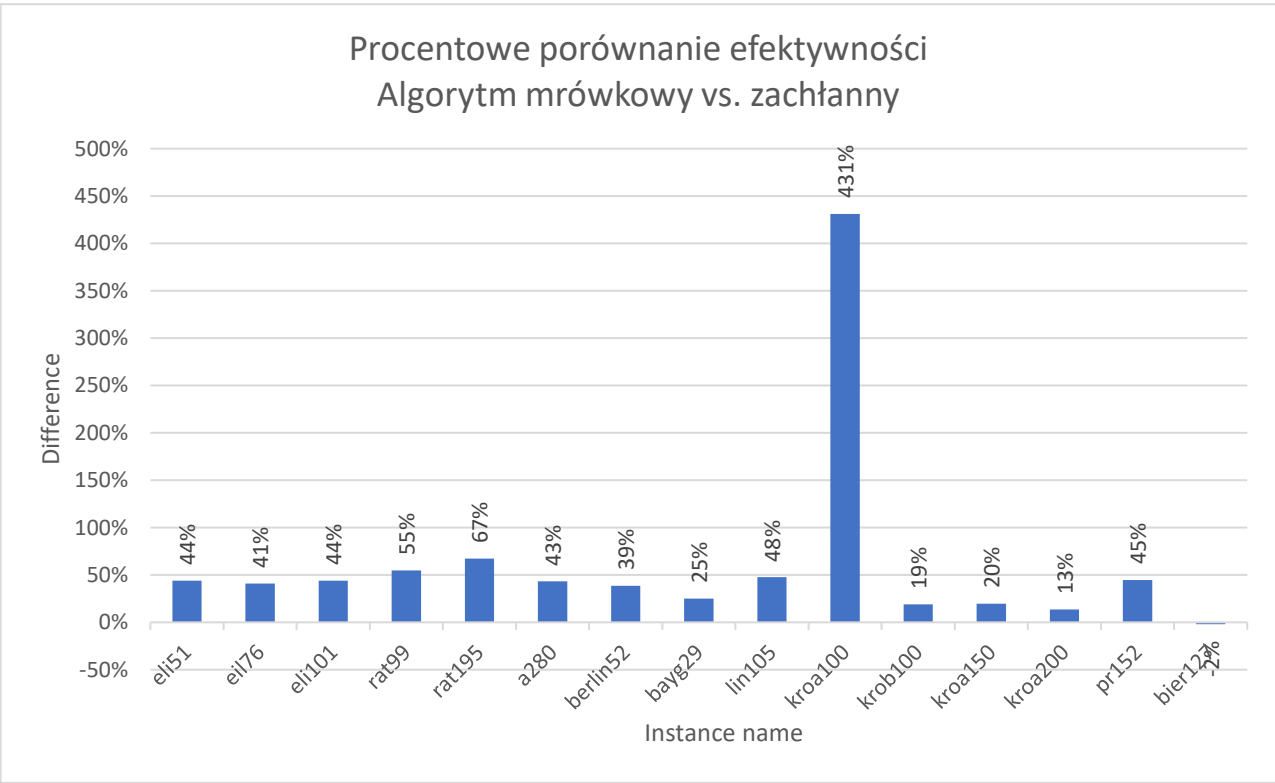
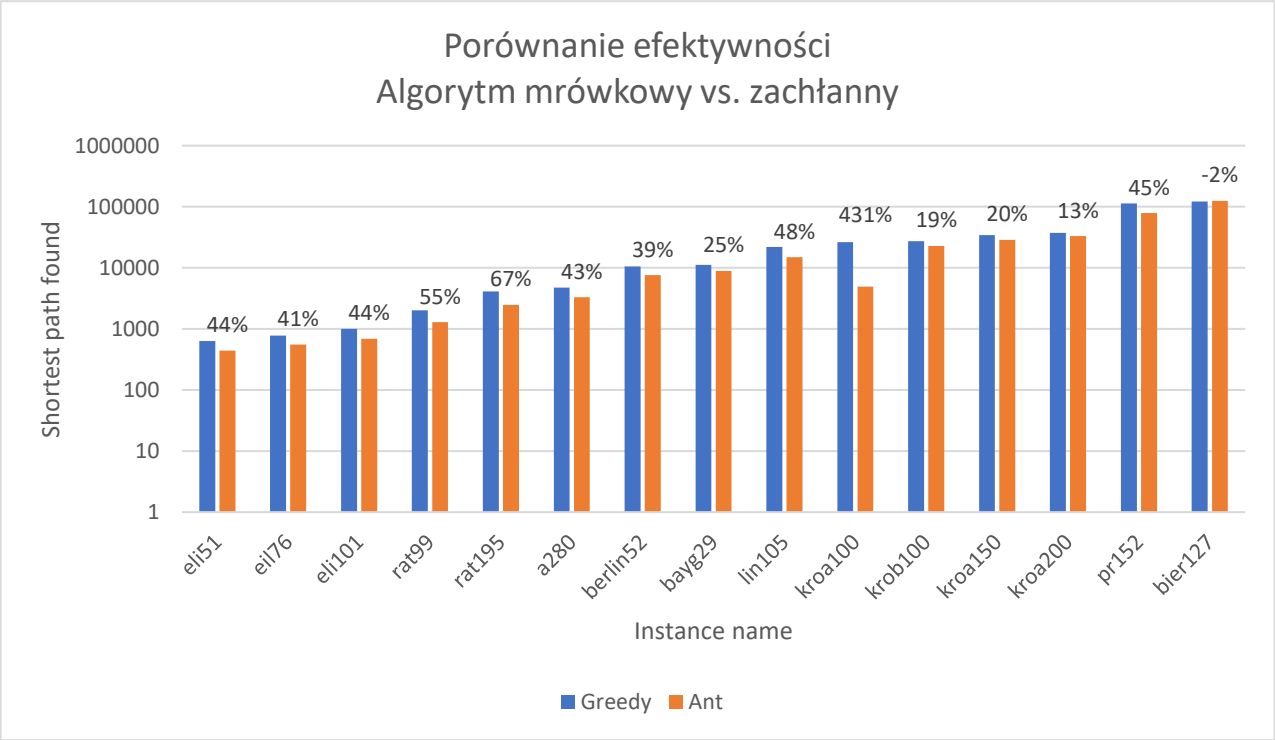
#### 5. Finalizacja – efekt działania algorytmu:

Algorytm zakończył się po ze znalezionym dystansem 387.7 (12.4% lepszy niż początkowy). Choć został on osiągnięty już po 44 iteracjach.



Wykresy

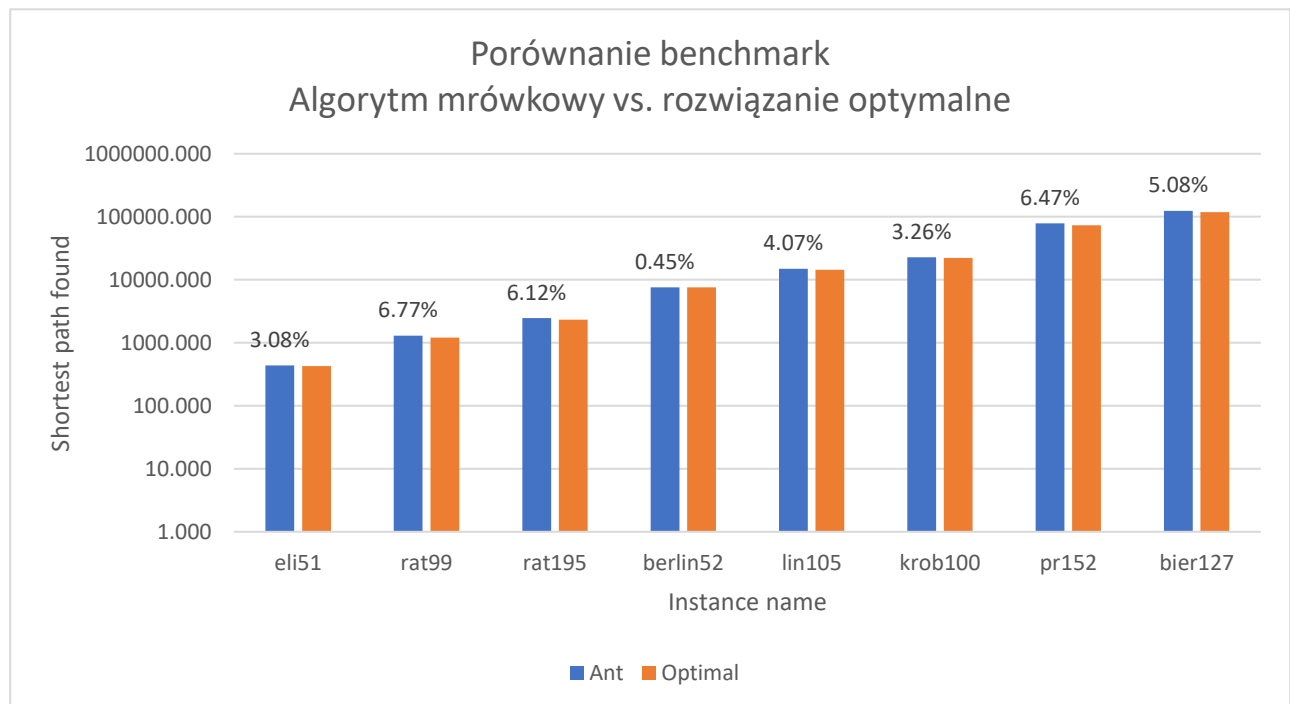
1. Porównanie optymalizowanej wartości algorytmu mrówkowego z algorytmem zachłannym:



Wykresy zostały sporządzone na podstawie tej tabelki:

Name	Greedy	Ant	Percent
eli51	631.662	439.111	43.85%
eil76	778.976	553.231	40.80%
eli101	994.722	690.486	44.06%
rat99	2003.02	1292.980	54.92%
rat195	4126.24	2465.190	67.38%
a280	4749.72	3318.270	43.14%
berlin52	10495.9	7576.250	38.54%
bayg29	11151.4	8926.390	24.93%
lin105	22076.1	14963.600	47.53%
kroa100	26184	4931.570	430.95%
krob100	27206.6	22862.500	19.00%
kroa150	34319	28694.200	19.60%
kroa200	37435.8	33004.200	13.43%
pr152	113515	78449.900	44.70%
bier127	121777	124288.000	-2.02%

2. Wartość błędu względnego algorytmu w stosunku do wartości optymalnej:



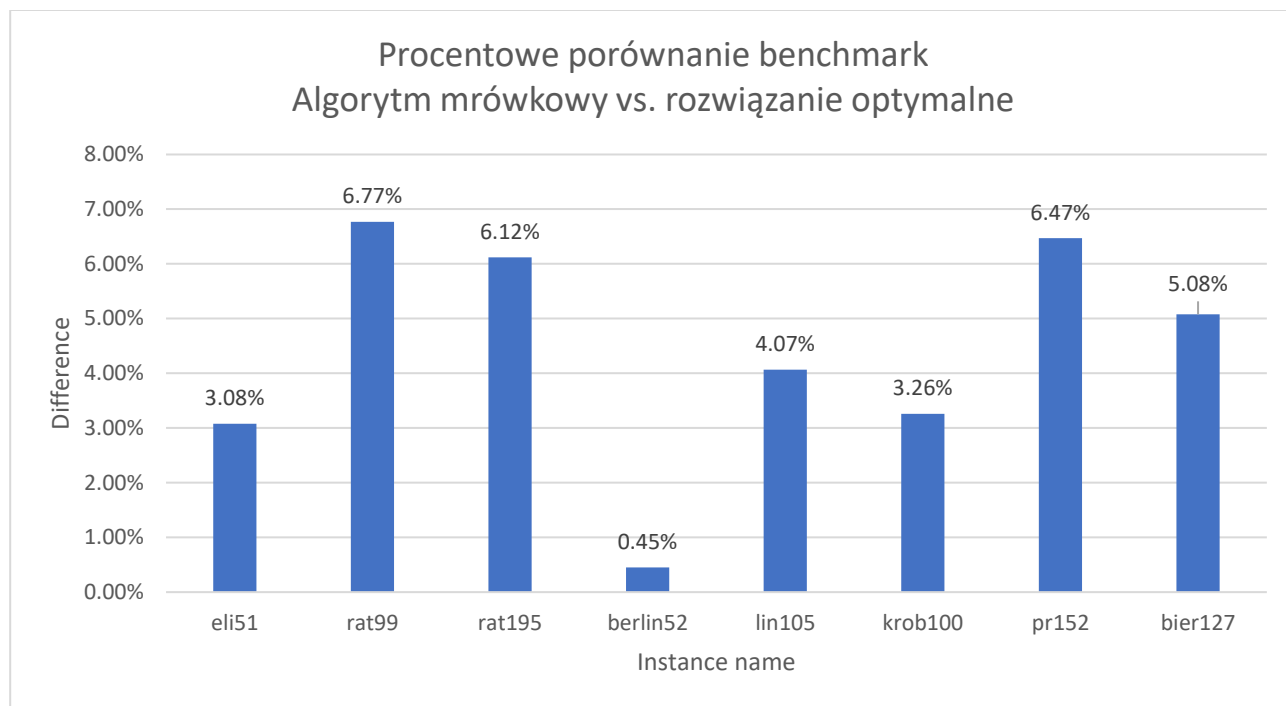


Tabela z danymi:

Name	Ant	Optimal	Percent
eli51	439.111	426	3.08%
rat99	1292.980	1211	6.77%
rat195	2465.190	2323	6.12%
berlin52	7576.250	7542.000	0.45%
lin105	14963.600	14379.000	4.07%
krob100	22862.500	22141	3.26%
pr152	78449.900	73682	6.47%
bier127	124288.000	118282	5.08%

3. Wyniki dla poszczególnych instancji z tabeli wyników:

<b>Berlin52</b>	<b>7576.25</b>
<b>Bier127</b>	<b>124288</b>
<b>Tsp1000</b>	<b>32079.7</b>
<b>Tsp500</b>	<b>100598</b>
<b>Tsp250</b>	<b>13525.8</b>