

Algorytmy Metaheurystyczne

Lista 1

Szymon Janiak

November 28, 2023

Opis problemu

Obliczamy wagę minimalnego drzewa rozpinającego, średnią wartość uzyskanego rozwiązania, średnią liczbę kroków poprawy oraz najlepsze uzyskane rozwiązanie dla 3 sposobów z wykorzystaniem algorytmu Local Search

1. Local Search użyty na cyklu stworzonego z minimalnego drzewa rozpinającego
2. Local Search użyty na losowej permutacji
3. Local Search użyty na losowej permutacji z wybieraniem najlepszego sąsiada z n losowo wybranych

Waga uzyskanych najlepszych rozwiązań

nic	MST	TSP	<i>local_search</i> 1	<i>local_search</i> 2	<i>local_search</i> 3
xqf131	476	756	540	564	574
xqg237	906	1380	1097	1156	1160
pma343	1183	1810	1476	1524	1585
pka379	1160	1851	1431	1494	1461
bcl380	1453	2307	1748	1775	1773
pbl395	1132	1753	1374	1400	1430
pbk411	1189	1793	1418	1486	1492
pbn423	1210	1873	1468	1537	1537
pbm436	1277	1970	1529	1562	1565
xql662	2249	3571	2713	2805	2809
xit1083	3256	4996	3834	4063	4016
icw1483	4017	5927	4668	5020	5112
djc1785	5550	8282	6440	6749	7208
dcb2086	5958	9186	7063	7321	7476
pds2566	6964	10673	8219	8611	9830

Przykład	<i>local_search</i> 1		<i>local_search</i> 2		<i>local_search</i> 3	
	średnia		średnia		średnia	
	liczba popraw	suma wag	liczba popraw	suma wag	liczba popraw	suma wag
xqf131	28.8	701.9	62.3	712.4	51.9	722.4
xqg237	37.0	1483.7	130.5	1415.8	112.2	1445.9
pma343	75.8	1650.3	201.9	1684.7	183.3	1717.1
pka379	78.9	1716.3	224.7	1745.9	204.9	1742.7
bcl380	62.7	1949.3	223.7	1917.5	189.2	1913.5
pbl395	80.0	1771.5	227.8	1829.0	193.4	1876.2
pbk411	81.6	1837.9	242.7	1888.5	203.6	2181.15
pbn423	72.3	1873.9	249.0	1921.6	219.8	2081.5
pbm436	97.85	2562.5	256.8	2312.0	217.4	2363.1
xql662	122.6	3690.4	405.4	3813.3	349.1	3917.5
xit1083	191.6	4312.2	693.7	4721.2	623.9	4826.1
icw1483	278.8	5034.6	973.4	5301.2	834.5	5718.2
djc1785	382.6	6733.8	1193.9	6872.1	1032.6	7764.5
dcb2086	390.9	7454.2	2392.6	7457.8	1222.2	8888.3
pds2566	457.3	8671.4	1742.9	8701.8	1526.0	10687.5

Wnioski

Z wyników można wywnioskować, że rozwiązanie tworzone poprzez *local_search* na podstawie MST daje najlepsze wyniki. Do tego widzimy znaczą poprawę względem zwykłego TSP. Testowanie części otoczenia zmniejsza średnią ilość przeprowadzonych popraw, jednak prowadzi do gorszych wyników.

Wizualizacje





