Journal of Al-Nahrain University Vol.17 (2), June, 2014, pp.213-218 Science

The Double Node Local Optimizer for Travelling Salesman Problem

Azhar Waleed Hammad
Department of Mathematics and Computer Applications, College of Sciences,
Al-Nahrain University, Baghdad-Iraq.

Abstract

In this paper we introduce a new local optimizer consists of two operations, swap and insert. Our new local optimizer can improve the travelling sales man shortest route problem obtained by metaheuristic algorithms directly and also those routes found by using other local optimizers. To see how well the new local optimizer, it was tested on two problems, the first is a benchmark problem and it is succeeded in reducing the published optimum route, the second problem is handling the problem of finding the shortest route for Arab capitals. Also, the double node local optimizer was used to improve the route obtained by nearest neighborhood method.

Introduction

The travelling sales man problem (or TSP) is a well-known scheduling problem, it first appearances in literature was in a 1757 paper by the great Leonhard Euler, today there are many important cases of practical problems that can be formulated as TSP like drilling of printed circuits boards, data transmission computer network, image processing, pattern recognition, the overhauling of gas turbine engines, in material handling in a warehouse, vehicle routing problem,…, etc [1].

Although TSP are simple but they aredifficult to solve and the available methods (exact or approximate) can only solve small problems and fail to solve large (practical) problems in reasonable time. In 1972 Richard M. Karp showed that the problem of finding a Hamiltonian cycle was NP-complete, which implies the NP-hardness of TSP, and because of that many researchesusemetaheuristic algorithms to solve TSP like genetic algorithm, ant colony, particle swarm optimization,…etc, to solve the TSP, the main disadvantage associated with metaheuristic algorithms is that the solution obtained cannot be guaranteed to be the shortest one, because metaheuristic algorithms can easily be trapped in local minimum, so local optimizer are used to improve the solution obtained by metaheuristic algorithms [2].

The double node local optimizer

Generally speaking there are two types of optimizers; global and local. Global optimizers search solution space for global minimum or maximum and are designed to balance between diversification and intensification of the solution space in order to find the global optimum. The main weakness point of global optimizers it can be easily fall in local optima, but even if it fall the solution it obtained still considered close to (global) optimum, the other type of optimizers i.e. local optimizer are used to improve the result obtained by global optimizers by searching the neighborhood of the global optimizers solution. The most famous local optimizers are the hill climb algorithms, and it is always a good idea to use more than one local optimizer in series to improve the solution in hand [3].

To illustrate the double node local optimizer suppose we has a certain TSP that consist of (n) citiesand we use as metaheuristica global optimizer (like ant colony, genetic algorithm,…etc.) to find the shortest route which can be represented by the sequence (D, n_1,n_2,…,n_{i-1},n_i,n_{i+1},…,n_n,D)where D means depot…...(route 1)

The first step in double node local optimizer is to perform the swap operation, i.e. replacing city n_i by n_j where i=1,2,..,n, j=1,2,…,n and $j \neq i$, this will result the following route: D, n_1, n_2,…, n_{i-1}, n_j, n_{i+1},…, n_{j-1}, n_j,$n_{j=1}$,…,n_n,D …..(route2)

We can notice that there are two problems in the resulted (i.e in (2)), they are:-

1- City n_j appear two times.
2- City n_i absent.

To solve these two problems we do the following:

1- Remove the old n_j city from the route, so the rout will be :-
D, n_1, n_2,…, n_{i-1}, n_j, n_{i+1},…, n_{j-1},$n_{j=1}$,…,n_n,D…..(route 3)

2- Reinsert city n_i in the route between city n_r, n_{r+1}, where r=1,2,…,n, so the route will be:-

D, n_1, n_2,…, n_{i-1}, n_j, n_{i+1},…, n_r, n_i, n_{r+1},…, n_{j-1},$n_{j=1}$,…,n_n, D.....(route4)

3- Recalculate the distance of the last route (4) and if it is less than the shortest route found yet (2) then this new route (4) will be considered as the new shortest route and neglect the previous one (2), otherwise we neglect route (4), and continue with swap operation for another city

It worth to note here that our new local optimizer differ from other known local optimizers like, swap, insert, 2-opt and other hill climb algorithms that it make two move at the same time (like 3-opt. local optimizer), while other localoptimizer make only one move, and that why we call it double node local optimize.

Fig.(1) illustrates pseudo code of the double node local optimizer.

Input:

x-axis for each city.
y-axis for each city.
GBR: global best route.
N: no. of cities to be linked by shortest route.

Output:

1. Sequence of the (n) cities represent shortest route.
2. Shortest distance.

Algorithm:

1. GBR=entering the route and calculate distance of the initial route.
2. For i=1 to n (to represent positions in route).
   For j= 1 to n (to represent cities).
   a. If city (i)= city (j) then exit
   b. Swap city(i) with city (j)
   c. Delete j from the route
   d. Reinsert city (i) at each possible location
   e. Calculate the new route distance
   f. If new route (distance) < GBR(distance) then GBR=new route found
      Go to 2
   Next j
   Next i
3. Print the latest shortest route found (i.e. GBR) and it's distance.

Fig. (1) Pseudo code of the double node local optimizer.

Test problems

In order to check the efficiency of our double node local optimizer and find how good it is, we test it on two problems. The first one is a benchmark problem known as Eil 101 which consists of 101city, this bench mark problem has optimum published route of 642.3095, we take the published route and apply the double node local optimizer and we find new shortest route of 640.4222, the new route details is shown in Table (1) below, other information about this instances such as distances, cities (x, y) coordinates can be found with much detail in www.iwr.uni_heidelberg.de/groups/comopt/software/TspLIB95.

Table (1)
Shortest route of bench mark problem, Eil 101 after applying the double node local optimizer.

bench mark problemname	*New shortest route after applying Double node local optimizer*
Eil 101	1-69-27-101-53-28-26-12-80-68-29-24-54-4-55-25-39-67-23-56-75-41-22-74-72-73-21-40-58-13-94-95-97-87-2-57-15-43-42-14-44-38-86-16-61-85-91-100-98-37-92-59-93-99-6-89-52-18-83-60-5-84-17-45-8-46-47-36-49-64-63-90-32-10-62-11-19-48-82-7-88-31-70-30-20-66-65-71-35-34-78-81-9-51-33-79-3-77-76-50-1

The second problem we handle to test the goodness of the double node local optimizer is to find shortest route between Arab capitals state. In first step we collect information about Arab capitals statedistances using the website www.distancefromto.net, and the collected distances are listed below in Table (2).

Table (2)
Arab capital Distances.

Arab Capitals	*Abu Dhabi*	*Amman*	*Baghdad*	*Beirut*	*Damascus*	*Doha*	*Jerusalem*	*KuwaitCity*	*Manama*	*Muscat*	*Riyadh*
Abu Dhabi	0	1986.32	1381.17	2107.67	2021.84	301.18	2044.64	836.34	427.55	434.85	773.99
Amman	1986.32	0	808.78	219.15	176.31	1691.46	72.31	1189.89	1558.86	2405.15	1327.51
Baghdad	1381.17	808.78	0	829.72	754.16	1129.35	880.96	555.04	989.75	1751.43	984.58
Beirut	2107.67	219.15	829.72	0	86.06	1820.09	237.5	1284.48	1683.28	2515.1	1491.97
Damascus	2021.84	176.31	754.16	86.06	0	1734.06	219.38	1199.64	1597.31	2429.91	1407.67
Doha	301.18	1691.46	1129.35	1820.09	1734.06	0	1748.41	574.45	140.47	734.95	488.21
Jerusalem	2044.64	72.31	880.96	237.5	219.38	1748.41	0	1251.44	1617.09	2465.61	1373.81
KuwaitCity	836.34	1189.89	555.04	1284.48	1199.64	574.45	1251.44	0	434.71	1231.32	533.22
Manama	427.55	1558.86	989.75	1683.28	1597.31	140.47	1617.09	434.71	0	853.9	422.47
Muscat	434.85	2405.15	1751.43	2515.1	2429.91	734.95	2465.61	1231.32	853.9	0	1205.97
Riyadh	773.99	1327.51	984.58	1491.97	1407.67	488.21	1373.81	533.22	422.47	1205.97	0
Sana'a	1468.5	2029.35	2000.86	2241.41	2172.24	1342.57	2042.65	1607.65	1378.82	1760.92	1074.54
Tunis	4407.66	2421.34	3128.34	2315.82	2399.53	4112.12	2363.73	3597.9	3980.17	4824.77	3720.05
Tripoli	4102.53	2136.98	2902.18	2072.61	2150.35	3802.43	2071.94	3323.37	3676.85	4530.07	3379.74
Mogadishu	2678.9	3473.98	3484.72	3691.39	3628.63	2671.23	3476.82	3055.77	2749.69	2791.11	2528.86
Rabat	5952.93	3972.02	4697.73	3879.89	3962.43	5654.24	3910.74	5156.28	5525.84	6376.29	5239.38
Nouakchott	7857.07	5406.92	6205.59	5382.13	5453.16	6959.34	5336.5	6561	6852.05	7691.58	6487.46
Djibouti	1877.06	2342.21	2400.62	2559.66	2497.34	1768.16	2345.46	2031.28	1808.04	2136.96	1499.24
Moroni	4203.49	4921.99	5013.83	5141.13	5086.65	4213.41	4914.39	4600.09	4294.97	4268.67	4070.97
Khartoum	2485.79	1858.83	2312.44	2063	2035.13	2254.02	1825.77	2206.99	2217.26	2866.01	1797.9
Cairo	2367	497.21	1300.31	587.21	615.62	2066.08	425.62	1619.13	1943.95	2797.75	1640.25
Algiers	5041.37	3055.23	3761.61	2950.77	3034.53	4745.23	2996.91	4232.88	4613.83	5459.36	4347.84

Arab Capitals	*Sana'a*	*Tunis*	*Tripoli*	*Mogadishu*	*Rabat*	*Nouakchott*	*Djibouti*	*Moroni*	*Khartoum*	*Cairo*	*Algiers*
Abu Dhabi	1468.5	4407.66	4102.53	2678.9	5952.93	7857.07	1877.06	4203.49	2485.79	2367	5041.37
Amman	2029.35	2421.34	2136.98	3473.98	3972.02	5406.92	2342.21	4921.99	1858.83	497.21	3055.23
Baghdad	2000.86	3128.34	2902.18	3484.72	4697.73	6205.59	2400.62	5013.83	2312.44	1300.31	3761.61
Beirut	2241.41	2315.82	2072.61	3691.39	3879.89	5382.13	2559.66	5141.13	2063	587.21	2950.77
Damascus	2172.24	2399.53	2150.35	3628.63	3962.43	5453.16	2497.34	5086.65	2035.13	615.62	3034.53
Doha	1342.57	4112.12	3802.43	2671.23	5654.24	6959.34	1768.16	4213.41	2254.02	2066.08	4745.23
Jerusalem	2042.65	2363.73	2071.94	3476.82	3910.74	5336.5	2345.46	4914.39	1825.77	425.62	2996.91
KuwaitCity	1607.65	3597.9	3323.37	3055.77	5156.28	6561	2031.28	4600.09	2206.99	1619.13	4232.88
Manama	1378.82	3980.17	3676.85	2749.69	5525.84	6852.05	1808.04	4294.97	2217.26	1943.95	4613.83
Muscat	1760.92	4824.77	4530.07	2791.11	6376.29	7691.58	2136.96	4268.67	2866.01	2797.75	5459.36
Riyadh	1074.54	3720.05	3379.74	2528.86	5239.38	6487.46	1499.24	4070.97	1797.9	1640.25	4347.84
Sana'a	0	4121.96	3684.1	1487.93	5499.87	6395.99	429.83	3013.39	1252.75	2105.63	4704.7
Tunis	4121.96	0	518.41	5291.63	1571.14	3299.43	4269.46	6416.41	3238.12	2092.03	635.02
Tripoli	3684.1	518.41	0	4801.47	1860.96	3344.55	3803.77	5901.23	2742.91	1740.25	1021.99
Mogadishu	1487.93	5291.63	4801.47	0	6472.77	6920.99	1131.77	1546.43	2059.7	3456.18	5806.96
Rabat	5499.87	1571.14	1860.96	6472.77	0	1993	5569.51	7343.84	4439.87	3599.16	946.06
Nouakchott	6395.99	3299.43	3344.55	6920.99	1993	0	6313.46	7301.6	5162.58	4943.59	2790.63
Djibouti	429.83	4269.46	3803.77	1131.77	5569.51	6313.46	0	2619.97	1163.92	2342.98	4824.31
Moroni	3013.39	6416.41	5901.23	1546.43	7343.84	7301.6	2619.97	0	3255.51	4822.37	6841.46
Khartoum	1252.75	3238.12	2742.91	2059.7	4439.87	5162.58	1163.92	3255.51	0	1618.93	3747.5
Cairo	2105.63	2092.03	1740.25	3456.18	3599.16	4943.59	2342.98	4822.37	1618.93	0	2713.39
Algiers	4704.7	635.02	1021.99	5806.96	946.06	2790.63	4824.31	6841.46	3747.5	2713.39	0

The second step is to find an initial shortest route between these states using global optimizer and we choose the nearest neighborhood algorithm, in fact we find 22 routes each one begins with one of the 22 Arab capitals, the result (i.e.route distances) of these initial routes are shown in Table (3) below, the table also shows the result when we improve the initial shortest route using the double node local optimizer, the table also show the percentage of improvement by applying the double node local optimizer, one can easily note that in some cases the double node local optimizer canachievean improvement of 14.5%.we find that the shortest route after applying the double node local optimizer to the initially 22 routes was (23058.59) and the route detail are listed below in Table (4).

Table (3)
Arab capital states shortest routes obtained by nearest neighborhood algorithm and the improved shortest routes by double local optimizer.

Route Begging City	*Nearest neighborhood shortest route*	*Double node shortest route*	*Improvement percentage*
Abu Dhabi	30505.02	26025	14.686
Kuwait city	27860.82	23842.26	14.424
Doha	29615.13	26732.69	9.733
Muscat	27876.87	26025	6.643
Manama	27820.84	26042.44	6.392
Mogadishu	24581.13	23058.59	6.194
Moroni	24581.13	23058.59	6.194
Riyadh	27370.85	26025	4.917
Algiers	24217.5	23058.59	4.785
Amman	25017.89	23842.26	4.699
Jerusalem	24904.4	23842.26	4.265
Damascus	24902.31	23842.26	4.257
Beirut	24788.44	23842.26	3.817
Khartoum	24866.59	23918.21	3.814
Tunis	24761.49	23842.26	3.712
Sana'a	26512.93	26087.68	1.604
Baghdad	25260.22	24908.45	1.393
Djibouti	27679.24	27452.39	0.820
Rabat	23095.64	23058.59*	0.160
Nouakchott	23095.64	23058.59*	0.160
Tripoli	23879.31	23842.26	0.155
Cairo	23879.31	23842.26	0.155

**** shortest route between Arab capital states***

Table (4)
Detailed Arab capital states shortest route.

Baghdad-Kuwait_city-Manama-Muscat-abu_dhabi-Doha-Riyadh-Sana'a-Djibouti-mogadishu-Moroni-Khartoum-Nouakchott-Rabat-Algiers-Tunis-Tripoli-Cairo-Jerusalem-Amman-Beirut-Damascus.

Conclusion

In this paper a new local optimizer called the double node local optimizer, it aimed to improve the solution (i.e. shortest route) of TSP obtained by metaheuristic algorithms or those improved by other local optimizer. The double node local optimizer consists of two operations swap and reinsert. We apply our new local optimizer for two problems, the first one is a bench mark problem called Eil 101 which has a published shortest route and by using our local optimizer we were able to improve it, the second problem we used to test our local optimizer is to find shortest route among the 22 Arab capital states, we apply first themetaheuristicnearest neighborhood algorithm, then improve it by applying the double node local optimizer. So, it was able to achieve an improvement of 14.5%.

For further work we suggest to apply this local optimizer (i.e. the double node) to other combinatorial optimization problems in order to get better estimate of its efficiency, these problems like machine schedule, capacity vehicle routing problems … etc.

References

[1] Tarzjan M. S., Khademi M. Mohhammad R., Akbarzadeh T. and Moghaddam H.A., “A Novel Constructive- optimizer Neural Network for the traveling Salesman Problem”, IEEE transaction on System, Man, and Cybernetics, Vol. 37, No. 4, 2007

[2]. Applegate D.L., Bixby R.E., Chv´atal V., Cook W.J., “The Traveling Salesman Problem, A Computational Study”, Princeton University Press, Princeton and Oxford, 2006.

[3]. Showkat F.F., “A Modified ANT System Optimization Algorithm for the Capacitated Vehicle Routing Problem”, Journal of Al-Nahrain University Science Vol.14 (1), pp.161-170, 2011.

الخلاصة

في هذا االبحث قمنا بأيجاد محسن موقعي جديد يتكون من عمليتي التبديل والحشر، وان هذا المحسن الموقعي قد تمكن من تحسين الحلول المقدمه لمسألة رجل المبيعات المسافرمن قبل الخوارزميات الاخرى وكذلك الحلول المقدمة من قبل المحسنات الموقعية الاخرى ولغرض اختبار مدى كفاءة المحسن الموقعي الذي اوجدناه قمنا بتطبيقه على مسألتين لرجل المبيعات، الاولى هي مسألة عالمية معروفة بأسم (Eil 101) وتمكنا من تحسين الطريق الاقصر المنشور، اما المسألة الثانية فهي تحسين الحل الذي وجدناه من تطبيق خوارزمية الجار الاقرب لايجاد اقصر طريق بين عواصم الدول العربية، وتبين لنا ان المحسن الموقعي ثنائي العقدة قد تمكن في بعض الاحيان من تحسين الحل لنسبة ١٤،٥%.