Modified Grey Wolf Optimization to Solve Traveling Salesman Problem

Dibbendu Singha Sopto[1], Safial Islam Ayon[1], M. A. H. Akhand[1], and N. Siddique[2]

[1] Dept. of CSE, Khulna University of Engineering & Technology, Khulna, Bangladesh

[2]School of Computing, Engineering and Intelligent Systems, Ulster University, United Kingdom

***Abstract*—Traveling Salesman Problem (TSP) is arguably the most familiar combinatorial optimization problem. TSP is also very popular to check proficiency in any newly developed optimization method. In addition, the optimization methods which are developed for other tasks (e.g., numerical optimization), also test their proficiency in TSP. This study investigates a new technique to solve TSP based on a recently developed optimization method inspired by the hunting and social behavior of the grey wolf pack. Standard Grey Wolf Optimization (GWO) is developed for numerical optimization. In this study, GWO is modified and updated to solve TSP; and Swap Operators (SOs) and Swap Sequence (SS) are considered to adapt GWO for TSP. SS is an arrangement of several SOs in which each one holds two particular positions of a solution that might be swapped to make a new solution. In the proposed method, each grey wolf is considered as a TSP solution and SS is considered to update the solution. In every iteration, a new tour is formed by swapping the two position of a solution indicated by the SS to the previous solution. The proposed technique is tested on a several numbers of benchmark TSPs and final results are compared to the other prominent methods. Experimental consequences show that the proposed strategy is a decent technique to resolve TSP.**

Index Terms—Traveling Salesman problem, Swap Sequence, Swap Operator, Grey Wolf Optimization

I. Introduction

Travelling Salesman Problem (TSP) is probably the most familiar and largely used combinatorial optimization real-world problem. TSP is the problem faced by a man who is a salesperson who makes a journey from a specific city and travels to every other city in the shortest conceivable path. The salesman cannot visit any city twice before returning to the starting city [1]. TSP demonstrates all the parts of combinatorial optimization and comes under the set of NP-hard. If one can easily found a proposed solution for a specific problem in polynomial time then it is called *NP* problem. And if it as difficult as any NP problem then it is called *NP*-hard problem [2]. There are numerous real-life applications of TSP such as vehicle routing, computer wiring, X-Ray crystallography, printed circuit panel, and order selecting problem in warehouses [1].

TSP is also very popular to check proficiency in optimization methods. Nearly every year new method for resolving various problems such as engineering, optimization which are verified on the TSP, which is known as a standard test bench and in recent time interest has been grown-up to resolve it in modern methods. Ant Colony Optimization (ACO) [3] is the prominent one to answer TSP and is shown to perform well. In addition, the optimization methods which are developed for other optimization task (e.g., numerical optimization) are also modified to solve TSP [4, 10, 11]. A number of methods are also developed based on Particle Swarm Optimization (PSO) [4, 10, 11]. Among the methods, Swap Sequence (SS) based PSO methods are performed well to solve TSP.

This paper investigates a new technique to solve the TSP based on a recently developed optimization method using Grey Wolf Optimization (GWO) [6]. In the recent years, GWO is used by the researchers in the various fields. GWO is used to work with multi-verse optimizer [7], non-convex economic load dispatch problem [8], Hypercube interconnection network [9], and many other sectors. In this study, GWO is modified and updated to solve TSP, and Swap Operators (SOs) and SS are considered to adapt GWO for TSP.

GWO algorithm is a population-based meta-heuristics algorithm [6]. Like all others meta-heuristics algorithm, GWO starts with the uniformly generated random position of the grey wolves. The positions are updated in every iteration. There are some classifications in the society of the grey wolf. They maintain a hierarchy of their society. The leader of the folks of a grey wolf is called Alpha wolf, the wolf right next to the hierarchy is a beta wolf, and then there is Delta wolf is in the hierarchy. The rests are called Omega wolves. These leading three wolves play the key role for making any decision or making an update of the current position of the wolves. In the society of grey wolves, the hierarchy is maintained strictly and the best hunt agents which are Alpha, Beta and Delta work along to make the decisions. The existing best agents work for the duration until they are updated. In this situation, the other wolves work as the command of their leaders. They are called Omega wolves and they updated their position or value based on the location of the alpha, beta and delta wolves respectively.

Standard GWO is developed for numerical optimization. The components of GWO are updated and hence a new method is proposed to solve TSP in this study. The position of each grey wolf is considered a solution. Every grey wolf gives a solution individually. In the system, SS is considered to update the solution. A SS is a set of several SOs and each one holds two particular positions of a solution that should be swapped. All the calculated SOs of an SS are used on a specific solution preserving order and hence the application of the SS converts the TSP tour into another tour. The SOs are generated using the exact location of a specific grey wolf as well as the location of other best hunt agents (alpha wolf, beta wolf, and delta wolf) of the group.

Rest of the paper has been organized as follows. Section II narrates the proposed technique to solve TSP using GWO algorithm. Section III elaborates the experimental outcomes of the proposed method and compares the performance with other methods. Finally, Section IV presents a brief conclusion of the paper.

II. Proposed Grey Wolf Optimization to Solve TSP

This portion of the paper discusses proposed Grey Wolf Optimization (GWO) to solve TSP. In the suggested system, SS is used to update a solution. Following subsections explained the operators in detail to make the paper more complete.

A. Swap Operator in Solving TSP

A SO [4, 10-11] is a couple of indexes that imply two cities which may be swapped in a tour solution. Assume a solution of a TSP contain four cities in a sequence as defined by S = (4 – 1 – 3 – 2) and SO is SO (1,3), then the modified solution S' is:

$$S' = (4 - 1 - 3 - 2) + SO\ (1,3) = (3 - 1 - 4 - 2)$$

Here '+' means to apply the swap sequence in the solution.

A Swap Sequence [4, 10-11] is a set of one or additional SO(s) that may be applied on a specific solution one after another consecutively.

$$SS = (SO_1, SO_2, SO_3, \ldots, SO_n)$$

Here SO_1, SO_2, SO_3, …, SO_n are SOs. The SS may also get from solutions S_1 and S_2within the following equation.

$$SS = S_2 - S_1 = (SO_1, SO_2, SO_3, \ldots, SO_n)$$

Here '-' means want to calculate SO_s of SS on solution S_1 to S_2. Suppose two solutions are $S_1 = (1 - 2 - 3 - 4 - 5)$ and $S_2 = (2 - 3 - 1 - 5 - 4)$ then SS = *SO* (1,2), *SO* (2,3), *SO* (4,5).

It is citable that various SSs acting on the same solution may generate a similar new solution. All these SSs are named the equal set of SSs. Among all the SSs, the SS which has the minimum SOs is called Basic Swap Sequence (BSS).

Suppose two SS are: SS_1 = *SO* (2,3), *SO* (3,2), *SO* (4,1), *SO* (5,3), *SO* (3,4) and SS_2 = *SO* (4,1), *SO* (5,3), *SO* (3,4) is applied on S_1 = (e-a-b-c-d) independently, the result is S_2 = (c-a-e-d-b). Consequently, SS_2 is the BSS. Using $S_2 - S_1$ it will also be found.

Partial Search (PS) is one kind of SS, but here one SO is applied and calculated the result. Then the next SO is applied and continued. Here, every individual result is remembered and at last select the best one among them.

B. Modified GWO to solve TSP

The proposed GWO to solve TSP works with the Partial Search (PS) method. Algorithm 1 shows the basic steps of the proposed method and a brief description of the algorithm is given below.

1) Initialization

Like all other population-based swarm algorithm, GWO initiates the population with a number of random solutions and tries to enhance those solutions at each iteration step. In the initialization stage, the number of populations are defined and a termination criterion is developed. The tour cost is also calculated. Also, allocates a random solution (i.e., tour) to each of the particles. At the beginning stage, the fitness values (i.e., tour cost) of each particle is calculated and the particles of best three tour cost are considered as the best hunt agents and called Alpha, Beta, and Delta respectively.

Algorithm 1: GWO Algorithm to solve TSP

Step 1: Initializing the population,

Step 2: For each particle X_t in the population,

a. Find out all the swap sequence using Eq. (1)
b. Calculate the Basic Swap Sequence of the generated SS
c. Update Solution X_t using Partial Search manner using Eq. (2)
d. Update the best hunt agents in each iteration if there is any superior solution than the current agents.

Step 3: Jump to **Step 2** until a termination criterion is found.

Step 4: After all the iterations and operations the best solution (X_n) will be considered as the result.

2) Solution update of the Grey Wolfs

In each iteration step, GWO updates the position of each particle (i.e., Omega wolf) based on the position of the best hunt agents by generating a velocity (i.e., Swap Sequence) and applying the SS to the current solution. In the paper, when the SS are applied to a solution, then PS technique is applied. The velocity SS that it calculates using the following equation,

$$D = c_1 . |X_\alpha - X_t| + c_2 . |X_\beta - X_t| + c_3 . |X_\delta - X_t| \tag{1}$$

$$X_{(t+1)} = X_t + D \tag{2}$$

Here, X_α, X_β, X_δ indicates an alpha wolf, beta wolf, and delta wolf respectively. Eq. (1) indicates the SS which is gained from Alpha, Beta, and Delta wolf. Here c_1, c_2, c_3 are random variables which cut a portion of the whole SS to work with. Using Eq. (2) apply the SS to the current solution to generate a new solution.

For the velocity (i.e., position) calculation of a solution (i.e., grey wolf), GWO uses the solutions of the best hunt agents (X_α, X_β, X_δ) of the swarm. The calculated velocity is not applied to a position of the particle to find the new position like a usual method. But GWO applies the calculated velocity as the process of PS. Before applying PS, a BSS is applied to minimize the SS. With the PS, each SO of SS makes an individual tour solution, and particle takes the best one as its position among those. Finding the best tour is done by the partial search and that is based on the position of the best three hunt agents are the main feature of GWO.

The tour cost of each new solution of a particle is tested with those best three hunt agents. If the tour cost is found better, then among the best hunt agents the worst one will be skipped. GWO applies the PS operation on (X_t) for improving it than the previous and then it makes an update of the other parameters. Taking the global best agents as three unique particles, the process based on the chosen particles should be helpful and useful to enhance the whole outcomes of GWO with a better result. The $X\alpha$ is considered as the outcome of the whole operation. The tour cost of each new position (X_t) of a particle is compared with X_α, X_β, and X_δ. And, finally update them if $X(t+1)$ is shown a better result than the previous one.

3) Termination Criteria

GWO examines termination criterion after the finishing of each iteration and terminates if the criterion is found. Usually, an adequately good tour cost or a defined number of iterations is taken as the termination criteria. If the termination criterion is not found, GWO continuously updates the positions of the particles again and again as indicates the looping process to Step 2 from Step 3 in algorithm 1.

III. EXPERIMENTAL STUDIES

In this section, the outcomes of the proposed method to solve TSP using GWO compare to ACO, Genetic Algorithm (GA) [5,14] and Producer Scrounger Method (PSM) [12] to solve TSP. ACO and GA were two well-studied methods of TSP; on the other hand, PSM was the recently developed best-suited method for TSP. Here all the algorithms were tested on 15 different benchmark TSPs. For a fair comparison, the experimental methodology is chosen carefully.

A. Benchmark TSP Data and Experimental Methodology

In the above-mentioned examination, a set of 15 benchmark problems are taken from TSPLIB [13] where the number of cities are between 14 to 100.The experiment was built on Visual C++ of Visual Studio 2017. For appropriate understanding, tests have been led on a particular computer (HP ProBook 450G2, Intel(R) Core (TM) i5-5200U, CPU @ 2.20 GHz CPU, RAM 4 GB) with Windows10. For a reasonable examination, the investigational methodology was selected deliberately.

B. Experimental Analyses

In this section, the experimental analyses of GWO were compared to three prominent existing methods which are already implemented to solve TSPs. We considered ACO, GA, and PSM to compare the outcome of the GWO. We did this using two parameters: i) Effects of population size, and ii) Effects of total iteration. We chose Eil51 dataset, where the number of cities is 51, and for ACO the number of ants and the number of cities are equal. The results were taken as the average value of 10 individual runs.

Figure 1 shows the tour cost of various population sizes. Here, the population size varied from 5 to 500. For ACO, the number of ants and the number of cities are equal (i.e. 51) as it is desired. For all the problems, the total number of iterations was fixed at 500. It was observable from the Fig. 1 that when the population size was small (e.g. 5, 10) then all the methods gave worst tour cost. But increasing the population size the results were enhanced. Meanwhile, for a specific problem, population size was permanent in ACO; it demonstrated an unchanging performance in Fig. 1 and the performance was worst among the three methods.

Figure 2 demonstrates the accomplished the tour costs of ACO, GA, PSM, and GWO for different iterations. Here the population size was permanent at 100 but the iteration number was varied from 20 to 1000 for all methods. It was seen from Fig. 2 that when the iteration number was less than GWO, GA and PSM showed the worst result and the result was improving when the iteration number was increased up to a certain value. After that certain value, the changing result was not significant. In short, modified GWO has demonstrated the capacity to accomplishing better outcome in fluctuating iteration number and population size.

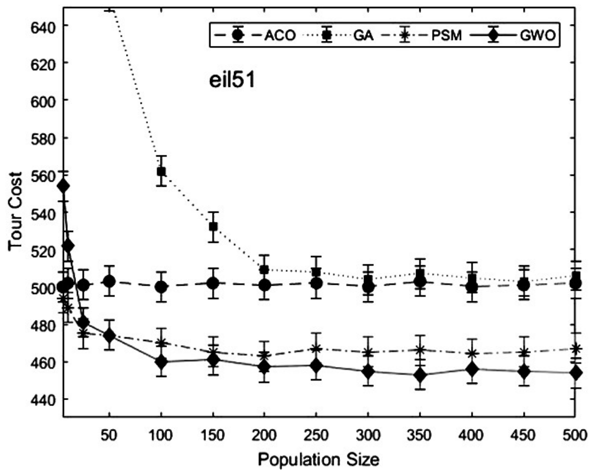

Figure 1. Variation effect of population size on tour cost.

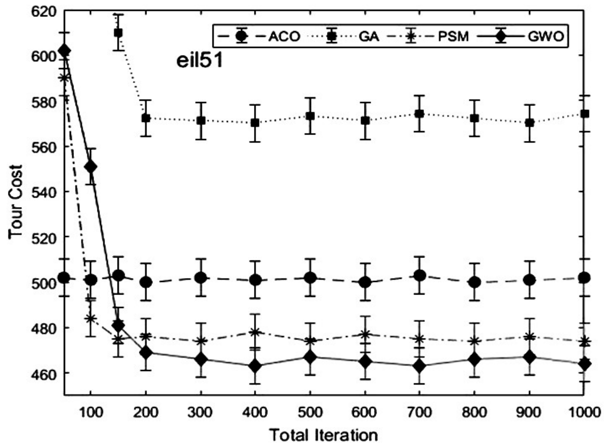

Figure 2. Variation effect of total iteration on tour cost.

C. Experimental Results and Comparison

This section compares the experimental results of GWO to ACO, GA, and PSM. In the above section, Fig. 1 and Fig. 2 shows that when the population size was greater than 100 and the iteration number was greater than 500, the result did not change significantly. So for comparison, the iteration number was fixed at 500 and population size was 100 for all the problems.

Table I compares the outcome of ACO, GA, PSM, and GWO for 15 different benchmark TSPs. Here all the problem was run 10 independent times. In Table I, of the four methods the best tour cost (i.e., smallest value) is indicated as the bold-face type and the worst tour cost (i.e., biggest value) is indicated as underlined. In the lowest portion of the table, a summary of the comparison was shown.

In Table I, average tour cost for all the problems demonstrate that GWO was better than GA, and PSM, but worse than ACO. The average tour cost for GWO was 4512.08 for the selected 15 problems. The achieved average tour cost for GA, ACO, and PSM was 7063.50, 4349.97 and 4858.51 respectively. ACO and GA showed the worst result in 2, and 13 problems respectively. On the other hand, PSM and GWO did not show any worst result. GWO showed the best result in 9 problems out of 15 problems.

TABLE I. COMPARISON BETWEEN THE PROPOSED GWO WITH GA, ACO, AND PSM TO SOLVE BENCHMARK TSPS

		Average Tour Cost (Standard Deviation)				Best Tour Cost			
Sl.	**Problem**	**GA**	**ACO**	**PSM**	**GWO**	**GA**	**ACO**	**PSM**	**GWO**
1	burma14	31.83 (0.89)	31.21 (0.0)	30.89 (0.07)	**30.87** (0.00)	**30.87**	31.21	**30.87**	**30.87**
2	ulysses16	74.79 (0.73)	77.13 (0.0)	74.2 (0.24)	**73.99** (0.00)	74.0	77.13	**73.99**	**73.99**
3	gr17	2458.36 (157.25)	**2332.58** (0.0)	2375.39 (66.24)	**2332.58** (0.00)	2332.58	2332.58	2332.58	2332.58
4	gr21	3033.82 (337.79)	2954.58 (2.0)	2838.22 (248.07)	**2714.65** (127.32)	**2672.27**	2949.81	**2672.27**	**2672.27**
5	ulysses22	79.62 (4.29)	86.81 (0.08)	76.68 (1.0)	**76.08** (0.28)	76.09	86.74	**75.51**	**75.51**
6	gr24	1402.01 (122.53)	**1267.13** (0.0)	1372.57 (71.18)	1289.23 (35.13)	**1249.82**	1267.13	**1249.82**	**1249.82**
7	fri26	689.49 (27.43)	646.48 (0.0)	675.24 (36.24)	**644.67** (13.19)	647.78	646.48	**635.58**	**635.58**
8	bays29	9981.49 (490.87)	9964.78 (0.0)	9917.59 (391.84)	**9219.40** (146.52)	9336.82	9964.78	**9076.98**	**9076.98**
9	hk48	16033.31 (1170.13)	12731.07 (81.41)	13870.94 (952.95)	**12117.05** (472.46)	14040.66	12699.86	12239.3	**11183.46**
10	eil51	592.3 (31.61)	504.83 (3.07)	474.58 (21.23)	**463.29** (8.61)	524.18	499.92	**438.7**	455.24
11	berlin52	10413.61 (690.02)	**8088.95** (11.58)	8865.08 (407.97)	8289.11626 (212.57)	9184.19	**8046.06**	8109.91	8048.91
12	st70	1203.35 (76.16)	**748.65** (7.23)	845.4 (47.65)	800.14 (27.68)	1015.0	**734.19**	767.65	752.84
13	eil76	926.4 (48.47)	**601.77** (3.12)	631.58 (27.84)	629.24 (18.58)	805.78	595.58	**591.89**	604.32
14	gr96	1092.04 (75.36)	**590.67** (7.61)	618.68 (37.33)	660.48 (28.49)	950.18	567.52	**564.47**	593.38
15	kroa100	57940 (3568.51)	**24623.01** (80.58)	30210.57 (2164.08)	28340.42 (1645.64)	51446.8	**24504.9**	26419.8	25983.8
	Average	**7063.50**	**4349.97**	**4858.51**	**4512.08**	**6288.47**	**4333.59**	**4351.96**	**4251.30**
	Best /Worst	**0/13**	**7/2**	**0/0**	**9/0**	**3/8**	**3/5**	**10/0**	**8/0**

The best result from different runs may use the outcome of a method. Therefore, minimum tour cost achieved from 10 runs is also compared in Table I for better understanding. GWO outperformed ACO, GA, and PSM based on the lowest tour costs showed in Table I. GWO showed the average lowest tour cost 4251.30 for the 15 problems. The average minimum tour costs of ACO, GA, and PSM were 4333.59, 6288.47 and 4351.96 respectively. But GWO showed 8 best result among the 15 problems. On the other hands PSM showed 10 best results, here 7 results are same for GWO and PSM. Based on the Best/Worst summary, GWO achieved the best result for 8 cases and did not show any worst result.

IV. CONCLUSION

In this paper, a modified Grey Wolf Optimization (GWO) is introduced to solve TSP. Mainly Standard GWO is a function optimization algorithm, but here the algorithm is modified in such a way that it can solve TSP which is a popular combinatorial optimization problem. In GWO, the grey wolf individual holds a solution. In every iteration, the grey wolf tries to update his result. The result is updated when Swap Sequence (SS) is applied to a particular solution. The proposed GWO showed a better result in most of the cases compared to GA, ACO, and PSM results. The possible future research opportunity is also opened by this analysis. It will be interesting to employ PS on other methods and remain as a future study. Therefore, a different method to identify an optimal subset of SOs owing to achieve the better result is also an open challenge after this study.

REFERENCES

[1] R. Matai, S.P. Singh, and M.L. Mittal, "Traveling Salesman Problem: AnOverview ofApplications, Formulations, and Solution Approaches," Traveling Salesman Problem, Theory and Applications, Edited by D. Davendra, InTech, pp 1-24, 2010.

[2] K.L. Hoffman, M. Padberg, and G. Rinaldi, "Traveling Salesman Problem," Encyclopedia of Operations Research and Management Science, pp. 1573-1578, 2013.

[3] Z. Chi, S.S. Hlaing, and M.A. Khine, "Solving Traveling Salesman Problem by using Improved Ant Colony Optimization Algorithm," International Journal of Information and Education Technology, vol. 1, pp. 404-409, December 2011.

[4] K. P. Wang, L. Huang, C. G. Zhou, and W. Pang, "Particle swarm optimization for traveling salesman problem," Proc. International Conference on Machine Learning and Cybernetics, pp. 1583–1585, November 2003.

[5] Z. Hua, and F. Huang, "A variable-grouping based genetic algorithm for large-scale integer programming," Information Science, vol. 176, pp. 2869–2885, 2006.

[6] Seyedali Mirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, "Grey Wolf Optimizer," Advances in Engineering Software, vol. 69, pp. 46-61, 2014.

[7] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Abdolreza Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization", The Natural Computing Applications Forum, 2015

[8] Vikram Kumar Kamboj, S. K. Bath, and J. S. Dhillon, "Solution of non-convex economic load dispatch problem using Grey Wolf Optimizer, "The Natural Computing Applications Forum, 2015.

[9] Ameen Shaheen, Azzam Sleit, and Saleh Al-Sharaeh, "Travelling Salesman Problem Solution Based-on Grey Wolf Algorithm over Hypercube Interconnection Network," Modern Applied Science, vol. 12, no. 8, 2018.

[10] X. Wei, Z. Jiang-Wei, and Z. Hon-lin, "Enhanced Self-Tentative Particle Swarm Optimization Algorithm for TSP," Journal of north China electric power university, vol. 36, no. 6, pp. 69–74, 2009.

[11] J. Zhang, and W. Si, "Improved Enhanced Self-Tentative PSO Algorithm for TSP," Proc. Sixth IEEE International Conference on Natural Computation, Yantai, Shandong, pp. 2638–2641, August 2010.

[12] M. A. H. Akhand, Pintu Chnadra Shill, Md. Forhad Hossain, and A. B. M. Junaed, "Producer-Scrounger Method to Solve Traveling Salesman Problem," Intelligent systems and applications, 2015.

[13] TSPLIB – a library of sample instances for the TSP. Available: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB9/

[14] Saad Alharbi and Ibrahim Venkat, "A Genetic Algorithm Based Approach for Solving the Minimum Dominating Set of Queens Problem.,"Journal of Optimization, 2017.