2020 11th International Conference on Information and Communication Systems (ICICS) 978-1-7281-6227-0/20/$31.00 ©2020 IEEE 10.1109/ICICS49469.2020.239528

Solving Multiple Traveling Salesmen Problem using Discrete Pigeon Inspired Optimizer

1st Hadeel Alazzam
Department of Computer Science
University of Jordan
Amman, Jordan
hdy9150096@ju.edu.jo

2nd Abdulsalam Alsmady
Department of Computer Engineering
Jordan University of Science and Technology
Irbid, Jordan
aralsmady15@cit.just.edu.jo

3rd Wail Mardini
Department of Computer Science
Jordan University of Science and Technology
Irbid, Jordan
mardini@just.edu.jo

***Abstract*—Multiple Traveling Salesmen Problem (MTSP) is an extended version of the popular traveling salesman problem. In MTSP there are m salesmen that should travel n cities with minimum cost (distance). All the salesmen start their trips from a depot city and back to the same city, each city must be visited once. MTSP is an optimization problem that researchers tried to solve using meta-heuristic algorithms. In this paper, a discrete pigeon inspired optimizer is proposed to solve MTSP. The proposed algorithm evaluated using five instances from TSPLIB benchmark and compared its results with four algorithms from state-of-the-art. The results indicate that the proposed algorithm outperformed the examined algorithms in most cases.**

***Index Terms*—Multiple Travelling Salesmen Problem, Pigeon Inspired Optimizer, TSPLIB**

I. Introduction

The traveling salesman problem (TSP) is an optimization concept that is aimed at finding the shortest route that visits every city exactly once and back to the starting point. It tries to answer the question that given a list of cities and the distance between two neighboring cities, then what is the shortest route incurs less cost and time [1]. TSP is an NP-hard problem notably due to the complexity of finding the route with the minimum cost possible, as well, as the time takes to arrive at the solution. The majority of the solutions to TSP are obtained using meta-heuristics algorithms notably Ant Colony, Genetic Algorithm, particle swarm optimization, and simulated annealing among others [2]. TSP is applied in a variety of fields including planning, manufacturing, logistics, and transport [1].

Multiple Travelling Salesmen Problem (MTSP) is an extension of the TSP concept whereby several salesmen are expected to visit at least one city once. MTSP stipulates that m salesmen must visit n cities at least once. The objective of MTSP is to minimize the overall cost of the route taken by all the salesmen [3]. Regardless of the individual routes taken by each salesman, the final destination must be the same. The MTSP concept is comparatively harder than TSP in the fact that the former can involve a fixed or non-fixed number of routes and destinations while the latter requires the salesman to use a fixed route to reach a fixed destination [3]. Similar to TSP, MTSP is an NP-complete problem that is solved using evolutionary computing techniques. MTSP is also solved using the same category of algorithms used to solve the TSP approach.

The complex nature of TSP and MTSP makes it extremely difficult to be solved using traditional computing methods, which makes it appropriate to use meta-heuristics algorithms to obtain relevant solutions [4]. Meta-heuristics algorithms are characterized by optimization assumptions and are thus designed to provide a solution to wide range of problems. In other words, meta-heuristics algorithms are not restricted to finding a perfect solution but rather perform optimization to obtain a near-perfect solution [5]. Meta-heuristic algorithms can be placed into several categories includes nature-inspired, population-based, memetic, hybrid, local search-based, and global search-based among others. The majority of meta-heuristic algorithms use popular optimization techniques such as natural selection, natural evolution, cooperation, communication, and neighborhood search [6]. Meta-heuristics algorithms use these concepts to improve the process of building an optimal route with the minimum cost possible.

The rest of this paper is organized as follows: Section II reviews the related works, Section III presents the proposed algorithm for solving MTSP, Section IV discusses the conducted results and Section V concludes the paper.

II. Related Works

Hosseinabadi et al proposed the development of a novel approach that uses a combination of hybrid meta-heuristics algorithms [7]. The framework uses both the Gravitational Emulation Local Search (GELS) and the Genetic Algorithm (GA) to solve the MTSP. The GELS-GA hybrid algorithm uses both the public and local searches to optimize the route selection and cost minimization processes. The experimental setup is aimed at evaluating and comparing the performance levels of GELS-GA against other meta-heuristics algorithms notably Ant colony Optimization (ACO) and Particle Swarm Optimization (PSO). The experimental results indicate that GELS-GA improves the computation time by 11% and route minimization by 27% when used to solve the MTSP.

Xu et al proposed and designed TPHA, which is a two-phase heuristic algorithm used to solve the MTSP [8]. The solution utilizes a combination of two algorithms; a k-means algorithm for grouping cities into clusters and GA for route planning. Further, the approach uses the concept of the roulette wheel in combination with the elitist strategy to optimize the route selection process. The main objective of using the TPHA approach is to optimize the location of visitors using a GPS-enabled mobile guide system. TPHA used improved versions of both the k-means clustering and GA algorithms to implement workload balance. The results are compared with other GA algorithms notably ACO. The TPHA manages to compute the total route taken by tourists to 150.7 Km compared to 159.4 Km for the GA algorithm.

Harrath et al managed to develop a novel hybrid approach known as AC2OptGA that uses a combination of three metaheuristics algorithms to solve the MTSP [9]. The approach uses a modified version of ACO, 2-Opt, as well as the GA. In the proposed approach, ACO is used for the generation of route solutions while 2-Opt is used for enhancing the generated solutions. Additionally, the GA is used to make further quality enhancements for the route solutions. Similar to the GELS-GA algorithm the basis of using three algorithms are motivated by the need to utilize both local and public searches. The experimental results are evaluated using several MTSP benchmarks with the outcome indicating the AC2OptGA outperforms both M-GELS and NMACO.

Modified GELS (M-GELS) is another novel approach that uses the combination of heuristics and concepts of physics to solve the symmetric MTSP. In this context, M-GELS use the local search concepts as well as gravitation, and velocity to improve performance [10]. The performance level of M-GELS depends on how fast the number of possible solutions can be generated. The algorithm uses three main parameters notably the maximum velocity, a radius of the vector space, and the number of iterations used to generate optimal routes. The results from the experimental setup are then compared to other optimization algorithms (ACO and SW+ ASelite). M-GELS produces a 13.7% improvement ration in terms of the distance covered and time taken to generate the required routes.

Yousefikhoshbakht, Didehvar, and Rahmati propose the development of a new and improved ACO (NMACO) that is used to solve the MTSP [11]. The proposed algorithm is aimed at addressing the weaknesses of ACO as evident is the low speed of computation as well as lack of global convergence. NMACO is enhanced using the 2-Opt algorithm, as well as, swap and optimization techniques. Unlike traditional ACO, NMACO utilizes a global search to optimize the process of obtaining the best solution. NMACO also makes it easy to create new lists and update the existing pheromones thus improving performance. The experimental results are then evaluated and compared against other meta-heuristics algorithms with the outcome indicating that NMACO produces superior quality compared to the classic ACO.

The framework developed by Shuai, Yunfeg, and Kai utilizes the combination of mutation and crossover to optimize and solve the MTSP [12]. The approach uses the NSGA-II framework to obtain a working balance between the route taken and the distance covered by a salesman. The algorithm uses the mutation function to maintain the search process within the local search while the crossover function is used to optimize the convergence process. When the performance levels are evaluated using several MTSP benchmarks, it is evident that the NSGA-II framework produces better results compared to state-of-the-art nature-inspired algorithms like ACO, P-ACO, and MACS [11].

In [13] a proposed approach for solving MTSP use modified versions of particle swarm optimization (PSO) algorithm have been introduced. The modified algorithms include APSO and HAPSO whereby the former utilizes the 2-Opt concept while the latter is based on greedy search functionality. The objective is aimed at evaluating the relative performance capabilities between the two modified versions of PSO. The experimental results are further compared with ACO and GA based on five MTSP benchmarks. Eventually, the results indicate that the performance level of HAPSO is slightly superior compared to APSO.

III. Proposed Discrete Pigeon Inspired Optimizer for Solving MTSP

In this section, a proposed discrete pigeon inspired optimizer for solving multiple TSP is introduced. The proposed algorithm considers finding the shortest path for traveling salesmen's overall trips while maintaining a balanced load between all salesmen trips. Pigeon inspired optimizer is a meta-heuristic algorithm inspired by the pigeon homing behavior that has been used in the first and second wars to carry messages. Pigeon inspired optimizer have been used by researchers to solve optimization problems such as air robot path planning [14], target detection [15], image restoration [16], feature selection [17] and many others [18]–[20].

Pigeon homing skills have been derived from two main operators: map and compass, and landmark operators. The homing skills of pigeon come from little magnetic particles located in its peak that give it the ability to find their way around. Also, the pigeon used the altitude of the sun as a compass to adjust its direction.

A. Solution Representation

The solution is represented by a one-dimensional array, where the length of the solution equal to the number of the cities. The value of each index represents the city id. The solution will be divided into multiple parts based on the number of salesmen, where each salesman will take a consecutive part of the solution. All salesmen trips will start from the same source and back to the same starting point. In this way, we assure you that all the salesmen take approximately the same load (number of cities). Fig. 1 shows an example of a solution representation for 10 cities. According to Fig. 1, the tour of salesman 1 is ($source \rightarrow 4 \rightarrow 2 \rightarrow 1 \rightarrow source$), the tour of salesman 2 is ($source \rightarrow 7 \rightarrow 5 \rightarrow 3 \rightarrow source$), and the tour of salesman 3 is ($source \rightarrow 6 \rightarrow 8 \rightarrow 10 \rightarrow 9 \rightarrow source$).

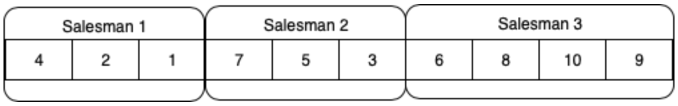

Fig. 1. An example of solution representation.

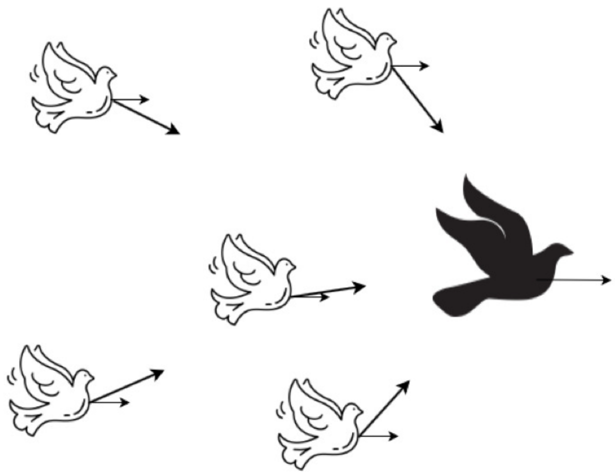

Fig. 2. White pigeons adjust their flying position toward the black one using map and compass operator [17].

B. Fitness Function

The fitness function is used to evaluate the quality of each pigeon (solution). The fitness value is based on the total distance of all salesmen trips. Equation 1 represents the fitness function used in this paper, where S represent the salesman trip, and Equation 2 present the euclidean distance used to calculate the distance between two cities.

$$Fitness\,Function = \sum Distance(S_1, S_2, ...S_n) \quad (1)$$

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

C. Mathematical Model

The mathematical model of Pigeon Inspired Optimizer (PIO) can be expressed by two main operators as follows [14]:

- Map and Compass operator
  Pigeons adjust their direction by relying on the sun altitude and the perception of the shape of the earth as a compass. The rules of this operator are defined by position X_i and velocity V_i of pigeon i. The velocity of each pigeon is updated using Equation 3, while the position is updated using a Sigmoid transfer function since the PIO is designed to solve a continuous problem and MTSP is a discrete problem. The Sigmoid function is used to discretize the PIO algorithm. Fig.2 visualize the map and compass operator, where the black pigeon is the best global pigeon X_g, and all other pigeons move toward it.

$$V_i(t+1) = V_i(t).e^{-Rt} + rand.(X_g - X_i(t)) \quad (3)$$

  where the R is the map and compass factor, and X_g is the position of the global best pigeon, which can be calculated by comparing all pigeon fitness value regarding a target fitness function.
  Based on the Sigmoid value calculated by Equation 4 the position of each pigeon will be updated in D-dimensional space with a percentage of $S(V_i(t))$ from the global best pigeon.

$$S(V_i(t)) = \frac{1}{1 + e^{\frac{-v_i}{2}}} \quad (4)$$

  Fig. 3 illustrates how the pigeon position will be updated in each iteration based on the $S(V_i)$ value. For example, suppose the $S(V_i)$ value for pigeon i were 0.3, this mean that the pigeon i will move toward the global best pigeon X_g by a percentage of 0.3. As Fig. 3 shows that the solution contains 10 cities, then pigeon i will take $0.3 * 10 = 3$ cities for X_g. During the updating process, the city id should be unique in the solution (replication is not allowed). for that, the rest of $X_i(t)$ solution will be taken and double check if one of the city already taken from X_g, then the values will be filled from the mirror part of solution in a consecutive manner as shown in Fig.3.
- Landmark operator
  Pigeons rely on their landmark neighboring when they get close to their destination. To formulate the landmark operator mathematically, the pigeons or solutions ordered in non-increasing order based on their fitness value. Only half the number of the pigeon participates in each iteration to calculate the desirable destination X_c.
  Equation 5 illustrates the number of pigeon in each generation, where t represents the current iteration. Equation 6 presents the desirable destination calculation. All the solutions will be updated or move toward the desired destination X_c as illustrated in Equation 7.

$$Np(t+1) = \frac{Np(t)}{2} \quad (5)$$

$$X_c(t+1) = \frac{\sum X_i(t+1).Fitness(X_i(t+1))}{N_p \sum Fitness(X_i(t+1)} \quad (6)$$

$$X_i(t+1) = X_i(t) + rand.(X_c(t+1) - X_i(t)) \quad (7)$$

IV. Results and Discussion

In this section, the results of the proposed algorithm are presented and compared with the results of the selected algorithms presented in [13]. To evaluate the proposed algorithm, five instances from TSPLIb presented in Table I were chosen, which the same instances used by [13] with their optimal solutions. The first city in each instance was chosen to be the depot city.

The algorithms compared to the proposed discrete PIO are the Genetic Algorithm (GA),ACO ,APSO and HPSO [13]. All the examined algorithms evaluated using a different number of salesmen (two, three and four salesmen).

Table II presents the parameter setting used for PIO. While Table III illustrates the results for all examined algorithms on

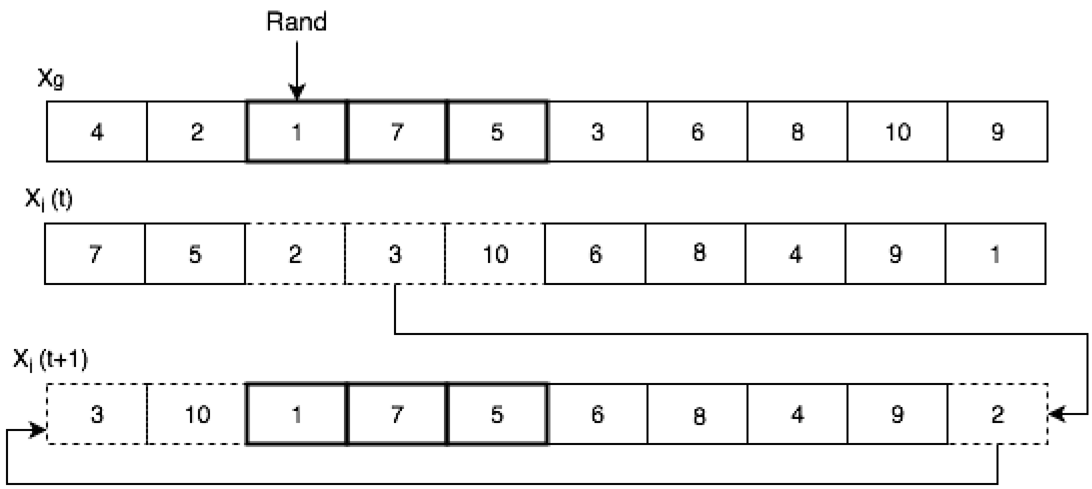

Fig. 3. Example on updating the pigeon position X_i.

TABLE I
TSP INSTANCES FROM TSPLIB [21]

#	Instance	Number of cities	Optimal Solution
1	rat99	99	1211
2	att48	48	10628
3	bier127	127	118282
4	pr76	76	108159
5	berlin52	52	7542

TABLE II
PIO PARAMETERS SETTINGS.

Parameter	Value
Number of Pigeons (Np)	128
Map and Compass Factor (R)	0.09
Number of Iterations	300
Number of Salesmen	(2, 3 and 4)
Initial Velocity for pigeon i (V_i)	Rand (0,1)

five instances from the TSPLIb. According to Table III it is clear that the proposed PIO for solving MTSP outperformed all examined algorithms in most cases. Using two salesmen the PIO achieved the best results on five selected instances, and has results close to optimal especially on "berlin52" were the distance for all two salesmen were 7651 and the optimal solution for this instance is 7542.
For three salesmen experiment, also the PIO achieved the best results among the other examined algorithm, but it is clear that the distance of three salesmen trips is longer than the trips of two salesmen. This can be thought of as a trade-off between the distance and the required time to travel to the cities. So, using more salesmen will increase the overall distance trips but it will cut the time.
The last part of the table presents the results of four salesmen experiment. As the results show that the ant colony optimizer has better results in two instances: "att48" and "berlin52" respectively. While the PIO has better results in the rest instances. Here, we can notice that the overall distance is larger than both experiments three salesmen and two salesmen.

V. CONCLUSION

Multiple traveling salesman problem is an extended problem of the traditional traveling salesman problem. Where the scenario consists of more than one salesman that has to travel over many cities with the lowest cost as possible. In this paper, a discrete PIO algorithm is proposed to solve the multiple salesmen problem. The PIO algorithm mimics the homing skill of pigeon for finding their homeland. The proposed algorithm evaluated and compared with four algorithms from state-of-the-art works namely GA, APSO, HPSO, and ACO. All the examined algorithms used to solve five instances from the TSPLIB using several numbers of salesmen (three, four and five). The proposed algorithm outperforms all other examined algorithms in most TSPLIB instances and scenarios.

REFERENCES

[1] K. L. Hoffman, M. Padberg, G. Rinaldi *et al.*, "Traveling salesman problem," *Encyclopedia of operations research and management science*, vol. 1, pp. 1573–1578, 2013.
[2] H. A. Abdulkarim and I. F. Alshammari, "Comparison of algorithms for solving traveling salesman problem," *International Journal of Engineering and Advanced Technology*, vol. 4, no. 6, pp. 76–79, 2015.
[3] T. Bektas, "The multiple traveling salesman problem: an overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209–219, 2006.
[4] D. Gupta, "Solving tsp using various meta-heuristic algorithms," *International Journal of Recent Contributions from Engineering, Science & IT (iJES)*, vol. 1, no. 2, pp. 22–26, 2013.
[5] E. Çela, V. G. Deineko, and G. J. Woeginger, "The multi-stripe travelling salesman problem," *Annals of operations research*, vol. 259, no. 1-2, pp. 21–34, 2017.
[6] M. Ondřej, "Comparison of metaheuristic methods by solving travelling salesman problem," in *Proceedings of the 9th International Scientific Conference INPROFORUM: Common challenges-Different solutions-Mutual dialogue*. Jihočeská univerzita v Českỳch Budějovicích, 2015.
[7] A. A. Hosseinabadi, M. Kardgar, M. Shojafar, S. Shamshirband, and A. Abraham, "Gels-ga: hybrid metaheuristic algorithm for solving multiple travelling salesman problem," in *2014 14th International Conference on Intelligent Systems Design and Applications*. IEEE, 2014, pp. 76–81.

TABLE III
EXPERIMENTAL RESULTS

Number of Salesmen	Instance	GA	Ant Colony	APSO	HPSO	PIO
2 Salesmen	rat99	2487.64	3980.43	1662	1427	**1321.43**
	att48	50725.81	71151.36	43424	36891	**31748.11**
	bier127	282343.86	425016.29	155651	125480	**124532.7**
	pr76	184176.1	309514.32	143560	120490	**117432.20**
	berlin52	11066.69	16354.02	9482	7994	**7651**
3 Salesmen	rat99	1970.48	3328.02	2018	1680	**1320.67**
	att48	49709.78	51032.81	51510	40637	**33812.43**
	bier127	257228.63	349770.9	187575	129621	**125632**
	pr76	170857.76	265550.49	174485	138096	**121530.4**
	berlin52	10898.75	11726.73	11149	8876	**7995.32**
4 Salesmen	rat99	1945.36	2814.74	2353	1963	**1803.5**
	att48	47083.53	**42169.88**	58445	67170	43654.13
	bier127	233708.3	294273.77	207814	141496	**140732**
	pr76	168717.69	207333.11	201022	159964	**143453.98**
	berlin52	11736.74	**8820.24**	12952	13602	9651.83

[8] X. Xu, H. Yuan, M. Liptrott, and M. Trovati, "Two phase heuristic algorithm for the multiple-travelling salesman problem," *Soft Computing*, vol. 22, no. 19, pp. 6567–6581, 2018.

[9] Y. Harrath, A. F. Salman, A. Alqaddoumi, H. Hasan, and A. Radhi, "A novel hybrid approach for solving the multiple traveling salesmen problem," *Arab Journal of Basic and Applied Sciences*, vol. 26, no. 1, pp. 103–112, 2019.

[10] A. S. Rostami, F. Mohanna, H. Keshavarz, and A. A. R. Hosseinabadi, "Solving multiple traveling salesman problem using the gravitational emulation local search algorithm," *Applied Mathematics & Information Sciences*, vol. 9, no. 3, pp. 1–11, 2015.

[11] M. Yousefikhoshbakht, F. Didehvar, and F. Rahmati, "Modification of the ant colony optimization for solving the multiple traveling salesman problem," *Romanian Journal of Information Science and Technology*, vol. 16, no. 1, pp. 65–80, 2013.

[12] Y. Shuai, S. Yunfeng, and Z. Kai, "An effective method for solving multiple travelling salesman problem based on nsga-ii," *Systems Science & Control Engineering*, vol. 7, no. 2, pp. 108–116, 2019.

[13] S. D. Gulcu and H. K. Ornek, "Solution of multiple travelling salesman problem using particle swarm optimization based algorithms," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 7, no. 2, pp. 72–82, 2019.

[14] H. Duan and P. Qiao, "Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning," *International Journal of Intelligent Computing and Cybernetics*, vol. 7, no. 1, pp. 24–37, 2014.

[15] C. Li and H. Duan, "Target detection approach for uavs via improved pigeon-inspired optimization and edge potential function," *Aerospace Science and Technology*, vol. 39, pp. 352–360, 2014.

[16] H. Duan and X. Wang, "Echo state networks with orthogonal pigeon-inspired optimization for image restoration," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 11, pp. 2413–2425, 2015.

[17] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer," *Expert Systems with Applications*, p. 113249, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417420300749

[18] S. Zhang and H. Duan, "Gaussian pigeon-inspired optimization approach to orbital spacecraft formation reconfiguration," *Chinese Journal of Aeronautics*, vol. 28, no. 1, pp. 200–205, 2015.

[19] H. Qiu and H. Duan, "Multi-objective pigeon-inspired optimization for brushless direct current motor parameter design," *Science China Technological Sciences*, vol. 58, no. 11, pp. 1915–1923, 2015.

[20] Y. Deng and H. Duan, "Control parameter design for automatic carrier landing system via pigeon-inspired optimization," *Nonlinear Dynamics*, vol. 85, no. 1, pp. 97–106, 2016.

[21] G. Reinelt, "Tsplib—a traveling salesman problem library," *ORSA journal on computing*, vol. 3, no. 4, pp. 376–384, 1991.