©2019 Published in 7th International Symposium on Innovative Technologies in Engineering and Science 22-24 November 2019 (ISITES2019 SanliUrfa - Turkey)
https://doi.org/10.33793/acperpro.02.03.31

A New Approach Based on K-means Clustering and Shuffled Frog Leaping Algorithm to Solve Travelling Salesman Problem

[1]Murat KARAKOYUN
[1]Faculty of Engineering, Department of Computer Engineering, Necmettin Erbakan University, Turkey

Abstract

The Travelling Salesman Problem (TSP), which is a combinatorial NP-hard problem, aims to find the shortest possible path while visiting all cities (only once) in a given list and returns to the starting point. In this paper, an approach, which is based on k-means clustering and Shuffled Frog Leaping Algorithm (SFLA), is used to solve the TSP. The proposed approach consists of three parts: separate the cities into *k* clusters, find the shortest path for each cluster and merge the clusters. Experimental results have shown that the algorithm get better results as the number of cluster increase for problems that have a large number of cities.

Key words: Shuffled Frog Leaping Algorithm, K-means Clustering, Travelling Salesman Problem

1. Introduction

The TSP, which is a combinatorial NP-hard problem and one of the mostly studied optimization problems, was firstly formulated in 1930. The aim of the problem is to find the shortest possible path which is starts and ends with same city, provided that visiting all cities only once. Although, TSP is generally used as a benchmark problem of the optimization to analyze the performance of the algorithms, it is applied on many real world problems [1-3]. It can be seen that there are many researches and works about TSP in literature. Especially in last years, many meta-heuristic algorithms were applied on the specified problem. Chaudhari and Thakkar [4] applied ant colony optimization (ACO), firefly algorithm (FA), particle swarm optimization (PSO) and genetic algorithm (GA) on the travelling salesman problem. The performance comparison of the algorithms was done and they stated that ACO and GA performed better than the other algorithms. Hatamlou [5] worked on the TSP to compare the performance of black hole (BH) algorithm with other metaheuristic algorithms' (ACO, PSO and PSO) performances. Based on the experimental results, it is stated that BH algorithm gives better results than other algorithms. Besides, BH is faster in terms of run time. Saji and Riffi [6] compared the performance of bat algorithm (BA) with traditional and metaheuristic algorithms on travelling salesman problem. Gupta et al. [7] proposed a discrete and greedy version of whale optimization algorithm (WOA) to solve the travelling salesman problem. They stated that the proposed algorithm (GWOA) is more prominent then the other algorithms – standard WOA and GA – with its successful results.

In this paper, a new approach based on clustering is proposed to solve the TSP. The motivation of this study is to show the effect of the clustering approach on the travelling salesman problem. For this purpose, k-means algorithm was used as clustering algorithm and shuffled frog leaping

*Corresponding author: Address: Faculty of Engineering, Department of Computer Engineering, Necmettin Erbakan University, Konya, Turkey. E-mail address: mkarakoyun@erbakan.edu.tr, Phone: +905423245453

algorithm was used to search the shortest path for each cluster. As last step, Prim algorithm was used to merge the clusters.

The rest of this paper is organized as follows: Firstly, the definition of the TSP, basic steps of the k-means algorithm, the details of the SFLA, an abstract of the Prim algorithm and a review of the proposed approach were done in Section 2. Then, the experimental results were presented in Section 3. Finally, a discussion of the paper was done in Section 4.

2. Materials and Method

In this paper, the TSP is discussed as a problem to be solved. To solve the problem, an approach based on clustering was proposed.

2.1. The travelling salesman problem

The TSP is a combinatorial optimization problem which has a salesman has to visit all cities only once and back to starting city. Since there is no algorithm to solve the TSP in polynomial time, it is a problem that belongs to NP-hard problem class. The minimum expected time to achieve shortest path is exponential. The purpose of the salesman is to find an answer for the following question: "For a given list of cities, what is the shortest possible tour that visits all cities only once and returns to the origin city?" [2, 7-9]

The TSP can be represented as a connected graph $G = (N, E)$, where vertices *(N)* of the graph represent the set of n cities must be visited, and the edges *(E)* represent the paths which connecting all cities. Besides, an edge's length is the path's distance. The distance between ith and jth cities d_{ij} is usually calculated by Euclidean distance. Eq. (1) represents the mathematical formula of the distance an edge $(i, j) \in E$ [5-8].

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

In this paper, five benchmark problems with different number of cities were used. The datasets were taken from [10]. The details about the benchmark problems, which were used in this work, were presented in Table 1.

Table 1. Benchmark problems from TSP library

Problem	# City	Opt. tour length
Berlin52	52	7542
Ch130	130	6110
Tsp255	255	3919
Pcb442	442	50784
Pr1002	1002	259045

2.2. K-means clustering algorithm

K-means [11] is one of the most popular algorithms for non-hierarchical clustering problem. The purpose of the algorithm is to separate *N* data objects into *k* cluster. In k-means algorithm, the data objects are clustered according to the distance between each other. K-means works according to the traditional clustering dialectic, and so that each data object can belong to only one cluster. K-means algorithm tries to find the centroids which have minimum distance with the data objects belong to them [12-14].

The algorithm starts with a random initialization of centroids. Then, for each data object, the nearest cluster is found and the object assigned to that cluster. After assigning of all data objects to the nearest cluster is done, the position of the all centroids is calculated by using new members of clusters. The assigning of the objects to the clusters and updating of the position of the clusters are applied in a loop. Algorithm stops when there is no more change for centroids or in case of finishing criteria is provided [12, 14].

The following four steps organized by Velmurugan to summarize the k-means algorithm [15]:

1. Initialize *k* centroids randomly within the solution space.
2. Assign each data object to the nearest cluster.
3. Update the position of the centroids according to the new members.
4. Repeat steps 2 and 3 until finishing criteria is provided.

2.3. Shuffled frog leaping algorithm

The SFLA, which is a population based, memetic meta-heuristic and iterative optimization algorithm, was first proposed to solve the discrete problems by Eusuff et al. The algorithm was modelled based on social behaviors of the frog population. In social life of frogs, the main goal is getting maximum food by using minimum moves [16, 17].

Each frog in population represents a possible solution to the current problem and the fitness value of that solution means the closeness of the frog to the food. The SFLA starts with a randomly generated population. Then, according to the objective function which is specified for the optimization problem, the fitness value of the each member is calculated and the population is sorted by the fitness value from best to the worst. After the sorting process, the population is separated into memeplexes and a memetic evolutionary process is done for each memeplex. In memetic evolutionary, the aim is to improve the position of the worse members by using the local best or global best. After memetic evolutionary process is done, the memeplexes are gathered and shuffled. If the finishing criterion is provided, the algorithm returns the best position. Otherwise, the population is sorted and separated into memeplexes for the next iteration [16-19]. A flowchart of the SFLA is given in Fig. (1).

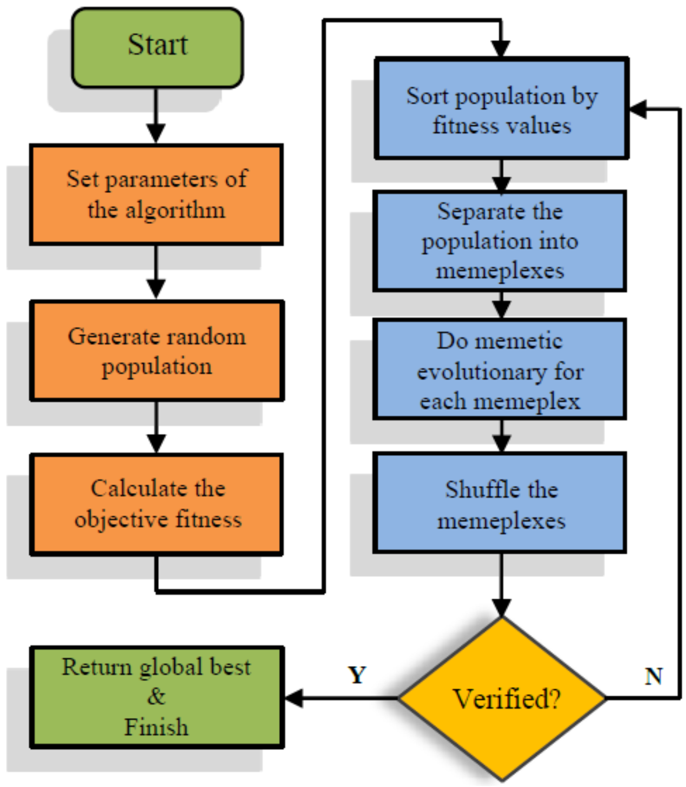

Figure 1. The flowchart of the SFLA

2.4. Prim algorithm

Prim algorithm [20], which was first proposed in 1930 by Jarnik, is a greedy algorithm that tries to find a minimum spanning tree (MST) in a weighted undirected graph. The algorithm finds a subset of the edges which includes all vertexes with a minimized cost of the total weight. The algorithm starts with a random (or arbitrary) vertex and adding the vertex which has lowest cost at each step. In later times, the algorithm improved by Prim in 1957 [21] and then by Dijkstra in 1959 [22].

2.5. The proposed algorithm

In this paper, an approach, which is based on k-means clustering and shuffled frog leaping algorithm, is used to solve the TSP. The proposed approach consists of three parts: separate the cities into *k* cluster, find the shortest path for each cluster and merge the clusters. The flowchart of the proposed algorithm is given in Fig. (2).

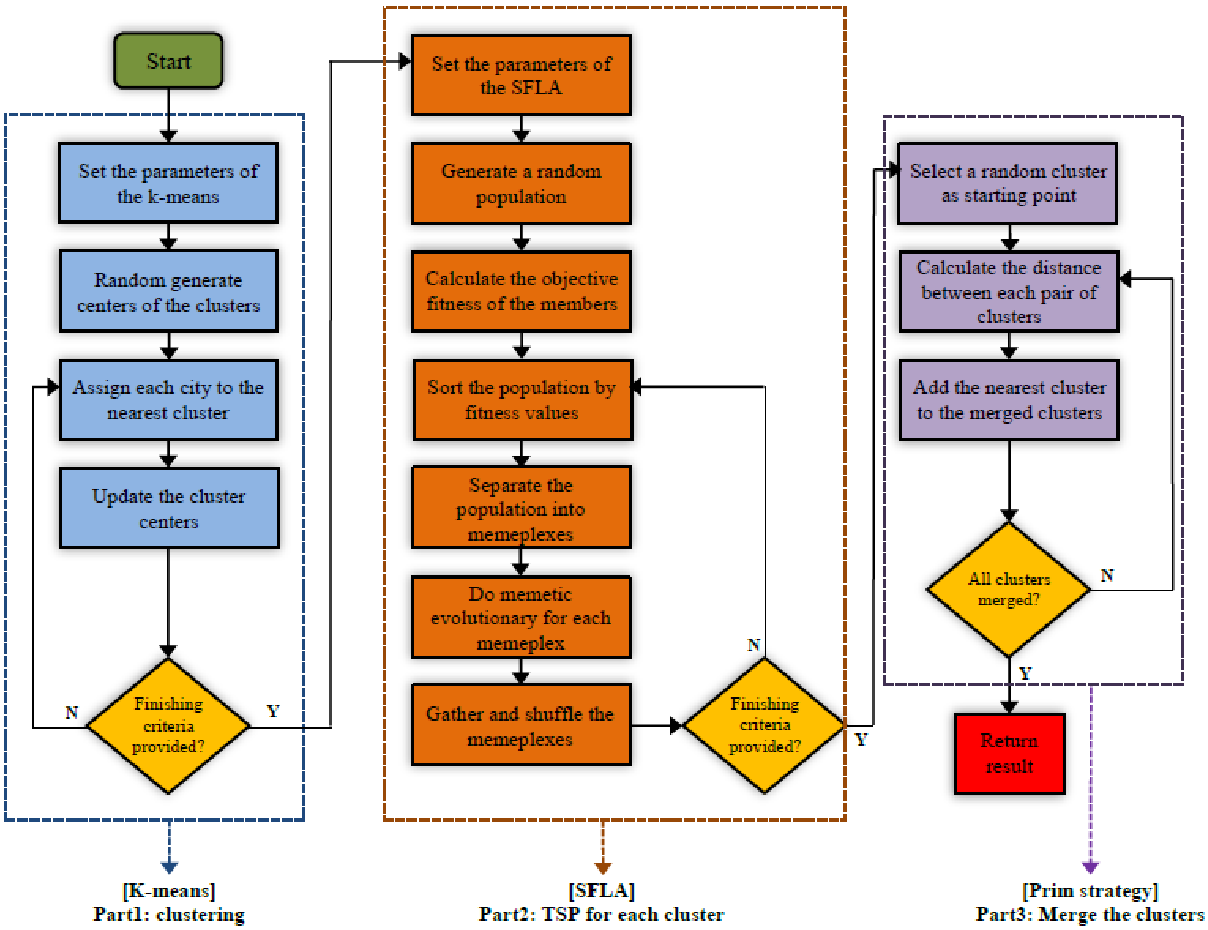

Figure 2. The flowchart of the proposed algorithm

3. Results

In this paper, a cluster based approach applied on five benchmark problems of the TSP. Different number of cluster ($k = 1, 2, 4, 8$) was tried and the results which obtained in different cluster number were compared. Iteration number for k-means algorithm was chosen as 100. For each problem the algorithm worked 10 times. The parameters and the value of the parameters of the SFLA are given in Table 2.

Table 3 shows the experimental results of the proposed algorithm. According to the results in Table 3, it can be seen that when the number of clusters increases, the success of the algorithm decreases in the problems with low number of cities. On the other hand, in problems with a large number of cities, increasing the number of clusters increases the success of the algorithm. Based on these results, it can be said that it may be beneficial to apply the clustering approach to the problems with large number of cities for travelling salesman problem.

Table 2. Parameters of the SFLA

Parameter	Sign	Value
Population	F	100
Memeplex number	m	10
Member of memeplex	n	10
Member of sub-memplex	q	5
Iteration number	I	250
Memetic evolutionary iteration	N	5
Maximum step size (%)	Smax	45

Table 3. The results of the proposed algorithm in different *k* values for 10 runs

Problem	k	Best	Worst	Average	Best Known
Berlin52	1	7742	8338	**8000**	7542
	2	7847	8244	8073	
	4	8407	9605	8960	
	8	10262	11324	10799	
Ch130	1	7593	11124	9801	6110
	2	6496	7099	**6748**	
	4	6648	6959	6766	
	8	6946	7777	7514	
Tsp255	1	16395	17922	17129	3919
	2	5759	6277	5975	
	4	4244	4466	**4364**	
	8	4339	4493	4421	
Pcb442	1	454009	501034	475524	50784
	2	234344	261769	250486	
	4	78015	86485	81529	
	8	59400	63088	**60882**	
Pr1002	1	4753163	4923296	4834934	259045
	2	2782208	2960086	2887670	
	4	1387717	1460742	1420568	
	8	535591	633299	**578593**	

4. Conclusion

In this paper, an approach based on k-means clustering algorithm and shuffled frog leaping algorithm was proposed for travelling salesman problem. The experimental results showed that, the clustering approach can be useful in the problem with large number of cities. For next studies, other metaheuristic algorithms which have been proposed for the combinatorial problems can be applied with clustering approach. On the other hand, parallel programing can be used to minimize run time of the system.

References

[1] A. H. Halim and I. Ismail, "Combinatorial optimization: comparison of heuristic algorithms in travelling salesman problem," *Archives of Computational Methods in Engineering*, vol. 26, pp. 367-380, 2019.

[2] C. K. Joshi, T. Laurent, and X. Bresson, "An Efficient Graph Convolutional Network Technique for the Travelling Salesman Problem," *arXiv preprint arXiv:1906.01227*, 2019.

[3] S. S. Juneja, P. Saraswat, K. Singh, J. Sharma, R. Majumdar, and S. Chowdhary, "Travelling Salesman Problem Optimization Using Genetic Algorithm," in *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019, pp. 264-268.

[4] K. Chaudhari and A. Thakkar, "Travelling Salesman Problem: An Empirical Comparison Between ACO, PSO, ABC, FA and GA," in *Emerging Research in Computing, Information, Communication and Applications*, ed: Springer, 2019, pp. 397-405.

[5] A. Hatamlou, "Solving travelling salesman problem using black hole algorithm," *Soft Computing*, vol. 22, pp. 8167-8175, 2018.

[6] Y. Saji and M. E. Riffi, "A novel discrete bat algorithm for solving the travelling salesman problem," *Neural Computing and Applications*, vol. 27, pp. 1853-1866, 2016.

[7] R. Gupta, N. Shrivastava, M. Jain, V. Singh, and A. Rani, "Greedy WOA for Travelling Salesman Problem," in *International Conference on Advances in Computing and Data Sciences*, 2018, pp. 321-330.

[8] E. L. Lawler, J. K. Lenstra, A. H. Rinnooy Kan, and D. B. Shmoys, "The traveling salesman problem; a guided tour of combinatorial optimization," 1985.

[9] V. Tongur and E. Ülker, "The analysis of migrating birds optimization algorithm with neighborhood operator on traveling salesman problem," in *Intelligent and Evolutionary Systems*, ed: Springer, 2016, pp. 227-237.

[10] TSPLIB. (2019, 02 Sep.). *TSP Library Benchmarks*. Available: http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html

[11] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 1967, pp. 281-297.

[12] M. Capó, A. Pérez, and J. A. Lozano, "An efficient approximation to the K-means clustering for massive data," *Knowledge-Based Systems*, vol. 117, pp. 56-69, 2017.

[13] P. Arora and S. Varshney, "Analysis of k-means and k-medoids algorithm for big data," *Procedia Computer Science,* vol. 78, pp. 507-512, 2016.

[14] M. KARAKOYUN, A. SAGLAM, N. A. BAYKAN, and A. A. ALTUN, "Non-locally color image segmentation for remote sensing images in different color spaces by using data-clustering methods," *image,* vol. 10, p. 11, 2017.

[15] T. Velmurugan, "Performance based analysis between k-Means and Fuzzy C-Means clustering algorithms for connection oriented telecommunication data," *Applied Soft Computing*, vol. 19, pp. 134-146, 2014.

[16] M. M. Eusuff and K. E. Lansey, "Optimization of water distribution network design using the shuffled frog leaping algorithm," *Journal of Water Resources planning and management,* vol. 129, pp. 210-225, 2003.

[17] M. Karakoyun and A. Babalik, "Data clustering with shuffled leaping frog algorithm (SFLA) for classification," in *International Conference on Intelligent Computing, Electronics Systems and Information Technology (ICESIT 2015)*, 2015, pp. 25-26.

[18] V. K. Jonnalagadda and V. K. D. MALLESHAM, "Bidding strategy of generation companies in a competitive electricity market using the shuffled frog leaping algorithm," *Turkish Journal of Electrical Engineering & Computer Sciences,* vol. 21, pp. 1567-1583, 2013.

[19] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization," *Engineering optimization*, vol. 38, pp. 129-154, 2006.

[20] V. Jarník, "O jistém problému minimálním," *Práca Moravské Prírodovedecké Spolecnosti,* vol. 6, pp. 57-63, 1930.

[21] R. C. Prim, "Shortest connection networks and some generalizations," *The Bell System Technical Journal*, vol. 36, pp. 1389-1401, 1957.

[22] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik,* vol. 1, pp. 269-271, 1959.