Sigma J Eng & Nat Sci 37 (4), 2019, 1421-1434

Sigma Journal of Engineering and Natural Sciences
Sigma Mühendislik ve Fen Bilimleri Dergisi

Research Article

TESTING OF TREE-SEED ALGORITHM ON P-MEDIAN BENCHMARK PROBLEMS•

İbrahim Miraç ELİGÜZEL[1], Eren ÖZCEYLAN*[2], Cihan ÇETİNKAYA[3]

[1] *Gaziantep University, Department of Industrial Engineering, GAZIANTEP;* ORCID: 0000-0003-3105-9438
[2] *Gaziantep University, Department of Industrial Engineering, GAZIANTEP;* ORCID: 0000-0002-5213-6335
[3] *Adana Alparslan Türkeş Science and Technology University, Department of Management Information Systems, ADANA;* ORCID: 0000-0002-5899-8438

Received: 20.04.2019 Revised: 16.07.2019 Accepted: 13.09.2019

ABSTRACT

This paper presents an application of tree-seed algorithm (TSA) -which is based on the relation between trees and their seeds- on the P-median benchmark problems. To the best knowledge of the authors, this is the first study which applies TSA to the P-median problem. In this paper, different P-median problem instances are generated to show the applicability of the TSA. The experimental results are compared with the optimal results obtained by GAMS-CPLEX. Also, TSA is applied on data sets from OR-Library, and then the obtained and known optimal results are compared. The comparisons demonstrate that the TSA can find optimal and near-optimal values for the small and medium-sized problems, respectively.
Keywords: Location and allocation, meta-heuristic, P-median problem, tree-seed algorithm.

1. INTRODUCTION

The P-median problem is one of the well-known problems in facility location problem types. Numerous versions of the P-median problem are defined in the literature and it has been shown that, it belongs to the class of NP-hard problems [1]. To find acceptable solutions for the P-median problem in a reasonable time, genetic algorithm [2]; simulated annealing [3]; ant colony optimization[4]; tabu search algorithm [5]; artificial bee colony [6] and particle swarm optimization [7] are applied by the researchers. Besides aforementioned well-known meta-heuristic approaches, there have been also other approaches, which can be applied or tested for facility location problems. One of the approaches includes improvement, such as genetic algorithm proposed by Alp et al. [8]. With respect to the study, modified genetic algorithm has desired properties such that simplicity, generating solutions and being fast by improving decision process of population size and crossover part. In the study, proposed algorithm consists of three main components which are sub-gradient column generation, core heuristic and aggregation

* Corresponding Author: e-mail: erenozceylan@gmail.com, tel: (342) 317 26 18
• This paper is a revised and expanded version of a paper entitled 'Application of Tree-Seed Algorithm to the P-Median Problem' presented at the International Conference on Artificial Intelligence towards Industry 4.0 in İskenderun, Turkey, 15–17 November 2018.

heuristic [9]. Another study that utilizes method consists of genetic algorithm with two dimensional representation and it manages to reduce the cost by 40.55% in the proposed case [10]. One of the example for application of P-median problem in real life is published by Afify et al. [11]. In this aspect, another application can be given as planning of emergency service backup systems in o rder to provide the response time and service quality [12]. The main contribution of this study is giving an evolutionary learning technique that provides receiving optimal or near optimal solution in a short time for facility location problems under disruption.

TSA is one of the recent nature based meta-heuristic approaches that is proposed by Kiran (2015). There are only a few papers in the literature which uses TSA such as; structural damage identification problem [14], li-ion battery parameter problem [15], power system problem [16] and uncapacitated facility location problem [17].

In this paper, TSA is applied to the P-median problem for the first time in the literature. Aim of this paper is to demonstrate the applicability of TSA on P-median problem. Therefore, from the gathered results, it can be said that applying further improvements and modifications on TSA makes it possible to reach optimal solution or near optimal solution in affordable time. Our motivation in this study is to apply a novel approach to solve P-median problem which is Tree-seed algorithm. The main reason behind our motivation is that TSA has not been applied on P-median before. The rest of the paper is structured as follows. Mathematical model of the P-median problem is given which is utilized as fitness function in TSA in Section II. The steps of TSA are presented in Section III including population generation and general behavior of TSA in solution space. Computational results of TSA on several P-median data sets are shown in Section IV and finally, conclusion and suggestions for further research are given in Section V.

2. P-MEDIAN PROBLEM

Facility location problems can be classified; according to their objective functions (minisum, minimax, covering constraints); time horizon as static and dynamic; hierarchy in facilities and; containing stochastic elements as probabilistic and deterministic [18]. In Figure 1, classification of facility location problems is demonstrated.

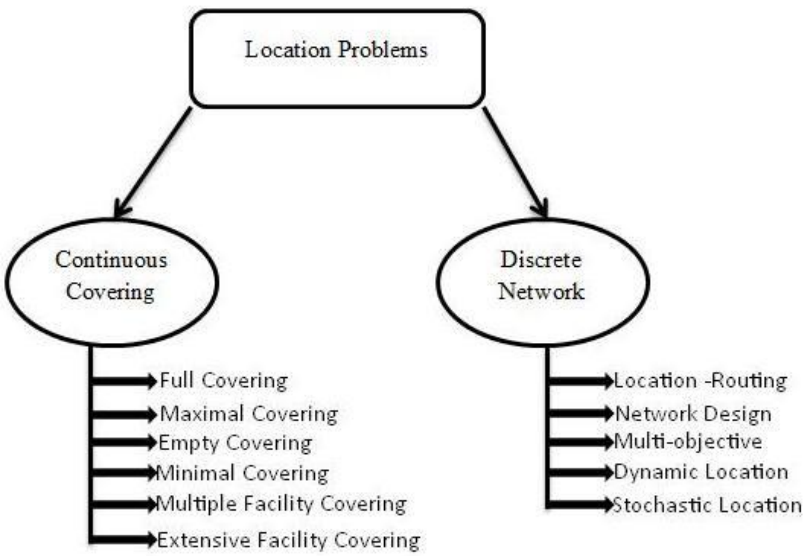


Figure 1. Classification of location problems [19; 20]

The P-median problem is a network problem that was originally designed and has been extensively applied to facility location problems [21]. The P-median problem seeks to define the number of *p* candidate facility (source nodes) to be opened, and which customers (nodes) will be assigned to each facility while minimizing the total distance/cost [22]. The problem formulation is given as follows [23]:

Decision variables:

$$y_k = \begin{cases} 1, if\ source\ node\ k\ is\ selected\ (\forall k \in K) \\ 0, otherwise \end{cases}$$

$$x_{ik} = \begin{cases} 1, if\ demand\ node\ i\ is\ assigned\ to\ source\ node\ k\ (\forall i \in K, \forall k \in K) \\ 0, otherwise \end{cases}$$

Parameters:

d_{ik} = Distance between source node *k* and demand node *i*
p = Number of source node to be opened

Objective function:

$$Min\ Z = \sum_{i \in I} \sum_{k \in K} d_{ik} x_{ik} \quad \text{(Eq. 1)}$$

Subject to

$$\sum_{k \in K} x_{ik} = 1 \qquad \forall i \in K \quad \text{(Eq. 2)}$$

$$x_{ik} \le y_k \qquad \forall i \in K, \forall k \in K \quad \text{(Eq. 3)}$$

$$\sum_{k \in K} y_k = p \quad \text{(Eq. 4)}$$

$$y_k, x_{ik} \in \{0, 1\} \qquad \forall i \in K, \forall k \in K \quad \text{(Eq. 5)}$$

The objective function (1) is to minimize total distance. Constraint (2) provides the assignment of each demand node to a source node, while Constraint (3) provides the assignment of demand nodes to the opened source nodes. Constraint (4) determines the number of source nodes which should be opened. Constraint (5) is the constraint of the decision variables.

3. THE TREE-SEED ALGORITHM

TSA is the population based meta-heuristic algorithm which is inspired from the trees and their seeds. Initialization, seed production mechanism and replacement procedure are the main parts that the algorithm contains. In the initialization part, trees are placed in the search space and their fitness values are calculated. After the initialization part, seed number for each tree is decided and comparison between seeds and tree is done with respect to their fitness values. In the last part, comparison is done between the trees and their seeds according to the fitness values. After comparison, if the seed has a better fitness value, seed takes the place of parent tree. Otherwise, parent tree preserves its place. Another aspect is the parameters that required to be decided according to problem type, parameters can be expressed as *ST* (control parameter between [0-1] interval), *NS* (number of seeds produced for tree), *D* (dimensionality of problem) and *max_FEs* (referred as iteration) [17]. *NS* and *ST* are used for convergence and exploitation purposes. It can be summarized as "when *ST* is close to *1*, the algorithm provides a fast convergence and this is useful for lower dimensional optimization problems. If *NS* is a higher number, the effective local search is provided" [17]. Following flow chart (Figure 2) demonstrates the working principle of the TSA.

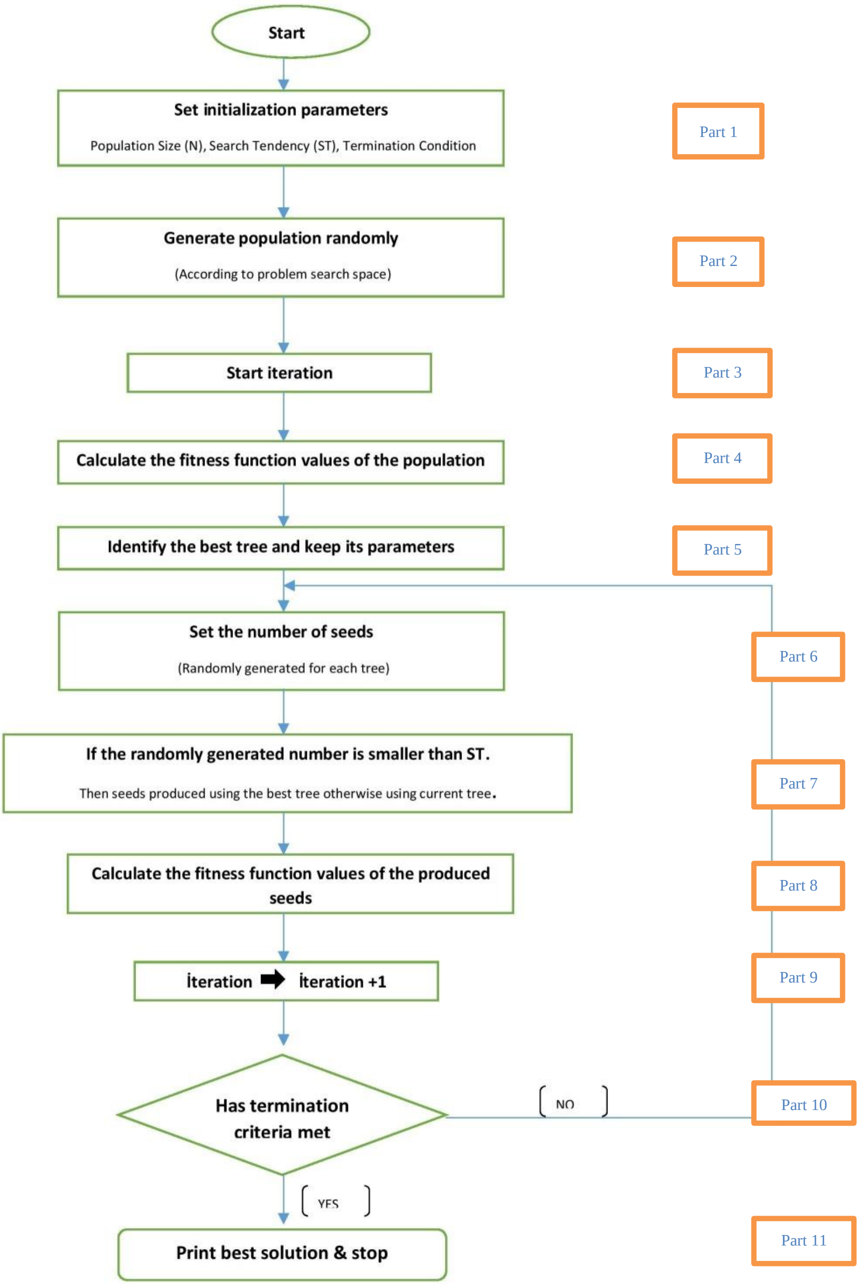


Figure 2. TSA flow chart [17]

First two parts in the flow chart belongs to the initialization phase. In that process, trees are generated randomly in solution space and parameters are settled according to the problem type.

Following five parts belongs to the seed production mechanism phase, which includes working procedure of seed production and selecting best seeds to be tree in the next generation. The last parts consist of replacing the seed that has a better value than its parent.

4. COMPUTATIONAL ANALYSIS

In order to investigate the performance and accuracy of applied TSA, 10 P-median problems with different node sizes are generated. All the test problems are coded and solved in MATLAB 2018b. To make a comparison, the test problems are also run by using GAMS-CPLEX optimization package program. The parameters of TSA ST, NS and *max_FEs* are decided as 0.1, 100 and 100000 respectively. All of the runs are conducted on a server with 2.7 GHz Intel Core processor and 2 GB of RAM. The results of two applications are given in Table 1. First three columns in Table 1 show the problem name, size and the number of *p*, while the other four columns show the objective function values and computation times required for GAMS-CPLEX and TSA, respectively. The gap between two objective functions values are given in the last column. After the test application of TSA on generated data, TSA is applied on the forty Pmed data sets from OR-Library and comparison is made in the same way with the randomly generated data set. Results for Pmed data sets are demonstrated in Table 2. The gap is the percentage of the deviation from the optimum solution for the obtained best solution from the algorithm and it is calculated as follows:

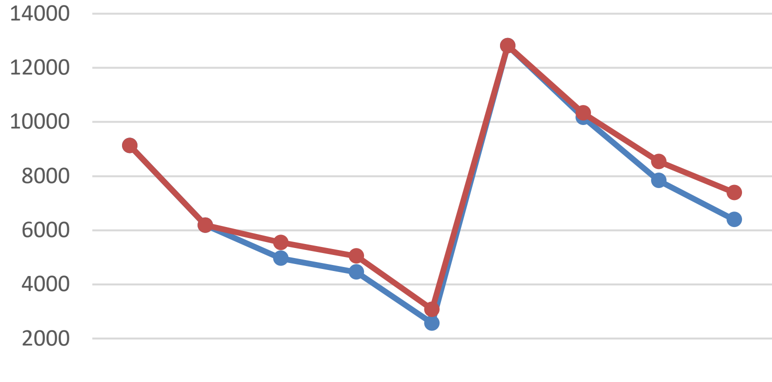
$$Gap = \frac{f(sol)-f(opt)}{f(opt)} x100 \quad \text{(Eq. 6)}$$

where *f(opt)* is the optimal solution obtained by GAMS-CPLEX, *f(sol)* is the best solution obtained by the TSA at the end of the run. In Table 1, the objective function values obtained by GAMS-CPLEX are optimal. According to the Table 1, TSA can find the optimal solutions in 3 problems. However, the optimal solutions cannot be obtained in the rest 7 problems. The gap values between two solutions are shown in the last column of Table 1. As can be seen, 23.56% gap value which is found in last test problem is the maximum value. In Table 2, it is illustrated that TSA is applicable for uncapacitated P-median problems. From the Table 2, it can be seen that application of TSA on small and medium size data sets is more effective and efficient. For the big data sets, deviation from the optimal solution gradually increases. In addition, the performance of TSA is inversely correlated with number of the medians. For the data sets that include five medians, TSA find the optimal and near optimal solutions. However, with the increase in number of medians, deviation from optimal solution inclines.

Table 1. Results of the TSA and GAMS-CPLEX

Problem	# of nodes	*p*	GAMS-CPLEX		Tree-seed algorithm		Gap
			Obj. Func. Value	CPU Time	Obj. Func. Value	CPU Time	
Pm1	100	5	9121	0.62	9121	1.61	0.00
Pm2	100	10	6186	0.88	6186	162.16	0.00
Pm3	100	15	4964	0.93	5544	224.01	11.68
Pm4	100	20	4456	1.00	5047	170.98	13.26
Pm5	100	25	2571	1.00	3080	106.12	19.80
Pm6	200	5	12809	6.24	12809	33.65	0.00
Pm7	200	10	10163	7.75	10324	605.17	1.58
Pm8	200	15	7830	2.30	8537	147.56	9.03
Pm9	200	20	6389	4.19	7388	598.67	15.64
Pm10	200	25	5182	2.63	6403	201.60	23.56

In Figure 3, the TSA and GAMS-CPLEX results are compared. The data-set which is utilized for test on TSA starts with small sized data and then, gradually become bigger accordingly Pm1 to Pm10. The GAMS-CPLEX results are the optimum ones. Therefore, from the Figure 3, it can be said that TSA algorithm is affective for small and medium sized data-sets. However, for the large sized data sets, deviation from the optimum solution rapidly increases.



	1	2	3	4	5	6	7	8	9
GAMS-CPLEX	9121	6186	4964	4456	2571	12809	10163	7830	6389
TSA	9121	6186	5544	5047	3080	12809	10324	8537	7388

GAMS-CPLEX TSA

Figure 3. Comparison results

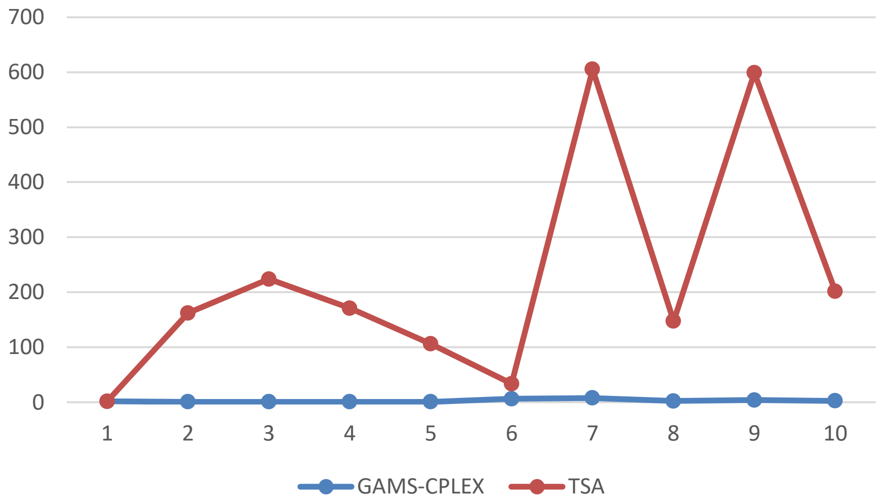


Figure 4. CPU time comparison

The CPU time for both GAMS-CPLEX and TSA is given in Figure 4. From the Figure 4, it can be realized that GAMS-CPLEX follows straight line. However, for TSA, line decreases for data-sets with low number of medians and shows a behavior of climbing when the number of medians begins to increase. Data-sets which are utilized in this study can be downloaded from http://ibs.gantep.edu.tr/duyuru/files/articles/pmed-matrisler12669.rar.

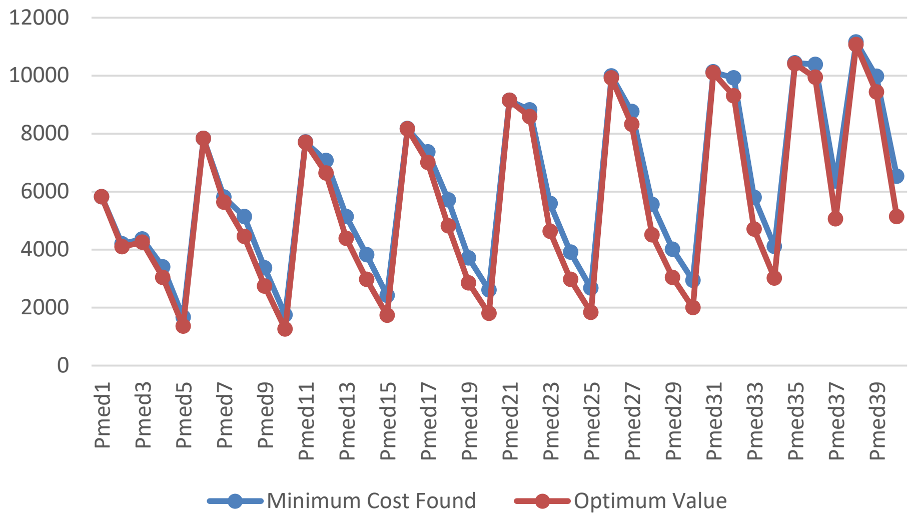


Figure 5. Results of comparison with OR-Library data sets

In Figure 5, it is shown that the results from application of TSA on OR-Library data sets and optimal solutions for these data sets retrieved from OR-Library. From the Figure 5, it can be said that the Gap increases with the increase in number of medians. Also, it can be seen that the Gap is affected by the size of the data sets but not as much as effect of the number of medians. For the Pmed (1-6-11-21-26-31-35-38), optimal or near optimal solutions are found. When we take these data sets into consideration, it can be easily said that these data sets include low number of medians. For the CPU times, it can be observed that it is affected by both size of the data sets and number of the medians. In Table 2, examining the last column, it can be acknowledged that the size of data sets and number of medians is effective on CPU time. Results show that, with the increase in number of medians, deviation also increases from the optimal solution accordingly. In addition, optimal and near optimal solutions can be reachable with the current state of the TSA. By taking aforementioned interpretations of results into consideration, it can be said that TSA can cope with the convergence to optimal solution depending on the searching place in solution space. However, exploration part of TSA can be considered as weak by examined the increasing deviation with increasing number of medians and nodes.

Table 2. Results of the TSA from OR-Library

P-median Data	Total Number of	Number of	Minimum Cost	Optimum	Gap	Average CPU Time
Pmed1	100	5	5819	5819	0	2.84
Pmed2	100	10	4194	4093	2.46	6.49
Pmed3	100	10	4359	4250	2.56	2.89
Pmed4	100	20	3398	3034	11.99	11.03
Pmed5	100	33	1666	1355	22.95	8.43
Pmed6	200	5	7824	7824	0	6.99
Pmed7	200	10	5801	5631	3.01	12.21
Pmed8	200	20	5134	4445	15.50	29.00
Pmed9	200	40	3362	2734	22.97	50.39
Pmed10	200	67	1740	1255	38.64	21.74
Pmed11	300	5	7702	7696	0.07	11.26
Pmed12	300	10	7068	6634	6.54	24.79
Pmed13	300	30	5133	4374	17.35	35.81
Pmed14	300	60	3820	2968	28.70	55.14
Pmed15	300	100	2413	1729	39.56	132.05
Pmed16	400	5	8164	8162	0.024	126.94
Pmed17	400	10	7367	6999	5.25	453.64
Pmed18	400	40	5710	4809	18.73	251.86
Pmed19	400	80	3706	2845	30.26	431.62
Pmed20	400	133	2601	1789	45.38	1081.05
Pmed21	500	5	9138	9138	0	76.30
Pmed22	500	10	8813	8579	2.72	1572.18
Pmed23	500	50	5580	4619	20.80	13327.3
Pmed24	500	100	3900	2961	31.71	1318.5
Pmed25	500	167	2665	1828	45.78	1659.14
Pmed26	600	5	9985	9917	0.68	104.10
Pmed27	600	10	8751	8307	5.34	212.23
Pmed28	600	60	5555	4498	23.49	2104.86
Pmed29	600	120	4005	3033	32.04	1498.68
Pmed30	600	200	2933	1989	47.46	3123.6
Pmed31	700	5	10123	10086	0.36	173.97
Pmed32	700	10	9917	9297	6.66	232.03
Pmed33	700	70	5797	4700	23.34	1469.07
Pmed34	700	140	4108	3013	36.34	6560.7
Pmed35	800	5	10437	10400	0.35	270.9
Pmed36	800	10	10383	9934	4.51	245.47
Pmed37	800	80	6355	5057	25.66	3525.99
Pmed38	900	5	11154	11060	0.84	144.45
Pmed39	900	10	9965	9423	5.75	594.63
Pmed40	900	90	6518	5128	27.10	5719.36
Average CPU time and Gap for application of TSA on Pmed1-40					16.32	1167.24

4.1. Experimental Design for TSA

From Figure 6 to Figure 15, the comparison is done between the parameters by increasing Search Tendency (ST) which leads fast convergence and *max_FEs* leading incline in iterations. In general, when ST increased, gathered solution from the algorithm decreased and became close to the optimal value or the optimal solution is found. For testing the effect of change in parameter on obtained results from the TSA, Pmed1 to Pmed10 datasets are taken into account. For the small size of datasets, effect of the parameter setting is not clearly observed because of reaching optimal solution without any need of increasing iteration or ST.

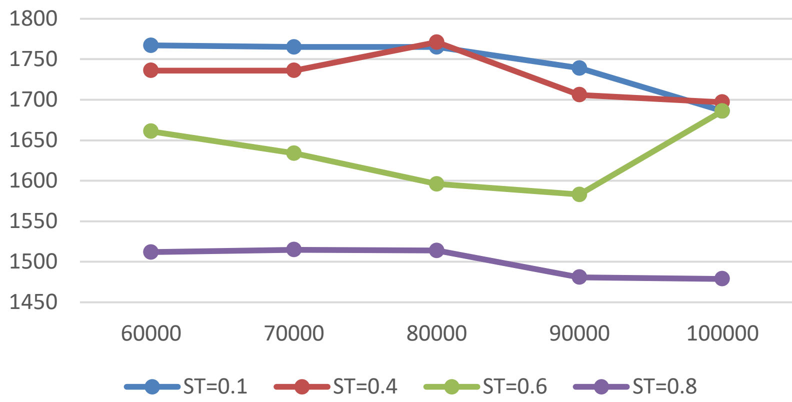


Figure 6. Pmed10 parameter alteration

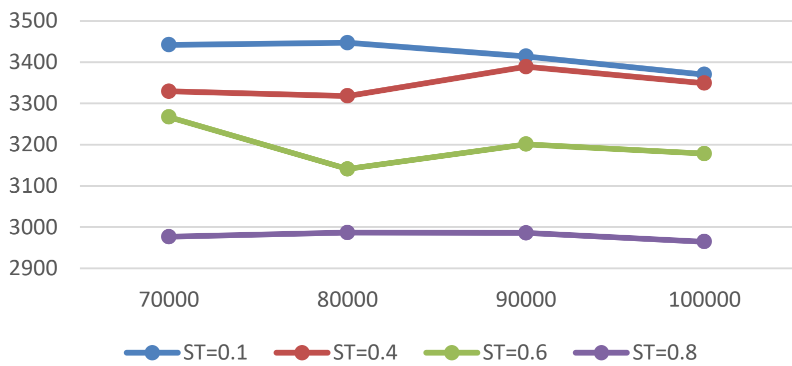


Figure 7. Pmed9 parameter alteration

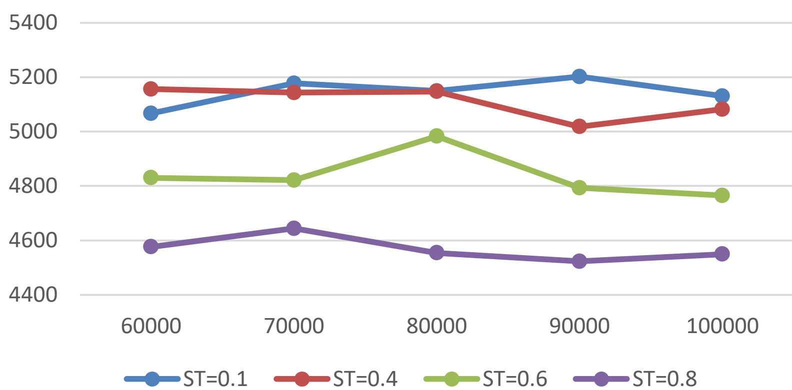


Figure 8. Pmed8 parameter alteration

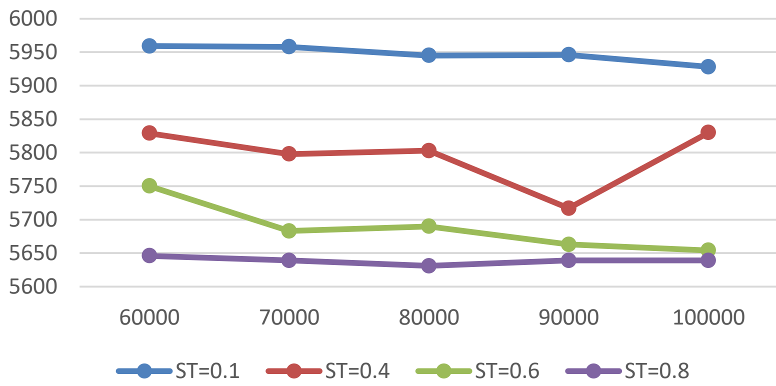


Figure 9. Pmed7 parameter alteration

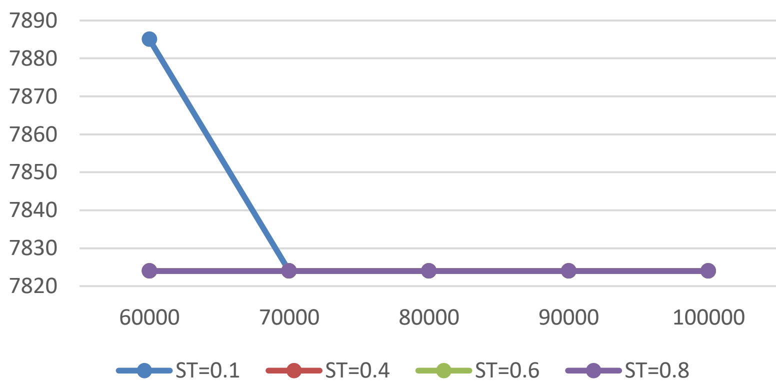


Figure 10. Pmed6 parameter alteration

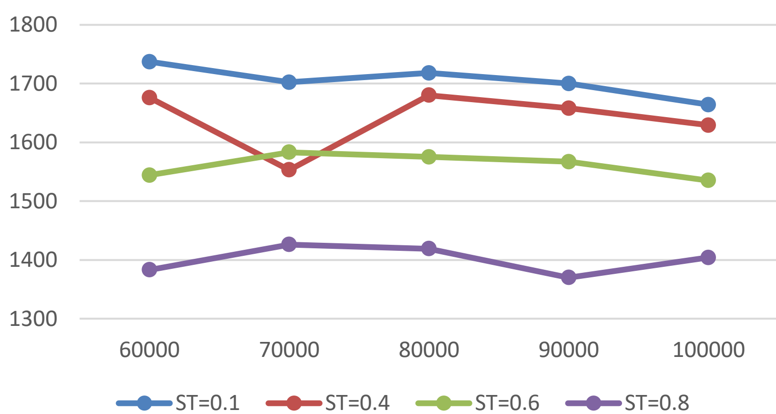


Figure 11. Pmed5 parameter alteration

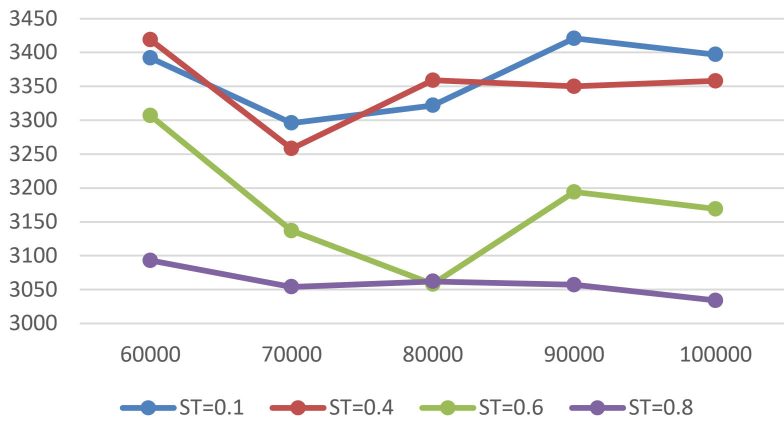


Figure 12. Pmed4 parameter alteration

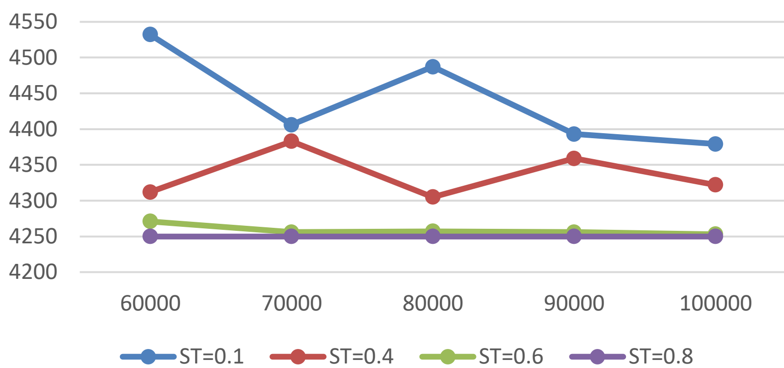


Figure 13. Pmed3 parameter alteration

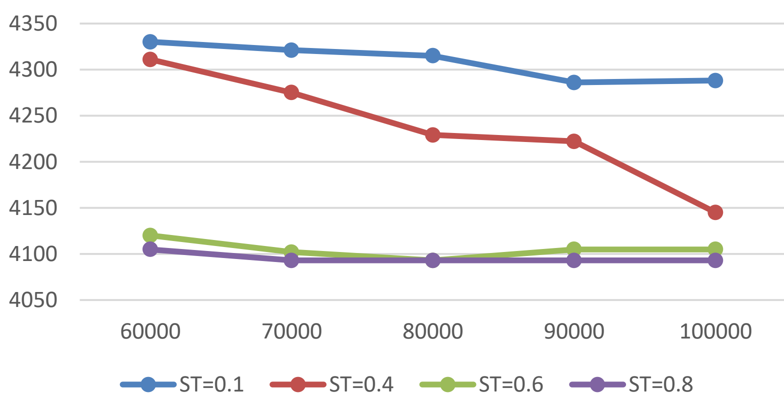


Figure 14. Pmed2 parameter alteration

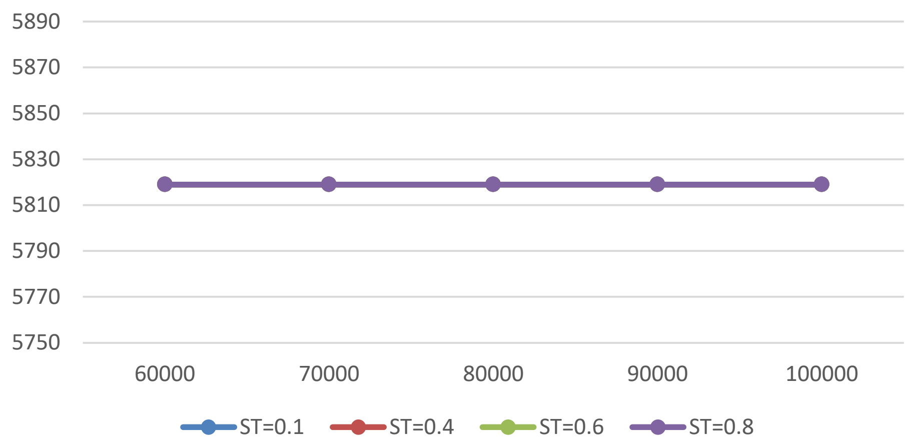


Figure 15. Pmed1 parameter alteration

For the Pmed2 and Pmed3, it can be expressed that after ST=0.6, the number of iterations do not have a major effect on the results. However, for ST less than 0.6, the results converge the optimum value rapidly with an increase in number of iterations. For both datasets (Pmed2-3), increase in ST gives a better solution close to the optimal. For the Pmed4 and Pmed8, there is a drastic decrease, when ST=0.8. The common feature for both datasets is having 20 centers. Therefore, it can be said that for datasets that have medium or large sized number of centers, ST=0.8 provides better improvement, means incline in ST provides a better improvement in results medium and large sized center datasets compare to datasets contains small number of centers. In addition, for the small value for ST, number of iteration has an effect. However, for the big values for ST, number of iterations has slightly affects the results.

5. CONCLUSION

In this paper, one of the novel meta-heuristic approaches, namely TSA, is applied to the P-median problem for the first time in the literature. The main objective of this study is to test the applicability of TSA on P-median problems. To do so, small and medium sized test problems are generated. The obtained results are compared with the optimal results obtained by GAMS-CPLEX. In addition, TSA is applied on forty data sets received from OR-Library. While the optimal solutions are achieved for a few test problems, near optimal solutions are obtained for the medium sized test problems. Further work can include modifications of the TSA that covers the updating the step stone solution to move next feasible solution by managing inheritance from the parent tree and providing better initial population. Therefore, TSA can be improved for the large sized test problems. During the application of TSA, confronted obstacles derives from the structure of TSA design which is for continuous problems, inadequate properties of used computer and integration of P- median problem into TSA. Shortcomings of the study can be expressed as applied TSA shows a low performance on large sized problems compared to medium and small sized problems considering the elapsed time and Gap.

Acknowledgement

We are thankful to the anonymous two reviewers and the editor for all of their valuable comments which helped us a lot in improving the paper significantly.

REFERENCES

[1] L. A. N. Lorena and E. L. F. Senne, "A column generation approach to capacitated p-median problems," *Comput. Oper. Res.*, vol. 31, no. 6, pp. 863–876, 2004.

[2] M. Herda, "Parallel Genetic Algorithm for Capacitated P-median Problem," *Procedia Eng.*, vol. 192, pp. 313–317, 2017.

[3] F. Chiyoshi and R. D. Galvão, "A statistical analysis of simulated annealing applied to the p-median problem," *Ann. Oper. Res.*, vol. 1–4, no. X, pp. 61–74, 2000.

[4] F. O. De França, F. J. Von Zuben, and L. N. De Castro, "MAX MIN Ant System and Capacitated p-Medians: Extensions and improved solutions," *Inform.*, vol. 29, no. 2, pp. 163–171, 2005.

[5] E. Rolland, D. A. Schilling, and J. R. Current, "An efficient tabu search procedure for the p-Median Problem," *Eur. J. Oper. Res.*, vol. 96, no. 2, pp. 329–342, 1997.

[6] B. Jayalakshmi and A. Singh, "A hybrid artificial bee colony algorithm for the p-median problem with positive/negative weights," *Opsearch*, vol. 54, no. 1, pp. 67–93, 2017.

[7] G. Lin and J. Guan, "A hybrid binary particle swarm optimization for the obnoxious p-median problem," *Inf. Sci. (Ny).*, vol. 425, pp. 1–17, 2018.

[8] O. Alp, E. Erkut, and Z. Drezner, "An Efficient Genetic Algorithm for the p-Median Problem," *Ann. Oper. Res.*, vol. 122, no. 1–4, pp. 21–42, 2003.

[9] P. Avella, M. Boccia, S. Salerno, and I. Vasilyev, "An aggregation heuristic for large scale p-median problem," *Comput. Oper. Res.*, vol. 39, no. 7, pp. 1625–1632, 2012.

[10] P. Prima and A. Aniati Murni, "Optimization of school location-allocation using genetic algorithm," in *Proceedings of 2018 the 8th International Workshop on Computer Science and Engineering*, 2018, pp. 750–755.

[11] B. Afify, S. Ray, A. Soeanu, A. Awasthi, M. Debbabi, and M. Allouche, "Evolutionary learning algorithm for reliable facility location under disruption," *Expert Syst. Appl.*, vol. 115, pp. 223–244, 2019.

[12] M. Karatas and E. Yakıcı, "An analysis of p -median location problem : Effects of backup service level and demand assignment policy ☆," *Eur. J. Oper. Res.*, vol. 272, no. 1, pp. 207–218, 2019.

[13] M. S. Kiran, "Expert Systems with Applications TSA : Tree-seed algorithm for continuous optimization," *Expert Syst. Appl.*, vol. 42, no. May, pp. 6686–6688, 2015.

[14] Y. Zhao, J. Liu, Z. Lyu, and Z. Ding, "Structural damage identification based on residual vectors and tree-seed algorithm," *Zhongshan Daxue Xuebao*, vol. 56, no. 4, pp. 46–50, 2017.

[15] W. J. Chen, X. J. Tan, and M. Cai, "Parameter Identification of Equivalent Circuit Models for Li-ion Batteries Based on Tree Seeds Algorithm," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 73, no. 1, 2017.

[16] A. A. El-Fergany and H. M. Hasanien, "Tree-seed algorithm for solving optimal power flow problem in large-scale power systems incorporating validations and comparisons," *Appl. Soft Comput. J.*, vol. 64, pp. 307–316, 2018.

[17] A. C. Cinar and M. S. Kiran, "Similarity and Logic Gate-Based Tree-Seed Algorithms for Binary Optimization," *Comput. Ind. Eng.*, vol. 115, no. December 2017, pp. 631–646, 2018.

[18] R. D. Galvão, "Uncapacitated facility location problems: contributions," *Pesqui. Operacional*, vol. 24, pp. 7–38, 2004.

[19] F. Plastria, “Continuous Covering Location Problems,” in *Facility Location:Applications and Theory*, no. January, D. Zvi and H. Horst W., Eds. Springer, 2002.
[20] J. Current, M. Daskin, and D. Schilling, *Discrete Network Location Models*, no. January 2016. 2002.
[21] S. Mete, Z. A. Cil, and E. Özceylan, “Location and coverage analysis of bike-sharing stations in university campus,” *Bus. Syst. Res.*, vol. 9, no. 2, pp. 80–95, 2018.
[22] J. Reese, “Solution methods for the p-median problem: An annotated bibliography,” *Networks*, vol. 48, no. 3, pp. 125–142, 2006.
[23] J. C. Teixeira and A. P. Antunes, “A hierarchical location model for public facility planning,” *Eur. J. Oper. Res.*, vol. 185, no. 1, pp. 92–104, 2008.