DOI: https://doi.org/10.71058/jodac.v9i6001

JOURNAL OF DYNAMICS AND CONTROL

ISSN: 1672 - 6553

JOURNAL OF DYNAMICS AND CONTROL

VOLUME 9 ISSUE 6: 1 - 10

OPTIMIZATION OF FUZZY TRAVELLING SALESMAN PROBLEM USING PARAMETRIC ANALYSIS WITH MATLAB PROGRAMMING

Jay Chandra Yadav[1], Mohammad Rizwanullah[2]

[1]Research Scholar, [2]Associate Professor
Department of Mathematics and Statistics,
Manipal University Jaipur, India

JOURNAL OF DYNAMICS AND CONTROL
DOI: https://doi.org/10.71058/jodac.v9i6001

OPTIMIZATION OF FUZZY TRAVELLING SALESMAN PROBLEM USING PARAMETRIC ANALYSIS WITH MATLAB PROGRAMMING

Jay Chandra Yadav[1*], Mohammad Rizwanullah[2]

[1]Research Scholar, [2]Associate Professor

Department of Mathematics and Statistics, Manipal University Jaipur, India

**Corresponding Author: jay.211051026@muj.manipal.edu*

Abstract: *This paper addresses the optimization of the Travelling Salesman Problem (TSP) using parametric analysis. The TSP is an NP-hard problem where the total distance and time traveled by an agent (e.g., salesman delivering pizzas from a center to multiple centers in a city) are minimized. In the present work, a fuzzy optimization perspective has been adopted by considering center-to-center distances as parametric variables. Using α-cuts, we converted the fuzzy TSP into a crisp optimization problem by slicing the fuzzy numbers at various levels of certainty. MATLAB programming is used to analyze how parametric modifications in the variables lead to changes in the optimal solution. At the end of a numerical illustration is given to justify the proposed method and solution algorithm.*

Keywords: *Optimization, Traveling salesman, Fuzzy system, Parametric analysis, Tour network*

1. Introduction

The delivery of a certain product from several sources to multiple destinations at the lowest possible cost is the focus of transportation difficulties. There are several optimization challenges in a variety of fields, including economics, engineering design, production systems, and the sciences. To investigate the practical, extremely useful optimization problems, which is a need for effective computational techniques and reliability. It can be applied to solve optimization problems in a variety of application domains [6].

One of the well-known NP-hard combinatorial optimization classes is the Traveling Salesperson Problem (TSP). The Travelling Salesperson Problem (TSP) has been solved by numerous renowned researchers using a variety of optimization techniques, including [2], [10], [18], [21], [22], [23], and [24]. Assume for the moment that a list of towns or cities along with separations between every pair of cities. So that the shortest trip that only makes one stop in each city must be found. Assume, for instance, that the salesman at Domino's Pizza has the responsibility of traveling in Jaipur city to multiple other centres just once to sell his or her goods before returning to the starting city. The salesman's primary objective is to find the shortest route that will minimize the overall distance and time [4], [7]. The salesman's ability to carry out their task efficiently and successfully is the main obstacle to this issue. A TSP solution's performance can be examined using techniques like situational analysis, parameter variation, and sensitivity analysis, helping to discover principal factors and direct decision-making in ambiguous situations.

2. OR Tool

OR-Tools is a versatile optimization tool that supports various algorithms. It is allowing users to tackle different types of optimization problems. Its time efficiency is high, allowing for faster problem-solving in practical applications and research fields. It helps to solve complicated problems. OR-Tools integrates well with other tools and allows you to use multiple programming languages, making it accessible to researchers and developers. As an open-source project, it is continually updated and improved by the community. OR-Tools is a fast and portable tool for combinatorial optimization. It is a method of one of the generic probabilistic C++, JAVA, or PYTHONS, which are crucial techniques for resolving problems of the NP-hard class. They are also employed to find an approximate solution for most global optimization issues.

DOI: https://doi.org/10.71058/jodac.v9i6001

3. Mathematical Formulation

Reducing the overall distance travelled is the TSP's goal.

Equation (1) used to express distance travelled:

$$Minimize\ \ Z = \sum_{(i,j)\in T} d_{ij} \,.\, x_{ij} \tag{1}$$

- Where:T is the collection of every city pair (i.e., every edge in the graph).
- d_{ij} is distance between city i and city j.
- x_{ij} is a binary decision variable:

$$x_{ij} = \begin{cases} 1, \text{if the route from city } i \text{ to city } j \text{ is chosen} \\ 0,\ \text{otherwise} \end{cases} \tag{2}$$

Z is the total distance of the tour.

Constraints

i.) *Visit Each City Exactly Once:* Each city must be visited exactly once, and the salesman must return to the starting city. Mathematically, this can be stated as follows (3):

$$\sum_{(i\neq j)\in T} x_{ij} = 1\ \ \forall i \tag{3}$$

This ensure that from each city i, exactly one path is taken.

ii.) *Return to the Starting City*: Equation (4) is true since the salesman needs to go back to the starting city (4).

$$\sum_{(j\neq i)\in T} x_{ji} = 1\ \ \forall j \tag{4}$$

As a result, the salesman will arrive in city j precisely once.

iii.) *Subtour Elimination*: To avoid the creation of subtours (tours that do not include all cities), we add a constraint. For a TSP with n cities, we use the following subtour elimination constraints (using the Miller-Tucker-Zemlin formulation) in equation (5):

$$u_i - u_j + n \cdot x_{ij} \leq n - 1\ \forall i,j \quad (i \neq j) \tag{5}$$

Where u_i is an additional variable used to order the cities in the tour.

3.1. Fuzzy optimization to formulate the TSP.

The fuzzy total distance is now the objective, which we define as its minimisation. Equation (6) can be used to express the tour's total fuzzy distance:

$$Z(\tilde{x}) = \sum_{(i,j)\in T} \tilde{d}_{ij} \tag{6}$$

where T is the collection of tour routes. The goal is to find the tour T that minimizes this fuzzy total distance.

The fuzzy objective can be managed using ranking methods or α-cuts. Using α-cuts, we convert the fuzzy problem into a crisp optimization problem by slicing the fuzzy numbers at various levels of certainty (e.g., 0.2, 0.5, 0.7) and then solving the resulting deterministic TSP for each level.

DOI: https://doi.org/10.71058/jodac.v9i6001

4. Parametric Analysis and Application

The parametric approach allows for adaptive decision-making and flexibility, ensuring minimal total time and distance for the delivery process. Parametric study, where supply and demand parameters were varied, can vary the distances between centers within their fuzzy ranges. We can perform parametric analysis by solving the TSP for different alpha-cut levels:

At $\alpha = 1$, we use the "core" of the fuzzy numbers, treating distances as crisp values in their most certain range.

At $\alpha = 0$, We make use of the entire range of distance uncertainty.

We solve the TSP for each alpha-cut by minimizing the total distance using the OR-Tools approach.

5. Numerical Illustration

A travelling salesman in a pizza center of Jaipur City who needs to cover from Malviya Nagar center to all the other centers in Jaipur once and only once to deliver his pizza and return to the starting center. The salesman wants to find the shortest possible route by which the total distance and total time taken will be minimized. The big challenge of this problem is that how can the salesman manage the route?

The TSP is solved using parametric analysis, where a parametric approach is introduced to the given distance matrix.

Table 1. Distance Matrix

	Malviya Nagar	GT	WTP	Mansarovar	DP Metro	MGF	Raj Mandir	Pink Square	Vaishali Nagar	VDN	Triton Mall	Jhotwara
Malviya Nagar	0	0.2	0.3	5	7	8	8	10	12	12	14	18
GT	0.2	0	0.1	5.1	7	8	8	10	12	12	14	17
WTP	0.3	0.1	0	5	7	8	8	10	11	11	14	18
Mansarovar	5	5.1	5	0	3	2	2	3	5	5	7	11
DP Metro	7	7	7	3	0	1.5	1	2	4	4	6	10
MGF	8	8	8	2	1.5	0	1	2	4	4	6	12
Raj Mandir	8	8	8	2	1	1	0	2	4	4	10	10
Pink Square	10	10	10	3	2	2	2	0	2	1	4	8
Vaishali Nagar	12	12	11	5	4	4	4	2	0	1	2	6
VDN	12	12	11	5	4	4	4	1	1	0	2	4
Triton Mall	14	14	14	7	6	6	10	4	2	2	0	2.5
Jhotwara	18	17	18	11	10	12	10	8	6	4	2.5	0

5.1. The Distance Matrix's Parametric Definition

Some of the centres' distances will be treated as parameters instead of constants. For instance, real-world circumstances like traffic may cause modest variations in the distances between nodes like Malviya Nagar and GT. To investigate their effect on the ideal TSP solution, these parameters can be changed.

5.2. Analysing Parametric:

We take into consideration LPP, where the data is linearly dependent on a single parameter. Parametric analysis of LPP is the process of solving such LPPs for a subset of parameter values that fall inside a certain range. We consider only those cases in which the parameter affects either:

(i) The objective function coefficients c_j or

(ii) The right-hand side resource vector b.

Objective Function of Parametric Analysis: With parametric distances, the objective function becomes:

DOI: https://doi.org/10.71058/jodac.v9i6001

$$Minimize\ Z(\alpha,\beta,\gamma)$$

$$= \sum_{(i,j)\in T} d_{ij}(\alpha,\beta,\gamma).x_{ij} \quad (7)$$

where $d_{ij}(\alpha,\beta,\gamma)$ represents the parametric distance between cities i and j, were the distances,
d-MG., d-GW., and d-vv. vary based on parameters α, β, and γ, respectively.
$Z(\alpha,\beta,\gamma)$ is the total distance for the optimal tour under the parametric values.

We introduce the parametric variation by focusing on some uncertain distances. Let us say the distances between *Malviya Nagar* and *GT*, *GT* and *WTP*, and *Vaishali Nagar* and *VDN* might vary due to traffic or road conditions. These distances are represented by parameters that have a minimum and a maximum value.

Distance between *Malviya Nagar* and *GT*:

$$d_{MG}(\alpha) = [0.1,0.2,0.3]$$

Distance between *GT* and *WTP*:

$$d_{GW}(\beta) = [0.05,0.1,0.15]$$

Distance between *Vaishali Nagar* and *VDN*:

$$d_{VV}(\gamma) = [0.8,1.0,1.2]$$

5.3. Procedure for Applying the Parametric Approach

Step (i): *Specify Fuzzy Parameter Ranges*

We provide a range of variation in the distances between these centres:

- $d_{MG} \in [0.1,0.3]$
- $d_{GW} \in [0.05,0.15]$
- $d_{VV} \in [0.8,1.2]$

Step (ii): *Determine Several Parameter Values*

OR-Tools is used to solve the TSP for each of the parametric variables. For various parameter choices, we run the TSP solution and observe how the ideal trip varies. Using parametric changes of certain distances between cities (pizza centres), we are attempting to solve the Traveling Salesman Problem (TSP). To return to the starting city, the TSP seeks the shortest path that makes precisely one stop in each city.

Here, we introduce parametric values for three key distances:

- d_{MG} : distance between *Malviya Nagar* and *GT*
- d_{GW}: distance between *GT* and *WTP*
- d_{VV}: distance between *Vaishali Nagar* and *VDN*

We compute the TSP solution for different values of these distances (minimum, average, and maximum) and observe how the optimal tour and the total distance change.

1. *For the* ***minimum*** *values of the parameters:*
   $d_{MG} = 0.1,\ d_{GW} = 0.05,\ d_{VV} = 0.8$
2. *For the* ***average*** *values:*
   $d_{MG} = 0.2,\ d_{GW} = 0.1,\ d_{VV} = 1.0$
3. *For the* ***maximum*** *values:*
   $d_{MG} = 0.3,\ d_{GW} = 0.15,\ d_{VV} = 1.2.$

DOI: https://doi.org/10.71058/jodac.v9i6001

In the context of the TSP, we modify the distance matrix for each case, updating the values of the parameters d_{MG}, d_{GW}, and d_{VV}.

Case 1, the distance matrix becomes:

$$D_{min} = \begin{bmatrix} 0 & 0.1 & \dots & & \\ 0.1 & 0 & 0.05 & \dots & \\ \dots & 0.05 & 0 & \dots & \\ \dots & \dots & \dots & 0.8 & \dots \end{bmatrix}$$

each case, we use the corresponding values of d_{MG}, d_{GW}, and d_{VV}.

Step(iii): Compute the Optimal Tours

After computing the optimal tours by MATLAB software, we get the output.

6. Result and discussion

MATLAB COMMAND WINDOW

```
>> tsp_parametric
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.65
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.65
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.65
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.7
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.7
Optimal Tour:
1  2  3  5  9  11  12  10  8  7  6  4  1
Minimum Distance: 30.7
Optimal Tour:
1  5  9  11  12  10  8  7  6  4  3  2  1
Minimum Distance: 30.75
Optimal Tour:
1  5  9  11  12  10  8  7  6  4  3  2  1
Minimum Distance: 30.75
Optimal Tour:
1  5  9  11  12  10  8  7  6  4  3  2  1
```

DOI: https://doi.org/10.71058/jodac.v9i6001

Minimum Distance: 30.75

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.75

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.75

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.75

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.8

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.8

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.8

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.85

DOI: https://doi.org/10.71058/jodac.v9i6001

Optimal Tour:

1 3 2 5 9 11 12 10 8 7 6 4 1

Minimum Distance: 30.9

Optimal Tour:

1 3 2 5 9 11 12 10 8 7 6 4 1

Minimum Distance: 30.9

Optimal Tour:

1 3 2 5 9 11 12 10 8 7 6 4 1

Minimum Distance: 30.9

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.95

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.95

Optimal Tour:

1 5 9 11 12 10 8 7 6 4 3 2 1

Minimum Distance: 30.95

Optimal Tour: This indicates the sequence of cities (pizza centers in this case) that form the shortest possible route for the TSP. The cities are represented by numbers, with city 1 (Malviya Nagar) serving as the starting and finishing point.

Optimal Tour: 1 2 3 5 9 11 12 10 8 7 6 4 1

According to this order, the salesperson begins in city 1 (Malviya Nagar), travels to cities 2, 3, 5, 9, 11, 12, 10, 8, 7, 6, and 4, and then circles back to city 1 to finish the journey.

Minimum Distance: Using the current set of parametric distances, this is the total distance covered on the ideal tour. A particular variant of the fuzzy parameters is associated with each distance (e.g., distinct values for the distances between certain cities like Malviya Nagar and GT). 30.65 is the minimum distance. Accordingly, in that specific parametric, the tour's total distance is 30.65 units (kilometers).

6.1. An examination of the findings:

Multiple Runs: You are seeing different results for each set of parameter values because in the result (tsp_parametric function) loops over different parametric values (for the distances between certain pairs of cities, such Malviya Nagar to GT, GT to WTP, and Vaishali Nagar to VDN). The algorithm determines the best tour and matching minimal distance for every set of distances.

Repetition of Results:

The similar ideal tour is visible for the initial several parametric sets (between 30.65 and 30.65):

The ideal tour:

1, 2, 3, 5, 9, 11, 12, 10, 8, 7, 6, 4

DOI: https://doi.org/10.71058/jodac.v9i6001

This indicates that this specific trip stays ideal for these parametric values, with only little changes in the overall distance (from 30.65 to 30.7).

Following minor adjustments to the parameters, the ideal tour becomes:

The ideal tour:

1, 5, 9, 11, 12, 10, 8, 7, 6, 4, 3, 2, 1

This indicates that a slightly larger total distance (from 30.7 to 30.75) was the new optimal tour that the computer discovered for the subsequent set of parametric variables.

Parametric Variation and Solution Stability: The best solution (tour) may or may not vary as the parametric variables do. Sometimes the same tour produces the best results for slightly different distances, which results in modest differences in the minimum distance (30.65 vs. 30.7). In some instances, the system locates an alternative tour that is comparable but has a marginally greater overall distance (e.g., 30.75 vs. 30.85).

Increasing Distance Gradually: As the parametric distances are changed, we can see that the distances progressively get longer:

30.65 → 30.7 → 30.75 → 30.8 → 30.85 → 30.9 → 30.95

The overall distance of the ideal route is gradually increasing as a result of the growing values of the parametric distances between cities, according to this.

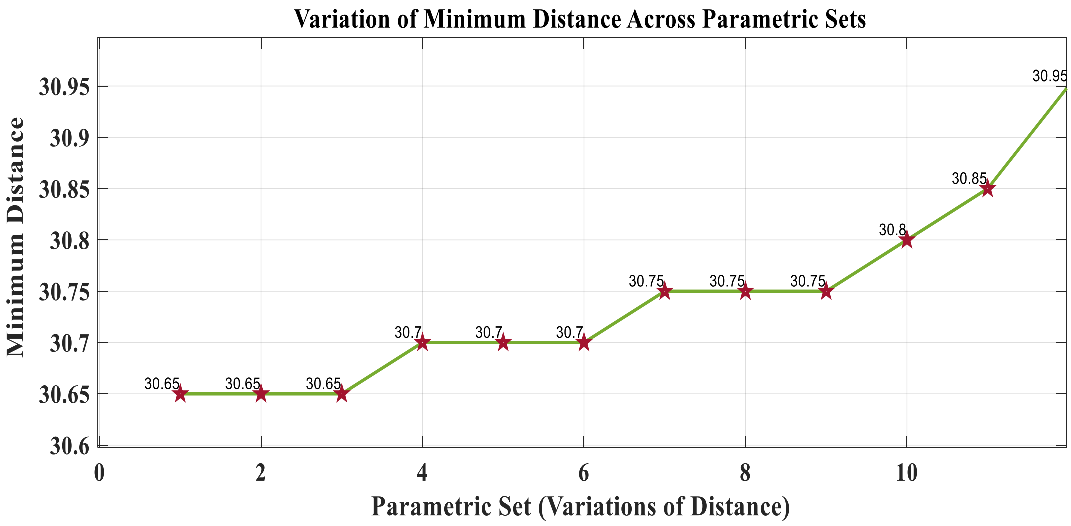

Figure 1. Variation of Minimum distance across Parametric sets.

The X-axis parametric set represents different combinations of parametric values. A specific set of values for the distances between specific city pairs is represented by each point on the X-axis, and the lowest distance of the ideal TSP tour for each parameter set is shown on the Y-axis. The graph will demonstrate how changing the parameter values affects the minimum distance. Most likely, as a result of shifting parametric values, we saw a slow rise in the minimum distance.

The minimum distance, 30.65, keeps the same for the first few parametric sets (1 to 3). The minimum distance marginally rises to 30.7 as we proceed to sets 4 through 6, suggesting that the parametric variables have begun to impact the solution. From parametric sets 7 to 12, the minimum distance keeps becoming longer, coming in at 30.95 for the last set. We can see from this graph how responsive the TSP solution is to variations in specific distance values between cities. Before growing, it also shows how stable the answer is for a few ranges (constant distance for parametric sets 1-3 and 4-6).

Use footnotes sparingly (or not at all) and place them at the bottom of the column of the page on which they are referenced. Use Times New Roman 9-point type, single-spaced. To help your readers, avoid using footnotes altogether and include necessary peripheral observations in the text (within parentheses, if you prefer, as in this sentence).

7. Conclusion

In this study, we present a comprehensive approach to solve the Travelling Salesman Problem (TSP) using parametric analysis under a fuzzy environment. By introducing variable distances between pizza delivery centers using OR-Tools to solve the proposed TSP. The result shows that slight changes in parameters can lead to variations in the overall distance and optimal route. In the result we explained and highlighted the stability of the solution across parameter ranges. The result demonstrates the sensitivity analysis of the TSP to changes in specific distance values. This method can be effectively utilized for practical applications, such as delivery routing problems, where conditions like traffic may alter travel times.

Acknowledgments

Funding: There is no funding given to this research work by any agency.

Data availability Statement: Not applicable.

Declaration

Conflict of Interest: The author declares that they have no conflict of interest.

References

[1] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, (**1983**), pp. 671–680.

[2] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, "Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search," *Appl. Soft Comput.*, vol. 11, no. 4, (**2011**), pp. 3680–3689. https://doi.org/10.1016/j.asoc.2011.01.039

[3] S. W. Lin, F. Y. Vincent, and S. Y. Chou, "Solving the truck and trailer routing problem based on a simulated annealing heuristic," *Comput. Oper. Res.*, vol. 36, no. 5, (**2009**), pp. 1683–1692. https://doi.org/10.1016/j.cor.2008.04.005

[4] R. Zhang and C. Wu, "A hybrid immune simulated annealing algorithm for the job shop scheduling problem," *Appl. Soft Comput.*, vol. 10, no. 1, (**2010**), pp. 79–89. https://doi.org/10.1016/j.asoc.2009.06.008

[5] B. Cakir, F. Altiparmak, and B. Dengiz, "Multi-objective optimization of a stochastic assembly line balancing: A hybrid simulated annealing algorithm," *Comput. Ind. Eng.*, vol. 60, no. 3, (**2011**), pp. 376–384. https://doi.org/10.1016/j.cie.2010.08.013

[6] L. M. Gambardella and M. Dorigo, "Ant-Q: A reinforcement learning approach to the traveling salesman problem," in *Machine Learning Proceedings 1995*, Morgan Kaufmann, (**1995**), pp. 252–260. https://doi.org/10.1016/B978-1-55860-377-6.50039-6

[7] T. Stützle and H. Hoos, "MAX-MIN ant system and local search for the traveling salesman problem," in *Proc. IEEE Int. Conf. Evol. Comput. (ICEC'97)*, (**1997**), pp. 309–314. https://doi.org/10.1109/ICEC.1997.592327

DOI: https://doi.org/10.71058/jodac.v9i6001

[8] J. Zhu, T. Rui, H. Fang, J. Zhang, and M. Liao, "Simulated annealing ant colony algorithm for QAP," in *Proc. 8th Int. Conf. Nat. Comput.*, (**2012**), pp. 789–793. https://doi.org/10.1109/ICNC.2012.6234519

[9] F. H. Khan, N. Khan, S. Inayatullah, and S. T. Nizami, "Solving TSP problem by using genetic algorithm," *Int. J. Basic Appl. Sci.*, vol. 9, no. 10, (**2009**), pp. 79–88.

[10] S. S. Ray, S. Bandyopadhyay, and S. K. Pal, "New Genetic operators for solving TSP: Application to microarray gene ordering," in *Pattern Recognition and Machine Intelligence, PReMI 2005, Proc.*, vol. 1, Springer, (**2005**), pp. 617–622.

[11] R. Thamilselvan and P. Balasubramanie, "A genetic algorithm with a Tabu search (GTA) for traveling salesman problem," *Int. J. Recent Trends Eng.*, vol. 1, no. 1, (**2009**), pp. 607.

[12] I. Stojanovska and A. Dika, "Notice of Retraction: Impact of the number of generations on the fitness value and the time required to find the optimal solution in standard GA applications," in *Proc. 3rd Int. Conf. Comput. Sci. Inf. Technol.*, vol. 1, IEEE, (**2010**), pp. 589–593. https://doi.org/10.1109/ICCSIT.2010.5563975

[13] A. Misevičius, "A modified simulated annealing algorithm for the quadratic assignment problem," *Informatica*, vol. 14, no. 4, (**2003**), pp. 497–514.

[14] T. Peng, W. Huanchen, and Z. Dongme, "Simulated annealing for the quadratic assignment problem: A further study," *Comput. Ind. Eng.*, vol. 31, no. 3–4, (**1996**), pp. 925–928. https://doi.org/10.1016/S0360-8352(96)00265-3

[15] P. P. Yip and Y. H. Pao, "A guided evolutionary simulated annealing approach to the quadratic assignment problem," *IEEE Trans. Syst. Man Cybern.*, vol. 24, no. 9, (**1994**), pp. 1383–1386. https://doi.org/10.1109/21.310514

[16] G. Paul, "Comparative performance of Tabu search and simulated annealing heuristics for the quadratic assignment problem," *Oper. Res. Lett.*, vol. 38, no. 6, (**2010**), pp. 577–581. https://doi.org/10.1016/j.orl.2010.09.009

[17] M. S. Hussin and T. Stützle, "Tabu search vs. simulated annealing as a function of the size of quadratic assignment problem instances," *Comput. Oper. Res.*, vol. 43, (**2014**), pp. 286–291. https://doi.org/10.1016/j.cor.2013.10.007

[18] G. Dueck and T. Scheuer, "Threshold accepting: A general-purpose optimization algorithm appearing superior to simulated annealing," *J. Comput. Phys.*, vol. 90, no. 1, (**1990**), pp. 161–175. https://doi.org/10.1016/0021-9991(90)90201-B

[19] M. M. Flood, "The traveling-salesman problem," *Oper. Res.*, vol. 4, no. 1, (**1956**), pp. 61–75. https://doi.org/10.1287/opre.4.1.61

[20] M. Malek, M. Guruswamy, M. Pandya, and H. Owens, "Serial and parallel simulated annealing and Tabu search algorithms for the traveling salesman problem," *Ann. Oper. Res.*, vol. 21, (**1989**), pp. 59–84.

[21] Saad and S. A. Abass, "A parametric study on transportation problems under fuzzy environment," (**2002**). http://hdl.handle.net/10576/7992

[22] M. Nilofer and M. Rizwanullah, "The New Approach to Traveling Salesman Problem using Branch and Bound Method with case study of Domino's Pizza Centers," *Adv. Fuzzy Math.*, vol. 12, no. 4, (**2017**), pp. 1017–1033.