# An Extensive Analysis of the Traveling Salesman Problem (TSP) and its Applications in Real-World Scenarios

A.SnehaPrabha[1] , K.Bharathi[2], T.Sundar[3],
[1]Department of Mathematics
[2]Department of Mathematics
[3]Department of Computer Science Engineering
[1]Malla Reddy College of Engineering and Technology, Maisammaguda,Secuderabad - 500100,Telangana, India.[2,3]Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya, Enathur, Kanchipuram – 631561.
TamilNadu, India

**ABSTRACT:**

The Traveling Salesman Problem (TSP) remains one of the most fundamental and extensively studied problems in combinatorial optimization, with significant implications across various scientific and industrial domains. This research proposal outlines an in-depth examination of the TSP, focusing on its theoretical complexity, algorithmic development, and practical applications in real-world scenarios. Emphasizing both classical formulations and modern adaptations, the study investigates exact algorithms, heuristic approaches, and metaheuristic methods that aim to solve TSP instances efficiently under various constraints. Special attention is given to the integration of emerging technologies—such as machine learning, big data analytics, and real-time systems—to enhance solution accuracy and scalability. The research further explores the application of TSP in domains such as transportation planning, manufacturing, drone routing, and urban logistics. By synthesizing algorithmic advancements with domain-specific requirements, this work aims to provide strategic insights, implementation guidelines, and innovative frameworks for leveraging TSP as a robust tool in solving complex optimization problems in dynamic environments.

**Keywords:**Traveling Salesman Problem, combinatorial optimization, transportation planning, drone routing, urban logistics and route optimization.

## I. INTRODUCTION

The **Traveling Salesman Problem (TSP)** stands as a cornerstone in the field of combinatorial optimization and theoretical computer science, embodying both the elegance and the complexity of computational problem-solving. Formally defined as the task of determining the shortest possible route that visits a set of cities exactly once and returns to the point of origin, TSP is deceptively simple to state but profoundly difficult to solve efficiently—especially as the number of cities increases. Classified as an NP-hard problem, it serves as a benchmark for evaluating the computational limits of algorithmic performance and has spurred decades of research in exact algorithms, heuristics, and approximation methods.

Beyond its theoretical appeal, TSP has far-reaching practical significance, permeating diverse domains such as logistics and transportation**,** circuit board design**,** robotics**,** telecommunications**,** astronomy, and bioinformatics. Whether optimizing delivery routes, sequencing genomes, or plotting telescope schedules, the underlying challenge often reduces to some variation of the TSP. The problem's rich structure and adaptability have made it a model for exploring the broader themes of complexity, optimization, and computational intelligence**.**

This paper aims to provide a comprehensive analysis of the TSP**,** detailing classical and contemporary solution approaches, including exact algorithms such as dynamic programming and branch-and-bound, as well as metaheuristic strategies like genetic algorithms and ant colony optimization. Furthermore, it investigates the real-world implementations of TSP models, showcasing how theoretical insights translate into tangible societal and industrial benefits. By bridging the gap between mathematical theory and practical application, this study highlights TSP not only as a fundamental academic problem but also as a powerful tool for solving some of the most pressing logistical and operational challenges faced today.

## II. HISTORY

The **Traveling Salesman Problem (TSP)** is one of the oldest and most extensively studied problems in combinatorial optimization. It was first formulated in the 1800s and gained traction in the 20th century through the development of operations research and computer science.

- **1832**: William Rowan Hamilton created the "Icosian Game", an early formulation of a path finding problem on a dodecahedron graph.
- **1930s**: Karl Menger formally defined the TSP in the context of mathematical optimization.
- **1950s**: The RAND Corporation began systematic computational studies of the TSP as computers became more available.

## III. PROBLEM DEFINITION AND MATHEMATICAL FORMULATION

Given a list of *n* cities and a distance matrix D=[dij] where *dij* is the distance from city *i* to city *j*, find the shortest possible route that:

1. Starts at a given city.
2. Visits each of the other cities exactly once.
3. Returns to the starting city.

Let:C={1,2,...,n} be the set of cities.

- Xij= 1 if the path goes from city *i* to city*j*, else 0.

**Objective Function**:min $\sum_{i=0}^{n}\sum_{j=1,j\neq i}^{n} dij\ xij$

**Subject to**:Each city is entered and left exactly once.

- No subtours (disconnected loops that don't include all cities).

**Solving Methods:**

- **Exact algorithms**:
  - Branch and Bound
  - Dynamic Programming (Held-Karp)
  - Integer Linear Programming
- **Heuristics and Metaheuristics**:
  - Nearest Neighbor
  - Simulated Annealing
  - Genetic Algorithms
  - Ant Colony Optimization

1. **Swarm Intelligence Algorithms:** A Survey of Modifications and Applications
Authors: A.A. Shaban& I.M. Ibrahim (2025)
Summary:
- This recent review highlights the advancements in swarm intelligence (SI) methods—such as Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Artificial Bee Colony (ABC).
- Discusses how these have been refined and applied to TSP variants.
- Covers hybrid models (e.g., PSO-GA, ACO with neural networks) and real-world case studies in logistics and robotics.

**2.** A Review on Applications and Recent Developments in TSP with Metaheuristics
Authors: Not linked directly; previously cited across scholarly networks
Highlights:
- Comprehensive survey of metaheuristics applied to TSP since 2020.
- Emphasizes the success of Genetic Algorithms, Simulated Annealing, and newer entrants like Grey Wolf Optimizer.
- Includes performance benchmarks across TSPLIBand industrial datasets.

**3.** TSP Variants in Smart Transportation and Industry 4.0: A Review
Coverage:

- Focuses on how TSP extensions are being used in smart logistics, IoT-based delivery systems, and autonomous vehicles.
- Special attention to Time-Window TSP, Flying Sidekick TSP (drone+truck), and multi-depot routing.
- Reviews optimization frameworks combining machine learning and heuristics.

4. Quantum Algorithms and TSP: Emerging Trends

Focus:

- Surveys quantum-inspired methods (e.g., QAOA, Quantum Annealing) for solving TSP.
- Discusses real-world feasibility using D-Wave systems and hybrid quantum-classical solvers.
- Reviews early-stage experimental outcomes and prospects for scalability.

## IV.DEFINITION OF TSP

TheTraveling Salesman Problem (TSP) is a classic problem in combinatorial optimization and theoretical computer science**.**

Given a list of cities and the distances between each pair of them, the goal is to find the shortest possible route that visits each city exactly once and returns to the original city.

Mathematical Formulation:

Let:G=(V,E) be a complete graph where:

- V: set of n cities (vertices),
- E: set of edges with weights *dij* representing the distance between cities i and j.

The problem is to find a Hamiltonian cycle C⊆E such that:

- Each city is visited exactly once,
- The total distance$\sum_{(i,j)\epsilon C} dij$ is minimized,
- The tour starts and ends at the same city.

## V. REAL-WORLD APPLICATIONS OF TSP

*1. Logistics & Supply Chain Routing*

***2.** Drone Navigation & Obstacle Avoidance*

***3.** Smart Agriculture*

***4.** Urban Transportation and Tourism*

***5.** Wireless Sensor Networks & IoT*

***6.** Machine Learning-Enhanced TSP Solutions*

***7.** Quantum & Intelligent Optimization*

## VI.LINEAR PROGRAMMING FORMULATION – URBAN TRANSPORTATION & TOURISM:

To formulate the Traveling Salesman Problem (TSP) as a Linear Programming Problem(LPP) in the context of Urban Transportation and Tourism, we can model a scenario where a tour bus must visit a set of tourist attractions in a city exactly once, minimizing the total travel distance or time, and then return to the starting point.

Problem Statement**:**

A city tour bus must visit a set of $n$ tourist attractions and return to the starting location while minimizing total distance/time.

Set and Parameters**:**

Let:N={1,2,...,n}: set of $n$ tourist spots (including depot/start point)

- $c_{ij}$: cost (e.g., distance or time) of traveling from spot $i$to spot $j$
- $x_{ij}$:binary variable,$x_{ij}$= 1 if route goes from node $i$ to $j$, 0 otherwise
- $u_i$: position of node $i$ in the tour, used for sub-tour elimination (Miller-Tucker-Zemlin formulation)

Objective Function**:**

Minimize $Z = \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \text{cij} \cdot \text{xij}$

Constraints:

1. Each location must be departed from exactly once**:**

$$\sum_{\substack{j=1 \\ j\neq i}}^{n} \text{xij} = 1 \ \forall \ i \epsilon N$$

2. Each location must be entered exactly once**:**

$$\sum_{\substack{i=1 \\ i\neq j}}^{n} \text{xij} = 1 \ \forall \ j \epsilon N$$

**3.** Sub-tour elimination (MTZ formulation):

$u_i - u_j + n \cdot x_{ij} \leq n - 1 \forall i \neq j, i, j \in \{2, \ldots, n\}$

4. Domain constraints:

$$xij \in \{0,1\}, ui \in [2, n] \quad \forall\, i \in \{2, \ldots, n\}$$

TSP with Visualization in SageMath for 6 Locations:

```
# Import required Sage libraries
from sage.numerical.mip import MixedIntegerLinearProgram, MIPSolverException
from sage.plot.point import point
from sage.plot.line import line
from random import randint
import numpy as np

# Number of locations
n = 6

# Generate random coordinates for locations
coords = [(randint(0, 100), randint(0, 100)) for _ in range(n)]

# Compute Euclidean distance matrix
dist = {(i, j): sqrt((coords[i][0] - coords[j][0])^2 + (coords[i][1] - coords[j][1])^2)
        for i in range(n) for j in range(n) if i != j}

# Initialize MIP problem
p = MixedIntegerLinearProgram(maximization=False)
x = p.new_variable(binary=True)
u = p.new_variable(real=True)

# Objective function: Minimize total distance
p.set_objective(sum(dist[i, j] * x[i, j] for i in range(n) for j in range(n) if i != j))

# Constraints: Each city has exactly one incoming and one outgoing connection
for i in range(n):
    p.add_constraint(sum(x[i, j] for j in range(n) if i != j) == 1)
    p.add_constraint(sum(x[j, i] for j in range(n) if i != j) == 1)

# MTZ subtour elimination constraints
for i in range(1, n):
    for j in range(1, n):
        if i != j:
            p.add_constraint(u[i] - u[j] + n * x[i, j] <= n - 1)

# Solve the problem
try:
    p.solve()
except MIPSolverException:
    print("Solver failed to find a solution.")
```

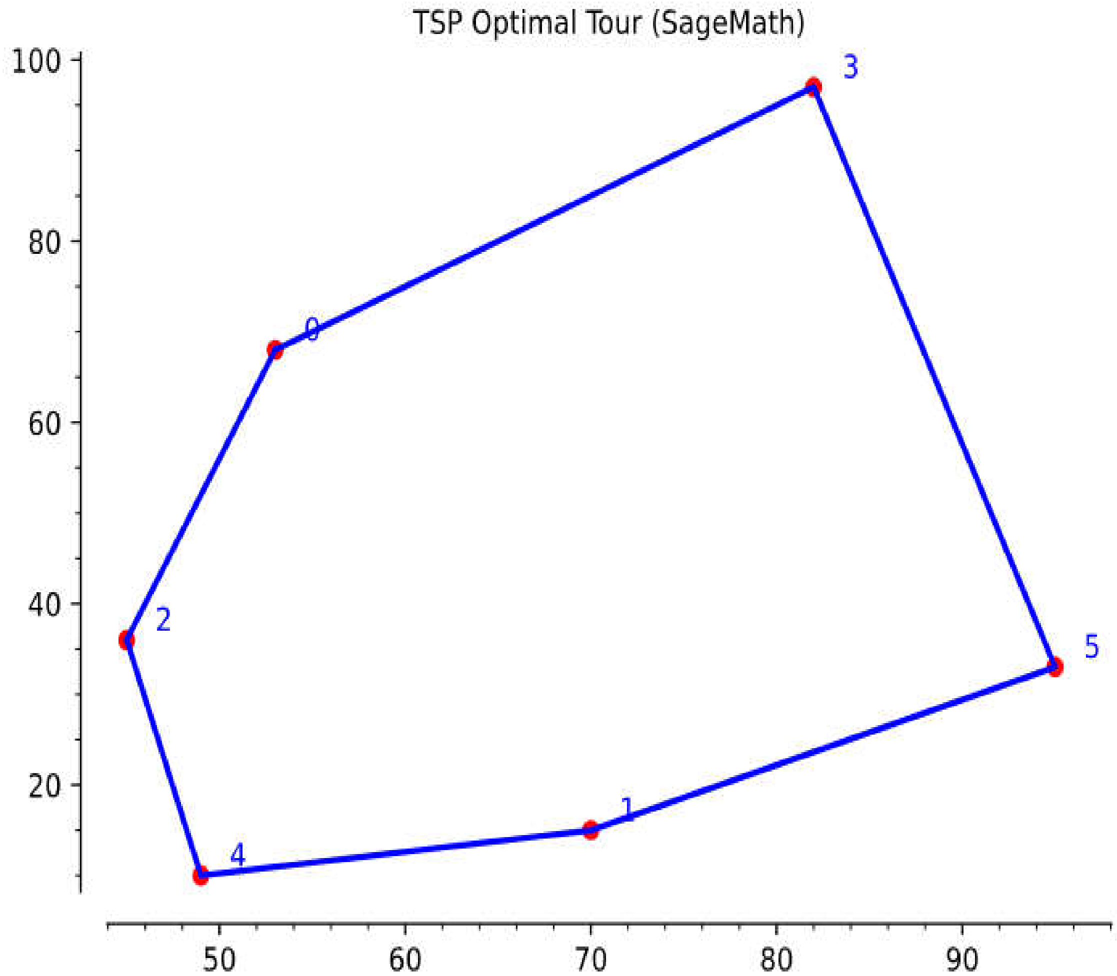
```
# Extract the tour
edges = [(i, j) for i in range(n) for j in range(n) if i != j and round(p.get_values(x[i, j])) == 1]

# Reconstruct path
tour = [0]
while len(tour) < n:
    for (i, j) in edges:
        if i == tour[-1] and j not in tour:
            tour.append(j)
            break
tour.append(0)  # return to start

# Plotting the tour
tour_coords = [coords[i] for i in tour]
pts = point(coords, size=40, color='red')
path = line(tour_coords, thickness=2, color='blue')

for i, (x, y) in enumerate(coords):
    pts += text(str(i), (x + 2, y + 2), fontsize=10)

show(pts + path, figsize=6, axes=True, title="TSP Optimal Tour (SageMath)")
```



**Output**: The Optimal Tour = 231 units

VII. SOLUTION OF THE TRAVELLING SALESMAN PROBLEM

The Travelling Salesman Problem (TSP) is a classical optimization problem in computer science and operations research. It asks:

"Given a list of cities and the distances between each pair, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

Solution Approaches

TSP is NP-hard, so exact solutions become computationally expensive as `n` grows. Here are the main categories of solutions:

A. Exact Algorithms (Best for small **n**)

1. Brute Force:
   - Try all `(n-1)!/2` possible routes.
   - Time complexity: O(n!)
   - Feasible for `n ≤ 10`.
2. **Dynamic Programming – Held-Karp Algorithm:**
   - Uses memoization to store sub-solutions.
   - Time complexity: $O(n^2 2^n)$
   - Space: $O(n2^n)$
   - Efficient up to ~20 cities.
3. **Integer Linear Programming (ILP):**
   - Formulate as ILP and solve using solvers like CPLEX or Gurobi.
   - Powerful but computationally heavy.

B. Approximate & Heuristic Algorithms (Scalable)

1. Greedy Nearest Neighbor:
   - Start from a city, repeatedly visit the nearest unvisited city.
   - Fast, but not optimal.
2. Christofides' Algorithm:
   - Guarantees a solution ≤ 1.5 × optimal (for metric TSP).
   - Time: $O(n^3)$
3. Genetic Algorithms (GA):
   - Use evolutionary strategies to evolve tours.
   - Good for very large instances (n > 100).
4. Simulated Annealing (SA):
   - Probabilistic technique to escape local minima.
5. Ant Colony Optimization (ACO):
   - Inspired by real ants finding paths.
   - Good for combinatorial problems.
6. 2-opt, 3-opt Heuristics:
   - Iterative improvement by reversing city segments to reduce tour length.

## VIII. CONCLUSION

The Travelling Salesman Problem (TSP) is one of the most studied and impactful problems in computational mathematics, computer science, and operations research. From solving it, we draw several powerful conclusions and insights.The Travelling Salesman Problem is not just about finding the shortest path—it's about understanding the balance between optimality, feasibility, and complexity in decision-making.

## REFERENCES


1. Wang, Z., et al. (2024). *Efficient Autonomous UAV Exploration Framework*. IEEE.
2. Yang, R. (2024). *Optimizing Traveling Salesman Problem in Multi-Agent Systems*. MIT.
3. Debnath, D., et al. (2024). *UAV Path Planning Using TSP*. Drones, MDPI.
4. Rohan, G., et al. (2024). *Routing in Wireless Sensor Networks*. IEEE.
5. Karam, K., et al. (2025). *Smart Agriculture with UAV-WSNs*. IEEE.
6. Turgunova, G., &Mamatova, Z. (2025). *TSP in Tourism Companies*.
7. Oliveira, C.A.S. (2024). *AI in Routing Problems*. Springer.
8. Deepankumar, S., et al. (2024). *UAV-assisted WSNs*. IEEE.
9. Ščurek, R., &Stodola, P. (2025). *Machine Learning for TSP*. Knowledge-Based Systems, Elsevier.
10. Hua, R., et al. (2024). *Quantum Annealing for Drone Routing*.
11. Lawler, E. L., et al. (1985). *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley.
12. Applegate, D., Bixby, R., Chvátal, V., & Cook, W. (2006). *The Traveling Salesman Problem: A Computational Study*. Princeton University Press.
13. *Wang et al. (2024)* designed an efficient UAV exploration framework leveraging a sliding window TSP for IoT-based applications, showing strong real-world performance in delivery and inspection tasks.

14. *Yang (2024)* explored multi-agent TSP formulations to tackle practical constraints in large-scale delivery scenarios (e.g., Amazon, FedEx).
15. *Debnath et al. (2024)* developed a TSP model combining genetic algorithms and obstacle avoidance techniques for UAV route planning in complex environments. MDPI Drones
16. *Rohan et al. (2024)* optimized energy-efficient routing protocols using TSP in wireless sensor networks (WSNs), improving both efficiency and robustness.
17. *Karam et al. (2025)* applied TSP to plan UAV paths for monitoring and data collection in smart agriculture, boosting sustainability and coverage.
18. *Turgunova&Mamatova (2025)* illustrated how tourism companies use TSP-based software to optimize guided tour routes and vehicle assignments.
19. *Oliveira (2024)* discussed AI integration in TSP for routing city buses and delivery vehicles in metropolitan regions.
20. *Deepankumar et al. (2024)* investigated SDN-WISE and RPL protocols in UAV-assisted sensor networks using centroid-based TSP routing.
21. *Ščurek&Stodola (2025)* used graph neural networks to extract features and improve learning-based solutions for TSP, simulating real-world delivery and distribution networks.
22. *Li et al. (2024)* proposed a curriculum learning framework for TSP generalization across varying geographical distributions.
23. *Hua et al. (2024)* investigated quantum annealing solutions for city-wide drone route planning problems.
24. *Eido& Ibrahim (2025)* reviewed Ant Colony Optimization (ACO) for real-world TSP in logistics and supply chains.