RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Drone 3D Pathfinder: Optimized Navigation via A*, TSP & Knapsack

Simulates smart drone navigation in 3D space using A*, TSP, and Knapsack algorithms to find optimal, obstacle-free routes for multiple delivery goals with weight constraints.

By: Hitarth Mehra 1RV23CS100
Advay Gujar 1RV23CS094

Department of CSE, RVCE

Guide: Prof. Ganashree K C

Problem Statement

No Combined Simulator

No existing simulator combines A*, TSP, and Knapsack.

Limited UI Tools

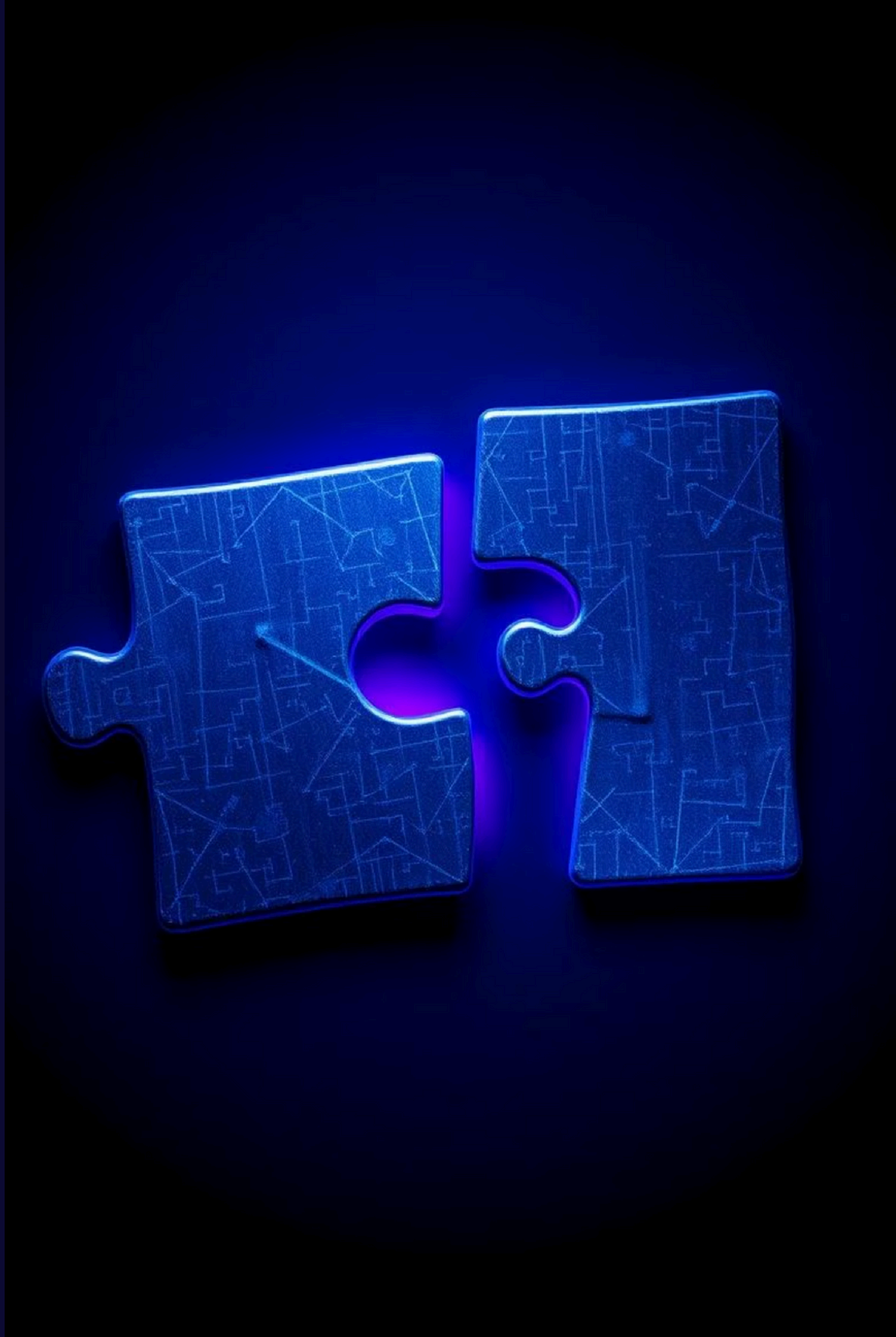
Limited UI-based tools for students to visualize 3D routing and obstacle avoidance.

Unreachable Destinations

Existing solutions do not handle unreachable destinations or blocked paths gracefully.

No Delivery Prioritization

No mechanism to prioritize delivery points based on value-to-weight ratio.



Introduction

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Navigating Complex Environments

Drones face significant hurdles in complex 3D spaces, including numerous obstacles, strict payload limitations, and the need to reach multiple destinations efficiently. Current approaches often tackle these issues in isolation.

Unified Simulation Platform

There's a pressing demand for a comprehensive simulation system that seamlessly integrates real-time navigation, intelligent delivery selection, and optimized route planning, all presented through clear, intuitive visualizations.

A* for Optimal Paths

Our solution leverages the A* algorithm to precisely determine the shortest, most efficient paths for drones, enabling them to skillfully navigate around obstacles and reach their destinations quickly.

TSP for Efficient Routing

We apply the Traveling Salesperson Problem (TSP) to optimize multi-goal routing. This ensures the most efficient sequence of visits to various delivery points, drastically reducing total travel distance.

Knapsack for Payload Optimization

The Knapsack problem is integrated to intelligently select the most valuable items for delivery. This strictly adheres to the drone's weight and volume capacities, ensuring highly efficient payload management.

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Literature Survey (with Base Paper)

Ref. No.	Paper Title	Authors	Year	Relevance to Our Work	Gap Identified
[1]	A Survey of 3D Space Path-Planning Methods and Algorithms	Luo et al.	2022	Base paper — review of 3D navigation methods	No implementation combining A*, TSP, and Knapsack in one system
[2]	A Modified Genetic Algorithm for Solving the Multi-objective Drone Routing	Liu, Qi & Zhu	2018	Applies GA to multi-goal drone routing	Non-deterministic; lacks weight-aware filtering and exact routing
[3]	An Efficient 3D Pathfinding Algorithm Using Improved A*	Rao & Kapoor	2016	Improves 3D A* performance in obstacle-heavy maps	Doesn't handle multi-goal routing or value-based delivery optimization
[4]	Sampling-based Algorithms for Optimal Motion Planning	Karaman & Frazzoli	2011	Introduces RRT* and PRM for 3D pathfinding	Probabilistic model; no guaranteed shortest path or delivery filtering
[5]	Real-Time UAV Delivery Route Optimization Using Edge Computing	Zhao et al.	2020	Uses edge computing for UAV route decisions	No clear integration with Knapsack or 3D obstacle handling
[6]	Deep Q-Learning for 3D Drone Navigation	Shi et al.	2021	Uses RL for learning-based navigation	Requires training, lacks real-time deterministic control and value constraints
[7]	Warehouse Drone Scheduling with Constraints	Caccamo et al.	2021	Solves dispatch in warehouse environments	Focuses on scheduling, not 3D spatial navigation or visualization

✅ Base Paper: Luo et al., 2022 – "A Survey of 3D Space Path-Planning Methods and Algorithms" This work inspired our combined use of spatial planning and optimization techniques.

Objectives

Simulate Real-time 3D Drone Routing

Simulate real-time 3D drone routing with obstacles.

Implement Key Algorithms

- A* → Navigate between two points
- TSP → Optimal multi-goal visit
- Knapsack → Choose highest value deliveries under weight constraint

Build a GUI

Build a GUI using Streamlit + Plotly for rotation, metrics display.

Highlight Unreachable Cases

Highlight unreachable cases with user feedback.

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Methodology

Process Flow

1. Obstacle and grid initialization
2. Mode Selection: A* only, TSP (order optimization), TSP + Knapsack (goal + order)
3. Route Computation: A* for all legs, Total distance = ∑(segment distances)
4. Display 3D plot with: Color-coded markers for start, goal, obstacles, Route segments, Metrics panel (path length, total value, weight)

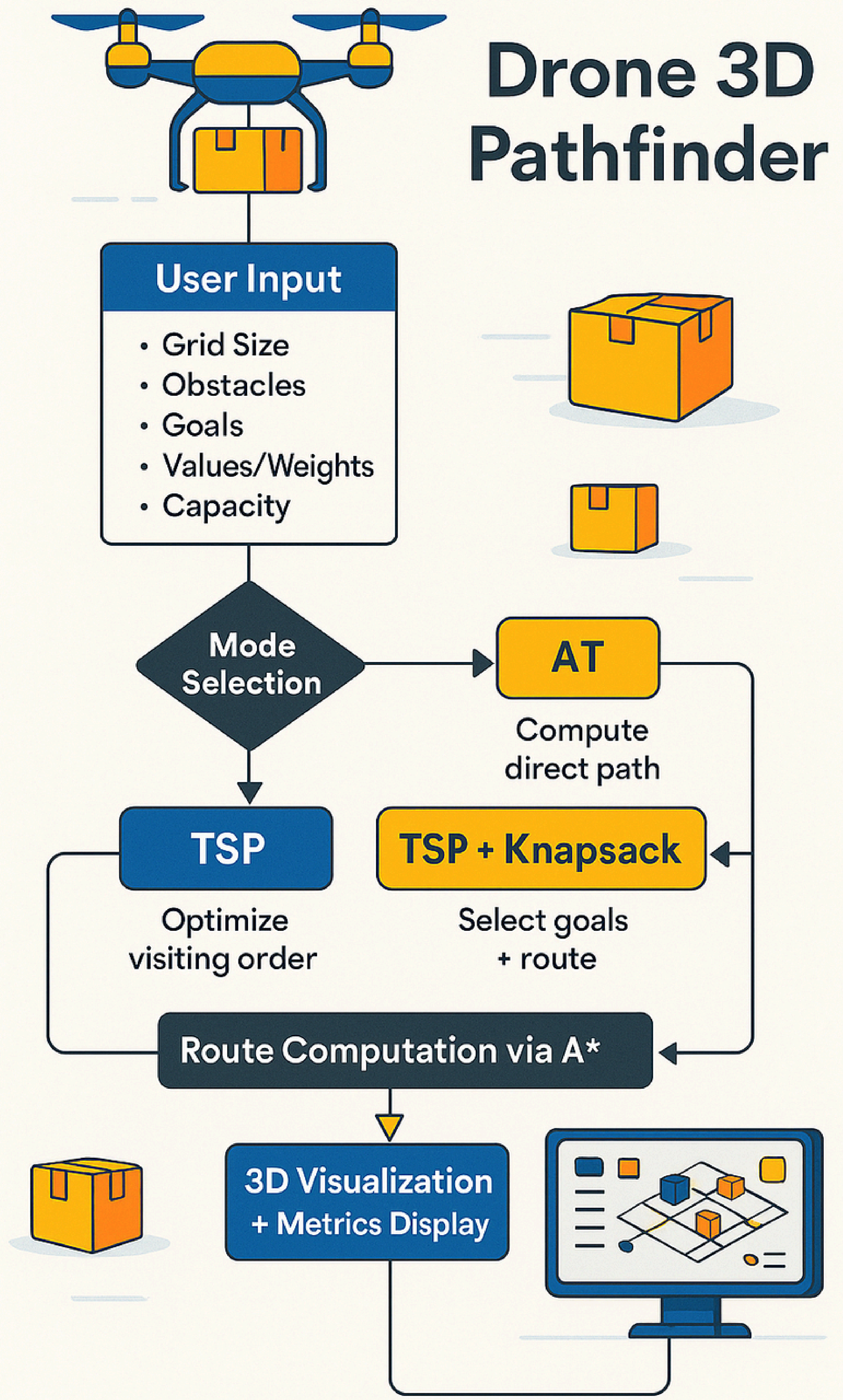
Tools

Modules:

- Python
- Streamlit (UI)
- Plotly (3D Graph)
- NumPy
- PriorityQueue

Algorithms:

- Heuristic Search (A*)
- Dynamic Programming (Knapsack)
- BnB (TSP)



Algorithms Used in Drone 3D Pathfinder

This slide outlines the algorithms used in Drone 3D Pathfinder:

A* Pathfinding:

A* finds the shortest path using a cost function:

$$f(n) = g(n) + h(n)$$

where:

- g(n) = actual cost from start node to current node
- h(n) = heuristic estimate to the goal (like Manhattan or Euclidean distance in 3D)

The algorithm always expands the node with the lowest f(n) value.

Traveling Salesman Problem (TSP) Branch & Bound:

- The goal is to find the minimum-cost cycle that visits each destination exactly once.
- The Branch and Bound method:
- Explores all possible permutations of paths (like brute force), but prunes branches.
- It's an optimization over brute-force by not traversing unpromising branches.

0/1 Knapsack Dynamic Programming:

The problem: maximize total value without exceeding weight capacity.

The core recurrence relation is:

If weight[i] ≤ capacity:

dp[i][w] = max(dp[i-1][w], dp[i-1][w - weight[i]] + value[i])

Else:

dp[i][w] = dp[i-1][w]

It builds a table to track the best value for each weight capacity and item subset.

These algorithms, when used together, make the drone system not just functional, but efficient and intelligent.

Time & Space Complexity Analysis

This slide outlines the theoretical performance of the algorithms used in Drone 3D Pathfinder:

A* Pathfinding:

- Time Complexity: O(b^d), where b is the branching factor and d is the depth of the optimal path.
- Space Complexity: O(n), where n is the number of grid cells.

Traveling Salesman Problem (TSP) Branch & Bound:

- Time Complexity: O(n!), due to factorial number of permutations.
- Space Complexity: $O(n^2)$, mainly due to distance matrix and state tree.

0/1 Knapsack Dynamic Programming:

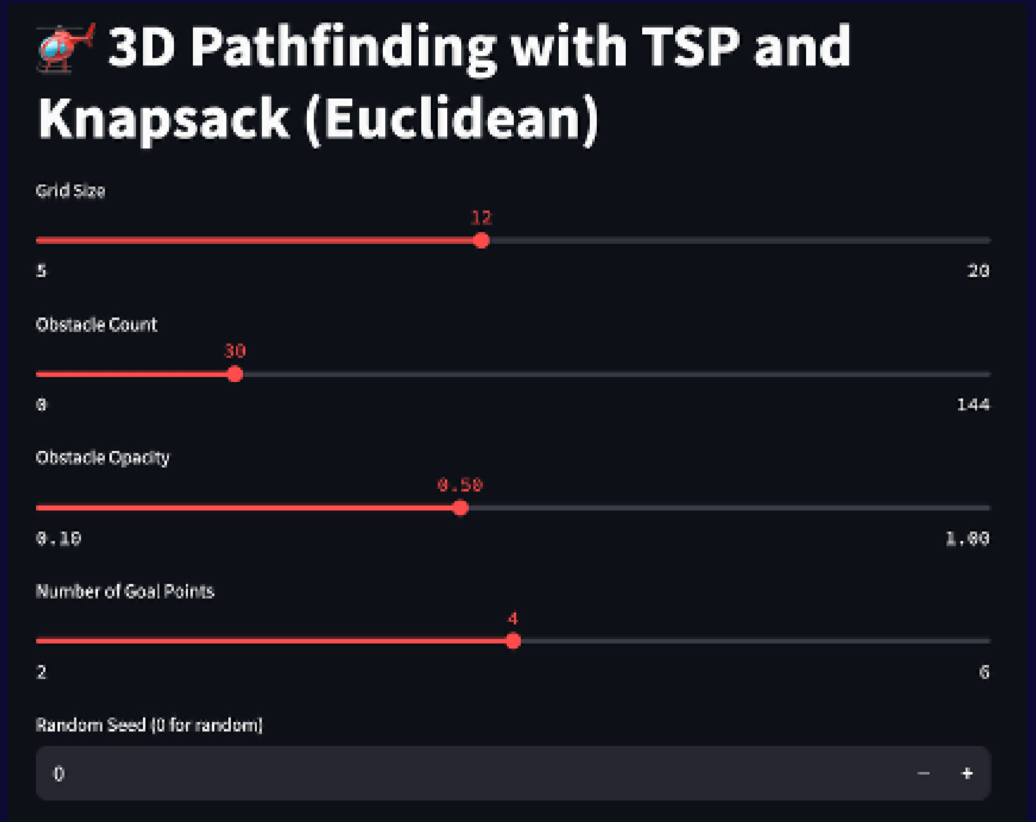
- Time Complexity: O(nW), where n is the number of items and W is the max weight capacity.
- Space Complexity: O(nW), due to the DP table used.

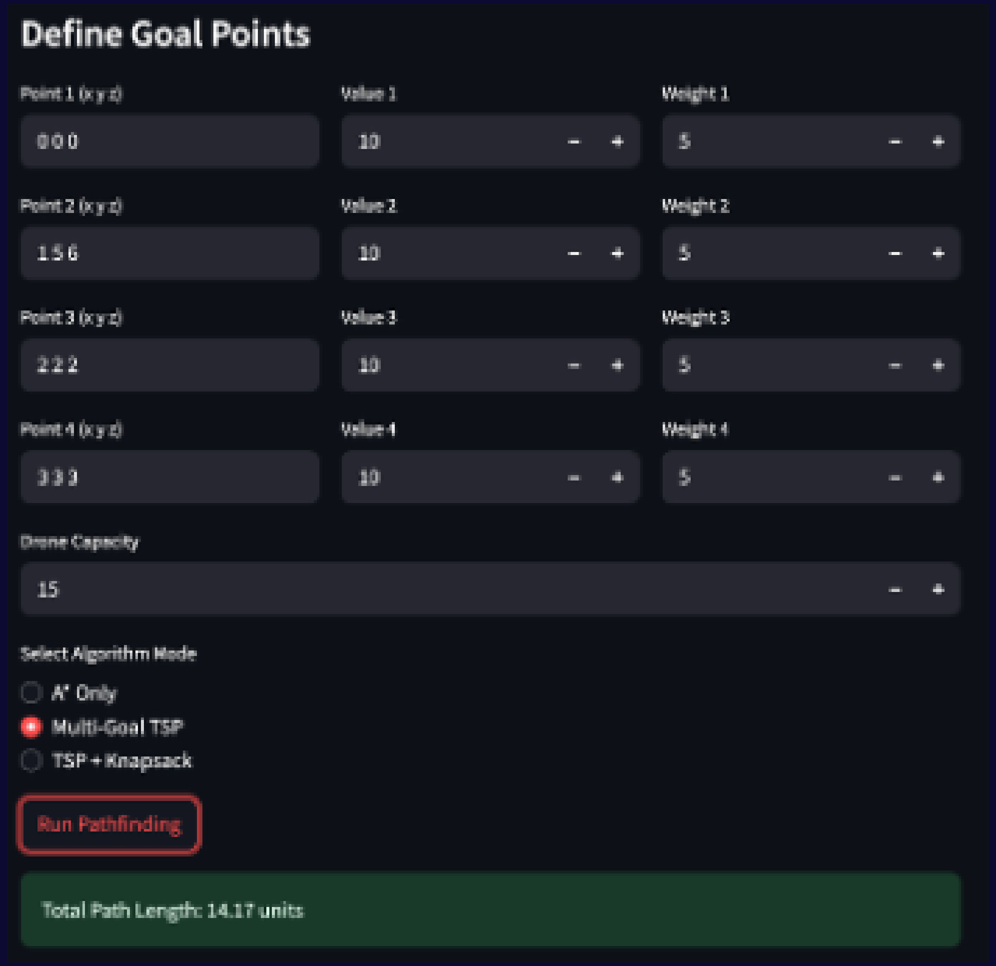
Each algorithm is chosen based on its ability to solve a specific challenge in drone navigation while keeping runtime acceptable for simulation-scale problems.

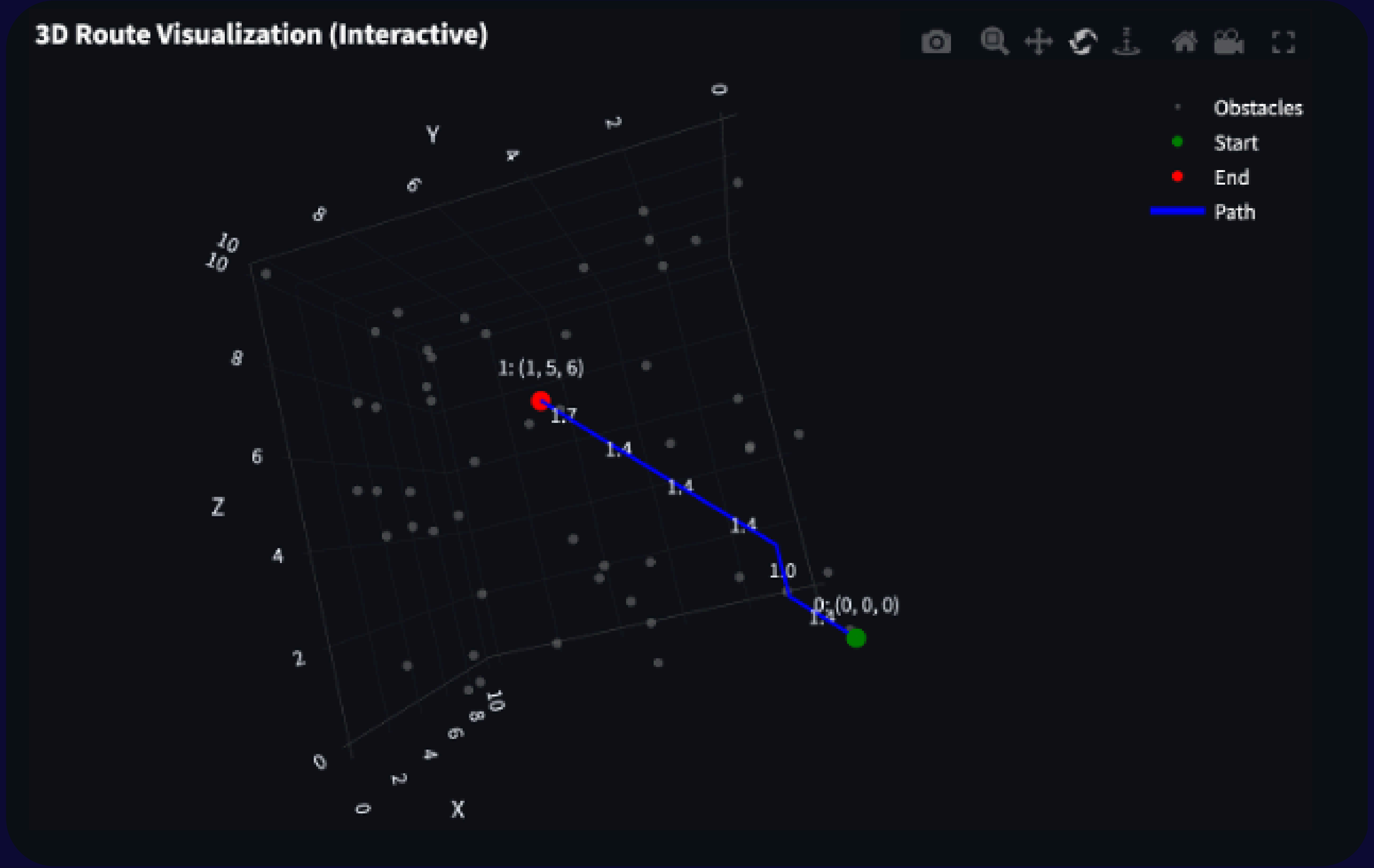
RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Results (Valid Output)

- 10x10x10 grid with 25 random obstacles
- Goals: G1, G2, G3 — selected via knapsack logic
- Path: Start → G1 → G2 → G3 → End





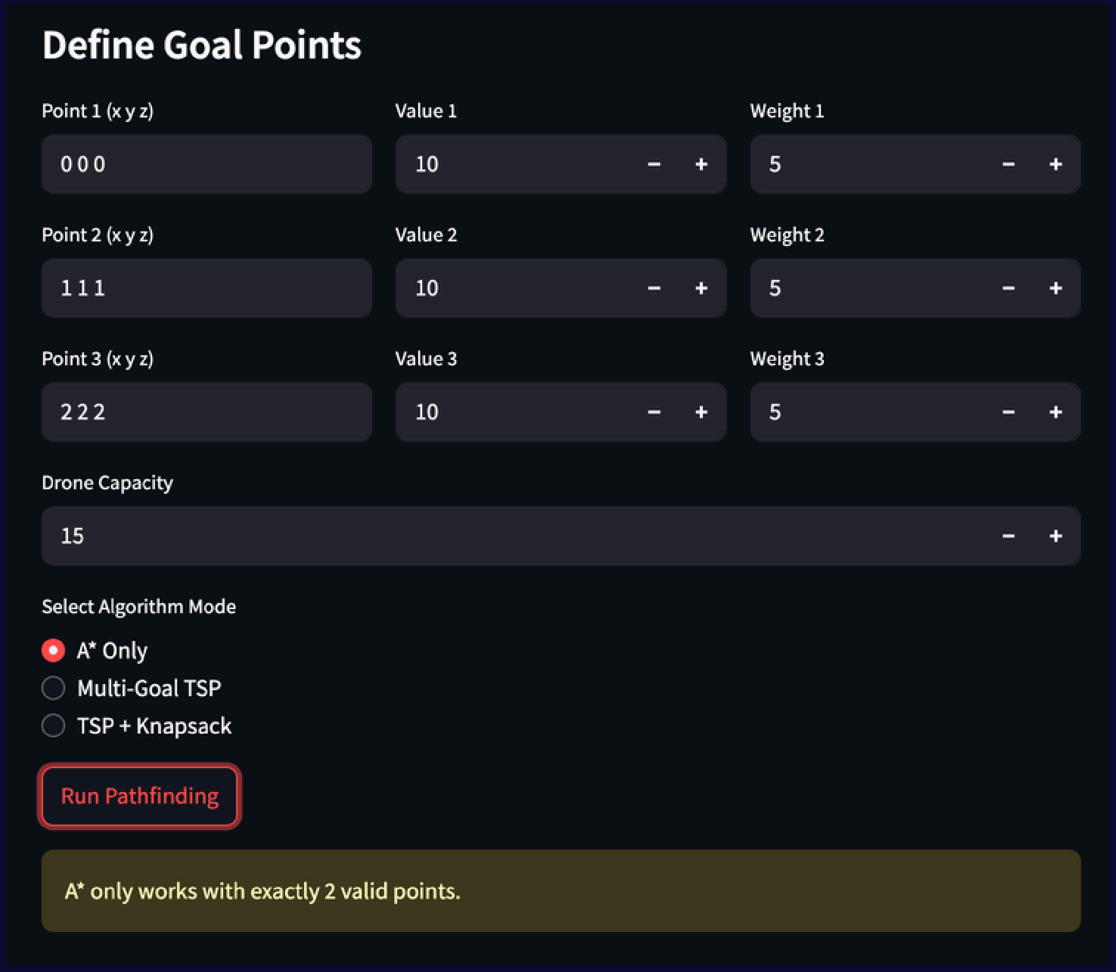


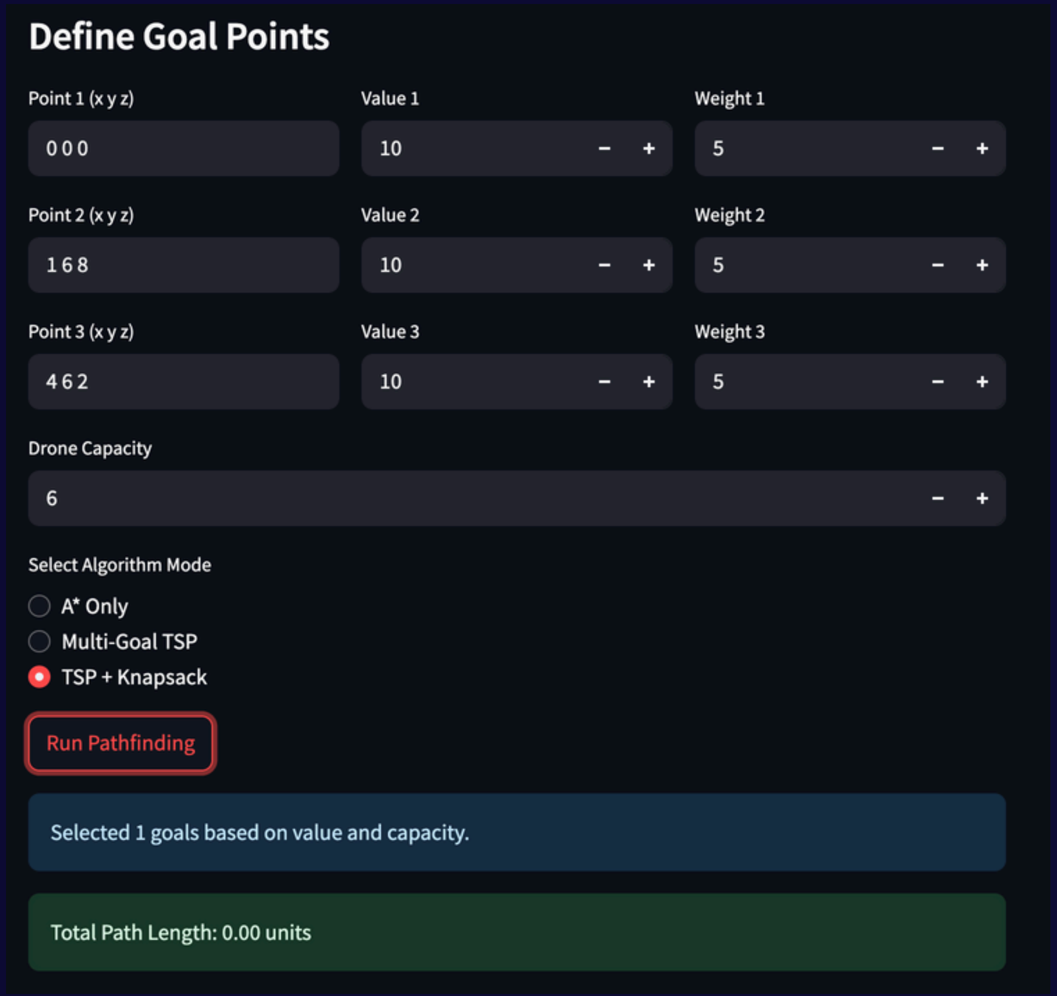
RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

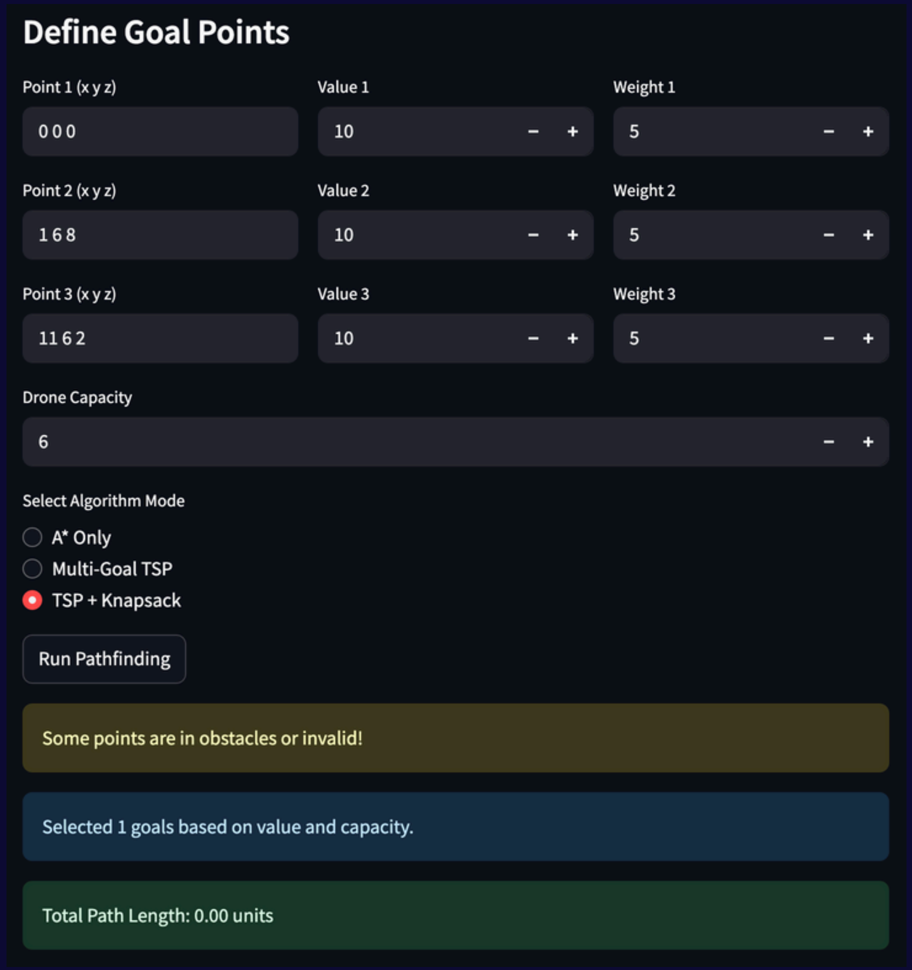
Results (Unreachable Goals Case)

- G3 unreachable due to obstacle barrier
- A* throws “No path found”
- UI marks that node as skipped
- TSP recomputes for reachable goals only

Graceful handling ensures system robustness under practical deployment-like conditions.







Conclusion

Drone 3D Pathfinder integrates three classical problems into a unified solution.

Accurate Pathfinding

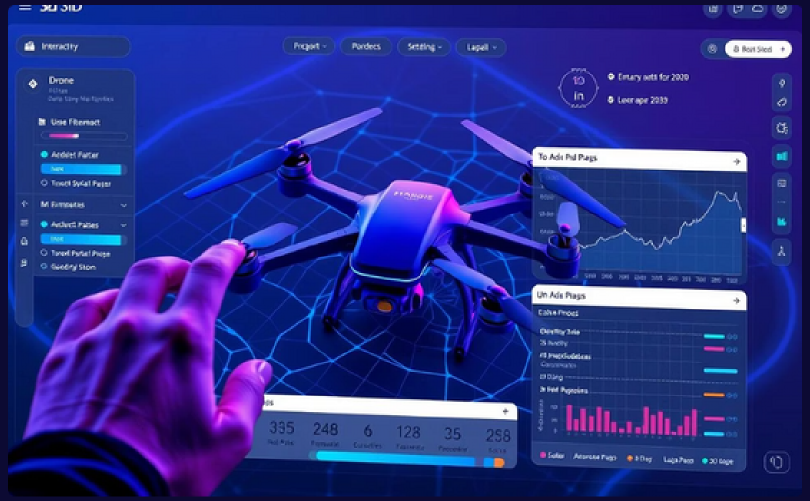
Successfully demonstrates precise pathfinding capabilities in complex 3D environments with various obstacles.

Efficient Route Optimization

Achieves highly efficient route optimization for multi-goal visits, minimizing travel distance and time.

Payload Prioritization

Implements effective payload prioritization, ensuring optimal delivery selection under defined weight constraints.



Interactive Visualization & Scalability

Enhances user understanding through interactive visualization, built on a scalable and modular architecture.

Future Enhancements

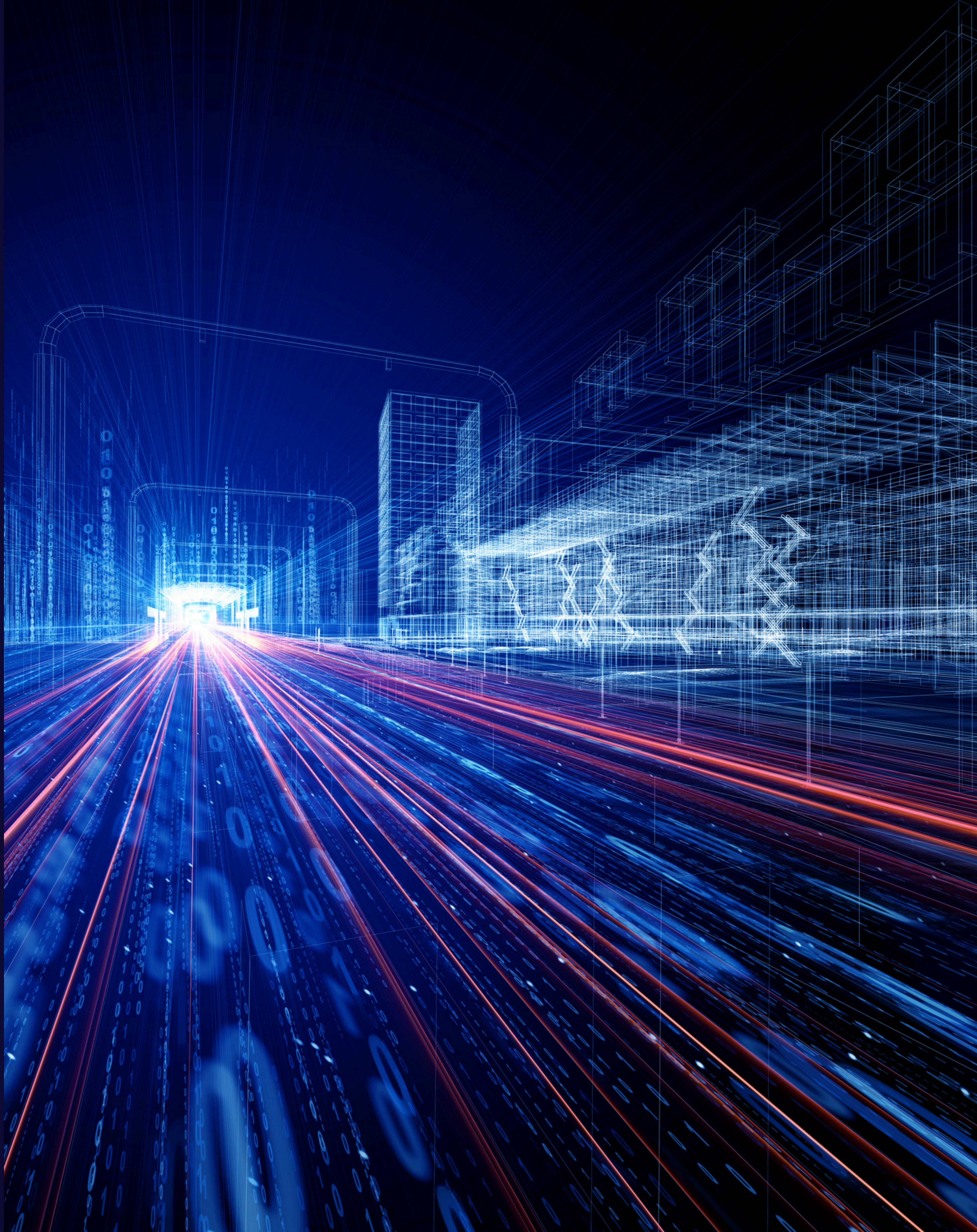
1. **Real-time GPS Feeds**
   Add support for real-time drone GPS feeds

2. **Dynamic Moving Agents**
   Replace obstacles with dynamic moving agents

3. **Multi-drone Coordination**
   Multi-drone coordination (fleet simulation)

4. **Battery-based Constraints**
   Add battery-based constraints (energy optimization)

5. **Drone Mission Planner App**
   Convert simulator to a drone mission planner app



References

[1] Z. Luo, C. Zhang, and Y. Wang, "A Survey of 3D Space Path-Planning Methods and Algorithms," Journal of Navigation and Positioning, vol. 10, no. 1, pp. 47–61, 2022.
[2] Y. Liu, Y. Qi, and Q. Zhu, "A Modified Genetic Algorithm for Solving the Multi-Objective Drone Routing Problem," IEEE Access, vol. 6, pp. 11069–11079, 2018.
[3] P. R. Rao and R. Kapoor, "An Efficient 3D Pathfinding Algorithm Using Improved A*," Procedia Computer Science, vol. 89, pp. 831–836, 2016.
[4] S. Karaman and E. Frazzoli, "Sampling-Based Algorithms for Optimal Motion Planning," The International Journal of Robotics Research, vol. 30, no. 7, pp. 846–894, 2011.
[5] X. Zhao, et al., "Real-Time UAV Delivery Route Optimization Using Edge Computing," in IEEE Transactions, 2020.
[6] J. Shi, Y. Wang, and D. Zhang, "Deep Q-Learning for 3D Drone Navigation," in Proc. of IEEE ICRA, 2021.
[7] M. Caccamo, A. Falcone, and L. Pappalardo, "Warehouse Drone Scheduling with Constraints," Robotics and Autonomous Systems, 2021.
[8] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, The Traveling Salesman Problem: A Computational Study. Princeton Univ. Press, 2007.
[9] S. Martello and P. Toth, Knapsack Problems: Algorithms and Computer Implementations. Wiley, 1990.