UAV Strategic Deconfliction in Shared Airspace

Objective

Design and implement a strategic deconfliction system that serves as the final authority for verifying whether a drone's planned waypoint mission is safe to execute in shared airspace. The system must check for conflicts in both space and time against the simulated flight paths of multiple other drones. The primary mission is defined by a series of waypoints that must be completed within an overall time window. **Extra Credit:** Extend your solution to a 4D simulation (3D spatial coordinates + time).

We expect you to use AI-assisted tools **(Claude Code, Cursor AI, Windsurf, Lovable, Replit, etc.)** to aid your development process and document how these tools helped expedite your work. The assignment is designed to be challenging yet achievable within the time limit when leveraging the latest AI tools and practices.

Scenario

A drone is scheduled to execute a waypoint mission within a specified overall time window (e.g., complete mission between *T_start* and *T_end*). The mission is represented by a series of waypoints (with spatial coordinates, and optionally altitude for 3D) that define the drone's intended route. Before takeoff, the drone queries a central deconfliction service that maintains the flight schedules of several other drones operating in the same airspace. These simulated drones have their own flight paths that may intersect with the primary drone's mission in both space and time.

Requirements

Input

1. **Primary Drone Mission:**

   - A series of waypoints defining the drone's route.
   - Each waypoint includes spatial coordinates (x, y). For extra credit (3D), include altitude (z).
   - One overall time window (start time, end time) during which the entire mission must be completed.

2. **Simulated Flight Schedules:**

   ○ A dataset (hardcoded or provided via a file) representing the flight paths of other drones.
   ○ Each simulated flight includes its own set of waypoints and associated timings so that trajectories may overlap in the spatiotemporal domain.

Functionality

● **Spatial Check:**
  Validate that the primary mission's path does not intersect with any other drone's trajectory within a defined safety buffer (minimum distance threshold).

● **Temporal Check:**
  Ensure that, within the overall mission window, no other drone is present in the same spatial area during overlapping time segments.

● **Conflict Explanation:**
  When conflicts are detected, return a detailed explanation indicating:

   ○ The location(s) and time(s) of conflict.
   ○ Which simulated flight(s) caused the conflict?

● **Query Interface:**
  Provide a simple interface (e.g., a Python function) that accepts the primary drone's mission and returns a status (e.g., "clear" or "conflict detected") along with conflict details.

● **Simulation & Visualization:**
   ○ Generate simulation graphs or animations that visually depict:
      ■ The primary drone's waypoint mission.
      ■ Trajectories of the simulated drones.
      ■ Highlighted areas and time segments where conflicts occur.
   ○ Produce videos or a series of plotted graphs to demonstrate multiple scenarios:
      ■ A conflict-free mission.
      ■ Cases where conflicts are detected and explained.
   ○ **Extra Credit:** Include 4D visualization (3D space + time) to show the spatio-temporal evolution of conflicts.

● **Scalability Discussion:**
  In your reflection document, explain what it would take for such a system to handle real data from tens of thousands of commercial drones. Outline the architectural changes and enhancements (e.g., distributed computing, real-time data ingestion pipelines, fault tolerance, scalability of conflict resolution algorithms) that would be necessary to support a large-scale deployment.

Deliverables

1. **Code Repository:**

   ○ A self-contained Python (or any other language) solution implementing the deconfliction system and simulation of multiple drone trajectories.
   ○ The code should be modular and well-documented.

2. **Documentation:**

   ○ **README:** Clear setup and execution instructions.
   ○ **Reflection & Justification Document (1–2 pages):**
      ■ Discuss your design decisions and architectural choices.
      ■ Explain how spatial and temporal checks were implemented.
      ■ Describe any AI integration, if applicable.
      ■ Provide a testing strategy and discuss edge cases.
      ■ Explain what would be required to scale the system to handle real-world data from tens of thousands of drones.

3. **Demonstration Video:**

   ○ A 3–5 minute video **WITH VOICEOVER** that demonstrates:
      ■ The working system in action.
      ■ Simulation graphs/animations of various scenarios (both conflict-free and conflict-present).
      ■ Clear explanations of how the deconfliction system identifies, resolves, and explains conflicts.
      ■ **Extra Credit:** A section showcasing any 4D (3D space + time) visualization features.

Evaluation Rubric

A. Code Quality and Architecture (35 points)

● **Modularity and Structure (10 points):**
  Code is logically organized into modules (e.g., data ingestion, spatial check, temporal check, visualization).
● **Coding Standards (10 points):**
  Adheres to industry standards with clear naming, formatting, and maintainability.
● **Architectural Decisions (10 points):**
  Design choices are well-justified, emphasizing scalability and robustness.
● **Readability and Comments (5 points):**
  Inline documentation that aids in understanding complex logic.

B. Testability and Quality Assurance (25 points)

- **Test Case Design (10 points):**
  Comprehensive tests covering various conflict scenarios.
- **Test Automation (5 points):**
  Use of automated testing scripts where applicable.
- **Robustness and Error Handling (5 points):**
  Proactive handling of potential failure modes and edge cases.
- **QA Thoughtfulness (5 points):**
  Evidence of iterative testing and quality assurance practices.

C. Effective Use of AI and Resourcefulness (20 points)

- **Innovative Use of AI (10 points):**
  If AI is used, it is integrated meaningfully to enhance the solution.
- **Critical Evaluation of AI Output (5 points):**
  Clear explanation of how AI contributions were validated and refined.
- **Self-Driven Learning (5 points):**
  Demonstrates rapid assimilation of new tools or concepts in solving challenges.

D. Documentation, Reflection, and Communication (20 points)

- **Solution Demonstration Video (10 points):**
  Quality and clarity in presenting the system, emphasizing design choices and simulation outcomes.
- **Reflection and Justification Document (10 points):**
  Thoughtful explanation of design decisions, AI usage, scalability discussion, and handling of edge cases.
- **Extra Credit for 4D Visualization:**
  Additional points will be awarded for solutions that successfully extend the simulation into 3D space with altitude as an extra dimension.