Large-Scale Simulation of a Distributed Sensing Network Supporting Regional Urban Air Mobility Operations

Keerthana Kannan*, Joshua Baculi†
KBR, NASA Ames Research Center, Moffett Field, CA 94035

Corey Ippolito‡
NASA Ames Research Center, Moffett Field, CA 94035

Vahram Stepanyan§
KBR, NASA Ames Research Center, Moffett Field, CA 94035

Thomas Lombaerts¶
KBR, NASA Ames Research Center, Moffett Field, CA 94035

Evan Kawamura‖
NASA Ames Research Center, Moffett Field, CA 94035

Urban Air Mobility (UAM) is set to transform transportation in densely populated regions like the San Francisco Bay Area. This paper introduces an innovative simulation approach to explore large-scale UAM scenarios, emphasizing the use of distributed sensing to enhance operational efficiency and safety. The Revolutionary Vertical Lift Technology (RVLT) model is employed as the framework for simulating complex interactions among multiple vehicles within urban landscapes. Strategically deployed ground sensor nodes enable distributed sensing, enhancing situational awareness and operational effectiveness. By integrating empirical data and geographical realism, the simulations provide a systematic analysis of the feasibility, efficiency, and safety considerations associated with UAM deployment in urban environments. Factors such as air traffic density and infrastructural requirements are thoroughly examined, offering actionable insights for policymakers and industry stakeholders. This paper aims to refine the structure and scenarios for large-scale simulations based on distributed sensing, thereby contributing to the advancement of UAM operations.

I. Introduction

Advanced air mobility (AAM) operating concepts, including frameworks for Urban Mobility Level (UML) 4+ and m:N operations, face significant barriers, particularly in autonomy, sensing, and perception[1]. UML 4+ operations represent advanced stages of UAM, where sophisticated systems for autonomy, sensing, and distributed network operations are essential for managing complex urban airspace and ensuring the safety and efficiency of high-density, autonomous flight operations. These challenges are not readily addressed by conventional techniques. The traditional monolithic approach to aviation system design, prevalent in current state-of-the-art aviation, may be insufficient for scaling to meet the needs of these innovative aviation markets.

To address these challenges, Distributed Sensing is being explored as a promising avenue. This research investigates the potential for overcoming barriers and addressing the complexities of advanced AAM and UAM concepts through distributed technologies and techniques. These technologies are associated with future interconnected systems of smart-enabled vehicles and airspaces that utilize wireless sensing networks and distributed design principles [2, 3].

Initial barriers for m:N and UML 4+ operations being investigated include:

---

*Embedded Software Developer, Intelligent Systems Division, Mail Stop 269-1, AIAA member.
†Systems Engineer, Intelligent Systems Division, Mail Stop 269-1.
‡Aerospace Scientist, Intelligent Systems Division, Mail Stop 269-1, AIAA member.
§Technical Advisor, Intelligent Systems Division, Mail Stop 269-1, AIAA Senior Member.
¶Aerospace Research Engineer, Intelligent Systems Division, Mail Stop 269-1, AIAA Associate Fellow.
‖Computer/GNC Engineer, Intelligent Systems Division, Mail Stop 269-1, AIAA member.

- **Airspace Surveillance and Conformance Monitoring:** This involves surveillance over low-altitude urban airspace corridors and vertiport terminal area airspaces, including independent airspace monitoring services for airspace control and m:N fleet management.
- **Precision Navigation:** This encompasses Precision Approach and Landing (PAL) and GPS-Free Position Navigation and Timing (PNT), supporting critical onboard services that traditional avionics struggle to achieve.
- **Autonomous Airspace Operations Services:** This includes services that support autonomous m:N fleet operations management and onboard self-autonomy services, such as onboard detect-and-avoid (DAA) services and decentralized autonomous airspace operations, including self-deconfliction, self-separation, self-sequencing, and self-spacing services.
- **Distributed Monitoring of Urban Environment and Weather Hazards:** This involves distributed networks for detecting local urban weather hazards, weather-related mapping services, and support for mitigation planning services essential for m:N fleet operations.
- **Distributed Systems Health Management:** This includes networks of sensors for assessing vehicle health, regional air traffic system health monitoring services, m:N fleet management monitoring services, and cybersecurity detection and mitigation services.
- **Onboard Aircraft Distributed Wireless Sensor Networks:** This involves onboard networks for air flow monitoring, structural monitoring of flexible structures, and distributed propulsion sensing/control.

By leveraging distributed sensing technologies, it is anticipated that the significant barriers and challenges to the realization of advanced AAM operating concepts can be addressed more effectively, paving the way for the successful implementation of these innovative aviation systems.

UAM is poised to revolutionize transportation in metropolitan areas, offering efficient, sustainable, and rapid aerial transportation solutions [4–7]. Simulation scenarios play a crucial role in understanding and optimizing UAM operations, particularly in complex urban environments like the San Francisco Bay Area. This paper presents a novel approach to simulation scenarios, which seamlessly transition between multi-vehicle and single-vehicle settings to capture the diverse dynamics of UAM operations.

Drawing upon the concepts of AAM [8], with a specific focus on UAM, the study leverages the RVLT model to simulate flights within the San Francisco Bay Area. The RVLT model, renowned for its accuracy in capturing vehicle dynamics and environmental factors, serves as a robust framework for simulation scenarios.

The characteristics and challenges of large scale multi-vehicle scenarios within the context of UAM operations are delineated. By incorporating real-world data and geographical features of the San Francisco Bay Area, the simulations provide insights into the feasibility, efficiency, and safety implications of UAM deployment.

Furthermore, the impact of various factors such as air traffic density, weather conditions, and infrastructure constraints on UAM operations in large-scale multi-vehicle scenarios is analyzed. Through comprehensive scenario-based simulations, the aim is to inform policymakers, urban planners, and industry stakeholders about the potential benefits and challenges associated with integrating UAM into urban transportation networks. This paper also explains the simulation architecture and the software structure of these large-scale scenarios.

In summary, this paper contributes to the growing body of research on UAM using distributed sensing by presenting versatile simulation scenarios tailored to the unique characteristics of the San Francisco Bay Area. By employing the RVLT model and considering large-scale multi-vehicle dynamics, valuable insights are provided to guide the development and implementation of UAM systems in urban environments.

II. Prior Work Overview

In the simulation paper [9], the team focused on constructing scenarios set within the San Francisco environment, limiting the implementation to just 2 vertiports for take-off and landing, and considering only a single flight in the simulation. The scenario involved the placement of approximately 7-8 sensors distributed along the route, with the pilot view camera onboard being the primary source of imagery, capturing scenes at the vertiport with the landing lights [10–12]. This initial setup served as a foundational framework for subsequent large-scale conversions. The current iteration of the project expands significantly, with around 50 vehicles simultaneously navigating to various destinations, each servicing nearly 22 established flight veriports. To support these expanded simulations, the team plans to deploy approximately 60-70 sensor ground stations, ensuring comprehensive coverage throughout the urban area. As a result, this paper marks a notable advancement over the previous work, with enhancements including a transition from a 1:1 vehicle to vertiport ratio to a more intricate M:N configuration, alongside an augmented sensor array and vehicle fleet. These improvements result in the creation of more realistic and detailed scenarios, particularly evident in the enhanced approach and landing procedures depicted in the current simulation. Hence the major requirements for the large-scale simulation are more number of vehicles, vertiports, routes, sensor stations, safety considerations and measures. Figure 1 displays the flight path alongside ground sensor placements. Meanwhile, Figure 2 serves as a reference to a 2023 paper's future steps, aiming to establish a densely populated sensor network. Figures 3 and 4 visually depict how the flight trajectory curves through urban areas with tall buildings. These images illustrate how the simulation accounts for urban infrastructure when determining the flight path.

Fig. 1 Route taken and sensors from prior work[9]

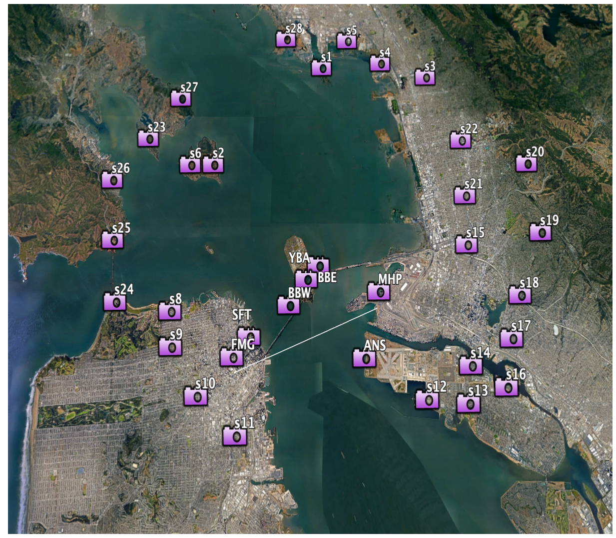


Fig. 2 Prospective endeavors explored in prior work[9]

Fig. 3 Curved trajectory from MHP to FMG

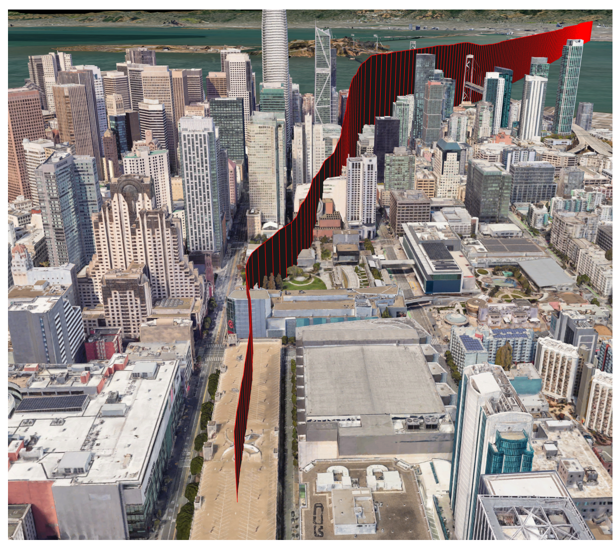


Fig. 4 Approach and Departure path in urban setting

The primary sensors considered for distributed sensing ground nodes encompass visual cameras, IR cameras, Lidars, radars, and multispectral cameras. Additionally, similar sensors are installed onboard to capture in-flight parameters and record flight trajectories [13]. The strengths and weaknesses of each sensor are outlined in table 1.

The other progressive improvement in this large scale simulation also includes the emulation of these sensors under consideration. Simulation helps is identifying the limitations of these thus saving money is real-time flight test failures.

Table 1 Sensor Pros and Cons

Sensor Type	Strengths	Limitations
Visual Camera	Affordable, easy to mount and just like human vision, they can easily distinguish shapes, colours, and quickly identify the type of object based on such information	Poor vision under extreme weather conditions
Radar	Resilient to weather conditions and great ability to detect and forecast moving objects in a vehicle's path	Struggles to precisely model object shapes, leading to identification challenges, Interference between multiple radar systems is a constraint due to radio wave interference, and Urban metal infrastructure can hinder radar waves, affecting performance
Lidar	Measures thousands of points simultaneously, enabling precise 3D modeling of the surrounding environment	Expensive, demands substantial computing power compared to cameras and radar, and susceptible to system malfunctions and software glitches

III. Single Flight Simulation Architecture

Figures 5 and 6 illustrate the simulation configuration for individual flights within the Reflection platform. It's important to note that this configuration is tailored for a single flight. However, in the current simulation setup, these configurations are replicated across all flights to emulate an ideal scenario where all flights are presumed to be identical in design and operation.

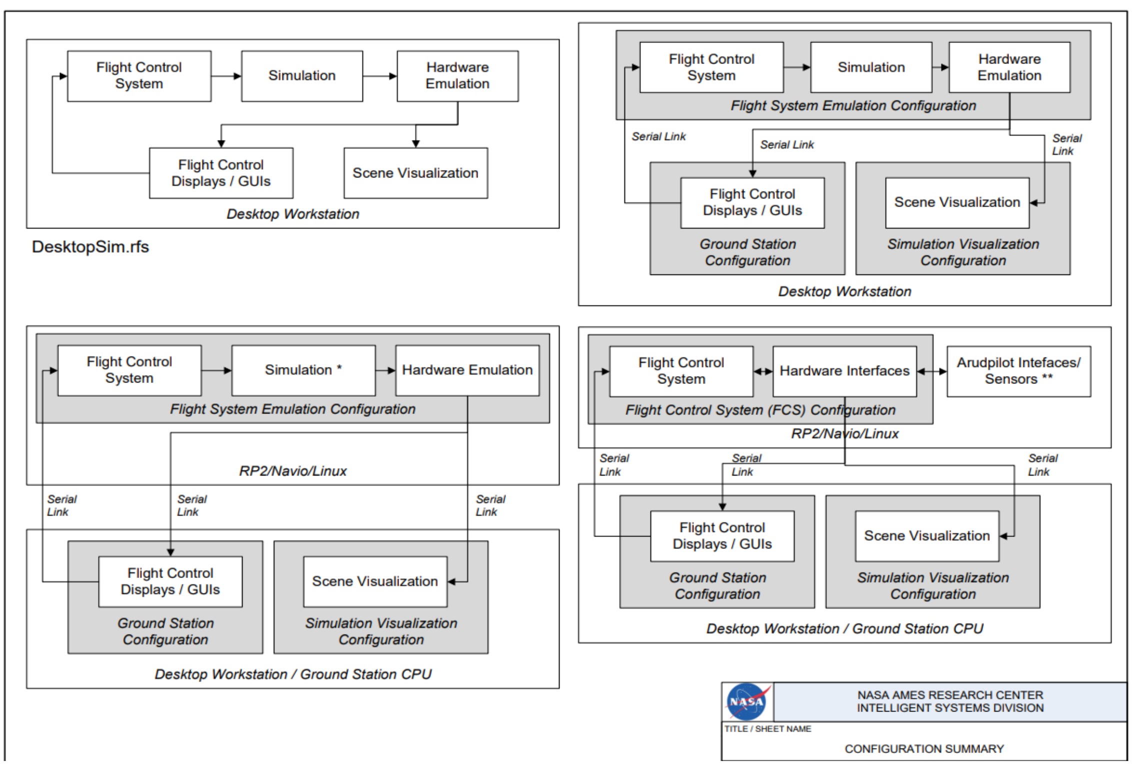


Fig. 5 Simulation Configuration[14]

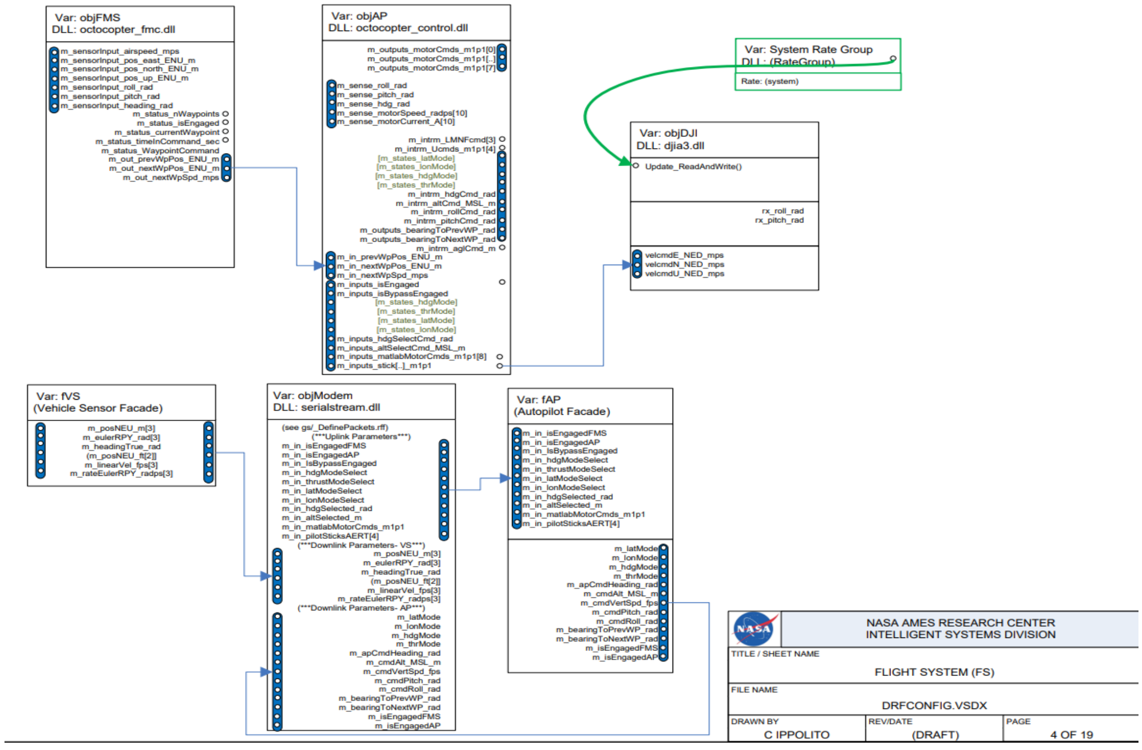


Fig. 6 Flight System Configuration[14]

This replication process involves duplicating the flight management system multiple times to ensure consistency across the simulated flights. By adopting this approach, the simulation aims to simulate a standardized environment where each flight adheres to the same set of parameters and operational procedures.

In essence, this methodology allows for the exploration of scenarios where uniformity among flights is assumed, enabling a focused analysis of system behavior and performance under controlled conditions. This simplification strategy is commonly employed in simulation studies to streamline the modeling process and facilitate comparative analysis across multiple scenarios.

A. Camera Emulation

Simulating videos for over 50 cameras dispersed across the expansive San Francisco area presents practical challenges. Additionally, the visualization software XPlane, while proficient, exhibits limitations when handling multi-vehicle scenarios. To overcome these obstacles and improve simulation fidelity, transitioning to more versatile platforms like MATLAB or Reflection becomes imperative. Leveraging the capabilities of these software, the simulation can efficiently emulate sensor outputs from each camera perspective, enabling realistic detection of flight activities. By harnessing MATLAB's computational prowess or Reflection's advanced modeling capabilities, the simulation gains the capacity to accurately replicate the intricacies of sensor data acquisition, enhancing the overall accuracy and effectiveness of the simulation framework. This shift not only addresses the constraints posed by XPlane but also opens avenues for more sophisticated analysis.

B. Radar Emulation

Placing hundreds of radars in an urban location filled with metal barriers and tall buildings without a proof of concept is impractical. Financially and logically, it does not make sense to implement such a setup in flight tests without understanding nuances such as potential radar signal interference and the optimal distance between radars. Therefore, emulating the radar is considered the best option, allowing error correction to be performed in the simulation before field deployment. This approach is especially crucial for radar systems, as incorrect usage can pose security threats.

The current radar model is designed to output four values: a flag indicating 1 if the target is detected and 0 if it is not detected, as well as the azimuth, elevation, and rho values. The algorithm first calculates the relative position between the target and the radar, computes the true azimuth, elevation, and range (rho), and then adds noise to these values. It checks if the target is within the field of view (FOV) and sets the output values accordingly. If the target is not in view, the azimuth, elevation, and range values are set to NAN, and it is indicated that the target is not in view. Figure 7 shows the flowchart of this emulator written in Reflection.

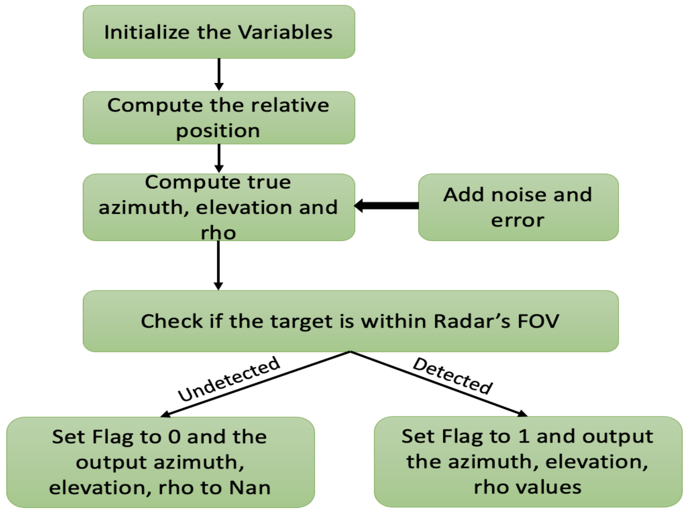


Fig. 7 Flow Diagram of the Radar Emulation

C. RVLT Flight Dynamic Model

1. Mechanism of the Battery-Powered Electric Drive Quadrotor

The battery-powered electric quadrotor is a promising design for UAM, exploring the potential of electric propulsion in a multi-rotor configuration. This approach offers advantages such as lower environmental impact, quieter operation, and reduced maintenance needs compared to conventional mechanical systems. Ref. [15] provides an overview of the key features, as well as the technology, trade-offs, and sensitivities associated with the NASA UAM quadrotor aircraft.

In this design shown in fig.8, the quadrotor is equipped with four electric motors, each positioned at the ends of the aircraft's booms. These motors are connected to local gear reduction units, which step down the high-speed, low-torque output of the motors to the required speed and torque for the rotors. These local gearboxes then feed power into a central combining gearbox, which distributes the energy to each rotor, ensuring an efficient and balanced power system [16].

The quadrotor employs collective control, which integrates the motors into a single propulsion group. This means that all motors share power and are interconnected, enabling the aircraft to maintain propulsion even if one motor fails. In the event of a motor malfunction, the remaining motors can compensate, ensuring the rotors still receive power and the aircraft can continue flying safely. This system provides redundancy, a crucial feature for ensuring reliability and safety.

One of the challenges with this design, particularly in forward flight, is managing the power demands between the front and rear rotors. Due to aerodynamic reasons, the rear rotors often require more power than the front ones. To solve this, the electric quadrotor uses identical motors for all four rotors. This uniformity allows for better power distribution and simplifies the design by eliminating the need for different-sized motors. With a single motor type across the entire system, the design becomes more cost-effective and efficient, and the aircraft can achieve higher speeds without the risk of overloading any individual motor.

The battery capacity is scaled to accommodate a larger payload, with the quadrotor capable of carrying up to six passengers. The design requires larger rotors and more powerful motors to support the increased weight and size of the aircraft. The battery system, typically made of high-capacity lithium-ion cells, stores the electrical energy required to power the motors. The power is transmitted through wiring and interconnects to each motor, ensuring that the aircraft can meet its power requirements during flight.

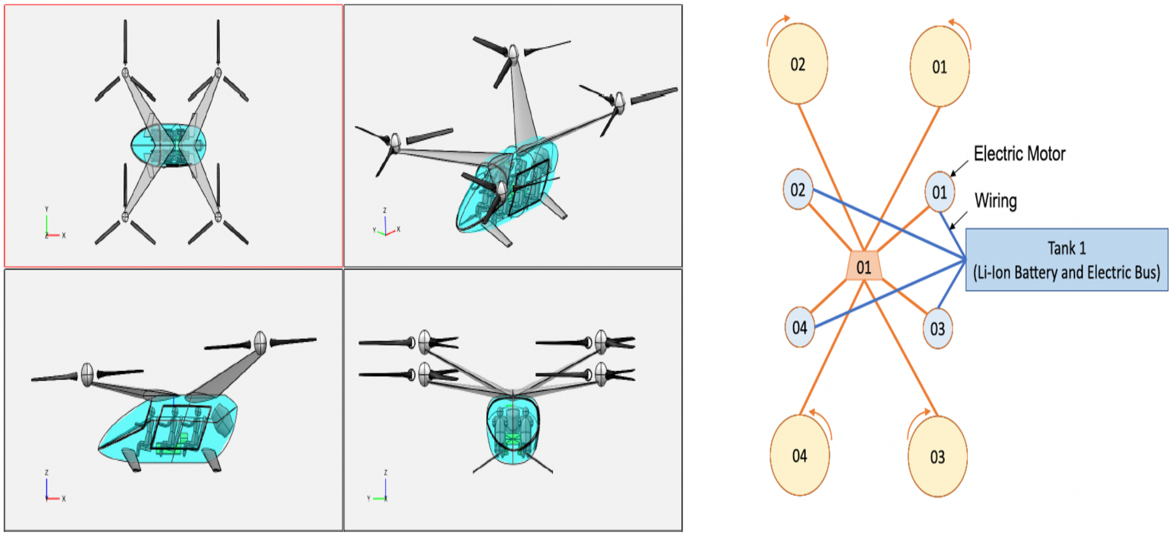


Fig. 8 Quadrotor concept vehicle (left) and Representation of Electric (right) Power Topologies for Quadrotor [16]

While the shift to electric propulsion provides clear environmental benefits, such as reduced emissions and noise, it also presents challenges. Electric motors tend to have lower energy density compared to conventional fuel systems, which can limit the aircraft's range and endurance. Therefore, the quadrotor design needs to balance power consumption with energy efficiency to ensure that it can complete its mission profile without running out of power prematurely.

The propulsion system integrates a combining gearbox, which helps distribute power evenly across the rotors and improves efficiency. This gearbox consolidates the power distribution into a single unit, reducing the complexity of the system and minimizing the number of critical components that could fail. This design is aligned with the principles of distributed electric propulsion (DEP), which enhances safety and performance by using multiple motors and rotors that can share the workload, providing greater redundancy.

In conclusion, the battery-powered electric quadrotor is an exciting development for UAM. It combines the benefits of electric propulsion with a straightforward and reliable power distribution system, offering a safe, efficient, and scalable solution for UAM. As technology continues to evolve, this design could serve as a model for future electric aircraft, demonstrating how electric propulsion can be scaled up to meet the needs of more complex and larger UAM missions while remaining environmentally friendly and cost-effective.

The figure 9 shows the working of the RVLT model. The flow of the RVLT mechanism begins with initializing the system by setting up the necessary parameters and loading external data, such as state vectors, mass values, and lookup tables. Once initialized, the flow proceeds to the Force Calculation step, where aerodynamic forces and moments, as well as gravitational effects, are computed based on the system's current state. These forces then feed into the Model Dynamics phase, where the equations of motion (EOM) are applied to calculate the state derivatives, representing how the system's state changes over time. The system's rigid-body dynamics and gravity effects are also taken into account here. If feedback control is involved, the flow moves to the optional Controller Logic step, where the system adjusts its behavior based on the error between the desired and actual states, computing control inputs to guide the system towards its target state. Finally, the outputs of the system including forces, moments, and state derivatives are returned as the results of the simulation or control process. These outputs can then be used for further analysis or real-time decision-making, completing the flow.

D. Flight Trajectory Planning and Generation

Trajectory planning is an essential part of unmanned aerial vehicle (UAV) mission management. It ensures that a UAV follows a pre-defined or dynamically calculated flight path from its departure point to its destination while optimizing for factors such as speed, fuel consumption, safety, and mission-specific constraints. The process involves defining waypoints (specific geographic locations that guide the UAV along its flight path), generating a flight trajectory, and ensuring the UAV adheres to the planned route during flight. This trajectory planning system accommodates

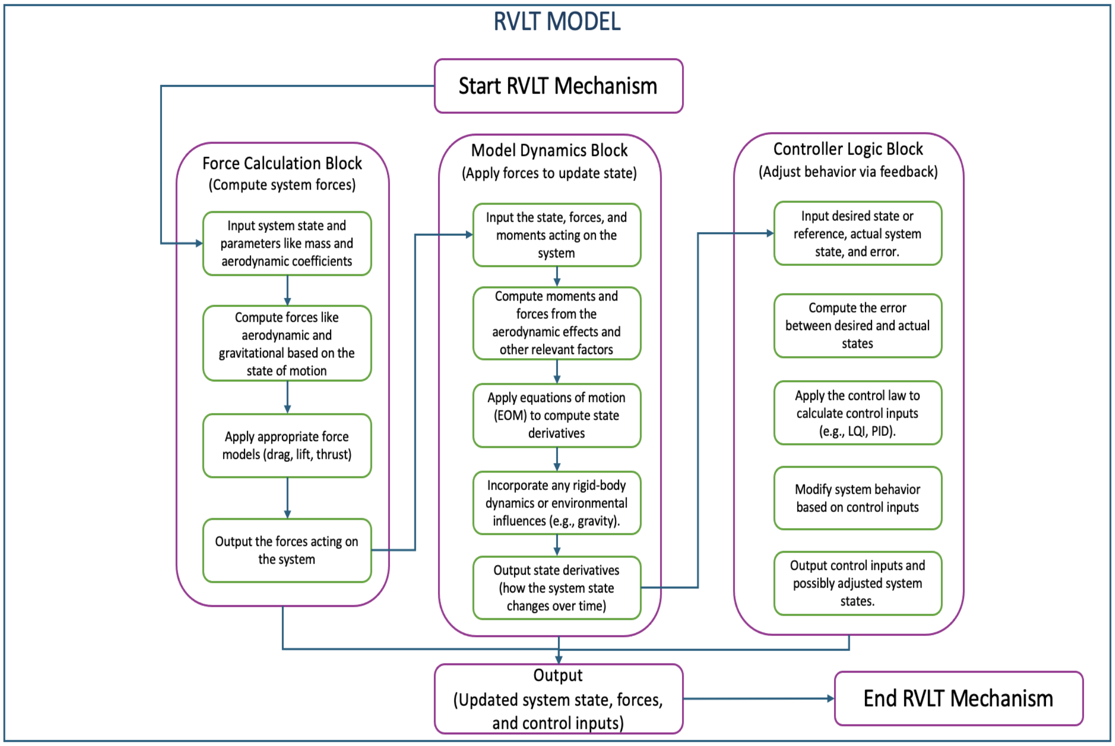


Fig. 9 RVLT Mechanism

dynamic adjustments to the UAV's path, allowing for real-time updates based on environmental conditions or mission changes.

1. Waypoint Definition and Loading

The trajectory planning process begins by defining a series of waypoints that the UAV must follow. These waypoints are critical for ensuring that the UAV stays on course throughout its mission. Waypoints can be specified either in geographic coordinates (latitude, longitude, altitude) or in local Cartesian coordinates, depending on the mission's requirements. The UAV system can accept waypoint data from various sources, such as static flight plans, dynamically generated paths, or user-defined input. A common format for representing waypoint data is JSON, which provides a structured way to store and transmit waypoint information.

For instance, the system uses functions like ParseTrajectoryFile to load waypoint data from a file, such as a JSON trajectory file. This file contains information on each waypoint, including position, altitude, speed, and any other relevant parameters (e.g., waypoint radius). The UAV system parses this data, converting it into usable flight plan data, which is then stored in memory for later use.

The system's Parse Trajectories function is specifically designed to read this JSON data and extract critical parameters such as the waypoint ID, position (in terms of North-East coordinates or latitude and longitude), altitude, airspeed, and other mission-specific parameters. These waypoints are then passed to the Flight Management Controller (FMC), which guides the UAV along the defined route.

2. Path Calculation and Optimization

Once the waypoints are defined and loaded, the next step in trajectory planning is to calculate the path that the UAV will follow. This often includes optimizing the trajectory to minimize travel time, fuel consumption, or energy usage while avoiding obstacles and adhering to operational constraints (e.g., airspace restrictions, no-fly zones). In the provided example, the system generates a combinatorial array of paths between departure and arrival vertiports (specific UAV take-off and landing locations), which is useful for defining possible flight routes.

The trajectory planning system considers the Euclidean distance between waypoints to determine the optimal route,

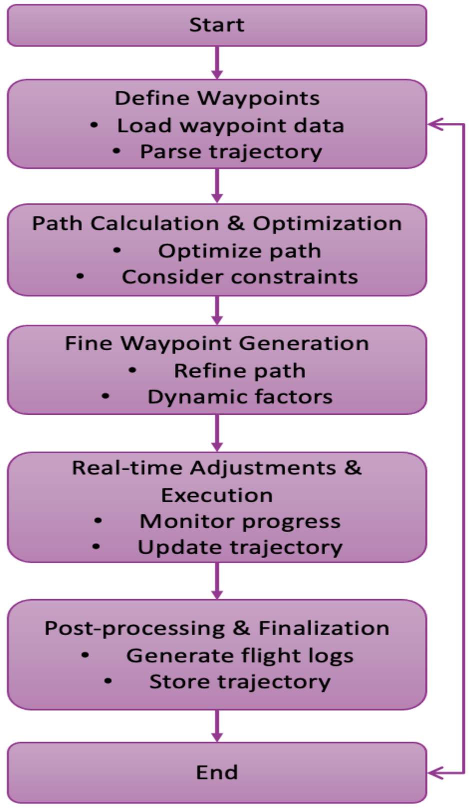


Fig. 10 Flow of Trajectory Planning and Generation

and path calculations are based on factors like the geodesic distance between two points (taking into account the curvature of the Earth). In the script, the function computes the relative North-East position of one waypoint with respect to another based on their latitude and longitude. This data is then used to calculate the total distance and estimated travel time.

3. Generating Fine Waypoint Trajectories

After calculating the initial path, finer waypoints are generated for the detailed trajectory. This step ensures smooth transitions between waypoints and accounts for various mission-specific factors such as altitude changes, speed adjustments, and time intervals. The flight system also considers various dynamic factors, such as real-time weather conditions, wind speed, and air traffic, which might necessitate adjusting the trajectory. For example, wind conditions might change a UAV's speed or cause deviations from the initial path, which can then be corrected by recalculating intermediate waypoints.

The trajectory generation involves defining parameters such as height, airspeed, timestamps, and the coordinates (latitude and longitude) for each waypoint. These values are then written into a JSON payload, which can be sent to a trajectory planning service to request a more precise path.

4. Real-time Adjustments and Execution

Once the fine trajectory is generated, the UAV's Flight Management System receives the detailed flight path and continuously monitors the UAV's progress. As the UAV follows the path, real-time updates ensure that the UAV stays on course by adjusting speed and altitude to match the planned trajectory. In addition, if any unexpected events occur (such as airspace congestion or a sudden change in weather), the system can dynamically adjust the UAV's path to ensure the mission is completed safely and efficiently.

For example, the script includes a reflection step where the coarse waypoints (which have larger distances and less precision) are logged into a reflection file, and the UAV is programmed to follow these waypoints during the flight. The coarse waypoints are cleared and new ones are added, which takes in the coordinates, height, and timestamps for each waypoint. This allows for further refinement during the execution phase of the flight.

5. Post-processing and Finalization

Finally, once the trajectory is completed and the UAV has reached its destination, the system stores the results for analysis and reporting. The data can also be used to generate flight logs, which provide a detailed overview of the UAV's performance and adherence to the planned trajectory.

In this case, the system also writes the coarse waypoint data to a separate JSON file, which can later be used for post-flight analysis or future mission planning.

In summary the trajectory planning workflow can be given as,

1) Define waypoints - Starting and destination points are specified, along with intermediate waypoints, using geographic or Cartesian coordinates.
2) Path Calculation - The system calculates the optimal flight path between waypoints, considering factors like distance, speed, and environmental conditions.
3) Fine Trajectory Generation - Intermediate waypoints are generated for smoother transitions and optimized flight paths.
4) Real-time Execution - The UAV follows the generated path, adjusting as necessary based on real-time conditions.
5) Post-Processing - Once the mission is completed, the trajectory data is stored for analysis and future reference.

By combining computational methods with real-time feedback mechanisms, the UAVâĂŹs trajectory planning system ensures that the vehicle can safely and efficiently navigate its mission, regardless of environmental factors or dynamic operational constraints.

IV. Multi-vehicle Simulation Scenarios

This section will describe the simulation scenarios that utilize distributed sensing concepts to enable UAM. In these simulations, multiple vehicles are modeled, each one replicating the same flight dynamics and system architecture as the single vehicle. Specifically, the Flight Control System, Flight Vehicle System, and Flight Management System are duplicated for all the vehicles, ensuring that each operates with the same performance characteristics and decision-making logic. The simulation includes a total of 50 flights, which are spread across 22 vertiports located throughout the San Francisco Bay Area, allowing for the testing of UAM operations in a diverse and complex urban environment. These scenarios are designed to assess the effectiveness of distributed sensing and coordination among multiple autonomous vehicles operating in close proximity.

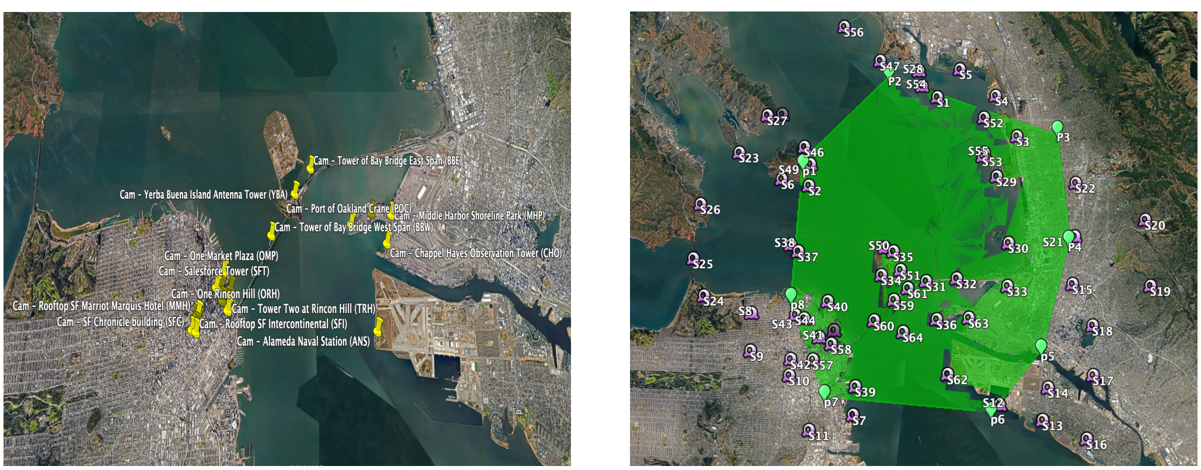
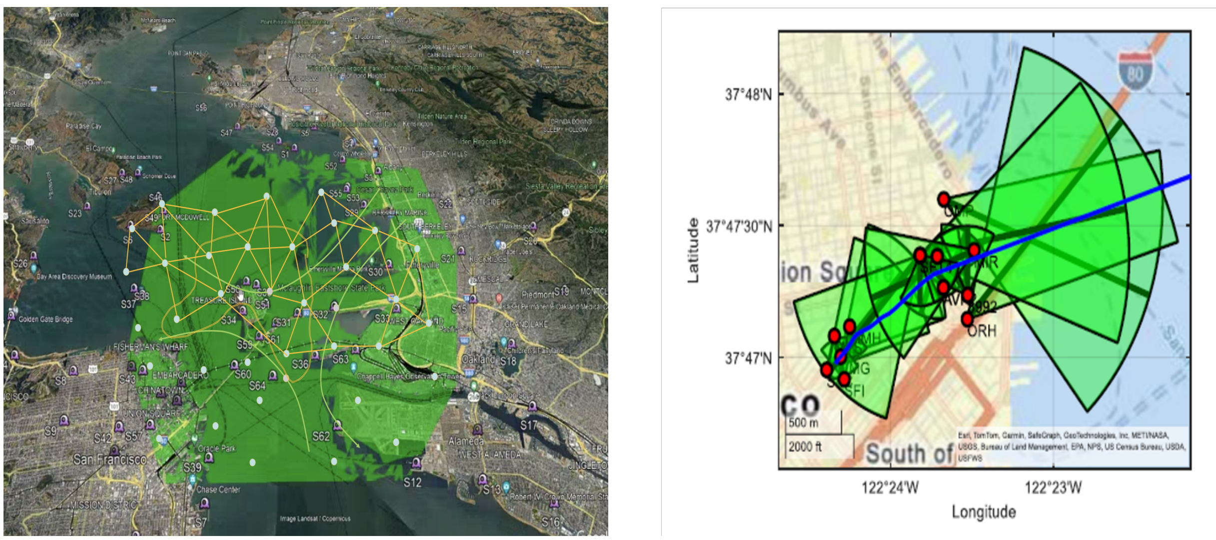


Fig. 11 Additional ground sensors spread throughout the routes

Fig. 12 Sensors at the landing site



(a) Overall coverage of the sensors placed throughout the San Francisco Bay area

(b) Viewing angles of the sensors at the landing site[9]

Fig. 13 Coverage by the dense sensor suite

A. Dense sensor population in the areas of interest

In the exploration of distributed sensing concepts for navigation without GPS, the simulation scenario incorporates the deployment of dense sensors within the area of interest. Specifically, additional cameras are strategically installed from Middle Harbor Shoreline Park (MHP) to Fifth & Mission Garage (FMG) to enhance surveillance capabilities. A minimum of four to five cameras are positioned to focus on the landing site at FMG, allowing for comprehensive monitoring of incoming flights and operational activities. This augmentation of sensor infrastructure enables the simulation to simulate and evaluate the efficacy of distributed sensing approaches in facilitating navigation and maneuvering within the designated airspace corridor. The table in 14 provides coordinates for all major vertiports considered in the San Francisco Bay Area. Meanwhile, Figure 14 offers a Google Earth depiction of these vertiports. Apart from this, a substantial number of additional vertiports have been incorporated into the simulation. Additionally, Figures 11, 12, and 13 depict the locations of sensors, emphasizing the dense coverage they provide. These visuals provide a detailed glimpse into the distribution of sensors throughout the area, showcasing the comprehensive monitoring network established for effective data collection and analysis.

No.	Vertiport	Latitude (°)	Longitude (°)
1	Pier 24	37.7890552	-122.3865241
2	Fifth and Mission Garage	37.783379	-122.405077
3	UCSF Medical Center at Mission Bay	37.7667022	-122.390365
4	Fort Mason Center	37.806666	-122.4310605
5	Fisherman's Wharf	37.8113183	-122.4203347
6	Ferry Building	37.795138	-122.3934823
7	Jack London Square	37.7941944	-122.2761147
8	Alcopark Parking Garage	37.8008194	-122.2653841
9	Kaiser Permanente Parking Garage	37.8230108	-122.2592473
10	Middle Harbor Shoreline Park	37.8058404	-122.3238733
11	Golden Gate Fields	37.8846806	-122.3143313

(a)

(b)

Fig. 14 Major Vertiports in the San Francisco Bay area[9]

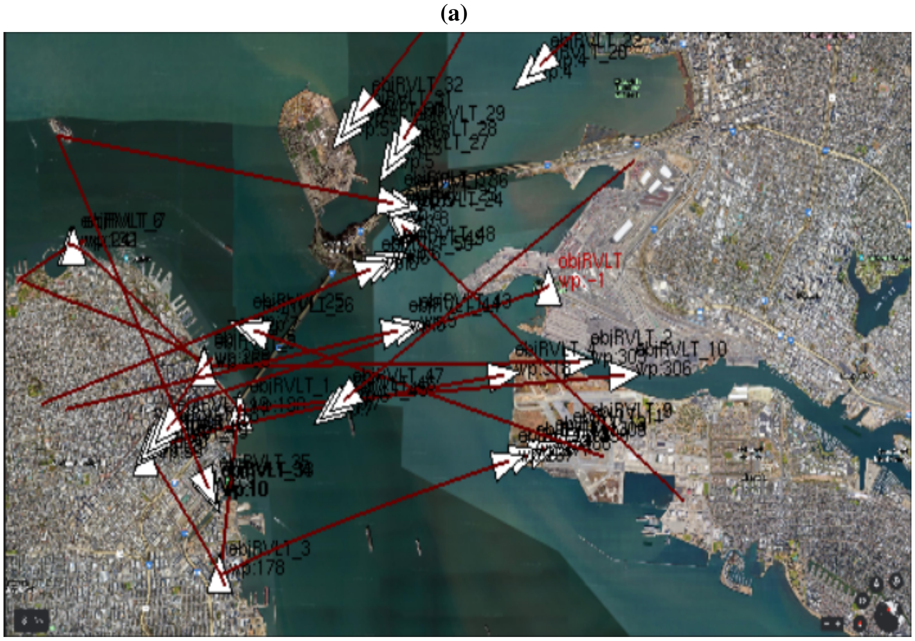
B. M:N Operations Scenarios

In the simulation scenario designed to explore the m:n concept in UAM, up to five flights are scheduled to be followed sequentially on the same route. To maintain operational uniformity, departures are evenly spaced, ensuring a consistent rhythm throughout the simulation. It ensures that even as the vehicles move together in tight formations, they can still adapt to changes in the environment or react to other vehicles in the convoy. Adding to the richness of the dataset, multiple fleets are deployed, with each being traveled in different directions. This deliberate choice enhances the versatility of the findings, allowing a wide spectrum of operational intricacies inherent in UAM systems to be captured. Figures 15 (a) and (b) display multiple convoys moving in different directions. Each convoy consists of 2-5 vehicles, moving in a straight line at regular intervals. These images offer a clear visualization of the coordinated movement of the convoys, highlighting their structured and synchronized behavior. Looking ahead, each vehicle will be equipped with an onboard sensor network that shares real-time data with other vehicles nearby. This sharing of information enables several important capabilities, including automated separation between vehicles and formation control, which are crucial for ensuring safe and efficient operations in busy urban airspaces. This concept to distributed sensing and m:n coordination is essential for making UAM operations safe and efficient at scale. Simulating these scenarios gives valuable insights into how multiple autonomous vehicles can work together seamlessly, without compromising safety or operational efficiency whether they're traveling in formation, reacting to unexpected obstacles, or adjusting to changes in traffic patterns.

C. Sensor Network for Multiple Approach and Departure Paths into/from Vertiports

Expanding the scope of research for M:N, the simulation integrates multiple approach paths leading into the FMG vertiport. By introducing additional approach paths, the simulation enhances the versatility and complexity of flight operations, allowing for a more comprehensive exploration of M:N concepts. This expansion facilitates a diverse range of flight trajectories and landing strategies, accommodating varying traffic densities and operational scenarios. Moreover, the inclusion of multiple approach paths provides a robust framework for evaluating the efficiency and effectiveness of distributed navigation and landing protocols within the UAM ecosystem. Ultimately, this feature broadens the horizons of research, offering valuable insights into optimizing airspace utilization and enhancing the scalability of distributed sensing in UAM systems. Figure 16 visually presents the various approach paths, all converging towards the FMG vertiport. Meanwhile, Figure 17 showcases the array of sensors positioned along one of these approach paths.

(a)



(b)

Fig. 15 Large-scale simulation scenario with convoys flying in different directions

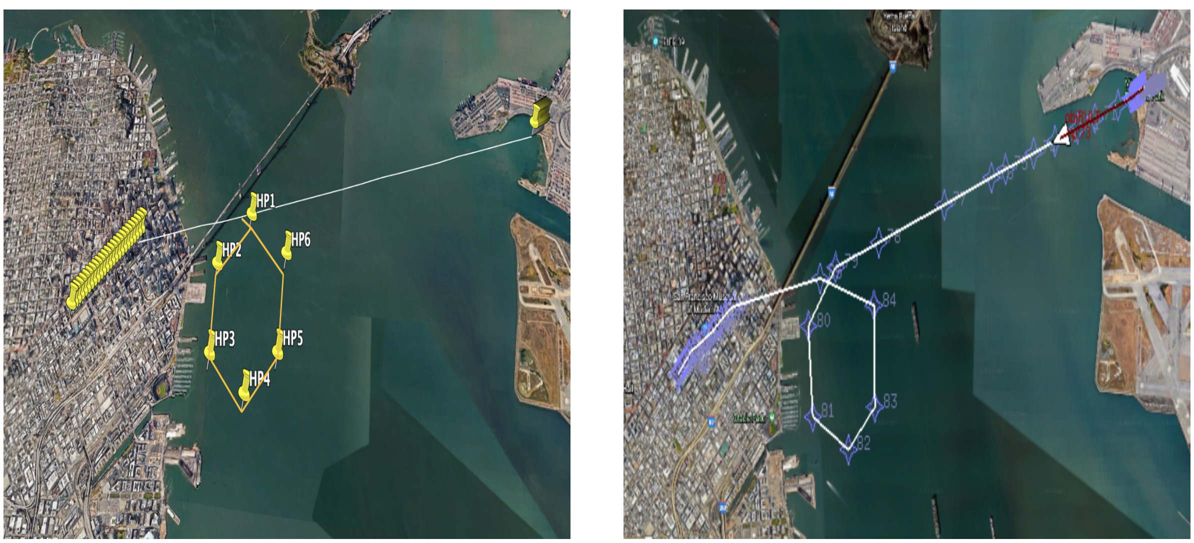
D. Holding patterns

Incorporating holding patterns into the simulation enriches the scenario by introducing a dynamic element to flight operations. When a flight approaches the landing phase but encounters an unavailable vertiport, the simulation orchestrates a transition into a holding pattern. In this scenario, the flight circles around the designated airspace until it receives clearance to land, ensuring safety and adherence to air traffic protocols. By simulating this racetrack pattern, the simulation explores alternative navigation strategies, providing valuable insights into the adaptability of UAM systems. This feature enhances the realism of the simulation, mimicking real-world scenarios where unforeseen circumstances necessitate temporary delays in landing procedures. While conventional aircraft use racetrack patterns to loiter around airports, there are no formal standards for holding patterns specifically designed for AAM aircraft awaiting landing at vertiports. To address this, the simulation adopts the same holding pattern standards used for conventional aircraft[17],

Fig. 16 Approach paths flying into FMG vertiport

Fig. 17 One approach path populated with sensors

simplifying the process of standardization. The holding pattern calculations are based on a bank angle of 35°, a cruise velocity of 60.96 m/s, and a load factor determined by 1/cos(35°), resulting in a turn radius of approximately 540.996 meters. In Figure 18, the racetrack holding pattern adopted by the simulated flight is depicted. This pattern, characterized by its elongated shape resembling a racetrack, is shown to be more suitable for densely populated urban environments compared to circular holding patterns.



(a) **(b)**

Fig. 18 Holding racetrack pattern

E. Dataset Utilization

The datasets generated are used to support various algorithms, including Vision-Based Distributed Sensing at Vertiports for approach and landing, and Optimal Sensor Placement for Enhanced Object Localization Accuracy. These datasets also provide valuable insights and lessons for the upcoming flight tests. One key application is an approach that operates without estimation feedback, addressing both the optimal sensor selection and the wireless communication network topology determination problems. This approach supports independent airspace surveillance by leveraging ground-based distributed sensing, computing, and wireless communication infrastructure.

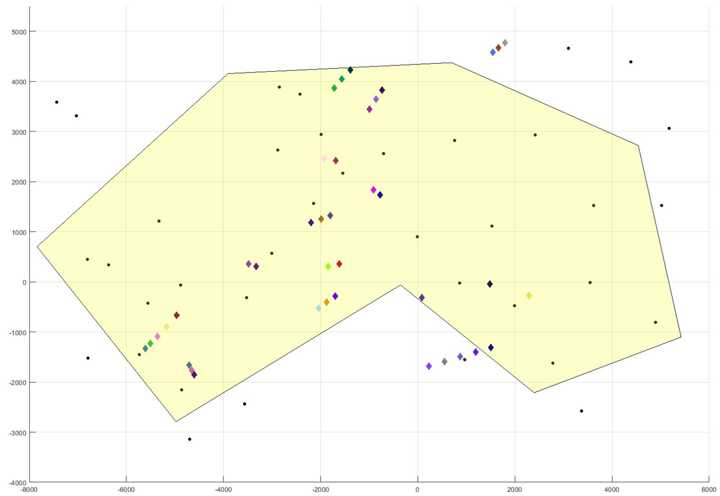


Fig. 19 Snapshot of the air traffic across the region of interest [18].

The selection criteria for sensors include maximizing airspace coverage with minimal resource usage, minimizing communication time and power consumption, while ensuring system observability and providing real-time, high-quality information to a monitoring observer. The developed algorithm employs a multi-objective optimization strategy, which balances trade-offs between conflicting objectives and allows for relaxed constraints to enable real-time implementation. It is implemented using existing computational tools such as the MATLAB Optimization Toolbox and the Graph and Network Algorithms Toolbox.

The approach is validated in a simulation environment using synthetic sensor data generated from the multi-vehicle flight scenario described in the paper. To briefly summarize the specific parameters, air traffic in this region is simulated using the Reflection Simulation Environment, with 50 air vehicles flying both individually and in formations at a speed of $5m/s$. These vehicles have varying takeoff and landing times, and are spaced with sufficient horizontal and vertical separation to avoid collisions. A snapshot of the air traffic projected onto the horizontal plane is shown in Figure 19, where black dots represent the locations of the ground nodes, and the colored diamonds indicate the 2D positions of the air vehicles. The saved vehicle trajectories are used to generate synthetic images of the air vehicles at a rate of 25 Hz, based on an in-house developed camera model. The camera has a 4:3 aspect ratio and is subject to additive zero-mean white noise, with variances of $\sigma_x^2 = 0.625,\ \sigma_y^2 = 0.469$ in pixels.

More details on the algorithms can be found in [18], [19], and [20].

V. Next Steps

The next steps involve conducting flight tests using ground nodes in collaboration with the NASA Smart Mobility team and the NASA Armstrong full-scale AAM aircraft flights. These tests will help validate the algorithms and datasets developed through simulation, ensuring their applicability and reliability in real-world scenarios. Additionally, discussions are underway to integrate the simulation outputs with other NASA teams. This integration aims to feed the results into other simulators, enhancing support for M:N operations and improving the overall ecosystem for UAM. By working closely with various NASA teams, the goal is to create a seamless and robust framework that leverages distributed sensing and advanced algorithms to optimize air mobility operations across multiple platforms.

VI. Conclusion

In conclusion, the development of distributed sensing-based simulation scenarios for UAM is a complex endeavor, integrating various factors to enhance the understanding and optimization of UAM operations. Each aspect, from exploring the m:n concept to incorporating holding patterns and multiple approach paths into vertiport navigation, adds depth and complexity to the simulation framework.

However, practical constraints such as the challenge of simulating videos for numerous cameras and the limitations of visualization software like XPlane necessitate adaptive measures, leading to prioritizing sensor emulation. Transitioning to more versatile platforms like Matlab or Reflection offers opportunities to overcome these challenges, enabling realistic emulation of sensor outputs and enhancing simulation fidelity.

By embracing innovative approaches and technologies, the simulation framework can evolve to address the diverse dynamics of UAM operations, ultimately facilitating informed decision-making for the integration of UAM into urban transportation networks.

Acknowledgments

The authors thank NASA Ames Research Center's TTT/P&DS, led by Code TI, for the research opportunity.

References

[1] Goyal, R., and Cohen, A., "Advanced Air Mobility: Opportunities and Challenges Deploying eVTOLs for Air Ambulance Service," *Applied Sciences 12*, 2022, p. 1183.

[2] Ippolito, C., Hashemi, K., Kawamura, E., Gorospe, G., Holforty, W., Kannan, K., Stepanyan, V., and et al, "Distributed Sensing and Advanced Perception Technologies to Enable Advanced Air Mobility," *AIAA SCITECH 2023 Forum*, AIAA, 2023, p. 0894.

[3] Ippolito, C. A., Martin, R. A., Kawamura, E., Gorospe, G., Wendy Holforty, K. K., Stepanyan, V., Lombaerts, T., Brown, N., and Dolph, C., "Enabling Smart Urban Airspaces through Distributed Sensing Techniques," *AIAA SCITECH 2025 Forum*, AIAA, 2025.

[4] Annie, C., Witzberger, K., Phojanamongkolkij, N., and Levitt, I., "Urban Air Mobility (UAM) Airspace Research Roadmap–Systems Engineering Approach to Managing Airspace Evolution Towards UML-4," *AIAA AVIATION 2022 Forum*, AIAA, 2022, p. 3401.

[5] Straubinger, A., Rothfeld, R., Shamiyeh, M., BÃijchter, K.-D., Kaiser, J., and PlÃűtner, K. O., "An overview of current research and developments in urban air mobilityâĂŞSetting the scene for UAM introduction," *Journal of Air Transport Management 87*, 2020, p. 101852.

[6] Thipphavong, D. P., Apaza, R., Barmore, B., Battiste, V., Burian, B., Dao, Q., and et al, M. F., "Urban air mobility airspace integration concepts and considerations," *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3676.

[7] Shish, K. H., Cramer, N. B., Gorospe, G., Lombaerts, T., Stepanyan, V., and Kannan, K., "Survey of Capabilities and Gaps in External Perception Sensors for Autonomous Urban Air Mobility Applications," *AIAA Scitech 2021 Forum*, 2021, p. 1114.

[8] *https://www.nasa.gov/aam*, Advanced Air Mobility Overview, last accessed November 2022.

[9] Keerthana, K., Baculi, J. E., Lombaerts, T., Kawamura, E., Gorospe, G. E., Holforty, W., Ippolito, C. A., Stepanyan, V., Dolph, C., and Brown, N., "A Simulation Architecture for Air Traffic Over Urban Environments Supporting Autonomy Research in Advanced Air Mobility," *AIAA SCITECH 2023 Forum*, AIAA, 2023, p. 0895.

[10] Evan, K., Kannan, K., Lombaerts, T., and Ippolito, C. A., "Vision-Based Precision Approach and Landing for Advanced Air Mobility," *AIAA SCITECH 2022 Forum*, AIAA, 2022, p. 0497.

[11] Evan, K., Dolph, C., Kannan, K., Lombaerts, T., and Ippolito, C. A., "Simulated Vision-based Approach and Landing System for Advanced Air Mobility," *AIAA SCITECH 2023 Forum*, AIAA, 2023, p. 2195.

[12] Thomas, L., Kannan, K., Kawamura, E., Dolph, C., Stepanyan, V., Gorospe, G. E., and Ippolito, C. A., "Distributed Ground Sensor Fusion Based Object Tracking for Autonomous Advanced Air Mobility Operations," *AIAA SCITECH 2023 Forum*, AIAA, 2023, p. 0896.

[13] Keerthana, K., Ippolito, C. A., and Holforty, W., "Flight Test Configuration of the Sensor Payload and the Ground Nodes in Distributed Sensing Frameworks," *AIAA SCITECH 2024 Forum*, AIAA, 2024, p. 2007.

[14] Stefan, S., Malpica, C., Lombaerts, T., Kaneshige, J., Withrow, S., Hardy, G., and Aires, J., "A modeling approach for handling qualities and controls safety analysis of electric air taxi vehicles," *AIAA AVIATION 2020 FORUM*, 2020, p. 3188.

[15] Johnson, W., Silva, C., and Solis, E., "Concept Vehicles for VTOL Air Taxi Operations," *AHS Specialists Conference on Aeromechanics Design for Transformative Vertical Flight, San Francisco, CA*, 2018, pp. ARC–E–DAA–TN50731.

[16] Silva, C., Johnson, W. R., Solis, E., Patterson, M. D., and Antcliff, K. R., "VTOL urban air mobility concept vehicles for technology development," *Aviation Technology, Integration, and Operations Conference*, 2018, p. 3847.

[17] Federal Aviation Administration, "Instrument flying handbook," U.S. Department of Transportation, 2010. URL `https://www.sheppardair.com/download/faa-h-8083-15.pdf`.

[18] Stepanyan, V., Ippolito, C., Kannan, K., Kawamura, E., and Lombaerts, T., "Optimal Communication Topology Construction and Sensor Selection for Independent Airspace Surveillance," *AIAA SCITECH 2025 Forum*, AIAA, 2025.

[19] Lombaerts, T., Kawamura, E., Kannan, K., Ippolito, C., Stepanyan, V., Gorospe, G., Dolph, C., Vitiello, F., and Causa, F., "A case study into optimal sensor placement for higher accuracy object localization for urban air mobility," *AIAA SCITECH 2025 Forum*, AIAA, 2025.

[20] Kawamura, E., and et al, "Vision-Based Distributed Sensing at Vertiports for Advanced Air Mobility and Urban Air Mobility Approach and Landing," *AIAA SCITECH 2025 Forum*, AIAA, 2025.