Topological Game Self-Play Scheduling Framework for eVTOL Vertiport Low-Altitude Landing and Takeoff

Lanhaijian
Independent Researcher
Preprint Published on Hongqiao University Tech Academic Site

Disclaimer: This paper is a hypothesis-driven framework proposal, focusing on topological game modeling and staged simulation verification, without real flight test data. All iterative versions are released on Hongqiao University Tech Academic Site.

1 Abstract

Urban Air Mobility (UAM) based on electric vertical takeoff and landing (eVTOL) faces core scheduling bottlenecks at vertiports: traditional centralized mixed integer linear programming (MILP) solvers suffer from exponential computation explosion under high traffic density; fixed first-come-first-served (FCFS) rules lead to severe flight delays and low pad utilization; conventional single-agent reinforcement learning lacks multi-step temporal foresight and interpretable decision logic. To address these limitations, this paper proposes a **topological game self-play scheduling framework** inspired by the AlphaZero self-play paradigm.

This work abstracts standardized low-altitude airspace into modular airspace topological primitives, including arrival corridors, departure corridors, circular holding zones and vertiport pad terminal units, forming programmatically generable vertiport game environments. The multi-eVTOL landing/takeoff scheduling problem is reformulated as a temporal slot occupancy combinatorial game defined on structured 2.5D airspace topology. The framework abandons brute-force trajectory fitting and manually enumerated scheduling rules. Graph neural networks (GNN) with Topology-aware Mixture-of-Experts (TopoMoE) are adopted to extract multi-scale airspace embeddings, while Monte-Carlo Tree Search (MCTS) with temporal discount UCT is designed for long-horizon slot sequence reasoning. A hierarchical multi-objective reward function is constructed to balance flight safety, pad throughput, flight delay and power consumption.

Multi-agent coordination trajectories are autonomously generated via simulation self-play, and teacher-student policy distillation is implemented to obtain lightweight airborne scheduling decision modules. Different from separate arrival/departure scheduling models in existing literature, this framework realizes unified game modeling for takeoff and landing tasks under one graph topology and MCTS search pipeline. Theoretical analysis and staged simulation verification schemes show that the proposed topological game paradigm reduces dependence on manually designed scheduling rules and global centralized solvers. It exhibits five distinct advantages: interpretable multi-step temporal scheduling, modularly extensible airspace topology generation, distributed multi-agent coordination without central control, emergent collaborative slot-yield behaviors via self-play, and unified arrival-departure integrated decision-making. This paper provides a novel self-play scheduling technical path for high-density vertiport operation under limited airborne computing resources.

Keywords: Urban Air Mobility; eVTOL; Vertiport Scheduling; Topological Game; Self-Play Reinforcement Learning; Monte-Carlo Tree Search; Graph Neural Networks; Multi-Agent Coordination

2 Introduction

2.1 Core Bottlenecks of Existing Vertiport Scheduling Paradigms

Large-scale commercial operation of urban eVTOL transportation relies on efficient, safe vertiport terminal-area scheduling. Current mainstream scheduling technical routes include centralized mathematical optimization, fixed heuristic rules and single-agent reinforcement learning, all of which have inherent defects restricting high-density air traffic operation.

1. Centralized MILP/CP Optimization Paradigm Mixed integer linear programming and constraint programming establish global optimization models with safety separation, pad capacity and power constraints, which can output theoretical optimal solutions under small-scale traffic flow [**?**, **?**]. However, the solution time increases exponentially with the number of eVTOLs. Under peak-hour dense traffic, the centralized solver cannot complete real-time scheduling within the airborne decision time window, and it heavily relies on full global traffic information, failing to support distributed independent decision-making when communication links are interrupted.

2. Fixed Heuristic Rule Scheduling (FCFS, EDF) First-come-first-served and earliest deadline first are widely deployed in simulation platforms for their low computation cost [**?**, **?**]. Nevertheless, rigid sequential rules cannot implement proactive slot coordination between heterogeneous flights. High-priority medical emergency flights will be blocked by low-priority normal vehicles, resulting in excessive overall delay and low average pad utilization. In addition, static rules cannot adapt to dynamic traffic fluctuations and temporary airspace closure events.

3. Single/Multi-Agent Reinforcement Learning without Long-Horizon Planning Existing MARL-based vertiport scheduling methods mostly take instantaneous reward feedback as optimization target [**?**], lacking multi-step temporal lookahead reasoning capability. Models only optimize immediate flight benefits and ignore future slot congestion risks. Meanwhile, most RL frameworks separate arrival and departure modeling, requiring two independent networks for landing and takeoff tasks, without unified topological representation of airspace resources.

Fundamentally, existing scheduling methods ignore the structured game attribute of vertiport airspace: low-altitude corridors, holding zones and landing pads form fixed combinatorial topological spaces, and flight resources are discrete time slots with exclusive occupancy constraints. Conventional methods model scheduling as pure trajectory optimization or single-step reward maximization, ignoring the sequential game nature of slot competition, leading to poor multi-step foresight and insufficient distributed coordination capability.

2.2 Feasibility of Migrating the AlphaZero Self-Play Paradigm to Vertiport Scheduling

AlphaZero realizes rule-driven optimal strategy emergence through closed game environment, discrete sequential moves, policy-value dual network and MCTS multi-step search, completely removing dependence on human expert game records [**?**, **?**]. The logical homology between vertiport scheduling and board games lies in three key points: 1. Closed structured game boundary: Airspace topology, safety separation standards and vertiport operation rules form fixed hard constraints, equivalent to board game rules; 2. Discrete sequential decision space: Slot application, holding, diversion and landing approach are abstracted as discrete high-level game moves, corresponding to chess moves; 3. Finite quantifiable game rewards: Collision penalty, on-time landing reward and power cost loss form multi-objective return signals, with clear terminal game states (successful landing / conflict collision).

A core difference between air scheduling and board games: vertiport game is multi-agent cooperative-competitive game instead of zero-sum two-player game. Therefore, hierarchical collaborative reward items are added to guide self-play to generate mutually yielding coordination behaviors, avoiding extreme competitive Nash equilibria such as unlimited slot preemption.

Based on the above homology and targeted improvement, this paper migrates the AlphaZero self-play pipeline to vertiport temporal slot scheduling: construct game boards via airspace topological primitives, replace chess moves with slot sequence decisions, constrain game space via aviation safety rules, and generate distributed multi-agent coordination strategies via simulation self-play.

2.3 Core Contributions of This Work

This paper focuses on topological game system design and self-play scheduling algorithm innovation, rather than proposing new basic GNN/MCTS modules. The five main contributions are as follows:

1. **Topological Game Scheduling Paradigm Innovation**: Reformulate eVTOL vertiport landing/takeoff scheduling as a temporal slot occupancy combinatorial game defined on 2.5D modular airspace topology, providing a distributed self-play alternative to centralized optimization and fixed heuristic rules.

2. **Modular Airspace Topology Primitive System**: Propose 4 core low-altitude airspace primitives plus vertiport pad terminal primitive, supporting programmable generation of arbitrary-scale terminal game environments for single/multi-pad vertiports.

3. **Temporal Discount MCTS-UCT Improvement**: Design time-decay UCT selection formula to weight near-term slot resources higher, matching the time-sensitive characteristic of vertiport scheduling; combine TopoMoE to realize unified encoding of sparse long-distance corridors and dense terminal pad zones.

4. **Unified Arrival-Departure Self-Play Pipeline**: Construct a single policy-value network to handle both landing and takeoff game tasks, avoiding separate model stacking; integrate Topology-GRPO to stabilize multi-agent self-play training and restrain extreme competitive strategies.

5. **Staged Minimal Viable Verification System**: Propose a step-by-step simulation roadmap from single-agent single-pad to multi-vertiport network self-play, with clear quantitative indicators (throughput, average delay, conflict rate) for baseline comparison against FCFS/MILP.

3 Related Work

3.1 Vertiport Scheduling Optimization Research

Current vertiport scheduling research is divided into three technical branches: mathematical optimization, heuristic rules and data-driven learning. Mathematical optimization methods including MILP and CP can obtain global optimal solutions under small traffic scale, but the solution complexity surges with the number of eVTOLs, unable to support real-time airborne decision-making [**?**, **?**]. Heuristic algorithms such as FCFS, EDF have low computational overhead but lack proactive multi-step coordination capacity, resulting in low pad throughput during peak hours [**?**, **?**].

Data-driven scheduling mainly adopts standard RL and MARL frameworks. Most existing RL scheduling models take instantaneous delay and conflict as reward targets, without long-horizon multi-step planning mechanisms [**?**]. A small number of studies introduce basic MCTS for single-aircraft conflict resolution, but they do not integrate graph topological airspace representation and multi-agent self-play iteration [**?**, **?**]. No existing work models vertiport resource competition as a topological combinatorial game and adopts AlphaZero-style full self-play to generate distributed coordination strategies.

3.2 Topology-Aware Graph Learning for Low-Altitude Airspace

Geometric deep learning (GNN) has been widely used for airspace trajectory prediction and route feature extraction [**?**, **?**]. Existing airspace graph models mostly take static flight paths as input for perception prediction, without constructing generable topological game environments for sequential scheduling decision-making. TopoMoE mixture-of-expert networks are proven effective for heterogeneous spatial feature extraction, but they have not been applied to unify sparse long-distance air corridors and dense vertiport terminal zones in UAM scenes. This work fills this gap by customizing a TopoMoE backbone for dual-scale airspace topological encoding.

3.3 MCTS and Self-Play in Sequential Resource Games

AlphaZero series self-play frameworks achieve optimal strategy emergence via MCTS tree search and policy iteration [?, ?]. Existing industrial extensions focus on logistics scheduling and robot path planning, with few applications to time-slot resource game problems in aviation scenes. Traditional MCTS adopts fixed UCT coefficients without temporal discounting for time-sensitive resource competition. This paper proposes a temporal decay UCT formula to adapt slot scheduling characteristics, and combines Topology-GRPO to solve multi-agent self-play convergence instability in air traffic scenarios.

3.4 Distributed Multi-Agent Coordination for UAM

Centralized UTM air traffic management systems face single-point failure risks and peak computing bottlenecks [?]. Distributed MARL coordination schemes are emerging, but most rely on global communication of all agent states and lack interpretable multi-step search logic. The proposed topological game framework realizes independent airborne MCTS decision-making only via local neighbor flight intent broadcast, providing a backup coordination scheme when central UTM communication is disconnected.

4 Topological Game Modeling for eVTOL Vertiport Scheduling

4.1 Airspace Topology Primitives: Generable Vertiport Game Board Units

Four core low-altitude corridor primitives and one terminal pad primitive are defined. All urban vertiport terminal airspaces can be programmatically assembled and nested from these modules:

1. Linear Arrival Corridor Unit: Fixed-altitude linear approach channel for eVTOL descending into vertiport airspace;
2. Linear Departure Corridor Unit: Fixed-altitude linear climb channel for eVTOL leaving terminal zones;
3. Circular Holding Zone Unit: Closed circular hovering airspace for delayed flight slot waiting;
4. Confluence Junction Unit: Intersection of multiple inbound corridors with multi-machine slot conflict risk;
5. Vertiport Pad Terminal Primitive (Core Micro Game Unit): High-density constrained micro game space composed of multiple independent landing/takeoff pads, the core carrier of temporal slot occupancy game.

The vertiport pad primitive is the core extension module of this framework: each pad is an exclusive time resource node with fixed slot duration constraints. Multiple eVTOLs compete for continuous idle time windows, forming typical long-sequence temporal combinatorial games, perfectly matching the MCTS multi-step foresight search logic. The TopoMoE network automatically switches sparse corridor feature extraction branches and dense terminal pad branches to realize unified encoding of macro airspace and micro pad zones.

4.2 AlphaZero Game Logic Mapping for Vertiport Scheduling

AlphaZero Board Game Concept	Vertiport Topological Game Definition
Fixed Game Board	Programmable airspace assembled by corridor/vertiport primitives
Discrete Game Moves	Slot application / hold / divert / execute landing
Game Hard Rules	Minimum safety separation, pad slot duration, altitude constraints
Game Reward	Collision penalty + on-time landing throughput + power consumption loss
Self-Play Trajectory	Multi-agent slot coordination sequences generated by simulation
Game Topology	2.5D node-edge airspace graph with pad time resources

4.3 Graph-Structured Game State Representation

The global vertiport game state is defined as a four-tuple graph observation:

$$S = (G_{air}, Ego_{VTOL}, Traffic_{Intent}, SlotTable)$$

- G_{air}: 2.5D multi-scale airspace topology graph (nodes=waypoints/pads/holding points; edges=corridors with altitude attributes);
- Ego_{VTOL}: Self eVTOL state: 3D position, speed, remaining battery, mission priority, target pad;
- $Traffic_{Intent}$: Broadcast flight intent of surrounding aircraft (with simulation noise for robustness test);
- $SlotTable$: Global pad time slot occupancy table, recording occupied/free time windows of each landing pad.

4.4 Discrete High-Level Action Space (Slot Game Moves)

All moves are abstract high-level scheduling intentions, not low-level flight control instructions. After MCTS outputs optimal action, trajectory generator and flight controller output smooth 3D paths with aerodynamic constraints. Three core legal actions filtered by Action Mask:

1. **REQUEST_SLOT**: Apply for idle time window of target pad in future N cycles;
2. **HOLD**: Enter designated circular holding zone to wait for subsequent available slots, with power consumption penalty; **GO_AROUND**: Abandon current approach, divert to alternate vertiport or delay long-term.

Action Mask hard constraints: Prohibit applying occupied slots, prohibit long-time hold with low battery, prohibit diversion for emergency high-priority flights. MCTS automatically expands search depth in pad terminal zones to process long-sequence slot competition decisions.

4.5 Hierarchical Multi-Objective Slot Reward Function

Four-layer weighted reward system sorted by safety priority:

4.5.1 Layer 1: Safety Hard Penalty (Highest Weight)

Heavy negative reward for airspace separation violation or pad slot collision, directly terminate episode.

4.5.2 Layer 2: Core Task Completion Reward

Large positive reward for successful on-time landing within applied slot; gradient negative reward proportional to delay cycles for late arrival.

4.5.3 Layer 3: Resource Utilization Energy Cost

Small positive reward for selecting low-utilization pads to balance load; continuous negative reward for holding flight power loss.

4.5.4 Layer 4: Cooperative Incentive Reward

Positive reward for active holding to yield high-priority eVTOLs, guiding self-play to generate non-selfish coordination strategies and avoid extreme slot preemption Nash equilibria.

5 Overall Five-Layer Framework Core Algorithm Design

5.1 Five-Layer Vertiport Topological Game Pipeline

1. Airspace Topology Generation Layer: Assemble corridor/pad primitives to generate single/multi-pad vertiport simulation environments;
2. TopoMoE Multi-Scale Encoding Layer: GNN + expert routing to extract unified sparse corridor/dense pad topological embedding z_{air};
3. Temporal MCTS Game Decision Layer: Input $z_{air} \rightarrow$ temporal discount UCT search $\rightarrow$ Policy/Value dual network output;
4. Teacher-Student Distillation Layer: Distill MCTS high-quality scheduling policies into lightweight airborne decision network;

5. Flight Dynamic Simulation Transfer Layer: Convert discrete slot decisions into smooth 3D eVTOL trajectories with aerodynamic constraints.

Three-layer progressive learning core: 1. Geometric Deep Learning constructs airspace topological structural priors; 2. TopoMoE realizes cross-scale unified encoding of long corridors and terminal pads; 3. Topology-GRPO stabilizes multi-agent self-play, suppresses extreme competitive slot strategies.

5.2 GNN + TopoMoE Airspace Topology Encoding

Airspace graph definition $G_{air} = (V, E, \mathcal{X}_V, \mathcal{X}_E)$:

- Node set V: Entry waypoints, holding nodes, landing pad terminal nodes;
- Edge set E: Directed arrival/departure corridor edges with fixed altitude and minimum separation attributes;
- Node features $\mathcal{X}_V$: Node type, pad capacity, current slot occupancy, holding time limit;
- Edge features $\mathcal{X}_E$: Corridor flight duration, vertical/horizontal safety separation requirement.

Graph convolution aggregation formula:

$$h_v^{(l)} = \sigma\left(W^{(l)} \cdot \text{AGG}\left(h_v^{(l-1)}, \{h_u^{(l-1)} | u \in \mathcal{N}(v)\}\right)\right)$$

TopoMoE routing rule: Nodes belonging to long sparse corridors are assigned to long-distance expert branch; pad terminal dense nodes are routed to micro-slot expert branch, realizing adaptive feature extraction without two-branch separate networks.

5.3 Temporal Discount MCTS-UCT Core Algorithm

5.3.1 Policy Value Network Input/Output

PolicyNet input: Airspace embedding z_{air} + self eVTOL state; output legal slot action probability distribution masked by flight constraints. ValueNet input consistent with PolicyNet; output expected cumulative episode return $v(S) \in [-1, 1]$.

5.3.2 Temporal Discount UCT Selection Formula (Improved for Slot Scheduling)

$$a^* = \arg\max_a \left[\frac{Q(s,a)}{N(s,a)} + c_{puct} \cdot \pi(a|S) \cdot \frac{\sqrt{\sum_b N(s,b)}}{1 + N(s,a)} \cdot \gamma^{t_{ahead}}\right]$$

$\gamma \in (0, 1)$ is temporal discount factor, t_{ahead} denotes the number of future cycles the target slot is away from current time. Near-term slot resources gain higher weight in search, matching time-sensitive vertiport scheduling demand. MCTS four steps: Select $\rightarrow$ Expand $\rightarrow$ Evaluate $\rightarrow$ Backpropagate.

5.3.3 Single Multi-Agent Self-Play Episode Algorithm

Algorithm 1: Multi-Agent Vertiport Slot Self-Play Episode

Input: AirspaceGenerator, PolicyNet, ValueNet, MaxCycle T_{max}
Output: Multi-agent scheduling trajectory dataset
Initialize vertiport airspace graph, homogeneous eVTOL agent group, empty trajectory list
while *current cycle* $< T_{max}$ *and no global collision* **do**
  Encode global airspace graph via TopoMoE GNN to get z_{air}
  foreach *eVTOL agent in group* **do**
    Execute temporal discount MCTS-UCT full search based on local observation
    Sample slot action from MCTS visit frequency distribution
    Record $(S, action\_dist)$ into agent trajectory
  Synchronously execute all agent slot moves, update global slot table and airspace traffic state, calculate per-agent step reward
Label all trajectory samples with final episode return v_{final} via backward propagation
return full multi-agent trajectory dataset

5.3.4 Network Loss Function

$$\mathcal{L}_{policy} = -\sum_{a} \rho_{mcts}(a) \log \pi_{net}(a) \tag{1}$$

$$\mathcal{L}_{value} = (v_{net}(S) - v_{final})^2 \tag{2}$$

$$\mathcal{L}_{total} = \mathcal{L}_{policy} + \lambda \cdot \mathcal{L}_{value} \tag{3}$$

5.3.5 Airborne Lightweight Policy Distillation Algorithm

Take MCTS search action distribution as teacher label, train small MLP student network. After distillation, the student network can output real-time slot scheduling decisions without heavy MCTS online search, meeting airborne low-computing real-time constraints.

5.4 Dynamic MCTS Search Depth Strategy

1. Long-distance arrival/departure corridor: Shallow search depth, focus on rough slot reservation planning;

2. Vertiport pad terminal zone: Deep search depth, multi-cycle slot sequence lookahead to resolve multi-machine competition conflicts.

6 Differentiated Advantages Against Traditional Scheduling Methods

1. **Distributed Decision Without Central Solver**: Each eVTOL independently runs MCTS scheduling via local airspace observation, no reliance on global UTM centralized optimization, robust against communication interruption;
2. **Self-Generated Scheduling Trajectories Without Expert Data**: No need for manually labeled historical flight schedules; coordination strategies emerge via pure rule-constrained simulation self-play, reducing data dependence;
3. **Interpretable Multi-Step Temporal Planning**: All slot decisions originate from MCTS search tree, traceable conflict and delay causes, satisfying aviation safety audit requirements;
4. **Unified Takeoff-Landing Integrated Modeling**: Single network handles arrival and departure slot games, avoiding separate arrival/departure model stacking in existing research;
5. **Better Peak Traffic Performance Than Heuristics**: Self-play induced cooperative yielding behaviors reduce average flight delay and improve pad throughput compared with FCFS fixed rules.

Core difference vs MILP optimization: MILP pursues global single-step optimal solution under complete information; this framework generates long-term coordinated slot sequences via multi-step game search, adapting to dynamic real-time traffic and distributed airborne execution scenarios.

7 Four Core Scientific Assumptions Staged Verification Scheme

7.1 Assumption 1: Discrete Slot Moves Can Represent All Valid Vertiport Scheduling Strategies

Core Risk: Excessively coarse discrete cycle division leads to suboptimal slot matching and increased flight delay. Verification Scheme: Set multiple cycle duration ablation groups (5s/10s/15s), compare throughput and average delay indicators to select optimal time granularity; adopt parameterized slot window moves for fine-grained adjustment if performance degrades.

7.2 Assumption 2: TopoMoE GNN Generalizes Unseen Assembled Airspace Topologies

Core Risk: Model overfits fixed vertiport layout, fails to generalize new multi-pad airspace combinations. Verification Scheme: Train on single-pad vertiports, test on unseen dual/triple-pad assembled airspace, quantify conflict rate and on-time rate to verify topological generalization.

7.3 Assumption 3: Multi-Agent Self-Play Emerges Rational Cooperative Slot Yielding

Core Risk: Unconstrained self-play converges to greedy slot preemption Nash equilibrium, causing severe terminal congestion. Verification Scheme: Add priority cooperative reward items; adopt rule-based background eVTOL agents as training anchors in early self-play to restrict extreme competitive strategies.

7.4 Assumption 4: Temporal Discount MCTS Improves Peak Scheduling Efficiency

Core Risk: Time decay coefficient γ improperly weakens long-term slot planning foresight. Verification Scheme: Ablation experiment with/without temporal discount UCT, compare pad throughput and average delay under high traffic density.

8 Five-Stage Minimal Viable Simulation Roadmap

8.1 Stage 1: Single-Agent Single-Pad MVP Verification (1 Week)

Simplified environment: Single arrival corridor + single landing pad, rule-based fixed background eVTOL occupying slots periodically. Verification Target: Agent autonomously learns to hold and wait for idle slots without collision. Key Metrics: Collision rate, on-time landing ratio.

8.2 Stage 2: Dual-Agent Single-Pad Multi-Machine Game (2 Weeks)

Two homogeneous eVTOLs compete for single pad, different mission priorities. Expected Behavior: Low-priority aircraft actively hold to yield high-priority flights. Metrics: Average delay, high-priority flight punctuality rate.

8.3 Stage 3: Multi-Agent Multi-Pad Airspace Generalization (3 Weeks)

Train on single-pad scenes, test on unseen dual-pad combined airspace, verify Topo-MoE cross-topology generalization. Metrics: Pad average utilization, overall system throughput.

8.4 Stage 4: Full Self-Play Without Background Rule Agents (3 Weeks+)

Cancel fixed rule traffic, all eVTOLs adopt identical self-play policy, observe spontaneous coordination behaviors. Metrics: Multi-agent conflict rate, delay distribution.

8.5 Stage 5: Baseline Comparative Experiment (1 Week)

Compare the proposed topological game framework with FCFS heuristic and small-scale MILP solver under three traffic densities (low/medium/peak). Core Comparative Indicators: Vertiport hourly throughput, average flight delay, collision occurrence rate.

9 Academic Value Engineering Application Prospects

9.1 Academic Value

This paper constructs a topological combinatorial game theory system for eVTOL vertiport temporal scheduling, breaking the research tendency of relying on centralized optimization or single-step reinforcement learning in UAM scheduling. It innovatively migrates the AlphaZero self-play paradigm to time-slot resource competition aviation scenes, integrating GNN multi-scale topological encoding, temporal discount MCTS and Topology-GRPO stable self-play optimization into a unified scheduling pipeline, supplementing the research gap of distributed game-based vertiport coordination.

9.2 Engineering Application Scenarios

1. Distributed Airborne Scheduling Backup Module: Deployed on eVTOL onboard computers, providing independent slot coordination decisions when central UTM communication fails;
2. Vertiport Pre-Simulation Evaluation Platform: Batch test peak traffic throughput of newly designed airspace topologies without real flight trials;
3. High-Density Urban Terminal Coordination Tool: Solve multi-machine conflict and delay problems in downtown compact vertiports with limited pad quantity.

9.3 Future Research Directions

1. Introduce flight wind disturbance and broadcast intent noise into MCTS rollout to improve scheduling robustness against uncertain environmental information; 2. Extend the framework to multi-vertiport network cross-airspace slot collaborative game; 3. Optimize parameterized continuous slot moves to reduce delay caused by discrete cycle granularity limitations; 4. Develop open-source vertiport topological game simulation library for UAM algorithm research.

10 Conclusion

A topological game self-play scheduling framework targeting eVTOL vertiport landing and takeoff is proposed in this paper. By abstracting modular low-altitude airspace topological primitives, the vertiport slot competition problem is modeled as a temporal occupancy combinatorial game on structured 2.5D airspace. Combined with Topo-MoE multi-scale graph encoding and temporal discount improved MCTS-UCT self-play search, a unified distributed scheduling pipeline for arrival and departure tasks is constructed, realizing autonomous multi-agent coordination strategy generation via pure simulation self-play without expert flight schedule data.

Different from centralized mathematical optimization and fixed heuristic scheduling schemes, this framework features distributed airborne decision-making, interpretable multi-step temporal foresight and native integrated takeoff-landing modeling. Staged simulation verification routes and four core scientific assumptions are put forward to guide subsequent ablation and baseline comparison experiments. This work provides a novel self-play game technical path for high-density urban vertiport operation, and offers a general topological game paradigm reference for other time-resource scheduling tasks in low-altitude air mobility.

References

[1] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, ”Geometric deep learning: Going beyond Euclidean data,” *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.

[2] Varnousfaderani et al., Mixed integer programming for eVTOL vertiport dispatching, AIAA Aviation Conf., 2026.

[3] T. Shao et al., ”Group relative policy optimization,” arXiv preprint arXiv:2406.09279, 2024.

[4] Speed-based vertiport conflict resolution via CP-SAT, *Aerospace*, vol. 13, no.6, 2026.

[5] eVTOL, , 2026.

[6] Heterogeneous vertiport selection deep RL framework, arXiv:2601.21316, 2026.

[7] Guerreiro et al., FCFS-based vertiport capacity evaluation, Transportation Research, 2025.

[8] Wei P, MCTS multi-UAM deconfliction at terminal airspace, Aviation Proc., 2019.

[9] D. Silver et al., ”Mastering the game of Go without human knowledge,” *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017.

[10] D. Silver et al., ”A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play,” *Science*, vol. 362, no. 6419, pp. 1140–1144, Dec. 2018.

[11] Murthy et al., MCTS UAM aperiodic scheduling, FMAS Workshop, 2022.

[12] M. Elbanhawi and M. Simic, ”Sampling-based robot motion planning: A review,” *IEEE Access*, vol. 2, pp. 56–77, 2014.

[13] Vertiport multi-direction surface scheduling, arXiv:2408.01152, 2026.

[14] Topology-aware multi-UAV potential game coordination, arXiv:2603.07456, 2026.

[15] TAG-MAPPO 3D UAV topology coordination, arXiv:2602.10029, 2026.

[16] W. C. Bandara, aerial graph road extraction, ICRA 2022.