UAVBench: An Open Benchmark Dataset for Autonomous and Agentic AI UAV Systems via LLM-Generated Flight Scenarios

Mohamed Amine Ferrag*[§], *Senior Member, IEEE*, Abderrahmane Lakas*, *Senior Member, IEEE*, and Merouane Debbah[1], *Fellow, IEEE*

Abstract—Autonomous aerial systems increasingly rely on large language models (LLMs) for mission planning, perception, and decision-making; yet, the lack of standardized, physically grounded benchmarks limits systematic evaluation of their reasoning capabilities. To address this gap, we introduce UAVBench, an open benchmark dataset comprising *50,000 validated UAV flight scenarios* generated through taxonomy-guided LLM prompting and multi-stage safety validation. Each scenario is encoded in a structured JSON schema encompassing mission objectives, vehicle configuration, environmental conditions, and quantitative risk labels, providing a unified representation of UAV operations across diverse domains. Building on this foundation, we present UAVBench_MCQ, a reasoning-oriented extension containing *50,000 multiple-choice questions* spanning ten cognitive and ethical reasoning styles—from aerodynamics and navigation to multi-agent coordination and hybrid integrated reasoning. This framework enables interpretable, machine-checkable assessment of UAV-specific cognition under realistic operational contexts. We evaluate 32 state-of-the-art LLMs, including GPT-5, ChatGPT 4o, Gemini 2.5 Flash, DeepSeek V3, Qwen3 235B, and ERNIE 4.5 300B, and find strong performance in perception and policy reasoning but persistent challenges in ethics-aware and resource-constrained decision-making. `UAVBench` establishes a reproducible, physically grounded foundation for benchmarking agentic AI in autonomous aerial systems and advancing next-generation UAV reasoning intelligence. To support open science and reproducibility, we release the `UAVBench` dataset (including labeled data), the `UAVBench_MCQ` benchmark, evaluation scripts, and all related materials on GitHub: https://github.com/maferrag/UAVBench.

Index Terms—Autonomous aerial systems, large language models, reasoning and decision-making, benchmark datasets, Autonomous AI Agents.

I. INTRODUCTION

Large Language Models (LLMs) are emerging as powerful tools for enhancing UAV autonomy. Recent studies have increasingly explored integrating LLMs into UAV systems to improve autonomy, decision-making, and communication. Several works demonstrate how LLMs can augment or replace traditional reinforcement learning and optimization frameworks, which often struggle with training complexity and low sample efficiency [1]. For example, LLMs have been applied to the Internet of Drones via hybrid decision-making frameworks that combine discovery Generation with structured knowledge graphs, allowing interpretable context-aware UAV control [2]. Other efforts employ LLM-guided reinforcement learning to address security and energy-efficiency trade-offs in heterogeneous UAV networks, achieving improved secrecy rates and robust trajectory optimization [3]. Similarly, LLM-based in-context learning has been introduced for intelligent data collection scheduling in UAV-assisted networks, outperforming baseline strategies while also revealing vulnerabilities to adversarial manipulation [4]. Additional work focuses on minimizing the age of information in UAV-assisted sensor networks using an evolutionary-optimization-assisted LLM, demonstrating superior routing efficiency under high node density conditions [5]. These developments highlight LLMs' ability to infuse adaptability, interpretability, and semantic reasoning into UAV decision pipelines.

Beyond single-agent autonomy, LLMs are also being leveraged for multi-agent UAV coordination and large-scale operational contexts. Recent frameworks utilize iterative structured prompting to optimize multi-hop UAV placements, thereby reducing computational overhead while maintaining near-optimal performance in network backhaul scenarios [6], [7]. Other works demonstrate the effectiveness of LLM-based in-context learning for flight resource allocation in wildfire monitoring, where real-time scheduling is critical to minimizing latency and data staleness [8]. Hierarchical architectures that combine high-altitude platforms and onboard UAV LLMs have been proposed for 3D aerial highway systems, providing both strategic access control and tactical maneuvering [9]. In the domain of swarm intelligence, LLM-driven role-adaptive frameworks enhance collaboration through semantic communication and dynamic role switching, improving task coverage and generalization in multi-UAV systems [10]. Likewise, urban trajectory planning approaches merge DRL with LLM reasoning to ensure safe, efficient, and regulation-compliant operations in low-altitude economic airspaces [11]. Together, these works demonstrate that LLMs are not merely auxiliary tools but are emerging as core enablers of intelligent, interpretable, and scalable UAV autonomy across diverse mission profiles [12].

Constructing intelligent agents capable of understanding natural language commands and translating them into navigation behaviors remains a central challenge in artificial intelligence [13]. Although vision-language navigation (VLN) has been extensively studied for ground robots, the aerial domain introduces far greater complexity. UAVs must operate within continuous three-dimensional environments characterized by

*Department of Computer and Network Engineering, College of Information Technology, United Arab Emirates University, Al Ain, United Arab Emirates.
[1]Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates.
§Corresponding author: `mohamed.ferrag@uaeu.ac.ae`

high degrees of freedom, varying altitudes, dynamic obstacles, and fluid environmental conditions such as wind and lighting changes [14]. These factors make path planning, spatial reasoning, and language grounding considerably more difficult than in ground-based systems. Moreover, aerial navigation demands fine-grained control over orientation, velocity, and stability, where small errors can lead to mission failure or collisions. Consequently, direct adaptations of ground-based VLN methods—typically optimized for discrete, planar movements—fail to capture the continuous, physics-driven nature of aerial motion and the real-time decision-making constraints inherent to flight [15].

Despite growing research attention, existing UAV VLN benchmarks and datasets remain limited in terms of realism, task diversity, and physical grounding. Many rely on simplified discrete actions, static environments, or low-fidelity simulators that neglect the continuous control challenges central to UAV operation [7]. This lack of physical and semantic richness hinders progress toward fully embodied aerial intelligence. Bridging this gap requires specialized platforms and datasets that integrate realistic flight dynamics, multimodal perception, and mission-level reasoning. Such resources would not only enable more accurate simulation of UAV flight behaviors but also facilitate the study of complex reasoning and language-grounded decision-making under real-world constraints. Addressing these limitations motivates the development of unified benchmarks such as UAVBench, which couple scenario-level realism with structured reasoning evaluation, paving the way for end-to-end research on autonomous, language-guided aerial navigation.

Unmanned aerial vehicles (UAVs) are increasingly deployed across domains, including disaster response, agriculture, environmental monitoring, traffic observation, and energy infrastructure inspection. However, most missions still depend on human-operated remote control, which is labor-intensive, error-prone, and costly [16]. Developing autonomous UAV agents that can perceive, reason, and act in complex environments is therefore a critical research objective. Compared to ground-based or indoor agents, UAVs face distinct challenges such as operating in large-scale, dynamic 3D environments, managing costly data collection, and requiring well-defined aerial-embodied tasks. Addressing these challenges requires specialized simulators, datasets, and evaluation frameworks that facilitate training and benchmarking of UAV embodied intelligence [17].

Beyond task execution, UAV autonomy requires advancing from fine-grained instruction-based navigation to high-level, goal-oriented cognition. Emerging approaches, such as Object Goal Navigation (ObjectNav), demonstrate the potential of semantic-driven navigation, in which agents reach mission-critical targets using abstract goals rather than detailed step-by-step instructions [18]. Although ObjectNav has been explored in indoor ground settings, its application to outdoor aerial environments remains underdeveloped. At the same time, human-like embodied cognition —processing continuous first-person visual streams for orientation, reasoning, and navigation —is largely absent from current UAV research. Urban airspaces, with their vertical mobility, dynamic obstacles, and dense semantic complexity, present new challenges for autonomous

navigation. To advance the field, it is imperative to establish systematic, standardized, and open benchmarks that evaluate the cognition embodied in UAVs and enable robust, scalable autonomy in real-world scenarios [19].

Our study is guided by the following research questions, designed to investigate how structured, physically grounded UAV scenarios and reasoning-based evaluation frameworks can advance the development of autonomous aerial intelligence:

Research Questions

- **RQ1:** How can a unified schema and taxonomy-driven generation framework ensure that large-scale UAV scenarios remain physically consistent, safety-aware, and semantically diverse for benchmarking autonomous flight intelligence?
- **RQ2:** What methods can be employed to systematically validate and risk-label automatically generated UAV scenarios to guarantee physical feasibility, schema compliance, and interpretability?
- **RQ3:** How can structured reasoning tasks derived from validated UAV scenarios be formulated to evaluate and compare cognitive, ethical, and operational decision-making in autonomous aerial systems?
- **RQ4:** How do distinct reasoning styles—spanning physical, navigational, ethical, and hybrid domains—affect the accuracy, generalization, and reliability of intelligent agents when performing UAV-related reasoning tasks?
- **RQ5:** To what extent do different model architectures and training paradigms influence consistency and grounded reasoning performance across diverse UAV mission contexts?

To address these research questions, we introduce `UAVBench`, an open benchmark dataset constructed from LLM-generated UAV flight scenarios for evaluating and training agentic AI models in autonomous aerial systems. `UAVBench` unifies scenario generation, validation, risk labeling, and reasoning into a single framework that systematically produces structured and physically consistent UAV missions. Each scenario is represented as a validated JSON specification capturing the UAV configuration, environment, mission objectives, airspace geometry, and safety constraints. The dataset integrates a multi-stage validation pipeline to ensure schema compliance, physical feasibility, and hazard-aware labeling, thereby enabling large-scale benchmarking of autonomous flight intelligence. Furthermore, we extend this dataset with `UAVBench-MCQ`, a reasoning-oriented benchmark that evaluates the cognitive, ethical, and operational decision-making capabilities of large language models (LLMs) in UAV contexts. The key contributions of this work are summarized as follows:

1) *Unified UAV Scenario Schema:* We propose a structured and mathematically defined schema that represents each UAV mission as a tuple encompassing simulation dynamics, vehicle configuration, environmental conditions, mission planning, and safety constraints. This schema ensures consistency, physical validity, and interoperability across diverse UAV applications.

2) *Taxonomy-Guided Scenario Generation:* We develop a taxonomy-driven LLM prompting mechanism that samples from a factorized space of mission types, airspace configurations, weather conditions, UAV designs, and payload categories. This approach yields a large-scale dataset, `UAVBench`, consisting of *50000 validated and physically*

TABLE I: Comparative coverage analysis of UAV embodied-intelligence benchmarks.

Work	Year	Scenario Design			Reasoning Scope			Evaluation
		Physical Realism	Validation & Risk	Mission Diversity	Physics/Navigati	Ethics/Safety	Hybrid Reasoning	Structured MCQs
Wang et al. [15] (OpenUAV)	2024	●	◐	○	●	○	○	○
Yao et al. [16] (AeroVerse)	2024	●	◐	●	●	○	○	○
Guo et al. [17] (BEDI)	2025	●	○	●	◐	○	○	○
Xiao et al. [18] (UAV-ON)	2025	●	○	◐	●	○	○	○
Zhao et al. [19] (UrbanVideo-Bench)	2025	◐	○	○	◐	○	○	◐
UAVBench / UAVBench_MCQ	2025	●	●	●	●	●	●	●

Symbols denote coverage levels: ●= fully covered, ◐= partially covered, ○= not covered.

consistent UAV flight scenarios that are semantically rich, safety-aware, and suitable for both training and evaluation.

3) *Multi-Stage Validation and Risk Labeling:* We introduce a systematic validation pipeline that enforces geometric, physical, and safety constraints on all generated scenarios. Each scenario is further annotated with quantitative risk levels and categorical safety tags (e.g., *Weather, Navigation, Energy, Collision-Avoidance*) derived from the detected hazards and environmental severity, forming a reproducible and interpretable benchmark for risk-aware UAV autonomy.

4) *UAVBench_MCQ (Structured Reasoning Benchmark):* We present UAVBench_MCQ, a reasoning-oriented extension of UAVBench containing *50,000 multiple-choice questions (MCQs)* systematically derived from validated scenarios. Each MCQ follows a standardized JSON schema and belongs to one of ten reasoning styles—*aerodynamics & physics, navigation & path planning, policy & compliance, environmental sensing, multi-agent coordination, cyber-physical security, energy management, ethical decision-making, comparative systems,* and *hybrid integrated reasoning*. The framework enforces grounded realism, structural completeness, and logical consistency to enable reproducible large-scale reasoning evaluation.

5) *Large-Scale LLM Evaluation:* We benchmark thirty-two state-of-the-art large language models (LLMs)—including GPT-5, ChatGPT 4o, Gemini 2.5 Flash, DeepSeek V3, Qwen3 235B, ERNIE 4.5 300B, and Mistral Medium 3.1—on the UAVBench_MCQ benchmark. The evaluation spans ten reasoning styles covering physical, navigational, ethical, and hybrid cognitive dimensions of UAV autonomy, revealing strong performance in perception and policy reasoning but persistent challenges in multi-agent coordination, energy management, and ethics-aware decision-making.

The remainder of this paper is structured as follows. Section II reviews previous studies on LLM-driven autonomy, UAV simulation datasets, and reasoning benchmarks. Section III describes the construction of the UAVBench dataset, including its taxonomy-guided scenario generation, schema definition, and multi-stage validation and risk-labeling pipeline. Section IV introduces UAVBench_MCQ, a reasoning-focused extension that formalizes ten styles of cognitive and ethical reasoning for UAV systems. Finally, Section V summarizes the main findings and discusses potential directions for future research in agentic and safety-aware UAV intelligence.

II. RELATED WORK

In recent years, substantial progress has been made toward developing benchmarks and platforms to evaluate embodied intelligence in unmanned aerial vehicles (UAVs). Various studies have proposed simulation frameworks, large-scale datasets, and evaluation methodologies, each addressing specific aspects, such as vision-language navigation, embodied cognition, and object-goal navigation. While these contributions have significantly advanced UAV autonomy, they exhibit notable differences in task definitions, experimental settings, and evaluation strategies.

A. Vision-Language Navigation Platforms for UAVs

Wang et al. [15] introduce OpenUAV, a simulation platform designed to advance vision-language navigation (VLN) for UAVs. Unlike prior benchmarks that oversimplify aerial navigation using discrete actions, OpenUAV provides realistic environments, continuous six-degrees-of-freedom (6-DoF) flight control, and algorithmic support for trajectory generation. Using this platform, the authors construct the first large-scale dataset of realistic UAV VLN trajectories (over 12,000), enriched with human-annotated paths and GPT-4–generated navigation instructions. To address the challenges of aerial search tasks, they propose the UAV-Need-Help benchmark, which introduces assistant-guided navigation with varying levels of support. Finally, they develop a UAV navigation LLM that integrates multiview images, language instructions, and assistant guidance to produce hierarchical trajectories, achieving significant performance gains over baselines but still trailing behind human operators.

B. Embodied Aerospace Intelligence

Yao et al. [16] introduce AeroVerse, a benchmark designed to foster the development of embodied aerospace intelligence. The authors present AeroSimulator, a drone simulation platform that models realistic urban scenes using Unreal Engine and AirSim, alongside two large-scale pre-training datasets: AerialAgent-Ego10k (real-world drone image-text pairs) and CyberAgent-Ego500k (virtual image-text-pose alignment data). They also define, for the first time, five downstream tasks of the UAV agent: scene awareness, spatial reasoning, navigation exploration, task planning, and motion decision, and provide corresponding fine-tuning datasets (SkyAgent-Scene3k, SkyAgent-Reason3k, SkyAgent-Nav3k, SkyAgent-Plan3k, and SkyAgent-Act3k). To evaluate UAV agent performance, the

authors propose SkyAgent-Eval, a GPT-4–based automated evaluation framework that complements traditional metrics such as BLEU and SPICE. Experimental results with multiple 2D/3D vision-language models highlight both the promise and limitations of existing approaches, underscoring the need for specialized aerospace embodied-world models.

C. Benchmarks for UAV-Embodied Agents

Guo et al. [17] propose BEDI, a framework to assess UAV-embodied agents (UAV-EAs). At its core is the Dynamic Chain-of-Embodied-Task paradigm, which models UAV behavior as a perception–decision–action loop and decomposes complex missions into measurable subtasks. Based on this paradigm, the benchmark defines five core skills—semantic perception, spatial perception, motion control, tool utilization, and task planning—and designs evaluation metrics for each. To ensure broad applicability, BEDI integrates both static real-world imagery and dynamic virtual environments (e.g., cargo delivery, firefighting, moving-target tracking), enabling agents to be tested under varied conditions. Importantly, it offers open interfaces for integrating custom UAV agents, promoting reproducibility and extensibility. Evaluations of several state-of-the-art vision-language models highlight their limitations in handling embodied UAV tasks, underscoring BEDI's role in establishing a systematic, open, and scalable benchmark for UAV embodied intelligence.

D. Object Goal Navigation in UAVs

Xiao et al. [18] introduce UAV-ON, a benchmark dedicated to instance-level Object Goal Navigation (ObjectNav) in outdoor aerial settings. Unlike prior UAV vision-and-language navigation benchmarks that rely on detailed, step-by-step instructions, UAV-ON defines more than 11,000 navigation tasks using semantic goal instructions that describe object category, approximate size, and visual attributes. The benchmark features 14 high-fidelity outdoor environments created with Unreal Engine and AirSim, spanning urban, natural, and mixed-use regions, and includes 1,270 annotated target objects placed according to realistic co-occurrence patterns. UAV-ON employs physically grounded continuous controls rather than teleport-based movements, requiring agents to integrate perception, obstacle avoidance, and semantic reasoning for safe navigation. The authors evaluate three baselines: a random policy, a CLIP-based heuristic agent, and their Aerial ObjectNav Agent (AOA), a zero-shot framework leveraging multimodal LLM reasoning. Results reveal that while LLM-based approaches excel at semantic exploration, they struggle with precise stopping and safe trajectory execution, leading to high collision rates across all methods.

E. Embodied Cognition in Urban Airspaces

Zhao et al. [19] present a benchmark specifically designed to assess embodied cognition in motion within complex 3D urban environments. The benchmark introduces a novel task suite of 16 tasks across four categories—recall, perception, reasoning, and navigation—each designed to test the embodied capabilities of Video-LLMs. To support these tasks, the authors collected 1,547 embodied drone video clips from real cities in Guangdong Province and from two simulators (EmbodiedCity and AerialVLN), and generated over 5,200 multiple-choice questions (MCQs) using a hybrid pipeline that combines LLM-based generation, blind filtering, and human refinement. Seventeen Video-LLMs, both open-source and proprietary, were evaluated under zero-shot and fine-tuned settings. Results show that state-of-the-art models achieve only 45% accuracy, and that causal reasoning is strongly correlated with recall, perception, and planning. The study highlights the challenges of embodied intelligence in urban airspaces and demonstrates the potential of simulation-to-real transfer through fine-tuning.

F. Comparative Analysis of UAV Benchmarks

Table I presents a comparative overview of recent UAV embodied-intelligence benchmarks, highlighting differences in design scope, reasoning coverage, and evaluation methodology. The analysis shows that prior work, such as *OpenUAV* [15] and *AeroVerse* [16], emphasizes physically realistic environments and vision-language navigation tasks but provides limited validation or risk modeling. Similarly, *BEDI* [17] and *UAV-ON* [18] advance embodied cognition and aerial navigation yet lack a unified schema or reasoning-based evaluation component. *UrbanVideo-Bench* [19] focuses on video understanding and multimodal reasoning within urban contexts but remains narrow in mission diversity and lacks standardized cognitive evaluation. In contrast, UAVBench and its reasoning extension UAVBench_MCQ provide comprehensive coverage across physical realism, validation and risk assessment, multi-domain reasoning, and structured evaluation. This unified design enables consistent, interpretable, and reproducible benchmarking of UAV intelligence across perception, planning, and decision-making dimensions.

Overall, these works represent significant progress toward advancing UAV autonomy by introducing specialized platforms, datasets, and evaluation frameworks. However, existing benchmarks still face several limitations. Many focus on narrow tasks, such as vision-language navigation or object-goal navigation, limiting their applicability to broader mission scenarios. Others remain heavily simulation-driven, limiting their ability to capture the full complexity of real-world aerial environments. In addition, task designs are often predefined and lack scalability, constraining the diversity of challenges UAV agents can face. Finally, the evaluation approaches remain fragmented, with limited emphasis on unified and systematic assessments of UAV intelligence across perception, reasoning, planning, and execution. These limitations highlight the need for more comprehensive, flexible, and realistic benchmarks—such as UAVBench and UAVBench_MCQ—that can better support the development of next-generation UAV-embodied intelligence.

III. UAVBench - Dataset Generation Methodology

Designing a benchmark for UAV autonomy requires scenarios that are both systematically diverse and scientifically rigorous. To achieve this, we formalize each scenario as a
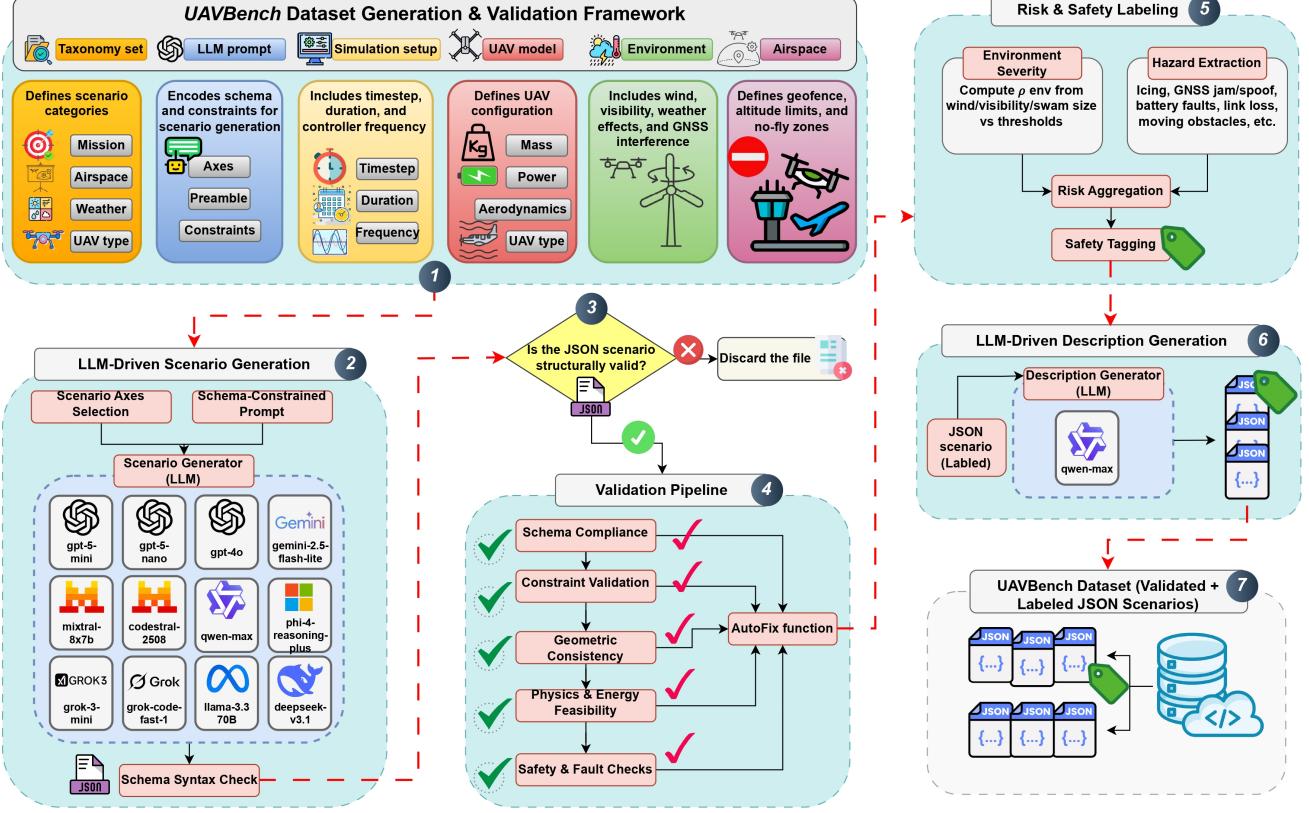
Fig. 1: UAVBench Dataset Generation, Validation, and Labeling Framework.

structured tuple that captures simulation dynamics, UAV configuration, environmental disturbances, mission objectives, and safety constraints. Table II summarizes the core mathematical notation used throughout this section.

Fig. 1 presents the complete `UAVBench` pipeline, illustrating the structured process for generating, validating, and labeling UAV scenarios. The framework begins with taxonomy-guided LLM scenario generation, followed by multi-stage validation that ensures schema compliance, geometric and physical feasibility, and safety consistency. Validated scenarios are then risk-scored and safety-tagged before being passed to an auxiliary LLM module that produces concise human-readable descriptions. The resulting dataset—composed of validated and labeled JSON files—is ready for benchmarking and simulation of agentic AI systems in diverse UAV mission contexts.

A. Scenario Schema Design

The benchmark relies on a structured schema that ensures that each scenario is both syntactically valid and physically realistic. To make this explicit, we formalize the schema using mathematical notation and provide detailed definitions of all symbols. A scenario is represented as follows.

$$\mathcal{S} = \langle \texttt{name}, \texttt{seed}, \Sigma, \mathcal{U}, \mathcal{E}, \mathcal{A}, \mathcal{X}_0, \mathcal{M}, \mathcal{T}, \mathcal{O}, \mathcal{W}, \mathcal{C}, \mathcal{D}, \mathcal{F}, \mathcal{L} \rangle, \tag{1}$$

where Σ defines the simulation parameters, $\mathcal{U}$ the UAV model, $\mathcal{E}$ the environment, $\mathcal{A}$ the airspace, $\mathcal{X}_0$ the initial spawn state, $\mathcal{M}$ the mission, $\mathcal{T}$ background traffic, $\mathcal{O}$ obstacles, $\mathcal{W}$ swarm teammates, $\mathcal{C}$ the control space, $\mathcal{D}$ safety thresholds, $\mathcal{F}$ injection faults and $\mathcal{L}$ communication constraints. The following subsections provide detailed definitions of each block.

1) Simulation Setup: The simulation setup controls the temporal structure of each scenario. It is defined as:

$$\Sigma = \langle \Delta t, N, f_c \rangle, \tag{2}$$

where Δt is the integration time step (s), N the number of discrete simulation steps, and f_c the controller update frequency (Hz). The constraints are:

$$\Delta t \in [0.01, 0.05], \quad N \geq 600, \quad f_c \geq 10. \tag{3}$$

The duration of the mission T is then:

$$T = N \cdot \Delta t. \tag{4}$$

These conditions ensure that each simulation runs for a non-trivial duration and at a temporal resolution appropriate for UAV dynamics. If Δt is too small, computational cost becomes excessive, while if it is too large, important dynamics may be missed. Similarly, $N \geq 600$ prevents trivial short-hop scenarios and aligns the dataset with real-world UAV missions that typically last several minutes. The controller frequency bound reflects the operational limits of autopilots such as PX4 and ArduPilot, grounding the benchmark in practical system architectures.

TABLE II: Notation used in the dataset generation methodology.

Symbol	Block	Definition
$\mathcal{S}$	Scenario	Full scenario tuple containing all blocks
name	Scenario	Scenario identifier string
seed	Scenario	Random seed for reproducibility
Σ	Simulation	Simulation setup tuple
Δt	Simulation	Integration time step (s)
N	Simulation	Number of simulation steps
f_c	Simulation	Controller update frequency (Hz)
T	Simulation	Total mission duration ($T = N \cdot \Delta t$)
$\mathcal{U}$	UAV	UAV configuration block
τ	UAV	UAV type (e.g., quadrotor, fixed-wing)
m	UAV	Mass (kg)
E_b	UAV	Battery energy (Wh)
V_f	UAV	Fuel volume (L)
ξ	UAV	Energy source type (battery, fuel, hybrid)
$v_{\max}$	UAV	Maximum velocity (m/s)
$\phi_{\max}$	UAV	Maximum tilt angle (deg)
r	UAV	Reserve energy fraction
$P(v, \dot{\mathbf{u}})$	UAV	Power consumption model
P_h	UAV	Hover power (W)
k_d	UAV	Drag coefficient
k_m	UAV	Maneuver coefficient
A_d	UAV	Rotor disk area (m^2)
C_D	UAV	Drag coefficient (fixed-wing)
AR	UAV	Aspect ratio
e	UAV	Oswald efficiency factor
S	UAV	Wing area (m^2)
$C_{L,\max}$	UAV	Maximum lift coefficient
v_{stall}	UAV	Stall speed (m/s)
$\mathcal{P}$	Payload	Set of payload elements
p_i	Payload	Payload element i
t_i	Payload	Payload type (e.g., lidar, camera, repeater)
m_i	Payload	Mass (kg)
P_i	Payload	Power (W)
$C_{dA,i}$	Payload	Drag–area coefficient (m^2)
μ_i	Payload	Mount position
$\mathcal{D}_i$	Payload	Data parameters
$\mathcal{O}_i$	Payload	Operating parameters
$\mathcal{C}_i$	Payload	Constraints
m_{tot}	Payload	Total mass ($m + \sum_i m_i$)
P_{payload}	Payload	Total payload power ($\sum_i P_i$)
$D_{\text{payload}}(v)$	Payload	Payload drag term
$\mathcal{E}$	Environment	Environment block
$\mathcal{E}_w$	Environment	Weather tuple
w	Environment	Wind speed (m/s)
ψ	Environment	Wind direction (deg)
g	Environment	Gust amplitude (m/s)
γ	Environment	Visibility condition
Φ	Environment	Atmospheric phenomena (hail, icing, etc.)
J_{GNSS}	Environment	GNSS jamming power (dBm)
$\mathcal{A}$	Airspace	Airspace definition
$h_{\min}, h_{\max}$	Airspace	Minimum and maximum altitude (m)
$\mathcal{P}_\ell$	Airspace	Polygonal geofence region
$\mathcal{X}_0$	Spawn	Initial UAV spawn state
$\mathcal{M}$	Mission	Mission block
σ	Mission	Mission type
$\mathcal{WP}$	Mission	Set of waypoints
κ	Mission	Path pattern (grid, corridor, orbit)
r_ℓ	Mission	Loiter radius (m)
B	Mission	Time budget (s)
ρ_{rw}	Mission	Runway requirement flag
Υ	Mission	VTOL transition profile
$\mathcal{T}$	Entities	Background traffic
$\mathcal{O}$	Entities	Moving obstacles
$\mathcal{W}$	Entities	Swarm teammates
$d_{\min}$	Entities	Minimum inter-UAV separation (m)
$\mathcal{C}$	Control	Control action set
CtrlOK$(\tau, \mathcal{A})$	Control	Predicate: UAV type τ has valid control set
$\mathcal{D}$	Safety	Safety thresholds
d_{sep}	Safety	Required separation distance (m)
$\text{TTC}_{\min}$	Safety	Minimum time-to-collision (s)
$\mathcal{F}$	Faults	Fault injection block
$(t_i, \varphi_i, \Delta t_i, s_i)$	Faults	Fault event tuple: start time, type, duration, severity
$\mathcal{L}$	Comms	Communication constraints (uplink, downlink, signal strength)
θ	Prompt	Axis tuple for LLM prompt
s, a, w, u, ν	Prompt	Scenario, airspace, weather, UAV type, and nonce
$\Pi(\mathbb{S}, \mathbb{C}; \theta)$	Prompt	LLM prompt construction function
$\rho(S)$	Risk	Risk score of a scenario S
$\sigma(S)$	Risk	Safety category label of a scenario S

2) UAV Configuration and Propulsion: The UAV block specifies the physical and energetic configuration:

$$\mathcal{U} = \langle \tau, m, E_b, V_f, \xi, v_{\max}, \phi_{\max}, r, \mathcal{B}, \mathcal{R}, \mathcal{A}_f, \mathcal{Z}, \mathcal{P} \rangle. \quad (5)$$

Here, τ is the UAV type, m mass (kg), E_b battery energy (Wh), V_f fuel volume (L), ξ energy source (battery/fuel/hybrid), $v_{\max}$ maximum velocity (m/s), $\phi_{\max}$ maximum tilt (deg), and r reserved energy fraction. The sub-blocks are: $\mathcal{B}$ battery model (e.g., hover power and coefficients), $\mathcal{R}$ rotorcraft parameters (e.g., rotor count, disk area), $\mathcal{A}_f$ fixed-wing/forward-flight aerodynamics, $\mathcal{Z}$ sensors (renamed to avoid conflict with $\mathcal{S}$), and $\mathcal{P}$ payload.

Energy consumption is modeled as:

$$P(v, \dot{\mathbf{u}}) = P_h + k_d v^3 + k_m \|\dot{\mathbf{u}}\|_2, \quad (6)$$

where P_h is hover power (W), k_d drag coefficient, v velocity (m/s), k_m maneuver coefficient, and $\|\dot{\mathbf{u}}\|_2$ the control-rate magnitude. The feasibility condition is:

$$\sum_{k=0}^{N-1} P_k \, \Delta t \le (1 - r) \, E_b \cdot 3600, \quad (7)$$

where P_k is the discrete power at step k and the factor 3600 converts Wh to joules.

Rotorcraft require rotor disk checks:

$$P_h \approx c_\eta \frac{m^{3/2}}{\sqrt{A_d}}, \quad (8)$$

where c_η aggregates propulsive efficiency factors and A_d is rotor disk area (m^2). For fixed-wing UAVs, aerodynamics are:

$$C_D = C_{D0} + \frac{1}{\pi \, AR \, e} C_L^2, \qquad v_{\text{stall}} \approx \sqrt{\frac{2mg}{\rho \, S \, C_{L,\max}}}, \quad (9)$$

with C_{D0} parasitic drag, AR aspect ratio, e Oswald factor, ρ air density, S wing area, and $C_{L,\max}$ maximum lift coefficient.

This block ensures UAVs are not arbitrary numerical constructs but physically realistic platforms. It encodes first-order aerodynamic relationships, preventing, for example, "impossible" rotorcraft from carrying large payloads with tiny rotors. By embedding both rotorcraft and fixed-wing models, the schema spans the full design space of UAVs used in research and industry. The inclusion of reserve fractions aligns with operational safety practices, which require UAVs to always retain energy for contingencies.

3) Payload Taxonomy and Examples: UAVBench introduces a comprehensive payload taxonomy that reflects the diversity of sensing, communication, delivery, industrial, and defense systems employed across modern unmanned aerial vehicle missions. Each payload is represented as

$$p_i = \langle t_i, m_i, P_i, C_{dA,i}, \mu_i \rangle, \quad (10)$$

where t_i denotes the canonical payload type (for example, lidar or thermal camera), m_i is the payload mass (kg), P_i is the electrical power requirement (W), $C_{dA,i}$ is the projected drag–area coefficient, and μ_i is the mounting position (belly, nose, top, wing, gimbal, bay, or tether). This standardized representation ensures that each payload contributes realistically

to the total UAV mass, power consumption, and aerodynamic profile.

The `UAVBench` taxonomy enumerates more than 200 canonical payload types, grouped into over 30 functional categories spanning the civil, scientific, and defense domains. The taxonomy covers a wide range of mission contexts:

- Imaging and sensing: optical, thermal, multispectral, hyperspectral, and advanced imaging such as lidar or synthetic-aperture radar.
- Communication and networking: radio relay, cellular base stations, satellite communication terminals, and ad-hoc mesh nodes.
- Industrial and environmental monitoring: methane and gas detection, pipeline inspection, and ground-penetrating radar.
- Public safety and emergency response: search-and-rescue thermal cameras, fire-mapping systems, and emergency beacons.
- Delivery and logistics: parcel and medical carriers, aerial drop systems, and life-raft deployment.
- Scientific and environmental research: radiation detectors, atmospheric sensors, and biosensors.
- Agricultural and ecological monitoring: multispectral and NDVI cameras, soil-moisture and crop-health sensors.
- Military and defense applications: electro-optical reconnaissance, laser designators, electronic-warfare systems, and CBRN detectors.

Table III presents representative payloads from these categories, illustrating the breadth of the `UAVBench` taxonomy and typical physical parameters used in simulation.

The inclusion of such a broad and standardized payload taxonomy enables `UAVBench` to generate mission scenarios with realistic, heterogeneous configurations—ranging from lightweight electro-optical cameras on micro-UAVs to multi-sensor payload suites on large fixed-wing platforms. By integrating payload mass, power, and aerodynamic drag directly into the simulation model, `UAVBench` enables reproducible, physically consistent benchmarking of autonomy, mission planning, and energy-aware flight control across both civilian and defense contexts.

4) Environment and Airspace: The environment $\mathcal{E}$ specifies weather and disturbances via the weather tuple

$$\mathcal{E}_w = \langle w, \psi, g, \gamma, \Phi \rangle, \qquad (11)$$

where w is wind speed (m/s), ψ wind direction (deg), g gust amplitude (m/s), γ visibility condition (categorical), and Φ atmospheric phenomena (e.g., icing, sandstorm). Optional electromagnetic effects include GNSS multipath, jamming power J_{GNSS} (dBm), and general EM interference.

Airspace $\mathcal{A}$ encodes altitude and lateral constraints. With $h_{\max} > h_{\min}$, vertical limits are enforced. Waypoints $\boldsymbol{w}_i$ must lie inside geofences:

$$\boldsymbol{w}_i \in \bigcup_\ell \mathcal{P}_\ell, \qquad (12)$$

where $\mathcal{P}_\ell$ are polygonal regions. No-fly zones are cylinders (static or dynamic), while runways are included for fixed-wing UAVs.

This block introduces environmental realism by constraining UAVs to atmospheric and regulatory conditions. Limiting wind speeds reflects the upper thresholds of UAV flight envelopes, while visibility levels capture operational categories such as VFR/IFR. Including electromagnetic effects such as GNSS jamming allows scenarios to simulate contested environments, aligning the benchmark with security and resilience studies. Altogether, this block integrates physical, meteorological, and regulatory realism.

5) Mission and Entities: The mission block $\mathcal{M}$ specifies task objectives:

$$\mathcal{M} = \langle \sigma, \mathcal{WP}, \kappa, r_\ell, B, \rho_{\mathrm{rw}}, \Upsilon \rangle, \qquad (13)$$

where σ is mission type, $\mathcal{WP}$ the set of waypoints ($3 \leq |\mathcal{WP}| \leq 6$), κ path pattern, r_ℓ loiter radius (m), B time budget (s), ρ_{rw} runway requirement flag, and Υ VTOL transition profile.

External entities enrich realism. Traffic $\mathcal{T}$ defines background UAVs, $\mathcal{O}$ moving obstacles, and $\mathcal{W}$ swarms. Swarm separation is enforced by:

$$\|\boldsymbol{x}_a(t) - \boldsymbol{x}_b(t)\|_2 \geq d_{\min}, \quad \forall t \in [0, T]. \qquad (14)$$

This block ensures that missions are neither trivial nor overly complex. Limiting waypoints to between three and six captures real-world planning tasks such as corridor inspections or grid-based surveys. The time budget introduces trade-offs between task completion and energy limits, mimicking operational decision-making. Including swarms and moving obstacles tests cooperative and reactive autonomy, which are critical features of next-generation UAV systems operating in dense airspaces.

6) Control, Safety, and Faults: The control block $\mathcal{C}$ specifies UAV action sets, either discrete or continuous. A predicate $\mathsf{CtrlOK}(\tau, \mathcal{A})$ ensures each UAV has sufficient degrees of freedom to remain flyable. For example, a fixed-wing UAV must support throttle, pitch, roll, and yaw.

Safety thresholds are captured as:

$$\mathcal{D} = \langle d_{\mathrm{sep}}, \mathrm{TTC}_{\min} \rangle, \qquad (15)$$

with violations defined as:

$$\min_{a \neq b} \|\boldsymbol{x}_a(t) - \boldsymbol{x}_b(t)\|_2 < d_{\mathrm{sep}} \quad \text{or} \quad \min_{a \neq b} \mathrm{TTC}_{ab}(t) < \mathrm{TTC}_{\min}. \qquad (16)$$

Faults are modeled as tuples $(t_i, \varphi_i, \Delta t_i, s_i)$: t_i start time (s), φ_i type (e.g., motor failure, GNSS jam), Δt_i duration (s), and s_i severity. Communication constraints $\mathcal{L}$ define uplink/downlink availability and signal thresholds.

This block allows UAVs to be benchmarked not only in nominal conditions but also in degraded environments. Safety thresholds align with the Unmanned Traffic Management (UTM) literature [20], ensuring comparability with regulatory concepts. The inclusion of fault injection makes the benchmark suitable for resilience testing, capturing how autonomy responds to failures such as GNSS denial or sensor corruption. This elevates the schema beyond static mission planning, positioning it as a tool for robustness evaluation.

TABLE III: Representative payload examples in `UAVBench` (excerpt from over 200 canonical types).

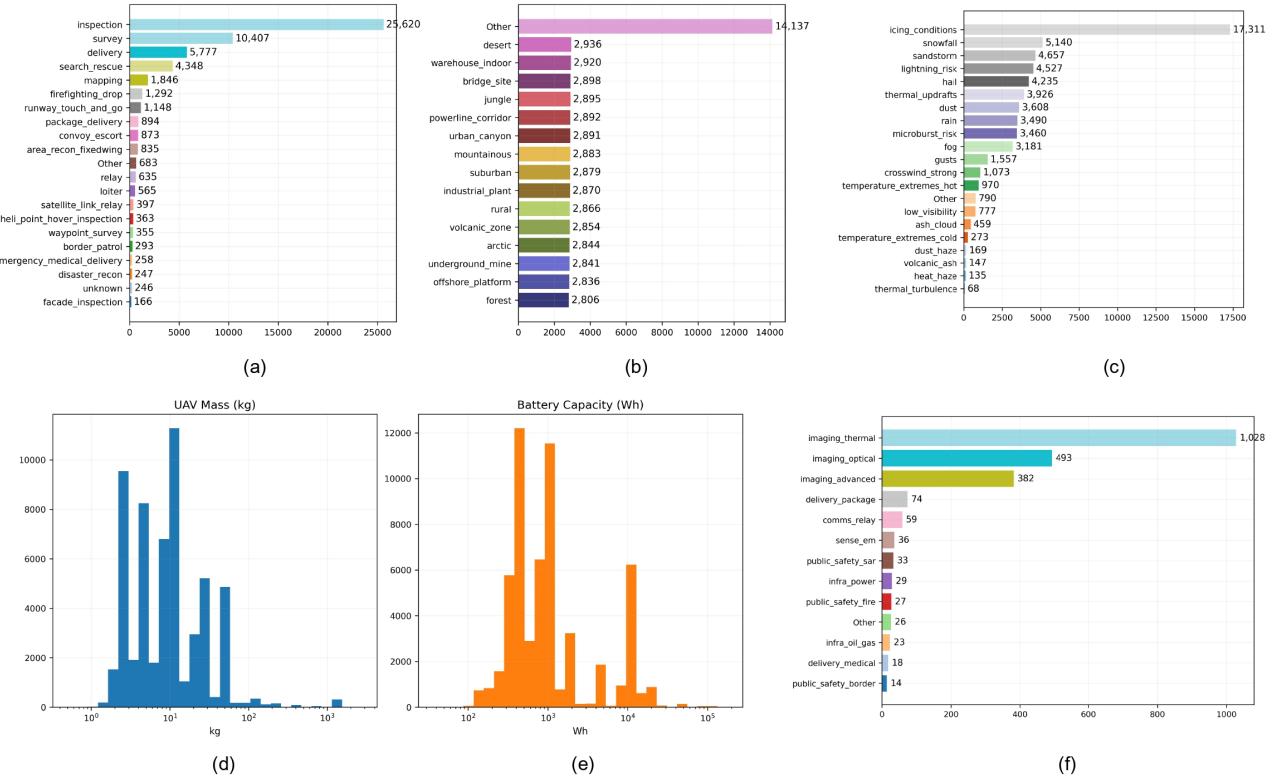| Category	Type (canonical)	Mount	Mass [kg]	Power [W]	Primary use
Imaging optical	Gimbaled camera	Gimbal	0.35	6	Structural inspection
Imaging thermal	Thermal camera (LWIR)	Nose	0.30	5	Night operations or SAR
Imaging multispectral	Multispectral camera	Wing	0.45	8	Vegetation and crop analysis
Imaging advanced	Lidar sensor	Belly	1.20	18	Terrain and infrastructure mapping
Communication relay	Radio repeater	Bay	0.60	10	Network extension
Communication link	Satellite communication terminal	Top	1.10	25	BVLOS operation
Industrial monitoring	Methane detection sensor	Belly	0.80	12	Oil and gas inspection
Industrial sensing	Ground-penetrating radar	Belly	2.50	25	Subsurface mapping
Public safety	SAR thermal camera	Gimbal	0.35	6	Search and rescue
Emergency response	Fire-thermal mapping system	Nose	0.50	10	Fire surveillance
Delivery	Parcel delivery system	Bay/tether	1.50	3	Urban logistics
Medical delivery	Medical supply carrier	Bay	2.00	4	Emergency medicine transport
Scientific	Gamma-ray spectrometer	Bay	1.00	9	Radiation monitoring
Agricultural	NDVI camera	Wing	0.40	7	Crop-health mapping
Ecological	Soil-moisture sensor	Belly	0.35	4	Environmental survey
Military ISR	Electro-optical reconnaissance system	Nose	1.80	20	Intelligence and surveillance
Military EW	Electronic-warfare jammer	Bay	2.20	30	Signal denial and counter-UAS

Fig. 2: Overview of `UAVBench` dataset composition and UAV design characteristics. (a) Mission types illustrating the diversity of operational scenarios; (b) Airspace types showing the range of environmental contexts; (c) Weather phenomena highlighting atmospheric complexity; (d) UAV mass distribution indicating variability in platform sizes; (e) Battery capacity distribution reflecting energy endurance profiles; and (f) Payload category frequencies summarizing the variety of onboard sensors and mission payloads.

B. Taxonomies and Prompt Design for LLM-Based Scenario Generation

To ensure that LLM-generated scenarios are both diverse and operationally realistic, we introduce structured taxonomies for scenarios, airspaces, weather, and UAV types. Each taxonomy organizes discrete tokens into meaningful categories, which are then embedded in the LLM prompt along with explicit constraints. This combination allows us to balance flexibility and validity in scenario generation. Fig. 2 presents an overview

of the `UAVBench` dataset composition and UAV platform characteristics. The distributions illustrate the dataset's diversity across mission types, airspace configurations, and weather conditions, highlighting its coverage of complex and realistic operational contexts. The UAV-specific statistics, including mass and battery capacity, reveal a wide range of vehicle sizes and endurance profiles, while payload diversity reflects the presence of multiple sensing modalities, including optical, thermal, and radar units. Collectively, these aspects demon-

strate UAVBench's suitability for benchmarking perception, autonomy, and risk-aware decision-making algorithms under heterogeneous aerial scenarios.

1) Scenario Taxonomy: Scenarios capture the mission-level intent of UAV operations. We organize them into categories such as inspection, delivery, reconnaissance, search and rescue, training, swarm coordination, safety-critical events, fire/hazmat, and maritime/offshore. Formally, we define a mapping

$$\mathcal{C}_S : c \mapsto \{s_1, s_2, \ldots, s_n\}, \tag{17}$$

where c is a scenario category (e.g., *inspection*) and $\{s_i\}$ is the set of scenario tokens in that category. This organization enables us to ensure that generated missions span different operational classes, from bridge inspections with multirotors to BVLOS mountain ridge flights with fixed-wing aircraft.

2) Airspace Taxonomy: Airspaces represent the environments in which UAVs operate. We partition them into four broad groups: urban, natural terrain, infrastructure corridors, and special constrained zones. This can be represented as

$$\mathcal{C}_A : a \mapsto \{e_1, e_2, \ldots, e_m\}, \tag{18}$$

where a is an airspace category and $\{e_j\}$ are the specific environments (e.g., `urban_canyon`, `desert`, `underground_mine`). This structure enables analysis of whether generated scenarios sufficiently explore both routine and extreme environments and ensures that waypoints remain consistent with airspace constraints.

3) Weather Taxonomy and Severity: Weather conditions influence both mission safety and UAV performance. We group tokens into precipitation, wind, visibility, icing/temperature extremes, electrical risks, and clear conditions. In addition, we define a severity function

$$\sigma_w : w \mapsto \{0, 1, 2, 3, 4\}, \tag{19}$$

where w is a weather token and $\sigma_w(w)$ gives its ordinal severity score (e.g., $\sigma_w(\text{clear}) = 0$, $\sigma_w(\text{rain}) = 1$, $\sigma_w(\text{icing\_conditions}) = 4$). This abstraction prevents the generation of implausible or unsafe combinations while preserving diversity across environmental conditions.

4) UAV Type Taxonomy: UAV platforms are classified into multirotors, rotorcraft, fixed-wing/gliders, and hybrid concepts. We model this as

$$\mathcal{C}_U : u \mapsto \{t_1, t_2, \ldots, t_k\}, \tag{20}$$

where u is a UAV family (e.g., *multirotors*) and $\{t_k\}$ are the concrete vehicle types (e.g., `quadrotor`, `hexacopter`). This taxonomy captures fundamental differences in dynamics and operational envelopes, ensuring that fixed-wing missions require runways, while rotorcraft missions can operate in constrained urban areas.

5) Prompt Integration: Finally, we integrate the taxonomies into the LLM prompt to guide scenario generation. Each prompt selects one element from each taxonomy and combines it with schema-level constraints. We formalize the axis tuple as

$$\theta = \langle s, a, w, u, \nu \rangle, \tag{21}$$

where $s \in \mathcal{C}_S$, $a \in \mathcal{C}_A$, $w \in \mathcal{C}_{\mathcal{E}}$, $u \in \mathcal{C}_U$, and ν is a random nonce that increases diversity. The prompt then embeds θ along with explicit constraints such as simulation duration, waypoint counts, or runway requirements. This structured design ensures that the LLM produces JSON objects that are not only valid but also operationally meaningful.

6) LLM Prompt Mechanics: We compose a single instruction that binds the taxonomy choices to the JSON schema and explicit guardrails. Let the prompt be $\Pi(\mathbb{S}, \mathbb{C}; \theta)$, where $\mathbb{S}$ is the verbatim JSON schema, $\mathbb{C}$ is the constraint bullet list, and $\theta = \langle s, a, w, u, \nu \rangle$ are the "axes": scenario type s, airspace a, weather w, UAV type u, and a nonce ν to decorrelate outputs. The function first samples θ from the canonicalized taxonomies, then interpolates u into type-conditional constraints (e.g., requiring `rotorcraft` and/or `aero` blocks), and finally concatenates *(i)* a strict preamble ("JSON only"), *(ii)* the axes header, *(iii)* the full schema $\mathbb{S}$, and *(iv)* the constraints $\mathbb{C}$. This "specification-by-example" design narrows the LLM's search space at generation time, significantly reducing out-of-range values (e.g., `dt`) and structural errors (e.g., missing geofence or runway). The nonce ν preserves diversity across repeated calls without weakening the constraints or the schema.

C. Validation Pipeline

The validation pipeline ensures that each scenario produced by the LLM is not only syntactically valid but also semantically consistent with the schema and physically plausible. This step is essential because large language models may generate well-structured outputs that nonetheless contain hidden inconsistencies or unrealistic mission details. By incorporating a multi-stage validation process, `UAVBench` transforms raw generative outputs into reliable and reusable benchmark assets. The process can be represented as Algorithm 1.

a) Discussion.: Algorithm 1 formalizes the multi-stage filtering process that ensures each generated scenario is valid. The input S denotes a candidate scenario, encoded as a structured mapping according to the schema $\mathcal{S}$ defined in Section III-A. The output is a Boolean validity flag $valid \in \{\texttt{true}, \texttt{false}\}$ indicating whether the scenario is accepted into the benchmark.

The first stage checks *schema compliance*. The set K contains all mandatory keys that must be present in every scenario, namely `name`, `seed`, `sim`, `uav`, `environment`, `airspace`, `spawn`, and `mission`. If any key $k \in K$ is missing from the domain $\text{dom}(S)$ of the scenario, the scenario is immediately rejected. This guarantees structural completeness and prevents parsing errors in downstream simulation.

The second stage enforces *constraint validation*. Let $s \in \mathcal{C}_S$ denote the mission type chosen in the scenario (e.g., `inspection`, `delivery`, `search_and_rescue`). For each mission type s, there exists a corresponding set of operational constraints $\mathcal{C}(s)$ that specifies which UAV types, airspace conditions, and weather profiles are admissible. The predicate $S \models \mathcal{C}(s)$ indicates that the scenario satisfies these rules. This step prevents illogical or unsafe pairings, such as fixed-wing aircraft operating underground or rotorcraft attempting satellite-relay missions.

The third stage verifies *geometric consistency*. Each waypoint $w = (x, y, z)$ is defined in three-dimensional space, where

---

Algorithm 1: Validation Pipeline for LLM-Generated Scenarios

Input: Scenario S generated by LLM
Output: Validity flag $valid \in \{\texttt{true}, \texttt{false}\}$
// Schema Compliance
Check that all required keys
  $K = \{\texttt{name}, \texttt{seed}, \texttt{sim}, \texttt{uav}, \texttt{environment},$
  $\texttt{airspace}, \texttt{spawn}, \texttt{mission}\}$ are present and
  well-typed;
if $\exists k \in K : k \notin dom(S)$ **then**
  | **return** false
else
  └ continue
// Constraint Validation
Let $s \leftarrow S.\texttt{mission.type}$ (where $s \in \mathcal{C}_S$);
Check that UAV type, airspace, and weather satisfy
  $\mathcal{C}(s)$;
if $S \not\models \mathcal{C}(s)$ **then**
  | **return** false
else
  └ continue
// Geometric Consistency
For each waypoint $w = (x, y, z)$ in S, verify
  $(x, y) \in G$ and $z_{\min} \le z \le z_{\max}$;
If any waypoint violates constraint: **return** false;
// Safety and Fault Checks
For all UAV pairs (i, j) compute distance d_{ij} and
  time-to-collision τ_{ij};
Check $d_{ij} \ge d_{\min}$ and $\tau_{ij} \ge \tau_{\min}$;
Validate each fault event (t_i, φ_i, s_i): $t_i \ge 0$, $s_i \le s_{\max}$;
If violations found: **return** false;
return true;

(x, y) are ground-plane coordinates and z is altitude above ground. The polygonal geofence $G \subset \mathbb{R}^2$ defines lateral bounds, while the altitude interval $[z_{\min}, z_{\max}]$ defines vertical limits. The condition $(x, y) \in G$ and $z_{\min} \le z \le z_{\max}$ ensures that all waypoints, spawn points, and landing sites remain within authorized operational boundaries. This prevents violations such as waypoints outside the geofence or below ground level, which would render scenarios infeasible in simulation.

The final stage applies *safety and fault checks*. For every pair of UAVs (i, j), the Euclidean distance d_{ij} and time-to-collision τ_{ij} are computed. These must satisfy $d_{ij} \ge d_{\min}$ and $\tau_{ij} \ge \tau_{\min}$, where $d_{\min}$ and $\tau_{\min}$ are safety thresholds defined in the schema block $\mathcal{D}$. Fault events are represented as tuples (t_i, φ_i, s_i), where t_i is the event start time, φ_i the fault type (e.g., motor failure, GNSS jam), and s_i the severity. The validator ensures that $t_i \ge 0$ and $s_i \le s_{\max}$, thereby excluding unrealistic cases such as an instantaneous catastrophic fault at mission start or a severity outside calibrated limits.

In summary, the validation pipeline acts as a layered filter that combines schema-level checks (K), operational constraints ($\mathcal{C}(s)$), geometric feasibility (G, $[z_{\min}, z_{\max}]$), and safety thresholds ($d_{\min}, \tau_{\min}, s_{\max}$). This guarantees that only structurally complete, logically coherent, spatially consistent, and operationally safe scenarios are admitted to the dataset. Such

rigor is crucial for benchmarking agentic AI systems, since it ensures that evaluation results reflect meaningful performance rather than artifacts of poorly constructed scenarios.

D. Risk & Safety Labeling

The labeling process assigns each validated scenario a discrete risk level and a categorical safety tag. This is implemented as a deterministic algorithm that combines hazard detection, environmental conditions, and mission context into a unified scoring procedure.

Algorithm 2: Risk and Safety Labeling Procedure

Input: Scenario $S = (H, E, M)$, where H = hazards,
  E = environment, M = mission parameters
Output: Risk level $\rho(S) \in \{0, 1, 2, 3\}$, Safety category
  $\sigma(S) \in \Sigma$
$F(S) \leftarrow$ detect hazards from H (e.g., icing, GNSS
  jamming, battery failure);
$v_{\text{wind}}, \gamma_{\text{vis}}, n_{\text{swarm}} \leftarrow$ extract environmental features from
  E;
if $F(S) \ne \emptyset$ **then**
  | $\rho_{\text{hazards}}(S) \leftarrow$ max severity of hazards in $F(S)$;
else
  └ $\rho_{\text{hazards}}(S) \leftarrow 0$;
$\rho_{\text{env}}(S) \leftarrow$ severity score based on thresholds:
  if $v_{\text{wind}} > v_{\text{th}}$ then add penalty;
  if $\gamma_{\text{vis}} =$ poor then add penalty;
  if $n_{\text{swarm}} > n_{\text{th}}$ then add penalty;
$\rho(S) \leftarrow \max(\rho_{\text{hazards}}(S), \rho_{\text{env}}(S))$;
$\sigma(S) \leftarrow$ assign category in Σ based on dominant hazard
  (e.g., Weather, Navigation, Energy,
  Collision-Avoidance);
return $\rho(S), \sigma(S)$;

Algorithm 2 formalizes the assignment of safety metadata to each scenario. The input $S = (H, E, M)$ decomposes a scenario into three blocks: hazard events H, environmental conditions E, and mission parameters M. The function $F(S)$ extracts the set of active hazards, such as icing events, GNSS jamming, or battery failures. If hazards are present, their maximum severity is recorded as $\rho_{\text{hazards}}(S)$; otherwise, this value defaults to 0. Environmental features are extracted as wind speed v_{wind}, visibility class γ_{vis}, and swarm size n_{swarm}, all of which are compared against operational thresholds $v_{\text{th}}, \gamma_{\text{th}}, n_{\text{th}}$ to compute an environmental risk contribution $\rho_{\text{env}}(S)$. The final risk score $\rho(S)$ is the maximum of hazard and environment contributions, thereby prioritizing catastrophic hazards while still capturing adverse operating conditions.

The second output, $\sigma(S)$, provides an interpretable categorical safety tag. The set Σ is partitioned into domains such as `Weather` (e.g., icing, lightning), `Navigation` (e.g., GNSS spoofing, link loss), `Energy` (e.g., low battery, fuel exhaustion), and `Collision-Avoidance` (e.g., separation breaches in swarms). The assignment rule maps each scenario to the dominant category associated with its highest-severity hazard or environmental stressor. This two-level labeling framework produces not only a scalar risk level $\rho(S) \in \{0, 1, 2, 3\}$

but also a categorical tag $\sigma(S) \in \Sigma$, enabling both coarse-grained benchmarking and fine-grained analysis of failure modes.

In summary, the risk and safety labeling step transforms raw scenario metadata into standardized, reproducible indicators. The quantitative score $\rho(S)$ facilitates statistical benchmarking across large datasets, while the categorical tag $\sigma(S)$ enhances interpretability by linking risk to root causes. Together, these labels make `UAVBench` suitable for evaluating agentic AI systems under both nominal and safety-critical conditions.

IV. EXPERIMENTS AND RESULTS

This section presents the experimental evaluation of reasoning in agentic AI–driven UAV systems using `UAVBench` and its structured extension `UAVBench_MCQ`. We outline the reasoning framework, describe the generation of structured MCQs, and report model performance across diverse reasoning domains to highlight current capabilities and remaining challenges in UAV autonomy.

A. Reasoning Styles and `UAVBench_MCQ` Framework

To evaluate reasoning in agentic AI–driven UAV systems, we extend the dataset with `UAVBench_MCQ`, a unified framework for structured reasoning and benchmarking. We define ten reasoning styles—covering aerodynamics, navigation, policy compliance, environmental sensing, multi-agent coordination, cyber-physical security, energy management, ethics and safety, comparative systems, and hybrid integration—each guided by a style-specific prompt and validation rule set. Validated `UAVBench` scenarios are transformed into self-contained JSON multiple-choice questions containing the scenario description, question, options, correct answer, rationale, and metadata. The framework enforces grounded realism, structural completeness, style-dependent option counts, and length limits to ensure consistent, large-scale, and programmatically gradable evaluation of UAV reasoning. Fig. 4 illustrates this pipeline, in which each scenario is mapped to a reasoning style, processed with LLM-based prompts, validated for schema and logic, and stored as standardized JSON objects for reproducible benchmarking.

B. UAVBench_MCQ: Structured Multi-Style MCQ Generation

`UAVBench_MCQ` transforms validated `UAVBench` scenarios into structured, interpretable, and machine-readable MCQs. Each MCQ is a self-contained JSON object that includes the scenario description, question, labeled options, the correct choice, rationale, and metadata such as style identifier, generator model, and schema version. The dataset is thus both human-interpretable and programmatically gradable, enabling large-scale benchmarking of UAV reasoning agents.

1) MCQ Representation: Each generated MCQ follows a standardized JSON schema:

$$q = \langle D, S, Q, \mathcal{O}, i^*, R, h \rangle, \tag{22}$$

where $\mathcal{O}$ represents the ordered set of candidate options, and i^* the single correct answer. The schema ensures backward compatibility across updates, traceability to source scenarios, and interoperability with automated evaluation pipelines.

2) Design Constraints: To maintain reliability and interpretability across reasoning styles, the generation process enforces strict constraints:

- *Grounded realism:* Each MCQ must reference only facts available in the original scenario JSON.
- *Structured completeness:* The fields `question`, `choices`, `correct_choice`, and `reason` are mandatory.
- *Consistency rules:* Exactly one correct option must exist; distractors must remain locally plausible but violate at least one constraint relevant to style S.
- *Compactness:* Question length ≤ 28 words; choice length ≤ 14 words.
- *Ethical schema:* For example, Style 8, seven options (A–G) are used to encode ethical trade-offs with explicit prioritization of human safety.

Algorithm 3: UAVBench_MCQ Multi-Style Generation Pipeline

Input: Validated scenario $\mathcal{S}$, reasoning style S
Output: Structured MCQ object q linked to $\mathcal{S}$
`// Stage 1: Scenario Description`
$D \leftarrow$ GENERATEDESCRIPTION($\mathcal{S}$);
    Invoke LLM with a style-specific system prompt to produce a concise description (≤ 10 sentences).;
    Sanitize and attach description to $\mathcal{S}$ for contextual grounding.;
`// Stage 2: MCQ Generation`
$q' \leftarrow$ MAKEMCQ(D, S) using style-specific prompt template.;
    Extract fields $\{Q, \mathcal{O}, i^*, R\}$; validate schema, label format, and distinct options.;
    If validation fails, retry up to $R = 3$ iterations.;
`// Stage 3: Metadata and Persistence`
Compute hash h and assemble
  $q = \langle D, S, Q, \mathcal{O}, i^*, R, h \rangle$.;
Save as
  `<scenario_name>_{S}_{h}_mcq.json`;
return q.

Algorithm 3 illustrates a modular, style-driven generation pipeline that enforces format, validity, and realism. The design separates descriptive grounding from question synthesis, ensuring that each LLM instance focuses on reasoning rather than scenario rewriting. The retry mechanism and schema validation safeguard against malformed or logically inconsistent outputs, while the hash-based persistence guarantees reproducibility and deduplication across large-scale generations.

`UAVBench_MCQ` provides an interpretable and standardized bridge between simulation-grounded UAV data and reasoning-based evaluation. By incorporating ten reasoning styles and structured JSON outputs, it enables both quantitative benchmarking and qualitative insight into LLMs' decision integrity. The schema's inclusion of metadata (e.g., `schema_version`, `style_id`, and `hash`) ensures transparent provenance tracking and long-term dataset evolution. Ultimately, `UAVBench_MCQ` advances the evaluation of UAV
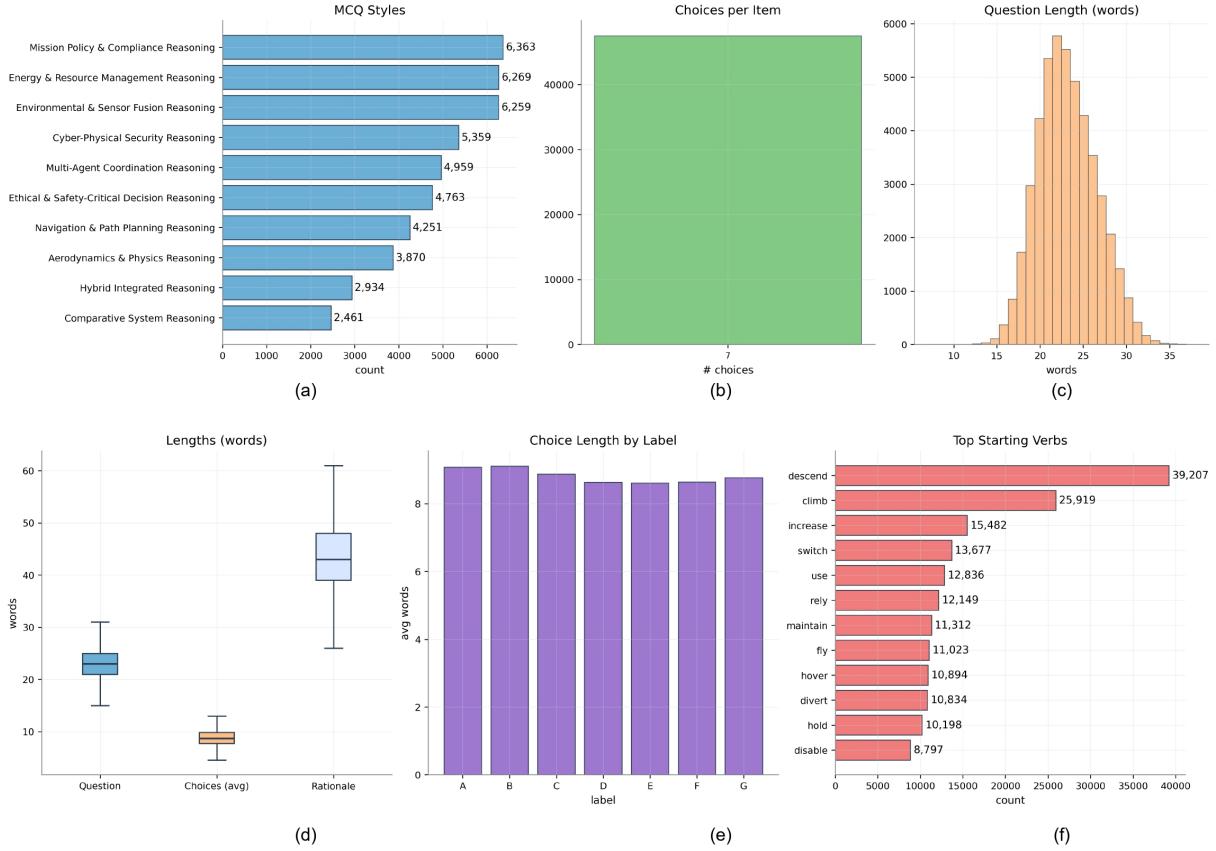
Fig. 3: Overview of `UAVBench_MCQ` dataset structure and linguistic statistics. (a) Distribution of multiple-choice question (MCQ) styles across reasoning domains; (b) Number of answer choices per question; (c) Distribution of question lengths in words; (d) Comparison of word counts for questions, averaged choices, and rationales; (e) Average choice length by option label (A–G); and (f) The most frequent starting verbs in the choice text. Together, these subfigures summarize the content balance, linguistic complexity, and stylistic diversity of `UAVBench_MCQ` items.

TABLE IV: Reasoning styles defined in `UAVBench` for comprehensive evaluation of UAV cognitive and ethical reasoning.

ID	Reasoning Style	Focus and Evaluation Scope
1	Aerodynamics & Physics Reasoning	Models flight mechanics, including lift, drag, thrust, and control stability. Evaluates physically plausible flight and awareness of aerodynamic constraints.
2	Navigation & Path Planning Reasoning	Tests trajectory optimization, obstacle avoidance, and spatial reasoning under time and energy constraints.
3	Mission Policy & Compliance Reasoning	Evaluates adherence to airspace regulations, operational limits, and mission rules (e.g., NFZ, BVLOS, privacy).
4	Environmental & Sensor Fusion Reasoning	Assesses understanding of environmental conditions, sensor fusion, and perception reliability under uncertainty.
5	Multi-Agent Coordination Reasoning	Focuses on cooperative UAV behavior, communication, and deconfliction among multiple agents in dynamic environments.
6	Cyber-Physical Security Reasoning	Evaluates response to spoofing, jamming, or sensor compromise, testing situational awareness and integrity preservation.
7	Energy & Resource Management Reasoning	Analyzes energy-efficient decision-making, load balancing, and mission prioritization under resource limitations.
8	Ethical & Safety-Critical Decision Reasoning	Captures moral trade-offs, safety-of-life priorities, lawful conduct, and responsible autonomy during emergencies.
9	Comparative System Reasoning	Compares UAV designs, control strategies, or architectures to infer performance trade-offs and optimal configurations.
10	Hybrid Integrated Reasoning	Integrates multiple reasoning domains (e.g., navigation + ethics + resource) to test multi-objective mission optimization.

autonomy by combining physical realism, cognitive depth, and ethical accountability within a reproducible benchmarking ecosystem.

Fig. 3 shows the composition and linguistic patterns of the `UAVBench_MCQ` dataset. It includes distributions of question styles, number of answer choices, and word-length statistics
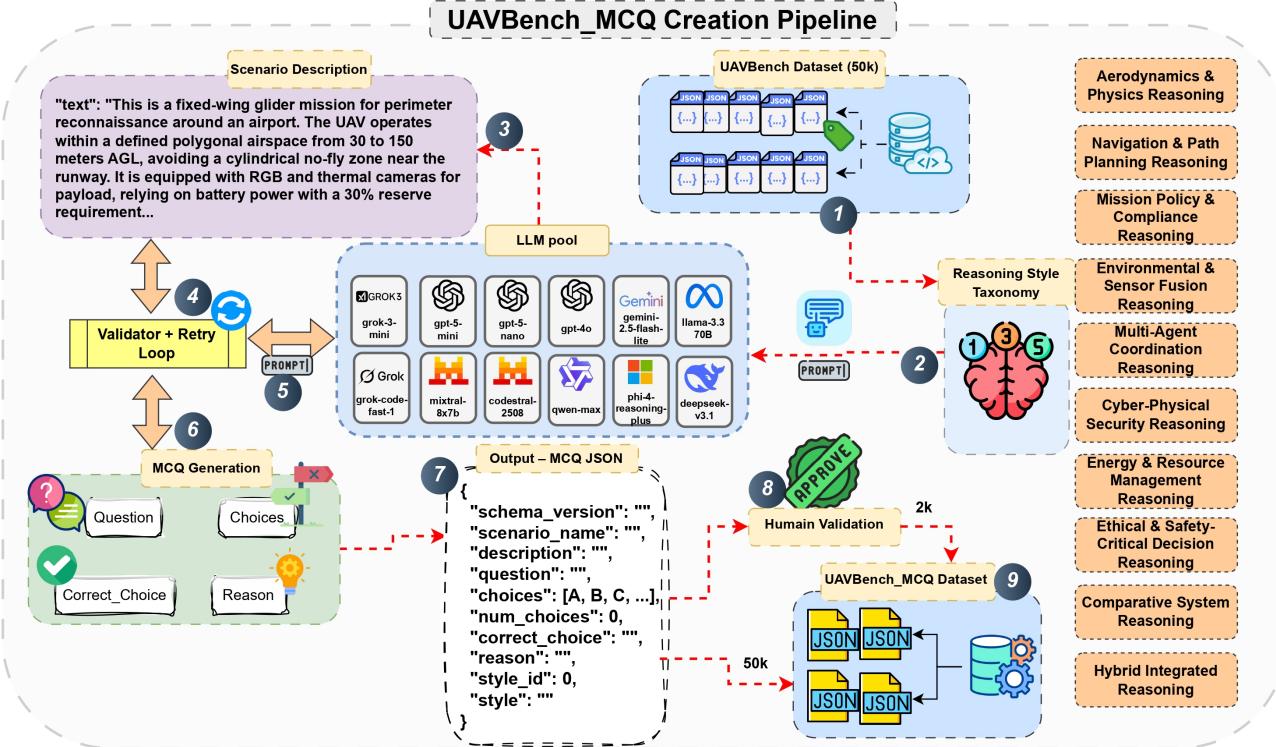
Fig. 4: `UAVBench_MCQ` Creation Pipeline.

TABLE V: Notation used in the `UAVBench_MCQ` generation process.

Symbol	Block	Definition
$\mathcal{S}$	Input	Validated `UAVBench` scenario in structured JSON format.
D	Input	Natural-language description derived from $\mathcal{S}$, summarizing mission type, UAV configuration, and constraints.
S	Context	Reasoning style identifier $(1-10)$ guiding prompt selection and validation logic.
Q	Output	MCQ question targeting reasoning consistent with style S.
$C = \{C_A, \ldots, C_G\}$	Output	Labeled set of candidate options; typically four or seven depending on style.
C^*	Output	Correct option satisfying physical, logical, or ethical constraints.
R	Reasoning	Explanation or rationale justifying the correct choice.
$\rho(Q)$	Risk	Embedded risk or severity level in the question context (low–critical).
$\Pi(D, S)$	Mapping	Prompting function transforming (D, S) into structured MCQ output.
h	Metadata	Content hash for versioning and deduplication.

for questions, choices, and rationales. The overall balance and variety across reasoning styles, lexical structures, and choice formulations illustrate the dataset's breadth and quality for UAV-related reasoning tasks.

C. Evaluation Metrics

To comprehensively assess reasoning performance across diverse UAV mission domains, we adopt a four-metric evaluation framework that captures both overall correctness and cross-style consistency. While raw *Accuracy* measures general task performance, it fails to distinguish between models that perform well in certain reasoning styles but poorly in others. To address this, we introduce three complementary statistics—*Mean Accuracy*, *Standard Deviation*, and the *Balanced Style Score (BSS)*—computed from the per-style accuracies $\{a_s\}_{s=1}^{S}$, where $S = 10$ denotes the number of reasoning styles defined in `UAVBench_MCQ`.

a) Accuracy (%): Overall accuracy measures the proportion of correctly answered multiple-choice questions across the entire benchmark:

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100, \tag{23}$$

where N_{correct} and N_{total} denote the number of correct responses and total questions, respectively. This metric captures aggregate correctness independent of reasoning style.

b) Mean Accuracy (%): To evaluate average performance across reasoning categories, we compute the mean of per-style accuracies:

$$\bar{a} = \frac{1}{S} \sum_{s=1}^{S} a_s, \tag{24}$$

where a_s represents the accuracy for reasoning style s. High $\bar{a}$ values indicate generally strong performance across all cognitive dimensions of UAV reasoning.

TABLE VI: Accuracy (%) on *Perception & Physical World* reasoning styles in UAVBench.

Model	Company	Size	License	(1) Aerodynamics & Physics	(4) Environmental & Sensor Fusion	Avg.
Qwen3 235B A22B (2507)	Alibaba	235B	Open	82.500	97.000	89.800
ChatGPT 4o	OpenAI	N/A	Proprietary	74.500	96.500	85.500
GPT-5 Chat	OpenAI	N/A	Proprietary	73.500	97.000	85.300
Qwen3 Max	Alibaba	N/A	Open	73.500	96.000	84.800
Mistral Medium 3.1	Mistral AI	N/A	Proprietary	72.500	94.500	83.500
ERNIE 4.5 300B A47B	Baidu	300B	Open	71.000	96.000	83.500
GPT-4.1 Mini	OpenAI	N/A	Proprietary	68.000	97.500	82.800
InternVL3 78B	OpenGVLab	78B	Open	69.000	96.500	82.800
GPT-4.1	OpenAI	N/A	Proprietary	69.500	95.500	82.500
GPT-4.1	OpenAI	N/A	Proprietary	69.500	95.500	82.500
Kimi K2	Moonshot AI	1T	Open	69.500	95.000	82.300
Claude-haiku-4.5	Anthropic	N/A	Proprietary	68.000	94.500	81.300
Phi 4 Reasoning Plus	Microsoft	14B	Open	65.500	97.000	81.300
Gemini 2.5 Flash	Google	391B	Proprietary	65.500	96.000	80.800
Qwen3 VL 8B Instruct	Alibaba	8B	Open	64.500	96.500	80.500
DeepSeek Chat V3 (0324)	DeepSeek	685B	Open	65.000	95.500	80.300
DeepSeek V3.1 Terminus	DeepSeek	N/A	Open	62.500	94.500	78.500
DeepSeek V3.2 Exp	DeepSeek	N/A	Open	61.000	95.000	78.000
Llama-4-scout	Meta	17B	Open	59.000	96.500	77.800
Grok 4 Fast	xAI	N/A	Proprietary	60.000	89.500	74.800
Qwen 2.5 7B Instruct	Alibaba	7B	Open	54.500	91.000	72.800
LFM 2 2.6B	Liquid AI	2.6B	Open	49.000	95.500	72.300
Gemma-3n-e4b-it	Google	4B	Open	49.000	94.500	71.800
Olmo 2 32B Instruct	AllenAI	32B	Open	49.000	91.500	70.300
LFM2-8B-A1B	Liquid AI	8B	Open	47.000	92.000	69.500
Llama 3.1 8B Instruct	Meta	8B	Open	46.000	92.500	69.300
Jamba-mini-1.7	AI21	N/A	Open	48.500	86.000	67.300
Llama 3.2 3B Instruct	Meta	3B	Open	37.500	86.500	62.000
Granite-4.0-h-micro	IBM	3B	Open	36.000	87.000	61.500
Claude Sonnet 4.5	Anthropic	468B	Proprietary	36.000	81.000	58.500
GLM-4.6	Z.AI	357B	Open	34.500	36.500	35.500
Qwen3-30B-A3B	Alibaba	30B	Open	3.500	7.500	5.500
Nemotron Nano 9B V2	NVIDIA	9B	Open	3.000	0.000	1.500

LLM parameters: `top_p = 1.0`, which is the nucleus sampling parameter (1.0 = all tokens considered); `max_tokens = 16`, which defines the maximum number of tokens generated; `temperature = 0.0`, which controls randomness (0 = deterministic output); and `max_retries = 5`, which specifies the maximum number of retry attempts in case of LLM failure.

c) Standard Deviation (%): To quantify performance consistency, we calculate the standard deviation of accuracies across all reasoning styles:

$$\sigma(a) = \sqrt{\frac{1}{S}\sum_{s=1}^{S}(a_s - \overline{a})^2}. \qquad (25)$$

Lower $\sigma(a)$ values indicate balanced reasoning ability, while higher values reveal specialization or weakness in certain domains (e.g., strong in physics but weak in ethics).

d) Balanced Style Score (BSS): Finally, we propose the *Balanced Style Score* (BSS) as an integrated indicator of both accuracy and balance. BSS combines the geometric mean of per-style accuracies with a penalty term for imbalance:

$$\text{BSS} = \left(\prod_{s=1}^{S}(a_s + \varepsilon)^{1/S}\right) \times \left(1 - \frac{\sigma(a)}{\overline{a}}\right), \qquad (26)$$

where ε is a small constant (10^{-6}) to avoid undefined logarithms. The first term rewards uniformly high performance, while the second penalizes uneven distribution across reasoning styles. BSS values lie in $[0, 1]$, with higher scores indicating both accurate and consistent reasoning—a key property for safe, reliable UAV autonomy.

Together, these four metrics provide a multidimensional assessment of LLM reasoning under UAVBench_MCQ. *Accuracy* captures raw task success, *Mean Accuracy* reflects general competence, *Standard Deviation* measures balance across reasoning styles, and *BSS* synthesizes them into a single interpretable metric that rewards models exhibiting both correctness and cross-domain consistency—an essential criterion for trustworthy UAV decision-making.

D. Performance on Perception and Physical World Reasoning

Table VI presents the results of model performance on UAVBench's *Perception & Physical World* reasoning tasks, which assess a model's capability to understand aerodynamics, environmental dynamics, and sensor fusion scenarios. Among all evaluated systems, *Qwen3 235B A22B* achieves the highest average accuracy of *89.8%*, outperforming leading proprietary models such as *ChatGPT 4o* (85.5%) and *GPT-5 Chat* (85.3%). Open-source models from Alibaba, including Qwen3 Max and Qwen3 VL 8B Instruct, consistently rank among the top performers, indicating the growing competitiveness of open models in physics-grounded reasoning. Proprietary systems from OpenAI and Mistral also demonstrate strong and stable results across both reasoning categories, suggesting robust internalization of physical and environmental relationships even in UAV-specific contexts.

Smaller, lightweight open models (e.g., *Llama 3.1 8B*, *Gemma-3n-e4b-it*, and *Olmo 2 32B*) exhibit a marked decline in performance, with average accuracies ranging from 61–70%. This trend suggests that reasoning over aerial dynamics and sensor-based perception remains highly dependent on model scale and domain-specific training. At the lower end, *Nemotron Nano 9B V2* and *Qwen3-30B-A3B* perform poorly (below 6%), revealing limited generalization to grounded physical reasoning. Across nearly all models, accuracies on *Environmental & Sensor Fusion* tasks exceed those on *Aerodynamics & Physics*, implying that current LLMs integrate perceptual and multimodal cues more effectively than they infer dynamic physical laws. Overall, these findings indicate that while large-scale, instruction-tuned models—both open and proprietary—are achieving near-human reliability in perceptual

reasoning, mastering fine-grained aerodynamics and UAV physics remains an open research challenge.

E. Performance on Planning, Coordination, and Resource Reasoning

Table VII reports the accuracy of leading LLMs on UAVBench's *Planning, Coordination & Resources* reasoning tasks, encompassing *Navigation & Path Planning*, *Multi-Agent Coordination*, and *Energy & Resource Management*. The results indicate that *Qwen3 235B A22B* again achieves the highest overall performance with an average accuracy of *76.5%*, demonstrating balanced competence across trajectory optimization, obstacle avoidance, and energy-aware planning. Proprietary models such as *GPT-5 Chat* (72.8%) and *ChatGPT 4o* (71.7%) follow closely, reflecting their strength in dynamic decision-making and temporal-spatial reasoning. Open-source systems like *Qwen3 Max* and *Phi 4 Reasoning Plus* also perform competitively, suggesting that well-tuned open models are closing the gap in complex reasoning domains. In contrast, *GPT-4.1* exhibits notably strong navigation performance (82.5%) but comparatively weaker coordination and resource management, suggesting a bias toward single-agent spatial reasoning.

Performance trends across subtasks reveal that *Navigation & Path Planning* generally yields higher accuracies than the other two categories, emphasizing that most LLMs handle structured spatial reasoning better than cooperative or resource-constrained scenarios. *Multi-Agent Coordination* and *Energy & Resource Management* tasks, which require distributed decision-making and trade-off optimization, remain challenging across all models, with even top performers achieving below 80%. Smaller open models such as *Llama 3.1 8B*, *Gemma-3n-e4b-it*, and *DeepSeek V3.2 Exp* average between 56–66%, while lightweight architectures like *Nemotron Nano 9B V2* and *Qwen3-30B-A3B* fall below 6%. These results collectively suggest that while frontier models demonstrate emerging capabilities in autonomous planning, true competence in cooperative multi-agent coordination and energy-aware mission optimization remains an open research frontier for both open and proprietary LLMs.

F. Performance on Governance, Ethics, and Security Reasoning

Table VIII summarizes model performance on UAVBench's *Governance, Ethics & Security* reasoning tasks, which evaluate compliance with mission regulations, ethical decision-making under safety-critical conditions, and robustness against cyber-physical threats. The *Qwen3 235B A22B* model leads with an average accuracy of *82.7%*, demonstrating exceptional competence in enforcing airspace policy and making high-stakes decisions. Proprietary models such as *ChatGPT 4o* (80.7%) and *GPT-5 Chat* (79.8%) closely follow, confirming their strength in ethical reasoning and operational rule interpretation. Open-source competitors like *Qwen3 Max* (80.5%) and *DeepSeek Chat V3* (78.8%) also perform robustly, demonstrating that governance-related reasoning is increasingly tractable for large open models. Interestingly, all high-performing models show particularly strong accuracy in *Cyber-Physical Security* reasoning (95–98%), suggesting that integrity-preservation

and threat-response scenarios are well-captured in large-scale pretraining corpora.

Across subtasks, however, *Mission Policy & Compliance* and *Ethical & Safety-Critical Decision* reasoning remain more challenging than security-focused reasoning. Even top-tier models exhibit a noticeable performance gap—approximately 20 percentage points—between regulatory or moral judgment and cyber-physical threat handling. This indicates that while LLMs can recognize and describe technical countermeasures (e.g., against spoofing or jamming), they still struggle with normative constraints, lawful autonomy, and ethical trade-offs under uncertainty. Smaller open models (e.g., *Llama 3.1 8B*, *Qwen 2.5 7B*, and *Olmo 2 32B*) yield averages between 60–65%, reflecting their limited abstraction capacity for contextually nuanced or policy-dependent reasoning. Overall, the results suggest that while modern LLMs have made major strides in UAV security interpretation, achieving human-level ethical alignment and mission-compliance awareness remains a critical and unsolved dimension of safe autonomous operation.

G. Performance on Systems and Integration Reasoning

Table IX presents `UAVBench` results for *Systems & Integration* reasoning, which includes *Comparative System Reasoning* and *Hybrid Integrated Reasoning*. These categories assess a model's capacity to compare UAV architectures, control designs, and mission configurations while optimizing across multiple reasoning domains. The top-performing model, *Qwen3 235B A22B*, attains an impressive *89.3%* average accuracy, demonstrating a strong holistic understanding of UAV system trade-offs and integration principles. Close competitors such as *ChatGPT 4o* (87.8%), *Qwen3 Max* (87.5%), and *GPT-4.1* (87.3%) exhibit similarly high proficiency, indicating that both open and proprietary models have achieved mature competency in systems-level reasoning. Notably, *Comparative System* reasoning yields the highest individual accuracies across all models—often exceeding 95%—suggesting that performance evaluation and architecture comparison are well-aligned with the statistical and analytic strengths of large LLMs.

However, the more complex *Hybrid Integrated Reasoning* task, which requires blending ethical, navigational, and resource-related reasoning to optimize multi-objective missions, remains a consistent bottleneck. Even the best-performing models score between 77–83%, underscoring the difficulty of integrating heterogeneous reasoning modes into cohesive decisions. Mid-range open models such as *Gemini 2.5 Flash*, *InternVL3 78B*, and *Phi 4 Reasoning Plus* maintain averages around 83–84%, while smaller architectures like *Llama 3.1 8B* and *Qwen 2.5 7B* show a steep decline to roughly 73–74%. At the lower end, models like *Granite-4.0-h-micro* and *Jamba-mini-1.7* struggle with integrated reasoning (below 70%), and minimal-capacity models such as *Qwen3-30B-A3B* and *Nemotron Nano 9B V2* fail almost entirely. Overall, while large-scale LLMs now excel in comparative system evaluation, achieving coherent integration across diverse UAV mission domains remains a critical next step toward fully autonomous, context-aware reasoning systems.

TABLE VII: Accuracy (%) on *Planning, Coordination & Resources* reasoning styles in UAVBench.

Model	Company	Size	License	(2) Navigation & Path	(5) Multi-Agent Coord.	(7) Energy & Resource	Avg.
Qwen3 235B A22B (2507)	Alibaba	235B	Open	81.500	76.500	71.500	76.500
GPT-5 Chat	OpenAI	N/A	Proprietary	78.000	72.000	68.500	72.800
ChatGPT 4o	OpenAI	N/A	Proprietary	80.500	70.000	64.500	71.700
Qwen3 Max	Alibaba	N/A	Open	77.000	70.500	65.000	70.800
GPT-4.1	OpenAI	N/A	Proprietary	82.500	67.000	62.500	70.700
GPT-4.1 Mini	OpenAI	N/A	Proprietary	75.500	71.000	64.500	70.300
Phi 4 Reasoning Plus	Microsoft	14B	Open	76.500	67.000	67.000	70.200
Kimi K2	Moonshot AI	1T	Open	67.500	71.500	70.000	69.700
Gemini 2.5 Flash	Google	391B	Proprietary	73.500	69.500	66.000	69.700
InternVL3 78B	OpenGVLab	78B	Open	71.500	67.500	68.000	69.000
Llama-4-scout	Meta	17B	Open	72.000	71.500	63.500	69.000
Qwen3 VL 8B Instruct	Alibaba	8B	Open	76.000	64.500	66.000	68.800
Claude-haiku-4.5	Anthropic	N/A	Proprietary	73.000	68.000	62.500	67.800
ERNIE 4.5 300B A47B	Baidu	300B	Open	71.500	68.000	63.000	67.500
Mistral Medium 3.1	Mistral AI	N/A	Proprietary	69.000	67.500	65.000	67.200
Gemma-3n-e4b-it	Google	4B	Open	63.500	63.500	71.000	66.000
DeepSeek Chat V3 (0324)	DeepSeek	685B	Open	68.500	65.500	63.000	65.700
Grok 4 Fast	xAI	N/A	Proprietary	69.500	59.500	58.000	62.300
DeepSeek V3.2 Exp	DeepSeek	N/A	Open	63.000	62.500	61.000	62.200
LFM 2 2.6B	Liquid AI	2.6B	Open	65.000	62.000	55.500	60.800
DeepSeek V3.1 Terminus	DeepSeek	N/A	Open	59.500	58.000	62.500	60.000
Llama 3.1 8B Instruct	Meta	8B	Open	57.000	58.500	59.500	58.300
Qwen 2.5 7B Instruct	Alibaba	7B	Open	60.500	52.000	61.000	57.800
Llama 3.2 3B Instruct	Meta	3B	Open	54.000	55.000	59.000	56.000
Olmo 2 32B Instruct	AllenAI	32B	Open	57.000	60.000	51.000	56.000
LFM2-8B-A1B	Liquid AI	8B	Open	65.500	57.000	41.000	54.500
Jamba-mini-1.7	AI21	N/A	Open	54.500	55.000	43.500	51.000
Claude Sonnet 4.5	Anthropic	468B	Proprietary	53.000	50.500	41.000	48.200
Granite-4.0-h-micro	IBM	3B	Open	50.000	43.500	50.500	48.000
GLM-4.6	Z.AI	357B	Open	31.500	47.500	32.000	37.000
Qwen3-30B-A3B	Alibaba	30B	Open	4.500	6.500	4.000	5.000
Nemotron Nano 9B V2	NVIDIA	9B	Open	1.000	0.500	1.000	0.800

LLM parameters: `top_p = 1.0`, which is the nucleus sampling parameter (1.0 = all tokens considered); `max_tokens = 16`, which defines the maximum number of tokens generated; `temperature = 0.0`, which controls randomness (0 = deterministic output); and `max_retries = 5`, which specifies the maximum number of retry attempts in case of LLM failure.

TABLE VIII: Accuracy (%) on *Governance, Ethics & Security* reasoning styles in UAVBench.

Model	Company	Size	License	(3) Policy & Compliance	(8) Ethical & Safety-Critical	(6) Cyber-Physical Sec.	Avg.
Qwen3 235B A22B (2507)	Alibaba	235B	Open	76.000	75.500	96.500	82.700
ChatGPT 4o	OpenAI	N/A	Proprietary	72.000	73.000	97.000	80.700
Qwen3 Max	Alibaba	N/A	Open	68.500	76.000	97.000	80.500
GPT-5 Chat	OpenAI	N/A	Proprietary	65.500	76.000	98.000	79.800
GPT-4.1	OpenAI	N/A	Proprietary	73.000	70.000	96.000	79.700
DeepSeek Chat V3 (0324)	DeepSeek	685B	Open	66.000	75.500	95.000	78.800
DeepSeek V3.2 Exp	DeepSeek	N/A	Open	61.000	77.500	96.000	78.200
Kimi K2	Moonshot AI	1T	Open	69.000	68.500	96.500	78.000
GPT-4.1 Mini	OpenAI	N/A	Proprietary	68.000	67.000	97.500	77.500
Gemini 2.5 Flash	Google	391B	Proprietary	62.000	71.500	97.000	76.800
InternVL3 78B	OpenGVLab	78B	Open	62.500	72.000	96.000	76.800
Mistral Medium 3.1	Mistral AI	N/A	Proprietary	59.000	75.000	96.500	76.800
Claude-haiku-4.5	Anthropic	N/A	Proprietary	68.000	67.000	95.000	76.700
DeepSeek V3.1 Terminus	DeepSeek	N/A	Open	59.500	72.000	96.000	75.800
Gemma-3n-e4b-it	Google	4B	Open	61.500	72.000	95.000	76.200
Phi 4 Reasoning Plus	Microsoft	14B	Open	57.000	73.500	96.500	75.700
Grok 4 Fast	xAI	N/A	Proprietary	60.500	69.500	94.000	74.700
Llama-4-scout	Meta	17B	Open	63.000	63.500	96.000	74.200
ERNIE 4.5 300B A47B	Baidu	300B	Open	59.500	68.000	94.500	74.000
Qwen3 VL 8B Instruct	Alibaba	8B	Open	62.500	62.000	97.000	73.800
LFM 2 2.6B	Liquid AI	2.6B	Open	57.500	57.500	94.500	69.800
Olmo 2 32B Instruct	AllenAI	32B	Open	54.500	61.500	88.500	68.200
LFM2-8B-A1B	Liquid AI	8B	Open	47.000	57.500	95.000	66.500
Qwen 2.5 7B Instruct	Alibaba	7B	Open	47.000	53.500	93.000	64.500
Llama 3.1 8B Instruct	Meta	8B	Open	45.000	57.000	91.000	64.300
Claude Sonnet 4.5	Anthropic	468B	Proprietary	48.000	49.000	91.000	62.700
Llama 3.2 3B Instruct	Meta	3B	Open	43.500	47.500	87.500	59.500
Granite-4.0-h-micro	IBM	3B	Open	37.500	54.000	87.000	59.500
Jamba-mini-1.7	AI21	N/A	Open	32.000	45.000	88.500	55.200
GLM-4.6	Z.AI	357B	Open	41.500	41.500	66.500	49.800
Qwen3-30B-A3B	Alibaba	30B	Open	4.000	6.000	10.000	6.700
Nemotron Nano 9B V2	NVIDIA	9B	Open	1.500	2.500	11.000	5.000

LLM parameters: `top_p = 1.0`, which is the nucleus sampling parameter (1.0 = all tokens considered); `max_tokens = 16`, which defines the maximum number of tokens generated; `temperature = 0.0`, which controls randomness (0 = deterministic output); and `max_retries = 5`, which specifies the maximum number of retry attempts in case of LLM failure.

H. Aggregate Performance and Cross-Style Balance

Figure 5 presents a comparative summary of the top fifteen large language models (LLMs) evaluated under the `UAVBench_MCQ` framework. The three subplots respectively illustrate mean accuracy, cross-style consistency (standard deviation of per-style accuracies), and the proposed Balanced Style Score (BSS). Each bar is annotated with the model name and its corresponding performance value, providing a clear view of how overall accuracy and reasoning balance vary across models. The results are ordered by BSS, which rewards models that are both accurate and consistent across reasoning styles.

In panel (a), the mean accuracy results show that the highest performing models, such as *Qwen3 235B A22B*, *ChatGPT 4o*, and *GPT–5 Chat*, achieve overall accuracies between 80% and

TABLE IX: Accuracy (%) on *Systems & Integration* reasoning styles in UAVBench.

Model	Company	Size	License	(9) Comparative System	(10) Hybrid Integrated	Avg.
Qwen3 235B A22B (2507)	Alibaba	235B	Open	95.500	83.000	89.300
Qwen3 Max	Alibaba	N/A	Open	96.500	78.500	87.500
ChatGPT 4o	OpenAI	N/A	Proprietary	96.500	79.000	87.800
Claude-haiku-4.5	Anthropic	N/A	Proprietary	94.000	80.500	87.300
GPT-4.1	OpenAI	N/A	Proprietary	97.000	77.500	87.300
GPT-5 Chat	OpenAI	N/A	Proprietary	95.500	77.500	86.500
GPT-4.1 Mini	OpenAI	N/A	Proprietary	95.000	77.000	86.000
Qwen3 VL 8B Instruct	Alibaba	8B	Open	94.500	76.000	85.300
Kimi K2	Moonshot AI	1T	Open	96.000	74.000	85.000
Mistral Medium 3.1	Mistral AI	N/A	Proprietary	95.000	74.500	84.800
InternVL3 78B	OpenGVLab	78B	Open	94.000	74.000	84.000
Phi 4 Reasoning Plus	Microsoft	14B	Open	93.500	74.000	83.800
Gemini 2.5 Flash	Google	391B	Proprietary	91.500	75.000	83.300
Llama-4-scout	Meta	17B	Open	92.500	73.500	83.000
Grok 4 Fast	xAI	N/A	Proprietary	96.000	69.500	82.800
DeepSeek Chat V3 (0324)	DeepSeek	685B	Open	93.500	71.500	82.500
ERNIE 4.5 300B A47B	Baidu	300B	Open	93.500	69.500	81.500
DeepSeek V3.1 Terminus	DeepSeek	N/A	Open	91.500	71.000	81.300
Gemma-3n-e4b-it	Google	4B	Open	91.500	71.000	81.300
LFM 2 2.6B	Liquid AI	2.6B	Open	89.000	72.000	80.500
DeepSeek V3.2 Exp	DeepSeek	N/A	Open	91.000	67.000	79.000
LFM2-8B-A1B	Liquid AI	8B	Open	91.000	65.000	78.000
Llama 3.2 3B Instruct	Meta	3B	Open	86.500	63.000	74.800
Qwen 2.5 7B Instruct	Alibaba	7B	Open	88.500	59.500	74.000
Llama 3.1 8B Instruct	Meta	8B	Open	90.000	56.500	73.300
Claude Sonnet 4.5	Anthropic	468B	Proprietary	79.500	55.000	67.300
Granite-4.0-h-micro	IBM	3B	Open	86.000	46.500	66.300
Olmo 2 32B Instruct	AllenAI	32B	Open	84.500	58.000	71.300
Jamba-mini-1.7	AI21	N/A	Open	85.500	54.500	70.000
GLM-4.6	Z.AI	357B	Open	48.000	37.500	42.800
Qwen3-30B-A3B	Alibaba	30B	Open	3.000	5.500	4.300
Nemotron Nano 9B V2	NVIDIA	9B	Open	1.500	2.000	1.800

LLM parameters: `top_p = 1.0`, which is the nucleus sampling parameter (1.0 = all tokens considered); `max_tokens = 16`, which defines the maximum number of tokens generated; `temperature = 0.0`, which controls randomness (0 = deterministic output); and `max_retries = 5`, which specifies the maximum number of retry attempts in case of LLM failure.
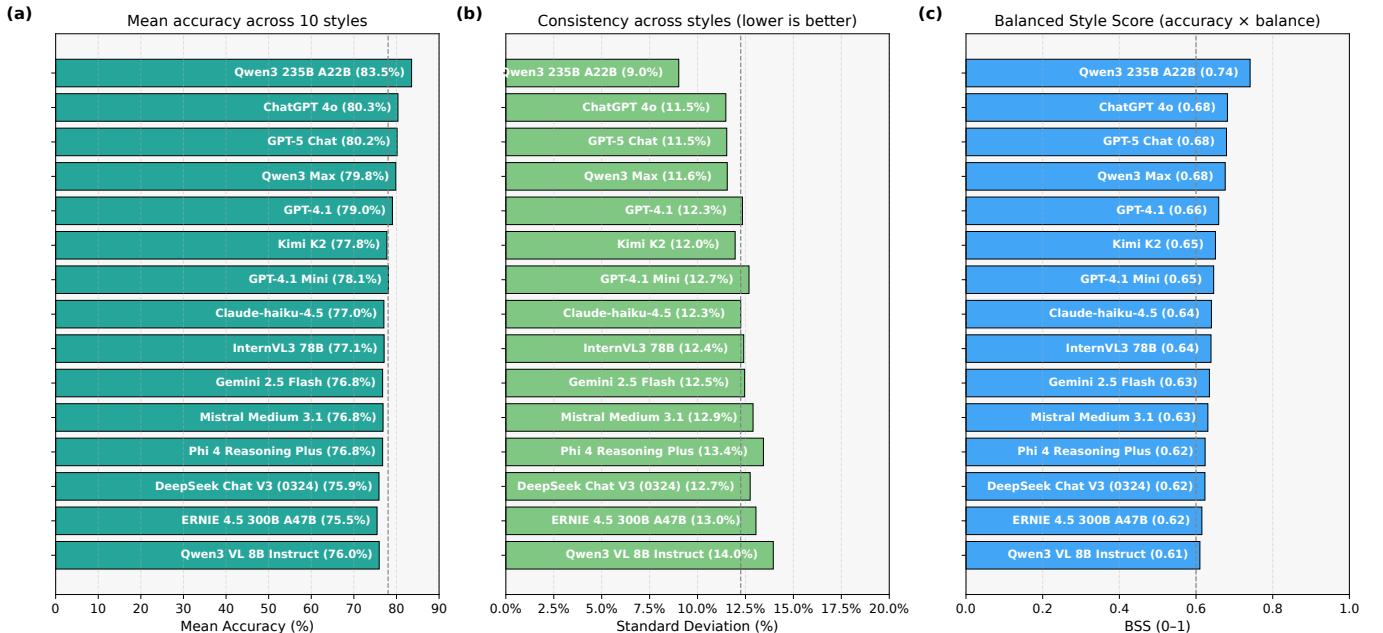

Fig. 5: Top 15 UAVBench_MCQ models ranked by Balanced Style Score (BSS). (a) Mean accuracy across ten reasoning styles, (b) cross-style consistency measured as the standard deviation of accuracies (lower is better; axis in %), and (c) BSS integrating both accuracy and consistency.

84%. This demonstrates that current frontier models maintain strong reasoning capabilities across most UAV mission contexts. Mid-tier systems, including *Qwen3 Max*, *GPT–4.1*, and *Kimi K2*, remain competitive with mean accuracies around 78%–80%, whereas smaller or lightweight models tend to cluster near 70%, confirming the dependence of complex UAV reasoning on model scale and specialization.

Panel (b) highlights cross-style consistency, expressed as the standard deviation of accuracies across the ten reasoning styles. Lower values indicate more balanced reasoning behavior. Here, the leading models exhibit deviation values below 12%, signifying stable performance across domains such as physics, planning, ethics, and system integration. In contrast, several mid-range models achieve similar mean accuracies but display

higher deviations, implying over-specialization in specific reasoning categories and reduced robustness when generalizing across mission types.

Panel (c) integrates these dimensions through the Balanced Style Score (BSS), a composite metric that multiplies geometric mean performance by a variance penalty. The results show that *Qwen3 235B A22B* attains the highest BSS of 0.74, followed by *ChatGPT 4o*, *GPT–5 Chat*, and *Qwen3 Max*, each scoring around 0.68. These findings suggest that models combining high accuracy with low cross-style variance achieve the most reliable overall reasoning behavior. Conversely, some models with respectable accuracy but larger variance suffer lower BSS values, reflecting uneven cognitive performance across domains.

Overall, the triptych visualization emphasizes that balanced reasoning, rather than raw accuracy alone, is crucial for evaluating UAV-oriented cognitive competence. High BSS values correspond to models that not only perform well on average but also maintain consistency across all reasoning categories, a property essential for dependable and safe autonomous decision-making.

V. CONCLUSION

This work introduced UAVBench, a large-scale, open benchmark for evaluating autonomous and agentic AI models in UAV systems. `UAVBench` integrates *50,000 validated UAV flight scenarios* constructed through LLM-driven prompt engineering and multi-stage validation, offering a unified schema that encodes environmental, operational, and safety dimensions of UAV missions. On top of this foundation, we developed UAVBench_MCQ, a structured reasoning benchmark containing *50,000 multiple-choice questions* distributed across ten reasoning styles, enabling interpretable and programmatically gradable evaluation of UAV-specific cognition.

Comprehensive evaluation of thirty-two leading LLMs demonstrated that frontier models achieve near-human performance in perception, policy, and physical reasoning, yet remain challenged by multi-agent coordination, energy management, and ethical trade-offs. These findings underscore both the progress and the limitations of current LLMs when applied to safety-critical aerial autonomy. Future extensions of `UAVBench` will incorporate multimodal sensor data, dynamic simulation rollouts, and temporal reasoning tasks, advancing toward a holistic evaluation framework for embodied, trustworthy, and context-aware UAV intelligence.

REFERENCES

[1] M. A. Ferrag, N. Tihanyi, and M. Debbah, "Reasoning beyond limits: Advances and open problems for llms," *arXiv preprint arXiv:2503.22732*, 2025.

[2] A. Sezgin and A. Boyacı, "Llm-powered uavs: A rag-based approach for safety-critical operations," in *International Conference on Intelligent and Fuzzy Systems*. Springer, 2025, pp. 577–584.

[3] L. Zheng, J. He, S. Y. Chang, Y. Shen, and D. Niyato, "Llm meets the sky: Heuristic multi-agent reinforcement learning for secure heterogeneous uav networks," *arXiv preprint arXiv:2507.17188*, 2025.

[4] Y. Emami, H. Zhou, S. Nabavirazani, and L. Almeida, "Llm-enabled in-context learning for data collection scheduling in uav-assisted sensor networks," *arXiv preprint arXiv:2504.14556*, 2025.

[5] B. Wei, R. Zhang, R. Jiang, M. Peng, and D. Niyato, "Laura: Llm-assisted uav routing for aoi minimization," *arXiv preprint arXiv:2503.23132*, 2025.

[6] Y. Wang, J. Farooq, H. Ghazzai, and G. Setti, "Multi-uav placement for integrated access and backhauling using llm-driven optimization," in *2025 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2025, pp. 1–6.

[7] Y. Li, R. Zhang, Y. Liu, G. Liu, D. Niyato, A. Jamalipour, X. Wang, and D. I. Kim, "Efficient onboard vision-language inference in uav-enabled low-altitude economy networks via llm-enhanced optimization," *arXiv preprint arXiv:2510.10028*, 2025.

[8] Y. Emami, H. Zhou, M. G. Gaitan, K. Li, and L. Almeida, "Frsicl: Llm-enabled in-context learning flight resource allocation for fresh data collection in uav-assisted wildfire monitoring," *arXiv preprint arXiv:2507.10134*, 2025.

[9] Z. Yan, H. Zhou, J. Pei, and H. Tabassum, "Hierarchical and collaborative llm-based control for multi-uav motion and communication in integrated terrestrial and non-terrestrial networks," *arXiv preprint arXiv:2506.06532*, 2025.

[10] Z. Wang, R. Li, S. Li, Y. Xiang, H. Wang, Z. Zhao, and H. Zhang, "Rally: Role-adaptive llm-driven yoked navigation for agentic uav swarms," *arXiv preprint arXiv:2507.01378*, 2025.

[11] Y. Gong, J. Fan, R. Zhang, D. Niyato, Y. Yao, and X. Chang, "Safe and economical uav trajectory planning in low-altitude airspace: A hybrid drl-llm approach with compliance awareness," *arXiv preprint arXiv:2506.08532*, 2025.

[12] K. C. Sekaran, M. Geisler, D. Rößle, A. Mohan, D. Cremers, W. Utschick, M. Botsch, W. Huber, and T. Schön, "Urbaning-v2x: A large-scale multi-vehicle, multi-infrastructure dataset across multiple intersections for cooperative perception," *arXiv preprint arXiv:2510.23478*, 2025.

[13] H. Zheng, N. Gao, D. Cai, S. Jin, and M. Matthaiou, "Uav individual identification via distilled rf fingerprints-based llm in isac networks," *IEEE Wireless Communications Letters*, 2025.

[14] L. Yuan, C. Deng, D.-J. Han, I. Hwang, S. Brunswicker, and C. G. Brinton, "Next-generation llm for uav: From natural language to autonomous flight," *arXiv preprint arXiv:2510.21739*, 2025.

[15] X. Wang, D. Yang, Z. Wang, H. Kwan, J. Chen, W. Wu, H. Li, Y. Liao, and S. Liu, "Towards realistic uav vision-language navigation: Platform, benchmark, and methodology," *arXiv preprint arXiv:2410.07087*, 2024.

[16] F. Yao, Y. Yue, Y. Liu, X. Sun, and K. Fu, "Aeroverse: Uav-agent benchmark suite for simulating, pre-training, finetuning, and evaluating aerospace embodied world models," *arXiv preprint arXiv:2408.15511*, 2024.

[17] M. Guo, M. Wu, J. He, S. Li, H. Li, and C. Tao, "Bedi: A comprehensive benchmark for evaluating embodied agents on uavs," *arXiv preprint arXiv:2505.18229*, 2025.

[18] J. Xiao, Y. Sun, Y. Shao, B. Gan, R. Liu, Y. Wu, W. Gua, and X. Deng, "Uav-on: A benchmark for open-world object goal navigation with aerial agents," *arXiv preprint arXiv:2508.00288*, 2025.

[19] B. Zhao, J. Fang, Z. Dai, Z. Wang, J. Zha, W. Zhang, C. Gao, Y. Wang, J. Cui, X. Chen *et al.*, "Urbanvideo-bench: Benchmarking vision-language models on embodied intelligence with video data in urban spaces," *arXiv preprint arXiv:2503.06157*, 2025.

[20] A. Hamissi and A. Dhraief, "A survey on the unmanned aircraft system traffic management," *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–37, 2023.