ICUAS 2027
UAV COMPETITION

Autonomous Drone Racing

RULEBOOK

Version 1.0

Publication date: 20/10/2026

Autonomous Drone Racing Competition:

Phase 1: simulation qualifiers using the LASER UAV System
Phase 2: in-person autonomous racing with Bitcraze Crazyflie drones

ICUAS '27 UAV Competition is organized by:

http://laser.ci.ufpb.br https://ufpb.br/

Contents

1 Introduction **2**

2 Eligibility criteria and team composition **2**

3 Competition scenario: autonomous obstacle-course racing **2**
3.1 Autonomy requirements 3
3.2 Race run 4

4 Performance benchmarks **4**
4.1 Benchmark 1: Course completion and gate accuracy 4
4.2 Benchmark 2: Race time 4
4.3 Benchmark 3: Safety and robustness 4

5 Competition platform **4**
5.1 Phase 1: LASER UAV System simulation 4
5.2 Phase 2: real Crazyflie platform 4

6 Code and data structures **5**

7 Competition timeline **5**

8 Phase 1: simulation qualifiers **5**
8.1 Submission and evaluation 5
8.2 Qualification and finalist selection 6

9 Phase 2: physical race finals **6**
9.1 Hardware integration and practice 6
9.2 Final race procedure 6
9.3 Safety measures 6

10 Scoring scheme **6**
10.1 Proposed scoring principle 6
10.2 Final scoring and tie-breaks 6

11 Disqualifying and penalized behaviors **7**

12 Evaluation transparency and rule updates **7**

1 Introduction

ICUAS'27 UAV competition is here again! This year's competition will focus on speed! The competition is organized by LASER from the Federal University of Paraíba. The competition will take place in two stages. The ICUAS 2027 UAV Competition is proposed as a two-phase autonomous drone racing challenge. In the qualifiers stage, teams will develop their solution in the ROS-Gazebo environment, and top teams will qualify for the finals stage.

In the autonomous drone racing challenge, teams will design their software enabling a small quadrotor to run a pre-defined course of gates, turns, and obstacles as quickly and reliably as possible without human piloting. The challenge emphasizes perception, trajectory planning, feedback control, robustness, and safe operation.

IMPORTANT:
Rules for the competition and scenario details are subject to change! Make sure to check the official repo for any updates: UAV Autonomous Racing Git and this rulebook regularly. All clarifications and FAQs will be publicly announced.

All communication regarding clarifications on scenario descriptions, rules, and scoring must be via the WhatsApp group discussions: ICUAS'27 WhatsApp Group.

The final scoring scheme, including time limits and penalties, for the simulation phase will be announced after the first evaluation runs.
The scenario and scoring scheme for the finals will be announced by the end of the simulation phase.

2 Eligibility criteria and team composition

The competition is open to any full-time BSc, MSc, and PhD students and others of a similar proficiency level. There is no fee to participate in the qualifier phase of the ICUAS '27 UAV Competition.

There is no limit for the number of team members for the simulation phase. The number of team members to participate in the finals will be limited for in-person attendance, but other registered team members will be allowed to support the on-site team remotely.

Each team must elect a Team Leader (TL) who will be responsible for communication with the organizing committee and referees. Given the dynamic nature of robotics competitions, which often evolve in response to participant feedback, teams will be given time to identify the optimal group to tackle the challenges.

Teams are required to register via the **Google Application Form**.

3 Competition scenario: autonomous obstacle-course racing

A single **Crazyflie-class quadrotor** with a frontal camera must autonomously fly from a designated start area through an ordered sequence of checkpoints or racing gates and finish in a designated zone at a given number of laps. Courses will resemble established drone-racing challenges while being scaled to the dimensions, dynamics, and sensing capabilities of nano-quadrotors, similar to a representation in Fig. 1. The heights of the gates may vary, but the format is always square/rectangle. The final gate sequence and pose are **[TBD]**.

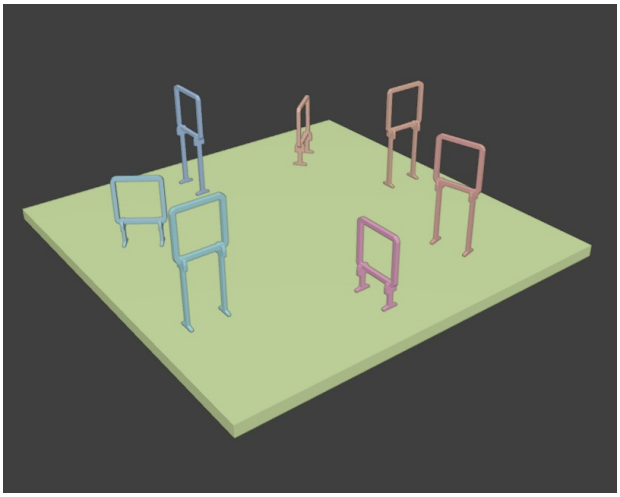

Figure 1: Autonomous race challenge representation with obstacles.

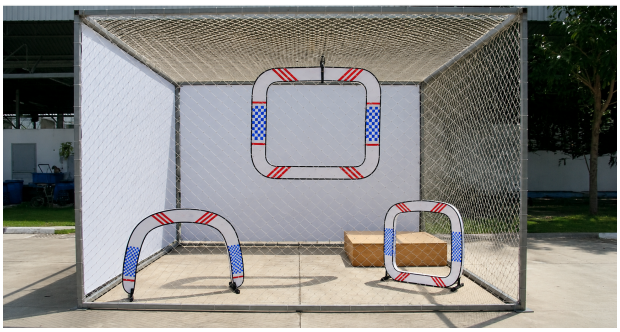

Figure 2: An example of the Arena.

The competition arena measures only 5 m x 5 m x 2 m (Width x Depth x Height) and has the same characteristics as common piloted drone race competitions. The arena has two walls covered with canvas (banner) and two walls with protective netting. The arena also has a protective ceiling. The arena in simulation reflects the real arena. All gates are the same color, and no markers are used for identification. The color of all gates is not provided.

Fig. 2 shows a concept drawing of the arena, with a takeoff/landing base. This base is made of wood and measures 1 m x 1 m x 0.5 m (Width x Depth x Height), and is positioned in a corner of the arena. In addition, the arena has seven structures with internal dimensions of 1 m x 1 m (Width x Height), with a 10 cm-thick border around the 1 m^2 hollow square. These structures are obstacles that the UAV must pass through.

3.1 Autonomy requirements

After an authorized start command, the drone must complete the race without manual steering or operator-selected waypoints. Autonomous takeoff, gate sequencing, obstacle avoidance, and landing/stop behavior are part of the challenge. In simulation, the state estimation will use the simulator's ground truth. During the real-robot race, a motion-capture system will provide state estimation.

Each drone will use a forward-facing camera to detect obstacles it must pass through. This camera will also be provided in simulation.

3.2 Race run

A run begins on the start command (for Phase 1) or the referee's start signal (for Phase 2). Timing starts at a defined start event and ends when the vehicle lands on the landing base after legally crossing the finish checkpoint, after a fixed number of laps.

Number of Laps: For both phases, the total number of laps the UAV must perform is 7.

A valid finish requires passing all mandatory checkpoints in the prescribed order and direction. Missing a gate requires a legal retry of that checkpoint, i.e., turning around the gate and entering through the normal entry direction; skipping it does not count as a complete run. Entering all gates is mandatory and must obey the gate order. The gate order is **[TBD]**.

4 Performance benchmarks

4.1 Benchmark 1: Course completion and gate accuracy

The primary objective is to traverse all required checkpoints in order. The evaluation records successful gate crossings, missed or invalid crossings, and whether the course was completed. A gate counts only when the UAV passes through the gate in the required direction, and in the correct order without touching (hitting) the gate walls.

NOTE: An invalid crossing is when the UAV enters the gate through its exit, i.e., in the inverse direction.

4.2 Benchmark 2: Race time

Among valid complete runs, lower elapsed time is better. In both phases (1 and 2), time is measured by the organizer's evaluation system, not by team-provided timestamps. Time penalties, if used, are added to elapsed time as specified in Section 10.

4.3 Benchmark 3: Safety and robustness

The UAV must remain inside the flight envelope and avoid collisions. A severe collision with the walls or any obstacle will invalidate the run. **Definition:** A severe collision is one that makes the UAV crash, and it is unable to fly.

5 Competition platform

5.1 Phase 1: LASER UAV System simulation

The required software foundation for Phase 1 is the **LASER UAV System**. However, both the simulation environment and the LASER UAV System can be downloaded from a single repository at the **Speed Race Competition GitHub**, in which a reproducible competition setup consisting of a pinned repository revision, installation instructions, race-world assets, Crazyflie-compatible vehicle configuration, launch procedure, supported middleware and simulator versions, a reference example, and the LASER UAV System can be downloaded and installed. **The system runs on a ROS 2, Ubuntu 22.04 system and simulates a Crazyflie UAV.**

5.2 Phase 2: real Crazyflie platform

The real robot race will use organizer-approved Bitcraze Crazyflie hardware with a motion capture system. The exact model, decks, firmware, battery, radio link, localization infrastructure, and permitted compute location (onboard or offboard) are **[TBD]**. Modifying airframes, propellers, or the race infrastructure is prohibited. Finally, the arena (race track), the start/stop, and the rules

will feature a similar environment to that of the simulation. Finally, the map (i.e., the order of obstacles) can change from the simulation environment to the real world challenge.

6 Code and data structures

Teams must submit the code required to build and run their autonomous racing solution, including configuration files, dependencies, and instructions, via the **Simulation Submission Form** that is **[TBD]**as a zip file before the deadline of Phase 1. Solutions should integrate with the competition-provided interfaces (expected to be ROS-based where applicable). Exact topic names, message types, command modes, coordinate frames, launch entry points, and submission packaging will be specified in a technical interface document.

At minimum, the Phase 1 submission must include:

- complete source code for the team's solution, plus a build manifest and dependency list;
- an automated launch entry point and non-interactive execution instructions;
- parameter/configuration files and any trained model weights needed at runtime;
- a brief README describing the algorithm, computational requirements and any required licenses;
- a statement of external services, network access or third-party components (if any).

Organizers must be able to rebuild and test the submission without private credentials, manual control input, or unavailable remote services. Teams retain ownership of their code; the code submission grants organizers permission to execute it for competition evaluation.

7 Competition timeline

All dates are intentionally open and will be fixed by the organizing committee.

Date	Milestone
20/10/2026	Rulebook and registration announcement
20/10/2026	Release of Phase 1 simulation package and sample course
15/01/2027	Registration deadline
28/02/2027	Final Phase 1 code submission deadline
01/04/2027	Phase 1 evaluation and announcement of finalists
14-16/06/2027	Phase 2 hardware briefing and practice window
17/06/2027	Phase 2 in-person race and awards

8 Phase 1: simulation qualifiers

8.1 Submission and evaluation

Each team submits one reproducible software package by the published deadline. Organizers execute the package on standardized hardware using the pinned LASER UAV System configuration. The same scoring protocol, checkpoint definitions, and resource limits apply to all teams. Organizers may run each solution at least three times to assess reliability. The best run will be the one with the lowest value of T_{adj} from (1).

The number of runs, treatment of failed runs, and tie-breaks will be published before the final submission window.

8.2 Qualification and finalist selection

The highest-ranked eligible teams qualify for Phase 2. The number of finalists and any minimum completion or safety thresholds are [**TBD**]. Organizers may request a technical demonstration or clarification to confirm that the submitted software runs autonomously and conforms to the rules.

9 Phase 2: physical race finals

9.1 Hardware integration and practice

Finalists will receive the approved hardware/software specification and a defined integration/practice period. The physical track will follow the announced obstacle categories and safety envelope, but may differ from the simulation qualification course. Any course-inspection time and permitted mapping or calibration procedures will be the same for all teams.

9.2 Final race procedure

Each team receives an equal number of official attempts ([**TBD**]). Before each attempt, referees inspect the drone and verify software readiness. The drone starts from the designated area and executes the course autonomously. A referee may immediately terminate flight if a safety boundary is violated or people/equipment are at risk. Only official attempts count toward final rankings.

9.3 Safety measures

The arena must be separated from spectators by suitable barriers or netting. Safety instructions from the race director take precedence over performance objectives. Any manual intervention beyond a safety stop invalidates the attempt unless the referees declare a technical rerun.

10 Scoring scheme

10.1 Proposed scoring principle

For teams whose drones have completed all runs, the primary ranking metric is the adjusted race time:

$$T_{\text{adj}} = T_{\text{elapsed}} + \sum_{j} n_j \, \Delta t_j, \tag{1}$$

where n_j is the number of penalized incidents of type j and $\Delta t_j = 5$ is the corresponding published time penalty (in seconds).

The ranking starts with the team with the lowest T_{adj} value ranking first and the team with the highest T_{adj} value ranking last.

Partial Runs: Partial runs will not be ranked. Any incomplete run will invalidate the entire try.

Penalties: Hits on obstacles/walls that are not considered a severe collision (i.e., that do not make the UAV crash and unable to fly) and missing entries are considered penalties. For each missing entry or each hit on an obstacle or wall, the team adds $n_j = n_j + 1$. **NOTE:** As stated in 3.2, if the UAV misses any gate during the run and does not perform reentry, the run is considered incomplete, and thus, invalid.

10.2 Final scoring and tie-breaks

Simulation qualifiers and physical finals may use different course layouts and time limits, but these must be announced before the relevant evaluation period. Phase 2, or the real-world final, determines the overall competition winner unless the organizing committee explicitly publishes

a combined-phase ranking. The tie-break order for complete runs is as follows: fewer collision penalties, faster unadjusted time, and an additional tie-break run.

11 Disqualifying and penalized behaviors

An attempt is invalid if the drone leaves the flight boundary, bypasses a mandatory checkpoint without recovery, receives prohibited manual piloting input, or violates an organizer-defined critical safety rule. A severe collision, unrecoverable crash, or emergency stop terminates the attempt; whether the attempt is recorded as invalid or incomplete will be fixed in the scoring appendix. Minor obstacle contact may incur a time penalty if safe continued flight is allowed. Code that cannot be launched in the announced evaluation environment may receive no valid Phase 1 score after any published troubleshooting grace period.

Deliberate interference with other teams, tampering with course infrastructure, exploiting private grading data, malicious code, unauthorized network access, or falsification of results may lead to disqualification from the competition following organizer review. Referee decisions, protests, and appeals procedures can be addressed to the e-mail: tiagopn@ci.ufpb.br.

12 Evaluation transparency and rule updates

Before each scoring phase, organizers will publish the official technical specification, allowed interfaces, resource limits, penalties, and tie-break procedures. Team submissions will be evaluated consistently, with a summary of results made available to participating teams. Clarifications affecting all teams will be posted publicly. Organizers reserve the right to revise this draft before the rules are frozen, and to apply emergency safety changes at any time with prompt notice.