DOI 10.70286/ISU-15.10.2025.012

ENVIRONMENT SELECTION FOR UAV SWARM SIMULATION WITH ONTOLOGY, LLM AND RL

Rohatskyi Ihor
Postgraduate student
Department of System Design
Ivan Franko National University of Lviv, Ukraine

Selecting an effective simulation environment for UAV swarm training and testing is essential to the success of autonomous multi-agent systems. The simulation environment must offer scalability, high-fidelity modelling, flexibility, and safety validation to ensure robust and efficient system performance, etc. Real-time responsiveness and interdisciplinary integration further enhance its ability to support advanced swarm autonomy. A well-designed simulation platform enables the testing of complex scenarios, facilitating the development of UAVs capable of tackling real-world mission challenges. By bridging the gap between simulation and practical deployment, the environment ensures that swarm systems can still operate effectively. Assuming, we are going to model a system whose high-level architecture is shown in Figure 1. Let's define some potential simulators according to the requirements of the proposed solution.

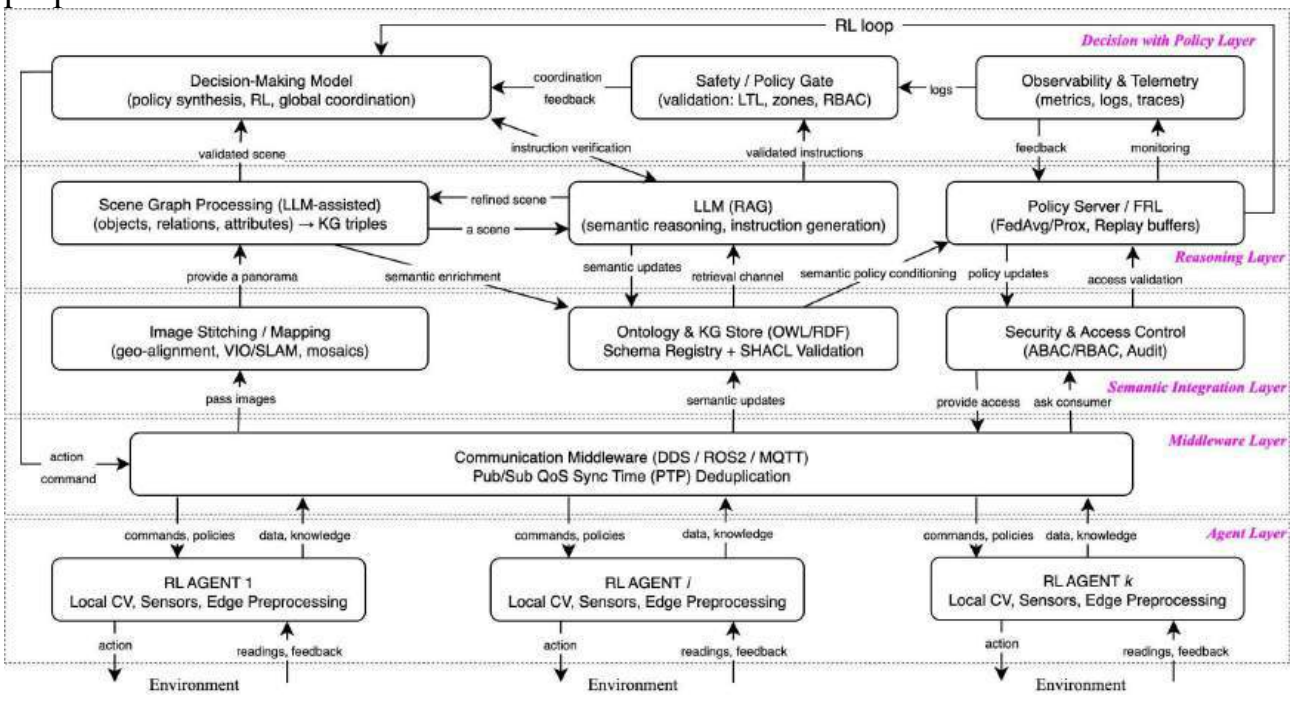

Fig. 1. High-level architecture of an ontology and RL integration into a distributed CV system

Each simulation environment offers unique strengths for specific aspects of UAV swarm research, from photorealistic modelling and fast algorithm prototyping to network interaction and control validation. Most platforms require integration with complementary environments to address their limitations in realism, scalability, or

sensor fidelity. Several potential environments and simulation frameworks can be considered for modelling and experimentation with UAV swarms.

AirSim is an open-source simulator developed by Microsoft, built on the Unreal Engine, supporting both aerial (UAV) and ground (automotive) vehicles. It provides realistic physics, photorealistic rendering, and full Python API access for integrating reinforcement learning and computer vision algorithms. [1]

Webots is a free and open 3D robotics simulator that supports Python controllers and full ROS/ROS2 integration. It allows multi-robot environments with camera and sensor simulation and is often used in educational and industrial robotics research. [2]

FANET-Sim is a simulation framework designed specifically for flying ad hoc networks (FANETs), focusing on the communication and coordination of UAV swarms. It integrates seamlessly with ROS and provides a modular setup for networked UAV simulations. [3]

CrazyChoir is a modular environment for simulating groups of Crazyflie quadrotors under ROS2. It supports multi-agent coordination, mission planning, and trajectory control through Python and ROS-based APIs. [4]

SPACE is a relatively new Python-based simulator for distributed multi-robot systems. It supports plug-ins and extensions for AI algorithms and offers a lightweight environment for multi-agent coordination experiments. [5]

The results of the comparison with the chosen criteria are shown in Figure 2.

Simulator	Swarm Support	Flight Physics / Dynamics	Sensor Simulation	Python API / RL Integration	SITL / Co-simulation	License / Openness	ROS / Middleware Support	Performance / Scalability	Documentation / Community
AirSim	Yes, supports multiple drones	Excellent (based on Unreal Engine physics)	Yes (cameras, sensors, depth and IMU data)	Has Python API (via RPC interface)	Supports SITL through PX4	Open source (MIT)	ROS Bridge supported	Good (depends on scene complexity)	Active community, strong documentation
Webots	Yes, multiple robots and UAVs	Moderate (ODE physics engine)	Yes (cameras, LiDAR, sensors)	Full Python controller support	Can be configured for SITL	Open source (Apache 2.0)	Full ROS integration	Moderate (dozens of agents)	Comprehensive docs, educational examples
FANET-Sim	Designed for UAV swarms and networks	Moderate (focused on network-level behavior)	Limited (sensors can be added manually)	Integrated with ROS, controllable via Python	Co-simulation via ROS possible	Open / academic use	Built-in network simulation model	Moderate (suitable for research)	Academic documentation and examples
CrazyChoir	Limited to Crazyflie drone platform	Medium (simplified quadrotor dynamics)	Yes (when integrated with Webots or Gazebo)	Full ROS2 and Python support	Allows mixed real-sim operation	Open source	Native ROS2 integration	Limited (tens of Crazyflie units)	Well-documented , good ROS2 support
SPACE	Built for multi-agent robotic systems	Moderate (simplified physics / abstract motion model)	Limited sensor simulation (early stage)	Plugin-based, Python-native framework	Co-simulation via custom API possible	Open source (academic, arXiv release)	Built-in lightweight communication environment	Good for algorithmic testing	Emerging project, growing community

Fig. 2. Table of comparison of the most widespread simulation environments

Additionally, there are open-source projects such as UAV Swarm Simulation [6] and Drone Swarm Simulation ("drone-swarms" in GitHub) project featuring real-time 3D visualisations of UAV coordination [7], which can also be considered as lightweight alternatives for rapid prototyping and algorithmic validation.

Among the reviewed simulation environments, AirSim and Webots emerge as the most suitable foundations for implementing the proposed ontology-driven, multi-agent computer vision architecture. AirSim provides highly realistic flight dynamics and a Python-based API that enables the integration of reinforcement learning and computer vision pipelines, making it ideal for the Agent and Middleware layers of the system. Webots, on the other hand, offers superior interoperability with ROS2, simpler multi-

agent configuration, and an open, extensible ontology framework that can host the Semantic Integration and Reasoning layers.

FANET-Sim demonstrates advantages in simulating communication and network behaviour, which supports the policy coordination aspect of the architecture. CrazyChoir is suitable for real-world validation due to its strong ROS2 support and direct interface with physical Crazyflie drones. Meanwhile, SPACE provides a flexible Python-native foundation for testing decision-making and semantic reasoning algorithms without hardware dependencies. Therefore, a hybrid approach combining AirSim for high-fidelity physical simulation and Webots or SPACE for semantic and reasoning integration would most effectively model the proposed distributed decision-making ecosystem.

Overall, the proposed solution (fig. 1), where ontological integration, RAG-LLM reasoning, and agent coordination are important, it is best to start with Webots as a core environment due to reasons:

- it is more easily extensible with Python modules;
- supports online knowledge updates and symbolic interaction between agents;
- allows rapid prototyping of semantic relationships and RL behaviours without GPU overload.

AirSim can be used at a later stage when we need to demonstrate a realistic scenario or validate CV algorithms on photorealistic data.

References

1. Microsoft. (2025). AirSim: Open-source simulator for drones, cars and more [Computer software]. Microsoft Research. https://microsoft.github.io/AirSim/
2. Cyberbotics Ltd. (2024). Webots: Open-source robot simulator [Computer software]. Cyberbotics. https://www.cyberbotics.com/.
3. Rathore, M. M., Ahmad, A., & Raza, M. (2023). FANET-Sim: A simulation framework for UAV ad hoc networks. Proceedings of the 8th ACM Workshop on UAV Networks and Systems. ACM Digital Library. https://dl.acm.org/doi/10.1145/3732365.3732396.
4. Pugliese, M., D'Andrea, R., & Caporale, M. (2023). CrazyChoir: Multi-drone coordination with ROS2 and Python [Preprint]. arXiv. https://arxiv.org/abs/2302.00716.
5. Serrano, F., Liu, X., & Aguado, A. (2024). SPACE: A Python-based simulator for distributed multi-agent coordination. arXiv preprint arXiv:2409.04230. https://arxiv.org/abs/2409.04230.
6. Prata, M. (2024). UAV Swarm Simulator: Multi-agent drone simulation environment [Computer software]. GitHub. https://github.com/matteoprata/UAV-Swarm-Simulator.
7. Jerome, J. (2024). Drone-swarms: Real-time 3D simulation for UAV coordination [Computer software]. GitHub. https://github.com/jeanjerome/drone-swarms.