M. Höser, K.-D. Büchter, "Simulation-based Testing of Service Drones in U-Space Environments", VEHICULAR 2022, 22-26 May, Venice, Italy, 2022.

Simulation-based Testing of Service Drones in U-Space Environments

Moritz Höser
Bauhaus Luftfahrt e. V.
82024 Taufkirchen, Germany
moritz.hoeser@bauhaus-luftfahrt.net

Kai-Daniel Büchter
Bauhaus Luftfahrt e. V.
82024 Taufkirchen, Germany
kai-daniel.buechter@bauhaus-luftfahrt.net

***Abstract*—With the introduction of U-Space, upcoming drone flight controllers are required to combine safety, autonomy and human interaction with evaluable regulatory compliance. In these regards, testing drones in virtual environments is a promising approach to enhance the development of drone controllers. With virtual drone test environments being actively developed, questions concerning general validity arise including a) how drone-testing scenarios should be described, set up and executed in a transferable manner and b) what conclusion developers and regulators can draw from such experiments. This work discusses potential benefits and interim results for creating, executing and later evaluating scenarios for virtual drone testing environments. The work includes early stages on the research on scenario content and simulation execution concept.**

***Index Terms*—drones, virtual testing, simulation, autonomy, U-Space, compliance, scenario.**

I. Introduction

The gradual introduction of the U-Space framework for drone operation is facilitating drone-based services and future airspace applications. Upcoming unmanned service drones are expected to operate comprehensibly and in an automated way while adhering to strict safety standards; their autopilot controllers will be part of a mobile, cyber-physical system that operates in an increasingly complex environment. This complexity poses major challenges to developers and regulators alike and may also raise concerns among the affected public.

Scenario-based testing and simulations can cover some of the requirements to evaluate system behavior in complex situations. The scenarios thereby have to fulfill requirements on reproducibility, interpretability and traceability.

The automotive sector is pushing ahead with scenario-based testing, among other things, in the pursuit of autonomous driving. However, these works are often only of limited use for drones. Drones in the Very Low-Level (VLL) airspace are confronted with the practical challenges in safely providing drone services on the one hand, and on the other hand meeting regulatory requirements of Unmanned Traffic Management (UTM).

This work presents considerations and first results on describing and executing simulation scenarios for service drones. Section II states the underlying research questions and prior research. Section III describes the results as presented in the work. Section IV ends with conclusion and future work.

II. Related work

A. Research question

Considering the objectives concerning UTM and drones, the authors define the following main research question:

RQ: What content should drone testing scenarios for UTM cover and how can it be described in a tool-independent manner? The RQ targets especially reproducibility, traceability and interpretability of test configurations, supporting the generation of valuable results through formalization of and automation in virtual testing conditions.

B. Related research

Related research is only briefly mentioned here due to limited space but can draw from a wide range of publications and ongoing research. Basic definitions on the subject of scenarios are summarized in [2], among others. Observations on the development cycle of scenarios and validation for automated systems and vehicles can be found in [1] and [3]. OpenScenario [5] is one example for an ongoing, domain-specific approach to establish a scenario description for the automotive industry. Several publications originating from [4] and [6] apply similar concepts to the aviation space.

III. Work and results

A. Challenges and requirements

Service drone platforms face a number of challenges: Firstly, the development of controllers for autonomous drones is undergoing active development, including the need to test real-time sensor data processing and algorithms for autonomous flight. Second, the concerns of future stakeholders should already be taken into account: in addition to developers and operators, these include UTM regulators, but also a concerned public, which will be indirectly confronted with the drones. Third, during and in addition to the actual tasks, drones will also face unexpected challenges that should be simulated to some extent in order to be able to master later situations without interventions and to make tests more convincing for the interested public.

B. Approach

The steps presented here center on the tasks from scenario content modeling up to scenario execution in the simulator.

 ISBN: 978-1-61208-974-4

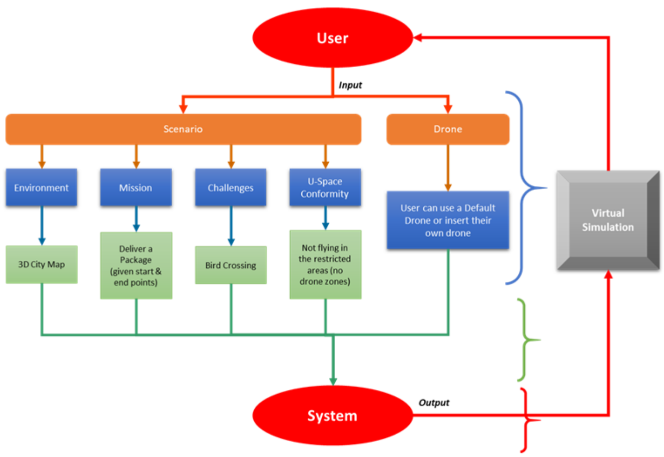


Fig. 1: Scenario elements (in blue) with examples (in green), in the context of a simulation system

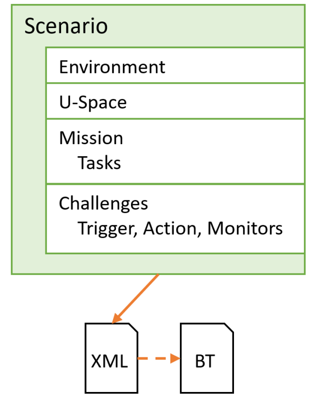


Fig. 2: The scenario elements are described in a scenario file format (XML and DSL). Next, the content is translated into a Behavior Tree (BT) for executing the scenario.

Fundamental for the modeling process are the considerations about the actual purpose of the scenarios in its application domain, which is reproduced among others in [1, p. 29]. Making use of scenario-based testing therefore starts with the *requirements* for the scenario, continues with its *implementation* and ultimately ends in the monitoring related tasks *verification and validation*, which provide the actual insight.

The current state of this research focuses on scenario modeling and implementation options, as it sets the stage for a meaningful analysis of monitoring results later on. In the best case, the work should allow other users and researchers to create own scenarios and achieve valuable and comprehensible simulation results for evaluation and development.

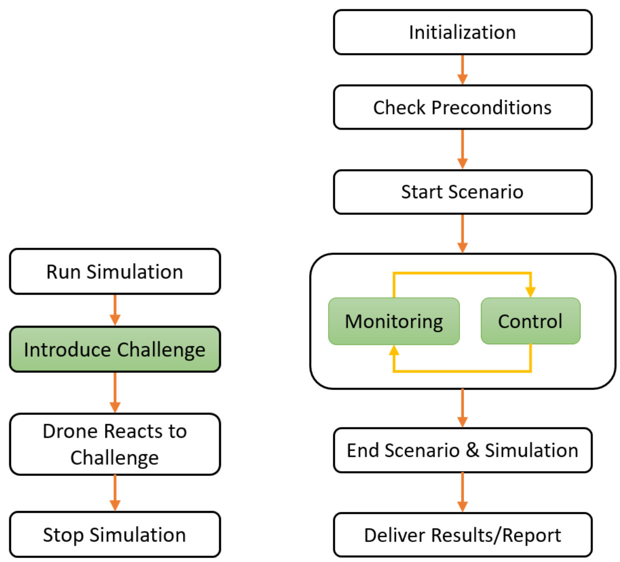
C. Results

The figures describe the early results of discussions and research on scenario contents, format and execution. In principle, unmanned drones with a specific task (mission) in an urban or rural area are to be considered. These drones operate in the VLL and must comply with a number of UTM regulations. In Europe, these rules are defined by the U-Space framework, which defines different airspace types and service levels.

The discussed requirements – summarized by the goal of testing drone capability and conformity with the prospect on verification and validation – lead to the *basic elements* of a testing scenario as depicted in Figure 1 and Figure 2.

The basic building blocks, in addition to the environment, are the main elements *U-Space, missions and challenges*. In the upcoming work, a data model and open specification for the format and content of these building blocks will be developed. In this process, findings from other groups can be drawn upon, including those from the automotive, aviation and simulation fields.

In particular, the so-called "Challenges" are difficult to integrate in the scenario model, since both variability and reproducibility are to be achieved at the same time. Challenges are intended to create an unexpected problem for the drone during or before mission execution. In addition to the laid out general procedure for challenges (Figure 3a), a classification and collection of challenges for drones is in development.



(a) Challenges can be introduced in scenarios and pose an unexpected event to the drone.

(b) The lifeline of a scenario during its simulation. At the core of the simulator, an execution engine monitors and controls the active scenario elements.

Fig. 3: Concept for challenges and scenario control

The execution of the scenarios poses a second challenge in simulation-based testing. A number of existing simulation environments from research and open source communities provide a robust foundation for simulation. In the context of this research towards drone algorithm development, the simulator Gazebo plus the Robot Operating System (ROS) is selected as a first development platform.

From an architectural viewpoint, scenario execution is best separated into a *scenario runner* component. Some elements of the scenario, such as mission execution, can and should

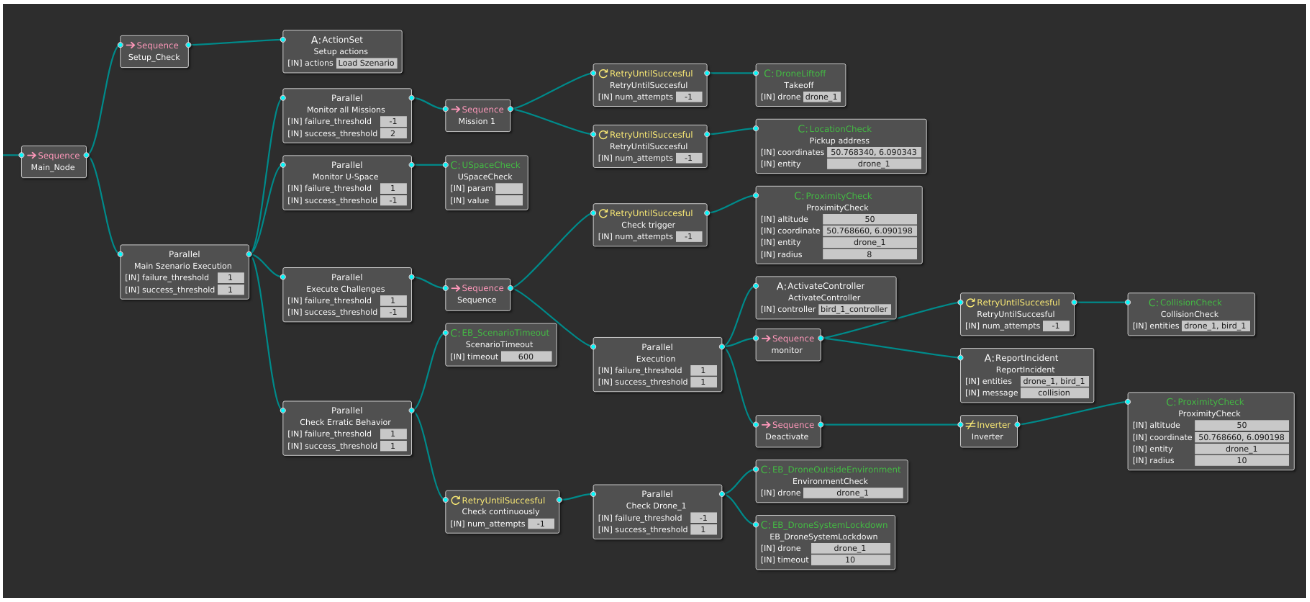


Fig. 4: Outlook on a behavior tree implementation to control the described scenario elements (shortened). Each branch on the 3rd level controls or monitors one category of scenario elements, while the last one checks for erratic behaviour (termination).

be observed passively, while others such as the challenges need an active intervention. This calls for a scenario interpreter component at the runner's core, similar to the concept depicted in Figure 3b. The particular attention to comprehensibility, but also adaptability and finally portability to other simulation environments, makes it preferable to handle the core interpreter independent from the simulator platform. Figure 4 shows work in progress to describe this interpreter by means of a behavior tree. Behavior trees, with their design for decisions and events plus their availability in simulators, have the potential to guide scenarios in a platform-independent way, but more research is needed in this area. The excerpt shows some part of the main tree to observe the scenario elements. Similar considerations have to be made for the monitoring component, which forms the basis for reporting and evaluation.

The upcoming work will focus on a evaluable scenario format description, a scenario editor, and further aspects of the scenario runner integration and monitoring component.

IV. Conclusion and Future Work

The presented work in progress investigates how simulation scenarios for virtual testing of service drones can be described and implemented. The research can build on similar work from the automotive sector and aviation scenario studies, but for drone services, there are specific requirements in scenario design and simulation. The main focus here lies on demonstrating drone conformity and capability, which is addressed with a direct integration of U-Space and service missions. Early results on a scenario description model and methods for scenario execution are presented with figures in the poster.

Future work will include a definite scenario format proposal. The next step is to support simulator execution with translation rules for the behavior tree. Tool support, like a scenario editor, will be necessary to establish a repository of scenarios and challenges to choose from. The plan is then to use the developed scenarios to investigate verification and validation methods for drones in an UTM setting.

Acknowledgment

This work is supported by ECSEL Joint Undertaking (JU) through the Project ADACORSA under grant agreement No 876019. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Germany, Netherlands, Austria, Romania, France, Sweden, Cyprus, Greece, Lithuania, Portugal, Italy, Finland, Turkey.

We thank our colleague Julia Schaumeier for her early contributions.

References

[1] S. Kalisvaart, Z. Slavik, and O. Op den Camp, "Using Scenarios in Safety Validation of Automated Systems," in Validation and verification of automated systems: Springer, 2020, pp. 27–44.

[2] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving," in 2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC): Spain, 2015, pp. 982–988.

[3] T. Menzel, G. Bagschik, and M. Maurer, "Scenarios for Development, Test and Validation of Automated Vehicles," in 2018 IEEE Intelligent Vehicles Symposium (IV): Changshu, 2018, pp. 1821–1827.

[4] S. Jafer, B. Chhaya, U. Durak, and T. Gerlach, "Formal Scenario Definition Language for Aviation: Aircraft Landing Case Study," in AIAA Modeling and Simulation Technologies Conference, Washington, D.C., 2016, pp. 2016–3521.

[5] ASAM e. V., OpenSCENARIO. [Online]. Available: https://www.asam.net/standards/detail/openscenario/ (accessed: April. 4 2022).

[6] J. A. Millan-Romera, J. J. Acevedo, A. R. Castano, H. Perez-Leon, C. Capitan, and A. Ollero, "A UTM simulator based on ROS and Gazebo," in The 2019 International Workshop on Research, Education and Development on Unmanned Aerial Systems (RED-UAS 2019): Cranfield, United Kingdom, 2019, pp. 132–141.