FADS: Framework for Autonomous Drones Safety: Certification, Benchmarks and Enforcement

Yash V. Pant, Max Z. Li, Rhudii A. Quaye,
Houssam Abbas, Rahul Mangharam, Megan S. Ryerson*

Abstract. This work aims to take a step towards certified autonomous Unmanned Aircraft Systems (UAS) operations in urban environments. To achieve this, we leverage a formal mathematical representation of a) The Federal Aviation Authority (FAA) prescribed safety requirements and b) the mission specific operational requirements. Through this representation, we develop an algorithm to generate trajectories for multi-rotor UASs that satisfy both the safety and operational requirements in a robust manner. In addition, we also show that contingencies (e.g. in the case of loss of communication) can be also be encoded within the same mathematical framework, and our algorithm allows for safe operation of UASs in an urban environment. The process of mathematically formalizing the FAA requirements and operational requirements will also lead to a database against which algorithms for autonomous requirements can be verified.

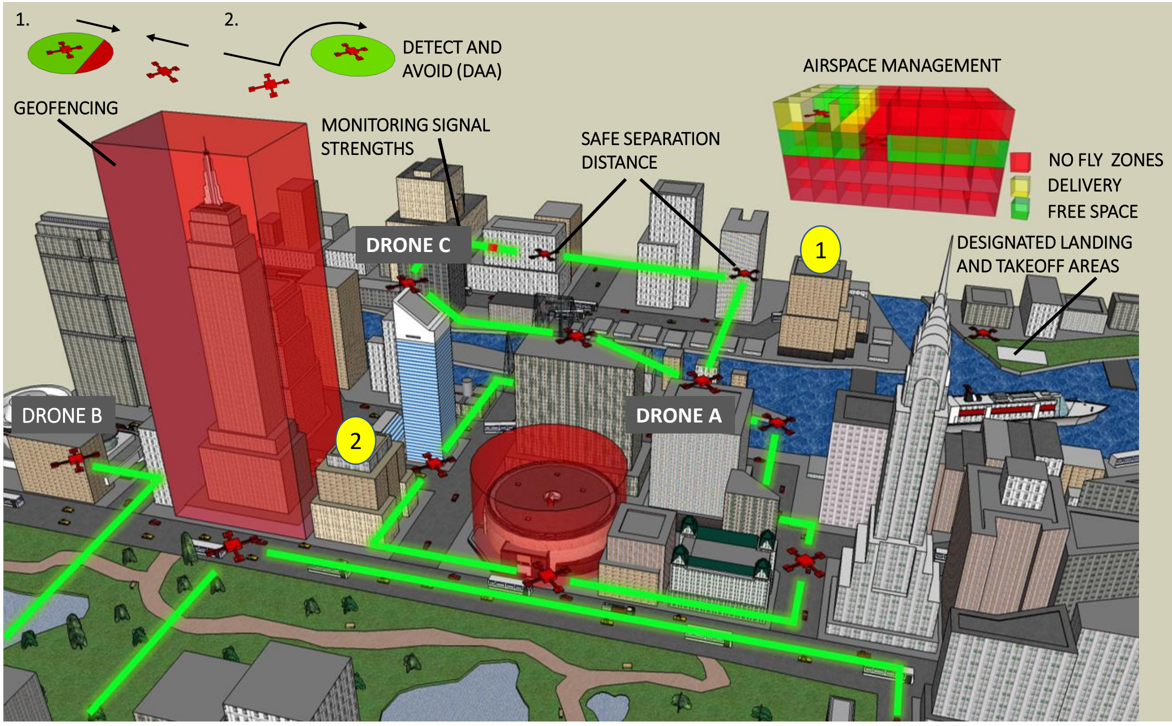

Figure 1. *Multiple autonomous drone missions in an urban environment. (Figure adapted from https://phys.org/news/2016-12-trafficsolutions-drones-singapore-airspace.html)*

1. Introduction.

FAA does not allow autonomous UAS operation [1, 2], except via case by case exemptions. However, increasing autonomy will inevitably lead to multiple fleets of

*Department of Electrical and Systems Engineering, University of Pennsylvania (habbas,rahul@seas.upenn.edu).

autonomous UASs sharing the same airspace (Fig. 1). In such a scenario, new techniques are needed to plan flights in a manner that provides safety as well as performance guarantees.

This work aims to show that existing FAA requirements (excluding ones on operator placements and line-of-sight), as well as contingency plans can be formally encoded in a mathematically sound manner by using Signal Temporal Logic (STL) [3]. By collecting safety and operational requirements from existing FAA guidelines [1, 4], as well as mission specific requirements for a variety of operational scenarios (e.g. surveillance, infrastructure monitoring etc.), and then representing them in STL, we aim to build a database of requirement benchmarks against which algorithms for autonomous operations can be checked. These requirements can consist of a combination of the following objectives:

1. Spatial objectives, e.g. geo-fenced no fly zones to avoid, or delivery zones to reach,
2. Temporal objectives, e.g. Hovering in a particular area for a predefined time window to monitor wireless signal strengths,
3. Reactive objectives, e.g. in case of a loss of link, UAS flies to a holding area or crash lands in a designated spot.

Additionally, we also present a technique [3] that uses these requirements to synthesize trajectories that satisfy the given requirements robustly.

Through tagging on additional requirements and/or relaxing existing ones, we can show that autonomous UAS operations are "safe" and "achievable" in urban and non-urban airspaces.

2. Towards bench-marking of requirements. Requirements for UAS operations can be broadly divided into two categories:

- **Safety requirements:** FAA rules, e.g. 14 CFR 107 [2] for UASs weighing less than 55lbs, and the FAA Concept of Operations (ConOps v2.0) [4] for heavier UASs suggest speed limits, altitude ceilings (and floors), distance to buildings, and distance to other UASs as non-negotiable safety requirements. Contingencies for cases such as loss of communication to UAS, intruder in airspace/deviation of other agents from flight plan and battery failure also fall under the safety umbrella.
- **Operational requirements:** While FAA requirements only cover safety, successful completion of a mission involves satisfying other operational, or mission based requirements. An example could be hover over a particular area to capture images at certain periods of time, or visit a set of regions in a sequential manner for surveillance.

While the safety requirements are of primary concern, successfully completing a mission would require satisfying both safety and mission requirements. We aim to capture all safety requirements specified in [1, 2], as well as mission specific requirements for a variety of conceivable missions. These will be compiled into a database of mission spe-

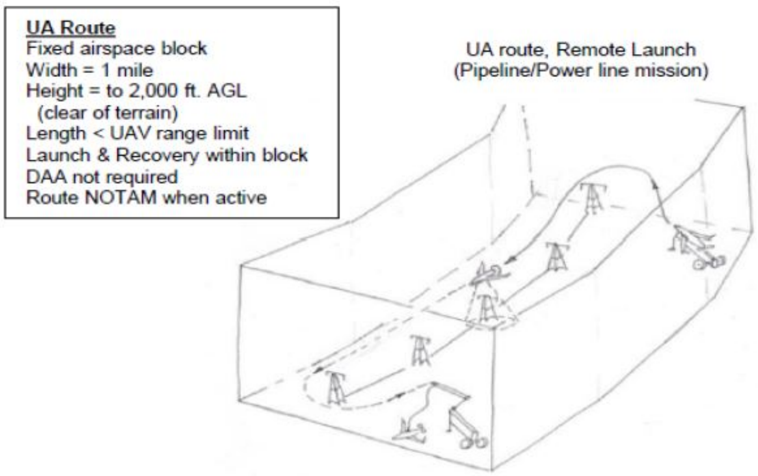

Figure 2. *Pipeline/powerline surveillance example, figure from [4].*

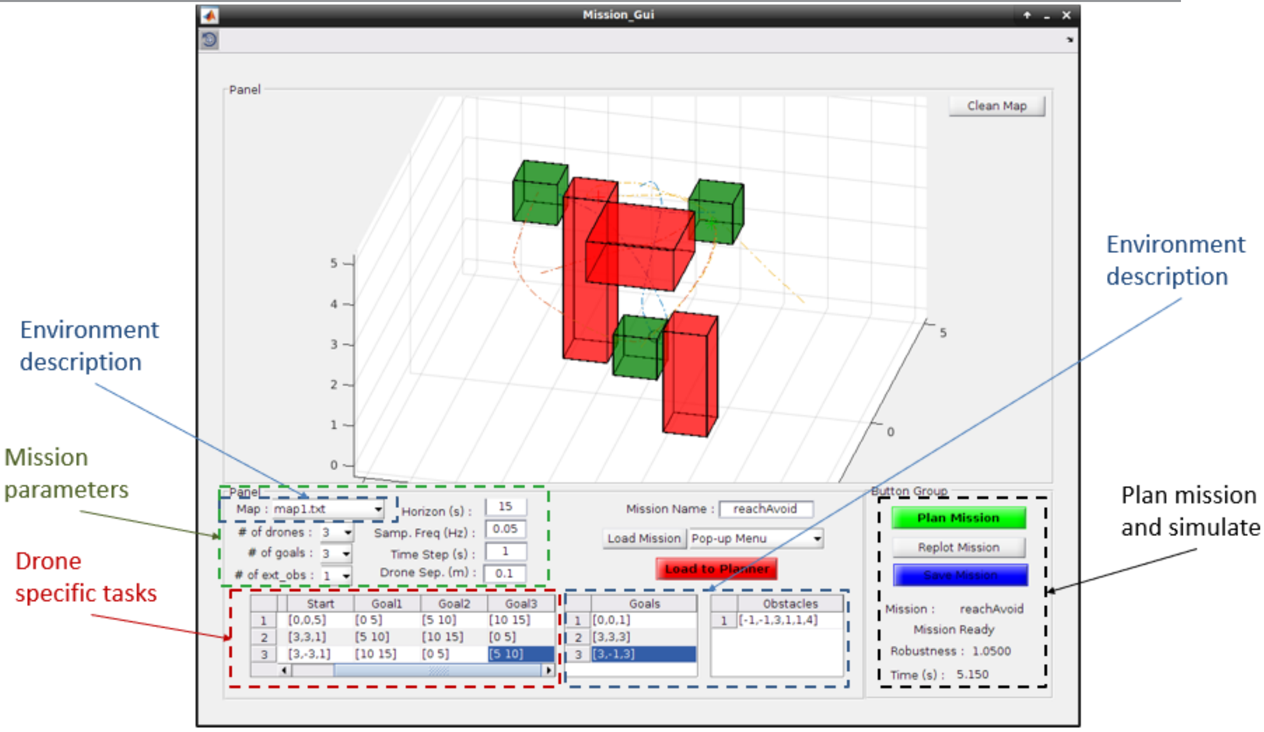

Figure 3. *GUI for mission specification, translation to STL, and trajectory generation.*

cific benchmarks, and translated into STL. As an example, we take a pipeline inspection mission in an non-urban environment from [1] and formalize the requirements:

Example: Figure 2 shows the pipeline inspection example from [4]. The safety requirements for the mission are: a) Operate within a block of width 1 mile (1600m) and length less than the UAS range (also say 1 mile). b) The UAS is not allowed to fly over 2000 feet (610m). c) The UAS has to take-off and land only from designated sites. d) An implicit time limit on how long the UAS can operate comes from the flight time for a fully-charged battery.

Mission requirements are as follows: a) Fly over each of the pump jacks in the drilling area in a sequential manner to collect pictures, and b) land in the designated area before battery runs out. These requirements can be encoded using the *always* and *eventually* operators in STL as outlined in [3].

3. Proposed methodology. To enable safe/certified UAS operations, our approach consists of the following steps: 1) Collect list of safety and operational requirements, 2) compose the relevant requirements into a STL mission formula, 3) Use the technique of [3] to generate trajectories for UASs that satisfy the mission with some guarantees. Figure 3 shows a graphical front end to specify a mission, translate it to STL, and then use the methods of [3] to generadrone platforms for research !gte trajectories for the UASs.

Example: The video in http://bit.ly/multi8mission shows a simulation for a case study in [3] that looks at a scenario where two fleets of 4 drones each share a common airspace and perform separate missions, a) A repeated surveillance mission (of regions in yellow) and

b) A package delivery (to region in blue) and return to base (region in green) mission, while also ensuring avoidance of the no-fly zones (in red) and minimum separation of drones. The video shows trajectories generated by our method that satisfy both the safety and operational requirements of this example.

Contingencies for cases like loss of communication, or low battery, can also be encoded in STL and incorporated into our method. More details are in [3].

Note, that in our case a guarantee would be that if UASs follow their computed trajectories within tubes of some radius 'r', then the specification is always satisfied, and in a continuous time paradigm. We can also interpret this robustness ('r') value as a guarantee on the (lower bound of) distance from infrastructure etc.

Through this formalization of requirements and guarantees on satisfying them, we can achieve a "certification" for any operator that wants to operate a fleet of UASs in an urban environment. Additionally, these formalized requirements can also be used for a) monitoring mission execution at run-time, b) checking if other algorithms result in trajectories that satisfy the specification, and c) for finding corner cases for existing planning algorithms where they fail to satisfy the given mission (through falsification [5]).

4. Discussion. On going work is on showing proof of concept through various example missions (along with contingencies for each case) for multi-rotor UASs, some of which can be found in [3] for missions involving one or multiple quadrotors. Additionally, a Graphical User Interface (Figure 3) is being developed to allow missions to be specified in a way that is directly translated to STL. This is followed by generating trajectories for UASs that satisfy the underlying specification.

REFERENCES

[1] Federal Aviation Administration, "Integration of civil unmanned aircraft systems (uas) in the national airspace system (nas) roadmap." https://www.faa.gov/uas/media/UAS_Roadmap_2013.pdf, 2013. Retrieved May 20, 2018.

[2] Federal Aviation Administration, "Faa suas part 107: The small uas rule." https://www.faa.gov/uas/media/faa-uas-part107-flyer.pdf, 2018. Retrieved May 20, 2018.

[3] Y. V. Pant, H. Abbas, R. A. Quaye, and R. Mangharam, "Fly-by-logic: Control of multi-drone fleets with temporal logic objectives," in *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*, ICCPS '18, (Piscataway, NJ, USA), pp. 186–197, IEEE Press, 2018.

[4] Federal Aviation Administration, "Faa's approach to unmanned aircraft systems (uas) concept maturation, validation & requirements development." https://www.faa.gov/about/office_org/headquarters_offices/ang/offices/tc/library/v&vsummit/v&vsummit2015/presentations/UAS%20Concept%20Maturation-Validation%20and%20Requirements%20Development_Sherri%20Magyarits.pdf, 2015. Retrieved May 20, 2018.

[5] T. Dreossi, T. Dang, A. Donze, J. Kapinski, X. Jin, and J. V. Deshmukh, "A trajectory splicing approach to concretizing counterexamples for hybrid systems," in *NASA Symposium on Formal Methods*, 2015.