Autonomous Air Traffic Control: The Fly-by-Logic Approach

Danyang Li   Advisor: Rahul Mangharam

1 Overview

The problem of safe planning and control for multi- drone systems across a variety of missions is of critical importance. This work focuses on encoding these requirements and missions to a mathematically framework. Based on the mathematical expression, our controller generate trajectories for the quadrotors in continuous-time. This approach avoids the oversimplifying abstractions found in many planning methods, allowing us to handle complex spatial, temporal and reactive requirements.

2 Problem: Autonomous Air Traffic Control

Dealing with a complex set of requirements to manage the airspace traffic.

- Minimum separation of aircrafts
- Velocity and acceleration limit
- Timely reaching goal
- Avoid No-Fly Zone

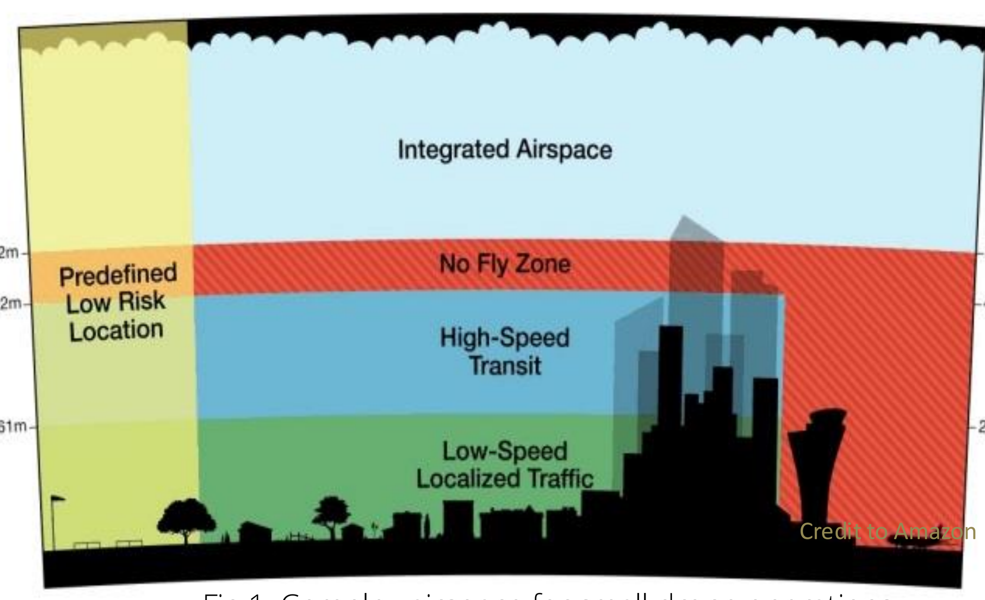


Fig 1. Complex airspace for small drone operations

Challenge 1: How do we encode such rules mathematically?

Challenge 2: How do we use this formulation to control real world dynamical systems?

3 The Control Architecture

Given various missions of a drone fleet, we could specify missions into parameters(time limits, velocity limits, goal zone, no-fly zone). The high-level controller generates a sequence of waypoints, position & attitude control. These parameters are sent out to the drones, and through a hierarchical control on-board control architecture, each drone would follow the trajectories perfectly with the continuous time behavior.

Application: Real-time control of multiple quadrotors for various missions

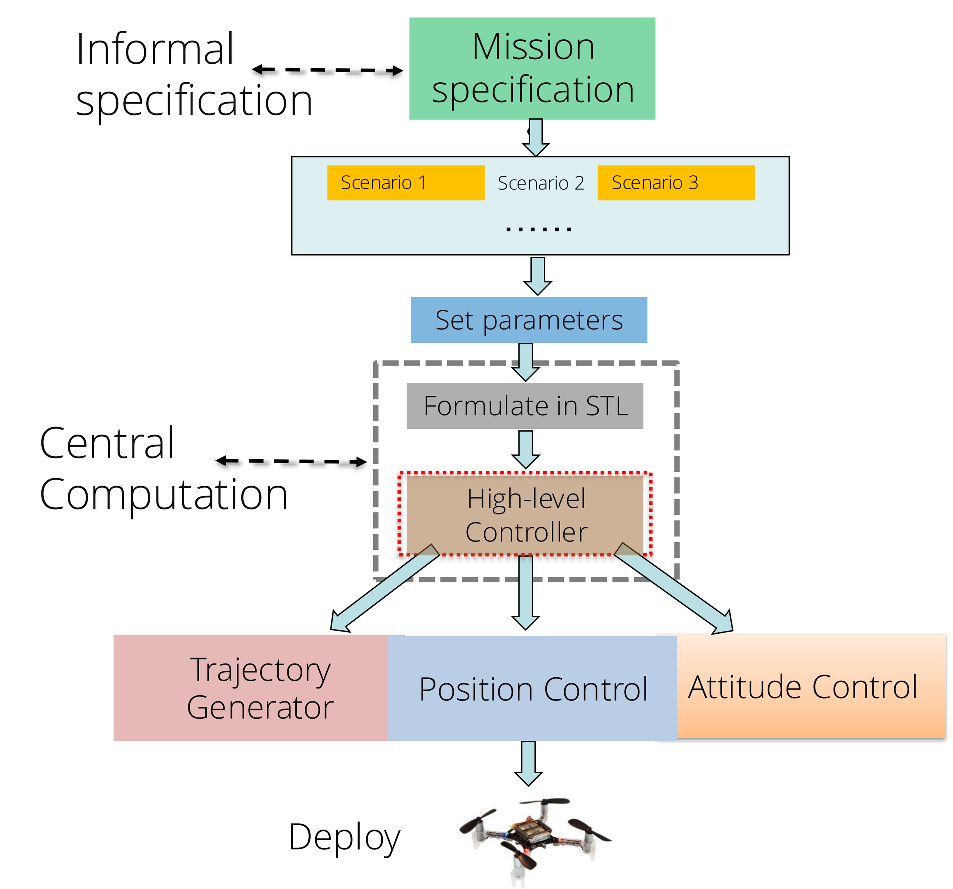


Fig 2. Control architecture for mission specification and control of autonomous systems

4 Specifications in Signal Temporal Logic(STL)

STL is a logic that allows the succinct and unambiguous specification of a wide variety of desired system behaviors over time.
An example of a 4 UAS reach-avoid mission

Temporal range   Logical AND   Logical negation

$$\phi = (\wedge_{d=1}^{4} F_{[0,10]}(p_d \in \text{Goal})) \wedge (\wedge_{d=1}^{4} G_{[0,10]} \neg \underbrace{(p_d \in \text{Unsafe})}_{\text{Atomic proposition}})$$
$$\wedge (\wedge_{d \neq d'} G_{[0,10]} ||p_d - p_{d'}|| \geq \delta)$$

- Always (G): Avoid Unsafe set and each other
- Eventually (F): Reach Goal set within 10 seconds
- *Mission level specification:* Combine above goals
- Specification horizon: Maximum period of time to finish goals

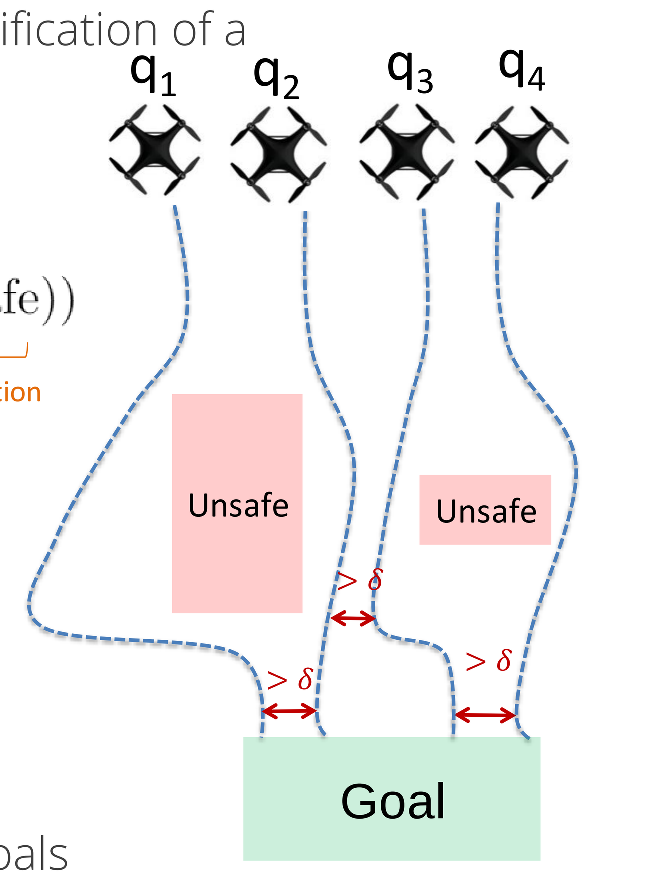


Fig 3. Four UAVs reach-avoid mission sketch

5 Control with Smooth Robustness

The goal is to find the maximum robustness of STL formula ϕ

$$P: \max_{\mathbf{u} \in U^{N-1}} \rho_{\varphi}(\mathbf{x})$$
$$\text{s.t. } x_{k+1} = f(x_k, u_k), \forall k = 0, \ldots, N-1$$
$$x_k \in X, u_k \in U \; \forall k = 0, \ldots, N$$
$$\rho_{\varphi}(\mathbf{x}) \geq \epsilon$$

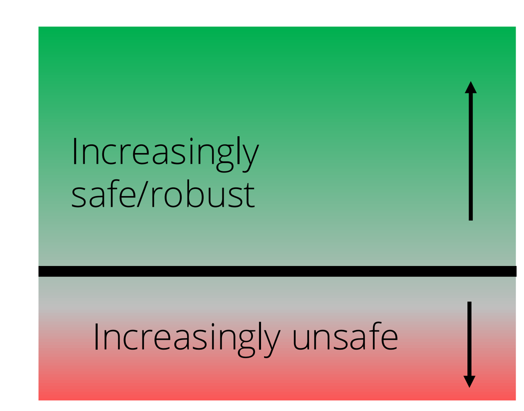


Fig 4. Robustness value corresponding to safety

Sign corresponds to satisfaction of specification

$$\varphi = \begin{cases} True, & \text{if } \rho_{\varphi} \geq 0 \\ False, & \text{otherwise} \end{cases}$$

Cannot maximize robustness as function is non-smooth due to signed distance and discrete max/min (fig. 4).

Smooth robustness $\widetilde{max}(\mathbf{x}) = \frac{1}{C} \log \sum_i \exp(C x_i)$

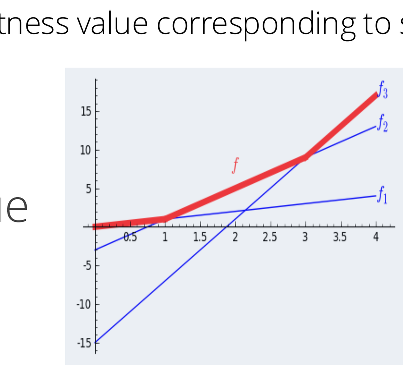


Fig 4. non-smooth maximum function

6 Planning in Sequential Method

Motivation: All at once planning method is not suitable for commercial application because it needs to share all mission information.

Possible Solution: Apply sequential planning method. Only one mission information needs to be shared to avoid collision.

Example of a 4 UAS reach-avoid mission formulating in *Sequential Method*:

$$\phi_1 = (F_{[0,10]}(p_1 \in \text{Goal})) \wedge (G_{[0,10]} \neg (p_1 \in \text{Unsafe}))$$
$$\phi_2 = (F_{[0,10]}(p_2 \in \text{Goal})) \wedge (G_{[0,10]} \neg (p_2 \in \text{Unsafe}))$$
$$\wedge (G_{[0,10]} ||p_2 - p_1|| \geq \delta)$$
$$\phi_3 = (F_{[0,10]}(p_3 \in \text{Goal})) \wedge (G_{[0,10]} \neg (p_3 \in \text{Unsafe}))$$
$$\wedge (\wedge_{d=1}^{2} G_{[0,10]} ||p_3 - p_d|| \geq \delta)$$
$$\phi_4 = (F_{[0,10]}(p_4 \in \text{Goal})) \wedge (G_{[0,10]} \neg (p_4 \in \text{Unsafe}))$$
$$\wedge (\wedge_{d=1}^{3} G_{[0,10]} ||p_4 - p_d|| \geq \delta)$$

7 Sequential Method vs. Centralized Method

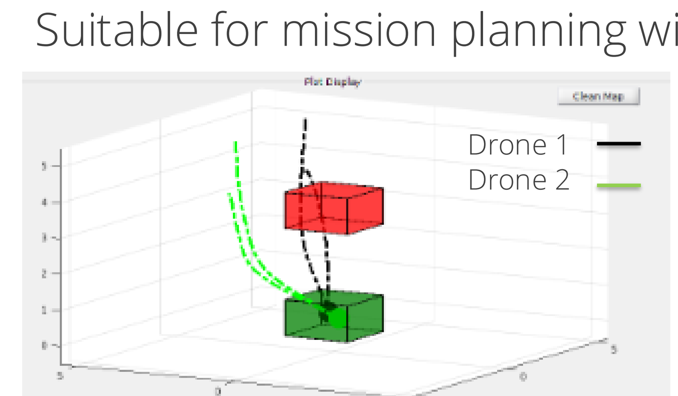
- Robustness slightly decrease

Number of drones	Robustness	
	Centralized	Sequential
2	0.4999	0.4475
6	0.5	0.4907
12	0	0

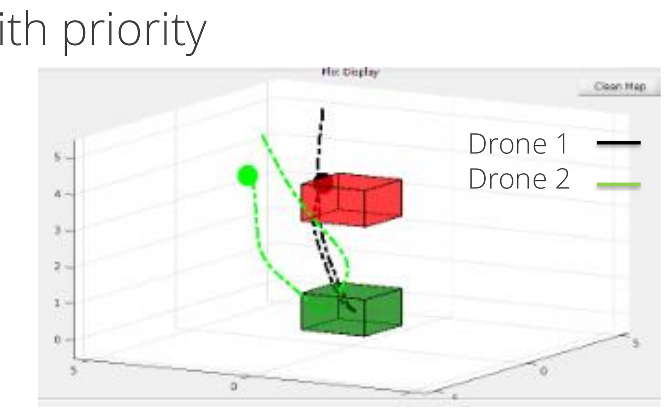
- Planning time dramatically decrease

Number of drones	Planning time(s)	
	Centralized	Sequential
2	16.396	9.46
6	100.778	60.623
12	223.041	86.063

- Suitable for mission planning with priority



Centralized



Sequential

The first drone in sequence always have the optimum trajectory.

- Minimum Robustness Guaranteed

Robustness goes to negative *at the same time* for both Sequential and Centralized Method

8 Experiment

Now implementing both indoor and outdoor experiments by assigning different tasks to UAVs.

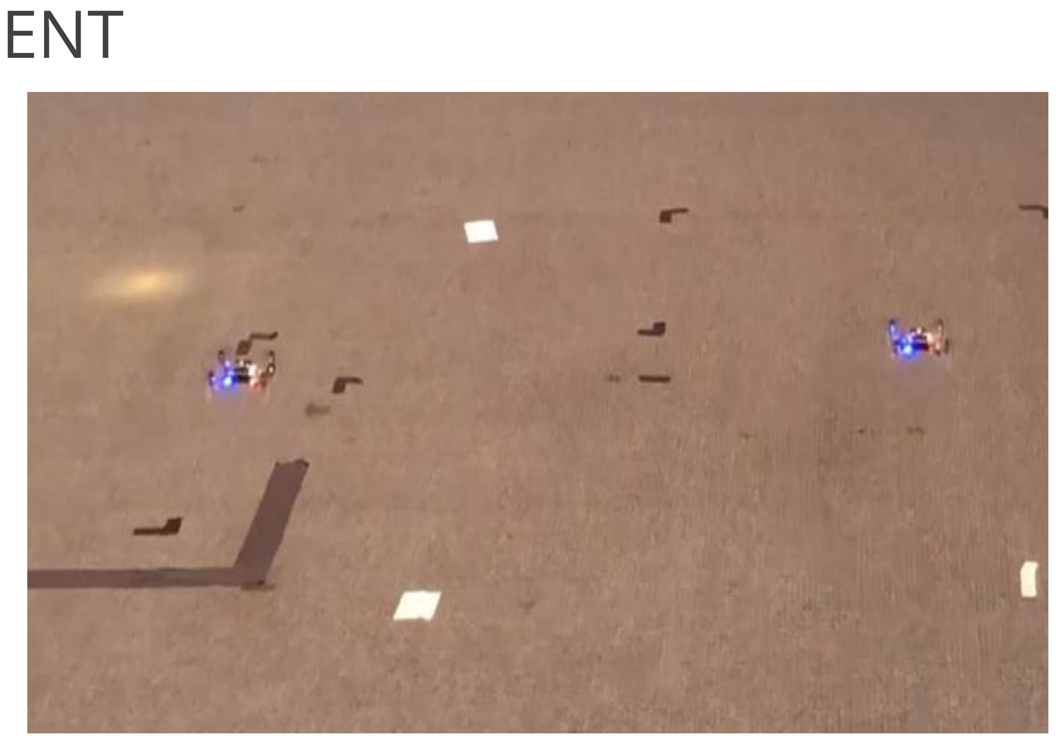


Fig 3. Crazyflies Indoor Using Vicon System

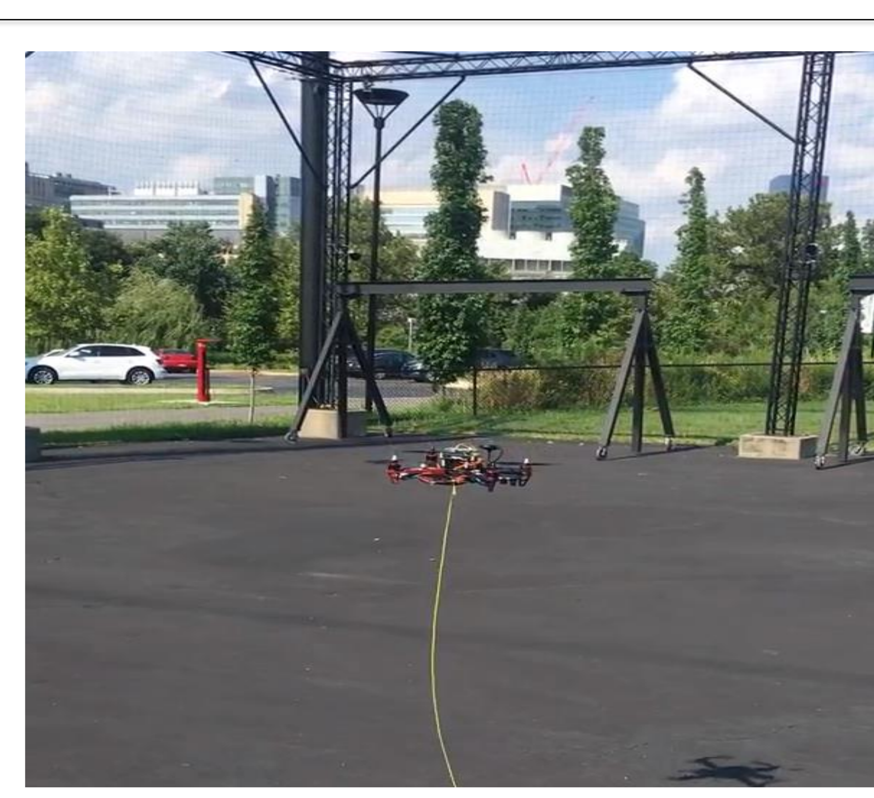


Fig 3. Drone Outdoor Using Pixhawk Board and Optical Flow Sensor

9 Conclusions

- Robust for dynamical systems with STL specifications.
- Through simulations as well as experiments on actual quadrotors, we show the applicability of a Real-time high-level controller in a hierarchical control scheme.
- Sequential planning method is a good way for commercial application to protect privacy information without losing safety guarantee.