Local Navigation for the Inspection of Steel Structures

Agatha Duranceau

Professor : Alcherio Martinoli
Assistant(s) : Lucas Wälti

Inspection operations often require the use of heavy equipment or putting human operators in potentially dangerous environments. The use of mobile robotics could allow for cheaper, faster and safer inspections, notably with UAVs that can evolve in three dimensions and thus access hard-to-reach areas. Over the recent years, the wide range of commercially available drones with flexible sensor integration possibilities have allowed a large increase in the use of UAVs for inspection. Nonetheless, inspection operations still often require a human operator to remotely control the drone at some point. Hence, fully autonomous flight in cluttered environments is still an active area of research. In this project, we have therefore focused on the navigation aspect of the inspection task, and explored ways that a quadrotor drone can perform path re-planning in real time to safely avoid obstacles in cluttered environments.

After performing a literature review of trajectory generation and path re-planning strategies, we have applied a state-of-the-art method to compute time-continuous smooth trajectories interpolating a sequence of desired waypoints for the drone to visit. Smoothness is beneficial to avoid saturation of the motor commands, as well as to preserve sensor measurement quality. To achieve smoothness, piecewise-continuous polynomial segments were used to generate the trajectory, and polynomial coefficients were found by minimizing the squared jerk or snap along the trajectory. To gain more insight on the proposed trajectory computation method, we have studied the impact of the different hyperparameters of the approach. The resulting trajectories were applied with both position and velocity control on a simulated drone.

In a second phase, we have proposed a path re-planning strategy for partially unknown environments and tested it in simulation. We have used Voxblox to incrementally build a map, using the depth camera embedded on the drone. The simulation starts with a trajectory interpolating a sequence of desired waypoints, and collisions are checked along the remaining part of the trajectory thanks to the Euclidian Signed Distance Field (ESDF) map generated by Voxblox. If a collision is met, a new waypoint is added to the collision-free space by moving the collision point following a search direction, which has a component along the gradient direction of the ESDF (i.e. moving away from the obstacle) and a component perpendicular to the gradient (i.e. moving along the obstacle). The search stops when the new point is far enough from the initial collision. Further improvements were made after observing the first experimental results.

Figure 1 shows an example of a re-planned trajectory that has been computed with the previously described method, where the drone is initially tasked to fly between two waypoints and find a trajectory allowing it to reach its goal, while avoiding a series of obstacles located along its path:

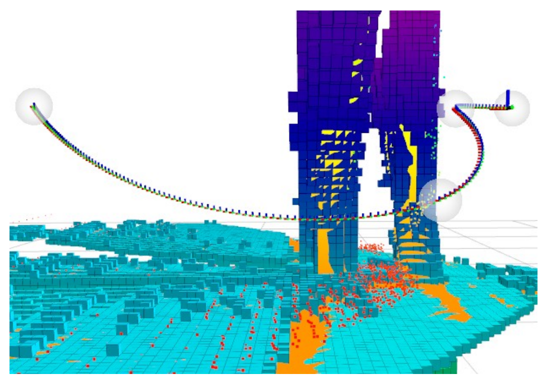

Fig. 1: Example of a re-planned drone trajectory

Overall, the drone was able to avoid the obstacles along its trajectory and reach its final goal. The ROS implementation of the approach makes it compatible with other multicopters. However, the proposed method contained several hand-tuned hyperparameters (such as collision distance thresholds, etc.) that might not be adapted to other scenarios, which limits the general robustness of the approach. Perfect localization was also assumed, which is not the case in real-world settings. Finally, the closed-form solution that was used to solve the optimization problem and compute the polynomial coefficients does not allow flexibility in the design of the cost function to avoid certain undesirable effects on the trajectory.