

# Homework 12

**Video:** [https://youtu.be/4Um4W\\_XyDfc](https://youtu.be/4Um4W_XyDfc)

**Self-Assessment:** 100 points out of 100

## **Notes:**

This video begins by demonstrating my point to point flight through a city. Because I precompute the animation does not happen at the same time as the RRT. As such the flow of the video goes as follows. The algorithm uses RRT to calculate a specified number of iterations of RRT. If the end point is within the step distance of a new node it is added to the tree. The algorithm then chooses the shortest path from the start to end point and that path is smoothed as much as possible. The animation then shows the aircraft following this path. Note that because the flight is performed at a constant altitude, some building (obstacles) are low enough to fly over.

The video then demonstrates my coverage algorithm. This functions very similarly to the point to point algorithm. It begins by generating the look ahead tree by iterating a specified number of steps though RRT. After doing this it chooses the node with the highest return value (the node that has been visited the least) and calculates a smoothed path to that node. This is shown real time. The simulation then propagates the dynamics to follows the aircraft's flight from its current position to the chosen node. Upon reaching that node, the cycle begins again by generating a new look ahead tree. This simulation is run for 1000 seconds of simulation time. Then the animation begins at the starting point and shows the aircraft flying the entire chosen path. Note that because the flight is performed at a constant altitude, some building (obstacles) are too high to fly over limiting the area that can be completely covered.

Note that all simulations are run with 5 m/s wind and moderate high-altitude turbulence. For speed in pre-calculation, this is performed with true states and not the EKF, which I have shown to work in previous homework. Also note that nodes are only added to the tree if a feasible dubbin's path exists between the closest node and the new node. This is why my tree is relatively sparse. I am pushing the limits of my algorithm by performing this in a relatively tight space (i.e. the space between buildings is small compared to the minimum turning radius of the aircraft).