29

Is there something major I'm missing regarding the Quadcopter project?

Nathaniel W 8 months ago

I've been stuck on the Train a Quadcopter to fly project for a while now and it's been really frustrating since I feel like the objectives/challenges for this project have not been communicated well. Looking in the MLND Slack, I see that many students are experiencing the same frustrations, but I don't see anyone having any breakthroughs, which seems to indicate that everyone is missing some big, important piece of information. The closest I've heard are people saying to submit it with an agent that doesn't actually fly, and maybe you'll pass.

I want to change that, but I don't have all the answers. Below is a list of all the small breakthroughs I've had while working on this project, in no particular order. My biggest achievement so far is that sometimes my quadcopter doesn't crash into the ground for 5 seconds straight, so take this all for what it's worth. Please share if there's anything useful that you realized that's not covered here, and hopefully we can all figure out this project together.

- The reward function seems to be the critical factor holding back the agent from learning properly; *do this first*.
- The starter code doesn't have any way to recognize if the target position was reached. (*Big Eureka moment right here*)
- The physics_sim doesn't have a floor, if you set `init_pos` too low and if the agent doesn't apply thrust right away, it will just fall below 0 and end the episode. (Thanks to Serchan Soydan in the Slack for this helpful nugget)
- Without specifying clear and large rewards for performing well, the agent might just decide it's better to crash early and end the episode sooner.
- Euler angles are in radians and here's what they mean:
- `x`/`phi` corresponds to yaw
- `y`/`theta` corresponds to roll
- `z`/`psi` corresponds to pitch
- The angles in the pose variable are all positive, so numbers like 6.1 are actually just a small negative variance from 0 (360 degrees = 2 Pi radians)