Agile Trajectory Generation for Tensile Perching with Aerial Robots

Progress Update

- Integrated the Spline portion.
- Fixed a bug with the plotting.
- Added confirmation at each stage and a choice of 5 different starting positions

Experiments

- Meeting with Atar/Kangle today 3pm to setup and verify that we have appropriate safety mechanisms.

Report Plan

- Main Contributions
  - PyBullet Simulation to approximately model a tether using a variety of pieces.
  - Learning from Demonstrations Integrations.
  - Set of Produced Trajectories optimised for speed.
  - Controller for Gazebo based on fixed time movement waypoints.

- Outcomes
  - Number of Demonstrations: Performed main training with 5 full demonstrations.
    - Performs well even with a single demonstration (Due to the way sampling between replay buffers is actually performed)
    - Reasonably resiliant to poor demonstrations - In
  - Comparison with Optimised Trajectory Approach
    - Uncertainty in live environments
    - In real-world environments a drone may not follow exactly a planned trajectory.
    - The NN approach is resilient to this:
      - If the drone doesn't follow exactly a planned path then using current positions it can reproduce points during operation.
      - This would not be possible in real-time using an optimisation based approach.
      - E.g. For the optimised approaching this can take up to 30 seconds to compute an optimised trajectory - this is not feasible to be adaptable during an actual flight.

Report Plan

- Introduction
- Ethical Considerations
- Background - mostly re-used from my interim report.
- Environment
  - Scenario Design
  - PyBullet Environment
  - Approximate Tether Modelling
    - Analyse the accuracy compared to real world experiments.

- Training
  - Wrappers and Effects
    - Waypoint - Description of fixed time waypoint system - effects on speed in comparison to real life.
    - Dimension - Effect of reducing dimension complexity.
    - Symmetry - Effects of assuming a symmetric environment - alternatives considered and their effects compared to the symmetrical design choice.
    - Other Wrappers: Briefly mention others with less details.
  - Training
    - Reward System Design
    - Algorithms - SAC, NAC, SACfD
    - Demonstrations - Comparison of different training techniques.
- Gazebo Offboard Controller
  - Running modes
  - Safety aspects for live environments

- Evaluation
  - Number of Demonstrations & Non-optimal demonstrations
    - As described in outcomes slide
  - Speed
  - Uncertainty in Live environments described above.
- Conclusion & Future Outlook

Overall Plan

- Report Deadline 17th June (2.5 weeks)
  - Today & Friday - Experiments
  - Aim to have full 1st draft of report by next week 12th June.
  - 5 days for final revisions.

Questions

Report

- Order of evaluation and implementation.
  - E.g. PyBullet Environment matching real-world tests - Visuals that show how the simulated tether and real tether match.
  - Is it better to include smaller evaluation pieces as I describe implementation or in a seperate section?