F450 Readme File

Chih-Chun Chen

April 8, 2022

1 Folders

1. **launch:** This folder includes launch files. Run files via "roslaunch f450 xxx.launch" command.
   - **f450_apm.launch:** This launch file includes the *mavors (rospackage) apm.launch* file, and the file is wrapped by a *namespace*, while can add an "name" before the rostopic/rosservice name. e.g. /mavros/imu/data to /uav1/mavros/imu/data. This is useful for multi-uav experiments. Run this file if the fcu has the apm firmware uploaded and multi-uav is needed. Arguments: name="uavID" default="uav1"; name="fcu_url" default="/dev/ttyUSB0"
   - **f450_test.launch:** Similar to **f450_apm.launch:**, the only difference is that the *mavors (rospackage) px4.launch* file is included. Therefore, use this file when the fcu has the px4 firmware uploaded.
   - **multi_uav_mavros_sitl_#.launch:** This launch file is run when running px4 gazebo software-in-the-loop (sitl) simulation. # represent the number of UAV added into the gazebo world. *"o"* represents the obstacle environment is used. The obstacle world is defined in f450 rospackage worlds folder obstacle_one.world file.
2. **scripts:** This folder includes python files. Run files via "rosrun f450 ros_x_pathplan.py" command.
   - **ros_f_pathplan.py:** This file allows the gcs to plan/command UAVs path with flocking algorithm. Pure flocking, 2D/3D, with/without downwash force, with/without adaptive model ← change all this modes and other parameters in the **Config.py** file. This file gets the final/initial/current position and other necessary information from outdoor_gcs (See *https://github.com/chenc159/outdoor_gcs* for details), and calculate the next desired position and return back to outdoor_gcs. Data transfer via rostopic.
   - **ros_o_pathplan.py:** This file allows the gcs to plan/command UAVs path with Optimal Reciprocal Collision Avoidance (ORCA) algorithm. Remember to git clone the open-source ORCA algorithm. The ROS part of this file is similar to **ros_f_pathplan.py**.
   - **Note: ros_f_pathplan.py** and **ros_o_pathplan.py** cannot be run simultaneously.

3. **sh:** This folder includes shell (.sh) scripts, which can run multiple files. Run files via "rosrun f450 xxx.sh" command.

   - **local_launch.sh:** This file allows the computer to ssh into the given onboard computer ip and run command. Currently, this file allows the computer to ssh into the onboard computer three times and connect pixhawk (px4.launch), run px4_command estimator , and run px4_command controller, respectively. Tmux is used for better reading. The first argument takes in the session id. The second argument takes in the onboard computer ip address. The third argument takes in the uav id (namespace). Run this file with the following example command: "rosrun f450 local_launch.sh ros1 odroid@192.168.2.179 uav1".
   - **local_launch_apm.sh:** Similar to **local_launch.sh**, but with apm running on the fcu.
   - **sitl.sh:** This file is used for multiple px4 UAV sitl gzebo simulation. First command line run the **multi_uav_mavros_sitl_#.launch** file. Gazebo environment shall appear, and multiple UAV shall appear. The next few line run the px4_command estimator and controller for all the UAV added in the gazebo. Then, the outdoor_gcs and path planning algorithm are called.
   - **sitl_apm.sh:** This file run the single Autopilot UAV sitl in the gazebo environment.