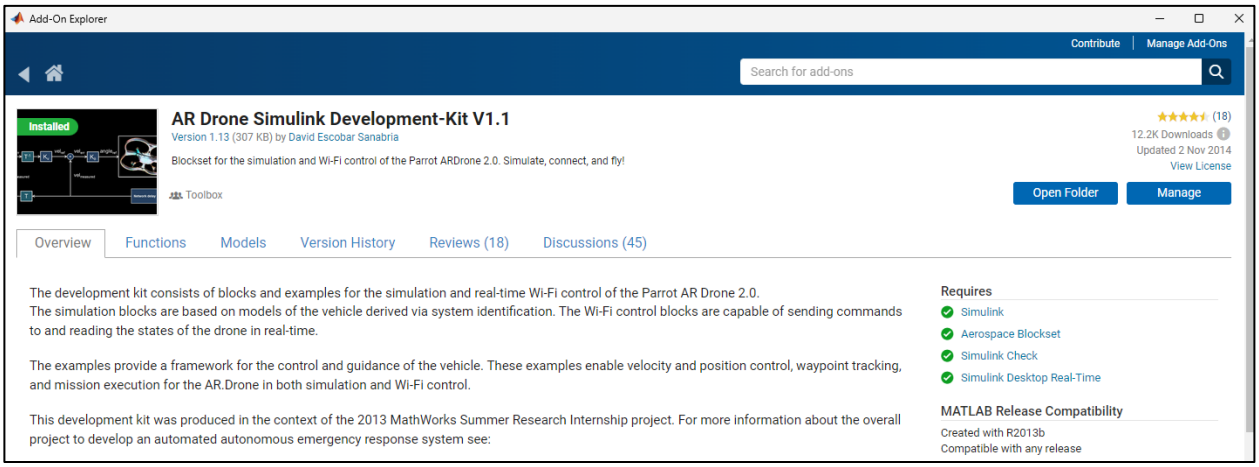
README

MATLAB README

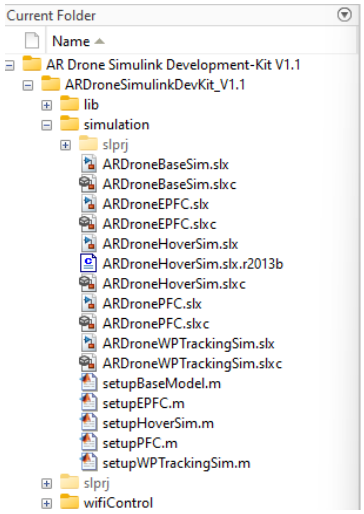
Steps to setup PFC and ePFC Simulink model:

1. Download the developed MATLAB files (setupPFC.m, setupEPFC.m) and the Simulink models (ARDronePFC.slx, ARDroneEPFC.slx).

2. Install AR Drone Simulink Development Kit V1.1 Add-On in MATLAB.



3. Go inside the library folder. Add the four developed files in */simulation* folder.



4. Double click on setupPFC.m or setupEPFC.m to view the script in a window. Click on the Run button. Wait for a few seconds as the respective Simulink model will load.

5. A tab with XY graph would load first. Click on the Run button and the XY trajectory of the drone would be plotted.

ROS README

ROS version: ROS Noetic
OS: Ubuntu 20.04

Developed package: epfc_controller
Dependency package: sjtu_drone

Setup dependency package:
$ sudo apt-get install libignition-math4-dev
$ cd <catkin_ws>/src
$ git clone https://github.com/tahsinkose/sjtu-drone.git
$ cd <catkin_ws>
$ catkin_make

To setup the developed package (epfc_controller), simply put it in src folder of catkin_ws and build it.

Steps to run different experiments:
For all experiments we first run a launch file which brings up gazebo. Then, we run one controller node.

Experiment 1:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_1.world*
$ roslaunch epfc_controller waypoints.launch

Pfc Controller node
pfc.py – uncomment lines 28-33 and comment lines 39-44
$ rosrun epfc_controller pfc.py

or

ePfc Controller node
epfc.py – uncomment lines 27-33 and comment lines 38-44 and 49-55
$ rosrun epfc_controller epfc.py

Experiment 2:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_2.world*
$ roslaunch epfc_controller waypoints.launch

Pfc Controller node
pfc.py – comment lines 28-33 and uncomment lines 39-44
$ rosrun epfc_controller pfc.py

or

ePfc Controller node
epfc.py – uncomment lines 38-44 and comment lines 27-33 and 49-55
$ rosrun epfc_controller epfc.py

Experiment 3:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_3.world*
$ roslaunch epfc_controller waypoints.launch

epfc Controller node
epfc_part2.py –
uncomment lines-37, 38, 106, 146, 163, 165.
comment lines-43, 44, 49, 50, 108, 147, 148, 164, 166, 167
$ rosrun epfc_controller epfc_part2.py

Experiment 4:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_4.world*
$ roslaunch epfc_controller waypoints.launch

ePfc Controller node
epfc_part2.py –
uncomment lines-43, 44, 106, 146, 147, 163, 165, 166.
comment lines-37, 38, 49, 50, 108,148, 164, 167
$ rosrun epfc_controller epfc_part2.py

Experiment 5:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_5.world*
$ roslaunch epfc_controller waypoints.launch

ePfc Controller node
epfc.py – uncomment lines 49-55 and comment lines 27-33 and 38-44
$ rosrun epfc_controller epfc.py

Experiment 6:

Launch file
Waypoints.launch - Change the world in line 7 to *waypoints_6.world*
$ roslaunch epfc_controller waypoints.launch

ePfc Controller node
epfc_part2.py –
uncomment lines-49, 50, 108, 148, 164,167.
comment lines-37, 38, 43, 44, 106, 146, 147, 163, 165, 166
$ rosrun epfc_controller epfc_part2.py