

四旋翼与无人车异构集群协同控制例程

1. 实验目的

通过四旋翼无人机与无人车的协同运动，理解多智能体系统的控制逻辑和通信机制。

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]；MATLAB2022B以上版本。

①：若使用Pixhawk 6X飞控，平台安装时的编译命令为：px4_fmu-v6x_default，推荐PX4固件版本为：1.12.3。其他配套飞控及编译命令请见：
<https://rflsim.com/doc/zh/1/Hardware.html>

- 硬件要求：笔记本/台式电脑1台^[2]。

①：推荐配置请见：<https://rflsim.com/>

3. 实验地址

例程目录：

[安装目录]\RflySimAPIs\10.RflySimSwarm\2.AdvExps\e6.Hetero_SwarmExps\1.3Cars3 UAVsSwarm

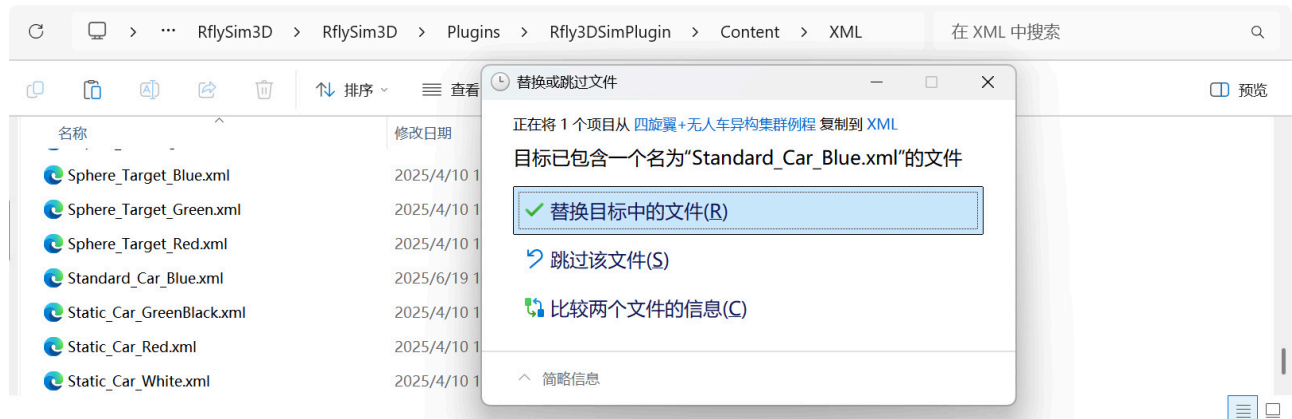
- [./Standard_Car_Blue.xml](#)：定义无人车模型参数的XML文件。
- [./UDPSimpleMultiVehicleSwarm.py](#)：主控制程序，实现无人机与无人车的协同控制算法。
- [./MultiVehicleRunALL.bat](#)：一键启动仿真环境批处理文件。
- [./Python38Run.bat](#)：Python运行环境批处理文件。

4. 实验内容或步骤

4.1 步骤1 替换XML文件

把Standard_Car_Blue.xml文件替换

到 *\\PX4PSP\\RflySim3D\\RflySim3D\\Plugins\\Rfly3DSimPlugin\\Content\\XML 文件夹内。



4.2 步骤2 启动仿真

双击运行MultiVehicleRunALL.bat脚本，会自启动1个RflySim 3D，1个QGroundControl，6个CopterSim，等待所有CopterSim信息栏出现PX4: GPS 3D fixed & EKF initialization finished. 字段。



4.3 步骤3 启动控制程序

打开Python38Run.bat脚本输入 python UDPSimpleMultiVehicleSwarm.py 并执行。

```
C:\WINDOWS\system32\cmd. x + v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python

F:\git\10.RflySimSwarm\2.AdvExps\6.Hetero_SwarmExps\1.3Cars3UAVsSwarm>python UDPSimpleMultiVehicleSwarm.py
0号飞机已到达指定高度
1号飞机已到达指定高度
2号飞机已到达指定高度
|
```

在RflySim3D查看实验效果。



5. 关键知识点

关键知识点1：多智能体协同控制（Multi-Agent Swarm Control）

`VehicleNum = 3` 和 `CarNum = 3` 定义了 3 架无人机和 3 辆无人车

无人机（ID 0-2）在达到目标高度后，将跟踪对应无人车（ID 3-5）的运动轨迹

```
1 | if flagI[j] == True:
2 |     mav.SendPosNED(MavList[j+3].uavPosNED[0], MavList[j+3].uavPosNED[1],
   |     mav.uavPosNED[2], 0)
```

无人车（ID 3-5）以固定步长（2.5米）沿X轴方向移动

```
1 | MavList[i].SendPosNEDNoYaw(MavList[i].uavPosNED[0]+2.5, MavList[i].uavPosNED[1], 0)
```

关键知识点2：PX4飞控的Offboard模式控制

初始化并进入 Offboard 模式：

```
1 | MavList[i].InitMavLoop(1) # 初始化 UDP 通信
2 | MavList[i].initOffboard() # 进入 Offboard 模式
3 | MavList[i].SendMavArm(True) # 解锁电机
```

关键知识点3：UDP 通信与仿真同步

查看此python文件查看具

体 `*:\PX4PSP\RflySimAPIs\RflySimSDK\ctrl\PX4MavCtrlV4.py`。

通信接口：

- 支持 UDP、串口(COM)、Redis 等多种通信方式
- 可与 PX4 飞控和 CopterSim 进行数据交互

飞行控制：

- 提供 Offboard 模式控制接口
- 支持位置、速度、加速度等多种控制模式

- 支持姿态控制

状态获取：

- 获取飞行器姿态、位置、速度等信息
- 获取 GPS 数据和传感器数据

仿真功能：

- 支持点质量模型仿真
- 可与 RflySim3D 进行交互

命令发送：

- 发送起飞、降落、模式切换等指令
- 发送 RC 遥控信号覆盖

30Hz 控制频率（仿真同步）

```
1 | timeInterval = 1/30.0 # 30Hz 控制频率
2 | while True:
3 |     lastTime = lastTime + timeInterval
4 |     sleepTime = lastTime - time.time()
5 |     if sleepTime > 0:
6 |         time.sleep(sleepTime) # 精确控制循环频率
```

关键知识点4：条件触发与状态机

无人机到达目标高度后触发跟踪逻辑

```
1 | if abs(mav.uavPosNED[2] - (-10)) < 2 and (flagI[j] == False):
2 |     flagI[j] = True # 标记"已到达目标高度"
```

无人车在无人机就位后才开始移动

```
1 | if flagI[i-3] == True:
2 |     MavList[i].SendPosNEDNoYaw(...) # 无人车前进
```

6.参考资料

1. [RflySim官方文档](#)

2. [PX4 Developer Guide](#)
3. [MAVLink Protocol Documentation](#)

I 7. 常见问题

I Q1: 在启动仿真时，CopterSim没有显示GPS 3D fixed信息，无法继续实验。

A1: 检查RflySim3D是否成功启动，确认网络连接正常，重启仿真程序。如果问题仍然存在，尝试重新安装RflySim工具链。

I Q2: 无人车与无人机协同工作不正常，无人机不能正确跟随无人车。

A2: 检查 [UDPSimpleMultiVehicleSwarm.py](#) 文件中的参数设置是否正确，特别是 VehicleNum和CarNum变量。确认代码中的索引对应关系正确。

I Q3: 在运行控制程序时出现连接错误或端口占用问题。

A3: 确保没有其他程序正在使用相同的UDP端口，关闭所有仿真程序后重新启动。检查防火墙设置，确保允许程序的网络访问权限。

-
1. <https://rflysim.com/> ↩
 2. 推荐配置请见: <https://rflysim.com/doc/zh/HowToInstall.pdf> ↩