

1、实验名称及目的

200 机质点集群实验：从模型精度的角度，使用高精度 6DOF 模型（CopterSim）+真实飞控系统（PX4）的软/硬件在环仿真闭环的方式，能够有效提高模型可信度，从而减小仿真与真机实验的差距。本实验基于 RflySim 平台实现在局域网内两台电脑 200 架质点模型的四旋翼飞机起飞和画圆飞行。

2、实验原理

为了提高单台电脑仿真集群飞机的数量，就需要降低模型精度并使用简化飞控模型。因此本平台在 Python 下开发出了质点多旋翼模型，只需 Python 和 RflySim3D 两个软件即可在单台电脑上实现百驾级别的无人机集群仿真。

3、实验效果

可实现局域网内两台电脑 200 个飞机先起飞并悬停，接着开始同步画圆。



4、文件目录

文件夹/文件名称	说明
NoPX4SITL200.bat	启动仿真配置文件
NoPX4SITL200PC1.py	电脑 A 实现功能主文件
NoPX4SITL200PC2.py	电脑 B 实现功能主文件
PX4MavCtrlV4.py	功能实现接口文件

5、运行环境

序号	软件要求	硬件要求	
		名称	数量

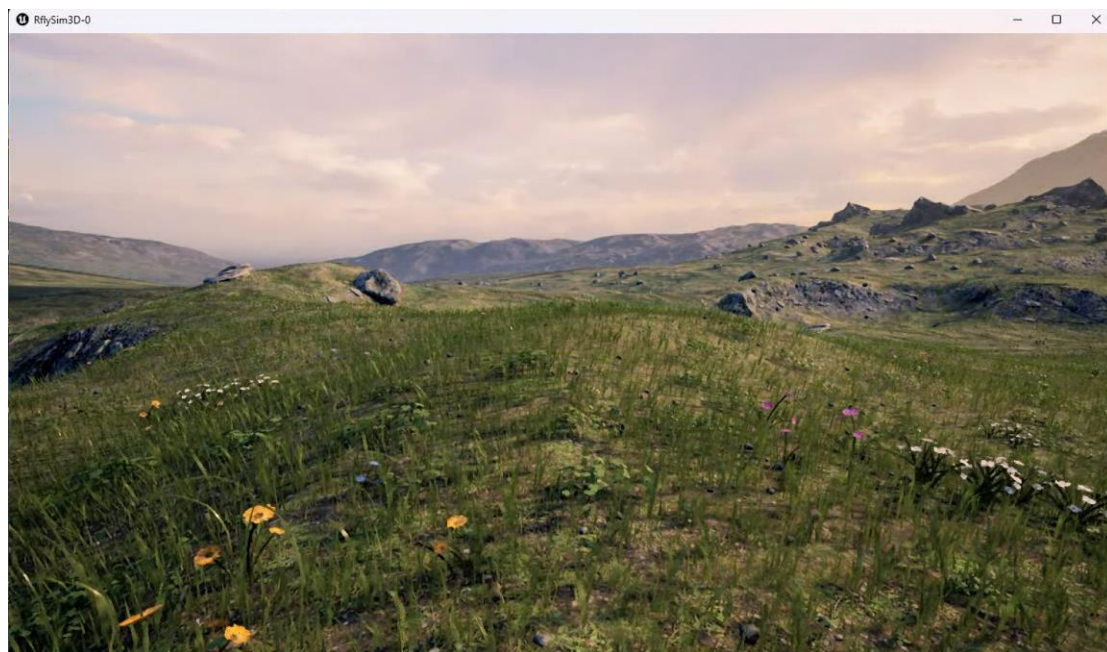
1	Windows 10 及以上版本	笔记本/台式电脑 ^①	1
2	RflySim 平台免费版		
3	Visual Studio Code		

①：推荐配置请见：<https://doc.rflysim.com/1.1InstallMethod.html>

6、实验步骤

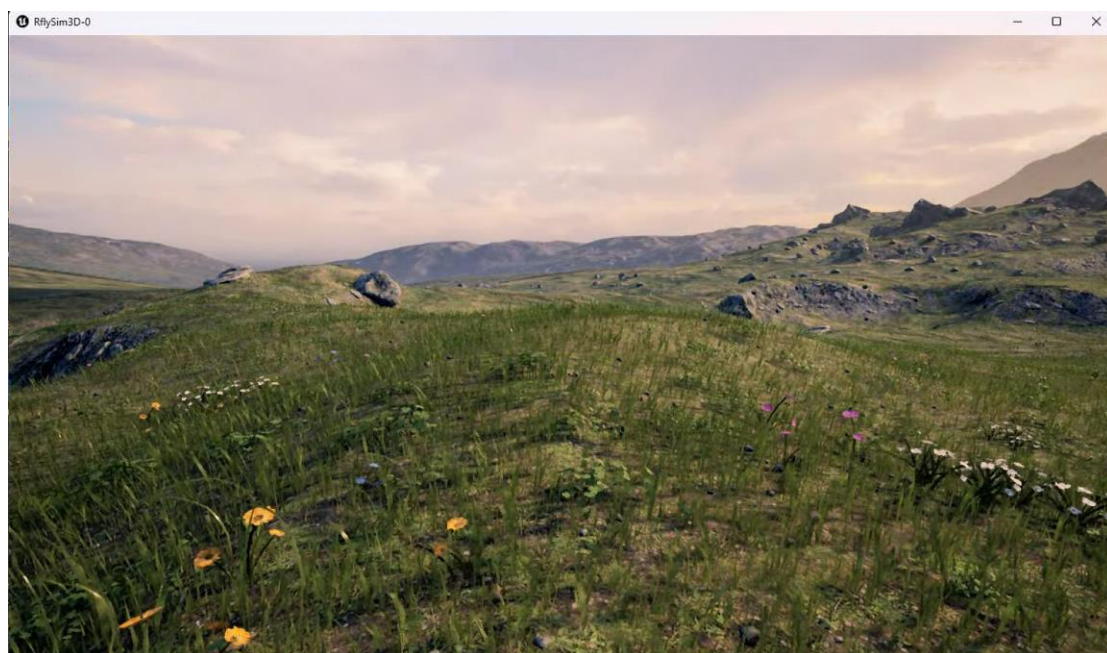
Step 1:

在电脑 A 上执行 NoPX4SITL200.bat 文件。只会出现 RflySim3D 软件。



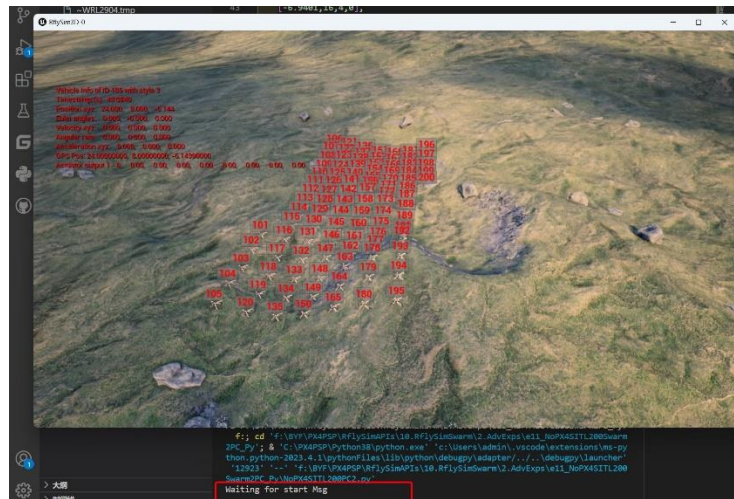
Step 2:

在电脑 B 上执行 NoPX4SITL200.bat 文件。只会出现 RflySim3D 软件。



Step 3:

在电脑 B 打开 VScode 本实验路径文件夹，运行 NoPX4SITL200PC2.py 文件，会在 RflySim 中看到生成的 100 架飞机，编号是从 101~200。并在 VScode 终端中显示：Waiting for start Msg。其效果如下所示：



Step 4:

在电脑 A 打开 VScode 本实验路径文件夹，运行 NoPX4SITL200PC2.py 文件，会在 RflySim 中看到生成的 100 架飞机，编号是从 1~100，可以在电脑 A、B 中看到 200 架飞机同时起飞并进行同时各自画圆。其效果如下所示：



7、参考文献

[1] 下面以单个飞机为例，介绍控制流程：

```

mav = PX4MavCtrl.PX4MavCtrl(20100) # 创建一号飞机实例
mav.initPointMassModel(-8.086,[0,0,0]) # 初始化质点模型循环
print((mav.uavPosNED,mav.truePosNED, # 打印数据
mav.SendPosNED(0, 0, -1.7, 0) # 发送目标位置
mav.SendMavArm(True) # 解锁飞控, 飞机起飞
time.sleep(5) # 代码暂停 5s, 飞机到达起飞点并悬停
mav.SendVelNED(0, 0, 1, 0) # 发送向下速度, 飞机降落
mav.EndPointMassModel() # 退出质点模型循环

```

[2] 代码解析如下: (与 4 飞机例子-[*PX4PSP\RflySimAPIs\10.RflySimSwarm\1.BasicExps\le9_NoPX4SITL4Swarm](#)-区别部分):

```

MavList=MavList+[PX4MavCtrl.PX4MavCtrl(20100+ii*2)] #建立飞机实例矩阵
InitPosList=[*****] # 配置飞机初值矩阵
MavList[i].initPointMassModel(InitPosList[i][0],InitPosList[i][1:4]) #通过矩阵初始化
Error2UE4Map = Error2UE4Map+[***] # 计算每个飞机起飞坐标系与 UE4 地图坐标系差值
MavList[0].sendUE4Cmd(b 'RflyChangeViewKeyCmd S' ) # RflySim3D 显示飞机数字标号
MavList[0].sendUE4Cmd(b 'RflyChangeViewKeyCmd T' ) # RflySim3D 显示飞机轨迹
MavList[i].SendPosNED(0, 0, -10, 0) # 飞机各自起飞到 10m 高 (以起飞点为坐标系)
targetPosE=np.array([-0.0,-15]) # 设置默认高度 15m, 所有飞机汇集到本坐标
targetPosE=np.array([10*math.sin(t/2+math.pi/2)-10,10*math.sin(t/2.0),-15]) # 生成圆形轨迹
targetPosE=targetPosE+Error2UE4Map[j] # 将圆形轨迹映射到各飞机起飞坐标系
mav.SendPosNED(targetPosE[0],targetPosE[1],targetPosE[2],0) #发送圆形轨迹
MavList[i].EndPointMassModel() # 各飞机退出仿真循环

```

[3] 关键代码解析

- 200 个飞机的初始位置和地形高度生成: 见[*PX4PSP\RflySimAPIs\10.RflySimSwarm\0.ApiExps\5.GetTerrainAPI](#) 例程, 关键点重复两次 100 飞机位置生成代码, 配置 “VehicleNum=100;TOTAL_COPTER=200;”, 针对电脑 1 设置 “START_INDEX=1;”, 针对电脑 2 设置 “START_INDEX=101;”, 运行 GenSwarmPos2PC200.m 可以得到两电脑的 InitPosList 列表
- NoPX4SITL200PC1.py 设置 “START_INDEX=1” 和 NoPX4SITL200PC2.py 设置 “START_INDEX=101”, 同时初始化脚本使用广播地址 “255.255.255.255”
MavList=MavList+[PX4MavCtrl.PX4MavCtrl(20100+ii*2,'255.255.255.255')]
- targetPosE 不经过 Error2UE4Map[j]的校准, 每个飞机画圆按照自己起飞坐标系来
- NoPX4SITL200PC2.py 中 “MavList[0].waitForStartMsg()”, 启用程序阻塞, 等到接收到来自局域网的开始仿真消息后, 才会继续执行后续的控制算法
- NoPX4SITL200PC1.py 中 “MavList[0].sendStartMsg()”, 发送开始仿真消息, 局域网内所有等待的程序收到本消息后, 会自动开始后续程序运行。

8、常见问题

Q1: 无

A1: 无