

多无人机集群筛选广播通信实验

1. 实验目的

本实验旨在实现一个多无人机集群环境下的筛选广播通信机制。通过该实验，学习如何构建一个高效的无人机集群通信系统，使得指定无人机可以根据需求主动请求其他无人机的状态数据，而被动无人机仅在接收到请求后才响应，从而避免全网广播造成的数据拥塞和资源浪费。

具体目标包括：

- 掌握无人机集群通信的基本原理和方法
- 学习基于请求-响应模式的高效通信机制
- 理解如何减少不必要的全网广播，优化网络资源利用
- 实现按需获取特定无人机状态数据的功能

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]。
- 硬件要求：笔记本/台式电脑1台^[2]。

3. 实验地址

例程目录：

[\[安装目录\]\RflySimAPIs\9.RflySimComm\0.ApiExps\e1.RflyNetAPIExps\8.BroadReqDemo](#)

- [Python38Run.bat](#)：RflySim Python运行脚本
- [RunAll.bat](#)：运行所有无人机控制脚本- [SITLRun3MavlinkFull.bat](#)：启动SITL仿真环境脚本- [UAV1Ctrl.py](#)：主控无人机（1号）控制脚本- [UAV_Passive_Ctrl.py](#)：被动无人机（2、3号）控制脚本

4. 实验内容或步骤

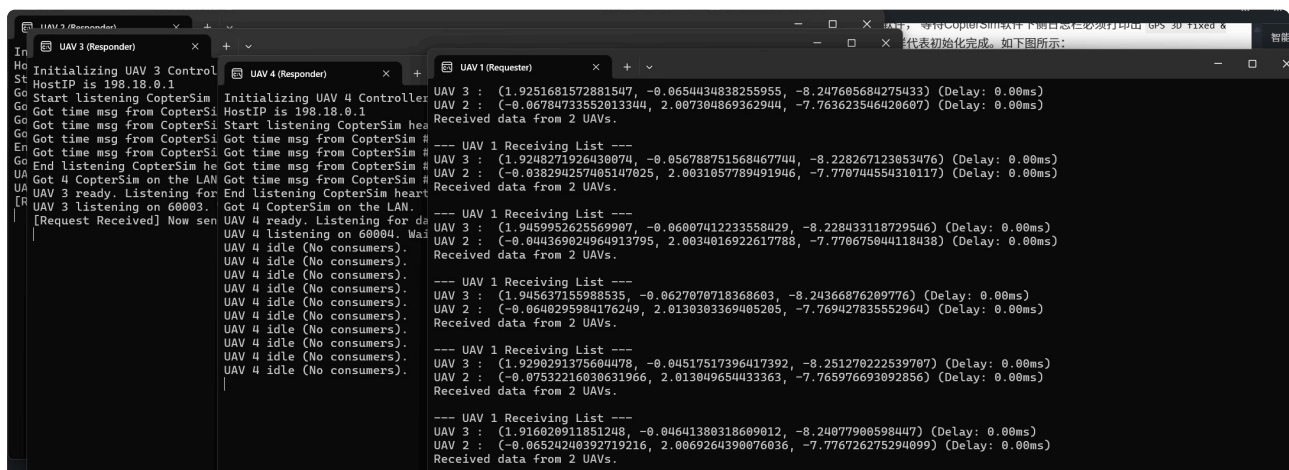
4.1 仿真初始化

双击运行SITLRun.bat文件，等待仿真环境初始化完成。脚本将会启动 1 个 QGC 地面站，1 个 CopterSim、1 个 RflySim3D 软件，等待CopterSim软件下侧日志栏必须打印出 GPS 3D fixed & EKF initialization finished 字样代表初始化完成。如下图所示：



4.2 仿真程序运行

直接双击 RunAll.bat 脚本，将直接运行 UAV1Ctrl.py、UAV_Passive_Ctrl.py 程序，其中 UAV_Passive_Ctrl.py 程序将初始化飞机2、3号飞机进行转发状态数据到1号飞机的程序中显示，如下图所示。



5. 关键知识点

本实验的核心思想是构建一个按需请求的无人机集群通信系统，其中1号无人机作为请求方，2、3号无人机作为被动响应方。这种机制避免了全网广播造成的资源浪费。

整体框架流程：

1. 1号无人机主动向2、3号无人机发送数据请求
2. 2、3号无人机在收到请求后，将其数据转发到1号无人机
3. 1号无人机接收并显示2、3号无人机的状态数据

主要程序解析：

UAV1Ctrl.py - 主控无人机程序：

此程序负责主动请求其他无人机的数据。

```
1 # 获取目标电脑IP并配置CopterSim回传数据到本电脑
2 TargetIP = req.getSimIpID(CopterID)
3 req.sendReSimIP(CopterID)
4
5 # 创建MAVLink通信实例
6 mav = PX4MavCtrl.PX4MavCtrler(CopterID,TargetIP)
7 net = NetSimAPIV4.NetSimAPI(mav)
```

核心功能是请求其他无人机数据：

```
1 # 启用网络转发功能，指定目标无人机ID
2 net.enUavForward([2,3])
3
4 # 开启请求模式：主动请求2，3号飞机的数据
5 net.StartReqUavData([2,3])
6
7 # 开启接收监听，监听自己的端口
8 net.StartNetRecOwn()
```

数据接收与显示：

```
1 while True:
2     for uav in net.UavData:
3         if uav.CopterID in [2, 3]:
4             print('UAV',uav.CopterID,': ',uav.uavGlobalPos, f"(Delay:
{uav.timeDelay*1000:.2f}ms)")
```

UAV_Passive_Ctrl.py - 被动无人机程序：

此程序使2、3号无人机处于被动响应状态，仅当接收到请求时才转发数据。

被动响应机制实现：

```
1 # 初始化并监听请求
2 net.StartNetRecOwn()
3
4 # 启动发送线程，但初始时不发给任何人
5 net.enNetForward([], '224.0.0.10')
6
7 # 监控请求者列表，动态添加发送目标
8 current_targets = net.netSendIDList
9 for target_id in current_targets:
10     if target_id not in known_targets:
11         target_port = 60000 + target_id
12         net.ForwardPort.append(target_port)
13         print(f"[Request Received] Now sending data to UAV {target_id} (Port
14 {target_port})")
15         known_targets.append(target_id)
```

通过这种方式，实现了按需通信，只有在被请求时才会建立数据转发通道，大大减少了网络流量和资源消耗。

6. 参考资料

1. [RflySim官方文档](#)
2. 无人机集群通信协议详解
3. MAVLink协议规范

7. 常见问题

Q1：运行过程中无法接收到其他无人机的数据。

A1：检查SITL仿真环境是否正确初始化，确认CopterSim软件下侧日志栏已显示 `GPS 3D fixed & EKF initialization finished`。同时检查网络配置是否正确，确保各无人机节点间可以正常通信。

Q2: 如何修改无人机ID?

A2: 在 `UAV_Passive_Ctrl.py` 中, 可以通过命令行参数指定无人机ID, 如 `python UAV_Passive_Ctrl.py 3` 表示将该无人机设为3号。也可以直接修改文件中的 `CopterID=2` 变量值。

Q3: 为什么采用被动响应机制?

A3: 被动响应机制避免了全网广播造成的网络拥塞。只有在被请求时, 无人机才会向特定目标转发数据, 这样可以有效节约网络带宽, 提高通信效率, 特别适用于大规模无人机集群场景。

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