

分布式局域网广播通信8机仿真实验例程-Simulink（分布式仿真集中式控制）

1. 实验目的

本实验旨在演示基于RflySim的分布式仿真系统，通过局域网实现多机协同仿真控制。具体目的包括：

1. 学习并掌握分布式仿真系统的构建方法
2. 理解基于局域网的多节点协同仿真原理
3. 掌握使用Simulink进行集中式控制的方法
4. 实现单台主机控制两台计算机共同完成8架无人机的协同飞行任务
5. 熟悉基于Git的分布式部署机制
6. 掌握仿真配置文件的设计与应用
7. 理解跨计算机的UDP通信机制在无人机仿真中的应用

通过本实验，用户能够深入理解分布式仿真系统的架构设计，掌握多机协同控制的关键技术，为复杂场景下的无人机集群仿真奠定基础。

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]，MATLAB 2022b及更高版本。
- 硬件要求：笔记本/台式电脑2，交换机，网线^[2]。

3. 实验地址

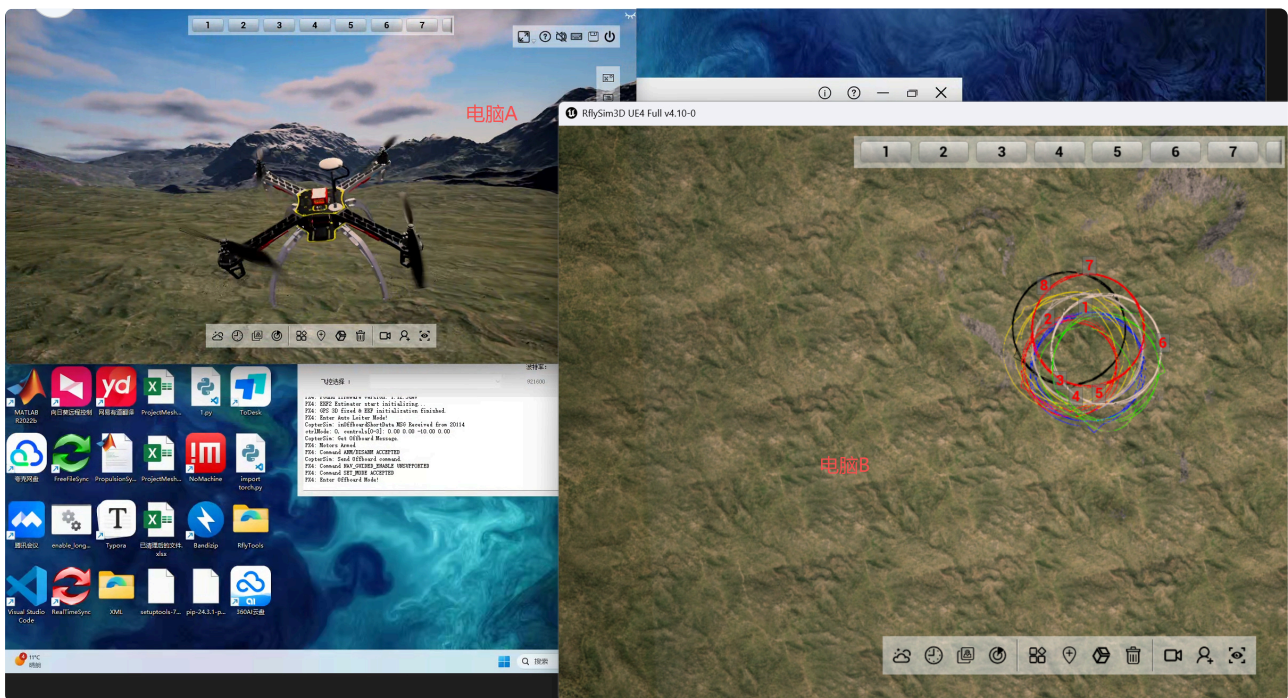
例程目录：

[\[安装目录\]\RflySimAPIs\10.RflySimSwarm\3.CustExps\e5.DSSwarmCtrls\3.CustIPDS8 UAVs\1.BroadDS8UAVs](#)

- `./DistSimMasterFlowCtrl.py`：分布式仿真主流程控制脚本，用于解析 `simulation_config.json` 配置文件，进行整个分布式仿真全生命周期的流程控制。
- `./simulation_config.json`：仿真配置文件，定义分布式仿真节点和各阶段任务。
- `./LaunchGitBareServer.bat`：启动Git裸仓库服务器的批处理脚本，用于分布式部署。
- `./Python38Run.bat`：Python运行环境设置脚本。
- `./deploys`：分布式部署文件夹。
- `./deploys/PC1`：PC1节点部署文件夹。
- `./deploys/PC1/SITLRunUdpSimple1_4.bat`：PC1启动1-4号无人机的批处理脚本。
- `./deploys/PC1/OneKeyStartExe.bat`：PC1一键启动程序的批处理脚本。
- `./deploys/PC1/CloseExe.bat`：PC1关闭程序的批处理脚本。
- `./deploys/PC1/RflyUdpUltraSimpleEight_Dist.exe`：PC1使用的可执行文件。
- `./deploys/PC2`：PC2节点部署文件夹。
- `./deploys/PC2/SITLRunUdpSimple5_8.bat`：PC2启动5-8号无人机的批处理脚本。
- `./Readme.pdf`：实验说明文档。

4. 实验内容或步骤

通过电脑A控制本机的4架无人机和电脑B的4架飞机进行分布式画圆。



4.1 步骤1：实验前准备

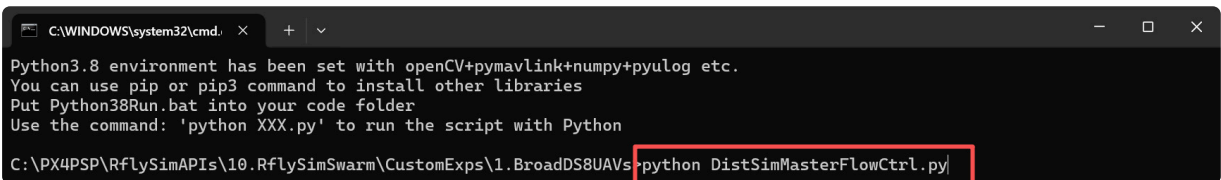
1. 根据

[安装目录]\PX4PSP\RflySimAPIs\10.RflySimSwarm\3.CustExps\e0.CustApiExps\2. DistSimCommAPIExps\1.HelloWorldTestExp\Readme.pdf"

进行配置DistSim软件，保证局域网内电脑可以相互通信，在进行以下操作。

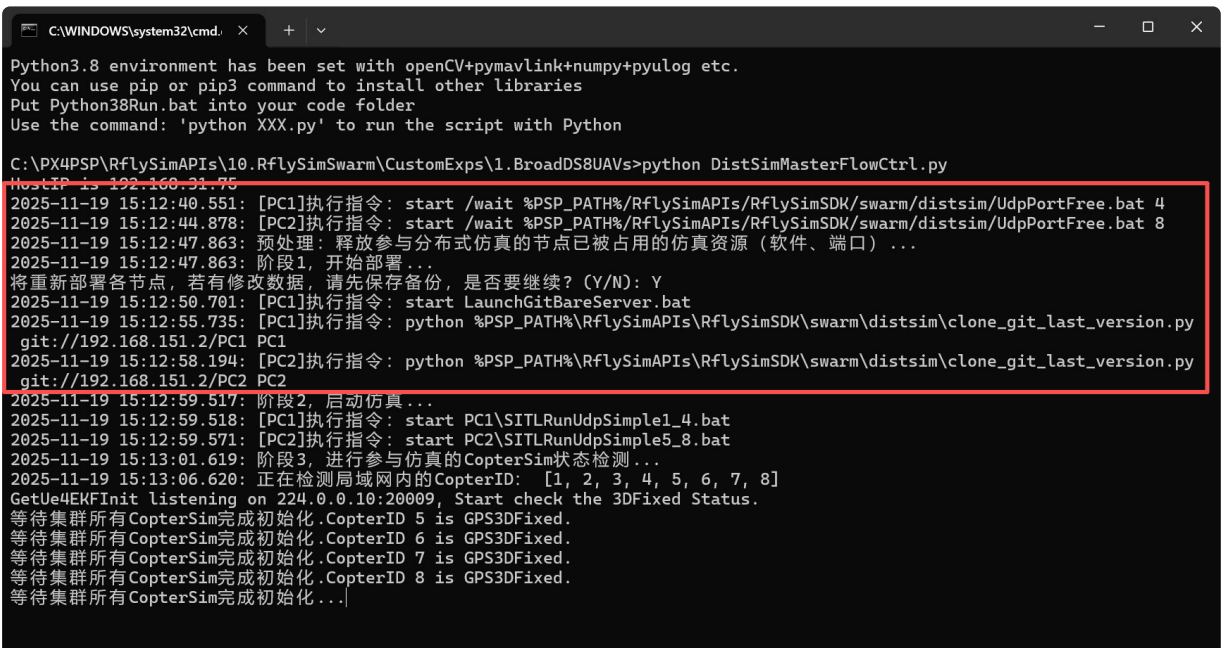
4.2 步骤2：启动软件在环仿真

1. 打开Python38Run.bat脚本，输入python DistSimMasterFlowCtrl.py，并运行。



```
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
C:\PX4PSP\RflySimAPIs\10.RflySimSwarm\CustomExps\1.BroadDS8UAVs>python DistSimMasterFlowCtrl.py
```

2. 可以在cmd命令行看到控制电脑A打开4个CopterSim，1个QGC，1个RflySim3D，电脑B打开4个CopterSim，1个RflySim 3D，开始部署电脑A和电脑B的代码。



```
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
C:\PX4PSP\RflySimAPIs\10.RflySimSwarm\CustomExps\1.BroadDS8UAVs>python DistSimMasterFlowCtrl.py
HostIP is 192.168.31.75
2025-11-19 15:12:40.551: [PC1]执行指令: start /wait %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\UdpPortFree.bat 4
2025-11-19 15:12:44.878: [PC2]执行指令: start /wait %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\UdpPortFree.bat 8
2025-11-19 15:12:47.863: 预处理: 释放参与分布式仿真的节点已被占用的仿真资源 (软件、端口) ...
2025-11-19 15:12:47.863: 阶段1, 开始部署...
将重新部署各节点, 若有修改数据, 请先保存备份, 是否要继续? (Y/N): Y
2025-11-19 15:12:50.701: [PC1]执行指令: start LaunchGitBareServer.bat
2025-11-19 15:12:55.735: [PC1]执行指令: python %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\clone_git_last_version.py
git://192.168.151.2/PC1 PC1
2025-11-19 15:12:58.194: [PC2]执行指令: python %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\clone_git_last_version.py
git://192.168.151.2/PC2 PC2
2025-11-19 15:12:59.517: 阶段2, 启动仿真...
2025-11-19 15:12:59.518: [PC1]执行指令: start PC1\SITLRunUdpSimple1_4.bat
2025-11-19 15:12:59.571: [PC2]执行指令: start PC2\SITLRunUdpSimple5_8.bat
2025-11-19 15:13:01.619: 阶段3, 进行参与仿真的CopterSim状态检测...
2025-11-19 15:13:06.620: 正在检测局域网内的CopterID: [1, 2, 3, 4, 5, 6, 7, 8]
GetUe4EKFFinit Listening on 224.0.0.10:20009, Start check the 3DFixed Status.
等待集群所有CopterSim完成初始化.CopterID 5 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 6 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 7 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 8 is GPS3DFixed.
等待集群所有CopterSim完成初始化...|
```

3. 在软件在环脚本脚本启动成功，自动通过局域网识别，电脑A电脑B所有的CopterSim的信息栏初始化成功，并执行程序。

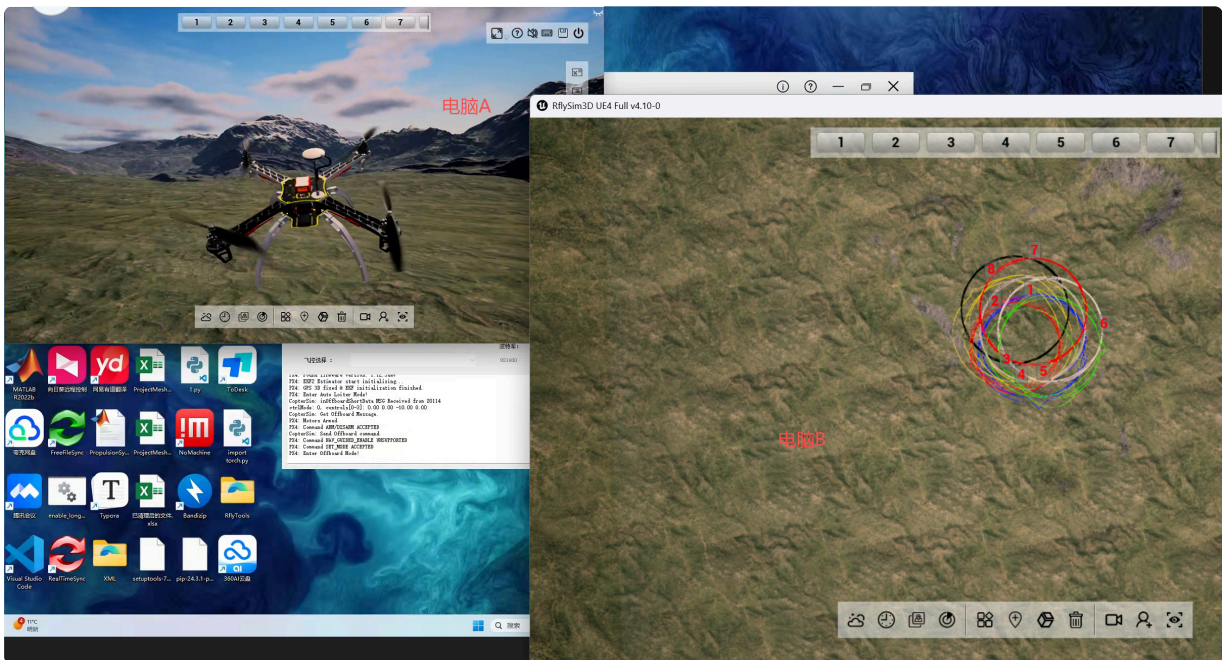
```

C:\WINDOWS\system32\cmd. x + v
2025-11-19 15:14:56.379: [PC2]执行指令: start /wait %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\UdpPortFree.bat 8
2025-11-19 15:14:59.260: 预处理: 释放参与分布式仿真的节点已被占用的仿真资源(软件、端口)...
2025-11-19 15:14:59.260: 阶段1, 开始部署...
将重新部署各节点, 若有修改数据, 请先保存备份, 是否要继续?(Y/N): y
2025-11-19 15:15:02.475: [PC1]执行指令: start LaunchGitBareServer.bat
2025-11-19 15:15:07.511: [PC1]执行指令: python %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\clone_git_last_version.py
git://192.168.151.2/PC1 PC1
2025-11-19 15:15:10.943: [PC2]执行指令: python %PSP_PATH%\RflySimAPIs\RflySimSDK\swarm\distsim\clone_git_last_version.py
git://192.168.151.2/PC2 PC2
2025-11-19 15:15:12.316: 阶段2, 启动仿真...
2025-11-19 15:15:12.316: [PC1]执行指令: start PC1\SITLRunUdpSimple1_4.bat
2025-11-19 15:15:12.361: [PC2]执行指令: start PC2\SITLRunUdpSimple5_8.bat
2025-11-19 15:15:14.409: 阶段3, 进行参与仿真的CopterSim状态检测...
2025-11-19 15:15:19.410: 正在检测局域网内的CopterID: [1, 2, 3, 4, 5, 6, 7, 8]
GetUe4EKFFinit listening on 224.0.0.10:20009, Start check the 3DFixed Status.
等待集群所有CopterSim完成初始化.CopterID 5 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 6 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 7 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 8 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 1 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 2 is GPS3DFixed.
等待集群所有CopterSim完成初始化.CopterID 3 is GPS3DFixed.
等待集群所有CopterSim完成初始化..CopterID 4 is GPS3DFixed.
All CopterSim is GPS3DFixed. Now we can start the simulation.
2025-11-19 15:15:59.435: 阶段4, 启动仿真对象控制脚本...
2025-11-19 15:15:59.435: [PC1]执行指令: start PC1\OneKeyStartExe.bat
2025-11-19 15:16:01.476: 阶段5, 进入用户操作...

```

输入任意指令, 结束分布式集群仿真, 关闭所有窗口|

4. 可以通过电脑A或电脑B的RflySim 3D查看到实验效果。



5. 关键知识点

关键知识点1: 分布式仿真系统架构设计

分布式仿真系统架构是本实验的核心设计理念。该架构将计算任务分散到多个节点上执行, 有效提升了系统的可扩展性和容错能力。在本实验中, 采用了主从式架构设计, 其中PC1作为主控节点负责整体流程控制和任务分配, 而PC1和PC2作为执行节点分别控制4架无人机。这种设计允许系统在增加更多无人机或计算节点时, 只需简单地添加新的执行节点, 而不需要重构整个系统。

关键知识点2：局域网多节点协同通信机制

局域网多节点协同通信机制是实现分布式仿真的关键技术之一。本实验采用UDP广播通信方式，使得无人机状态信息能够在局域网内实时传播，确保各节点数据的一致性。这种通信机制具有低延迟、高效率的特点，非常适合实时性要求高的无人机仿真场景。通过这种方式，所有节点都能及时获得最新的状态信息，从而做出正确的控制决策。

关键知识点3：基于Git的代码分发与版本控制

为了实现可靠的分布式部署，实验采用Git裸仓库机制进行代码分发和版本控制。通过 [LaunchGitBareServer.bat](#) 启动Git服务，各节点可以从中心仓库拉取最新的仿真配置和控制脚本，确保所有节点使用相同版本的代码。这种方法不仅简化了代码部署过程，还提供了良好的版本管理功能，便于追踪和回滚变更。

关键知识点4：UDP协议在无人机仿真中的应用

UDP协议因其无连接、低开销的特性，在无人机仿真领域得到广泛应用。本实验利用UDP协议传输无人机的状态信息和控制指令，实现高效的分布式协同仿真。相比TCP协议，UDP避免了连接建立和维护的开销，更适合传输频繁更新但允许少量丢包的实时数据。在本实验中，UDP协议确保了控制指令能快速传达至各仿真节点。

关键知识点5：集中式控制与分布式执行的实现方法

集中式控制与分布式执行相结合是本实验的重要特点。通过单一控制脚本 [DistSimMasterFlowCtrl.py](#) 统一调度各节点任务，但具体执行分布在不同计算机上。这种方式既保证了控制的统一性和一致性，又充分利用了多机资源提升整体性能。控制节点负责解析配置文件、分配任务和监控执行状态，而执行节点专注于运行具体的仿真任务，两者分工明确，协作高效。

6. 参考资料

1. [RflySim官方文档](#)
2. [RflySim安装指南](#)

I Q2: 在DistSimMasterFlowCtrl.py程序后，在部署代码的cmd命令行出现端口占用如何解决？

```
Run Git Command: ['wsl', '-d', 'RflySim-20.04', '-e', 'bash', '-lic', 'git status --porcelain']
成功删除: C:\PX4PSP\RflySimAPIs\10.RflySimSwarm\CustomExps\e4_FourUAVSwarmDistSim\e4_FourUAVSwarmDistSim2win\repos\uavSwarm-win
Run Git Command: ['wsl', '-d', 'RflySim-20.04', '-e', 'bash', '-lic', 'git clone --bare /mnt/c/PX4PSP/RflySimAPIs/10.RflySimSwarm/CustomExps/e4_FourUAVSwarmDistSim/e4_FourUAVSwarmDistSim2win/deploys/uavSwarm-win /mnt/c/PX4PSP/RflySimAPIs/10.RflySimSwarm/CustomExps/e4_FourUAVSwarmDistSim/e4_FourUAVSwarmDistSim2win/repos/uavSwarm-win']
Launch Git Bare Server Success.
Git Server is Running, Do not Close it.
[463] Could not bind to 0.0.0.0: Address already in use
[463] Could not bind to ::: Address already in use
fatal: unable to allocate any listen sockets on port 9418
Launch Git Bare Server Success, Access URL[git://127.0.0.1/xxxx]

C:\PX4PSP\RflySimAPIs\10.RflySimSwarm\CustomExps\e4_FourUAVSwarmDistSim\e4_FourUAVSwarmDistSim2win>
```

A2: 这个错误是因为端口被其他进程占用，可以通过以下命令查看占用端口的进程：`sudo lsof -i:9418`。如果查询不到端口进程，重新启动端口占用的windows电脑，在重新运行程序。

I Q3: 如何保证分布式仿真中各节点间的同步？

A3: 通过UDP广播通信机制实现各节点间的数据同步，确保所有节点能够及时获得最新的状态信息，同时主控节点统一协调各执行节点的任务执行时间。

1. <https://rflysim.com/> ←
2. 安装方式请见：<https://rflysim.com/doc/zh/HowToInstall.pdf> ←