

# 大模型语音控制实飞测试

## 1. 实验目的

验证语音识别转化自然语言指令在室内真实无人机上的闭环控制可行性 [researchgate.net](https://www.researchgate.net)

。

## 2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链<sup>[1]</sup>。
- 硬件要求：笔记本/台式电脑 1台；光学动量捕捉系统 1套（可选）；FS-J150飞机1架<sup>[2]</sup>

。

## 3. 实验地址

例程目录：

[\[安装目录\]\RflySimAPIs\6.RflySimExtCtrl\3.CustExps\4.VoiceandCtrlUAVFly](#)

- Main\_OpenAI\_api.py：主控制程序：OpenAI 调用、指令解析、UDP 包构建与发送
- OpenAI\_audio.py：实时语音识别模块：音频采集、WebSocket 交互、文本返回
- LLM\_UAV.slx / LLM\_UAV\_init.m：MATLAB/Simulink 在环仿真模型
- RflyUdpMavlinkRealSim.bat：PX4 SITL 启动脚本

## 4. 实验内容或步骤

本实验在室内光学动捕环境下（也可采用光流定位），使用语音指令控制真实无人机完成起飞、位置控制、降落等动作，验证了语音识别与大模型控制在实际飞行场景中的可行性。

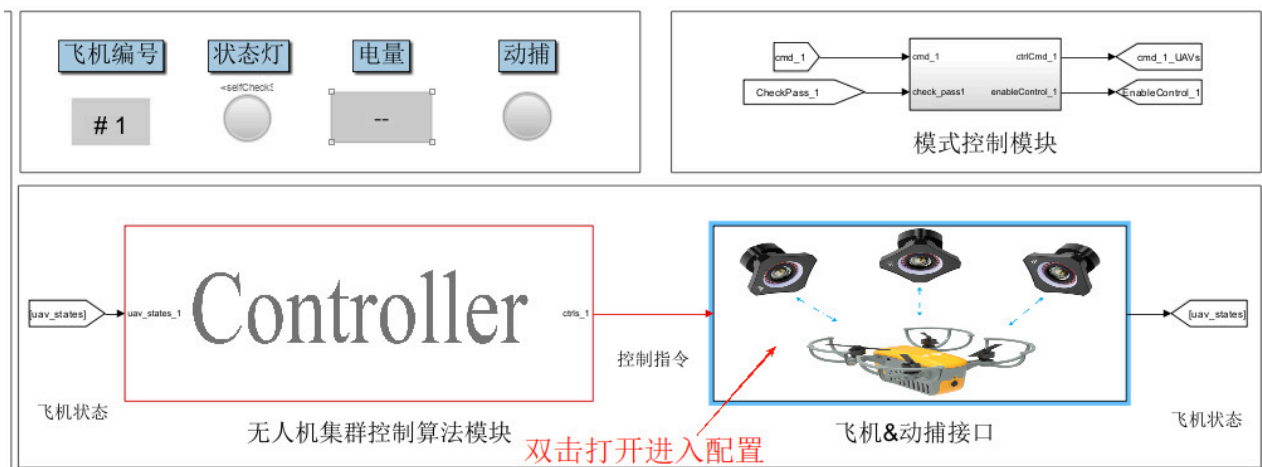
## 4.1 步骤1 Python环境切换

首先参照如下例程完成大模型API密钥与语音识别服务鉴权配置：[./Readme.pdf](#)。

## 4.2 步骤2 配置飞机环境并完成调试

参照[室内实飞实验.pdf](#)完成飞机调试和初步测试

在LLM\_UAV.slx文件中，实机测试需要进行无人机通信地址的配置，配置方法如下图所示：



LLM\_UAV ▶ Pa ▶

模块参数: RflyVrpnRecv1

RflyUdpReal API (mask) (link)

RflySimAPIs for multiple real uav data acquire and control

1. Base IP: The bash IP of the first UAV to receive MAVLink data
2. IP list: a string like 191.168.151.101,191.168.151.102 to specify the ip of each UAV
3. Vehicle number: Total vehicle number with StartID=1.  
For example Vehicle number=3 for a CopterID list [1 2 3]
4. CopterID List: a vector of CopterIDs, like [1 2 3] or [1:3] or [4 5 7]
5. Base UDP Port: The base port of the first UAV's mavlink port. You can also use a port list to specify the port of each UAV
6. RealflyMode: Checked for real uav control, the IP and port will auto increase 1 according to CopterID increase.
7. Scale for combination of virtual and real elements for X direction.
8. Scale for combination of virtual and real elements for Y direction.
9. Scale for combination of virtual and real elements for vertical direction.

Not checked for RflySim UDP simulation, the IP will not change and the port will +2 for each CopterID.

Input and output char arrays from buf for mavlink encode and decoder

参数 配置地址和端口

Vrpn IP Address  
127.0.0.1

Vehicle number or CopterID list (e.g., [1 3 4])  
[5 6]

Sample Time 1/30 0.033333  isSendVision

Drone base IP Address or IP list seperated by ','  
192.168.151.105,192.168.151.106

Drone base UDP Port or list  
15501

VR Scale X 1.0 Y 1.0 Z 1.0

确定(O) 取消(C) 帮助(H) 应用(A)

点击打开

RflyVrpnRecv  
send mavlink vision.  
VehicleNum [5 6]

RflyVrpnRecv1

RflyUdpMavlink  
Mavlink\_Real mode  
for Real fly.  
VehicleNum [5 6]

real\_data\_decoder4  
real\_data real\_bus States

real\_data\_decoder1  
real\_data real\_bus States

模块参数: RflyUdpMavlink1

RflyUdpReal API (mask) (link)

RflySimAPIs for multiple real uav data acquire and control

1. Base IP: The bash IP of the first UAV to receive MAVLink data
2. IP list: a string like 191.168.151.101,191.168.151.102 to specify the ip of each UAV
3. Vehicle number: Total vehicle number with StartID=1.  
For example Vehicle number=3 for a CopterID list [1 2 3]
4. CopterID List: a vector of CopterIDs, like [1 2 3] or [1:3] or [4 5 7]
5. Base UDP Port: The base port of the first UAV's mavlink port. You can also use a port list to specify the port of each UAV
6. RealflyMode: Checked for real uav control, the IP and port will auto increase 1 according to CopterID increase.  
Not checked for RflySim UDP simulation, the IP will not change and the port will +2 for each CopterID.
7. Auto Bind Udp, if enabled, the broadcast data will not be received, otherwise broadcast data will be received.

Input and output char arrays from buf for mavlink encode and decoder

参数 配置地址和端口

Base IP Address or IP list seperated by ','  
192.168.151.105,192.168.151.106

Vehicle number or CopterID list (e.g., [1 3 4])  
[5 6]

Base UDP Port or list  
15501

Sample Time 1/30 0.033333  RealflyMode  Auto Bind Udp

确定(O) 取消(C) 帮助(H) 应用(A)

点击打开

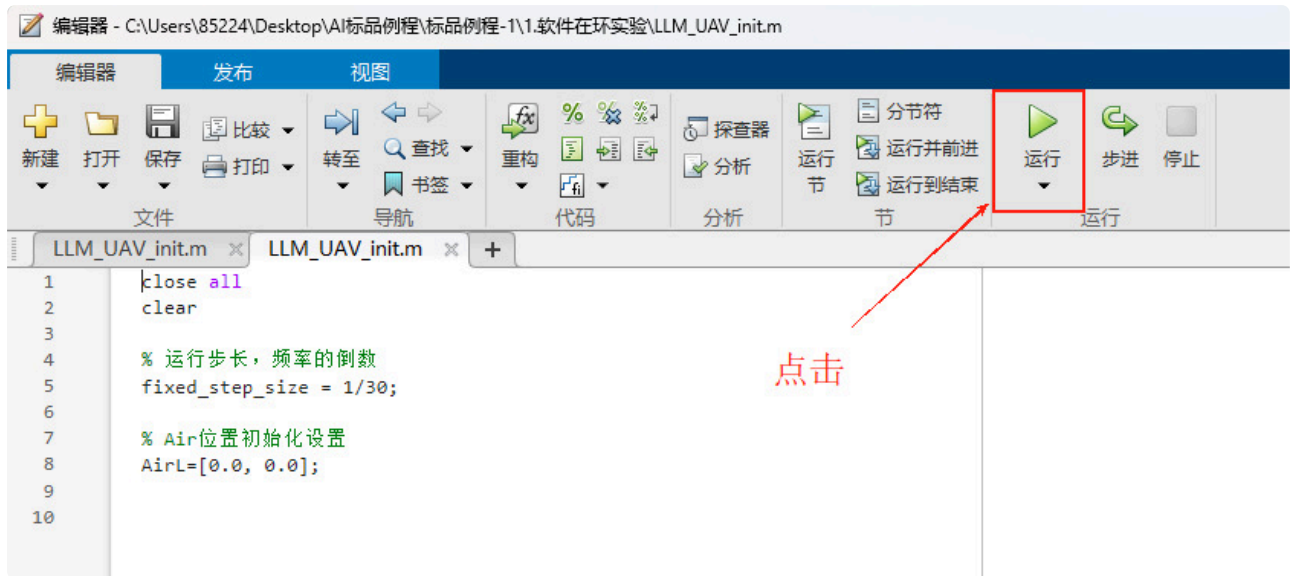
RflyVrpnRecv  
send mavlink vision.  
VehicleNum [5 6]

RflyUdpMavlink  
Mavlink\_Real mode  
for Real fly.  
VehicleNum [5 6]

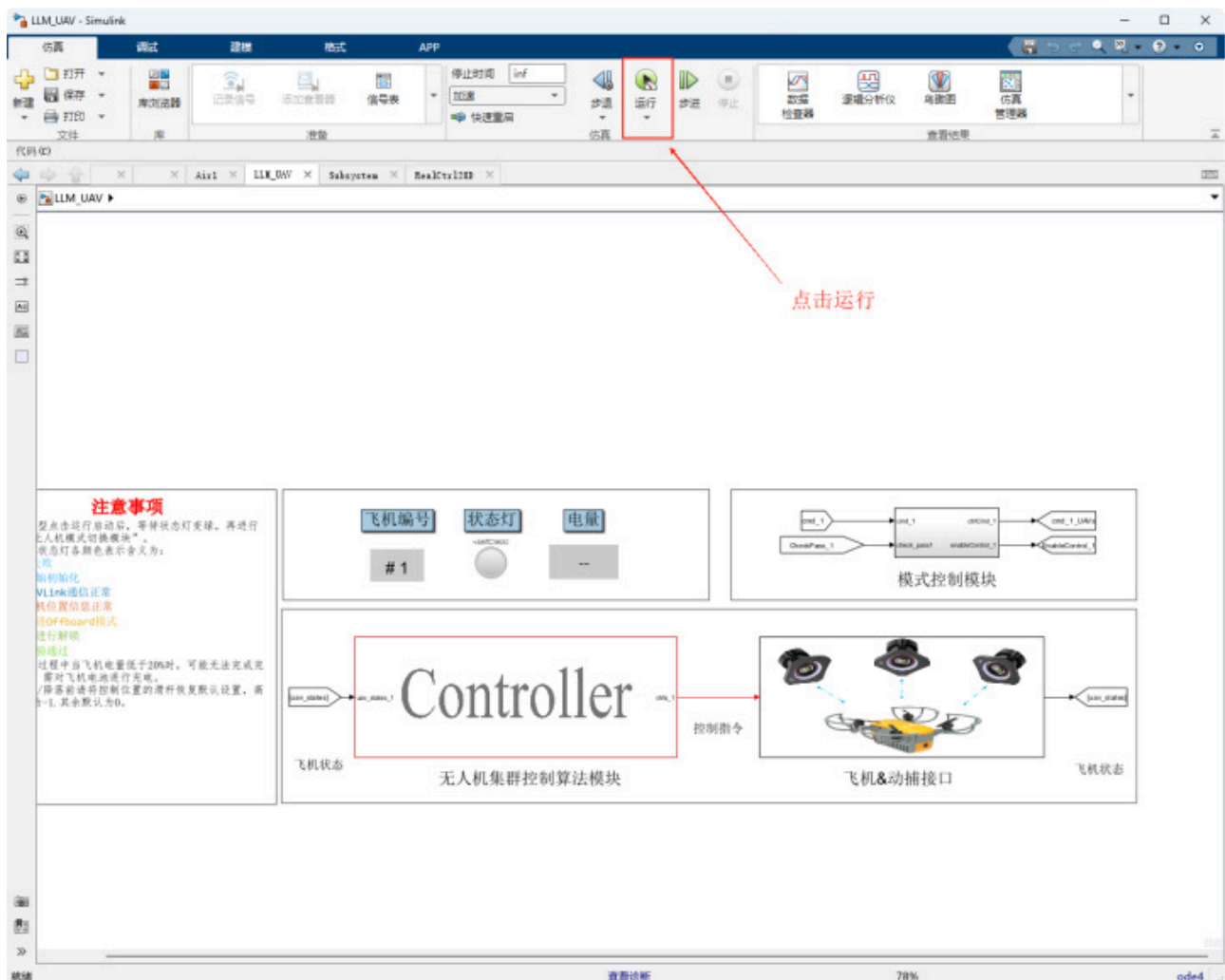
RflyUdpMavlink1

## 4.3 步骤3 运行外部控制程序

在同文件夹下，打开 LLM\_UAV\_init.m 文件，并在matlab中运行，进行参数初始化

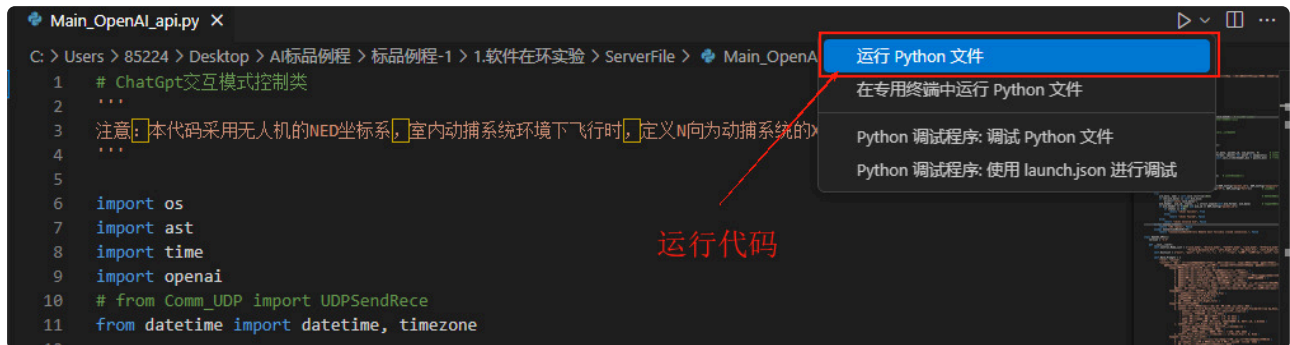


在同文件夹下，打开LLM\_UAV.slx文件，并在matlab中点击运行

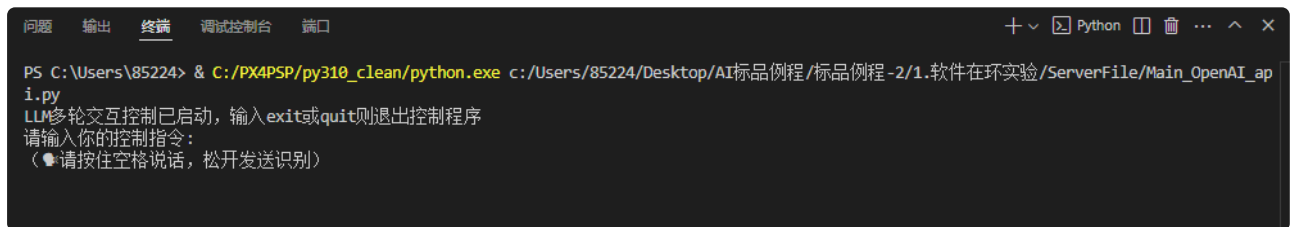


## 4.4 步骤4 启动大模型服务端并发送指令

使用原先配置好python环境的Visual Studio Code打开程序  
./ServerFile/Main\_OpenAI\_api.py; 并运行代码

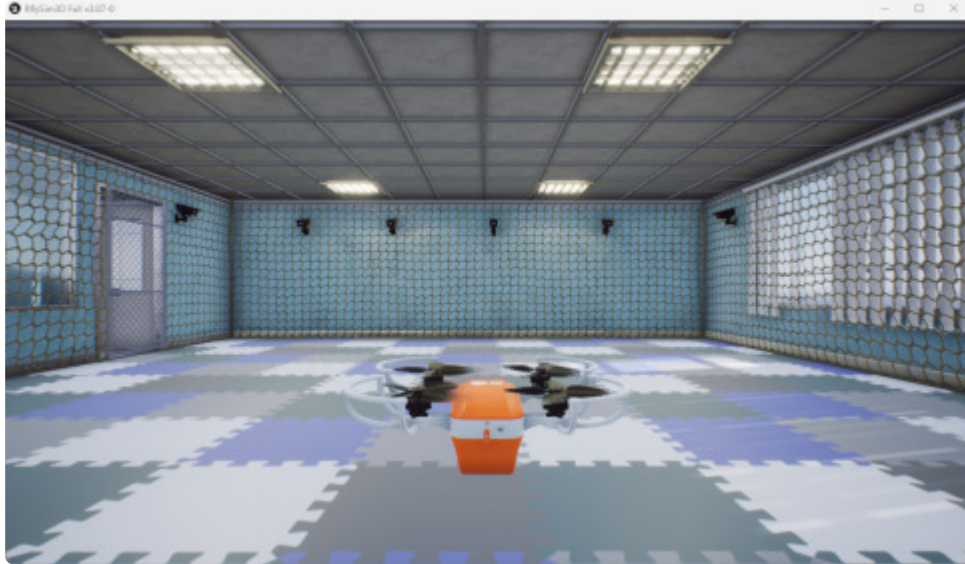


按住空格，说话输入语音指令（请确保电脑有可用的麦克风）。输入的指令如：



- 输入：**\*\*一键起飞\*\***。程序将会依次启动：开启Offboard模式、开启解锁模式、开启起飞模式；最后飞机将解锁并起飞升空
- 输入：**向左飞行1米**。程序将启动Left\_Right\_Fly，并按要求向左飞行1米
- 输入：**\*\*降落\*\***。程序将启动Land\_mode模式，飞机将降落到地面
- 输入：**锁定**。程序将启动Lock\_mode模式，飞机电机将停止转动
- 输入：**解锁**。程序将启动UnLock\_mode模式，飞机电机将解锁。





## 5. 关键知识点

### 5.1 关键知识点1：UDP/MAVLink 通信协议

理解 MAVLink 消息格式、心跳与命令子协议，以及基于 UDP 的点对点传输与重发机制 [mavlink.io](http://mavlink.io) [mavlink.io](http://mavlink.io)。

### 5.2 关键知识点2：二进制数据打包与 CRC 校验

掌握 `struct.pack("<HHBB", ...)` 等打包格式及 `zlib.crc32` 校验方法，确保控制指令在传输过程中的完整性与可靠性 [mavlink.io](http://mavlink.io) [mathworks.com](http://mathworks.com)。

### 5.3 关键知识点3：Simulink 硬件在环（HIL）配置

配置 HIL 模型与真实无人机的通信接口（Local/Remote IP、端口及字节序），并在线运行 Simulink 模型验证控制算法 [mathworks.com](http://mathworks.com) [mathworks.com](http://mathworks.com)。

### 5.4 关键知识点4：动捕定位与坐标系

熟悉 Vicon/OptiTrack 等室内动捕系统与 NED 右手坐标系的对齐方法，确保飞行位置与目标点的一致性，并控制位置误差在可接受范围内 [mdpi.com](http://mdpi.com)。

## 5.5 关键知识点5：飞行精度与轨迹跟踪

分析网络时延、数据解析延时及动捕抖动等误差来源，优化控制参数以提高点到点飞行精度及姿态稳定性 [mdpi.com](http://mdpi.com)。

## 5.6 关键知识点6：RflySim 平台移植

熟悉 RflySim 三维仿真到实机环境的配置差异与接口调整，快速调整模型与通信脚本，保证实验流程一致性 [discuss.px4.io](http://discuss.px4.io)。

# 6. 参考资料

1. [RflySim官方文档](#)
2. [MAVLink协议文档](#)
3. [PX4飞控文档](#)

# 7. 常见问题

## Q1：语音指令响应延迟过大怎么办？

A1：检查网络连接状况，确保WiFi信号稳定。同时检查大模型API调用是否正常，可能需要更换响应更快的服务商。

## Q2：无人机无法正确执行飞行指令？

A2：确认动捕系统定位正常，无人机位置信息准确。检查Simulink模型中通信参数配置是否正确。

## Q3：如何确保飞行安全？

A3：建议先在仿真环境中充分测试后再进行实飞试验，确保有紧急停机措施，并在开阔空间进行测试。

1. <https://rflysim.com/> ↩

2. 推荐配置请见: <https://rflysim.com/> ↩