

多机型多实例自动化测试

1. 实验目的

掌握自动测试平台多机型多实例自动化测试的基本结构和使用流程。

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]；Visual Studio Code。
 - 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://rflysim.com/doc/zh/1/Hardware.html>
- 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\7.RflySimPHM\2.AdvExps\e7_MultiFrameMultiInsExp

- [AutoTest.py](#)：自动测试脚本
- [Python38Run.bat](#)：Python程序执行脚本
- [model](#)：仿真用到的模型和bat脚本
- [data](#)：仿真数据存放处

4. 实验内容或步骤

4.1 步骤1：了解文件性质

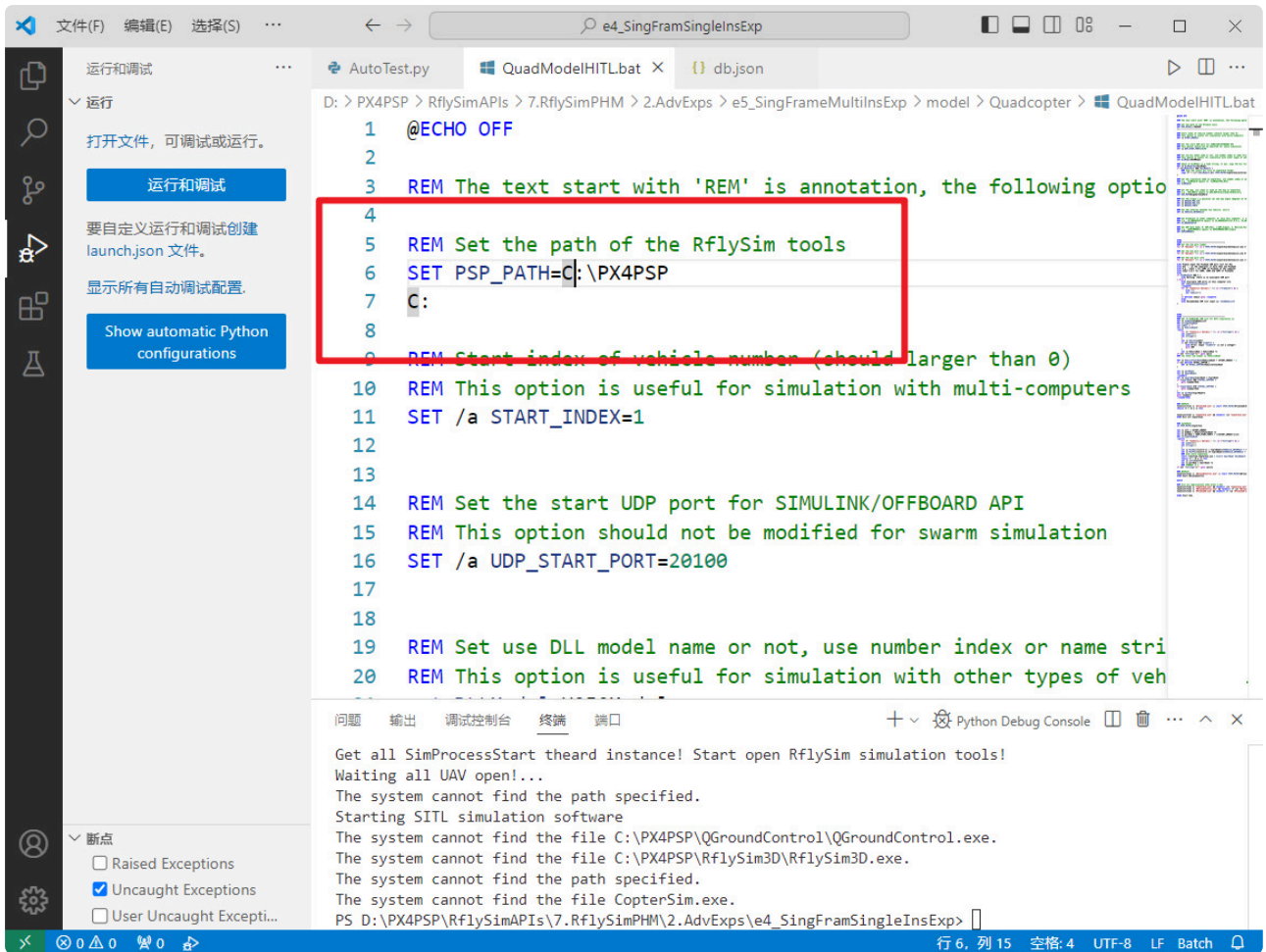
由于RflySimSaT的高度集成性，其只包含了一个主测试入口文件 [AutoTest.py](#) 文件和 db.json，其余的文件均放置在RflySim平台的公共库RflySimSDK中。

打开 [AutoTest.py](#) 文件，可以看到关键配置参数conf和mav,这两个参数通过列表维护不同的机型和实例数量。在本文件中设置为多机单实例模式；其中conf的机架数量为2，实例数量为2。

打开db_Quadcopter.json和db_Fixedwing.json，可以看到在"TEST_CASE"关键字后面的测试用例设置均为"1,2",由于为多机型多实例模式，因此不同机型的实例数均大于1，故json文件应设置为多机模式，用分号即可。此意为分别测试2个固定翼和2个四旋翼的1号和2号测试用例。注：当测试多机型时，不同机型的测试用例个数应相同。

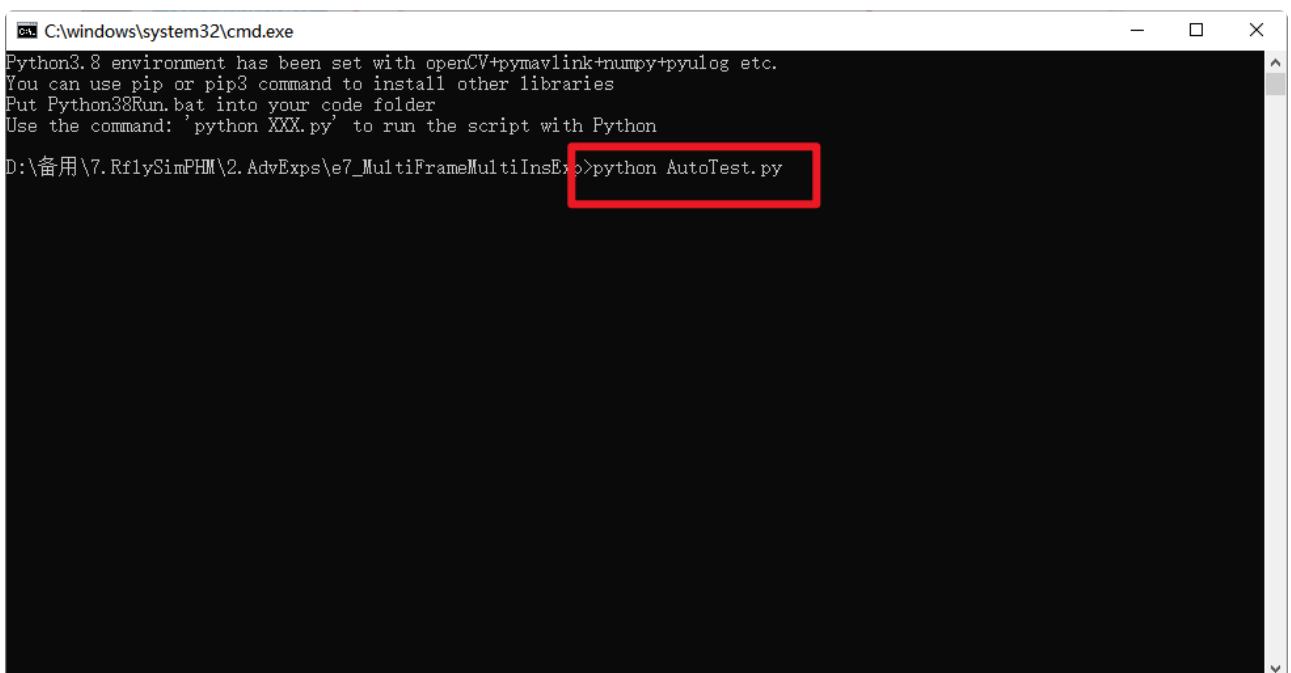
4.2 步骤2：配置仿真参数

根据安装平台路径修改"e6_MultiFrameSingleInsExp\model**.bat"以及"e6_MultiFrameSingleInsExp\model**.bat"。**(是指目录下的所有文件夹都要修改，.bat是指目录下的所有.bat文件都要修改)**



4.3 步骤3：运行python文件

在文件夹下，双击 `Python38Run.bat`，打开集成好的python环境，在该环境下运行 `AutoTest.py` 文件，输入 `python AutoTest.py`，运行 `AutoTest.py`。



可以看到本实验初始了2架四旋翼、2架固定翼，并分别自动化测试了2架四旋翼的测试用例1, 2和2架固定翼的测试用例1, 2，并自动化记录了测试数据，在 data\multi\mInstance\TestCase_1_1\Quadcopter文件夹和 data\multi\sInstance\TestCase_1_1\Fixedwing下。

```
Microsoft Windows [版本 10.0.19045.4651]
(c) Microsoft Corporation. 保留所有权利。

D:\PX4PSP\RflySimAPIs\7.RflySimPHM\2.AdvExps\e7_MultiFrameMultiInsExp>python AutoTest.py
Quadcopter Conf:
  Frame:Quadcopter
  Mode:Multiple machines
  MAV Num:2
  Test Cases:[[1], [2]]

Fixedwing Conf:
  Frame:Fixedwing
  Mode:Multiple machines
  MAV Num:2
  Test Cases:[[2], [1]]

self.mavid-1
mav1 all case: [1]
self.mavid-1
mav2 all case: [2]
self.mavid-1
mav3 all case: [2]
self.mavid-1
mav4 all case: [1]
Waiting all UAV arrive...
hello, mav1
hello, mav2
hello, mav3
hello, mav4
Get all SimProcessStart theard instance! Start open RflySim simulation tools!
```

名称	修改日期	类型	大小
multi	2024/8/9 11:27	文件夹	
TestInfo.html	2024/8/9 11:29	Microsoft Edge...	4 KB
TestResult.html	2024/8/9 11:29	Microsoft Edge...	6 KB
TestInfo.csv	2024/8/9 11:29	Microsoft Excel...	2 KB
TestResultInfo.csv	2024/8/9 11:29	Microsoft Excel...	3 KB

4.4 步骤4：检查数据

自动化记录了该测试用例的测试状态、测试结果等信息，并呈现了可视化结果，可以在 beta\data文件夹下的TestInfo.html和TestResult.html看到。

5. 关键知识点

5.1 关键知识点1：模块导入与路径配置

本实验可以对python脚本进行修改，实现平台的自动测试。

sys 和 os: Python标准库，用于系统相关操作。

PX4MavCtrlV4 和 AutoMavCtrl: 自定义模块，用于与PX4飞行控制器和自动控制环境交

互。

`sys.path.append(os.getcwd())`：确保脚本能够导入当前目录下的模块。

5.2 关键知识点2：自动配置参数

`conf`：一个列表，包含无人机的类型（四旋翼或固定翼）、仿真类型（软件在环SITL或硬件在环HITL）和模拟无人机的数量。

5.3 关键知识点3：无人机控制器实例管理

`mav`：一个列表，创建了多个PX4MavCtrler实例，每个实例代表一个无人机。

5.4 关键知识点4：地图与位置设置

`map`：包含地图名称和无人机的初始位置、偏航角和间隔。

5.5 关键知识点5：环境初始化与监控

`AutoEnv`：使用`AutoMavCtrl.InitMavAutoEnv`函数，初始化无人机的自动控制环境。

`AutoMavCtrl.MavMonitor()`：启动用于监控无人机状态的线程。

对于每架无人机，创建一个`AutoMavCtrl.AutoMavCtrler`实例，并调用`AutoMavLoopStart`方法启动自动控制循环。

`AutoMavCtrl.SimMonitor()`：启动仿真环境的监控。

6. 参考资料

1. [RflySim官方文档](#)
2. [PX4开发者指南](#)
3. [MAVLink协议文档](#)

I 7.常见问题

I Q1: 运行AutoTest.py时出现模块导入错误怎么办?

A1: 确保已正确配置Python环境，并且当前工作目录包含了所需的模块文件。可以尝试添加路径到sys.path中，或检查Python环境中是否安装了所需依赖库。

I Q2: 如何修改测试用例的数量和配置?

A2: 需要修改AutoTest.py中的conf配置参数，以及db_Quadcopter.json和db_Fixedwing.json文件中的"TEST_CASE"关键字后的值。注意不同机型的测试用例个数应相同。

I Q3: 多机型多实例仿真时性能下降明显怎么办?

A3: 多实例仿真会占用较多计算资源，建议适当减少实例数量，或升级计算机硬件配置。也可以调整仿真步长和数据记录频率来优化性能。

1. <https://rflysim.com/> ←

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