
dll 模型仿真初始化设置实验原理

1. 文件目录.....	1
2. 总体说明.....	1
2.1. 修改 bat 脚本实现更灵活的仿真初始设置	1
2.2. 利用 Python 接口自定义仿真初始设置	2
3. 关键功能的实现.....	3
3.1. 通过 Python 接口重置联机模式	3
3.1.1. 关键接口函数	3
3.1.2. 调用示例	3
3.2. 通过 Python 接口重设仿真模式	4
3.2.1. 关键接口函数	4
3.2.2. 调用示例	4
3.3. 通过 Python 接口重设 dll 模型	4
3.3.1. 关键接口函数	4
3.3.2. 调用示例	4
3.4. 通过 bat 脚本为不同载具模型设置初始位姿	4
4. 相关文献.....	4
附加资源.....	5

1. 文件目录

例程目录: [\[安装目录\]\RflySimAPIs\4.RflySimModel\0.ApiExps\14.inCopterData\](#)

文件夹/文件名称	说明
1.RegCopterSim\Readme.pdf	dll 模型仿真初始化设置 python 接口方式实验步骤
2.RunPosAttBat\1.MultiModelCtrl\Readme.pdf	不同 dll 模型仿真初始化设置 bat 脚本方式实验步骤
2.RunPosAttBat\2.UUV\Readme.pdf	
2.RunPosAttBat\3.CopterSimSILNoPX4\Readme.pdf	

2. 总体说明

2.1. 修改 bat 脚本实现更灵活的仿真初始设置

➤ 详细应用场景参考[1]中的[仿真一键启动 bat 脚本](#)

RflySim 平台的软/硬件在环仿真涉及多个程序的协作，都需要进行必要的初始配置，因此提供了 bat 脚本来预先配置各种启动参数，只需通过 bat 脚本来启动仿真就能自动配置所有仿真平台所需的初始参数。

但如果需要修改初始仿真条件，原有的普通软/硬件在环仿真启动 bat 脚本就需要再次编辑 b，重新启动仿真，这不够便捷。

这里提供两种修改后的 bat 脚本，一种可以在启动过程中指定载具（多旋翼、固定翼、无人车、水下潜航器等）的初始位姿状态、另一种可以同时启动多个载具仿真。

注意，这里重设初始位姿状态中也支持了高度偏移设置，即模型不必初始化在地面上。

2.2. IP 地址简介

如 10.100.122.2 就是一个 IP 地址。这个地址被点分隔为四个部分，每个部分 8 个 bit，所以 IP 地址总共是 32 位。

A类	0	网络号（7位）	主机号（24位）
B类	1	0	网络号（14位）主机号（16位）
C类	1	1	0网络号（21位）主机号（8位）
D类	1	1	10多播组号（28位）
E类	1	1	110留待后用（27位）

类别	IP 地址范围	最大主机数	私有 IP 地址范围
A	0.0.0.0 – 127.255.255.255	$16777214 = 2^{24} - 2 = 2^{3 \times 8} - 2$	10.0.0.0 – 10.255.255.255
B	128.0.0.0 – 191.255.255.255	$65534 = 2^{16} - 2$	172.16.0.0 – 172.31.255.255
C	192.0.0.0 – 223.255.255.255	$254 = 2^8 - 2$	192.168.0.0 – 192.168.255.255

C 类地址能包含的最大主机数量只有 254 个。而 B 类地址能包含的最大主机数量又太多。6 万多台机器放在一个网络下面，一般达不到这个规模，闲着的地址就是浪费。

折中的方式叫作无类型域间选路，简称 CIDR。这种方式打破了原来设计的几类地址的做法，将 32 位的 IP 地址一分为二，前面是网络号，后面是主机号。10.100.122.2/24，这种地址表示形式，就是 CIDR。后面 24 的意思是，32 位中，前 24 位是网络号，后 8 位是主机号。伴随着 CIDR 存在的，一个是广播地址，10.100.122.255。如果发送这个地址，所有 10.100.122 网络里面的机器都可以收到。另一个是子网掩码，255.255.255.0。将子网掩码和 IP 地址进行 AND 计算。前面三个 255，转成二进制所有位都是 1。1 和任何数值取 AND，都是原来数值，因而前三个数不变，为 10.100.122。后面一个 0，转换成二进制是 0，0 和任何数值取 AND，都是 0，因而最后一个数变为 0，合起来就是 10.100.122.0。这就是网络号。将子网掩码和 IP 地址按位计算 AND，就可得到网络号。

2.3. 利用 Python 接口自定义仿真初始设置

- 详细的初始化参数设置 python 接口参考[1] [仿真控制接口 ReqCopterSim.py](#)

通过调用 Python 接口，在仿真过程中重置初始化参数，可重置参数包括：

1) 是否联机

效果等价于在 bat 脚本中 `SET IS_BROADCAST=0 或 1`

IS_BROADCAST=0 表示只对本机有效，IS_BROADCAST=1 表示广播到局域网所有计算机。

2) 指定 CopterSim 回传数据 IP

同样等价于 bat 脚本中 `SET IS_BROADCAST= 目标 IP 地址，例如 192.168.3.1`

在 IP 模式下，IS_BROADCAST=0 相当于 IS_BROADCAST=127.0.0.1（只对本机有效），而 IS_BROADCAST=1 相当于 IS_BROADCAST=255.255.255.255（广播到所有网络设备）。

也可以使用逗号或分号作为分隔符来指定一个 IP 列表，以发送到具体的 IP 地址，例如：127.0.0.1,192.168.1.4,192.168.1.5。

3) 载具位置：北东地坐标系下的 x、y、z

4) 载具姿态：滚转、俯仰、偏航

5) UDP 通信模式

RflySim 平台的不同通信模式主要分为如下几种（可在 CopterSim 的 UI 中进行配置，其中 UDP 是传输层协议、Mavlink 是应用层协议）。

➤ UDP_Full:

Python 传输完整的 UDP 数据给 CopterSim，传输数据量小；CopterSim 收到数

据后，再转换为 Mavlink 后传输给 PX4 飞控；适合中小规模集群（数量小于 10）仿真。

➤ **UDP_Simple:**

数据包大小与发送频率比 UDP_Full 模式小；适合大规模集群仿真，无人机数量小于 100。

➤ **Mavlink_Full:**

Python 直接发送 MAVLink 消息给 CopterSim，再转发给 PX4，数据量较大适合单机控制；适合单机或少量飞机仿真，无人机数量小于 4；

➤ **Mavlink_Simple:**

屏蔽部分 MAVLink 消息包，并降低数据频率，发送数据量比 MAVLink_Full 小很多，适合多机集群控制；适合小规模集群仿真，无人机数量小于 8。

➤ **Mavlink_NoSend:**

模式下 CopterSim 不会向外发送 MAVLink 数据，此模式需要配合硬件在环仿真+数传串口通信，通过有线方式传输 MAVLink，此模式局域网内数据量最小，适合分布式视觉硬件在环仿真，无人机数量不限制

➤ **Mavlink_NoGPS:**

模式下 CopterSim 不会向外发送 MAVLink 数据和 GPS 数据。

- 6) 通过 DLL 模型名称重设 dll 模型
- 7) 通过 DLL 模型序号重设 dll 模型
- 8) 通过三维场景名称重设三维场景
- 9) 通过三维场景序号重设三维场景

3. 关键功能的实现

3.1. 通过 Python 接口重置联机模式（是否局域网联机）

3.1.1. 关键接口函数

- 详细的初始化参数设置 python 接口参考[1] [仿真控制接口 ReqCopterSim.py](#)

3.1.2. 调用示例

创建一个 CopterSim 状态获取实例，并监听 2s 钟，获取当前所有 CopterSim 列表数据

```
req = ReqCopterSim.ReqCopterSim()
```

获取 ID 和 IP 列表

```
IPList=req.getSimIpList()
```

获取指定 ID 的电脑 IP 地址

```
IP = req.getSimIpID(1)
```

下面展示，如何使用本接口，不需要知道目标电脑 IP 的情况下，能够连上远程电脑的 CopterSim

```
CopterID=1 # 计划仿真的飞机 ID
```

获取目标电脑 IP，并且配置 CopterSim 回传数据到本电脑

获取到指定 CopterID 的 CopterSim 所在电脑的 IP

```
TargetIP = req.getSimIpID(CopterID)
```

请求目标 CopterSim 将数据返回到本电脑

```
req.sendReSimIP(CopterID)
```

通过本接口，可以不用再去 bat 脚本里面填写 IP 地址了

注：使用本模式，也不需要再在 bat 脚本中开启联机模式

3.2. 通过 Python 接口重设 UDP 通信模式

3.2.1. 关键接口函数

➤ 详细的初始化参数设置 python 接口参考[1] [仿真控制接口 ReqCopterSim.py](#)

3.2.2. 调用示例

创建一个 CopterSim 状态获取实例，并监听 2s 钟，获取当前所有 CopterSim 列表数据

```
req = ReqCopterSim.ReqCopterSim()
```

获取 ID 和 IP 列表

```
IPList=req.getSimIpList()
```

获取指定 ID 的电脑 IP 地址

```
IP = req.getSimIpID(1)
```

通过本接口，可以强制修改 CopterSim 的 UDP_Mode，这里只有要时，才发送

```
new_udp_mode=2
```

```
req.sendReSimUdpMode(CopterID,new_udp_mode)
```

注，SITLRun 默认模式是 0 模式，也就是 UDP_Full，这里我们强制修改为 2 模式（M AVLink_Full 通信模式）

3.3. 通过 Python 接口重设 dll 模型

3.3.1. 关键接口函数

➤ 详细的初始化参数设置 python 接口参考[1] [仿真控制接口 ReqCopterSim.py](#)

3.3.2. 调用示例

将指定 CopterSim 的 dll 模型换成六旋翼

```
req.sendReSimDllName(CopterID,'HexarotorModelCTRL')
```

3.4. 通过 bat 脚本为不同载具模型设置初始位姿

```
SET /P PosXStr=Please enter the PosX (m) list: 存储用户输入的 X 坐标列表，单位是米。
```

```
SET /P PosYStr=Please enter the PosY (m) list: 存储用户输入的 Y 坐标列表，单位是米。
```

```
SET /P YawStr=Please enter the Yaw (degree) list: 存储用户输入的偏航角列表，单位是度。
```

```
SET /P AltStr=Please enter the PosZ (m) list: 存储用户输入的高度（Z 坐标）列表，单位是米。
```

```
SET /P RollStr=Please enter the Roll (degree) list: 存储用户输入的滚转角列表，单位是度。
```

```
SET /P PitchStr=Please enter the Pitch (degree) list: 存储用户输入的俯仰角列表，单位是度。
```

4. 相关文献

[1]. [..\API.pdf](#)

[2]. [..\..\6.RflySimExtCtrl\API.pdf](#)

附加资源

官方文档: RflySim 官方文档: <https://rflysim.com/doc/zh/>

社区交流: 加入 RflySim 技术交流群: 951534390

