

《智能无人机路径规划仿真系统》

使用说明书

目录

- 第一章 系统简介 1
- 第二章 系统安装说明 2
 - 1. 安装要求..... 2
 - 1.1 硬件环境 2
 - 1.2 软件环境 3
 - 2. 安装方法..... 3
- 第三章 系统使用说明 7
 - 1. 虚拟飞行..... 7
 - 1.1 启动 FlightGear 飞行模拟器..... 7
 - 2. 选择作战地图 9
 - 3. 作战控制台 10
 - 3.1 启动编程环境——作战平台..... 10
 - 3.2 仅加载 Mavproxy 控制台 10
 - 3.3 加载 Mavproxy 控制台与战场俯视图..... 11
 - 3.4 操作仿真系统执行一次飞行任务 15
 - 3.5 智能控制 18
 - 4. 地面站 19

第一章 系统简介

智能无人机路径规划仿真系统是一个具有操作控制精细、平台整合性强、全方向模型建立与应用自动化特点的软件。该系统的核心功能是通过仿真平台规划无人机航线，并进行验证输出，数据可导入真实无人机，使其按照规定路线精准抵达战场任一位置，支持多人多设备编队联合行动。系统以开源无人机仿真平台 SITL 为支撑，通过 FlightGear 渲染真实战场环境，集成了动力学模型建模、二维俯视、三维模拟、脚本控制、地面站监控、数据处理等功能，此外，仿真系统支持加载多种全球地图，模拟各大重点地域的三维环境，可应用于全球各处遥感监测的场景中。

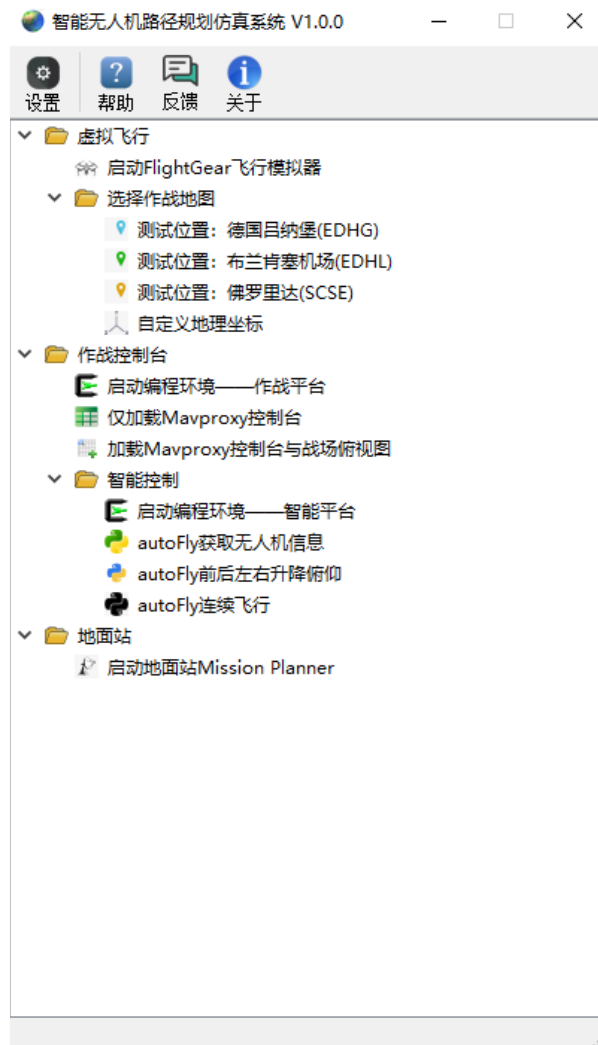


图 1-1 智能无人机路径规划仿真系统主界面

程序功能模块如图 1-2 所示，为了更加便于实际应用，软件环境从 Linux 重新编译到 Windows 系统下，当前版本除仿真三维环境模块外均使用 Python 语言编写，程序易维护、易修改。但 Python 本质仍为脚本语言，无法较好的渲染三维环境，为使仿真更加出色，三维仿真模块单独使用 C++ 语言进行编写，而后编译到 Python 环境中运行。下面是软件结构图（部分插件还未开发）：

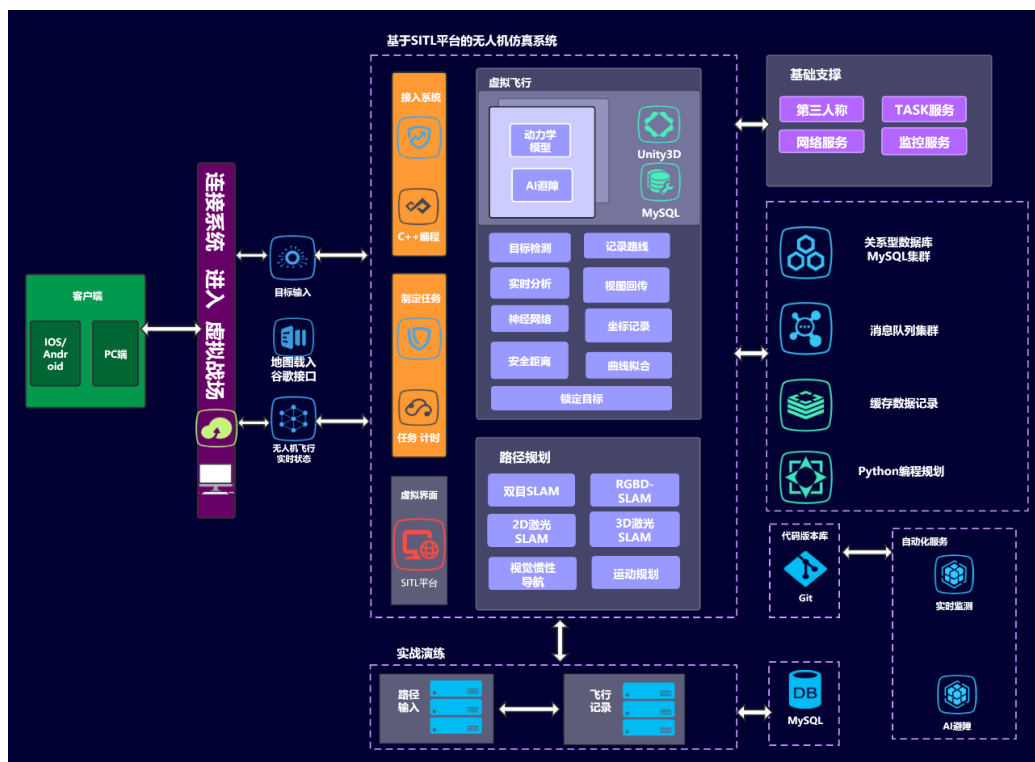


图 1-2 智能无人机路径规划仿真系统结构模块

第二章 系统安装说明

《智能无人机路径规划仿真系统》软件界面采用 PyQt5 开发，由于 Python 为非编译性语言，因此，Python 源代码的主体并未被编译为可执行程序，仅将程序的启动项编译为了二进制可执行程序。该系统的安装要求及安装方法详见下文。

1. 安装要求

1.1 硬件环境

处理器：主流的 Intel 或 AMD 处理器

内存容量: $\geq 4\text{GB}$

硬盘容量：≥2GB

1.2 软件环境

操作系统：Windows7 x64、Windows10 x64、Linux x64

2. 安装方法

本系统涉及到 Linux 系统与 Windows 系统的移植，多种可视化界面，需重新编译操作平台，因此环境搭建需要按照安装方法逐步执行，且软件路径选择默认较好，可避免很多路径麻烦。相较于 Linux 系统，在 Windows 系统中安装稍麻烦些，这里就演示在 64 位 Windows 系统中安装《智能无人机路径规划仿真系统》。软件安装包如图 2-1 所示：

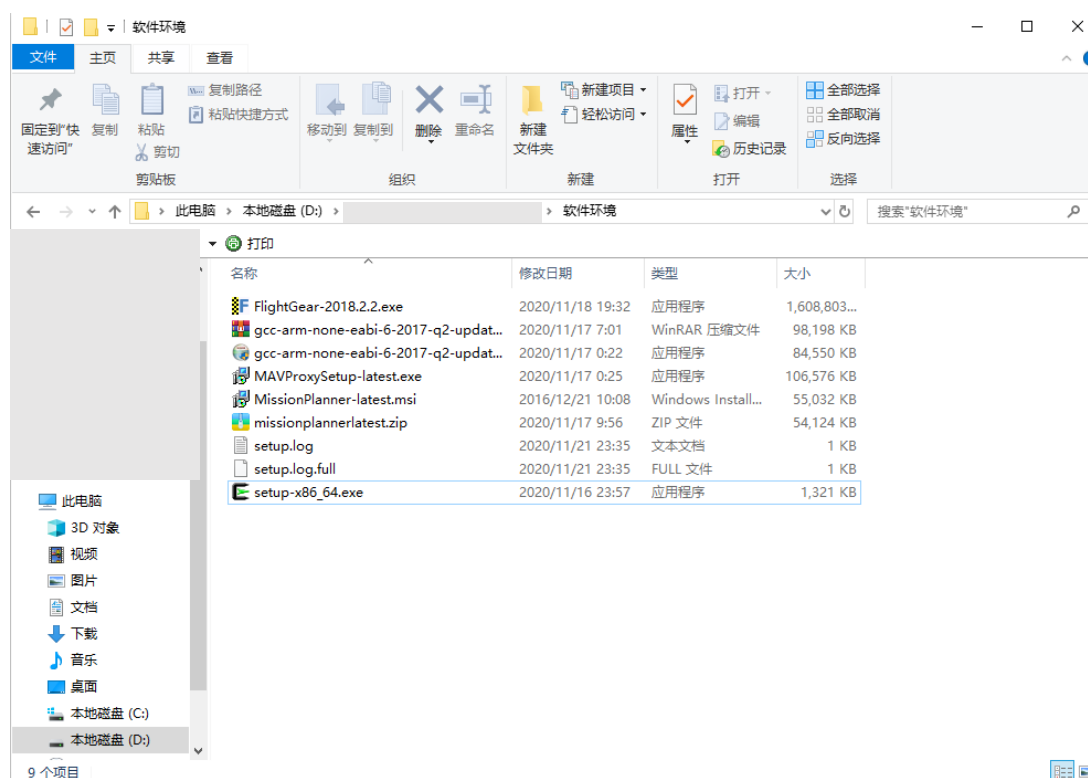


图 2-1 软件安装包

首先安装 Cygwin64 终端，将安装包放入相应位置，然后打开 cmd 终端（Windows 系统终端），输入：

```
setup-x86_64.exe -P autoconf, automake, ccache, gcc-g++, git, libtool, make, gawk, libexpat-devel, libxml2-devel, python36, python36-future, python36-1xml, python36-pip, libxslt-devel, python36-devel, procps-ng, zip, gdb, ddd
```

而后在添加 URL 项中添加国内开源镜像 <http://mirrors.163.com/cygwin/>，完成 Cygwin64 安装。如图 2-2 所示：

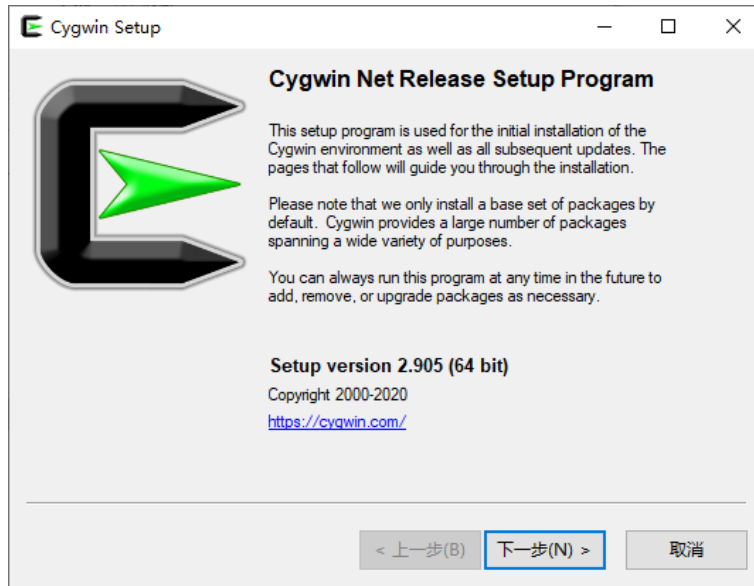


图 2-2 安装 Cygwin64

安装 GCC 软件，一路默认选项即可，直到最后一步，要选上 “Add path to environment variable”。

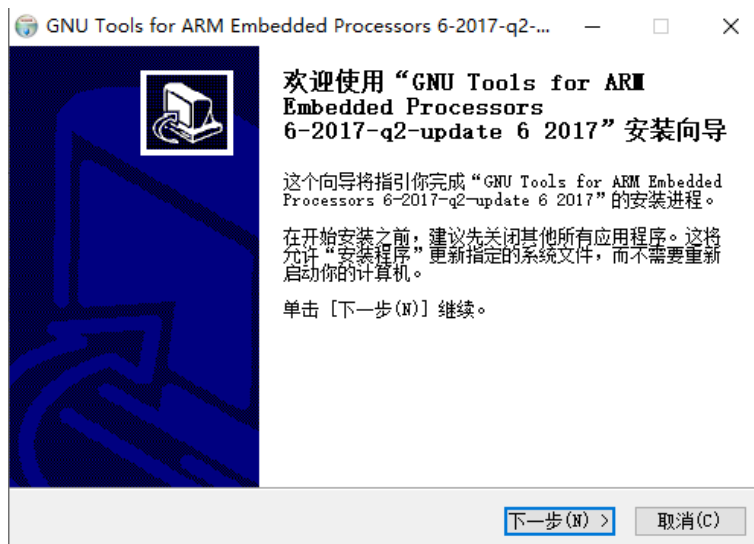


图 2-3 安装 GCC 软件

安装 Mavproxy 控制台，启动后一路选择默认即可。

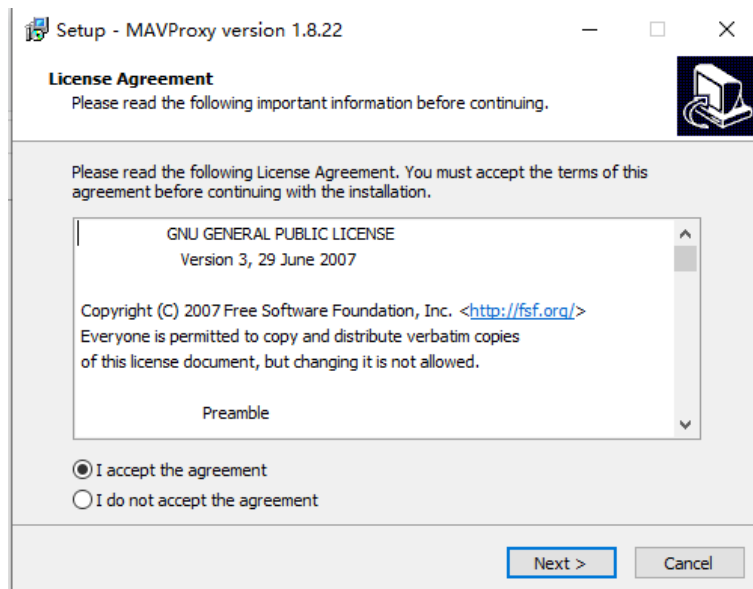


图 2-4 安装 Mavproxy 控制台

接下来配置 Python 编程环境，打开 Cygwin 后关闭，以生成初始化配置文件 initialisation files，再次打开 Cygwin，输入以下代码：

```
ln -s /usr/bin/python3.6 /usr/bin/python
```

```
ln -s /usr/bin/pip3.6 /usr/bin/pip
```

```
pip install empy pyserial pymavlink
```

```
ling@LAPTOP-5VSBCLQP ~
$ pip install empy pyserial pymavlink
Collecting empy
  Using cached empy-3.3.4.tar.gz (62 kB)
Collecting pyserial
  Using cached pyserial-3.4-py2.py3-none-any.whl (193 kB)
Collecting pymavlink
  Downloading pymavlink-2.4.10.tar.gz (4.0 MB)
    | 4.0 MB 51 kB/s
Requirement already satisfied: future in /usr/lib/python3.6/site-packages (from
pymavlink) (0.18.2)
Collecting lxml
  Downloading lxml-4.5.2.tar.gz (4.5 MB)
    | 4.5 MB 46 kB/s
Using legacy 'setup.py install' for empy, since package 'wheel' is not installed
.
Using legacy 'setup.py install' for pymavlink, since package 'wheel' is not inst
alled.
Using legacy 'setup.py install' for lxml, since package 'wheel' is not installed
.
Installing collected packages: empy, pyserial, lxml, pymavlink
  Running setup.py install for empy ... done
```

图 2-5 配置 Python 环境，过程可能较卡，多次安装即可

在打开的 cygwin 窗口下载 ardupilot 飞控源码，输入以下命令：

```
git clone https://github.com/ardupilot/ardupilot.git
```

```
cd ardupilot
```

```
git submodule update --init --recursive
```

而后编译和运行下载的 ardupilot 源码控件，进入 cygwin 的根目录，找到文件“.bashrc”，如我的目录为：“D:\cygwin64\home\Lenovo”。在“.bashrc”末尾添加：

```
export PATH = "$PATH:$HOME/ardupilot/Tools/autotest/"
```

编译 CubeBlack, copter, 仍在 cygwin 窗口下输入：

```
cd ardupilot
```

```
./waf configure --board CubeBlack
```

```
./waf copter
```

将以下四处路径加入系统环境变量 Path，方便软件调用（路径需按照自身实际），如图 2-6 所示：

Mintty: C:\cygwin64\bin

Mission Planner: C:\Program Files (x86)\Mission Planner\

Mavproxy: C:\cygwin64\home\Lenovo\ardupilot\ArduCopter

QT_QPA_PLATFORM_PLUGIN_PATH:

D:\智能无人机路径规划仿真系统\python-3.7.5-embed-amd64\Lib\site-packages

\PyQt5\Qt\plugins

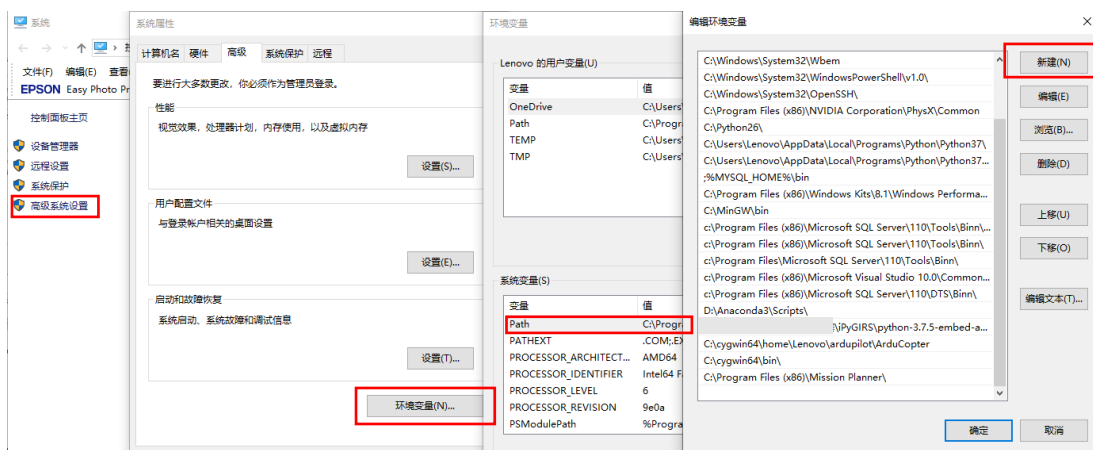


图 2-6 配置系统环境变量

命令执行完毕、路径添加好后，意味着《智能无人机路径规划仿真系统》的运行环境配置完成！

最后解压软件主包，如图 2-7 所示，解压“智能无人机路径规划仿真系统.zip”文件到自定义目录，即可得到如图 2-8 所示的程序文件，点击智能无人机路径规划仿真系统.bat 即可运行程序！

图 2-7 智能无人机路径规划仿真系统安装文件

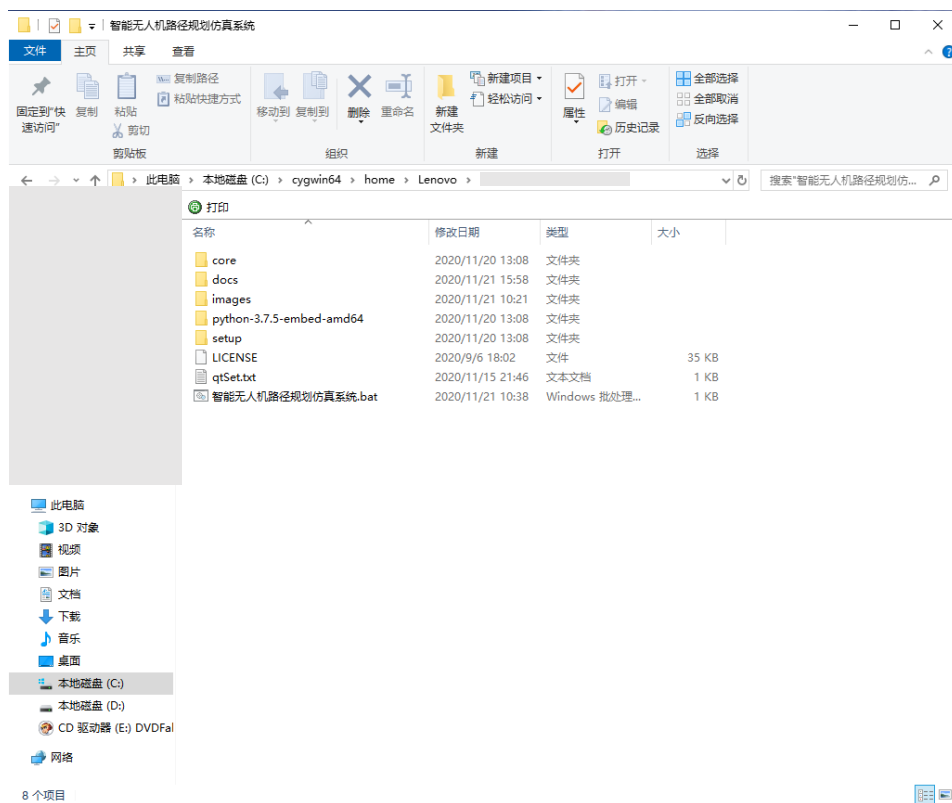


图 2-8 智能无人机路径规划仿真系统程序文件

该系统安装过程的注意事项如下：

- (1) 为后期维护方便，解压的目标目录尽量不要导入其他文件。
- (2) 为保证程序的正确运行，请勿更改解压后的文件（包括文件名、文件的相对路径）。

第三章 系统使用说明

1. 虚拟飞行

1.1 启动 FlightGear 飞行模拟器

智能无人机路径规划仿真系统是建立在信息流在各模块之间相互传递的基础之上。其第一步是加载飞行模拟器来模拟战场三维环境。模拟器允许使用者以第一人称或第三人称角度观察无人机的实时运动状态。双击启动“FlightGear 飞行模拟器”，程序将自动运行，进入默认地图：



3-1-1 FlightGear 飞行模拟器运行界面

FlightGear 飞行模拟器的运行界面如图 3-1-1 所示，该程序的参数说明如表 3-1-1 所示：

表 3-1-1 FlightGear 飞行模拟器内置工具使用说明

工具	使用说明
Scenery Download	该工具用以指定是否在运行期间下载地图，默认地图为美国旧金山机场，勾选后，可自动下载全球各大场所的三维环境，但速度较慢。
View	该工具的下众多选项可切换无人机视角，更改无人机、环境视野状态，有时为了使程序运行流畅，可以通过参数 View Options 减少一些不必要的场景出现。
Location	该工具及下方参数可选择无人机的地理位置，通过输入纬度、经度可将无人机放置于全球任一位置，还支持各大场所的切换，一般以机场、田野、农场等空旷场地为主。

Autopilot	该工具用以更改机场场景的各项设置，也对刷新更改后的场景，可自定义环境内各点航线，并进行统一管理。
Environment	该工具是指更改场景的天气，昼夜时间，树木数量以及风速、风向，在对实际环境模拟中起到很大作用。
Equipment	该工具用以无人机增添机上地图、航向标、GPS 定位、航向仪等其他飞行装备，第一人称使用时非常有模拟飞行驾驶员的感觉。
AI	该工具用以动态模拟环境内其他仿生物体，如客机、其他无人机、敌人、热气球等，增加场景的真实性。
Debug	该工具用以观察程序运行状态，如程序出现卡顿、崩溃等现象，该工具将记录错误瞬间，以备后期分析所用，还可分析正常运行下程序各参数的状态。

2. 选择作战地图

“选择作战地图”内设置了三处位置，包括其对应三维环境、经纬坐标，可供快速使用，它们分别是德国吕纳堡(EDHG)、布兰肯塞机场(EDHL)、佛罗里达(SCSE)。此外，还可直接输入纬度、经度、无人机放置高度、无人机朝向来精准启动模型。后台将输出使用者选择信息，运行界面如图 3-2-1 所示：

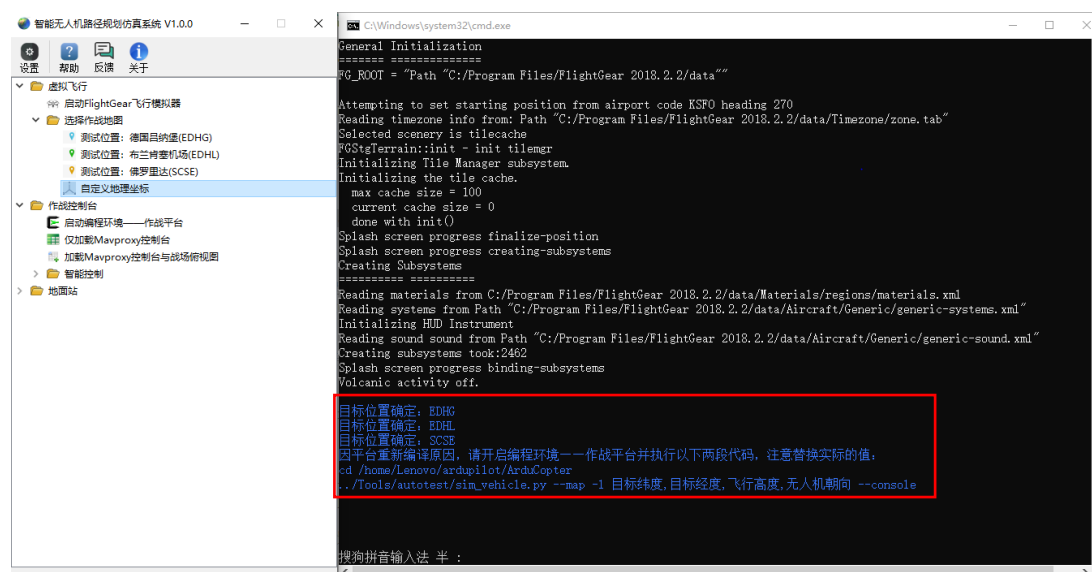


图 3-2-1 选择作战地图程序运行界面

由于平台重新编译原因，在选择“自定义地理坐标”时，无法自动化执行程序命令，需要打开编程环境输入代码，具体操作步骤已经打印在后台上。

3. 作战控制台

“作战控制台”是《智能无人机路径规划仿真系统》系统的核心模块，主要实现了遥感无人机模拟控制的功能，按照“启动编程环境”、“加载 Mavproxy 控制台”、“战场俯视图”、“智能控制”四个步骤对无人机进行多方位的控制。此外，该工具箱内还包含了脚本执行的功能，可自己开发无人机运动脚本，实现对无人机的智能控制。

3.1 启动编程环境——作战平台

根据上文所述，由于平台重新编译原因，程序从 Linux 系统移接到 Windows 系统后，终究还是存在不兼容的现象。部分启动代码仍需在 Linux 编程环境中执行，好在 Windows 系统在开发过程中自带了 mintty 程序，可以模拟 Linux 编程环境。我选择了 Cygwin64 Terminal 程序，是 mintty 的高配版本，相比之下 Cygwin64 Terminal 性能更好，功能更全，更加适合我们项目的启动。双击运行后界面如下所示：

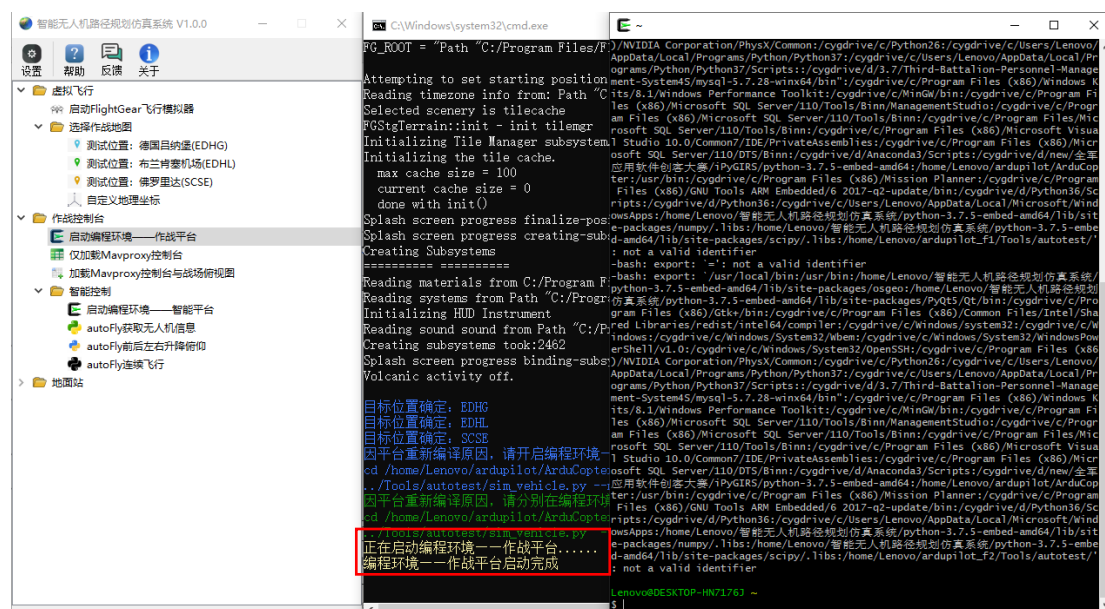


图 3-3-1 Cygwin64 Terminal 运行界面

3.2 仅加载 Mavproxy 控制台

“作战平台”程序启动后，选择“仅加载 Mavproxy 控制台”或“加载 Mavproxy 控制台与战场俯视图”，针对无人机的控制，是通过 Mavproxy 控制台实现的，战场俯视图可以让我们更好的观察无人机的运行状态，还可以对无人机的航向进行可视化规划，建议选择使用“加载 Mavproxy 控制台与战场俯视图”。

3.3 加载 Mavproxy 控制台与战场俯视图

如 3.2 所述,在选择两者之一后,程序后台将高亮显示向编程环境——作战的输入命令,复制粘贴到编程环境即可。

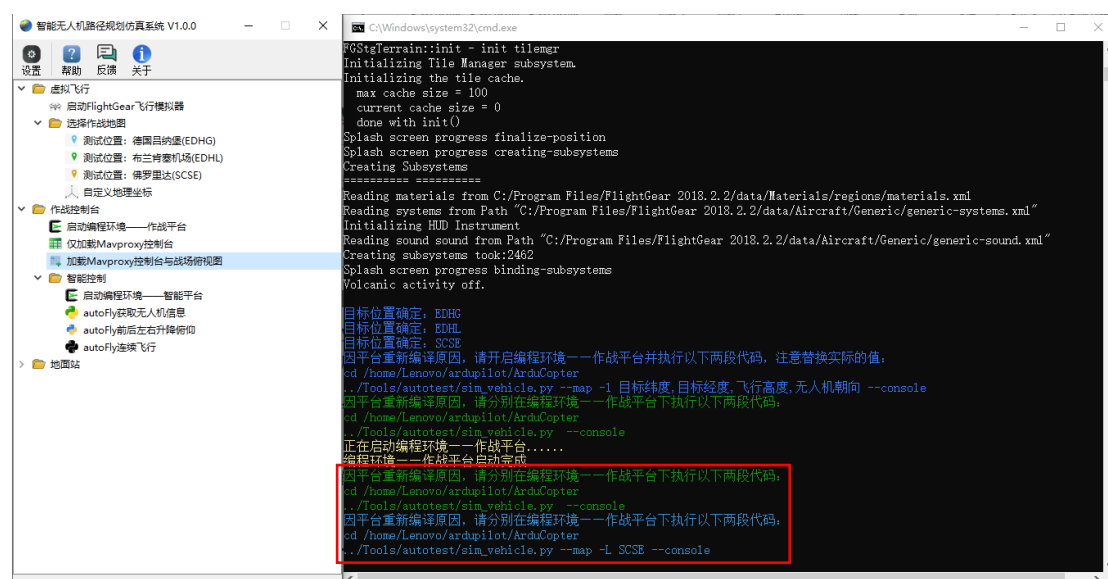


图 3-3-2 选择后后台提示输入代码



图 3-3-4 按照后台提示向作战平台输入代码并执行

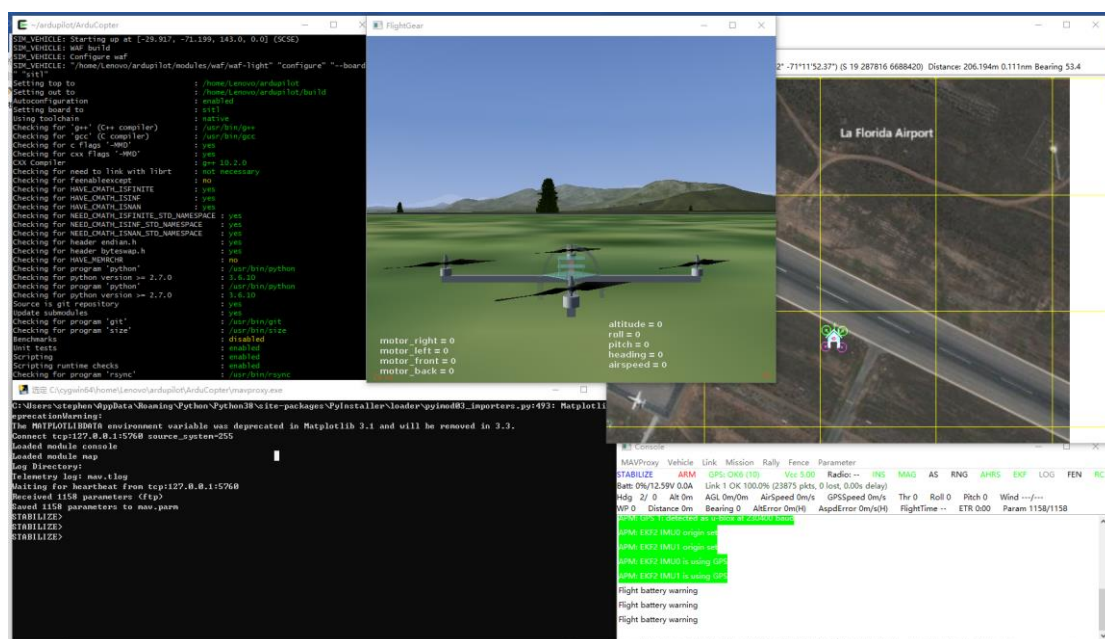


图 3-3-5 程序执行后加载出仿真控制界面

在作战平台执行后台提示的程序后，将启动仿真系统并加载 SITL 飞行控件，此时 FlightGear 飞行模拟器将刷新到我们指定地图位置，而后打开 Mavproxy 控制台，无人机状态终端，作战俯视图。

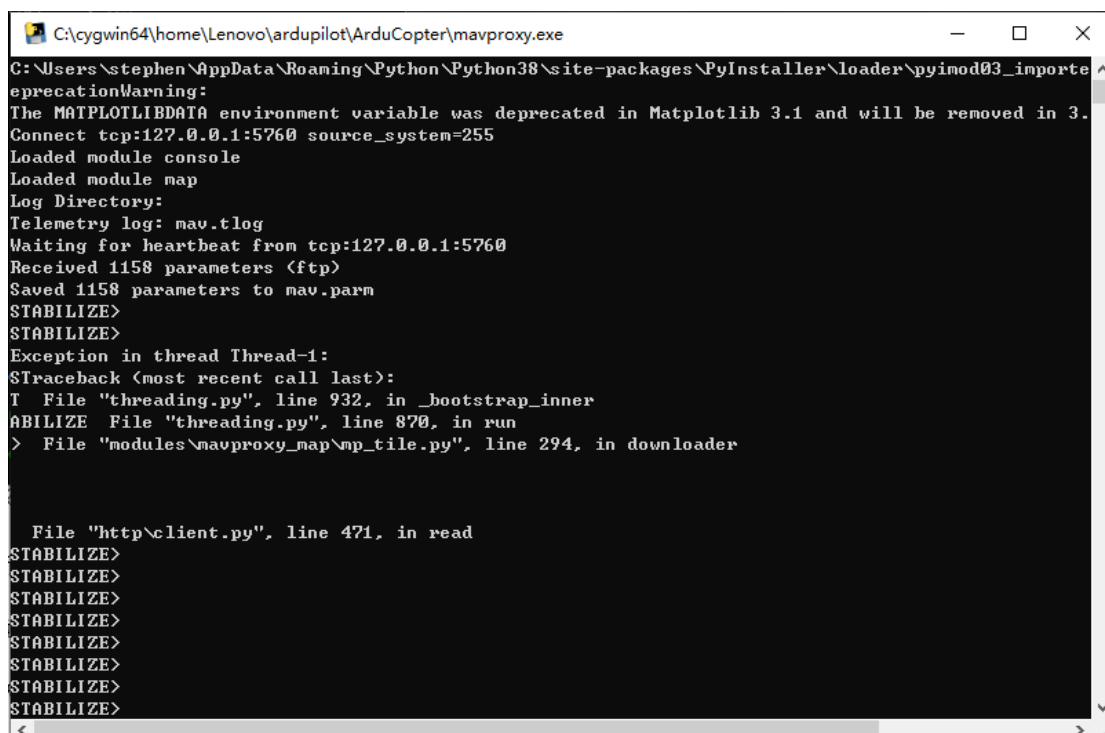


图 3-3-6 Mavproxy 控制台界面

Mavproxy 控制台运行界面如图 3-3-6 所示，部分较实用参数作用及设置说明如下：

表 3-3-1 Mavproxy 控制台参数使用说明

参数	使用说明
Mode	该参数用以指定无人机状态，STABILIZE 为稳定状态，此时无人机保持稳定不动，各项设备正常运转；GUIDED 为控制状态，此状态允许我们对无人机进行操作，本系统的大量实验也是在此状态完成的，一般按照解锁电机，原地起飞一定高度，执行任务的顺序驾驶无人机；AUTO 为自动状态，在设置好航点群后，可切换到此模式使无人机自动执行飞行计划；RET 为返回状态，当无人机出于此状态后，将从当前位置返回基地。
Arm	该参数用于对无人机上电机设备进行操控，无人机在起飞之前必须要电机进行解锁，在降落后长时间不运行无人机时会自动锁定电机，且该参数还实时显示了无人机供电情况，对无人机航行有很大影响。
Takeoff	该参数用以指定起飞高度，并完成无人机起飞动作，根据实际情况，无人机的起飞高度与风速有关，即影响无人机空速。
Wp	该参数用以指定无人机飞行路径中的各个航点，可输入经纬度来精确无人机航线，也可通过可视化界面确定，对任务的完成起到很大帮助。
Param	Param 参数用于指定各项命令所需的参数值，可设置画圆飞行的半径、弧线飞行的内径以及飞行时间等。
Out	Out 参数指定系统对外连接，可通过该命令开辟端口，连接地面站控制软件 MissionPlanner，外联真实无人机设备等。

图 3-3-7 演示了向 Mavproxy 控制台写入命令切换 GUIDED 模式，解锁无人机电机并起飞。

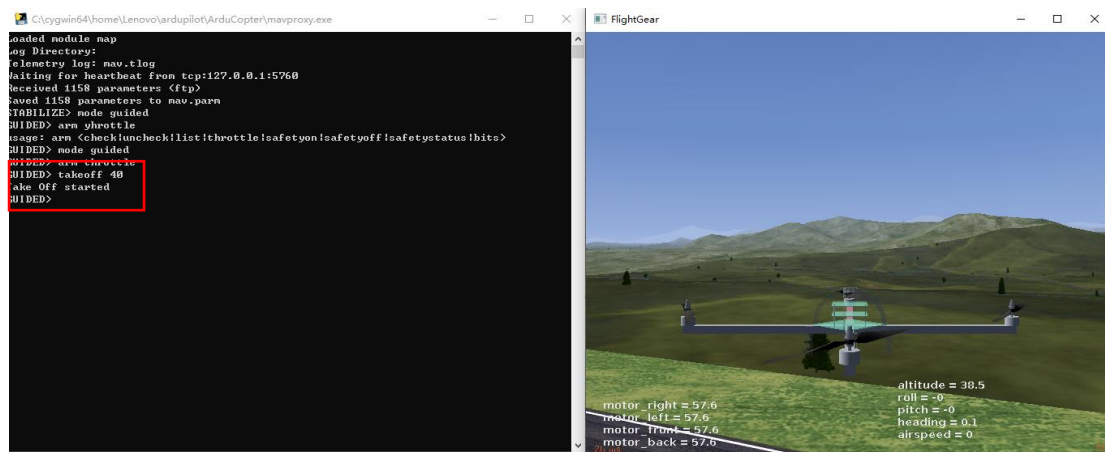


图 3-3-7 Mavproxy 控制无人机起飞

图 3-3-8 演示了向 Mavproxy 控制台写入命令后，无人机状态终端显示了无人机状态的改变，以及命令的执行情况。

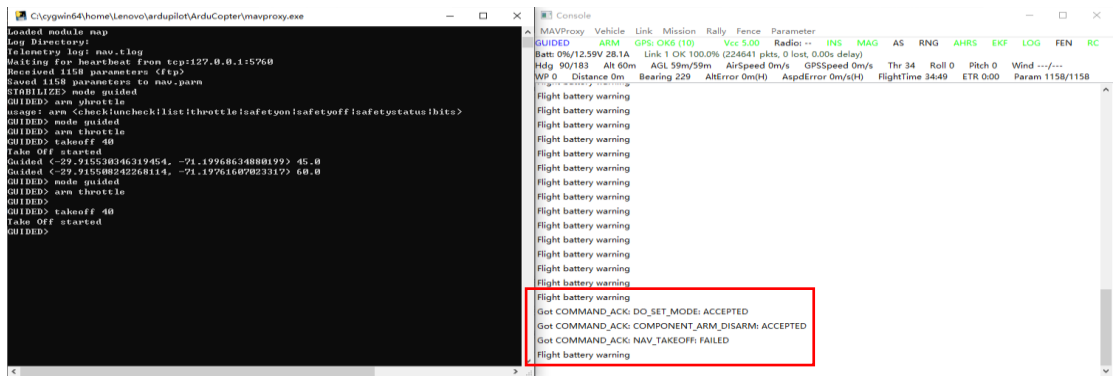


图 3-3-8 无人机状态控制台显示命令的接收情况

图 3-3-9 演示了通过在 Mavproxy 控制台下达飞行命令后，从作战俯视图及三维图像上无人机正在执行飞行任务的状态。

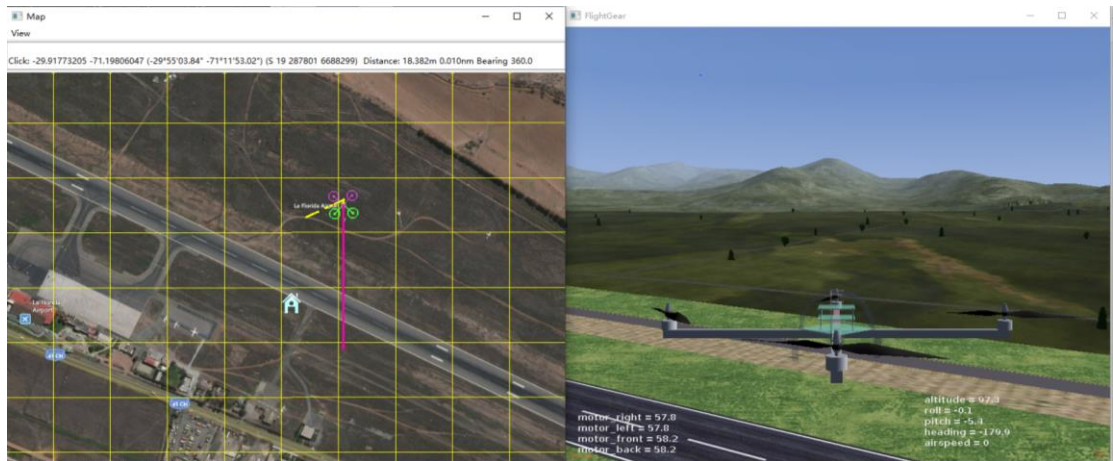


图 3-3-9 无人机正在执行飞行命令

图 3-3-10 演示了通过可视化界面执行无人机飞行命令，从作战俯视图上看无人机的航线已经改变。

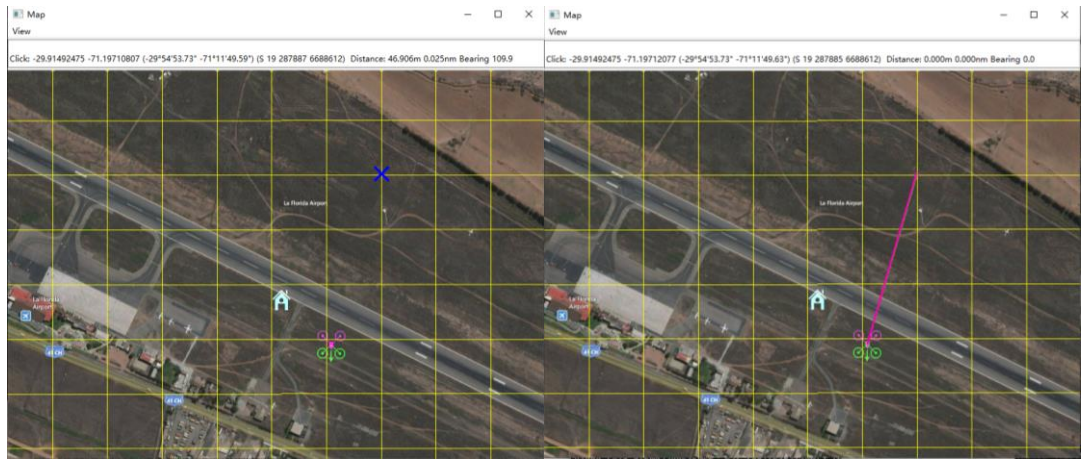


图 3-3-10 通过可视化界面执行飞行命令

3.4 操作仿真系统执行一次飞行任务

为说明智能无人机路径规划仿真系统的实用性，现以执行“围绕布兰肯塞机场飞行一周”飞行任务为背景，重新梳理一次如何使用仿真系统执行飞行任务。

首先打开智能无人机路径规划仿真系统项目文件夹，点击运行智能无人机路径规划仿真系统.bat 文件，打开软件界面，如图 3-4-1 所示：

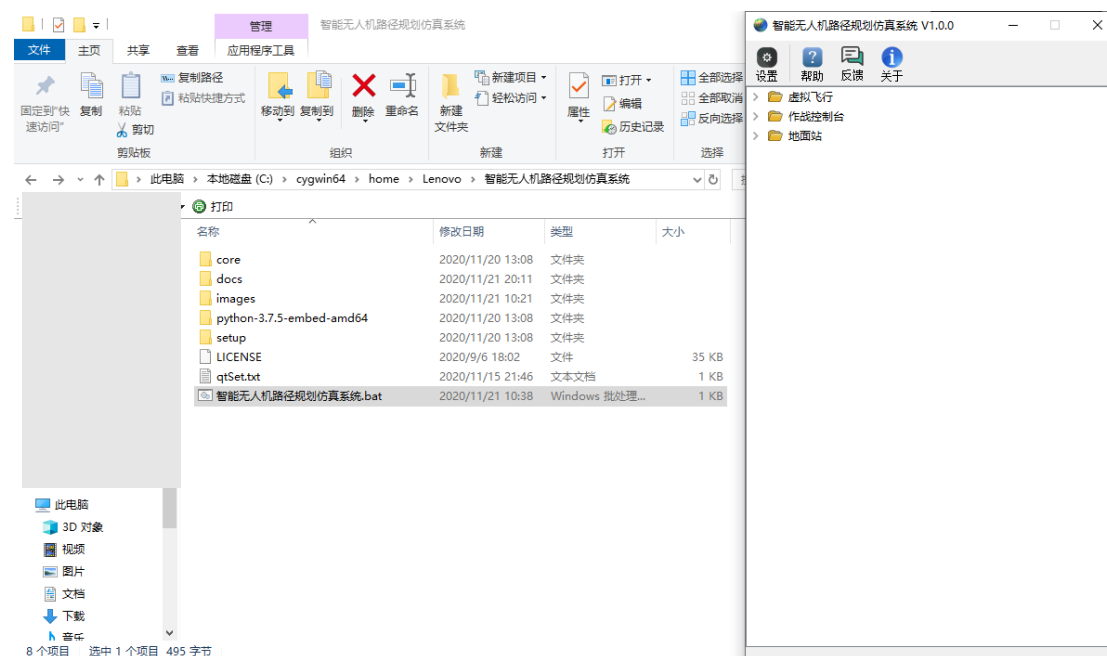


图 3-4-1 运行仿真系统

点击“虚拟飞行”文件夹，启动 FlightGear 飞行模拟器，而后在“选择作战地图”下双击“测试位置：布兰肯塞机场(EDHL)”，如图 3-4-2 所示：

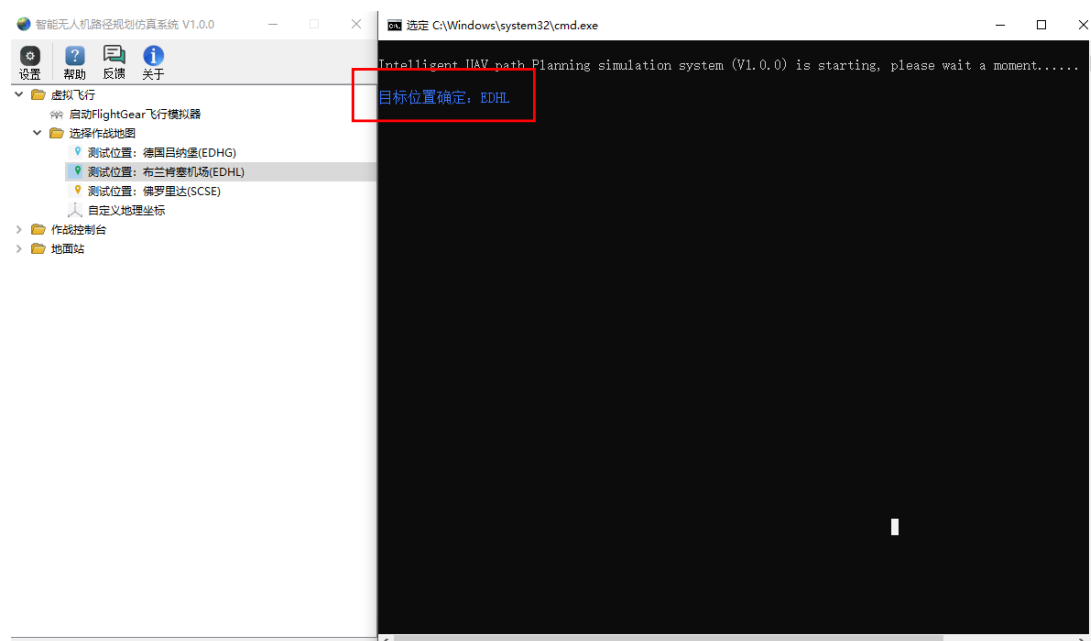


图 3-4-2 选择测试位置为布兰肯塞机场

打开“作战控制台”文件夹，启动作战平台，而后选择“加载 Mavproxy 控制台与战场俯视图”，后台将呈现相关代码，如图 3-4-5 所示：

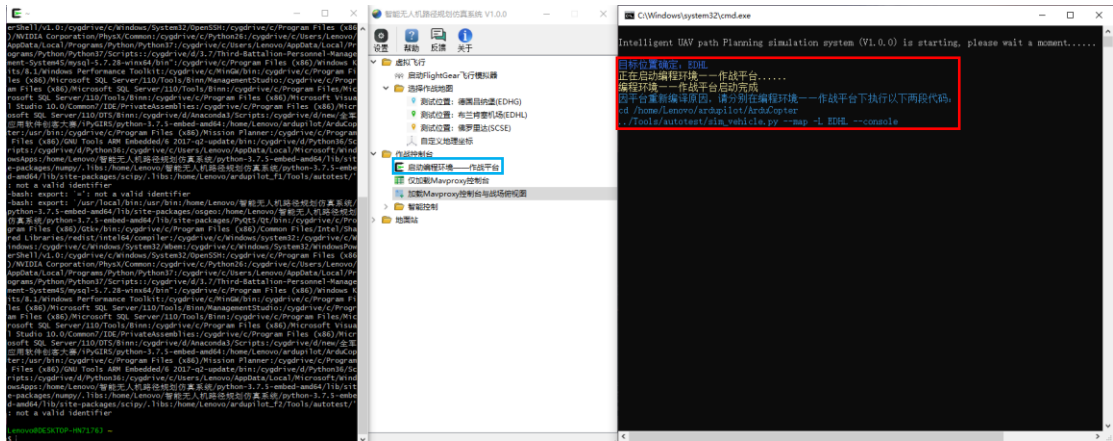


图 3-4-5 启动作战平台并选择控制台和俯视图界面

向作战平台输入在后台出现的提示代码并执行，打开控制台和作战俯视图，准备执行飞行命令，如图 3-4-6 所示：

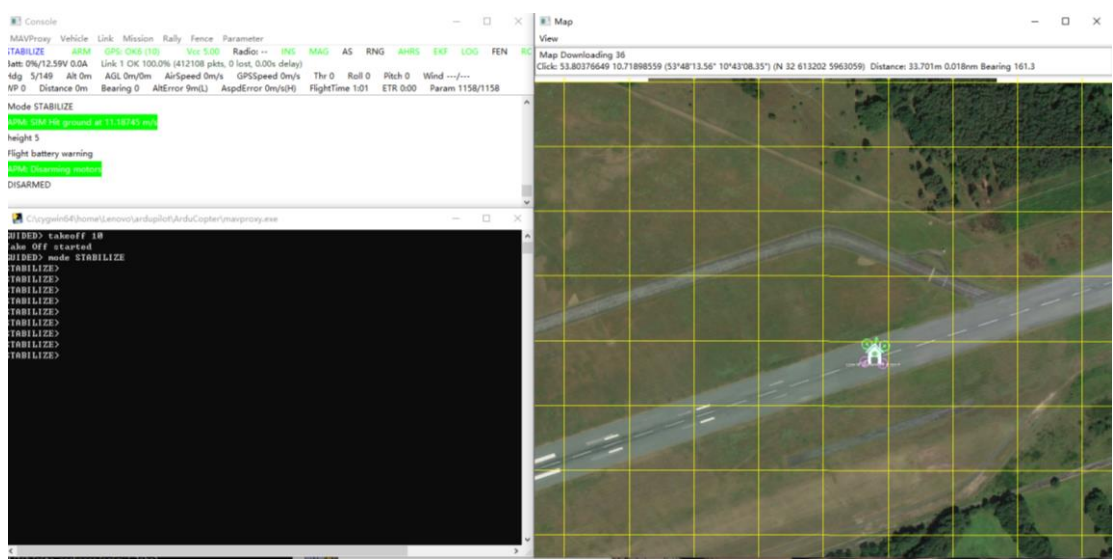


图 3-4-6 选择控制台和俯视图界面生成

想要执行飞行命令，首先应在 Mavproxy 控制台中执行命令切换控制模式 GUIDED，而后解锁电机，起飞一定高度，代码如下所示：

Mode GUIDED #切换控制模式为 GUIDED
Arm throttle #解锁电机，从三维视图可以看到无人机机翼开始旋转
Takeoff 40 #起飞高度 40 米

执行命令后，从三维视图中可以看到无人机状态已经达到指定飞行高度，如图 3-4-7 所示：

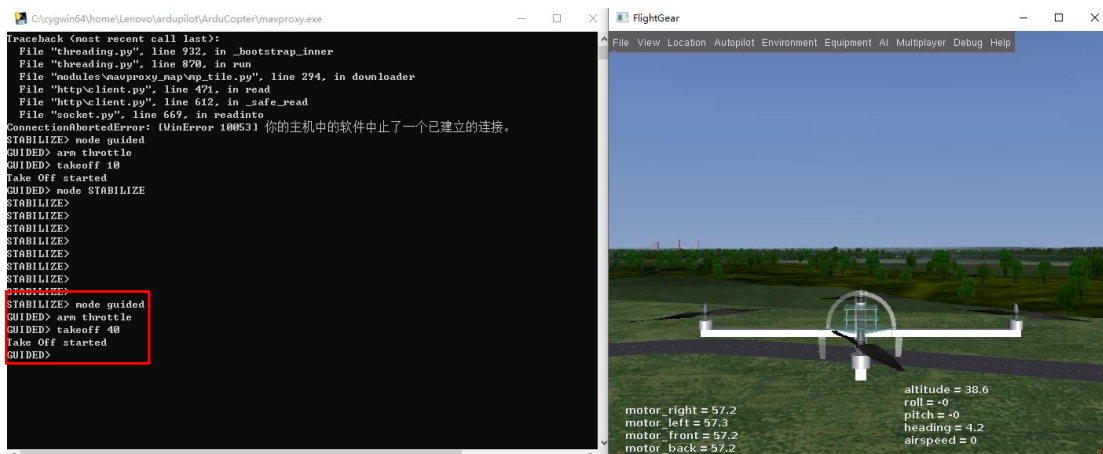


图 3-4-7 无人机起飞后已达指定高度

在到达指定高度后，可以通过无人机作战俯视图下达命令，使其执行任务，具体操作为：首先右击俯视图，在弹出菜单内选择 Mission，而后选择 Draw 命令，再次确认无人机航行时高度，在俯视图上围绕机场电机一圈，创建航点，最后在 Mavproxy 控制台中将无人机模式切换为 Auto，是无人机自动执行巡航任务。如图 3-4-8，3-4-9，3-4-10 所示：

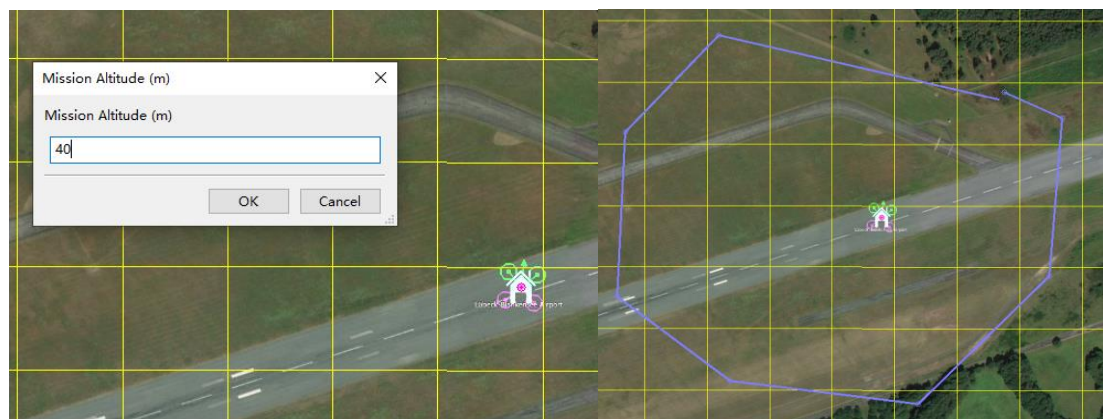


图 3-4-8 确认设定高度后，规划无人机航行路径

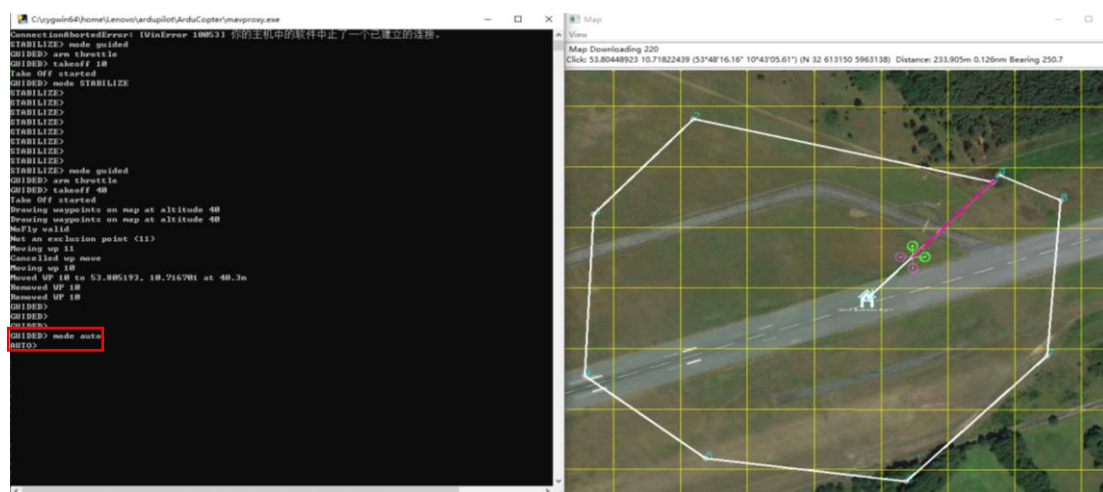


图 3-4-9 切换 Auto 模式，无人机自动进入规定航线

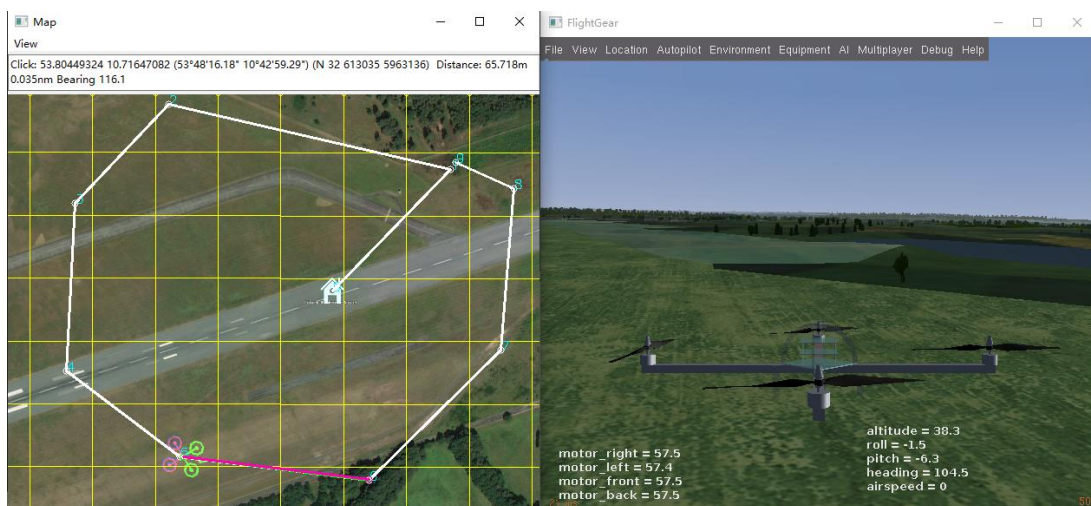


图 3-4-10 无人机正在按照规划航线完成飞行任务

任务完成！

3.5 智能控制

使用 Mavproxy 控制台操控无人机航行的缺点是操作过程较为繁琐，且在可视化界面上设置航点还不够精确。对于需要精确化航点的任务，以及减少操作流程，系统还提供了智能控制方法，即运行 Python 代码，快速启动无人机，直接控制飞行控件，对于连续飞行的任务，使用者只需提供相应的参数即可。

在作战平台启动的基础之上，双击“启动编程环境——智能平台”，另打开一个 Cygwin64 Terminal 终端，对应我提供的三个 autoFly 脚本，按照后台提示，执行相应的命令即可。如图 3-5-1，3-5-2，3-5-3 所示：

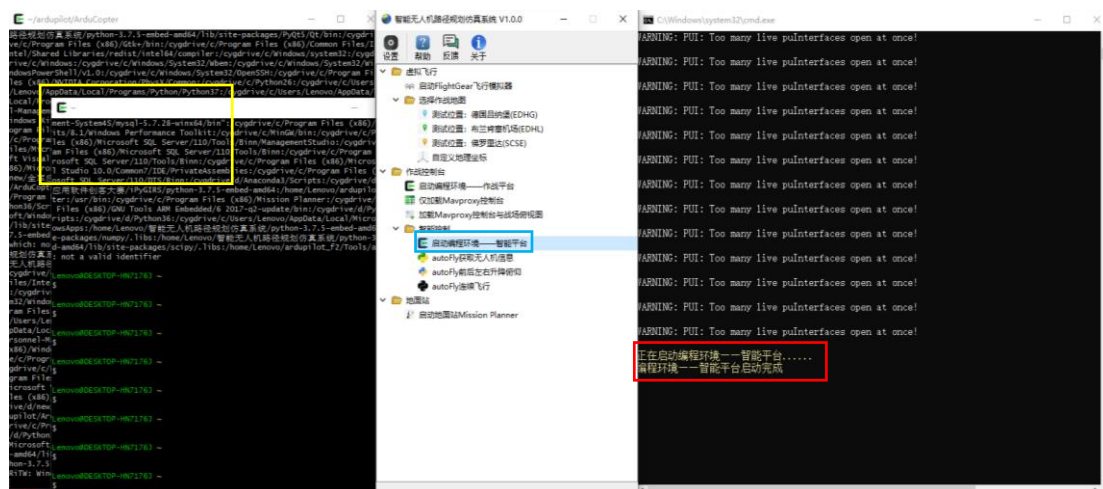


图 3-5-1 在作战平台的基础上，启动编程环境——智能平台

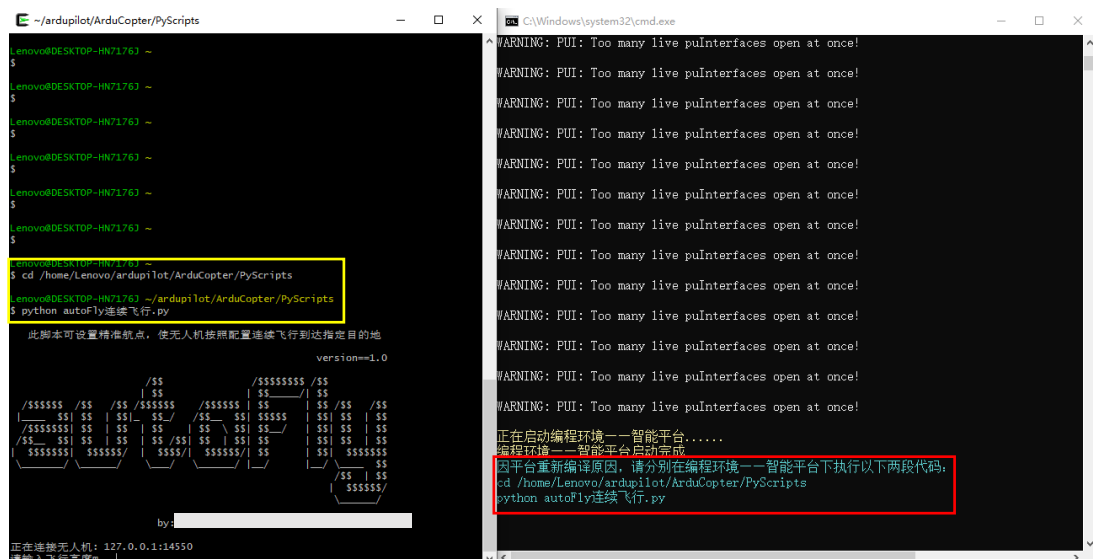


图 3-5-2 选择 autoFly 连续飞行脚本，执行后台命令

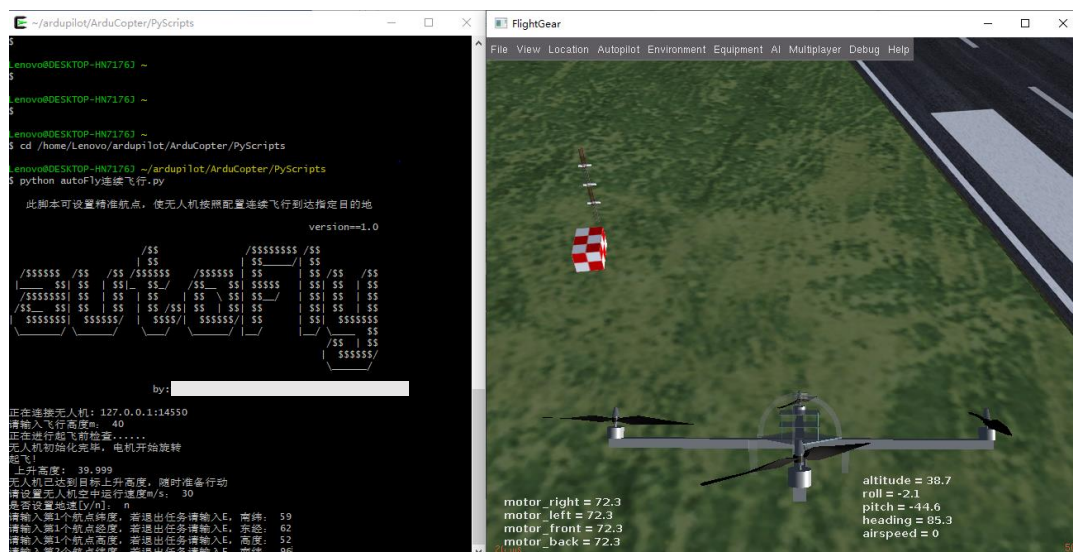


图 3-5-3 输入相应参数，无人机已经按照航点执行连续飞行任务

4. 地面站

地面站调试软件 MissionPlanner 界面与作战俯视图界面相似，但功能更加强大，它不仅可以控制无人机完成相应任务，还能对无人机所有参数设置进行可视化输入，软件可加载多种网络地图，地图界面更加清晰，且能够导出任务中的航点坐标，这意味在模拟环境中规划的路径可以导入真实无人机，使真实无人机能够在战场中迅速抵达全球任一目标位置。

在《智能无人机路径规划仿真系统》中双击“启动地面站 MissionPlanner”，启动程序，如图 4-1 所示：

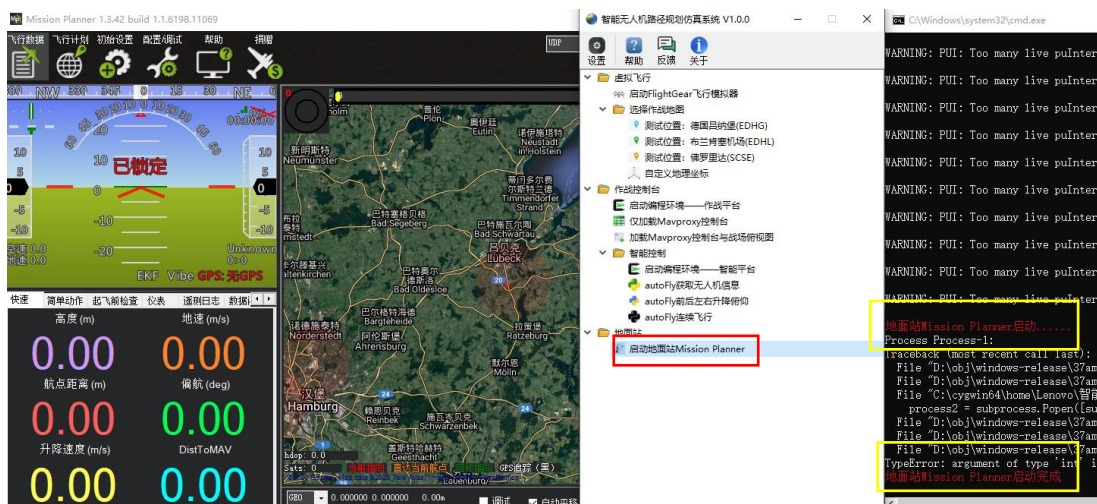


图 4-1 地面站 MissionPlanner 程序启动

连接地面站与 Mavproxy 控制台:向控制台输入 Output 命令,对外打开 14550 端口（其他未被占用的端口皆可），供地面站连接，如图 4-2 所示：

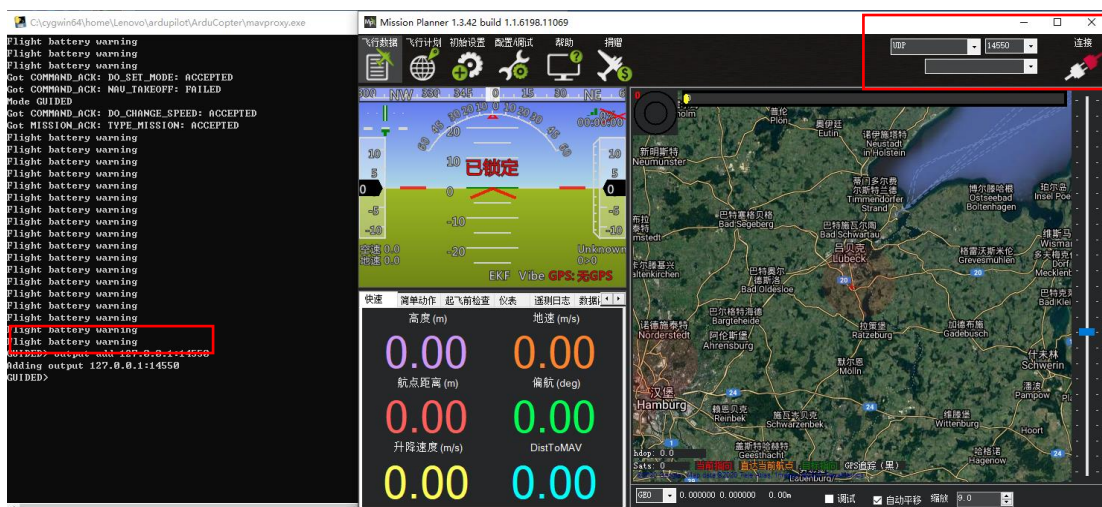


图 4-2 打开 14550 端口，连接地面站

在地面站程序中点击“连接”，并输入相应协议及端口，使地面站参与飞行任务，如图 4-3，4-4 所示：

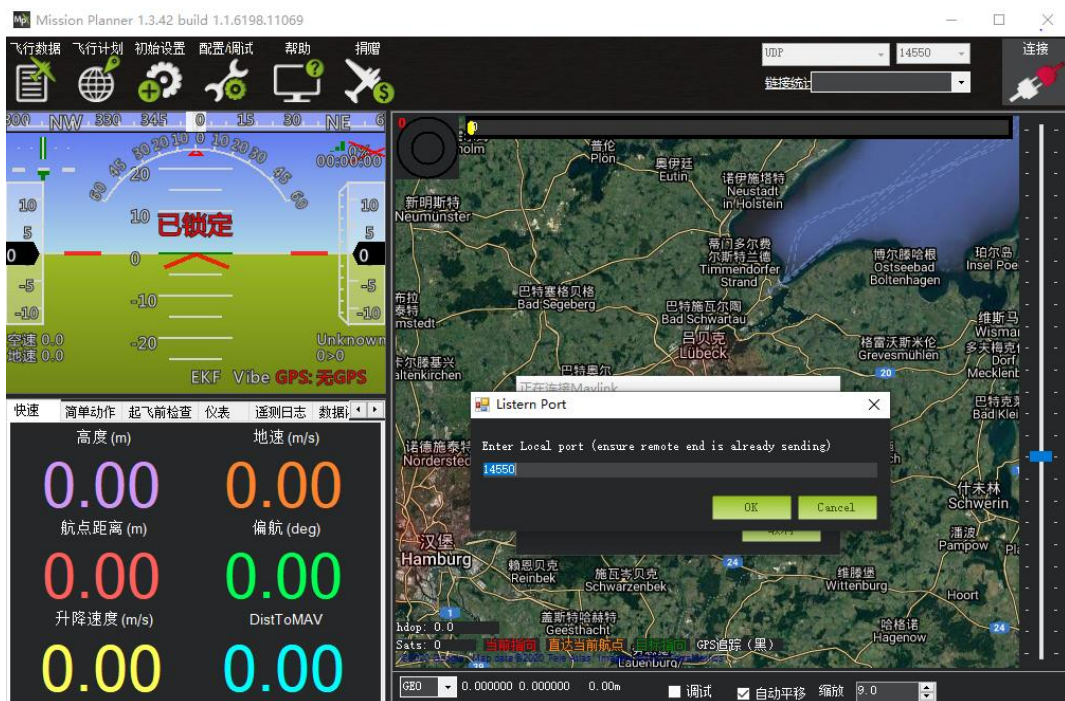


图 4-3 填写 UDP 协议，连接 14550 端口

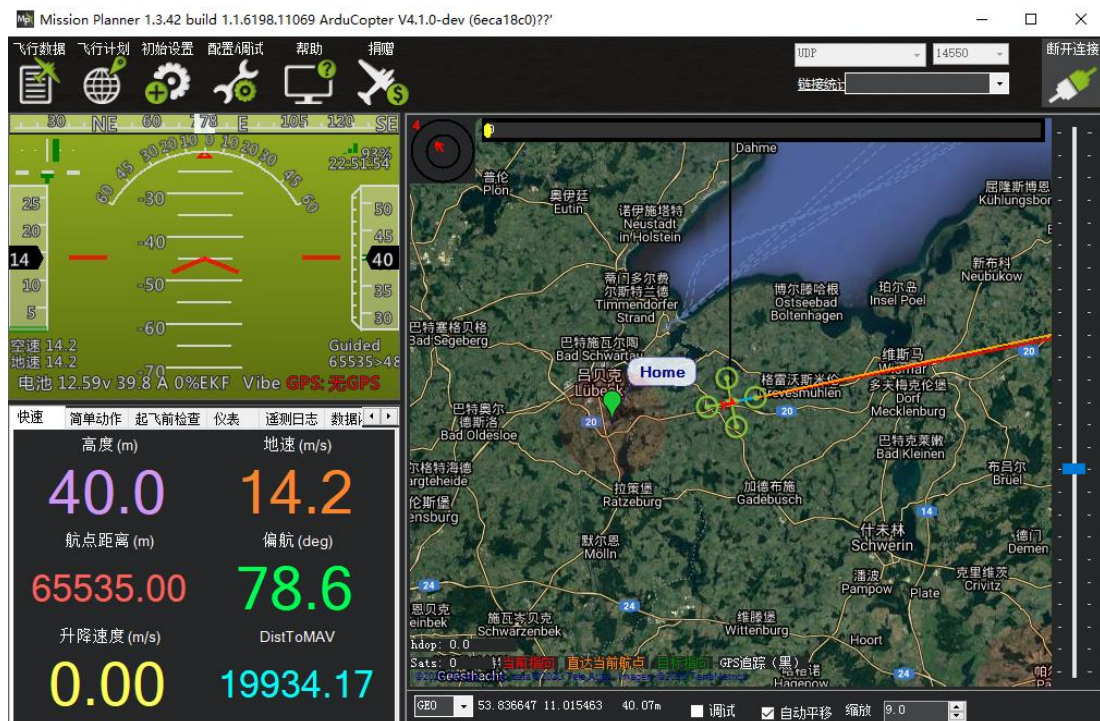


图 4-4 地面站连接成功

地面站 MissionPlanner 内基本涵盖了无人机所有参数，且有专门的飞行计划界面，支持网络地图，操作极其友好，适合军事任务的执行，各界面参数的作用大抵说明如表 4-1 所示：

表 4-1 地面站 MissionPlanner 程序各界面参数使用说明

参数	使用说明
飞行数据	该界面主要对飞行驾驶舱内参数，无人机行驶基本参数，实时飞行地图进行可视化，且可执行“自动”、“留待”、“返航”等简单动作，下方还包含了雷达、航速仪表、GPS 等指针，并对遥测日志、数据闪存日志进行记录。
飞行计划	该界面重点执行较为复杂的飞行计划，可在地图上对任务航线进行划分，并对航点进行细致区分，读取、写入外部航点，还可加载、保存航点文件，以备真实无人机使用。
初始设置	该界面主要是在软件层次上模拟安装硬件设备，并搭配多种装备，可选择相应固件在战场上进行模拟其他载体。
配置/调试	该界面主要是对无人机的默认选项进行配置，对地图围栏和各项基本参数、扩展参数、标准参数进行调整，针对使用者能力选择飞行难易模式。

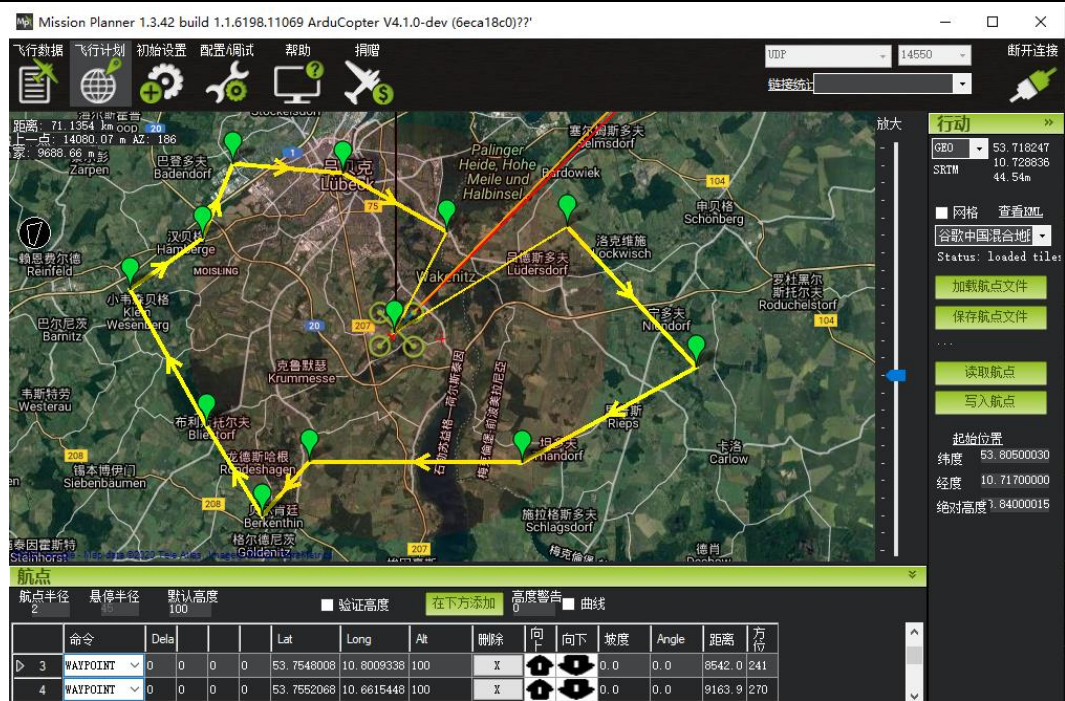


图 4-5 在地面站上规划飞行任务

多人多设备编队联合行动，是指服务器通过设置 Mavproxy 控制台中的 Out 命令，开辟多个端口供外界连接。当多个平台无人机同时连接服务器地图时，可完成编队战术行动，由于电脑资源限制，这里不再演示。