# 这周的工作：

看完墨子AI开发包的代码。

看了文档（接口文件，软件使用教程，案例说明）

并在墨子平台上运行了一个案例

学习pytorch中

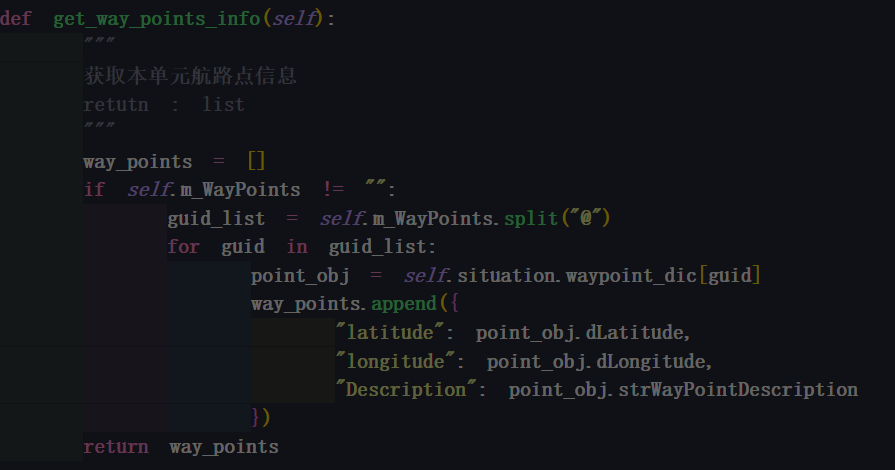
# 总结：

## 1.障碍体：

开发包里没给障碍物的定义方式（我觉得可以把障碍物也定义成Agent).  
 但是这个项目涉及到墨子平台上的想定文件，障碍物以怎样的方式出现在想定文件上是个问题

## 2.Agent定义方式：

上次基于平均场算法的设计中，智能体是按x,y,z三个坐标轴定义的。（大创申请书里出现的也是三维的  
 但是在平台上的定义方式是按经度，维度，以及它的朝向来定义的，朝向是一个 0 ~ 360 的角度，也就是说目前还是一个二维。



在后面看接口文档的过程中，我发现有height,altitude\_AGL(地理高度），CurrentAlt（当前高度）fPitch(倾斜角），fRoll(倾斜角），字样出现，我猜想可能有高度这一维度。然后，我边看文档边查开发包里的相关代码。

发现唯一涉及到的赋值语句就是这样：是否把高度直接设置为某个值

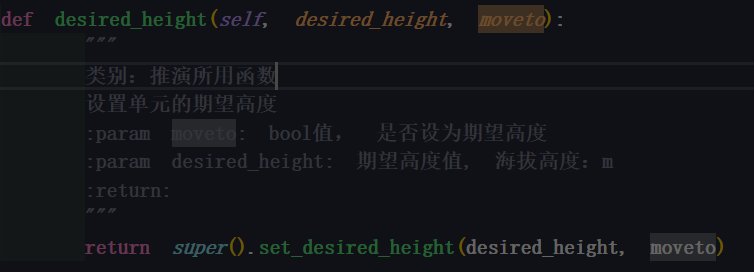


image-20210731160815675

上图是搜索height的结果）对于高度的使用基本上就是，是否把高度设为指定高度，并没有其他运算

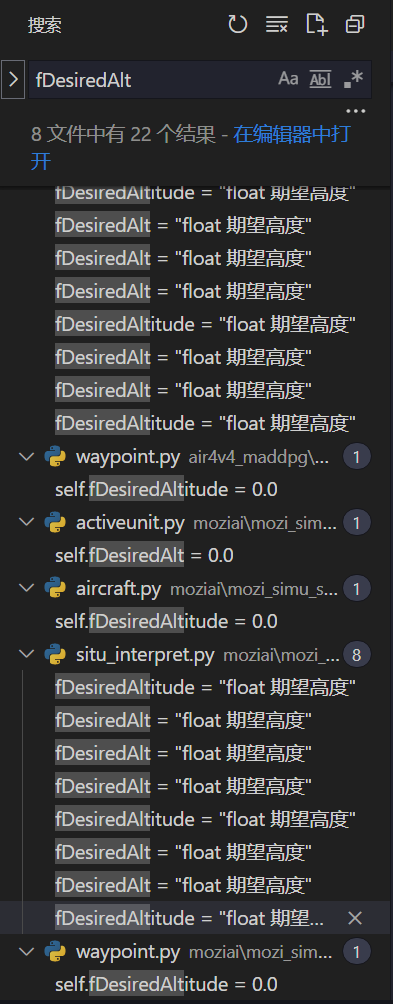


image-20210731163142525

搜到的基本就是这样的，我也点进每个代码看了，确实没有关于高度的运算，

也就是说，其实还是只能够提供二维的数据

另一方面，在开发包里提供的几个案例中，都是用经度，维度，朝向表示智能体的

如果时间充裕的话，可以结合墨子软件平台仔细看一下，到底有没有通过某些接口，传入高度进来，不过代码开发包并没有相关赋值语句，所以基本可以确定这个平台只能提供二维数据

目前，设计的基于MF-AC的球面算法 表示智能体的方式是： 经度 维度（0~2π）速度

## 3.执行的动作：

1）基于墨子平台实现的其他案例代码（基本是调用里面的ddpg算法包（基于pytorch模块实现）实现的，里面只有这一种算法，可以进化成maddpg），*初步判断可以根据接口写别的算法*

2）他们是按改变朝向的方式来执行动作的：  
 *当前朝向 = 上次朝向 + 改变的朝向；*  
 *然后再按这个朝向前进一段距离*（这个距离是一个参数，应该可以把它改成速度（所有智能体都一样或者是不一样））

上次平均场的设计的动作就是一个在三维空间的速度（如果墨子平台是二维的，这里会有冲突

本次的球面平均场，改进后基本和这个对上了

3）案例代码实现还有一个动作就是攻击敌人（如果敌人在攻击范围内）（案例都是己方攻击敌方的形式）  
 （不知道咱们这个涉及到敌人吗？  
 不过可以考虑把障碍物视作敌人，可以直接把障碍物搞掉）

## 4.奖励机制

基于墨子平台实现的其他案例代码的奖励机制：

reward = 距离目标的距离所带来的的奖励 + 自己是否存在的奖惩 + 智能体打掉目标会给智能体正的奖励（目标（敌人）是否存在的奖惩）

本次设计的回报函数是：智能体当前位置到目标位置的函数 + 智能体当前位置到距离其最近障碍物（或其他智能体）距离的函数

*这个距离其最近的智能体和障碍物怎么找是个问题*  
 似乎墨子AI开发包里的代码没给出实现，墨子代码里有一个sensor传感器，是基于guid（每个智能体的全局id）的，好像和这里的基于距离判断邻居有点不搭

如果是我自己实现，我暂时想不到比较好的实现方式，应该就是挨个算一遍把，纯遍历的方式

（可能需要仔细运行一些接口才能想到好办法）

## 5.关于平均动作：

与上面类似，**怎么找它的邻居是个问题**



我在代码里看到的比较有用的就是这部分，感觉传感器可能不太满足需求

我在看vicsek模型代码的过程中获得一点启发：  
 我看有的python代码中利用一些模块可以比较简洁地实现这个功能，不过能不能这样解决咱们的问题情境，我还需要再看看，因为距离的表示是结合墨子平台的，问题情境不一样吧  
 如果不用这个模块，可能需要自己写一写代码

主要是担心如果找邻居的算法写的不好（就是用的维度大，算法复杂度高），执行代码就会花很多时间，对于多Agent就失去意义了

有时间我再去看看平均场的那个代码实现，看看他是怎么实现找邻居的，当时没太看明白他是怎么找的邻居()

### 6.关于墨子平台

墨子平台（windows)太大了，运行起来有些占用内存

主要是为了做军旗推演的

这里面很多的案例代码都是一样的，只有所用的想定文件是不一样的

这里面dppg算法实现的接口，基本没有涉及到moziai的代码，对于后续算法开发很nice，只需要满足在最终的函数 接入的参数和传出的参数一致即可

windows平台只能用scene的想定文件，Linux只能用xml的想定文件

1）邓老师给我发了个案例实现（其实墨子AI开发包代码里也有一些案例实现）

我想运行一下这个案例，  
 上周打开这个软件显示验证码已过期没办法使用，给平台上发了两个邮箱没有收到回复。  
 在 bilibili.com 收到了一个曾经使用过这个软件的人的帮助，修改了验证码，可以成功使用了，但是软件上显示“华戍防务授权剩余时间：151天，可能后续还需要联系一下厂家

**（作战想定文件是作战仿真系统运行的基本依据和数据来源）**

然后，卡在了怎么把想定文件读入墨子平台，因为下的是windows版本的，只支持scen格式的想定文件

但是，这个案例实现的想定文件是xml格式的，我不太清楚怎么转成.scen格式或者怎么直接使用xml格式的想定文件（查看所有文档好像没办法把xml文档转成scen格式, 只能把scen转成xml）

由于这个案例实现是基于uav\_anti\_tank案例（无人机反坦克）改进而来的，所以我跑了一下这个无人机反坦克的案例，顺利跑起来了，心里也有底了

2）官网上的墨子平台（Linux版本）好像下载不了，虽然我应该也运行不了这个软件（因为我没有Linux的电脑，只能在虚拟机上运行Linux系统，内存有限，估计运行不了这么大的软件）

## 7.关于设备

我的显卡是AMD的，当时买电脑的时候没想到只有英伟达的显卡可以做深度学习，

这样只能用cpu进行运算，会慢很多。  
 如果只是这个项目是纯基于代码的实现，我可以在云计算平台上实现，比如矩池云和Google colab(这个还没试过)  
 （云计算平台一般只提供一个jupyter notebook，不提供其他软件），  
 这就解决显卡的问题了

现在项目的实现要基于墨子平台，要在本地训练，我的显卡就没办法了，只用CPU运算可能会慢一点

刚刚突然想到云电脑这回事，能不能在云电脑上装软件（mozi软件），也支持深度学习

9.还有就是，怎么结束训练（是否done)：自己是否存在 or 敌人是否存在

10.关于(Actor Critic) 网络暂时没啥问题，感觉都定义的差不多，可能要把代码写出来，根据运行结果再调整网络的参数吧

11.看了一下 能实现路径规划 而且还是 3D的平台 ，感觉不多的样子，不太好找，

不过可以在其他的路径规划的论文看看他们用的什么平台或者在什么情境下跑的代码  
 暂时感觉用墨子平台可能比较重，不方便多人相互协作，不过如果是基于无人机而且还要满足路径规划的需求，这个平台确实挺满足需求的

12.观察到的是经度，维度，朝向

13.moziai\_水面舰艇带直升机反潜案例:  
 状态空间分析：  
 在墨子推演系统返回的态势数据中，一架飞机的数据大概有 一百多维，维度较大，算法无法很快收敛。  
 我们选取最能影响任务成败的维度： 【经度，纬度，朝向】。  
 动作空间分析：  
 在墨子推演系统中，飞机可以执行的动作有几十种，我们选 取最能影响任务成败的动作：【朝向，是否投放声呐】，有的只选朝向  
 奖赏值分析：  
 假设目标区域大概已知，所以 1）接近目标区域会给智能体奖励。 2）智能体被击毁会给负的奖励。 3）智能体打掉目标会给智能体正的奖励。

“墨子”系统提供仿真推演所需的平台支撑和模型、数据等各类资源  
Python墨子开发包提供各类函数交互接口，实现“墨子”系统和AI处理系统之间的数据和指令的交互；  
人工智能（AI）处理系统实现针对决策参数和评价指标的深度强化学习

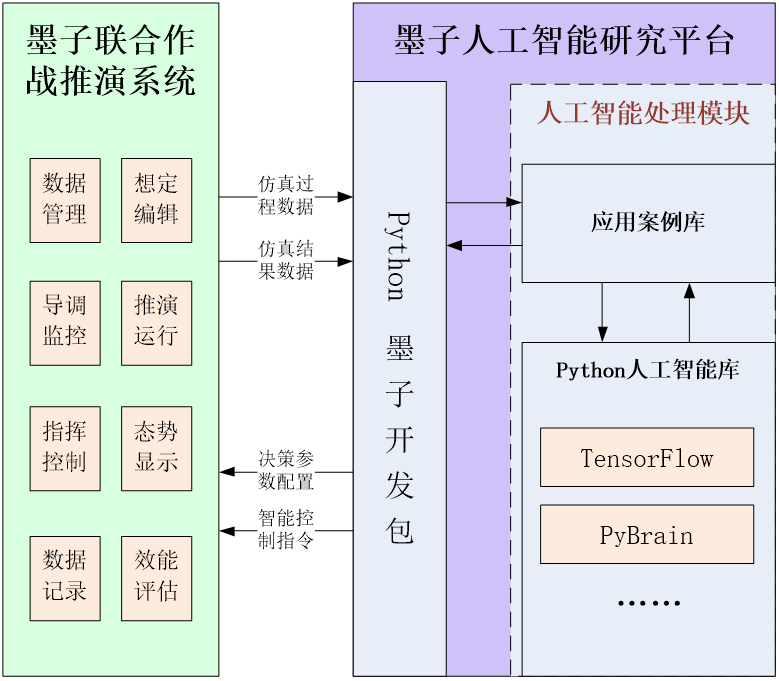


image-20210731150948912

Python人工智能客户端是指部署了Python人工智能开发环境的终端，通过调用墨子Python API，可以建立起面向MoziServer的控制连接，并驱动MoziServer创建及编辑推演、添加或修改单元及任务、执行动作命令并获得反馈及实时态势，从而开展军事人工智能训练或相关应用。

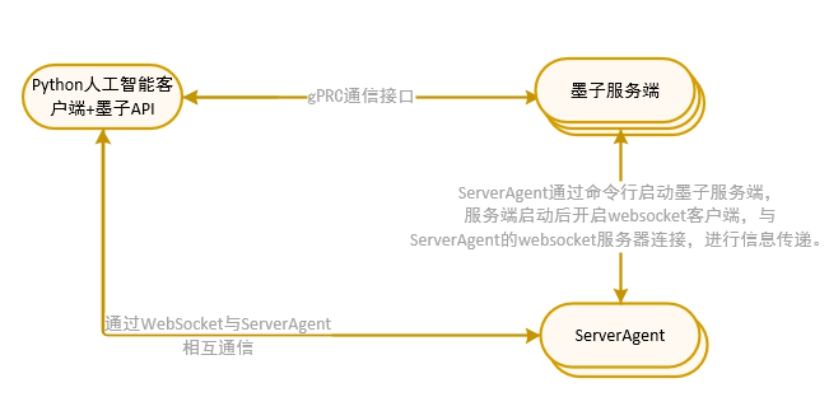


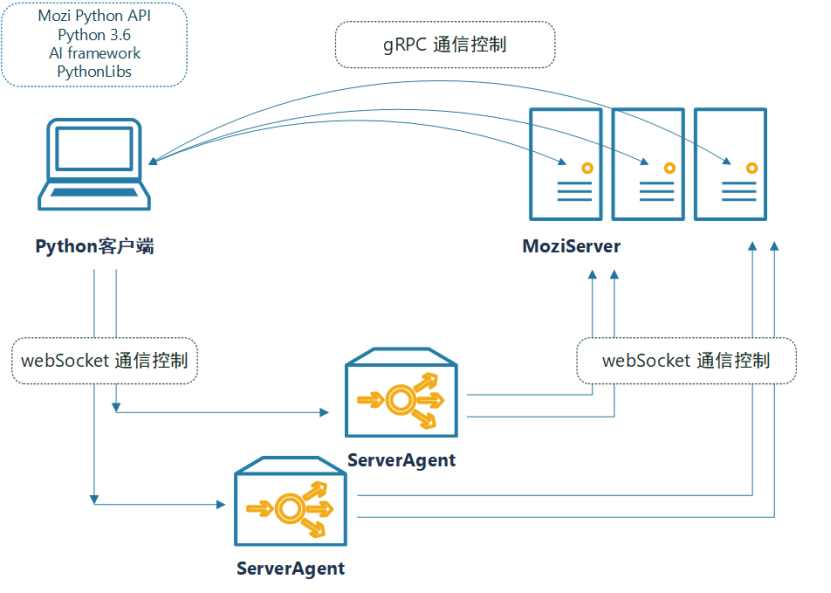
image-20210731151003431

图1-3 平台组件调用关系

1.3平台的使用模式  
1.3.1军事人工智能开发和训练模式  
使用墨子人工智能平台进行军事人工智能开发，主要有两种使用方式：  
（一）在单位内部部署平台，进行军事人工智能的开发与训练  
采用此种方式时，还可以按照开发规模分为三种配置方式：  
（1）一个Python客户端直接连接一个“墨子”系统服务端，两者甚至可以同属于一个操作系统，这种方式是最简单的，适合个人开发调试。



image-20210731151108646

（2）一个客户端，通过配置一个或多个ServerAgent，控制启动多个“墨子”系统服务端，此种方式适合大规模的训练。

（3）多个客户端通过ServerAgent，连接一个或多个“墨子”服务端，此种模式下，多个客户端可以在同一个推演中，控制不同的单元或者推演方，实现多个智能体的协同训练。

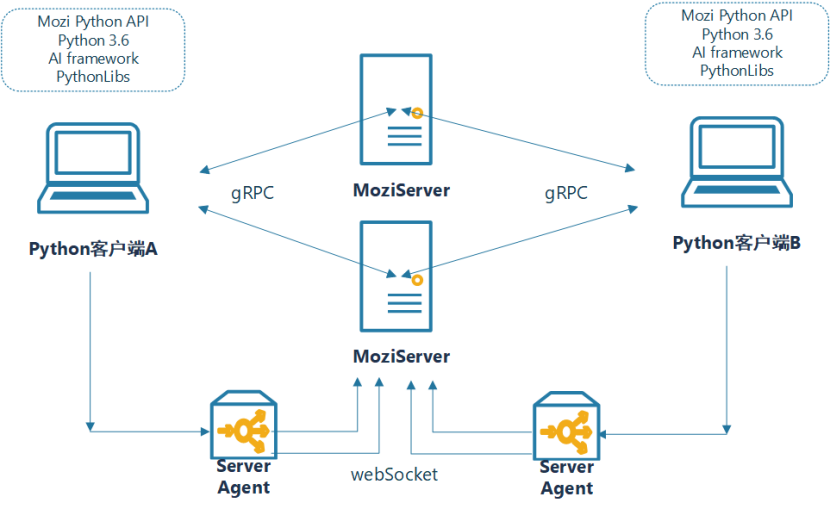


image-20210731151051084

## 在互联网上进行军事人工智能的开发和训练

此模式下，用户只需要安装必要的Python人工智能开发程序以及墨子客户端软件，即可通过公开的IP及端口，接入到云端的服务端资源，开展人工智能的测试研究功能。这个公开的人工智能服务端目前部署在阿里云上，具体IP地址为：59.110.163.138。(进了这个网站发现，是这样，不知道该咋注册)

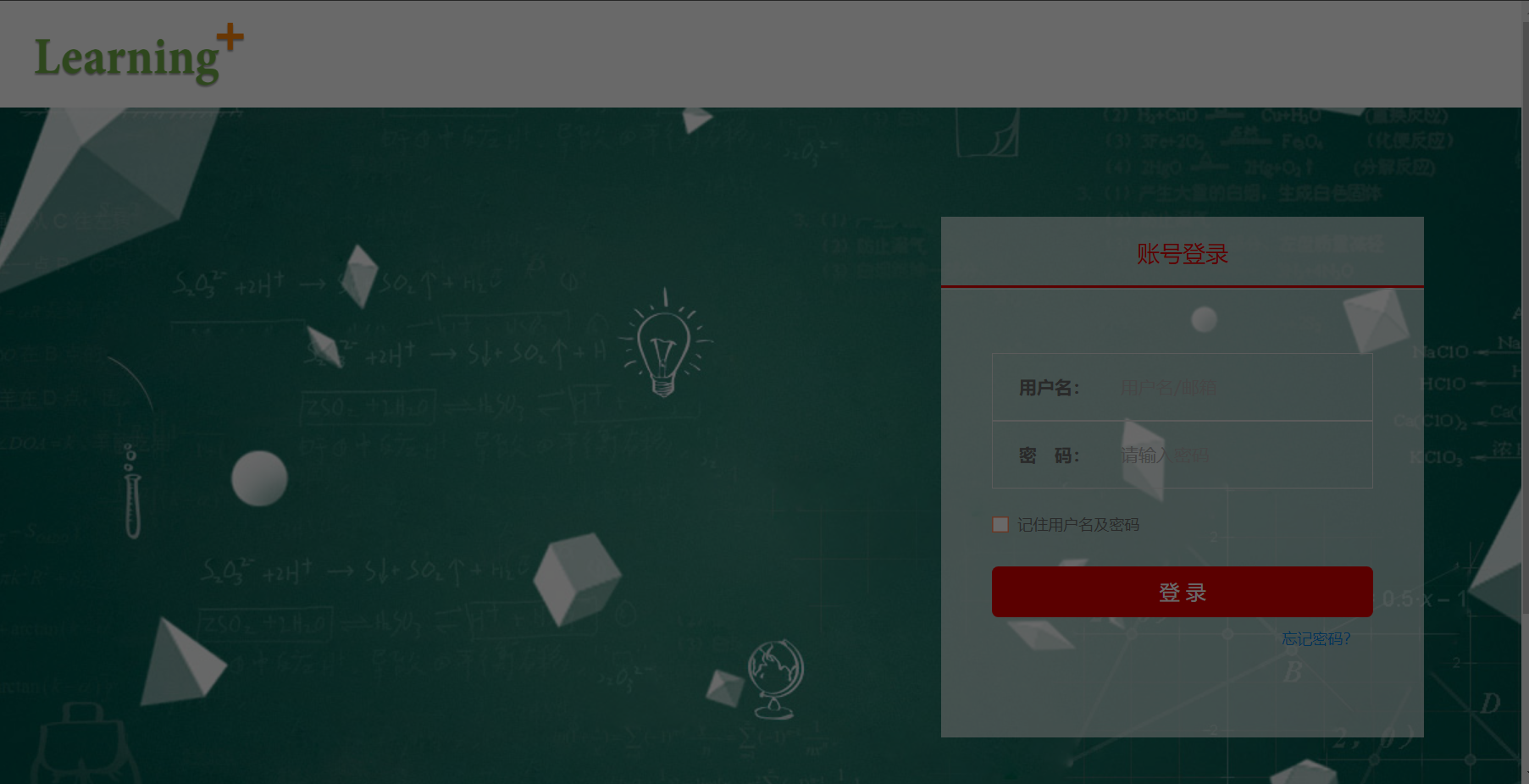


image-20210731151725172

目前“墨子”系统服务端、ServerAgent以及相应的通信接口均同时支持Windows及Linux，