



# Homework 2

## 要求

5 架四旋翼无人机保持旋转圆形编队，并且跟踪具有加速度，速度变化的领导者

## 问题分析

希望通过修改示例代码来达到题目要求，所以此处主要讨论示例代码和题目的不同，以发现并分析需要修改的地方

1. 状态偏移向量的时变性：示例代码和题目最大的区别在于无人机与领导者之间的状态偏移向量是时变的，因此需要先通过算法运行时间来确定状态偏移向量；
2. 编队补偿控制输入：示例代码没有非线性函数这一项，由于领导者是有非零加速度的，故题目需要加入这一项，也存在如何选择 **b** 的问题；
3. 编队通信拓扑图：示例代码中，无人机的通信拓扑图为全连通图，题目的参考通信图为环形通信图，每个节点可以收到相邻两个节点的信息，并且，只有 **1** 号无人机可以收到领导者的信息。但是实现的时候发现如果只有 **1** 号机可以收到领导者信息的话，控制难度过大，因此我在保持环形通信图的基础上，选择了让所有的节点都可以收到领导者信息，最终实现稳定控制

## 数学推导

代码的实现是以正确的数学推导为基础的，所以在此先推导必要的数学公式，再实现代码。通过问题分析模块可知，需要求解领导者的速度和编队补偿控制输入：

$$x_{0vx} = -0.5 \sin(0.1t) \quad (1)$$

$$x_{0vy} = 0.5 \cos(0.1t) \quad (2)$$

$$\dot{h}_{ivx} = -w^2 r \cos \left( wt + \frac{2\pi(i-1)}{5} \right) \quad (3)$$

$$\dot{h}_{ivy} = -w^2 r \sin \left( wt + \frac{2\pi(i-1)}{5} \right) \quad (4)$$

## 代码实现

此章节包含将示例代码修改为满足题目要求的解题代码所需要的关键修改步骤：

### 1. 计算领导者位置和速度：

```
center_px = 5 * math.cos(0.1 * elapsed_time)
center_py = 5 * math.sin(0.1 * elapsed_time)
center_vx = -0.5 * math.sin(0.1 * elapsed_time)
center_vy = 0.5 * math.cos(0.1 * elapsed_time)
```

### 2. 计算状态偏移：

```
self.data[uav]["hpx"] =
    self.r * math.cos(self.w * elapsed_time + 2 * math.pi * i / 5)
self.data[uav]["hpy"] =
    self.r * math.sin(self.w * elapsed_time + 2 * math.pi * i / 5)
self.data[uav]["hvx"] =
    -self.w * self.r * math.sin(self.w * elapsed_time + 2 * math.pi * i / 5)
self.data[uav]["hvy"] =
    self.w * self.r * math.cos(self.w * elapsed_time + 2 * math.pi * i / 5)
```

### 3. 计算时变编队补偿控制输入：

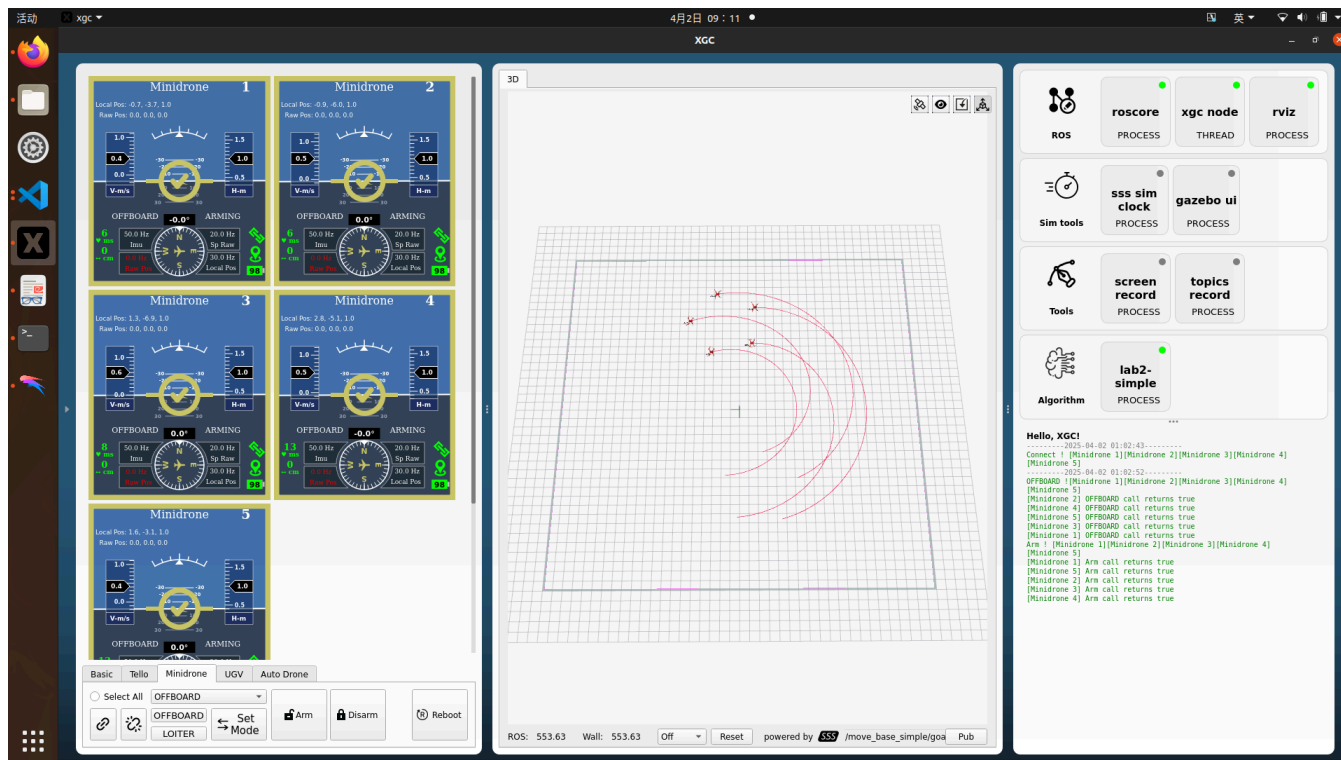
```
acc_x += self.w**2 * self.r * math.cos(self.w * elapsed_time + 2*math.pi*i/5)
acc_y += self.w**2 * self.r * math.sin(self.w * elapsed_time + 2*math.pi*i/5)
```

### 4. 计算非线性函数：

```
c = 1.0
b = 0.1
sigma = 0.01
phi = c + b / (math.sqrt(delta_x**2 + delta_y**2) + sigma)
```

## 结果展示

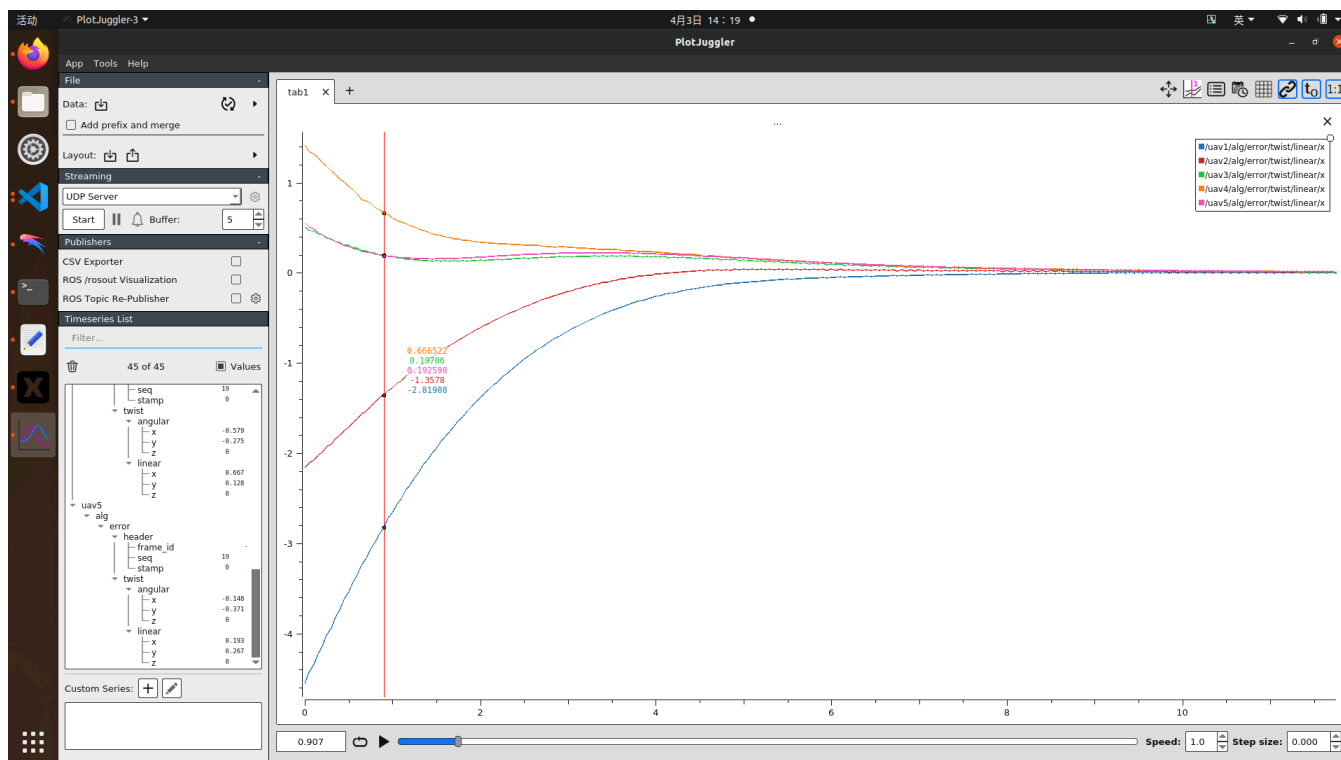
此处仅展示截图，详见视频：lab2\_result.webm



## 误差分析

此处仅展示截图，详见视频：lab2\_analyze.webm

(见下页)



1. 现象说明：5 台无人机的误差变化非常平滑，8s 后误差趋于 0，并且没有振荡
2. 现象分析：lab2 和 lab1 相比，补偿控制输入多了非线性函数： $\phi$  是近似的符号函数，这一项用来抑制领导者输入的不确定性，即模拟高频的切换来消除误差，在数值上的选择以大压小