

2、机械臂头部智能规划路线

一、API 知识准备

`send_coords([x, y, z, rx, ry, rz], speed, model)` 是用来控制机械臂头部以指定姿态移动到指定点。它主要用于实现智能规划机械臂头部从一个位置到另一个指定位置。 x, y, z 表示的是机械臂头部在空间中的位置（该坐标系为直角坐标系）， rx, ry, rz 表示的是机械臂头部在该点的姿态（该坐标系为欧拉坐标）。算法的实现以及欧拉坐标的表示需要一定的学术知识，这里不对其过多的讲解，我们只要懂得直角坐标系就可以很好的使用这个函数了。

二、API 简介

1、`send_coords([x, y, z, rx, ry, rz], speed, model)`

函数功能：智能规划路线，让机械臂头部从原来点移动到指定点。

参数说明：

$[x, y, z, rx, ry, rz]$ ：该参数中以 x, y, z 组合成一个空间直角坐标系，以机械臂底部为原点，前方为 x 正轴，右方为 y 正轴，上方为 z 正轴。 $[x, y, z]$ 表示为机械臂头部所在的位置。 $[rx, ry, rz]$ 表示的是机械臂头部的姿态，你可以调整机械臂头部的姿态然后使用 `get_coords()` 函数获取该点的姿态，这样你就不需要了解欧拉坐标系也能很好的使用该函数了。

`speed`：表示机械臂运动的速度。取值范围为 $0 \sim 100$ ，值越大速度越快。

model: 取值限定 0 和 1。0 表示机械臂头部移动的路径为非线性，即随机规划路线，只要机械臂头部以保持规定的姿态移动到指定点即可。1 表示机械臂头部移动的路径为线性的，即智能规划路线让机械臂头部以直线的方式移动到指定点。

2、get_coords()

函数功能：获取此时机械臂头部的空间坐标以及当前姿态。

返回值：返回的类型是包含六个 float 元素的 list 集合，前三个坐标为 x, y, z 表示机械臂头部的坐标，后三个坐标 rx, ry, rz 表示机械臂头部的姿态。

3、send_coord(id, coord, speed)

函数功能：仅单独修改机械臂头部空间坐标中 x, y, z 轴的其中一个坐标。

参数说明：

id: genre.Coord 类型数据，它表示的是机械臂头部的 X, Y, Z 轴。表示方式举例：X 轴 Coord.X.value。（可参考案例）

coord: float 类型数据，表示的是修改坐标值。

speed: 表示机械臂运动的速度。取值范围为 0~100，值越大速度越快。

三、代码内容

```
from pymycobot.mycobot import MyCobot
from pymycobot.genre import Coord
from pymycobot import PI_PORT, PI_BAUD # 当使用树莓派版本的 mycobot 时，可以引用这两个变量进行 MyCobot 初始化
import time
```

```
# 初始化一个 MyCobot 对象
mc = MyCobot(PI_PORT, PI_BAUD)
# 获取当前头部的角度以及姿态
coords = mc.get_coords()
print(coords)
# 智能规划路线，让头部以线性的方式到达[59.9,-65.8,250.7]这个坐标，以及保持[-50.99,83.14,-52.42]这个姿态
mc.send_coords([59.9, -65.8, 250.7, -50.99, 83.14, -52.42], 80, 1)
# 设置等待时间
time.sleep(1.5)
# 智能规划路线，让头部以线性的方式到达[59.9,-65.8,350.7]这个坐标，以及保持[-50.99,83.14,-52.42]这个姿态
mc.send_coords([59.9, -65.8, 350.7, -50.99, 83.14, -52.42], 80, 1)
# 设置等待时间
time.sleep(1.5)
# 仅改变头部的 x 坐标，设置头部的 x 坐标为-40。让其智能规划路线让头部移动到改变后的位置
mc.send_coord(Coord.X.value, -40, 70)
```