

代码

211250173 汪天佑

1. 基本功能的实现

我的代码主要通过一下思路来实现小车的靠墙移动.

1. 环境的感知
2. 决策的计算
3. 动作信息的发布 详见代码: /catkin_ws/src/f1tenth_simulator/node/safety.cpp文件

2. 速度为1.5m/s的实现:

在代码中动作信息发布的部分每次都只是将小车速度设为1.5即可

```
ackermann_drive_result.drive.speed = 1.5;  
drive_pub.publish(ackermann_drive_result);
```

效果见video1_slow

3. 速度为4m/s的速度实现:

在这个部分如果让小车始终以4m/s的速度前进的话会出现碰撞,所以我们这里综合拐弯时减速等机制来共同实现. 效果见video2_quick

4. 优化

4.1. 参数整定

KP、KD和KI是比例-积分-微分(PID)控制器的三个参数，它们分别代表比例、积分和微分项。：

KP (比例系数)

- 作用：KP是比例项的系数，它决定了控制器响应对当前误差的强度。较高的KP值会使系统响应更快，但也可能导致过冲（即超过目标点然后再回来），甚至系统不稳定。较低的KP值会使系统响应变慢，但能提高稳定性。

KD (微分系数)

- 作用：KD是微分项的系数，它基于误差的变化率（即误差的导数）来进行控制。微分控制考虑的是误差的变化速度, 微分项有助于减少系统的过冲和振荡，使系统更快地稳定下来。高的KD值可以使系统对快速变化更敏感，但过高可能导致噪声放大，引起不必要的控制动作。

KI (积分系数)

- 作用：KI是积分项的系数，它基于误差随时间的累积总和进行控制。积分控制的目的是消除稳态误差，积分项有助于确保系统最终能够到达所需的设定点，消除稳态误差。但是，过高的KI值可能导致系统变得过于敏感和不稳定，引起持续的振荡。

综合使用KP、KD、KI

通过在实验中不断探索KP,KD,KI的组合以发现更好地实验结果, 最后敲定了

```
#define KP 1.00
#define KD 0.001
#define KI 0.005
```

为适合本人本次实验的最佳参数组合

4.2. 速度控制(拐弯时降速)

代码实现:

```
if (abs(ackermann_drive_result.drive.steering_angle) > 20.0 / 180.0 * PI) {
    ackermann_drive_result.drive.speed = 2.0;
} else if (abs(ackermann_drive_result.drive.steering_angle) > 10.0 / 180.0 * PI) {
    ackermann_drive_result.drive.speed = 3.0;
} else {
    ackermann_drive_result.drive.speed = 5.0;
}
```

4.3 算法提高:

为了取得更好地效果,我使用了以下策略:

- 提前量探索:
考虑到激光雷达,以及信号触发等等的延迟, 在做出决策时, 程序会考虑到因为这些延时小车的偏移量. 由

```
#define LOOK_AHEAD_DIS 1.0
```

控制, 最终在反复地数据尝试中LOOK_AHEAD_DIS 定为1

- 速度的摸索

速度1	速度2	速度3	效果
2.0	3.0	5.0	若干圈均无碰撞,速度设为 '1'
1.0	3.0	5.0	无碰撞, 速度为 '0.95'
2.0	3.0	4.0	无碰撞,速度为 '0.85'

速度1	速度2	速度3	效果
2.0	3.0	6.0	第三圈出现碰撞, 速度为 '1.1'
.....			

最终综合碰撞情况和小车速度,选择速度1为 2.0,速度2 为3.0, 速度3 为5.0

- PID的参数摸索
 PID的取值探索是基于上述速度组合上的.

KP	KI	KD	碰撞情况	小车稳定性(即震荡情况)
1.00	0.001	0.005	无碰撞	较佳
1.00	0.002	0.005	出现碰撞	较不稳定
1.00	0.001	0.01	未出现碰撞	较不稳定,易出现持续震荡
.....				

- 异常数据发现和剔除 在对激光雷达数据的几何处理中, 通过检查是否为无穷大或非数值来处理激光雷达数据中的异常值,确保计算的准确性和可靠性