2. Lidar obstacle avoidance

1. Program Function Description

When the program is started, the cart will go forward, and when there is an obstacle in the detection range, it will adjust its attitude to avoid the obstacle, and then continue to go forward. If the handle node is activated, the R2 key of the handle can pause/enable this function.

2. Program code reference path

After entering the docker container, the source code of this function is located at.

```
/root/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_laser/yahboomcar_laser/laser_Avoidance_a1_R2.py
/root/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_laser/yahboomcar_laser/laser_Avoidance_4ROS_R2.py
```

View the source code according to the actual lidar purchased.

3. Program startup

3.1 Startup command

After entering the docker container, according to the actual model and lidar model, terminal input, the

```
#启动小车底盘 #Start the trolley chassis
ros2 run yahboomcar_bringup Ackman_driver_R2
#启动A1雷达 # Activate A1 lidar
ros2 launch sllidar_ros2 sllidar_launch.py
#启动4ROS雷达 # Started 4ROS lidar
ros2 launch ydlidar_ros2_driver ydlidar_launch.py
#启动雷达避障程序 A1雷达 # Initiating lidar obstacle avoidance program A1 lidar
ros2 run yahboomcar_laser laser_Avoidance_a1_R2
#启动雷达避障程序 4ROS雷达 # Initiating lidar obstacle avoidance program 4ROS lidar
ros2 run yahboomcar_laser laser_Avoidance_4ROS_R2
#启动手柄，如果需要的话 #Start the handle, if needed
ros2 run yahboomcar_ctrl yahboom_joy_R2
ros2 run joy joy_node
```

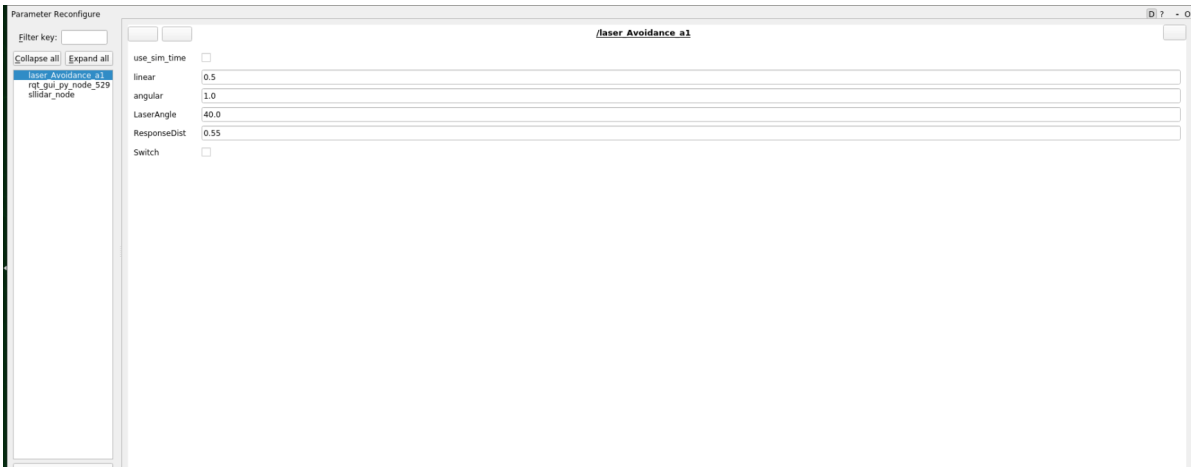
3.2. View the topic communication node map

docker terminal by typing.

```
ros2 run rqt_graph rqt_graph
```

It is also possible to set the size of the parameter, the terminal input, by means of a dynamic parameter regulator, the

```
ros2 run rqt_reconfigure rqt_reconfigure
```



The meaning of each parameter is as follows.

Parameter name	Parameter Meaning
linear	linear velocity
angular	angular velocity
LaserAngle	lidar detection angle
ResponseDist	Obstacle detection distance
Switch	Play switch

The above parameters are adjustable, except Switc, the other four need to be set when you need to be a decimal, modified, click on the blank space before you can write.

4. Core source code analysis

R2 models, A1 lidar source code as an example, mainly to see the lidar callback function, which explains how to get to each angle of the obstacle distance information.

```
def registerScan(self, scan_data):
    if not isinstance(scan_data, LaserScan): return
    ranges = np.array(scan_data.ranges)
```

```
    self.Right_warning = 0
    self.Left_warning = 0
    self.front_warning = 0

    for i in range(len(ranges)):
        angle = (scan_data.angle_min + scan_data.angle_increment * i) * RAD2DEG#
雷达的信息的angle是弧度制，这里要转换成角度进行计算
        #The angle of the lidar's information is in radians, here it is
converted to angles for calculation.
        if 160 > angle > 180 - self.LaserAngle:#angle是根据雷达的结构来设定判断范围的
            #angle sets the range of judgment based on the structure of the
lidar
            if ranges[i] < self.ResponseDist*1.5: #range[i]就是雷达扫描的结果，这里指
得是距离信息
            #range[i] is the result of the lidar scan, in this case the distance
information
                self.Right_warning += 1
        if - 160 < angle < self.LaserAngle - 180:
            if ranges[i] < self.ResponseDist*1.5:
                self.Left_warning += 1
        if abs(angle) > 160:
            if ranges[i] <= self.ResponseDist*1.5:
                self.front_warning += 1
        if self.Joy_active or self.Switch == True:
            if self.Moving == True:
                self.pub_vel.publish(Twist())
                self.Moving = not self.Moving
                    return
        self.Moving = True
```