

README

项目结构

- 【XING_Python_SDK&Demo】2.4.0.3142
 - NOKOV官方提供SDK demo，内附whl依赖和说明，按照要求安装运行即可
- LIMO用户手册.pdf
 - limo小车手册，其中注意如何调节遥控器使得小车是**麦克纳姆模式**
 - 建议看附录2, 3, 4 ros相关
 - 1.9 远程桌面连接，使用**NoMachine**远程连接小车
- xing_merge.py
 - 获取SDK数据、控制小车文件，需要安装上面提到的whl依赖

整体框架

使用NOKOV动捕设备获取相关参数，使用Ubuntu18.04安装ROS1控制小车并接收SDK数据

NOKOV, Ubuntu18.04, limo小车需在同一网段下，实验室已保证，**只需Ubuntu和小车都连接到epuck wifi即可**

相关细节

ROS1安装

[小鱼的一键安装系列](#) | [鱼香ROS \(fishros.org.cn\)](http://fishros.org.cn)

wsl控制相关

关闭防火墙

一定要关闭防火墙，终端中执行

```
sudo ufw disable
```

之后根据ip互相ping进行测试

limo小车系统

用户名：agilex

登录密码：agx

limo端开启底盘

在limo小车ubuntu上操作

终端中运行

```
roslaunch limo_base limo_base.launch
```

即可完成设置

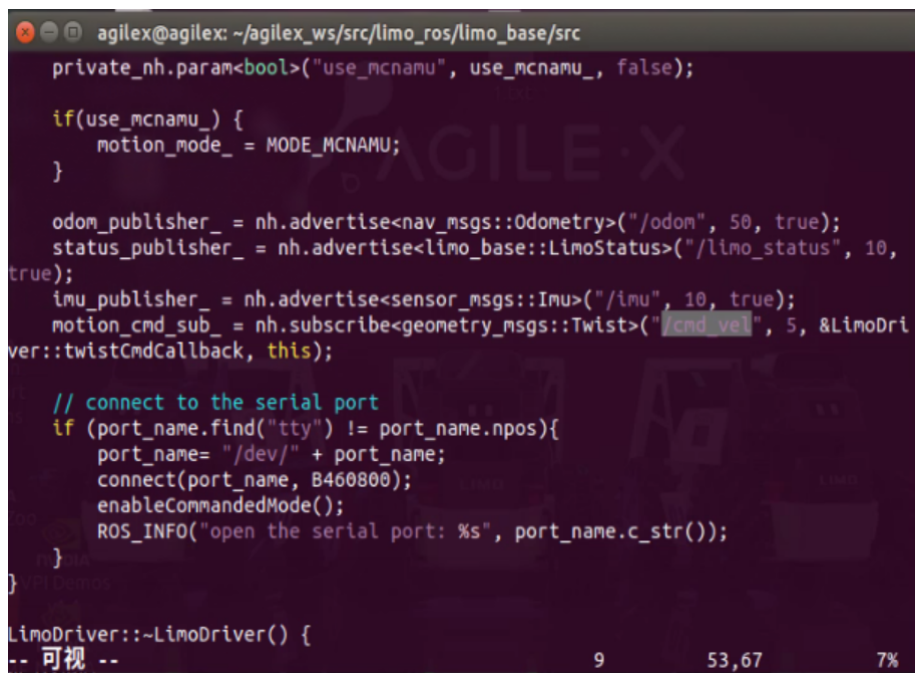
修改limo车名

在limo小车ubuntu上操作

1. 打开limo_driver.cpp文件，路径在：

```
/home/agilex/agilex_ws/src/limo_ros/limo_base/src
```

找到文件中的**motion_cmd_sub_**，修改后面的"/cmd_vel"改为"/xxxx"，xxxx为自己设置的名字，即下图高亮部分



```
agilex@agilex: ~/agilex_ws/src/limo_ros/limo_base/src
private_nh.param<bool>("use_mcnamu", use_mcnamu_, false);

if(use_mcnamu_) {
    motion_mode_ = MODE_MCNAMU;
}

odom_publisher_ = nh.advertise<nav_msgs::Odometry>("/odom", 50, true);
status_publisher_ = nh.advertise<limo_base::LimoStatus>("/limo_status", 10,
true);
imu_publisher_ = nh.advertise<sensor_msgs::Imu>("/imu", 10, true);
motion_cmd_sub_ = nh.subscribe<geometry_msgs::Twist>("/cmd_vel", 5, &LimoDriver::twistCmdCallback, this);

// connect to the serial port
if (port_name.find("tty") != port_name.npos){
    port_name= "/dev/" + port_name;
    connect(port_name, B460800);
    enableCommandedMode();
    ROS_INFO("open the serial port: %s", port_name.c_str());
}
}

LimoDriver::~LimoDriver() {
-- 可视 --          9          53,67          7%
```

2. 回到agilex_ws文件夹，进行编译

```
cd ~/home/agilex/agilex_ws
catkin_make
```

3. 重新启动小车ros master

```
roslaunch limo_base limo_base.launch
```

如果之前开启了ros master，必须关闭后重启才能使用修改后的名字进行控制

多车连接（暂时是一个，多车还没试过）

ROS是一个分布式系统框架，因此有必要介绍一下分布式多机通信的设置方法。分布式一般都是主从（master/slave）方式，因此需要两端都要进行设置。

作为示例的两台机器如下：

- master: 192.168.1.100
- client: 192.168.1.200

master及client分别为各自的hostname, 想要查询自己的hostname也非常简单，就使用hostname命令。

Master端设置

- 修改hosts文件

```
sudo gedit /etc/hosts
```

添加:

```
192.168.1.100 master
192.168.1.200 client
```

Client端设置

- 修改hosts文件, 方法与master相同

添加:

```
192.168.1.100 master
192.168.1.200 client
```

- 设置环境变量ROS_MASTER_URI

```
sudo gedit ~/.bashrc
```

添加

```
export ROS_MASTER_URI=http://192.168.1.100:11311
```

接下来在家目录终端中输入

```
source ~/.bashrc
```

测试

这个仅为测试, 打开limo小车master参照上文

在master上打开roscore, 并发送一条消息

```
$ rostopic pub /test_topic std_msgs/String "connected"
```

在client上看是否能收到消息

```
$ rostopic echo /test_topic
data: "connected"
---
```

启动车底盘

终止之前的roscore，在每个车上执行下面指令（还没有试过），这样才能保证能被控制，如果只是roscore小车无法移动

```
roslaunch limo_base limo_base.launch
```

对于多个小车，**只需要有一个master**，其他的按照client端设置即可，具体的名字通过hostname获取
同时在控制的ubuntu机器上也需要按照client的要求进行配置，但是不必开启roscore了