



innopolis  
UNIVERSITY

# System Identification and Simulation and Intelligent Mobile Robotics Coursework

Autonomous car-like robot navigation  
Sensor fusion for robot pose estimation

Filipenko Maksim, MS-AIR

# Problem

Inability to build a map from onboard sensor data and navigate in unknown environment while trying to estimate robot's position precisely.

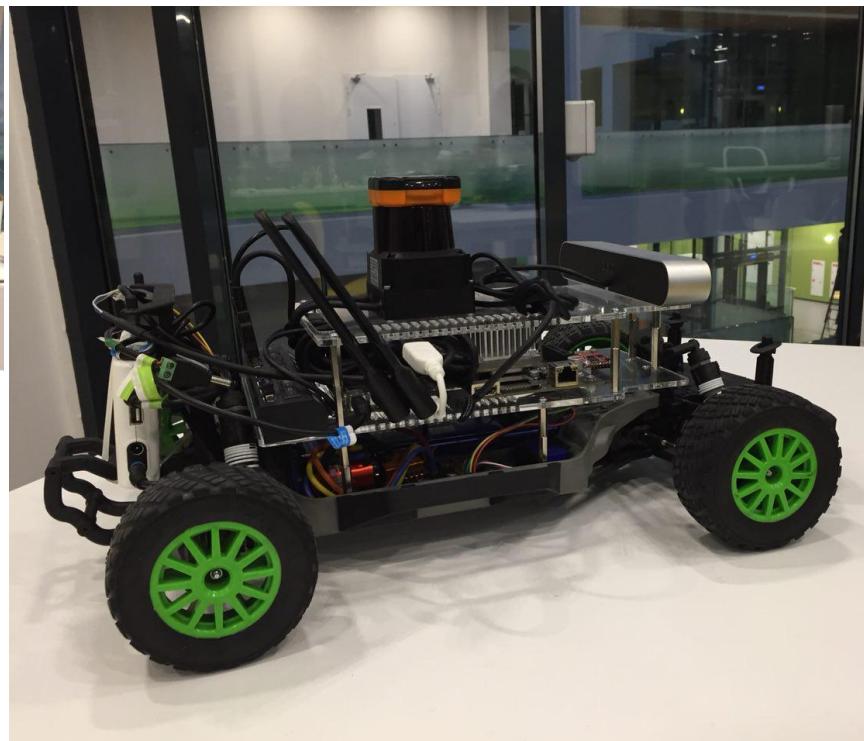
# Idea

All the sensors has strong and weak sides - can't estimate the position with the results of one of the sensors.

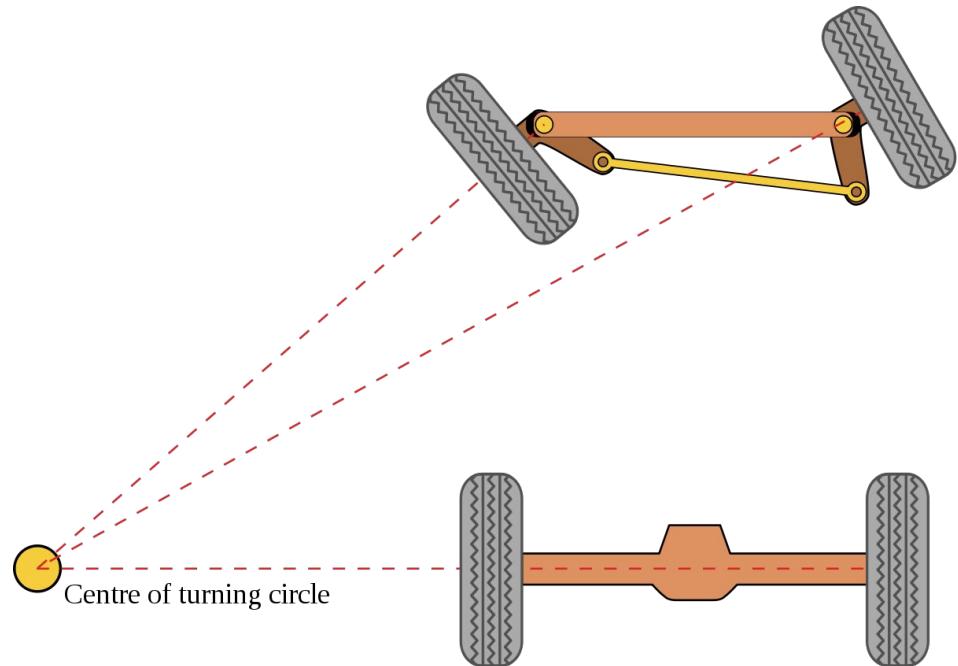
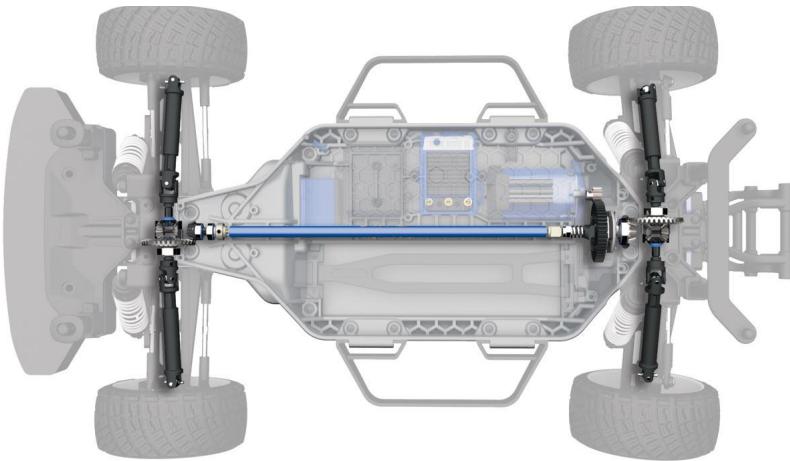
We can use multiple sensors and make strong sides of one sensors overcome weak sides of another one.

We are conducting experiments creating autonomous navigation for wheeled robot car

# Experimental setup - Labcar



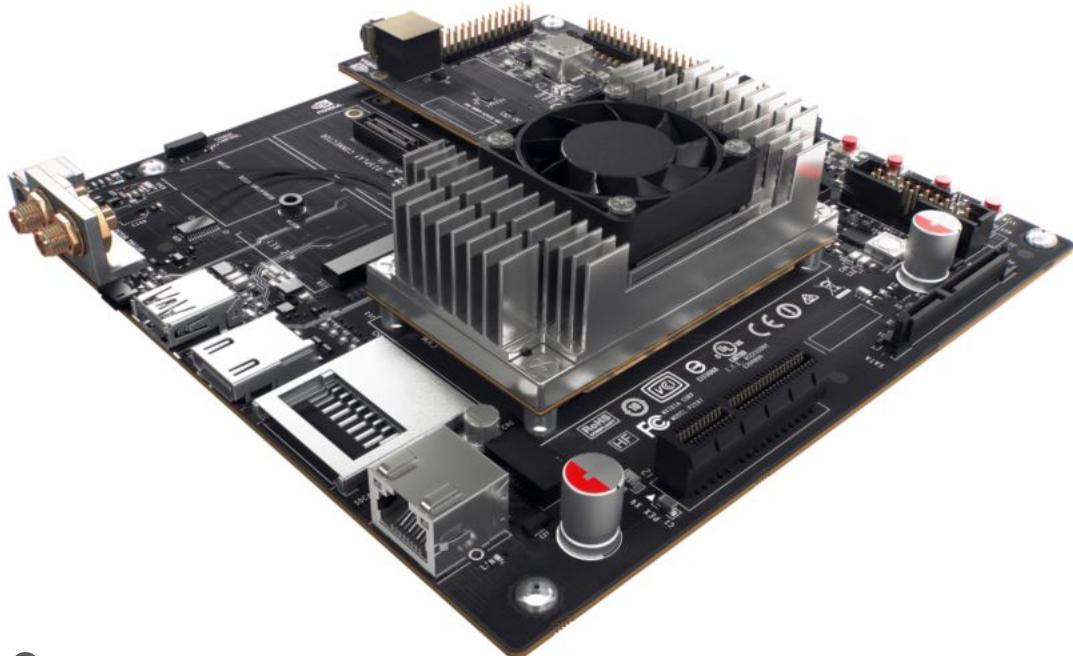
# Hardware - 4WD Traxxas #74076



- Efficient Shaft-Driven 4WD
- Low-CG Chassis

# Hardware - Nvidia Jetson TX1

Has custom Linux 14.04 core and custom ROS kinetic



- NVIDIA Maxwell ™
- 256 CUDA cores
- Quad ARM® A57
- Video: 4K x 2K 60 Hz
- Memory: 4 GB 64 bit LPDDR4 25.6 GB/s
- Other: UART, SPI, I2C, I2S, GPIOs

# Hardware - Driver & Motor SK-300058-04

Controlled in ROS by ackermann\_drive, reads messages from /ackermann\_msgs topic



- 60A, up to 380A bursts
- 1870 kV
- 17.5T

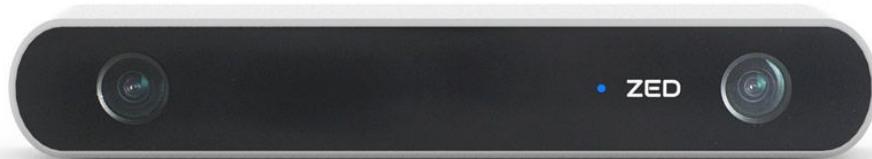
# Hardware - Hokuyo UTM-30LX



- 100mm to 30m
- 25 msec/scan
- 270° area scanning range

Controlled in ROS by hokuyo\_node, publishing messages to /scan topic

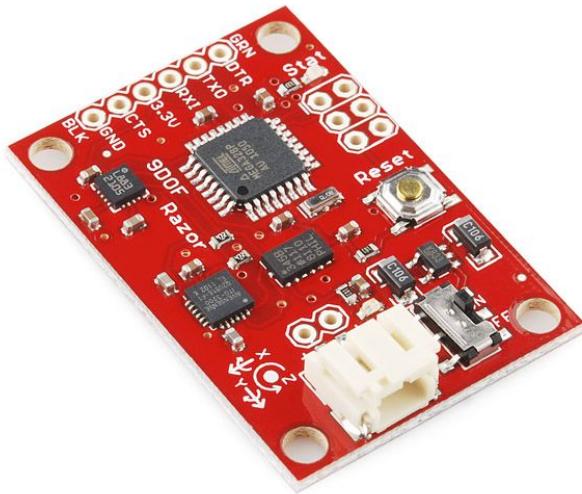
# Hardware - Stereolabs ZED camera



- High-Res and High Frame-rate 3D Video Capture
- Depth Perception at up to 20m
- 6-DoF Positional Tracking
- Spatial Mapping

Controlled in ROS by zed\_wrapper\_node, publishing messages to /camera topics

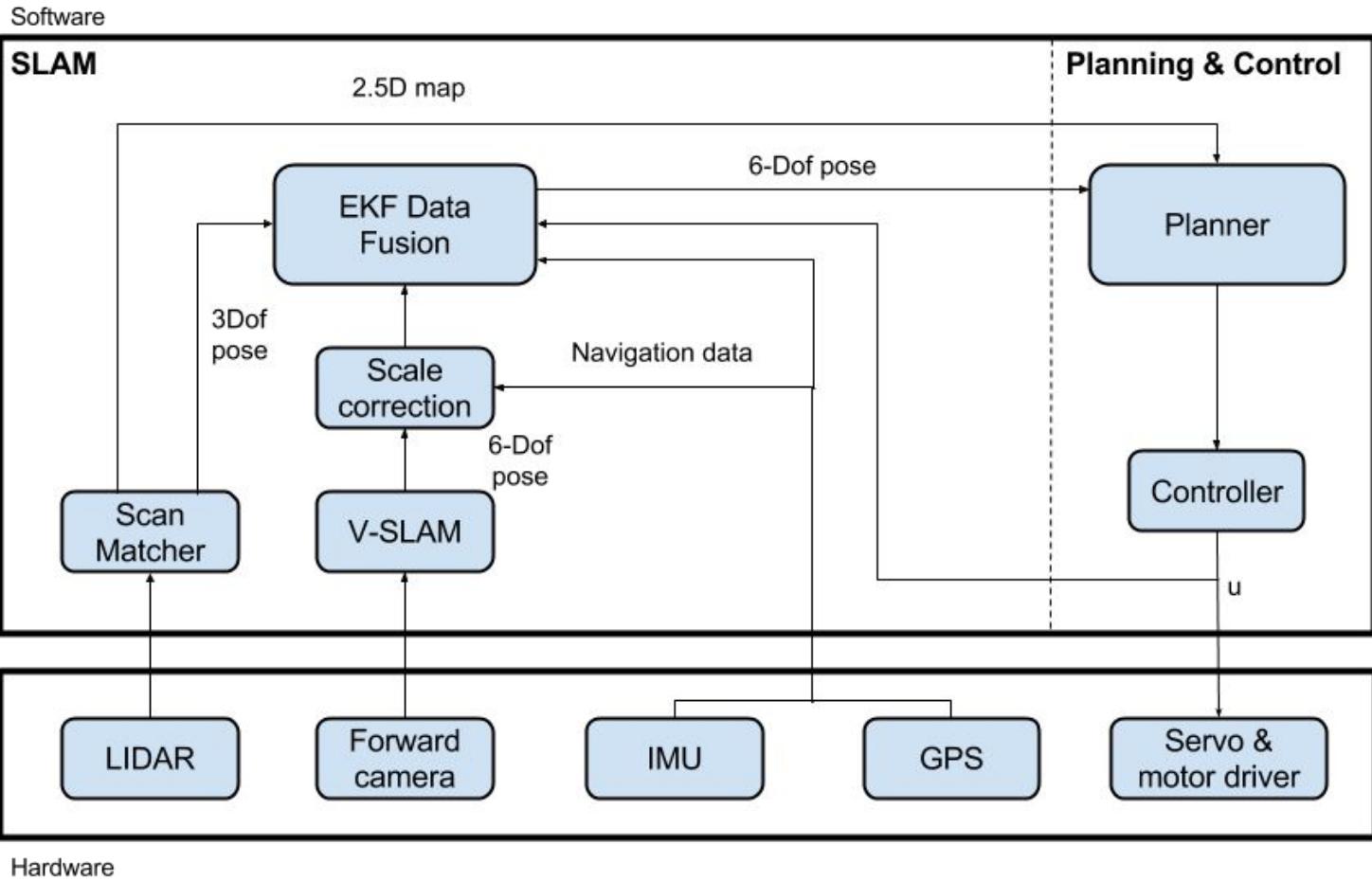
# Hardware - Razor IMU 9DoF SEN-10736



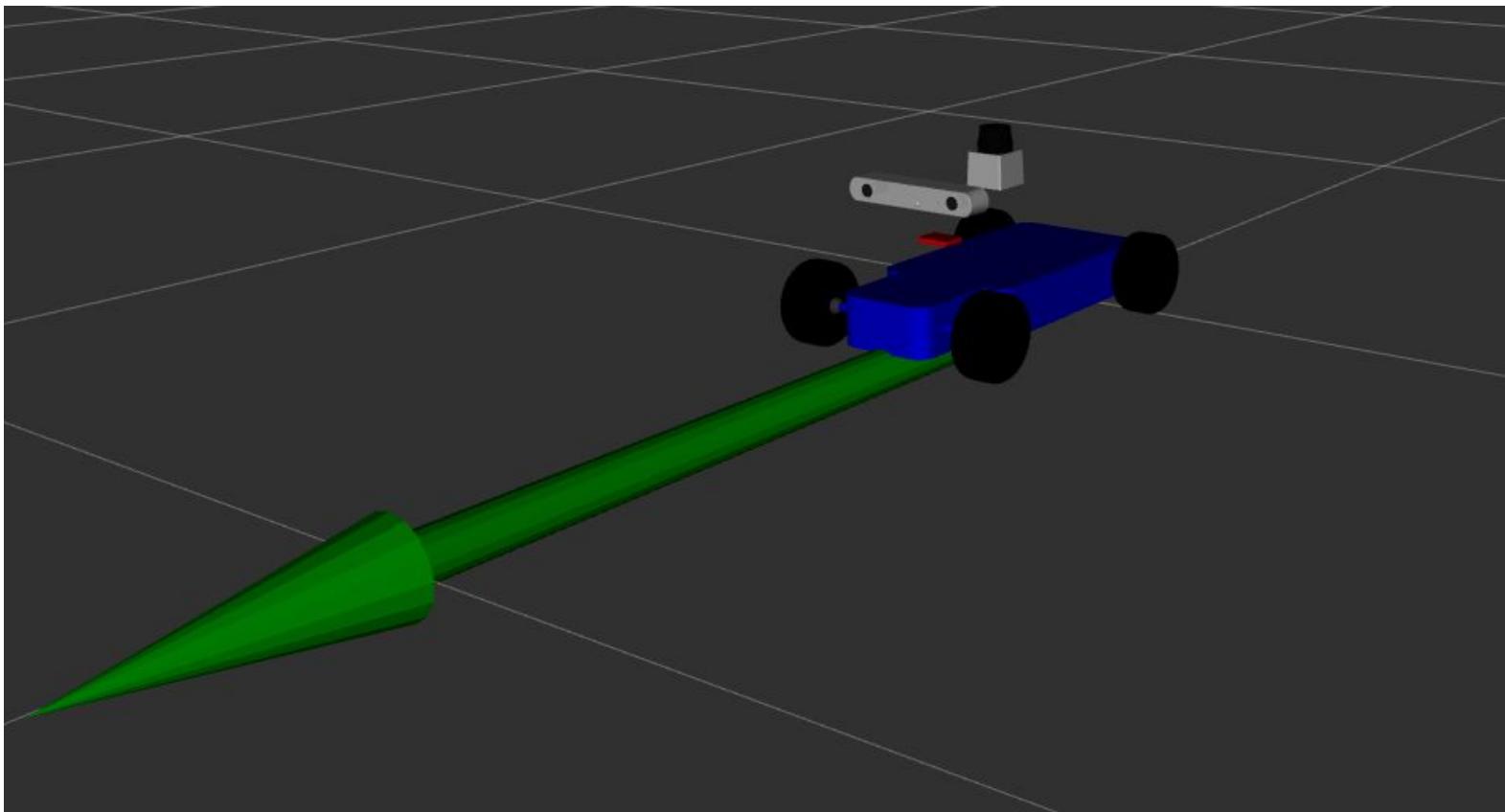
- triple-axis digital gyroscope,
- ±16g triple-axis accelerometer,
- triple-axis digital magnetometer

Controlled in ROS by `razor_imu_9dof`, publishing messages to `/imu` topics

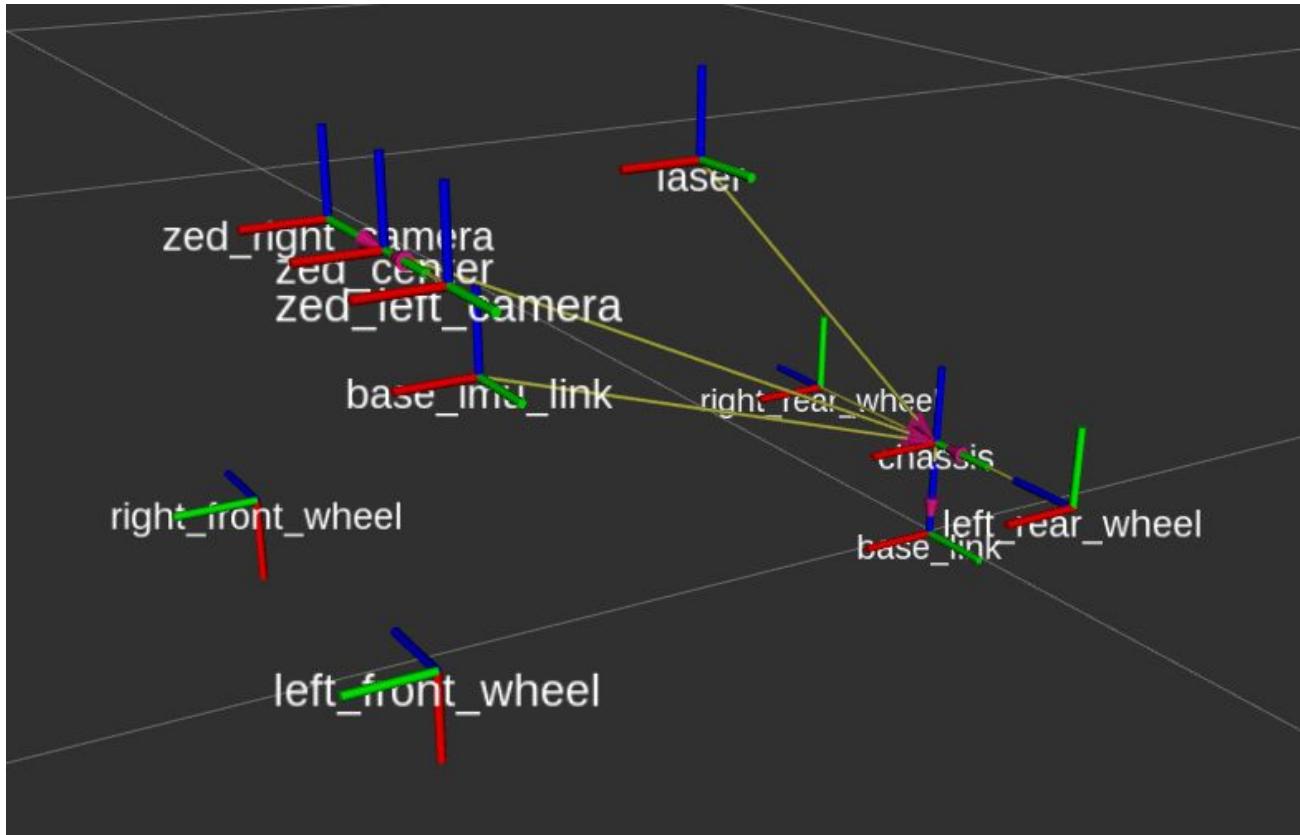
# General idea



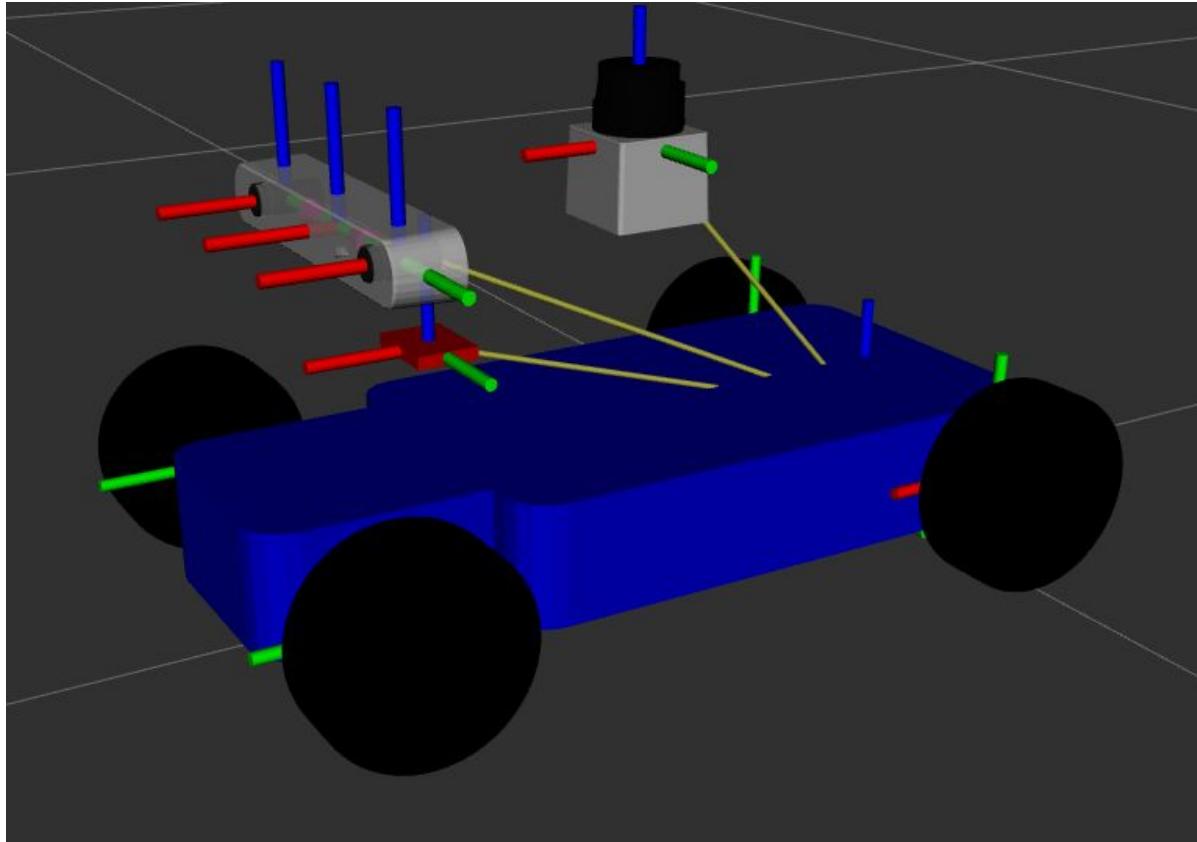
# Robot Modeling



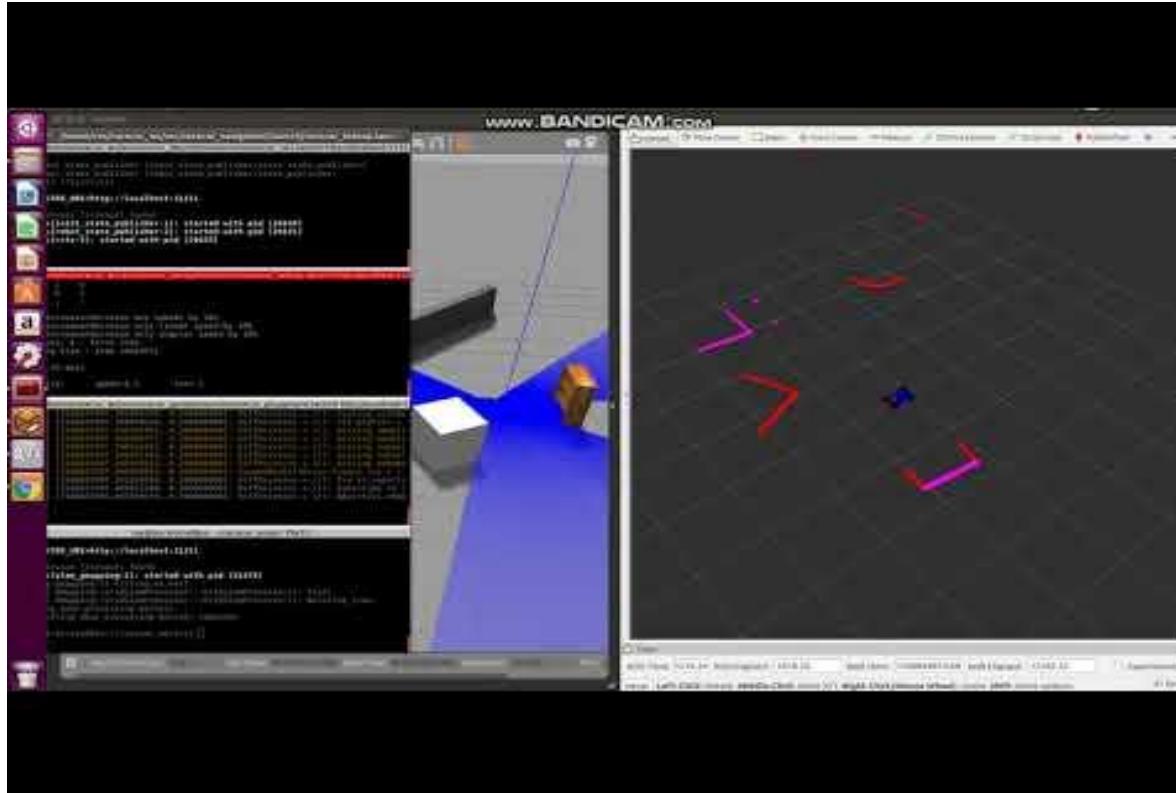
# Robot Modeling



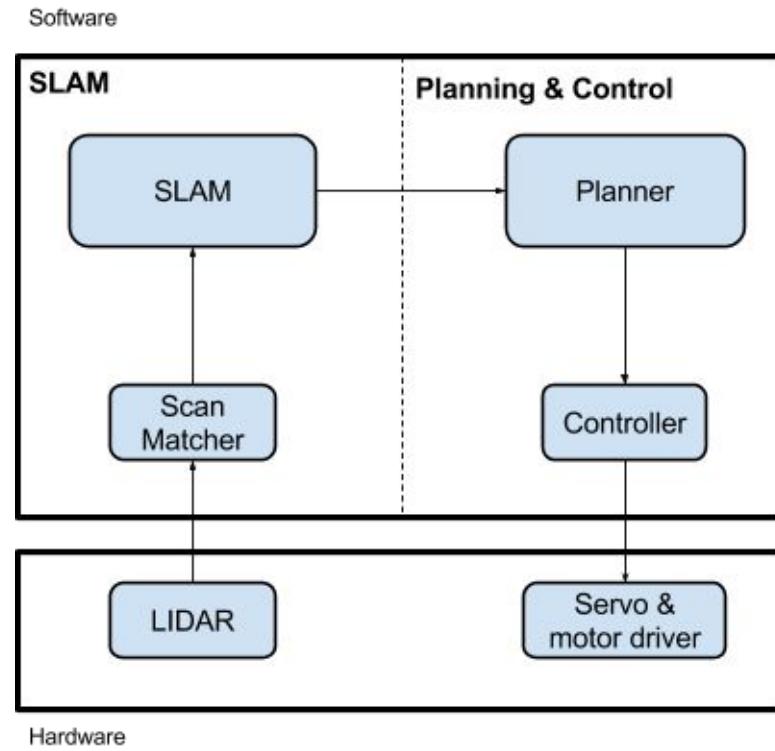
# Robot Modeling



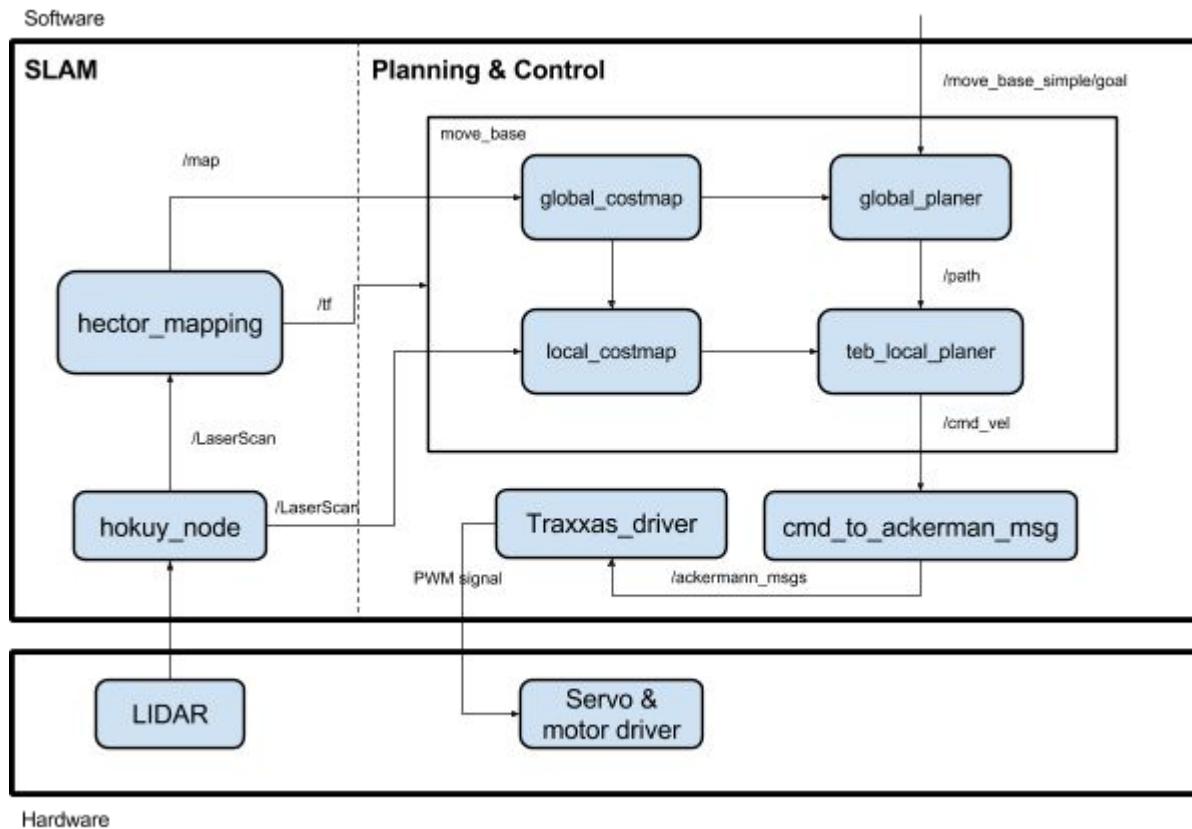
# Robot Modeling - Gazebo simulation (video)



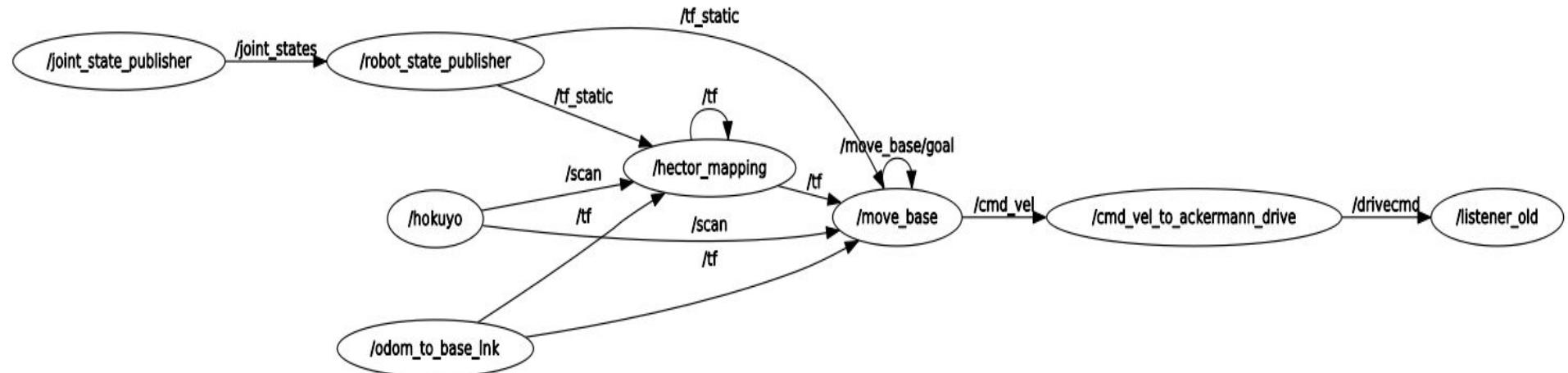
# Autonomous navigation - architecture



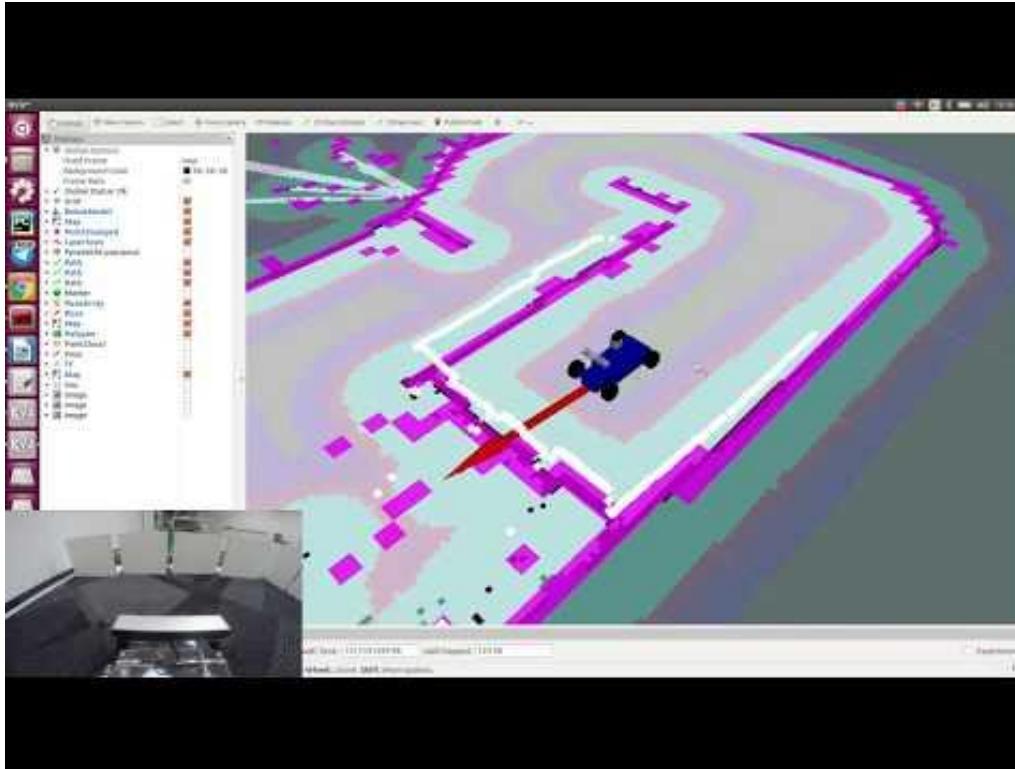
# Autonomous navigation - ROS architecture



# Autonomous navigation - ROS topics exchange

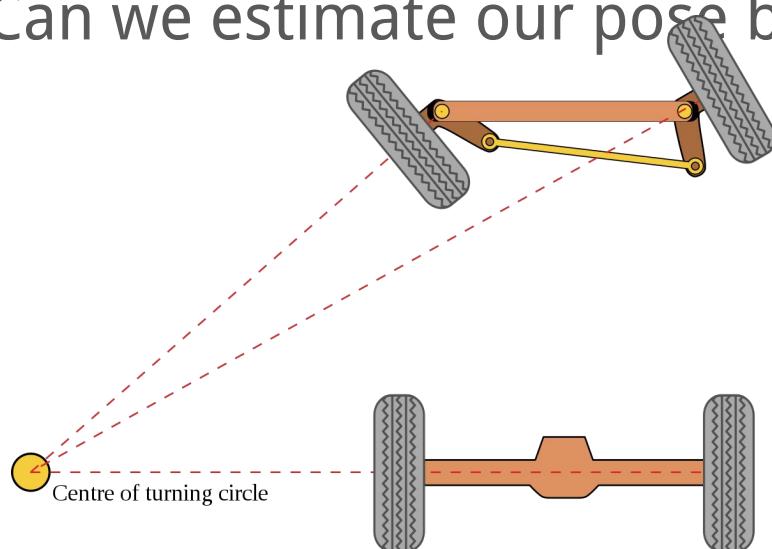


# Autonomous navigation - Result (video)

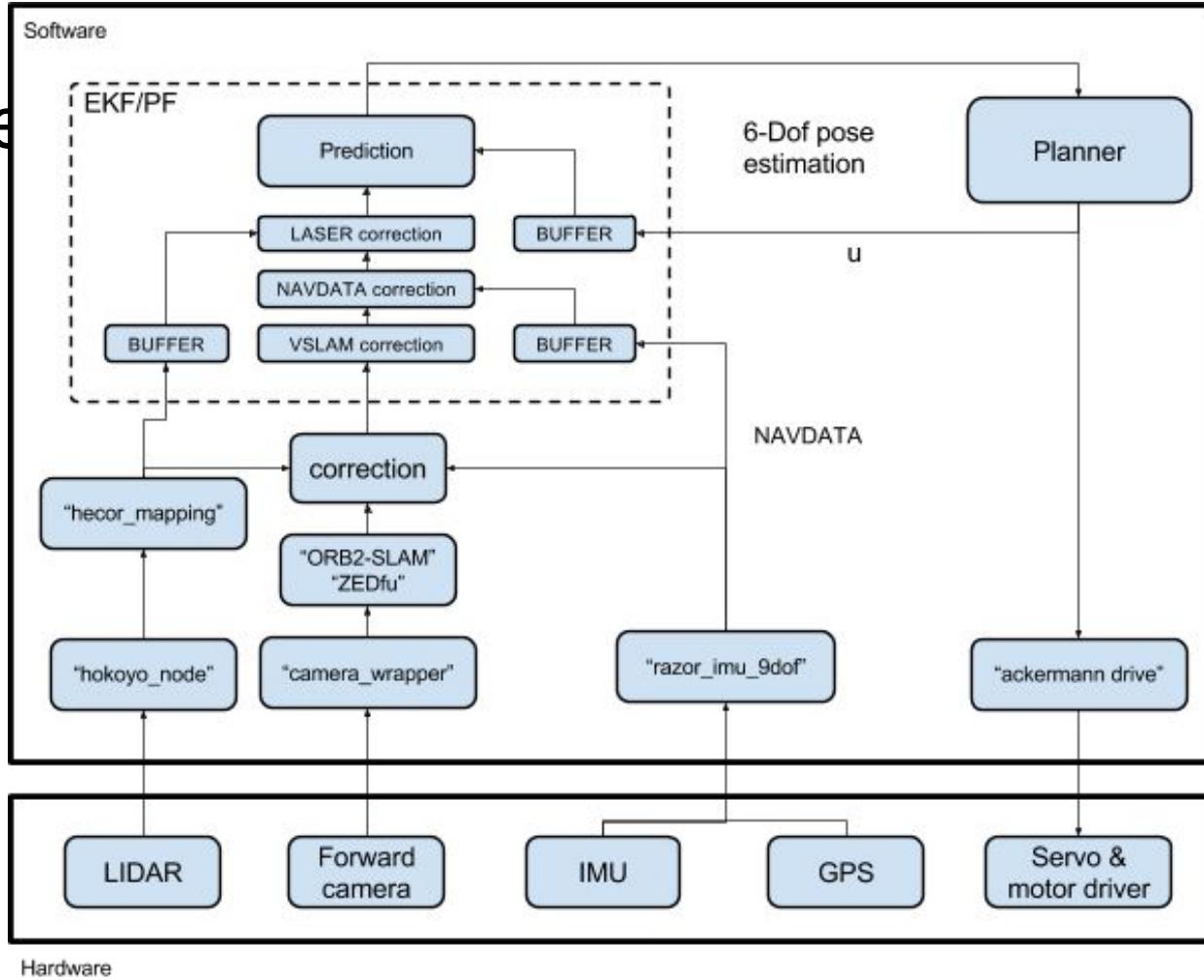


# Sensor fusion

Can we estimate our pose be



# Architecture



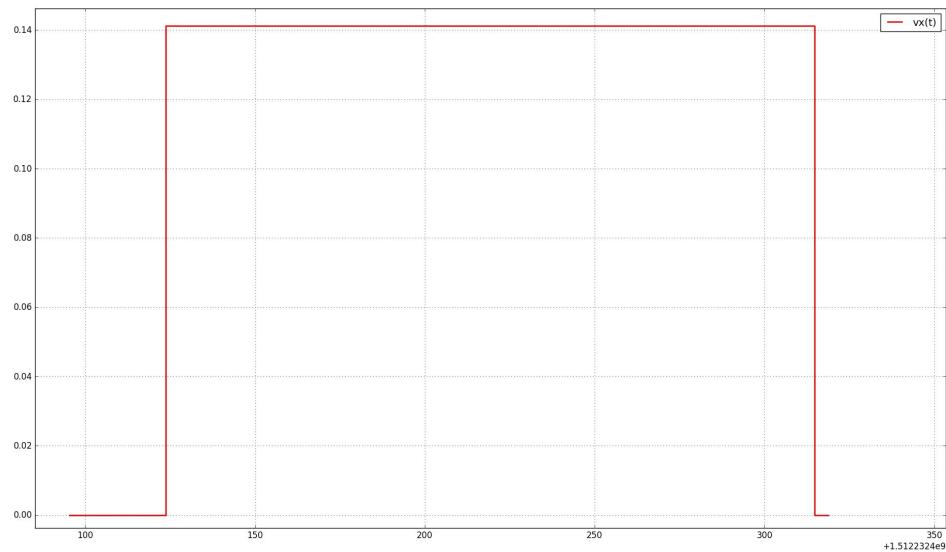
# Sensor calibration (video)



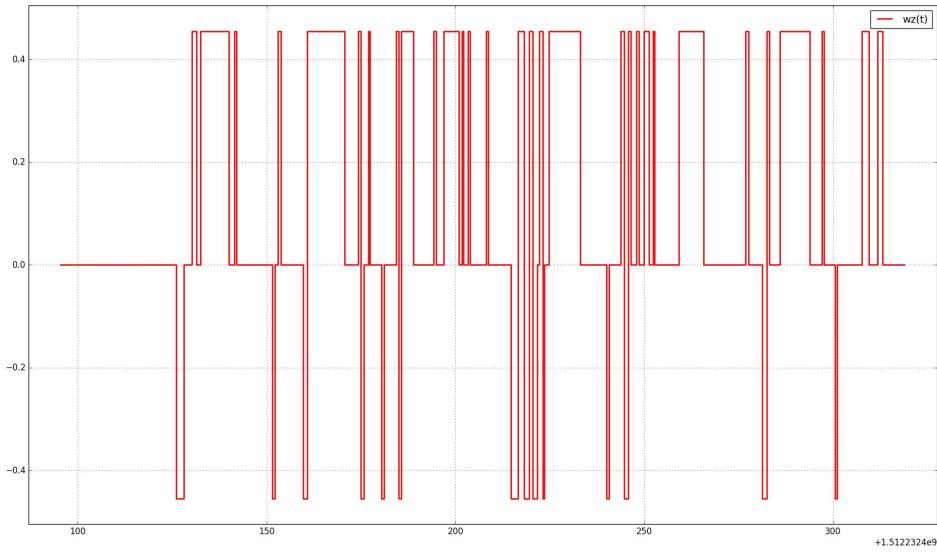
# Sensor fusion - Test environment (video)



# Data - Control commands

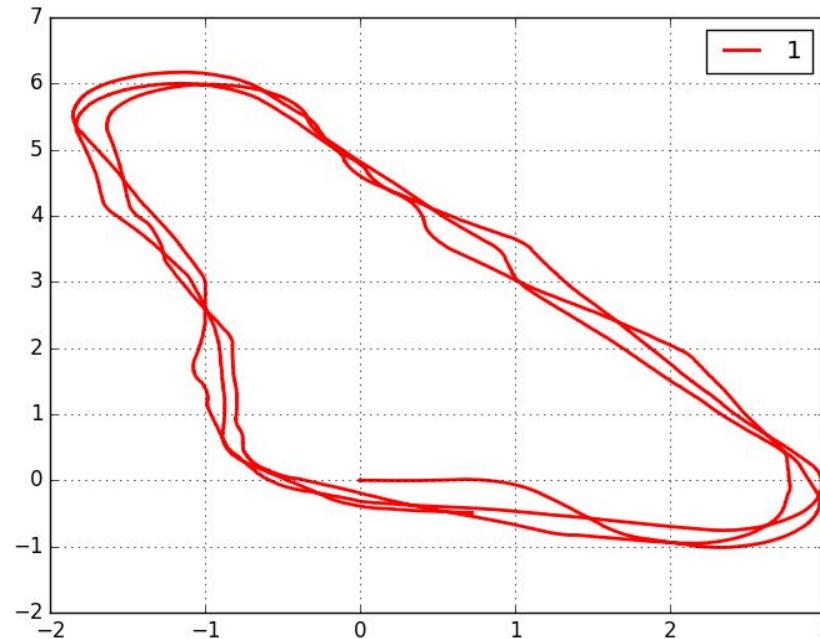


$vx(t)$  - linear speed in coordinate frame attached to the robot.



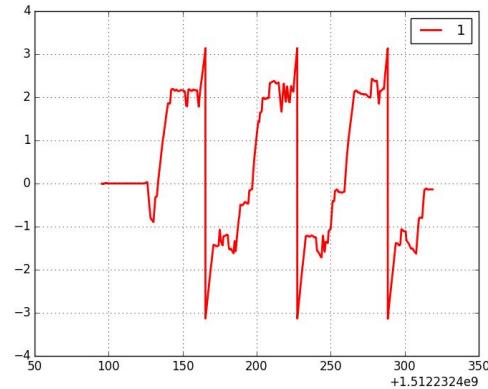
$wz(t)$  - speed of rotation.

# Data - Hector SLAM - LIDAR

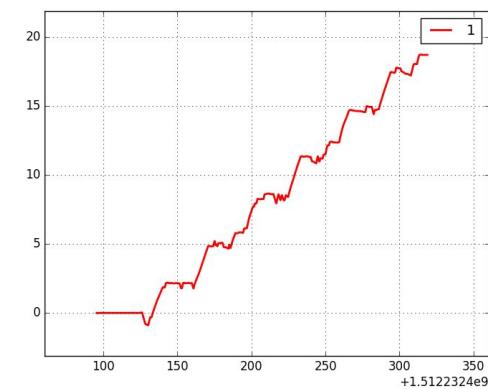


The trajectory of the robot (x,y).

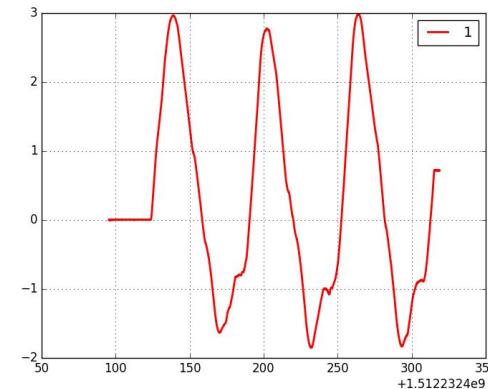
# Data - LIDAR hector\_SLAM



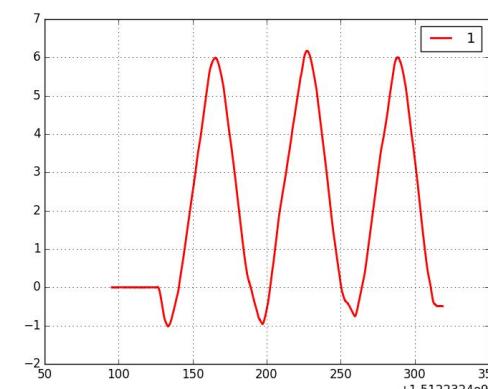
yaw(t) normalized



yaw(t)

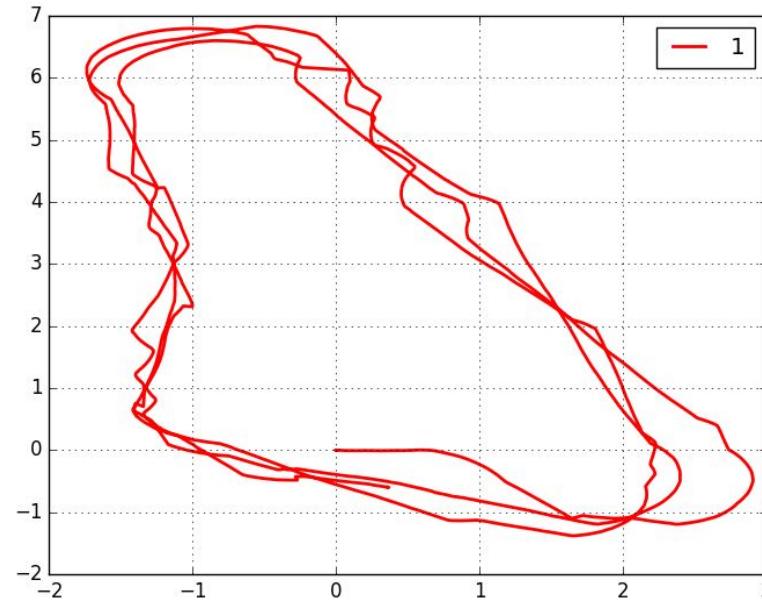


x(t)



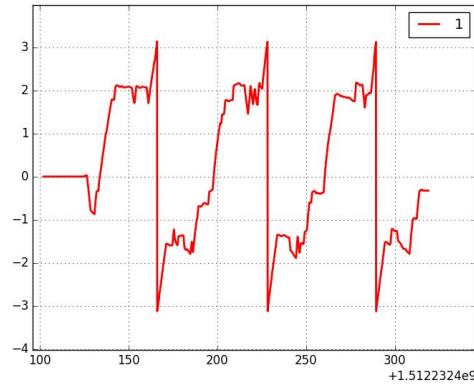
y(t)

# Data - ZED camera (ZEDfu)

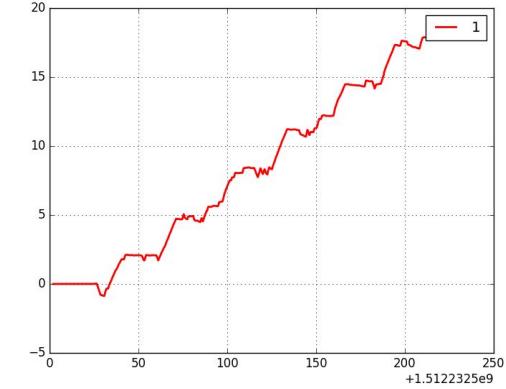


The trajectory of the robot (x,y)

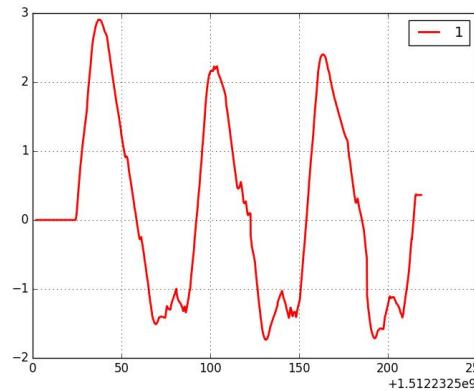
# Data - ZEDfu ZED camera



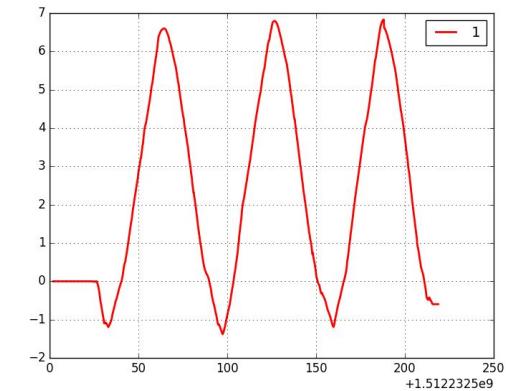
yaw(t) normalized



yaw(t)

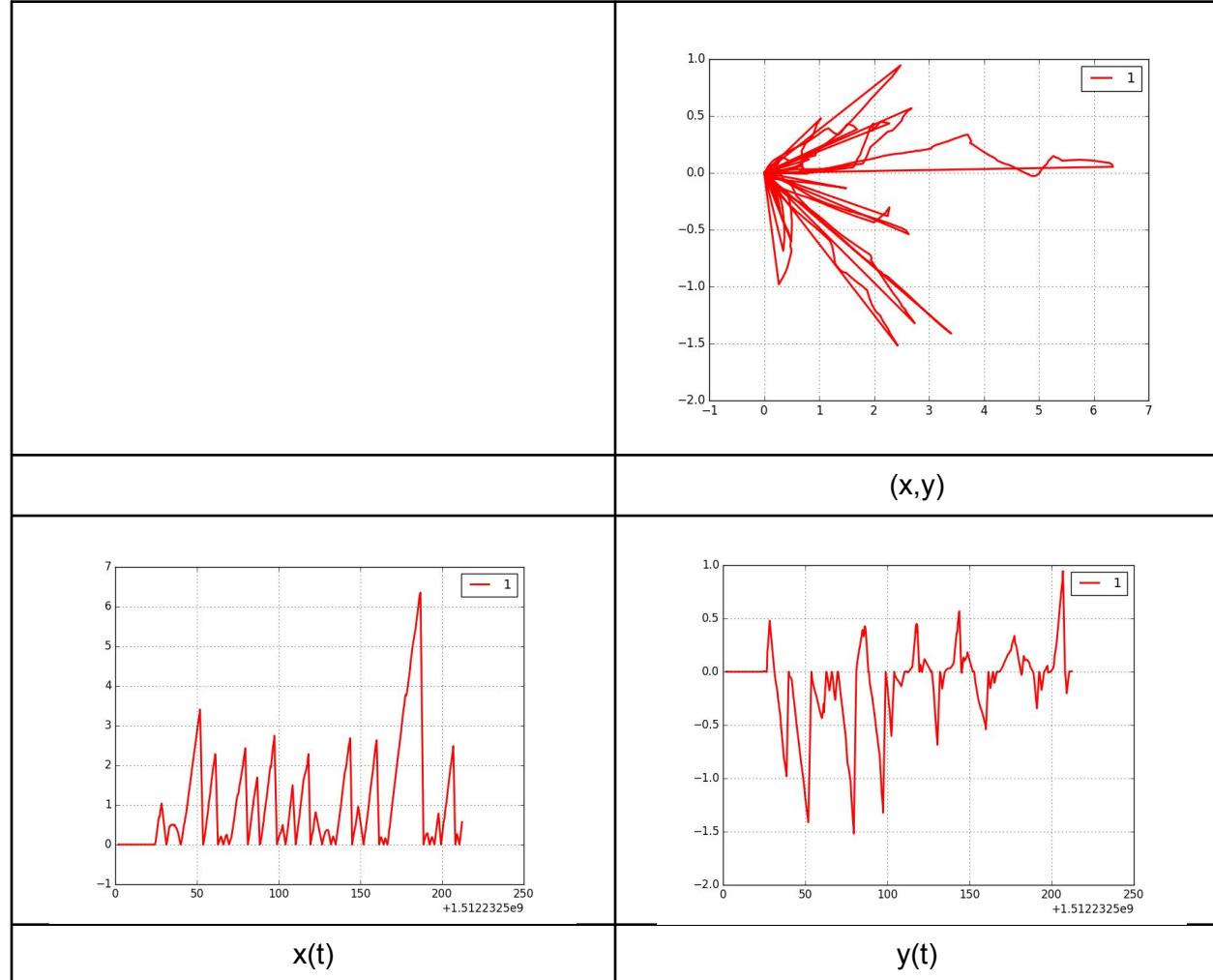


x(t)

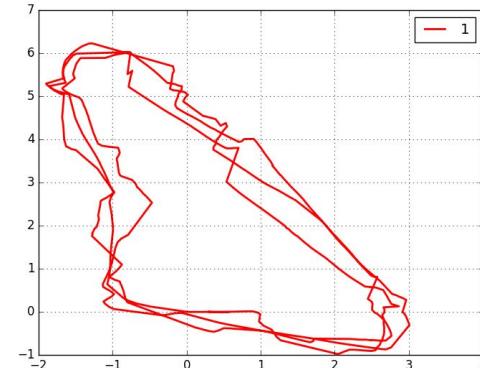
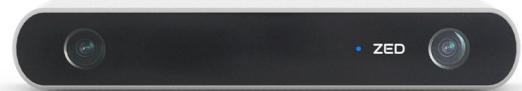


y(t)

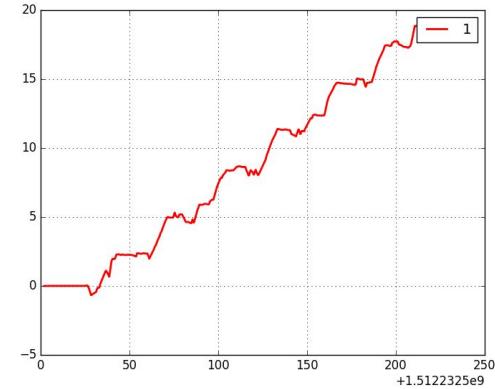
# Data - ORB SLAM ZED camera



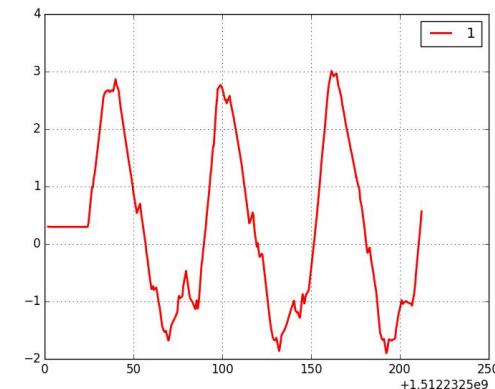
# Data - ORB SLAM ZED camera



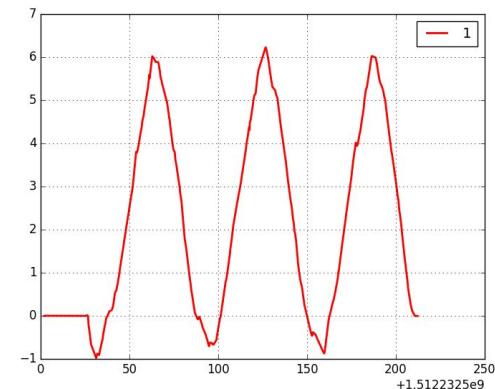
(x,y)



yaw(t)

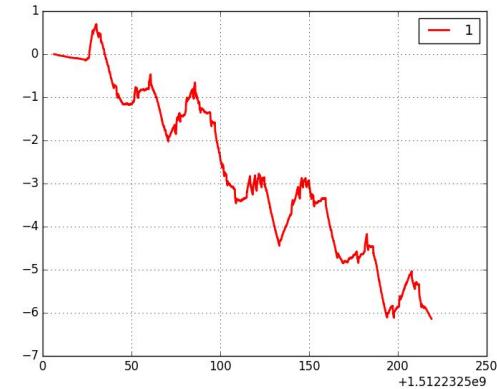
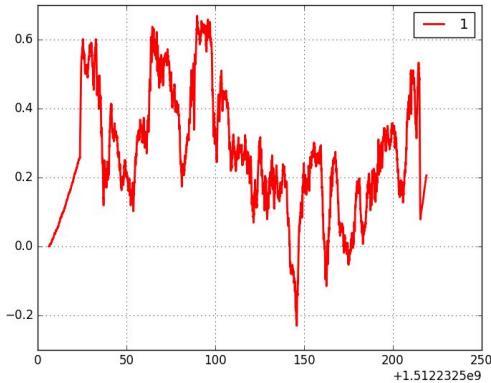
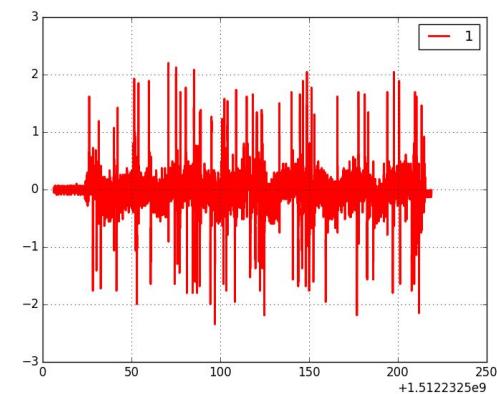
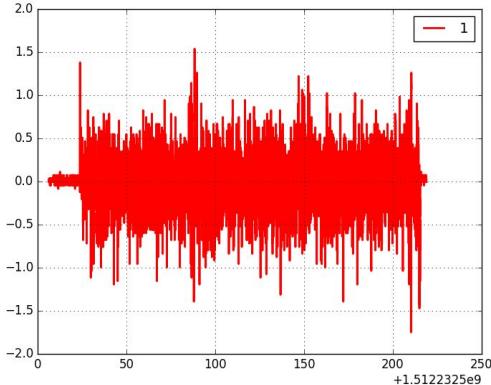


x(t)

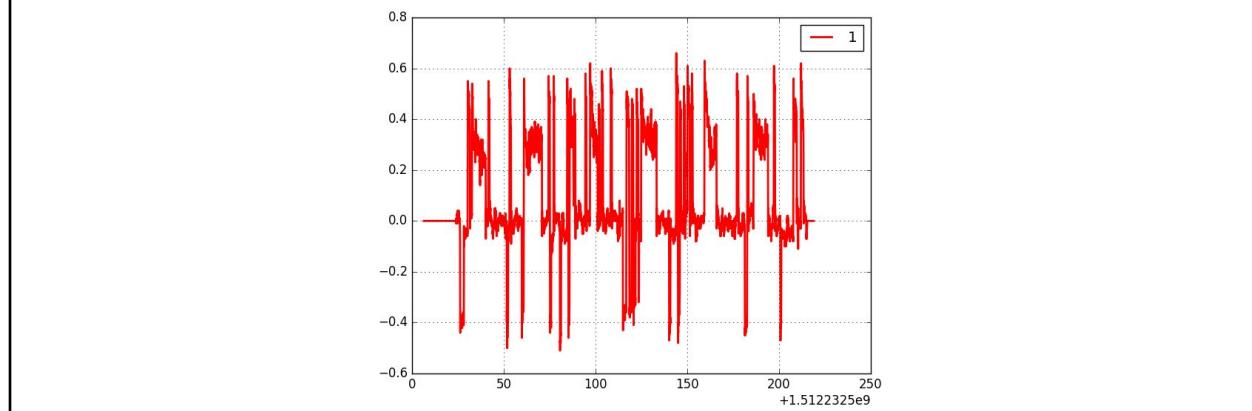
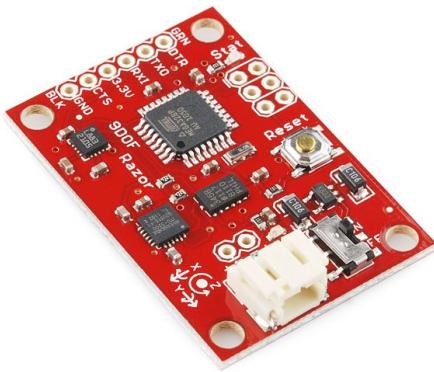


y(t)

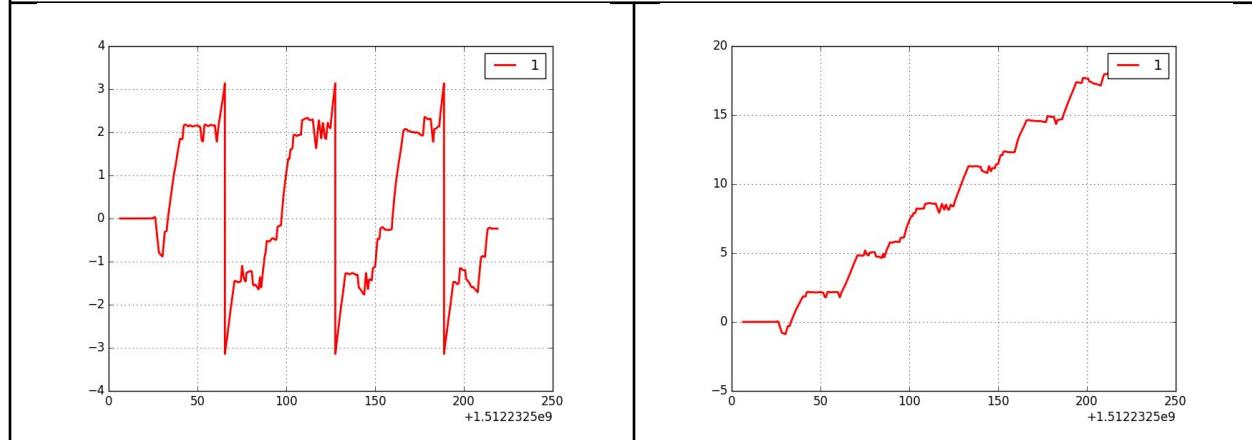
# Data - accelerometer



# Data - gyroscope



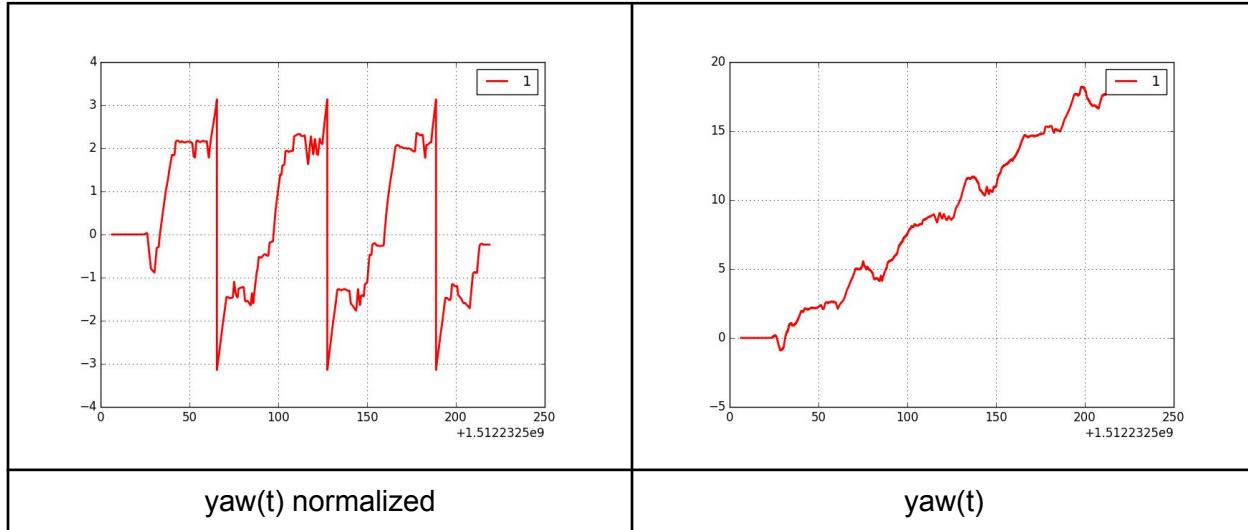
$w_z(t)$  - speed of rotation.



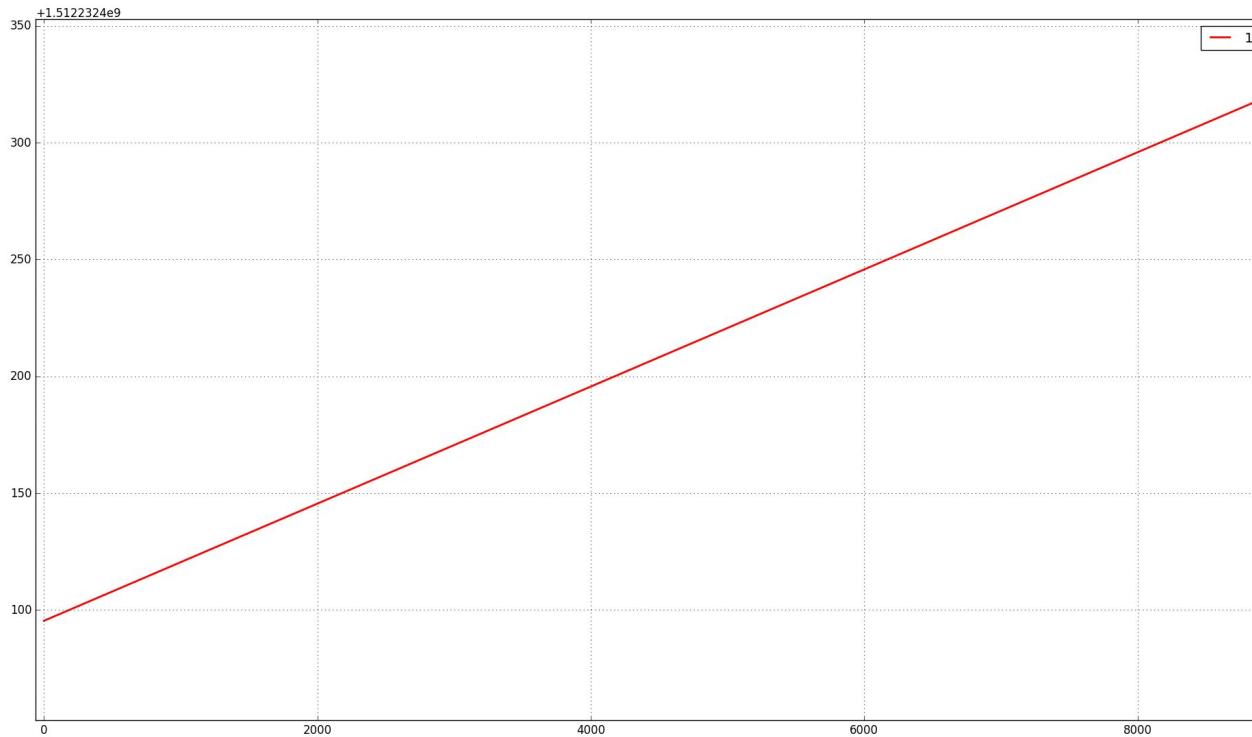
$yaw(t)$  - numerical first integral with normalization

$yaw(t)$  - numerical first integral

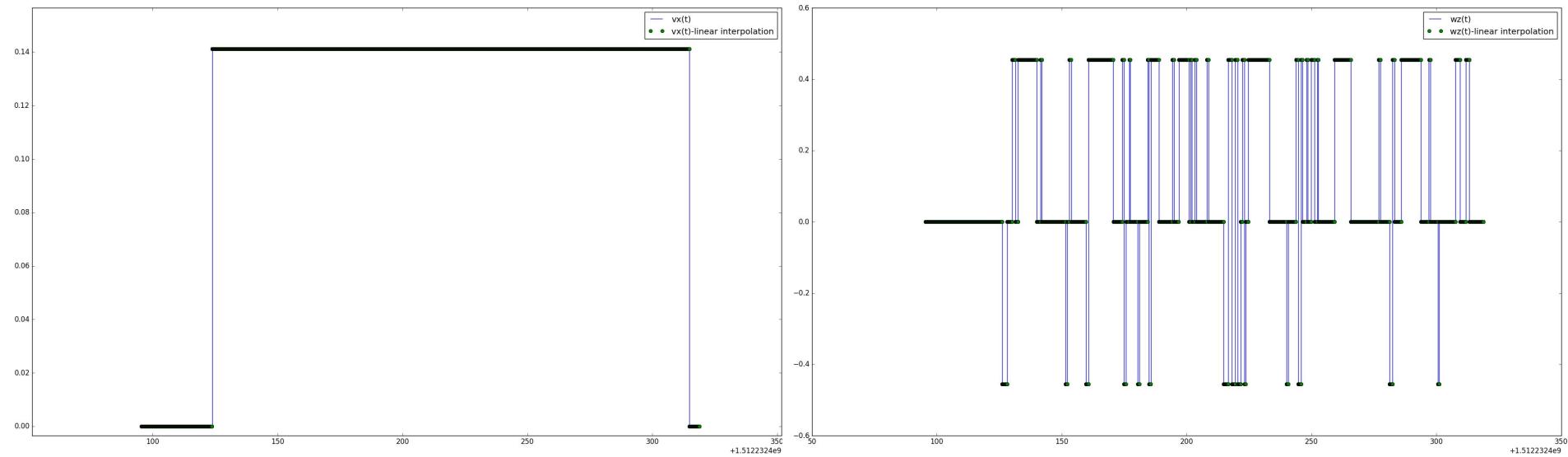
# Data - magnetometer



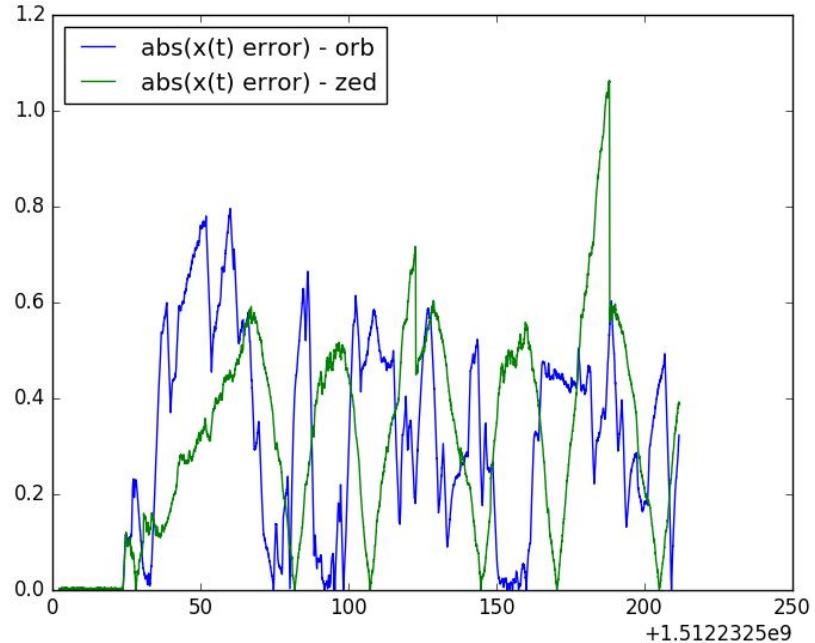
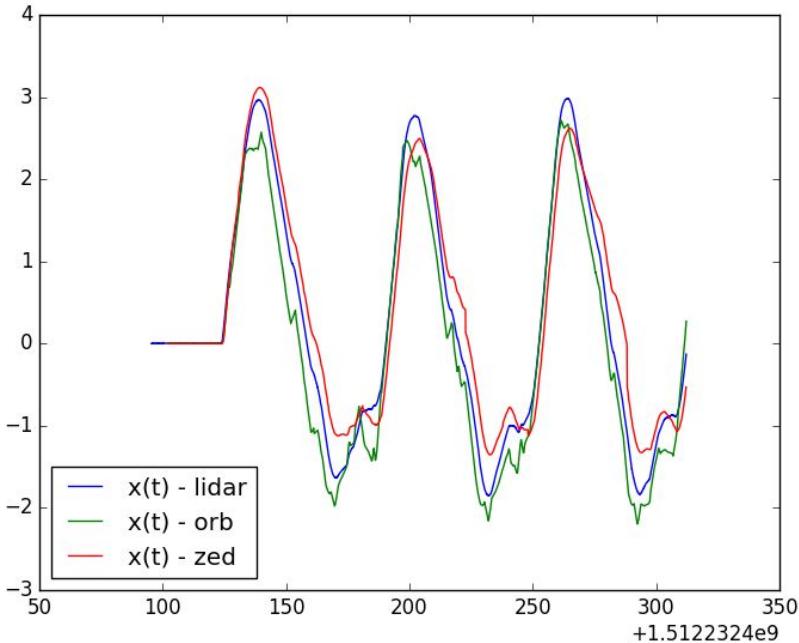
# Data interpolation



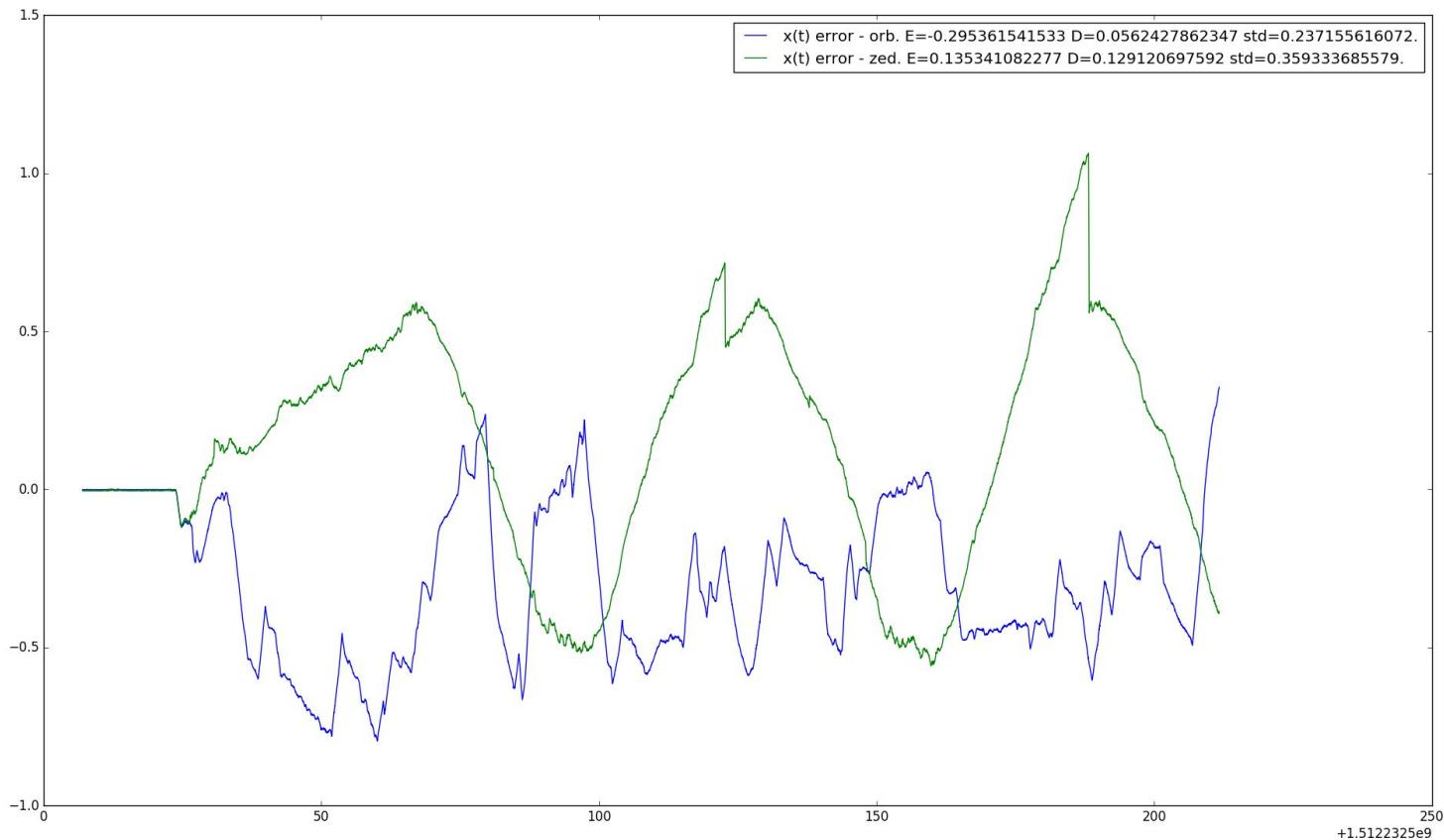
# Control command interpolation



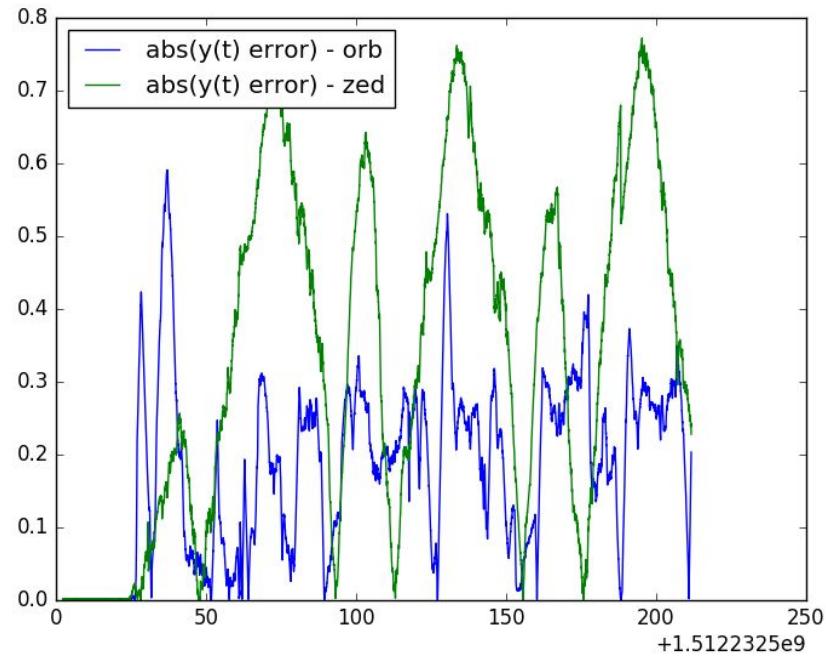
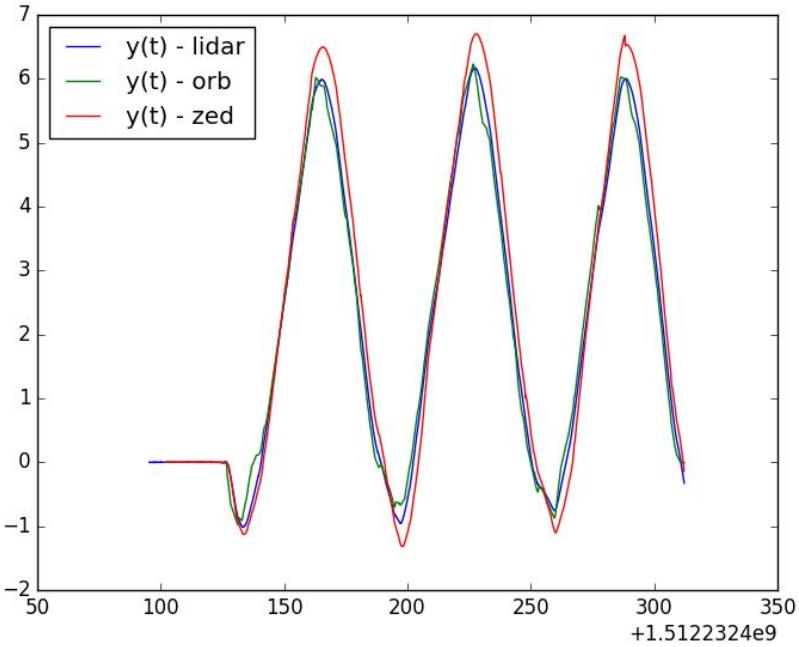
# Comparing - $x(t)$



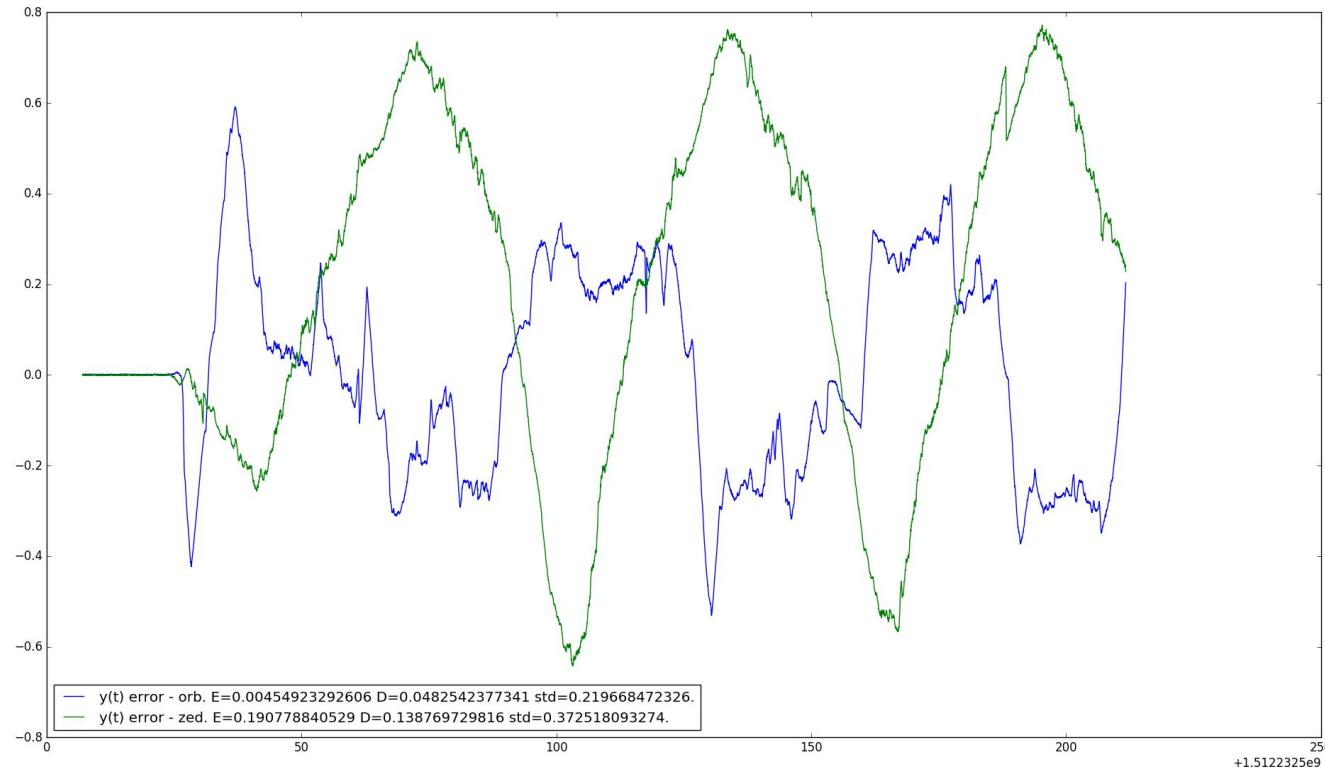
# Comparing - x(t)



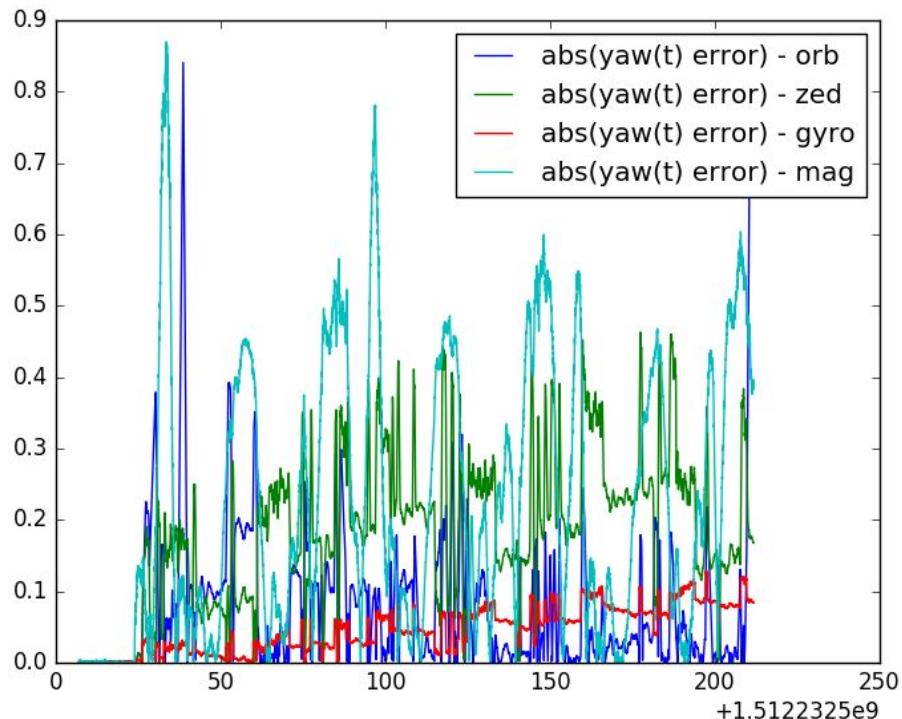
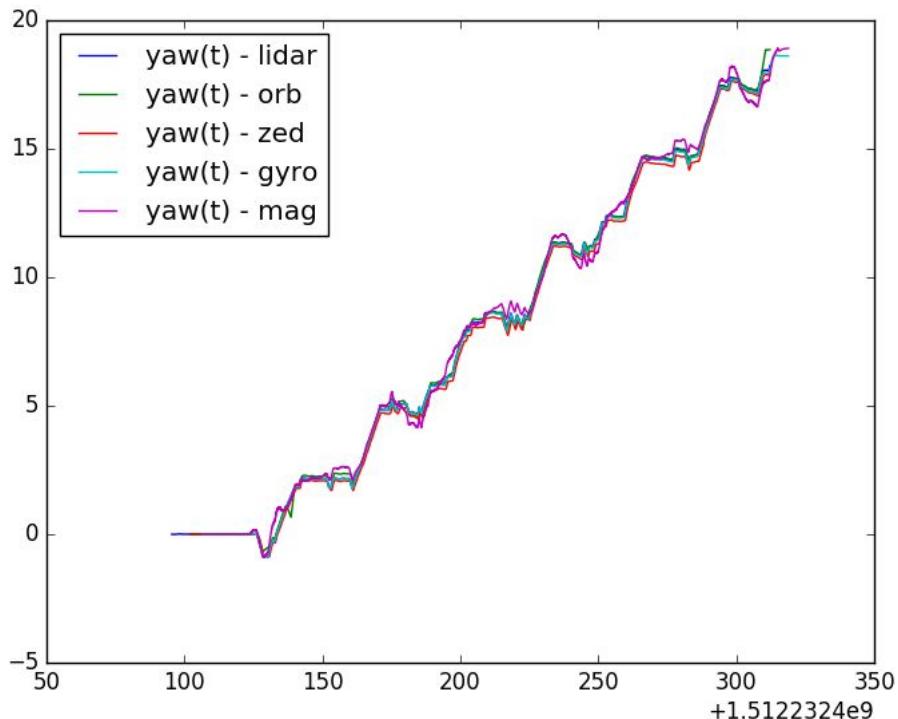
# Comparing - $y(t)$



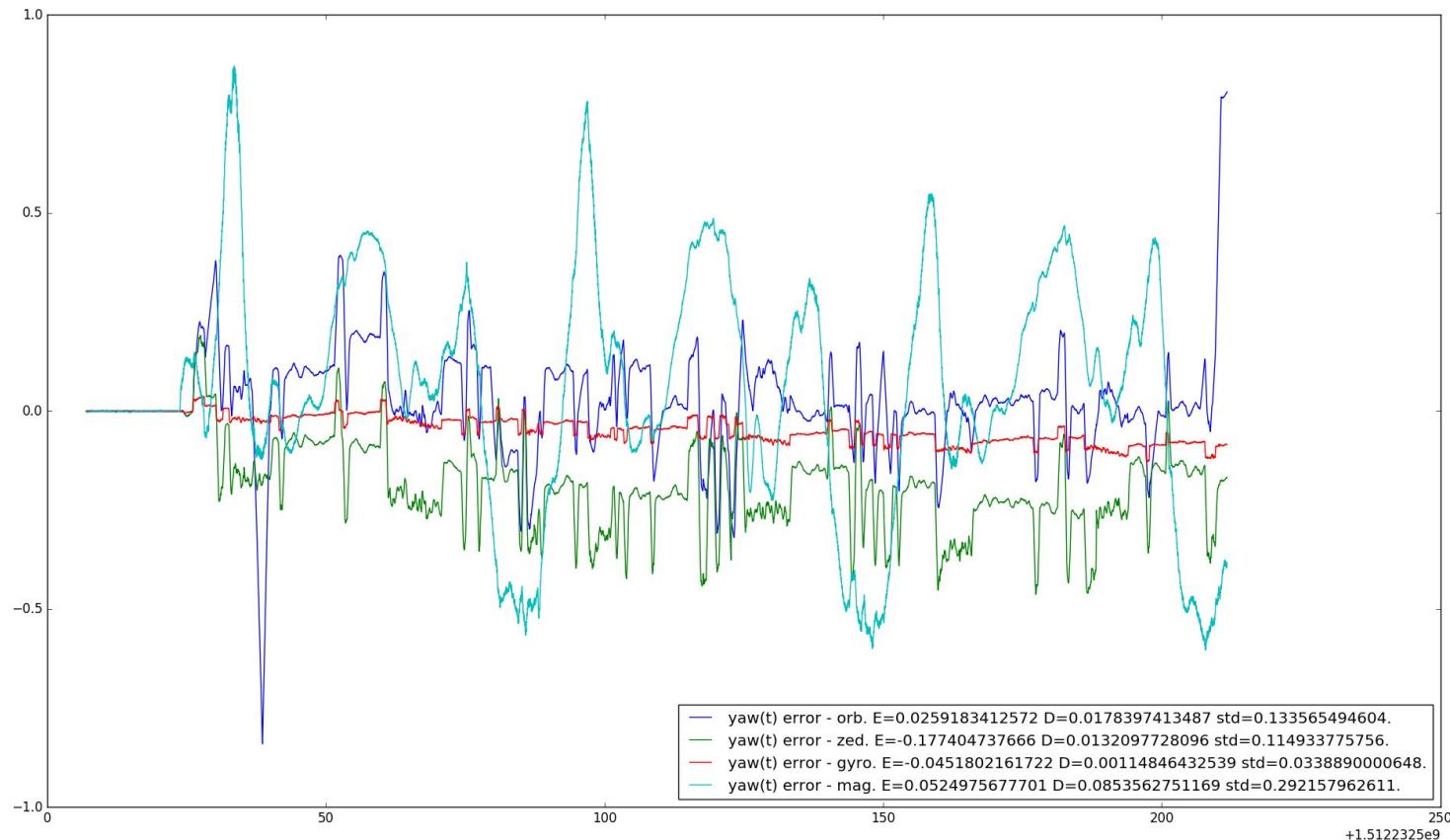
# Comparing - y(t)



# Comparing - yaw(t)



# Comparing - yaw(t)



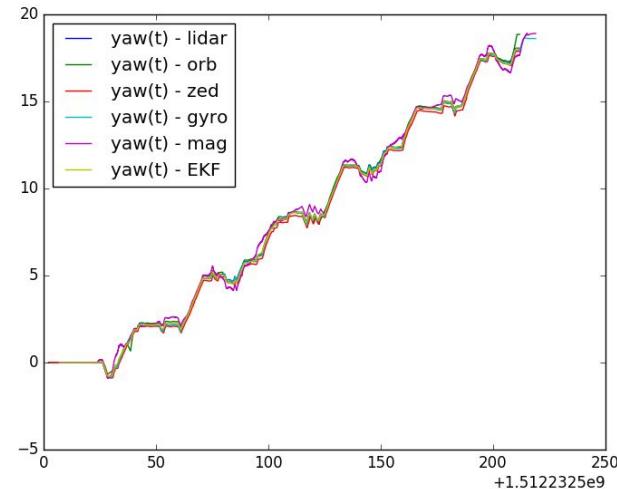
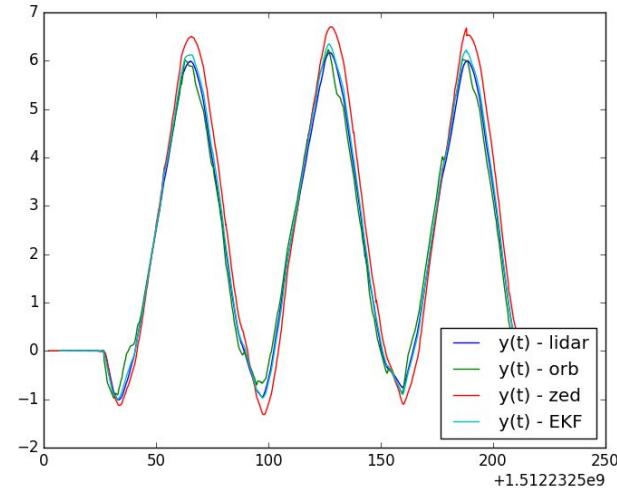
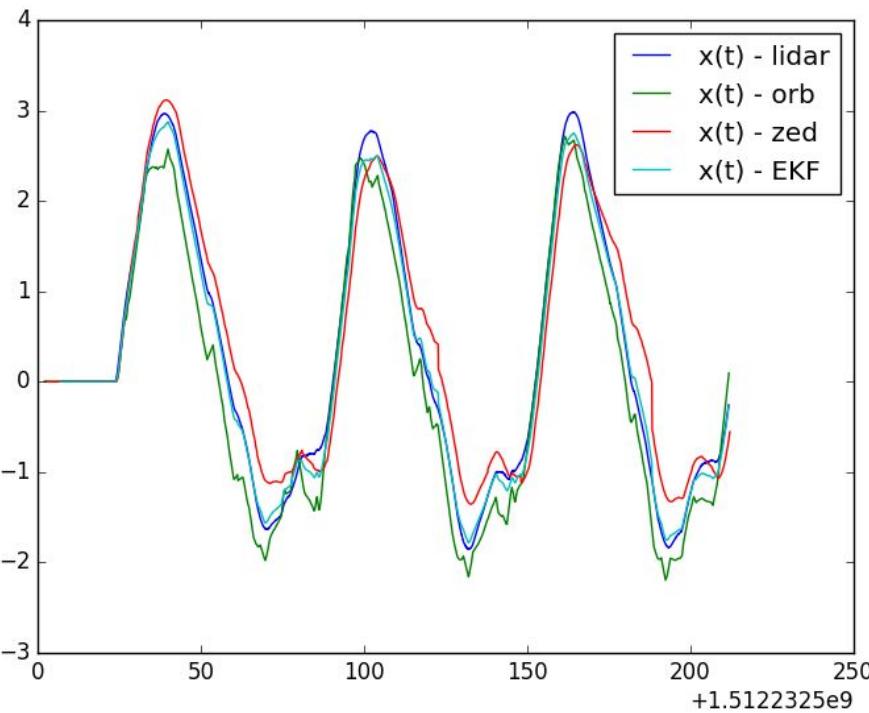
# Sensor fusion - motion model

$$yaw_i = yaw_{i-1} + w_{zi} \cdot dt_i$$

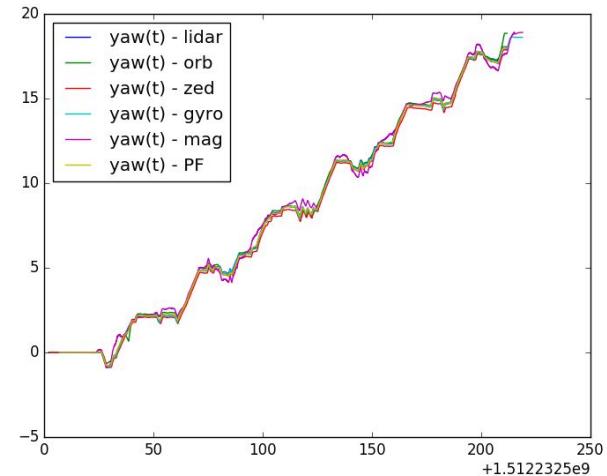
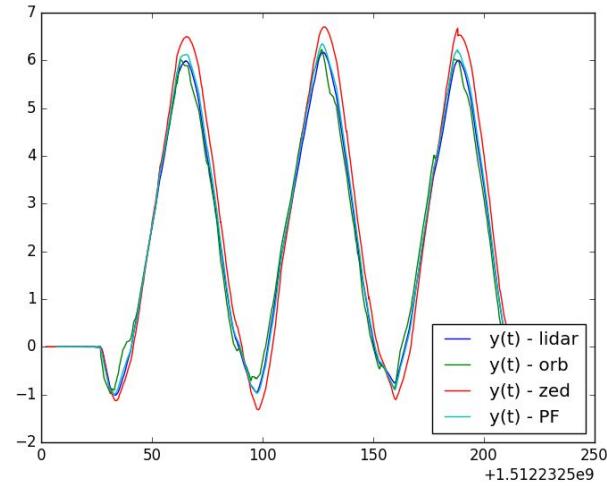
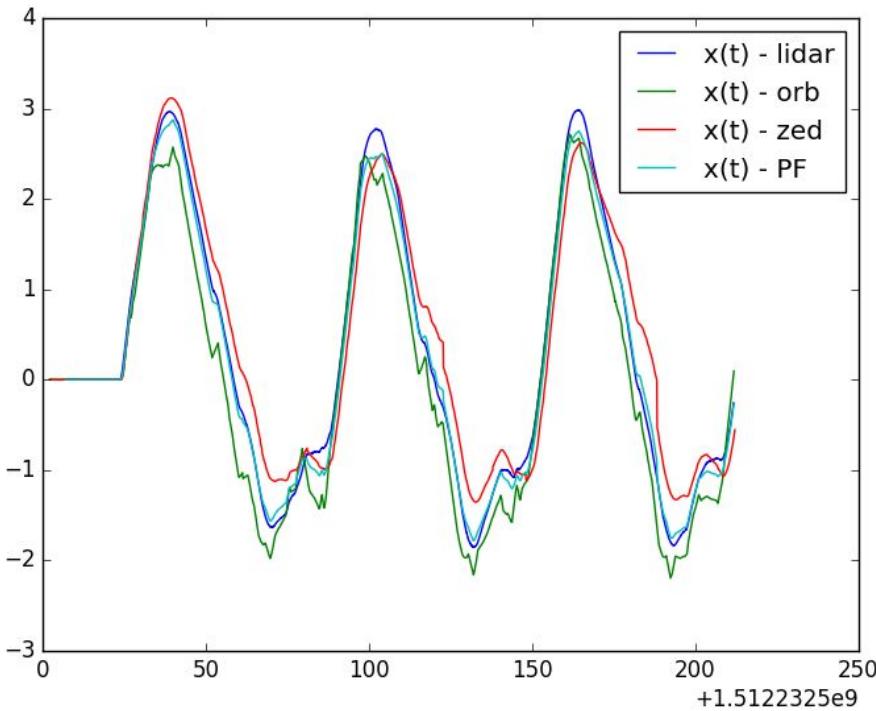
$$x_i = x_{i-1} + v_i \cdot \cos(yaw_i) \cdot dt_i$$

$$y_i = y_{i-1} + v_i \cdot \sin(yaw_i) \cdot dt_i$$

# Sensor fusion - EKF



# Sensor fusion - PF



# Results

- URDF robot model
- Autonomous navigated car-like robot
- Sensor fusion with EKF, PF
  - IMU
  - ZED camera
  - LIDAR

Thank you  
for your attention!