

Team 5: Tokyo Drift

Lab 3: Wall Follower



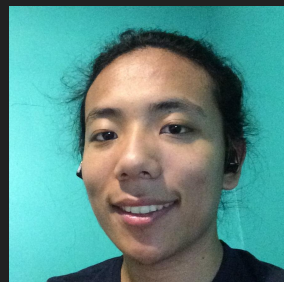
Eric Wieser



Ernie Ho



Miles Steele



Shi-Ke Xue



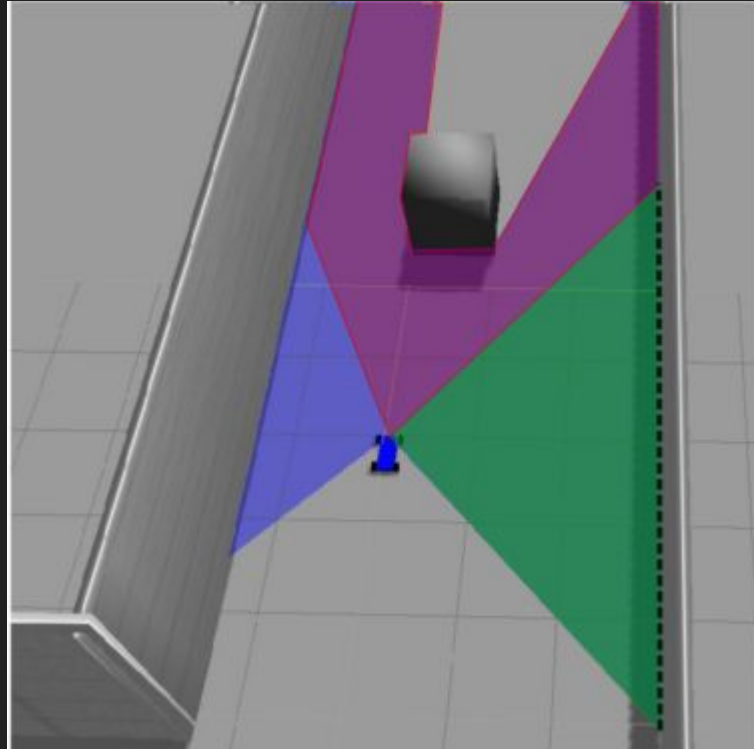
Steven
Homberg



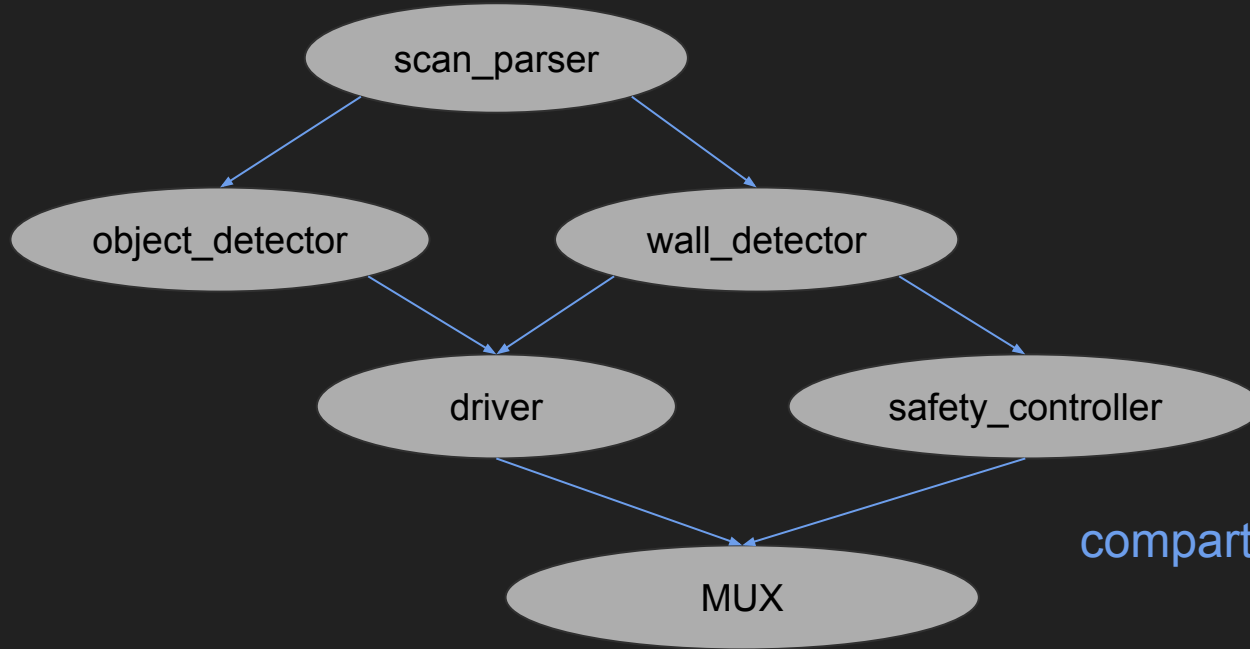
Winter Guerra

Goal: create a wall following pipeline

1. Obstacle Detection
2. Wall Detection
3. Safety Controller
4. Steering Controller



We focused on writing code that could be useful later




compartmentalized tasks


And also on writing and using tools that make code easier to write

rospyext

```
class ObjectDetectorNodeBefore(object):  
    def __init__(self):  
        self.pub_detect = Publisher('/object_detection', Bool)  
        Subscriber('/laser/scan', LaserScan, self.sub_scan)  
        self.distance_param = rospy.get_param('~distance_thresh', 1)  
  
    def sub_scan(self, scan):  
        detected = min(scan) < self.distance_thresh  
        self.pub_detect.publish(detected)
```



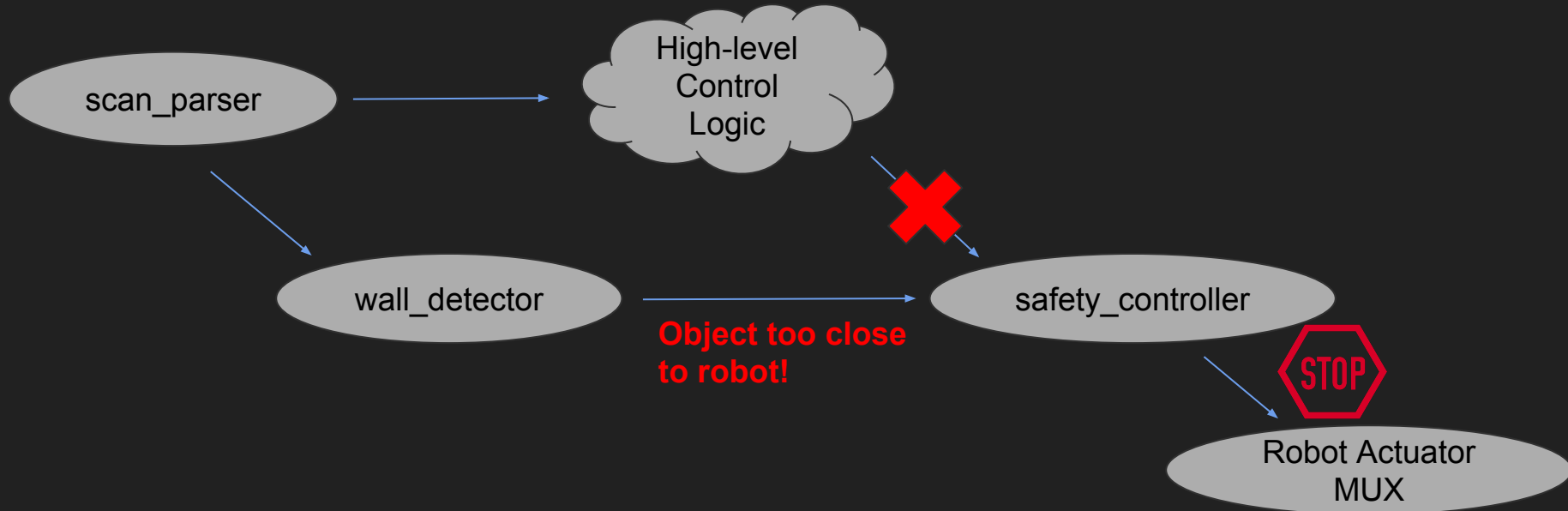
```
class ObjectDetectorNodeAfter(Node):  
    pub_detect = Publisher('/object_detection', Bool)  
    distance_thresh = Param(float, default=1)  
  
    @Subscriber('/laser/scan', LaserScan)  
    def sub_scan(self, scan):  
        detected = min(scan) < self.distance_thresh  
        self.pub_detect.publish(detected)
```



rospy.numpy_msg (builtin)

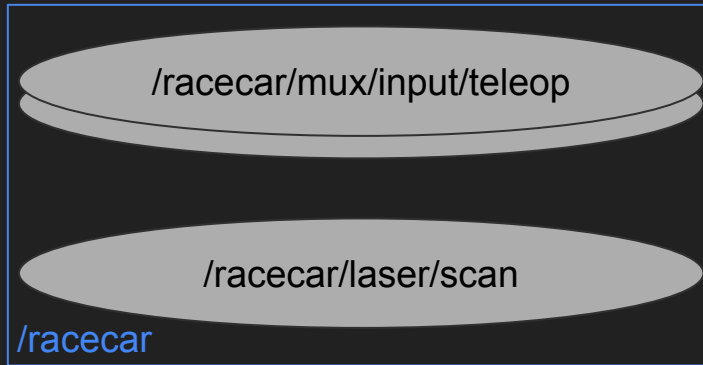
ros_numpy

To make development easier, our safety controller works independently of all higher-level robot logic

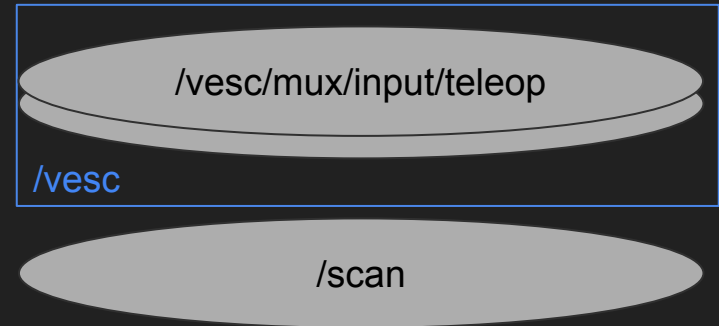


Simulator vs Reality: ROS Topics

Simulator

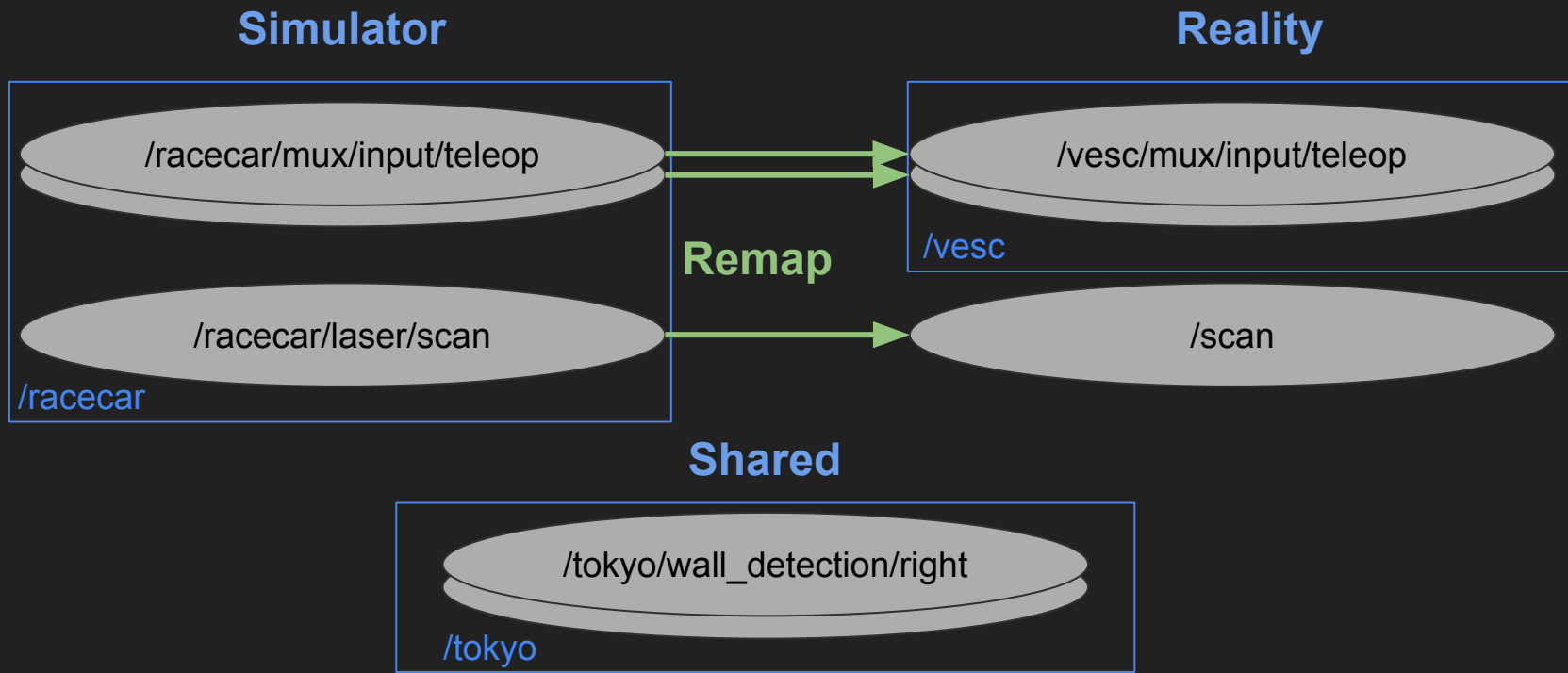


Reality



*mux = ackermann_cmd_mux

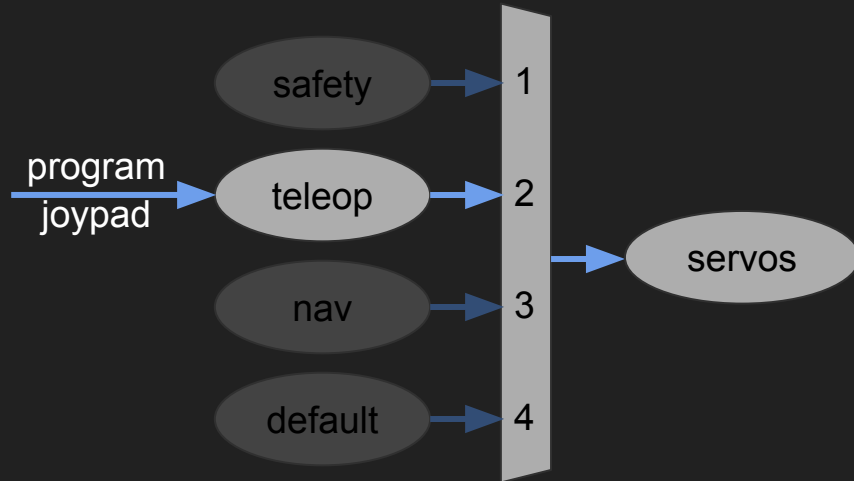
Simulator vs Reality: ROS Topics



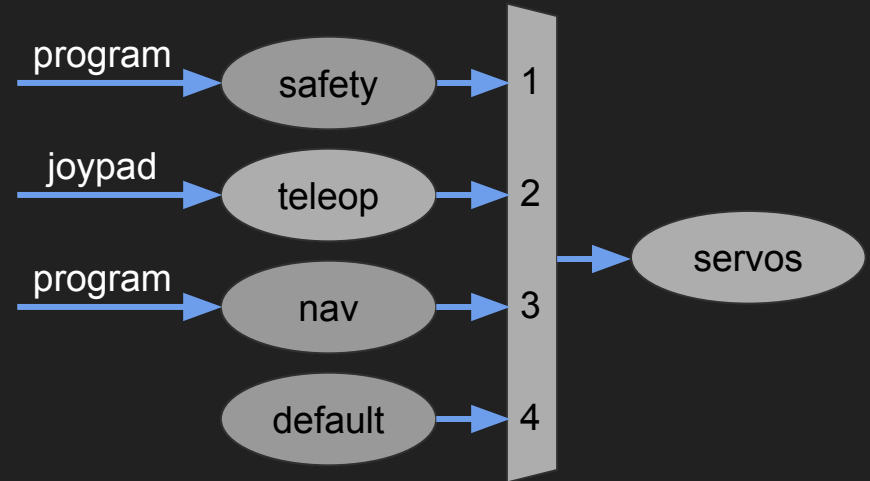
*mux = ackermann_cmd_mux

Simulator vs Reality: Command Mux

Simulator

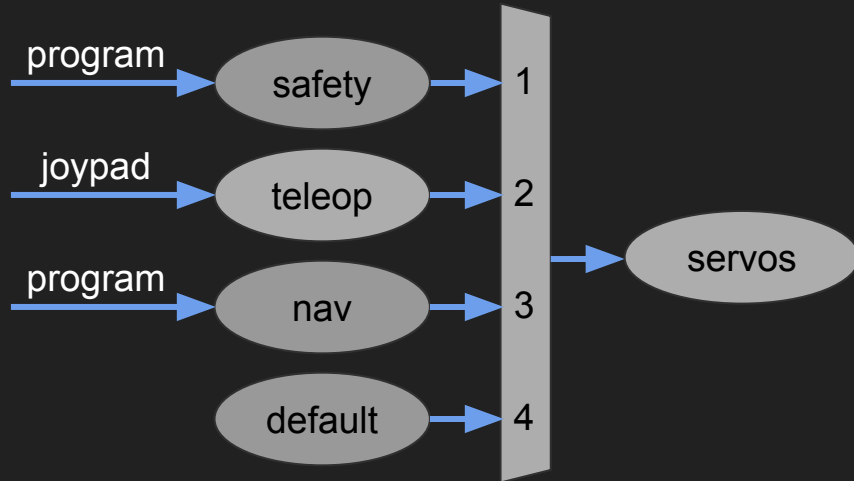


Reality

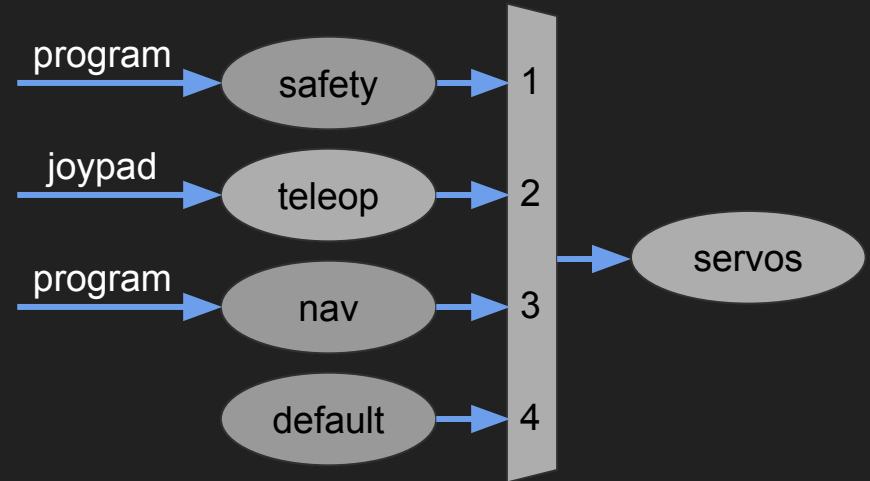


Simulator vs Reality: Command Mux

Simulator



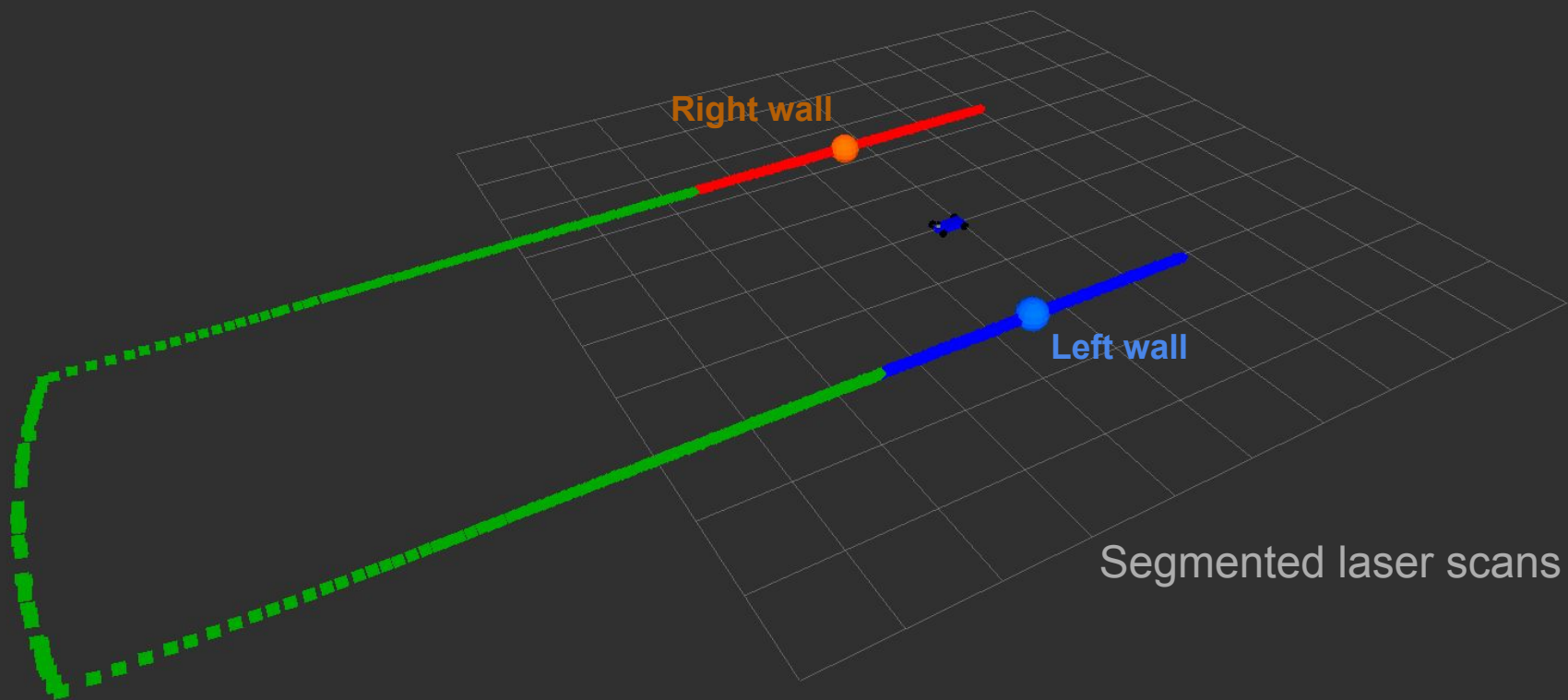
Reality



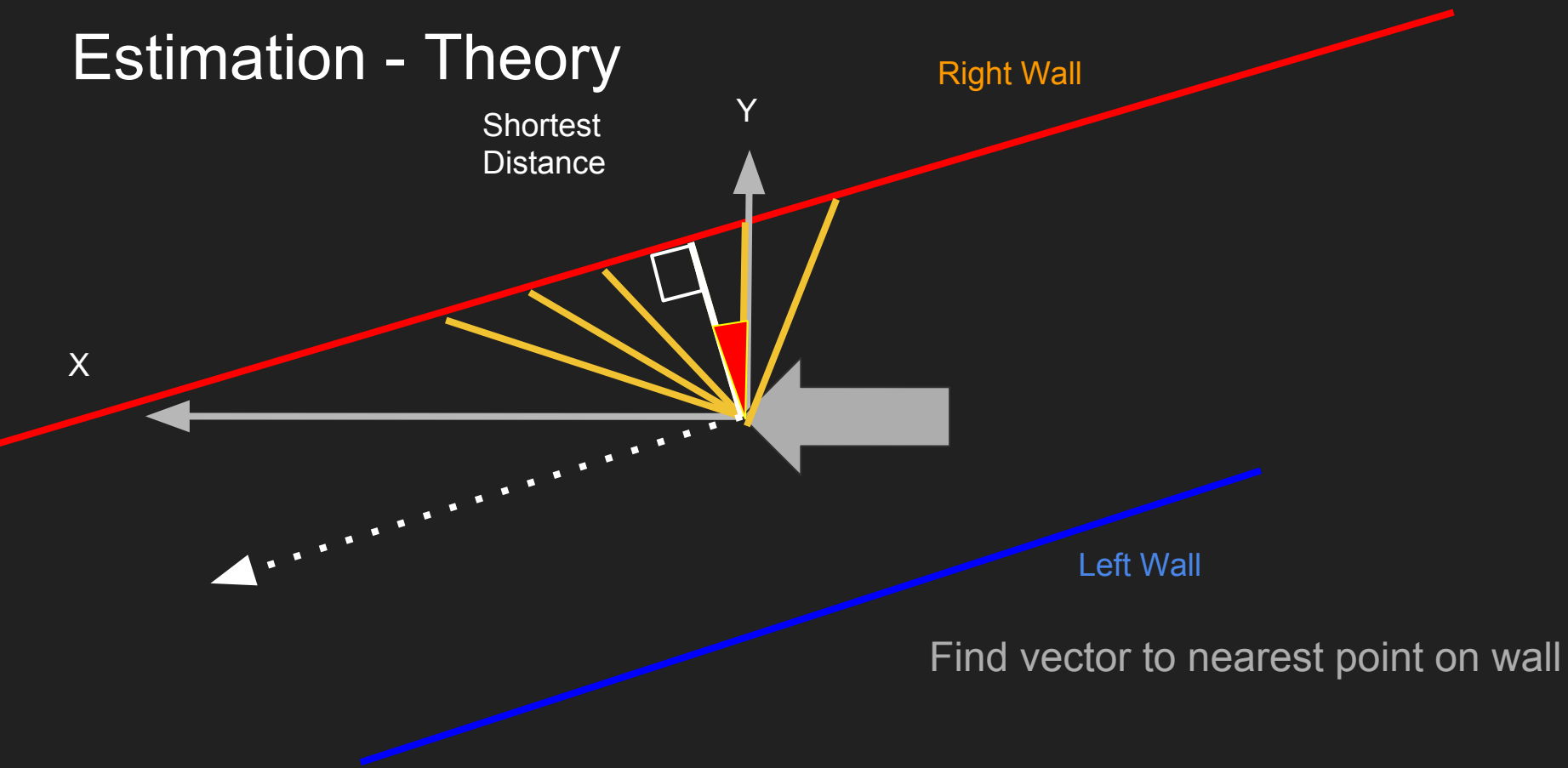
Patched the simulator.

Not a perfect solution.

Estimation Results - Simulator



Estimation - Theory



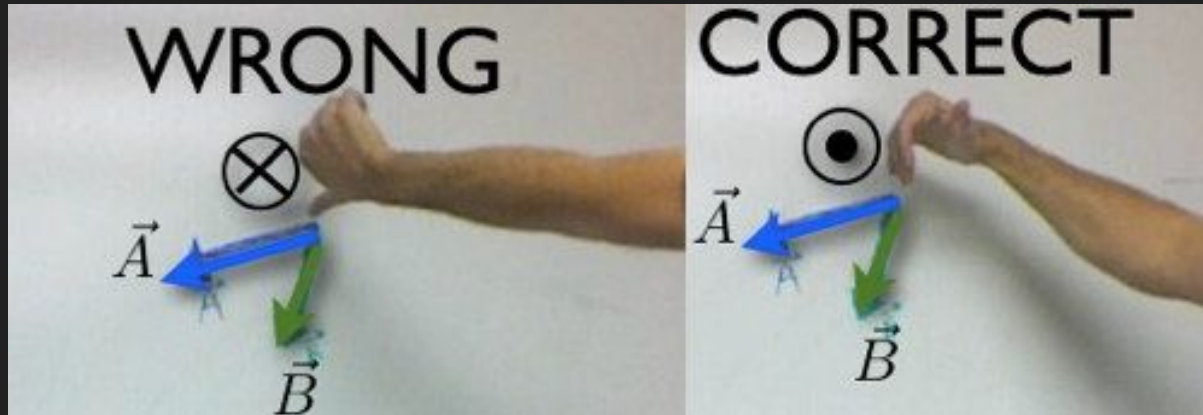
Control Results

- Actually a genius.
- Has input from environment.
- Making steady progress.



Lessons Learned

- Abhishek is still a hero.
- Third time lucky with the router
- Simulator is opposite to real world



Contributions

Eric: Wrote rospyext wrapper and other ros helpers, patched rqt

Ernie: Lab 2, laser to point cloud (failed...), book

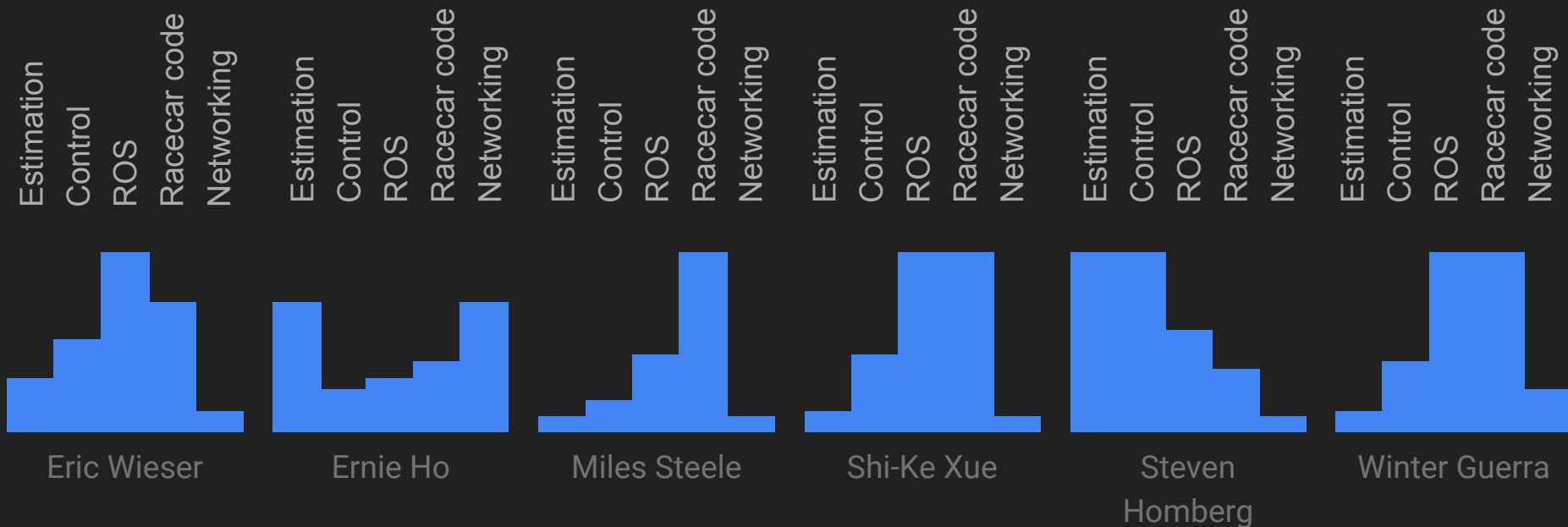
Miles: Wall follower, real-world namespace wrapping

Steven: Scan parser, object detector, wall detector

Winter: Safety controller, patched MUX into simulator

Shi-Ke: Finish Lab 2/Racecar Setup

Individual Learning



Graphs show normalized amount learned