Path Planning

DAY 3

Robotics Teaching Kit Modules

Module 1 Course Introduction	• Course Introduction and Overview • Introduction to Robotics • Introduction to Python • Introduction to ROS
Module 2 Sensors	• Sonar, Accelerometer, Gyroscope • Camera, Motors, Encoders
Module 3 Computer Vision	• Introduction to Computer Vision • Image Filtering • Image Moments
Module 4 Machine Learning	• Introduction to Machine Learning • Neural Networks • Caffe

Robotics Teaching Kit Modules

Module 5 Dead Reckoning	• Introduction to Dead Reckoning • Calculating Positions • Sensor Fusion and Kalman Filters
Module 6 Path Planning	• Introduction to Path Planning • A* Planning
Module 7 Control	• Introduction to Control Systems • PID Control
Module 8 Robot Localization	• Introduction to Robot Localization • Localization with Particle Filters
Module 9 Mapping	• Introduction to Mapping • SLAM
Module 10 Final Project	• Motivation • Harvester • Robot Catch the Ball Game • Color Follower

Planning

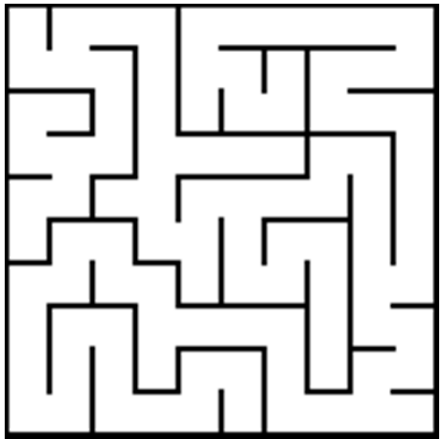
- Robots must make decisions that consider their entire environment
- Robots would be ineffective if they only consider their immediate sensor measurements
- Planning is the procedure of devising a strategy for achieving a goal based on a global perspective of the world.

Navigation

- One of the most important applications of planning is for navigation
- Navigation typically involves two levels of planning:
  - global plans
  - local plans

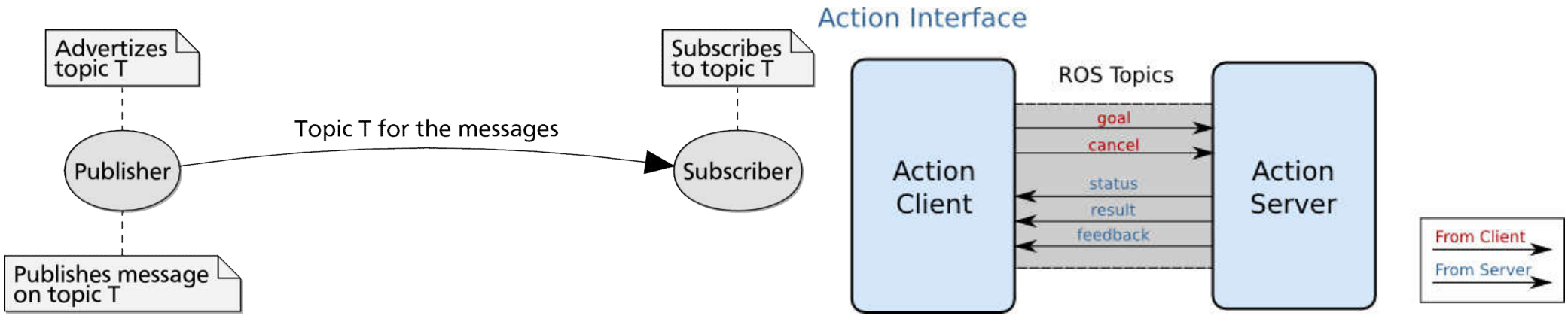
Navigation Maps

- Maps for navigation should show where the navigable regions are
- Navigation maps may also show the conditions of the areas
- If there are important items (like fuel stations or opponents), those may also be present in a map.



Introduction to Action in ROS

Recall Publish/Subscribe



Introduction to move_base

The move_base package provides an implementation of an action that, given a goal in the world, will attempt to reach it with a mobile base.

Configuration on navigation

http://wiki.ros.org/navigation/Tutorials/RobotSetup

Change Speed can be done here but will not covered in this workshop

Get Point of Interest

Use 2D Pose Estimation

- Rostopic echo /amcl_pose

- /amcl_pose show the current position

Quaternion

- The orientation of robot

Coordinate in map

move_base uses *Pose* to send goal

rosmsg info geometry_msgs/Pose

The center of map is (0,0)

Move Forward Avoiding Obstacle

Amcl (Adaptive Monte Carlo Localization)

Takes laser scan, map, transform (tf)

Normally initial position is (0,0,0)

This will automatic plan a route

Go to Specific Point in Map

We may define our destination in x,y coordinate

Steps using Gazebo

1. roslaunch turtlebot3_gazebo turtlebot3_stage_4.launch
2. roslaunch turtlebot3_navigation turtlebot3_navigation.launch map_file:=$HOME/workshop.yaml
3. rosrun <-workspace/src/package-> goforward.py

Steps in Turtlebot3

1. Connect Turtlebot
2. roslaunch turtlebot3_navigation turtlebot3_navigation.launch map_file:=$HOME/workshop.yaml
3. rosrun <-workspace/src/package-> goforward.py

Robotics Teaching Kit Modules

Module 1 Course Introduction	• Course Introduction and Overview • Introduction to Robotics • Introduction to Python • Introduction to ROS
Module 2 Sensors	• Sonar, Accelerometer, Gyroscope • Camera, Motors, Encoders
Module 3 Computer Vision	• Introduction to Computer Vision • Image Filtering • Image Moments
Module 4 Machine Learning	• Introduction to Machine Learning • Neural Networks • Caffe

Robotics Teaching Kit Modules

Module 5 Dead Reckoning	• Introduction to Dead Reckoning • Calculating Positions • Sensor Fusion and Kalman Filters
Module 6 Path Planning	• Introduction to Path Planning • A* Planning
Module 7 Control	• Introduction to Control Systems • PID Control
Module 8 Robot Localization	• Introduction to Robot Localization • Localization with Particle Filters
Module 9 Mapping	• Introduction to Mapping • SLAM
Module 10 Final Project	• Motivation • Harvester • Robot Catch the Ball Game • Color Follower

Thank you very much