CSCE574 – Robotics
Spring 2014 – Project 3

Assigned: March 3
Due: March 25

The purpose of this assignment is to give you chance to set up your Turtlebot robot and familiarize yourself with its operation.

Getting Ready

In addition to *AGITR*—you should now ideally be comfortable with all of the material in that book—the course website has some additional notes with details on our Turtlebot robots.

The Task

For this project you should . . .

- . . . get your robot working.

  After installing Ubuntu and ROS and configuring everything, create a launch file that includes all of the nodes mentioned in the accompanying notes.

- . . . write a program to monitor and publish the netbook's current WiFi signal strength.

  Publish this information on a new topic called `/wifi_ss` using `std_msgs/Int32` as the message type. You can get the data you need by opening and reading from a file called

  ```
  /proc/net/wireless
  ```

  The field you are looking for is called "Quality-link".

- . . . write another program that monitors the sensor information from the robot.

  Immediately generate an informative message whenever the robot generates a diagnostic message at the WARN or ERROR levels (show the details of the message itself), or when one of the robot's bump sensors or wheel drop sensors is triggered (mention which bump sensor or which wheel drop sensor). In addition, generate messages every five seconds showing percentage charge of the Create battery and the WiFi signal strength.
  For example, here's a snippet of output from a reasonable solution:

  ```
  [ INFO] [1331358828.627342623]: Wheel drop caster.
  [ INFO] [1331358828.631176162]: Wheel drop left.
  [ INFO] [1331358828.631280994]: Wheel drop caster.
  [ INFO] [1331358828.659960206]: Create battery status: 94%
  [ INFO] [1331358828.659960210]: Wifi signal strength: 64%
  [ INFO] [1331358828.660676083]: Wheel drop left.
  [ INFO] [1331358828.660786084]: Wheel drop caster.
  [ INFO] [1331358828.705600458]: Wheel drop left.
  [ INFO] [1331358828.705716745]: Wheel drop caster.
  [ INFO] [1331358828.710576962]: WARNING from Cliff Sensor: Near Cliff
  [ INFO] [1331358828.735038154]: Wheel drop left.
  [ INFO] [1331358828.735138517]: Wheel drop caster.
  ```

  Be prepared to execute this node on either the netbook or on your workstation.

- . . . be ready to teleoperate the robot and visualize its sensor data in rviz (including a robot model and faked laser scans) and the robot monitor, while these programs are running.

CSCE574 – Project 3 Cover Sheet

Name(s): _____ _____ _____ _____

Robot setup (30):

- □ Operating system installed.
- □ ROS installed and working.
- □ Wireless access point correctly configured to allow access to robot.
- □ Wireless network properly secured.
- □ ROS settings correct for multi-computer environment with core on robot.
- □ Can access netbook remotely via wireless, without using its keyboard.

Basic robot functions (25):

- □ create driver
- □ velocity command multiplexer
- □ robot description
- □ robot state publisher
- □ diagnostic aggregator
- □ extended Kalman filter
- □ Kinect breaker enabler
- □ Kinect driver
- □ depth image to laser scan
- □ (workstation) runtime monitor
- □ (workstation) teleoperation node
- □ (workstation) rviz: robot model
- □ (workstation) rviz: laser scans

WiFi signal strength node (10):

- □ Correctly reads WiFi signal strength.
- □ Publishes signal strength.

Sensor monitoring node (25):

- □ Receives to diagnostic messages.
- □ Correctly identifies warning and error diagnostic messages.
- □ Receives to sensor state messages.
- □ Correctly extracts wheel drops and bumps from sensor state messages.
- □ Receives to battery state messages.
- □ Receives to WiFi strength messages.
- □ Timing: Errors, warnings, bumps and wheel drops immediately.
- □ Timing: Battery and WiFi messages every 5 seconds.

Report (10):

- □ Report is complete and clear.
- □ Required sections exist under readily identifiable headings.
- □ Report is free of typos and grammatical errors.

Other comments:

Total: