Robot Sensors

Exercise 1: measure communication delay

ROS is designed with distributed computing in mind. A well-written node makes no assumptions about where in the network it runs, allowing computation to be relocated at run-time.

NOTE: When connecting multiple computers, you need only one ROS master. The ROS master will run on the TurtleBot. All other nodes must be configured to use the same ROS master, via `ROS_MASTER_URI`.

1. Connect two laptops to the TurtleBot.
2. On laptop 1,
    a. get the IMU measurements of the TurtleBot from the `/imu` topic;
    b. forward the IMU message on a new topic (e.g., `/imu_forwarded`) to laptop 2.
3. On laptop 2,
    a. get the IMU measurements from the `/imu_forwarded` topic;
    b. calculate the communication delay using `/imu_forwarded.header.stamp` field and `rostime("now")` (https://mathworks.com/help/ros/ref/time.html).

Exercise 2: video stream

TurtleBot3 is equipped with an 8-megapixel Raspberry Pi Camera. The camera can provide a high-definition video stream.

1. To start streaming the video on a ROS topic, besides running

    ```
    roslaunch turtlebot3_bringup turtlebot3_robot.launch
    ```

    in another terminal run

    ```
    roslaunch turtlebot3_bringup turtlebot3_rpicamera.launch
    ```

2. Receive the image in MATLAB: https://github.com/au-mobile-robots/Tutorials/blob/master/read image from camera/demo_grabImageFromRobot.m
3. Modify the script to show the live stream from the robot.