RBE 3002 Lab 2 - Mobile Robot Kinematics and Odometry

Robert Dabrowski
Robotic Engineering and
Computer Science
Worcester Polytechnic Institute
Worcester, Massachusetts 06109–2280
Email: rpdabrowski@wpi.edu

***Abstract*—In this lab, the basics of Robot Operating System (ROS) were explored. A single node was developed with the sole purpose of publishing some command velocities that accomplish some arbitrary trajectory. Basic mobile robot kinematics are also explored.**

I. Introduction

In order to fully explore robot navigation principles, Robot Operating System (ROS) is used to give a jump start in that process. Sensor calibration, reading and abstraction is all handled already and packaged up neatly for use by students. Using ROS, implementing two-wheeled robot kinematics is more about math than embedded programming. To bring this to the real world outside of code, we are also provided with Kobuki TurtleBots, which provide many resources for those programming them. This lab details the making of a node that sends command velocity messages for the robot to execute and follow a trajectory.

II. Methodology

The most technical part of this lab was performing the math, but after that implementation was fairly trivial. All of the figures below were taken from Lecture 4: Mobile Robot Kinematics, taught by Professor Chernova. [1]

Figure 1 below shows the initial diagram drawn to represent the TurtleBot. The "b" is the wheel base, measured to be 23 cm. "R" is the radius of rotation. "u" is the forward velocity and "w" is the angular velocity.

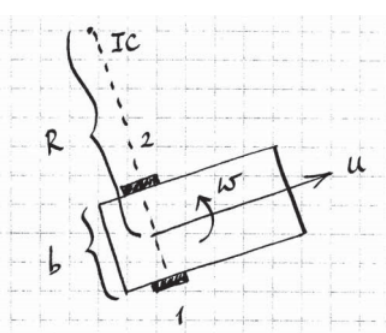

Figure 1: *Representation of two wheeled mobile robot.*

Figure 2 below shows how to calculate the forward velocities of each wheel based on each wheel radius and rotational velocity.

$$u_1 = r_1 \dot{\phi}_1 = r \dot{\phi}_1$$
$$u_2 = r_2 \dot{\phi}_2 = r \dot{\phi}_2$$

Figure 2: *Forward velocities of individual wheels.*

Figure 3 below shows how to calculate the forward velocity of the robot with respect to the rotational velocities of the wheels and wheel radius.

$$u = \frac{r}{2}\left(\dot{\phi}_1 + \dot{\phi}_2\right) \quad : \text{translational robot velocity}$$

Figure 3: *Forward velocity of the robot.*

Figure 4 below shows how to calculate the angular velocity of the robot.

$$u_1 = \omega\left(R + \frac{b}{2}\right) \quad \text{and} \quad u_2 = \omega\left(R - \frac{b}{2}\right)$$
$$\dot{\phi}_1 = \frac{\omega}{r}\left(R + \frac{b}{2}\right) \quad \text{and} \quad \dot{\phi}_2 = \frac{\omega}{r}\left(R - \frac{b}{2}\right)$$
$$\dot{\phi}_1 - \dot{\phi}_2 = \frac{\omega b}{r}$$
$$\omega = \frac{r}{b}\left(\dot{\phi}_1 - \dot{\phi}_2\right) \quad : \text{angular robot velocity}$$

Figure 4: *Calculating the angular velocity of the robot.*

Figure 5 below shows how to calculate the radius of rotation based on the previously mentioned equations.

To calculate R:

$$\dot{\phi}_2 = \frac{\omega}{r}\left(R - \frac{b}{2}\right) = \frac{r}{b}\left(\dot{\phi}_1 - \dot{\phi}_2\right) \cdot \frac{1}{r}\left(R - \frac{b}{2}\right)$$
$$R = \frac{b}{2} + b\,\frac{\dot{\phi}_2}{\dot{\phi}_1 - \dot{\phi}_2} = \frac{b}{2}\,\frac{\dot{\phi}_1 + \dot{\phi}_2}{\dot{\phi}_1 - \dot{\phi}_2} \quad : \text{turning radius.}$$

Figure 5: *Calculating the radius of rotation.*

See the attached code for my implementation of these equations and all the functions from the skeleton code.

III. Results

After hours of programming, my node can successfully have the TurtleBot execute a predefined trajectory. It is not terribly consistent because of the bang-bang control. I also did not try to implement the ramp function due to time constraints. As for the quality of the odometry readings of the robot, they seemed accurate but I did not actually measure them.

The only problems I encountered involved taking the starting skeleton code and figure out what direction to go in terms of researching the use of the various topics containing command velocity and odometry. For example, the skeleton code instructs us to use the ”tf” package, but I completed the assignment without it.

IV. Conclusion

In the future, I will need to spend more time on the lab so that I can implement the extra credit, which seems like the most important thing from this lab when looking back at it.

References

[1] Professor Sonia Chernova, *Lecture 4: Mobile Robot Kinematics*. Lecture. 2014.

RBE3002 Lab2: Mobile Robot Kinematics and Odometry

Name: Robert Dubrowski

Lab Sign-offs

TASK	Simulated*	Real	SIGNED BY	DATE/TIME
Prelab		✓	JSO	3/28/14
Spin Wheels	☑	☑		
Drive Straight	☑	☑		
Drive Arc	☑	☑		
Rotate	☑	☑		
Bumper Press Trajectory	☑	☑	JSO	3/3/14
Extra Credit	☐	☑		

*Demonstrating each capability on the simulator <u>or</u> on the real robot is sufficient. However, only 60% credit will be given for tasks shown only in simulation.

Grading Rubric

[100 points] All pre-lab and in-lab procedures are completed and demonstrated prior to the deadline. All deliverables and sign-off sheet are submitted on time. The code is well commented and structured.

Figure 6: *Signoffs. Done.*