Jaguar Lite Point Tracking

Lauren Lieu
Department of Engineering
Harvey Mudd College
Email: llieu@g.hmc.edu

Joshua Vasquez
Department of Engineering
Harvey Mudd College
Email: jvasquez@g.hmc.edu

March 5, 2013

***Abstract*—This paper describes the implementation of point tracking for the *Jaguar Lite Mobile Robot Platform*. Motion control of the autonomous vehicle is achieved using a Proportional-Integral-Derivative controller. In both simulation and hardware mode, the control law parameters and control gains have been tuned to demonstrate stable, precise motion towards desired goal poses.**

I. Introduction

Point Tracking refers to a robot's ability to transition from one pose to another on a known map of its environment. By modeling the robot's drive train with a motion model and by incorporating odometric sensor readings into this motion model, the robot can develop a trajectory between points. In the following paper, we incorporate both a differential drive-train motion model with a PID-controlled differential drive train to designate the Jaguar Lite robotic platform to track points on a two-dimensional environment.

A. Hardware Platform

The hardware platform of choice is a Jaguar Lite autonomous vehicle, sourced by Dr. Robot. This differential-drive platform is fairly rugged, and features a suite of sensory inputs: a 9 DOF Inertial measurement unit (IMU), two rotary encoders, a 240° field-of-view laser range finder, and an on-board webcam [2]. With a wireless wifi interface, the designer can implement navigation algorithms in *C#* within Microsoft Visual Studio to communicate with the Jaguar platform. Most importantly, the platform can be driven both indoors and outdoors.

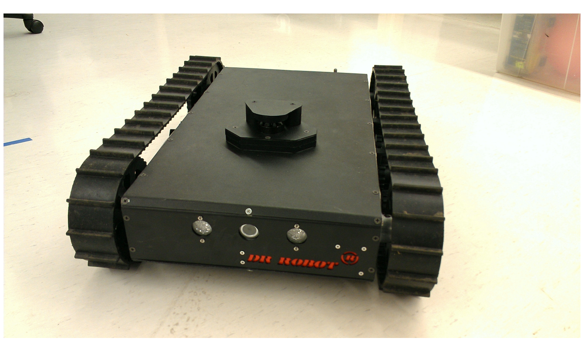

Fig. 1. The Jaguar Lite Platform

B. Terminology

This paper refers to specific definitions and usage of the following terms:

- **Pose** represents the robot's position: $[x, y, \theta]$ relative to a coordinate frame fixed to the environment that the robot navigates.
- **Nonholonomic Constraints** refers to a restricted motion path constrained by the physcial construction of the robot. On a two-dimensional coordinate frame, a robot that can freely rotate and translate is considered a holonomic robot.
- **Point Tracking** refers to the Jaguar Lite's ability to drive from one pose to another on a navigable coordinate frame.

II. Background

III. Problem Definition

IV. Control Design

A. Motion Model

Developing a method of navigating from one pose to another simplifies to determining the appropriate motor speeds of the Jaguar Lite at any given pose to form a trajectory as the robot pursues a goal pose. To determine the robot's motor outputs, an effective motion model which reflects the robot's nonholonomic constraints must be used.

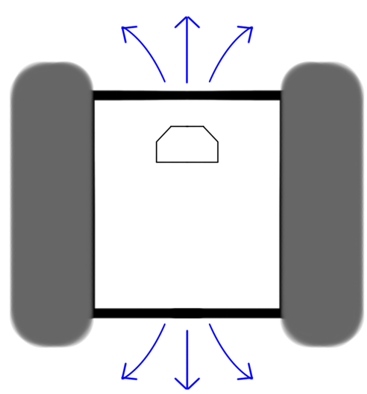

Fig. 2. Realizable Trajectories

The Jaguar Lite moves with a differential drive train. For this reason, the motion model cannot incorporate direct translation of the robot in a direction normal to the sides of the

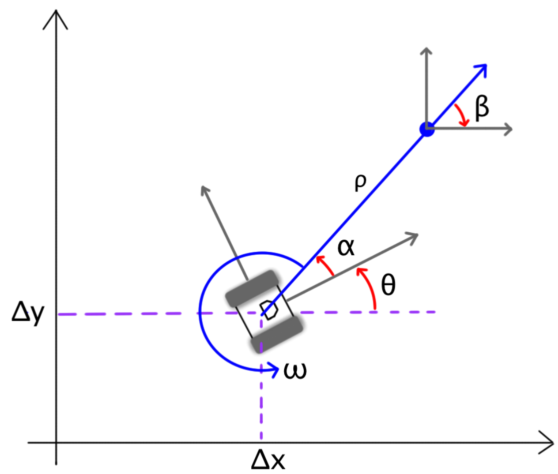

Fig. 3. Linear and Angular Errors

robot. Figure 2 illustrates possible trajectories, each of which is governed by the speed of the right wheel ω_1 and the left wheel ω_2.

Given a differential drive train, we can use the following coordinate transform to describe the robot's pose relative to the goal pose [4]:

$$\rho = \sqrt{\Delta x^2 + \Delta y^2} \tag{1}$$

$$\alpha = -\theta + atan2(\Delta y, \Delta x) \tag{2}$$

$$\beta = -\theta - \alpha \tag{3}$$

Equations (1), (2), and (3) x and y define the horizontal and vertical displacements respectively, relative to a fixed global coordinate frame. θ defines the angular displacement of the robot normal vector relative to the x axis. ρ defines the magnitude of the displacement from the robot's pose to the goal pose. α defines the error of the robot normal vector relative to the desired angle relative to the goal. β defines the angle of the goal vector relative to the horizontal. The angles of the coordinate transformation are shown in Figure 3.

The above equations quantify the error, or difference between the robot's pose and the desired pose. By calculating the kinematics in this coordinate frame and applying the control law, equations (4) and (5) can be produced, which define the Jaguar Lite's linear velocity v and angular velocity ω as a function of the error values ρ, α, and β.

$$v = k_\rho \rho \tag{4}$$

$$\omega = k_\alpha \alpha + k_\beta \beta \tag{5}$$

where k_ρ, k_α, and k_β are tuned gain constants.

Given that equations (4) and (5) model the robot's desired change in displacement over time, this motion control must be achieved on the Jaguar Lite platform through control of the left and right wheel rotations. Equations (6) and (7) fulfill this requirement.

$$\omega(t) = \omega_1 + \omega_2 \tag{6}$$

$$v(t) = L(\omega_1 - \omega_2) \tag{7}$$

where L is the radius of the robot. Thus, the overall left and right angular velocities, ω_1 and ω_2 respectively, may be determined by combining the previous equations to produce:

$$\omega_1 = \frac{v}{2L} + \frac{\omega}{2} \tag{8}$$

$$\omega_2 = -\frac{v}{2L} + \frac{\omega}{2} \tag{9}$$

B. PID Motor Control

The above equations rely on accurate determination and control of the left and right wheel speeds. To determine the actual wheel speeds the Jaguar Lite receives sensory input from quadrature encoders. Overall, the point tracking trajectory becomes far more accurate in implementation if the left and right motor speeds are controlled with a tight control loop. Thus, the desired left and right motor speeds (derived from ω_1 and ω_2) can be far more closely approximated by the actual Jaguar hardware. To achieve wheel-speed control, a PID controller was implemented to stabilize the individual wheel speeds of both motors.

At the lower level the Jaguar Lite's left and right wheel motors can change speed through pulse-width modulation, and a specified duty cycle. By tuning the P, I, and D control gains, the controller changes the duty cycle, effectively maintaining wheel velocities far closer to the desired values.

V. Results

The three control gains K_ρ K_α, and K_β weight the influence of each error parameter, iterating through time to measure the displacement of the robot relative to the goal pose. By tuning these parameters both in simulation and hardware modes, the desired correction velocities were achieved. In simulation, the tuned parameter values are $K_\rho = 1$ $K_\alpha = 3$, and $K_\beta = -2$ while in hardware the optimal values are $K_\rho = 4$ $K_\alpha = 4.5$, and $K_\beta = -4.5$.

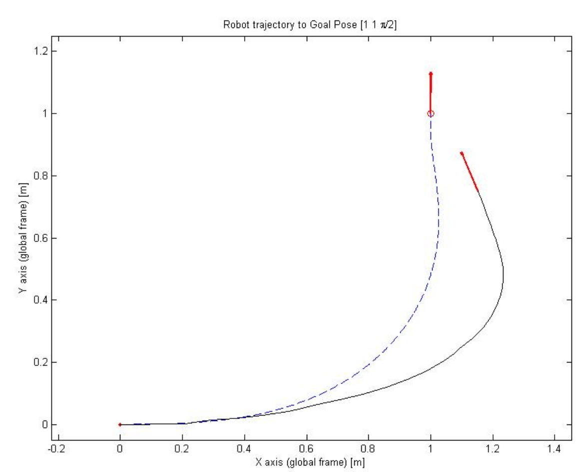

Fig. 4. Simulated robot trajectory (blue, dashed) and actual Jaguar Lite trajectory (black) with final poses (red).

For experimental testing, tuning the three P, I, D controller constants of the motor velocity controller also guided the Jaguar Lite to its goal pose. With the tuned control gains $P = 14$, $I = 5$ and $D = 12$ the robot successfully traverses from an initial pose at the origin to a goal pose within range of the actual desired final point. The trajectory of the robot in simulation and hardware is shown in Figure 4.

The presented motion model and control law does not encompass all cases of possible robot locomotion. In particular, this control design does not account for spinning the robot in place at a given pose when x and y remain fixed. This arises when the robot is already at the location of its goal pose, such that ρ is zero, but its orientation θ is different angle than the desired angle. Given the Jaguar Lite's ability to spin in place, this possibile method of movement is not encompassed by equations (1), (2), and (3). However, by removing the α term in equation (2), corrected desired velocities can be written as follows:

$$v = 0 \tag{10}$$

$$\omega = K_\beta \beta \tag{11}$$

Thus, redefining v and ω when ρ is sufficiently small allows the robot to rotate in place. The hardware implementation of this type of locomotion is limited depending on the terrain, however, simulation testing of this control design was successful.

VI. Conclusion

By implementing this control design and combining this with odometry-based localization, the Jaguar Lite can effectively traverse from one pose to a set goal pose. Furthermore, by simplifing trajectories into a series of desired poses, point tracking presents a powerful solution to generating trajectories for the robot to follow. Thus, the initial development of a motion model can effectively lead to the implementation of obstacle avoidance, tracking of waypoints and continuous path following on a known map.

Future work includes extending and applying point tracking capabilities to track straight and curved trajectories, and also incorporating odometry error characterization to correct for the drift identified in the Jaguar Lite encoder hardware.

Acknowledgment

The authors would like to thank Professor Christopher Clark for establishing a software framework for algorithm development and providing hardware resources from the Lab for Autonomous and Intelligent Robotics.

References

[1] http://newwww.hmc.edu/lair/E190Q/E190Q-Lab01-IntroToTheJaguar.pdf

[2] http://jaguar.drrobot.com/specification_lite.asp

[3] http://newwww.hmc.edu/lair/E190Q/E190Q-Lecture04-PointTracking.pdf

[4] Siegwart Roland, et al. *Introduction to Autonomous Mobile Robots* 2nd ed. 2011 MIT Press

[5] Thrun, Sebastian, et. all. *Probabilistic Robotics* 2005, The MIT Press.