Modeling and Calibrating a 4-wheel Skid-Steer Research Robot

by Gary Haas

Approved for public release; distribution unlimited.
NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer’s or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.
Modeling and Calibrating a 4-wheel Skid-Steer Research Robot

Gary Haas
Vehicle Technology Directorate, ARL
### 14. ABSTRACT

This report describes the calibration of a commercial Roboteq motor controller which was installed in an iRobot ATRV Jr research robot as part of a system upgrade. The objective of the calibration was to determine the transfer function between a command issued to the robot control system and the path followed by the robot and all relevant parameters. Implicit in the calibration is validation of the model on which the transfer function is based, which is a kinematic model of a 4-wheel skid-steer vehicle.

### 15. SUBJECT TERMS

Robot, skid-steer, calibrate
## Contents

List of Figures ................................................................. iv  
List of Tables ............................................................... iv  

1. **Background** ........................................................... 1  

2. **Determining Motor Controller Constants** .................. 1  
   2.1 Odometer ...................................................................... 2  
   2.2 Speed Setting ............................................................... 3  
   2.3 Turning Radius ............................................................. 4  

3. **Modeling and Calibrating** .......................................... 5  

4. **Results** ....................................................................... 7  

5. **Conclusions and Further Work** ................................... 8  

References ........................................................................ 9  

Distribution List ............................................................ 10
List of Figures

Figure 1. In 2-dimensional plan view, wheel centers (“tire-points”) describe a circular path about the center C, at left. At right, a single wheel is depicted in plan view, with vector components of its velocity. .................................................................6

List of Tables

Table 1. Measurements used in calculating conversion factor from wheel rotational velocity in revolutions per minute to Roboteq arbitrary speed units.................................................................3
Table 2. Examples of Roboteq controller response to combinations of channel 1 and channel 2 (linear and rotational velocity) setpoints. In the fourth example, the speed setting for the right wheel exceeds the maximum, so both settings are adjusted to maintain rotational velocity at the maximum possible translational velocity.................................................4
1. Background

The programmatic background and design of the research robot of this report is documented in (1). All work was performed at the Unmanned Vehicles Technology (UVT) Division of the U.S. Army Research Laboratory. The effort described herein followed as part of the tuning process, selecting parameters for the motor controller so that the robot performance can be specified in engineering units and executed faithfully by the robot. The objective is a transfer function between an instantaneous element of the desired path, specified as a pair of linear velocity and rotational velocity, and the corresponding command to the motor controller. The major components of this process are the specification of the kinematic model according to which the robot responds to inputs, and the selection of parameters peculiar to the motor controller board. The kinematics of a 4-Skid-Steer vehicle have been addressed in any number of academic papers and theses (2–4). The various descriptions of the model in the literature omit a couple of elements essential to the UVT implementation, so these elements are derived herein. These elements and the general approach to validating the model may be of general interest to others attempting similar projects. The section addressing calibration of the motor controller is specific to that controller and serves to document the specific steps and specific values determined for the UVT robot. In this section, only the general approach is likely to be of use to others, except for those integrating the same controller.

2. Determining Motor Controller Constants

The Roboteq controller (5) is configured as a closed-loop servo system using pulse-width modulation to control two direct current (DC) motors equipped with digital encoders. Each motor drives two wheels mechanically coupled together, one motor for the left wheels and one for the right wheels. The controller receives commands from the (external) control computer over a serial data port. During calibration the external computer was a laptop computer running the Roborun utility provided with the controller. In the intended use, the external computer will be an embedded computer on the ATRV (1).

The UVT robot is configured in the Roboteq “A & B speed mixed” mode (5), which specifies translational velocity and rotational velocity separately, in arbitrary units. A complete command to the Roboteq controller is of the form \{channel 1, channel 2\}, where “channel 1” is the value of the commanded translational velocity and “channel 2” is the value of the commanded rotational velocity. Details of this coordinated control will be discussed later. The conversion from engineering units (meters/second and radians/second) to the arbitrary units of the controller takes
place in an external computer programmed by UVT personnel. Parameters of the Roboteq controller affect this conversion and must be determined first.

The parameters in question are the “timebase” used by the controller in measuring and controlling motor speed and a “divider” applied to the accumulated encoder count (effectively an odometry measure). The controller manual does not explicitly describe how these parameters are used, so the following discussion is based on a certain amount of reverse engineering.

It appears that the timebase value sets a timer which defines a window within which encoder counts are accumulated. The faster the motor turns, the larger is the value of the count, to the maximum of the counter’s capability. At the end of the window, the number of counts is compared to the speed setpoint and the pulse-width fed to the motors is adjusted to zero the error between setpoint and count. This is a conventional design for this type of controller, and the Roboteq controller is assumed to conform to this design.

The value selected for the timebase determines the range of speeds to which the controller can be set, in this case, {-127:128}. Empirically it was determined that when the timebase is set to a value of 6, the range of setpoints substantially spans the range of speeds which can be attained by the motor. At a timebase value of 5, the motor does not reach its maximum speed at the maximum setpoint, while at a timebase value of 7 the maximum motor speed is reached at a value less than the maximum setpoint.

The “divider” value serves as a gain to the value reported by the odometry counter. This value was set to 128, the largest value accepted by the Roborun software, resulting in the lowest possible resolution of odometry. This resolution was later determined to be more than sufficient, and the value selected does not appear to be critical to this application.

2.1 Odometer

The next step in the calibration is the calculation of conversion factors between engineering units and the arbitrary units of the controller. Empirically it was determined that the counters (one each left and right, corresponding to the rotational displacement of the respective wheel sets) changed by approximately 44000 units during a single rotation of a wheel. The measurement of a rotation was determined by marking a line on the tire corresponding to an alignment with a pointer, zeroing the counter, then rotating the tire 360° using the Roborun utility and noting the counter value at the end. Several measurements were made for each wheel set. Alignment was not precise due to the crude measurement of alignment, but measures were within 0.5% of the mean. From this we conclude that a counter reading of 44000 corresponds to one revolution of the tire. Given a tire of nominal diameter 12” (= .304 m), one encoder count corresponds to 0.000857” of linear travel. The calibration was validated by commanding a straight-ahead translation and measuring the realized translation with a tape measure. Though the actual data has been lost, results were very close.
2.2 Speed Setting

With a means of measuring rotational displacement known, wheel rotational velocity could be calculated. Using a stopwatch to measure time and the Roborun counter display to measure displacement of both left and right wheels, the following findings were made. Elements of this calibration are shown in table 1.

Table 1. Measurements used in calculating conversion factor from wheel rotational velocity in revolutions per minute to Roboteq arbitrary speed units.

<table>
<thead>
<tr>
<th>Speed Setting</th>
<th>Displacement Left (counts)</th>
<th>Displacement Right (Counts)</th>
<th>Duration (s)</th>
<th>V (Wheel RPM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>141631</td>
<td>141052</td>
<td>50.44</td>
<td>3.83</td>
</tr>
<tr>
<td>127</td>
<td>2138332</td>
<td>2172213</td>
<td>30.43</td>
<td>96.58</td>
</tr>
</tbody>
</table>

Speed setting is for Roboteq channel 1 (translational velocity setpoint), while Roboteq channel 2 (rotational velocity setpoint) remains at 0. In the “A & B speed mixed” mode, this provides the same internal speed setpoint for each wheel set.

Measured wheel speed in revolutions per minute is calculated from the raw measurements of figure 1 as:

\[
V = \frac{\text{Displacement} \times \frac{\text{rev}}{44000 \text{ count}}}{\text{Duration} \times \frac{\text{min}}{60 \text{ sec}}} 
\]

The Roboteq speed setpoint, in arbitrary units, is calculated from the desired robot linear speed \( V \) as follows:

\[
\text{Setpoint} = \frac{V}{\pi \times \text{tire_diameter}} \times k \quad (1)
\]

The constant \( k \) is calculated from the data of figure 1:

\[ k = \frac{5}{3.83} \approx \frac{127}{96.58} \approx 1.3 \]
An alternate formulation using input speed $V$ in meters/second, which requires tire diameter in meters, is given as:

$$
Setpoint = \frac{V}{\pi \cdot \text{tire\_diameter}} \times k \times 60
$$

$$
= \frac{V}{\text{tire\_diameter}} \times k_2 = \frac{V}{\text{tire\_diameter}} \times 24.8
$$

(2)

$k_2 = 24.8$

2.3 Turning Radius

Turning radius for a skid-steered robot of track ‘T’ (distance between left and right tires across the direction of travel) is governed by the difference in velocity of each side of the robot, divided by the track. The Roboteq controller in “A & B speed mixed” mode decreases the internal speed setpoint of one wheel set by the setting of channel 2, while increasing the speed of the other by the same amount. If, for instance, channel 1 is set to 0 and channel 2 is set to 20, the setpoint to the right wheel set is forward at speed 20 and the left wheel set turns in reverse at setpoint 20, resulting in a pirouette about the center of the robot. If channel 1 is set to 20 and channel 2 is set to 20, the left wheels are stationary and the right wheels turn at setpoint 40.

This leads to the possibility of the controller issuing a motor speed setpoint greater than the maximum (127 or -126), which cannot be. Empirically it was determined that if the maximum is exceeded for either left or right wheel set, the controller reduces the translational speed so that rotational velocity is preserved while altering the translational velocity as little as possible (e.g., the side-to-side difference in rotational speed is preserved with one wheel set at its maximum speed, the other wheel set adjusted to maintain the maximum possible translational speed.).

<table>
<thead>
<tr>
<th>Channel 1</th>
<th>Channel 2</th>
<th>Left Speed</th>
<th>Right Speed</th>
<th>Left Speed (adj)</th>
<th>Right Speed (adj)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>20</td>
<td>-20</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>0</td>
<td>40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>20</td>
<td>20</td>
<td>60</td>
<td></td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>50</td>
<td>46</td>
<td>146 !!!</td>
<td>27</td>
<td>127</td>
</tr>
</tbody>
</table>

Table 2. Examples of Roboteq controller response to combinations of channel 1 and channel 2 (linear and rotational velocity) setpoints. In the fourth example, the speed setting for the right wheel exceeds the maximum, so both settings are adjusted to maintain rotational velocity at the maximum possible translational velocity.

Note that the channel 2 setting is related to linear wheel speed in engineering units by the same constant as calculated earlier for ground speed, e.g., [1] or [2].
3. Modeling and Calibrating

The kinematics of a four-wheel skid-steer vehicle have been addressed in any number of academic papers and theses. It is not uncommon to begin with a model based on a wheelchair-like differential drive robot (7). The model defines an instantaneous center of rotation (ICR) which lies on the extended line of the axle at a distance defined by the difference in velocity between the two wheels. This model is easy to understand because it obeys the assumptions usual for wheels, that is, that instantaneous motion can occur only perpendicular to the axle.

A four-wheel skid-steer vehicle, however, must violate this assumption whenever it turns. Because the wheels do not steer, they must slip or skid whenever the wheels on opposite sides turn at different rates. The tire-ground interface is difficult to model, it varies with the surface, and it plays a major role in the amount of slip and thus the rate of the turn. Several papers describing dynamic models are cited (2–4), along with calibration methods (6) and results for selected vehicles and surfaces. For the intended purpose of the UVT research vehicle, an elaborate model is not necessary. The vehicle of this report, when finished, will use Simultaneous Localization and Mapping (SLAM) to correct in closed-loop fashion for inaccuracies in the model. A simplified model relating commanded rotation and translation to realized rotation and translation within constrained conditions will suffice. To this end, analysis will proceed in the local coordinate system embedded in the robot, oblivious to any world coordinate system.

The constraint imposed is, slip at each wheel is equal so the center of rotation is at the geometric center C of the robot, at the intersection of the diagonals between the contact point of tires on opposite corners of the robot. Under these constraints, it is easy to separate the analysis of translation and rotation. Analysis of translation is trivial, so the discussion will focus on rotation of the robot about its geometric center. See figure 1.
At this point, it is convenient to represent the vehicle in 2-dimensional plan view. Each tire is represented by a point at the center of its contact patch, a tire-point. The diameter of rotation, labeled Dia in figure 1, is the resultant of the wheelbase B and the track T. The tangential velocity (VT) of each tire-point about the center C is the component of tire longitudinal velocity (VL) which is tangent to its path about the center. This can be formulated as:

\[ VT = VL \cdot \cos(Th) = VL \cdot \frac{T}{Dia} \]

And the length of the diameter Dia = \( \sqrt{T^2 + B^2} \)

The angle Th between the longitudinal velocity and tangential velocity is useful in visualizing the similar triangles, but the direct representation of the inverse cosine, e.g., \( T / Dia \), is more useful and will be used hence.

Rotation velocity of the tire-point about C in radians, denoted (VTr), is:

\[ \frac{VTr}{Dia/2} = 2 \cdot VL \cdot \frac{T}{T^2 + B^2} \]

At this point, the 3-dimensional tire is re-introduced.

From (2),

\[ VTr=2\left(\frac{Setpoint2 \cdot \frac{\text{tire\_diameter}}{k}}{k^2}\right) \cdot \frac{T}{T^2 + B^2} \]

So the channel 2 setpoint for a desired rotational velocity VTr is:
Setpoint2 = VTr \cdot \frac{k2 \cdot (T^2 + B^2)}{2 \cdot T \cdot \text{tire diameter}}

For the UVT robot, track and tire diameter in meters, and input rotational velocity VTr in radian/second, the computing formula is

\[ \text{Setpoint2} = VTr \times 32.5 \]

4. Results

Verification

Software was written to convert translational and rotational velocities in engineering units to arbitrary units according to the formulation described above, and to feed them to the Roboteq motor controller. A 5 m² test grid was laid out on the smooth concrete floor with masking tape at 1 meter intervals. Test trajectories of constant radius were defined as triples consisting of \{time, velocity, rotational velocity\} so that the robot would describe a circle, half-circle, or quarter-circle, beginning from a grid intersection and ending on a grid intersection, within the confines of the test grid. The expected radius of the circle and endpoint were computed from the following trajectories:

Radius = velocity / rotational_velocity = \{ 1, 2 m \}

Endpoint = (rotational_velocity * time) = \{ \pi/2, \pi, 2 * \pi \}

These trajectories were implemented using combinations of the following parameters:

Translational velocity = \{ 0.1, 0.15, 0.2, 0.3 m/s \}

Rotational velocity = \{ 0.1, 0.15, 0.2, 0.3 rad/s \}, both clockwise and counter-clockwise

Duration = \{ 15, 31, 63 s \}

Deviation from the expected trajectory was measured as variation from the expected grid-crossing values at multiples of \(\pi/2\). The measures were by means of measuring tape. Although crude, they were repeatable to the (admittedly low) standards of the test. They were sufficient to ascertain that, once the model was correct, the error was on the order of 2% of distance traveled along the path and less across the path. Angular error proved difficult to measure, but consistent success in approximating the starting point on full-circle trajectories indicates a tolerable error.
5. Conclusions and Further Work

Calibration of the UVT robot low-level controller was intended to assure coarse but sufficient accuracy in following a simple path. The path itself is to be prescribed by a path planner at a higher-level, enhanced by SLAM-based, closed-loop control. The successful execution of semi-circular and circular paths demonstrate that a simple model and correct calibration can provide control as intended, within a limited range of surfaces and velocities. With assurance of adequate accuracy from the lower-level control elements, research can progress in implementing higher-level control and autonomous behaviors built on a foundation of correct mobility.
References


