Interpreting Laser Rangefinder Data

Benjamin Kuipers

January 9, 2001

1 The SICK Laser Rangefinder

A laser rangefinder (LRF) sends an infrared laser pulse in a particular direction and measures the time delay for the reflection, giving a measurement of distance to the nearest obstacle. The laser rangefinders we use on the Intelligent Wheelchair, Vulcan, are manufactured by a German company named SICK.

The SICK laser rangefinder has a 180° field of view. Every 300-400 milliseconds, it scans its field of view at 1° intervals, to return a range-scan.

1.1 Low-Level Interface

The laser-scan driver software is continually collecting the results of laser scans. The low-level interface returns the most recent scan with a time tag.\(^1\)

\[ \text{laser-scan-left, laser-scan-right} \]

\[ \{ t_{\text{scan}}, r_1, r_2, \ldots, r_{180} \} \]

representing the current time and the 180 ranges, obtained from the left or right laser scanner, respectively.

2 Laser Rangefinders on Vulcan

Vulcan, the Intelligent Wheelchair, has two laser rangefinders mounted on the two front corners, facing outward at 45° from forward, so their fields of view overlap in front of the robot (Figure 1).

\(^1\) Unfortunately, this is not the interface currently provided on Vulcan.
Figure 1: The laser rangefinder geometry on the wheelchair. $\psi_L \approx +45^\circ$ and $\psi_R \approx -45^\circ$. 
Vulcan’s egocentric cartesian frame of reference has its origin at the center of the robot, and the positive x-axis facing forward, so the positive y-axis faces left. The zero angle is straight ahead, and positive angles increase to the left (counter-clockwise).

- \((x_L, y_L)\) and \((x_R, y_R)\) are the coordinates of the centers of the left and right laser rangefinders, respectively, in the robot’s egocentric cartesian frame of reference.

- \(\psi_L \approx +45^\circ\) and \(\psi_R \approx -45^\circ\) are the angles between the forward direction of the laser rangefinders and the forward direction of the robot.

- \((x, y, \theta)\) is the state vector (position plus orientation) of the robot with respect to some external (allocentric) frame of reference.

3 Egocentric Interpretation of Laser Rangefinder Scans

Let \(r_i\) be the range detected along the scan ray in position \(i\) from the left laser rangefinder. Our goal is to find the coordinates of the point detected at the end of this ray, in the egocentric frame of reference of the robot. (We will assume that all angles are in degrees.)

- Egocentric angle of \(r_i\) with respect to LRF forward: \(90 - i\).

- Egocentric angle of \(r_i\) with respect to Vulcan forward: \(\psi_L + 90 - i\).

- Endpoint coordinates wrt LRF center and Vulcan forward:
  \[
  \begin{pmatrix}
  r_i \cos(\psi_L + 90 - i) \\
  r_i \sin(\psi_L + 90 - i)
  \end{pmatrix}.
  \]

- Endpoint coordinates wrt Vulcan center, Vulcan forward:
  \[
  \begin{pmatrix}
  x_L \\
  y_L
  \end{pmatrix} + \begin{pmatrix}
  r_i \cos(\psi_L + 90 - i) \\
  r_i \sin(\psi_L + 90 - i)
  \end{pmatrix}.
  \]

- If \(r_j\) comes from the right laser rangefinder, the endpoint is:
  \[
  \begin{pmatrix}
  x_R \\
  y_R
  \end{pmatrix} + \begin{pmatrix}
  r_j \cos(\psi_R + 90 - j) \\
  r_j \sin(\psi_R + 90 - j)
  \end{pmatrix}.
  \]
4 Allocentric Interpretation of Laser Rangefinder Scans

Suppose we wish to find the endpoint of the same scan ray \( r_i \), but this time in an external (allocentric) frame of reference, within which the robot is at \((x, y, \theta)\). This is useful if, for example, we are building an occupancy grid from the scans.

- Extend the egocentric derivation to find the angle of the ray with respect to the allocentric frame of reference:
  \[
  \theta + \psi_L + 90 - \iota.
  \]

- Endpoint coordinates with respect to LRF center and allocentric angular reference:
  \[
  \begin{pmatrix} r_i \cos(\theta + \psi_L + 90 - \iota) \\ r_i \sin(\theta + \psi_L + 90 - \iota) \end{pmatrix}.
  \]

- In order to find the coordinates of the LRF centers in the allocentric frame of reference, we need to rotate the vectors \((x_L, y_L)\) and \((x_R, y_R)\) into the allocentric frame, through the rotation matrix:
  \[
  R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.
  \]

- Putting these together, the endpoint of ray \( r_i \) from the left LRF is mapped to the allocentric coordinates:
  \[
  \begin{pmatrix} x \\ y \end{pmatrix} + R \begin{pmatrix} x_L \\ y_L \end{pmatrix} = \begin{pmatrix} r_i \cos(\theta + \psi_L + 90 - \iota) \\ r_i \sin(\theta + \psi_L + 90 - \iota) \end{pmatrix}.
  \]

4.1 Scanning Free Space Along the Ray

When building an occupancy grid, it is useful to step along the ray \( r_i \) to identify the grid cells passed through on the way to the terminal point. The ray provides evidence that these cells are empty, just as it provides evidence that the endpoint cell is occupied.

We use an index \( k \) that steps through \([0, 1)\) to produce a series of points along the ray, separated by a third the width of a grid cell. The points are defined by:

\[
\begin{pmatrix} x \\ y \end{pmatrix} + R \begin{pmatrix} x_L \\ y_L \end{pmatrix} + k \cdot \begin{pmatrix} r_i \cos(\theta + \psi_L + 90 - \iota) \\ r_i \sin(\theta + \psi_L + 90 - \iota) \end{pmatrix}.
\]
These calculations describe space as $\mathbb{R}^2$, with real-valued coordinates. The occupancy grid breaks this up into discrete cells. Define the frame of reference and the scale so that points with integer coordinates correspond to the center of the cells. Then the mapping from real-valued coordinates to grid cell is a simple rounding operation to the nearest integer.

5 Mid-Level Interface

The mid-level interface takes the robot’s allocentric position and orientation $(x, y, \theta)$ as its only argument, and returns the set of points at the end of the rays in the current range scan, with respect to the allocentric frame of reference. If the points are desired in the egocentric frame, simply provide $(0, 0, 0)$ as the current position and orientation.

range_scan $(x, y, \text{theta})$ Returns a sequence of 721 values:

$$[t_{scan}, x_1, y_1, x_2, y_2, \ldots, x_{360}, y_{360}]$$

representing the current time and 360 points, 180 each from the left and right laser rangefinders.

6 High-Level Interface: The Occupancy Grid

The high-level interface creates a local occupancy grid by probabilistic mapping and incremental localization.

- **Mapping.** Given the current most likely value of $(x, y, \theta)$, update the occupancy grid from the information in the current laser scan.

- **Localization.** After a small motion, use the current occupancy grid and the current laser scan to determine the most likely value of $(x, y, \theta)$.

A separate document will describe the methods for these steps.

\[\text{Needs:}\]

- *Self-calibration algorithms for } \psi_L \text{ and } \psi_R.*