QUANSER
INNOVATE · EDUCATE

Concept Review

LiDAR Inverse Measurement Model

Why Model a LiDAR?

Modern autonomous vehicles make decisions on where to travel in their environment based on perception interpretation techniques. These techniques can be applied to a global understanding of the vehicle's position or a local planner in charge of deciding what areas are safe to drive on. In this document the inverse measurement model of a LiDAR is described.

LiDAR Inverse Measurement Model

A LiDAR inverse measurement model is typically used to produce the likelihood of a cell being occupied according to the latest sensor measurement: $p(m_{i,j}|\ \mathbf{z}_t, \mathbf{x}_t)$. This document is meant to accompany the Occupancy Grid Concept Review. From each scan, a LiDAR produces a list of s points in polar coordinates giving the range and angle relative to the lidar centre, namely:

$$\mathbf{z}_t = \begin{bmatrix} (\phi_{1,t}, r_{1,t}) \\ (\phi_{2,t}, r_{2,t}) \\ \vdots \\ (\phi_{s,t}, r_{s,t}) \end{bmatrix}$$

Where $\boldsymbol{z_t}$ is an array containing distance measurement r_i corresponding to each angle ϕ_i. $\mathbf{x}_t$ is the pose of the LiDAR in the world frame at time t. Each angle measurement is relative to the sensor's global heading. The distance values returned are restricted to lie within the range of 0 to r_{max} where r_{max} is the furthest distance that can be measured by the sensor. In a similar manner, the heading values are restricted to lie within the LiDAR's field of view: $0 \leq \phi \leq \phi_{max}$. A LiDAR's measurement envelope is visualized in Figure 1 with the relevant parameters labeled.

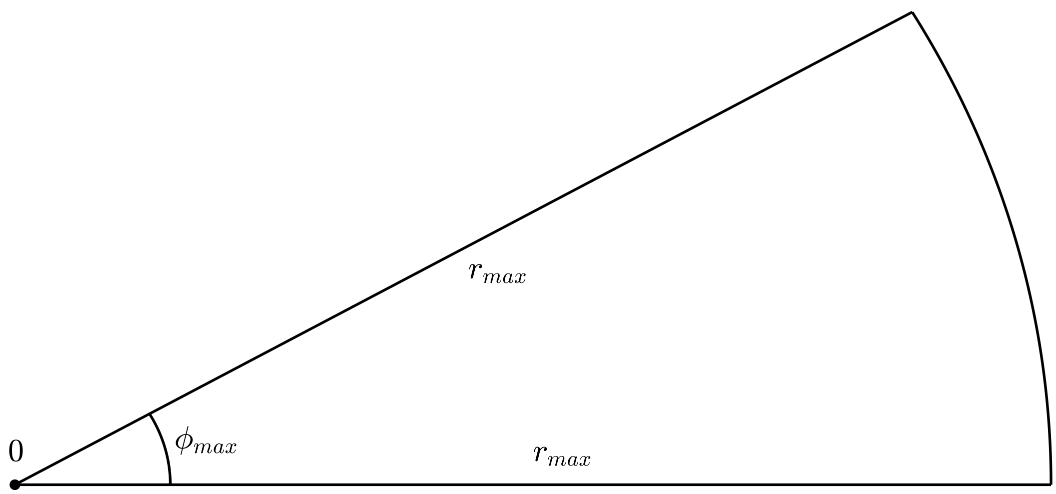

Figure 1: Relevant parameters relating to a LiDAR's measurement envelope.

Before we can update our occupancy grid map of the environment, we need to find a way to convert the scan information (given in polar coordinates) to the cartesian coordinates used for representing the world map. Additionally, we need to determine what the scan can tell us about the likelihood of each cell being occupied.

The complete inverse measurement process can be completed in three steps:

1. Construct a secondary grid of cells in the lidar frame.
2. Calculate the likelihood of each cell in the lidar frame being occupied.
3. Use bilinear interpolation to transform cell values from the lidar frame to the world frame.

Since the output of a lidar is given in polar coordinates, we can easily determine the likelihood of each cell in the polar grid being occupied. For example, consider the polar grid shown in Figure 2: scan points are represented with black dots, white cells indicate, low probability, red cells indicate high probability, and gray cells indicate prior probability.

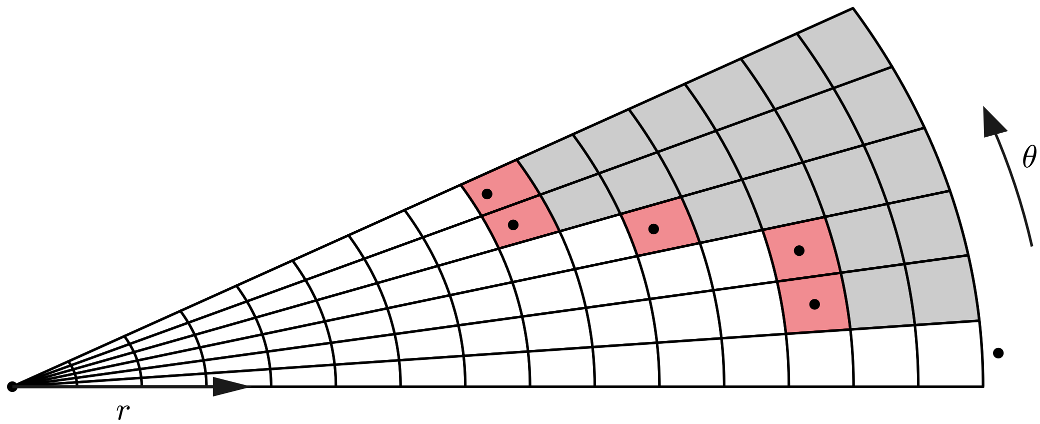

Figure 2: Occupancy grid from lidar scan in polar form.

When looking at the measurement envelope the probability of each cell can be described as,

- If a cell lies outside the field of view of the sensor, the likelihood is the prior p_0.
- If a cell lies between the sensor and a measured object, it has a low probability of being occupied, p_{free}.
- If a measurement lies within the area of a cell, the cell has a high probability of being occupied, p_{occ}.

The value of each cell can be determined as follows:

$$p(w_{\phi,r}|\mathbf{z}_t, \mathbf{x}_t) = \begin{cases} p_{free} & : \quad r < r_\phi \\ p_{Occ} & : \quad r = r_\phi \\ p_0 & : \quad r > r_\phi \end{cases}$$

Where $\mathrm{w}_{\phi,r}$ indicates the cell at range r and angle ϕ in the polar grid. If you imagine stretching out the polar grid shown in Figure 2, you can see that we can represent the polar grid as a 2d matrix, as shown in Figure 3. This makes it very easy to work with and compute.

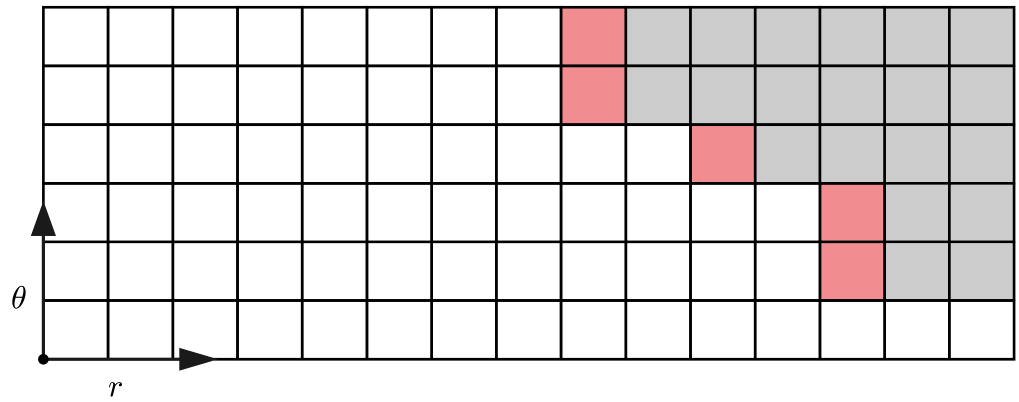

Figure 3: Polar grid represented as a 2D matrix.

The last step is to identify the likelihood of each cell in the world map, $m_{x,y}$ being occupied given the polar grid w. Visually, the relation between the two frames can be seen in Figure 4.

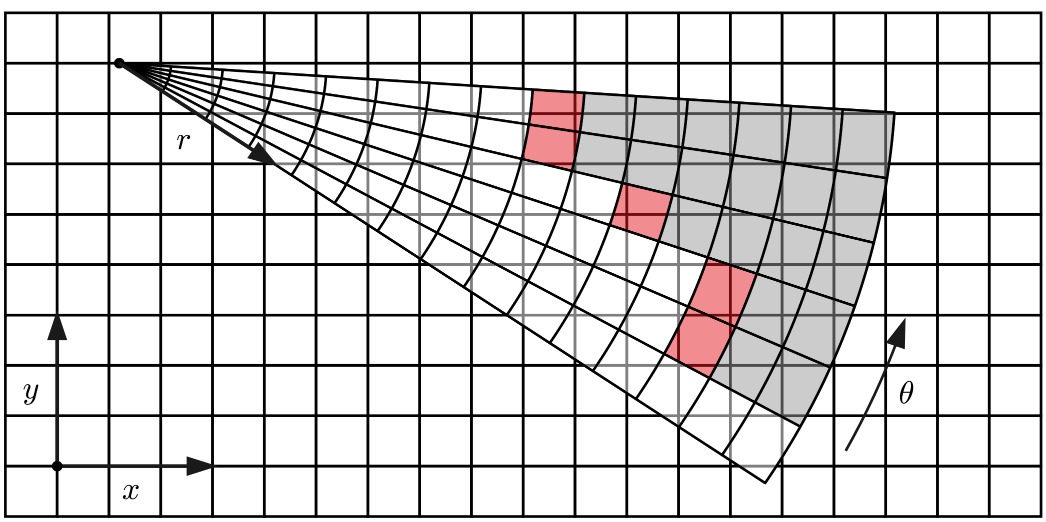

Figure 4: Polar Grid overlaid on top of a cartesian grid.

Given sensor location (x_t, y_t), the distance to cell i, j in the world map can be found as

$$r_{ij} = \sqrt{\left(c_{ij,x} - x_t\right)^2 + \left(c_{ij,y} - y_t\right)^2}$$

Where $(c_{ij,x}, c_{ij,y})$ represents the x and y grid locations for cell c_{ij}. The heading angle to each cell can be found as

$$\phi_{ij} = \text{atan2}\left(\frac{c_{s,y} - y_t}{c_{s,x} - x_t}\right) - \psi_t$$

With ψ_t representing the heading of the LiDAR in the global frame. Finally, the value of $p\left(m_{i,j} \middle| \mathbf{z}_t, \mathbf{x}_t\right)$ can be calculated using "`scipy.ndimage.map_coordinates`" to perform the bilinear interpolation with $p(w|\mathbf{z}_t, \mathbf{x}_t)$ and (r_{ij}, ϕ_{ij}) as input. The resulting map update is shown in Figure 5.

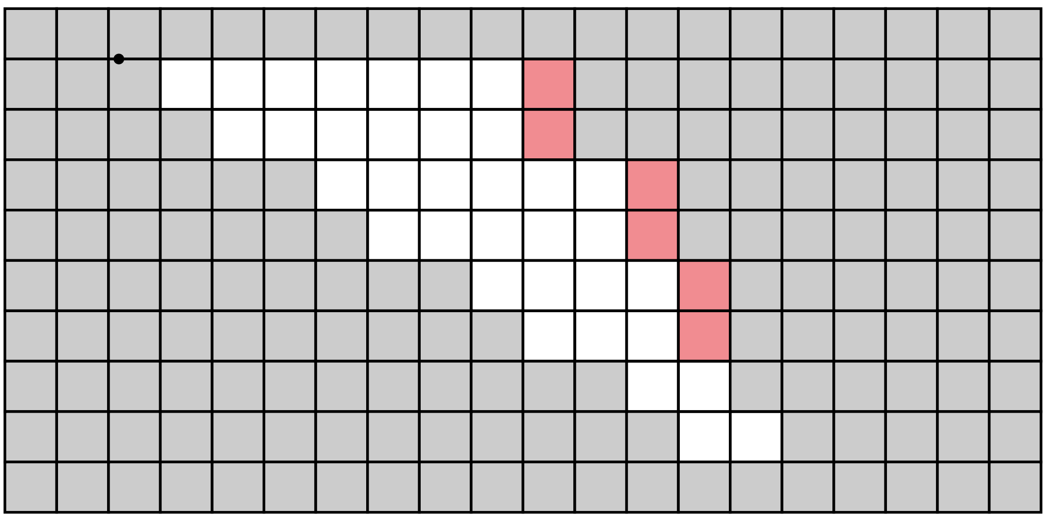

Figure 5: Post interpolation update of the world map.

References

Yguel, Manuel, Olivier Aycard, and Christian Laugier. "Efficient GPU-based construction of occupancy grids using several laser range-finders." *International Journal of Vehicle Autonomous Systems* 6.1-2 (2008): 48-83.

Homm, Florian, et al. "Efficient occupancy grid computation on the GPU with lidar and radar for road boundary detection." *2010 IEEE intelligent vehicles symposium*. IEEE, 2010.

Solutions for teaching and research. Made in Canada.