Populating Occupancy Grids from LIDAR Scan Data

Course 4, Module 2, Lesson 2 – Part 2

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Learning Objectives

- Create a simple Inverse Measurement Model
- Discuss an improvement using Bresenham line algorithm

Inverse Measurement Module

$$l_{t,i} = \mathrm{logit}\boxed{\left(p(m^i|y_t)\right)} + l_{t-1,i} - l_{0,i}$$

- State of the occupancy grid given a measurement
- So far we have only seen the following measurement model:

$$p\left(y_t \middle| m^i\right)$$

  - State of the occupancy grid given a measurement
- A inverse measurement model is needed!

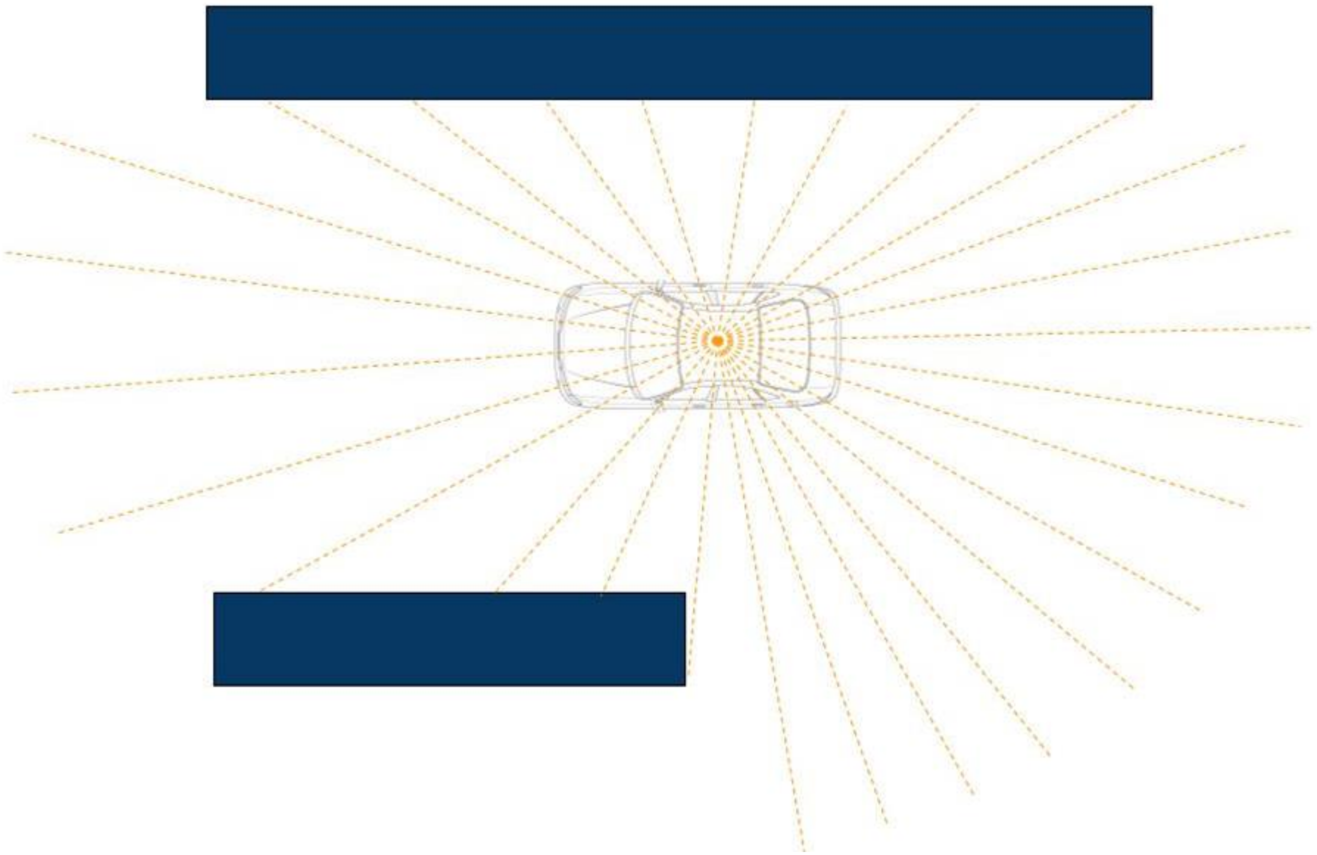
Inverse Measurement Module

- Scanner bearing:

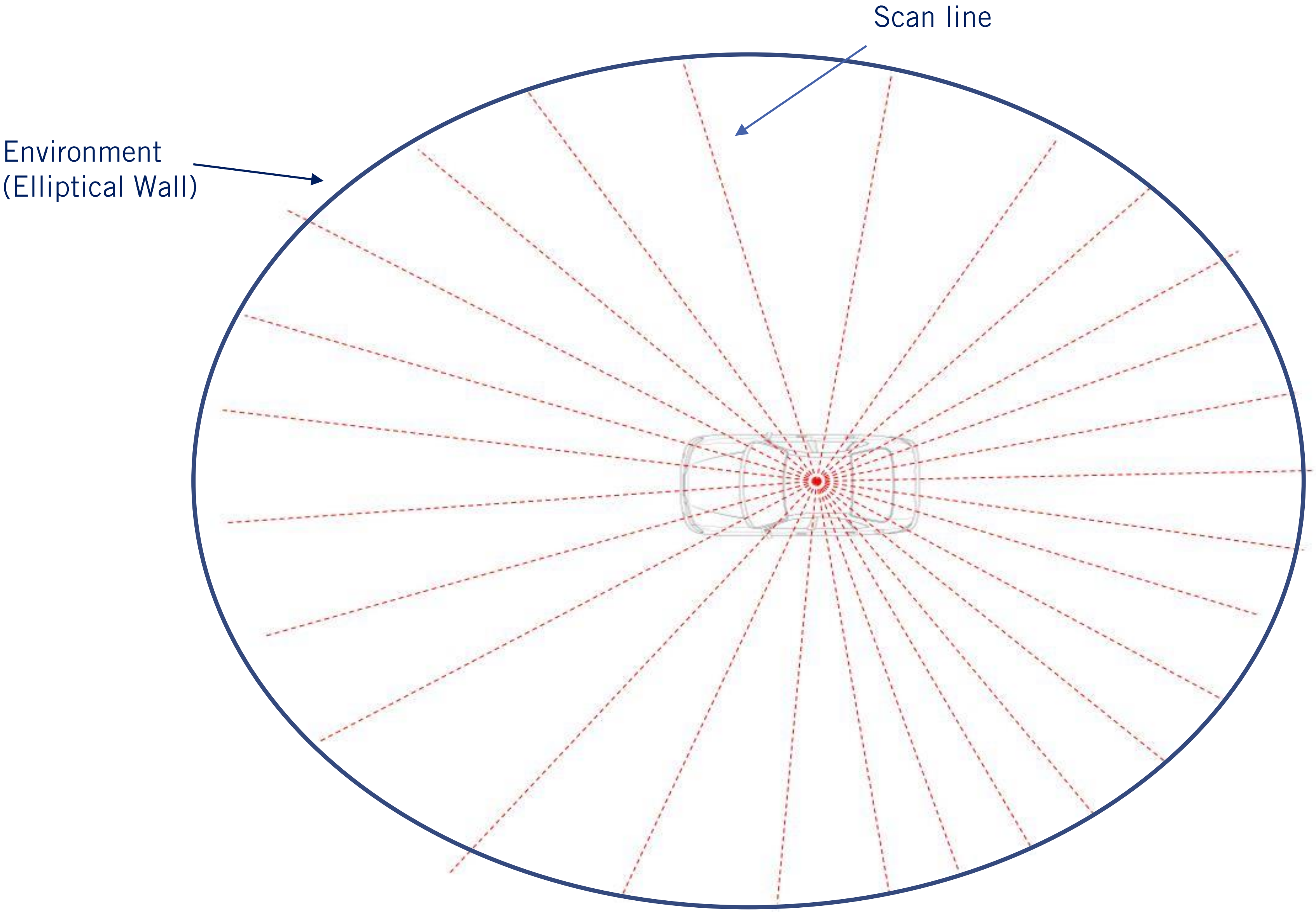
$$\phi^s = [-\phi^s_{max} \quad \cdots \quad \phi^s_{max}] \qquad \phi^s_j \in \phi^s$$

- Scanner ranges:

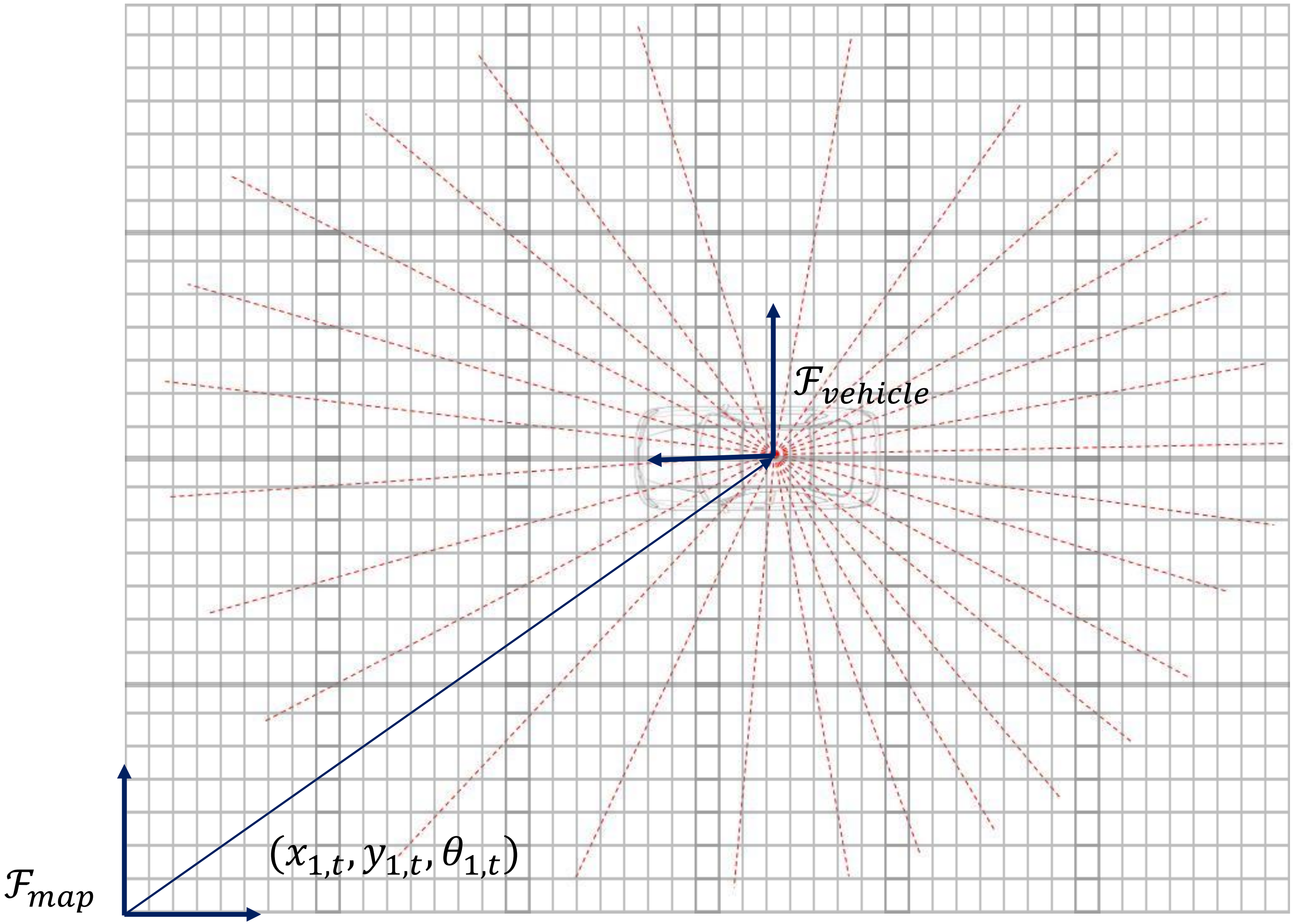
$$r^s = \left[r^s_1 \quad \cdots \quad r^s_j\right] \qquad r^s_j \in [0, r^s_{max}]$$



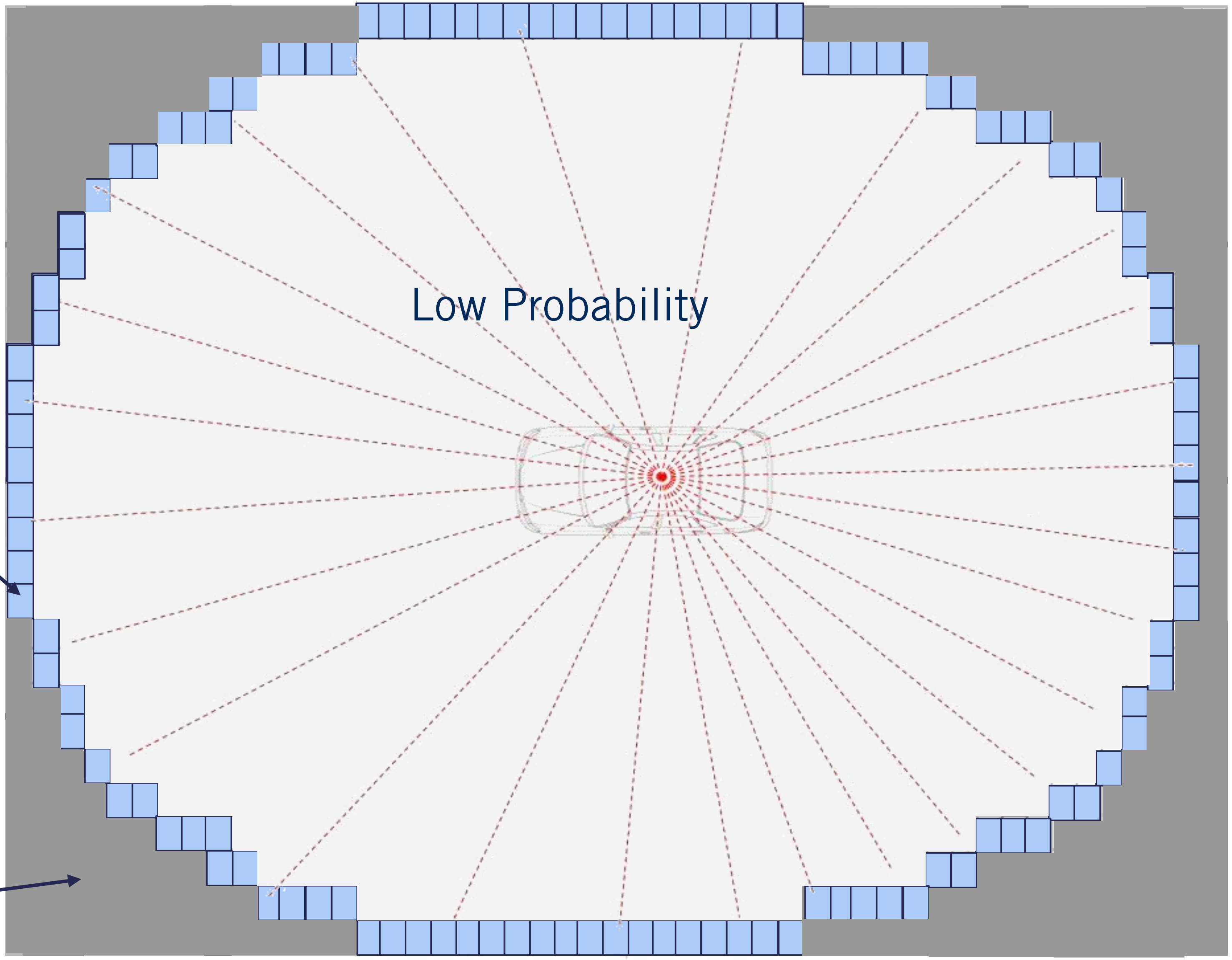
Inverse Measurement Module



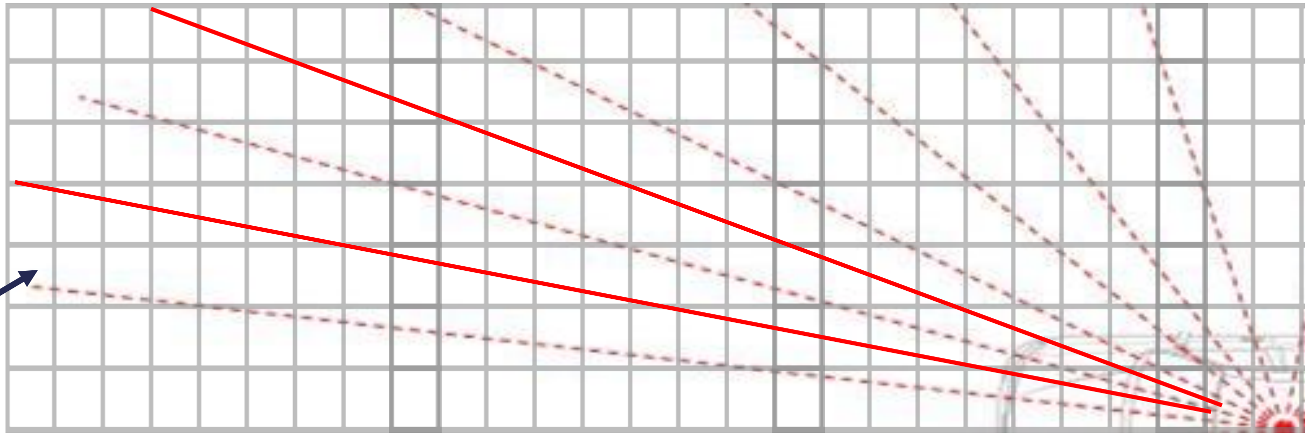
Inverse Measurement Module



Inverse Measurement Module



Inverse Measurement Module – To be fixed



(r^i, ϕ^i)

Closest relative bearing:

$$k = \operatorname{argmin}\left(\left|\phi^i - \phi_j^s\right|\right)$$

Relative range:

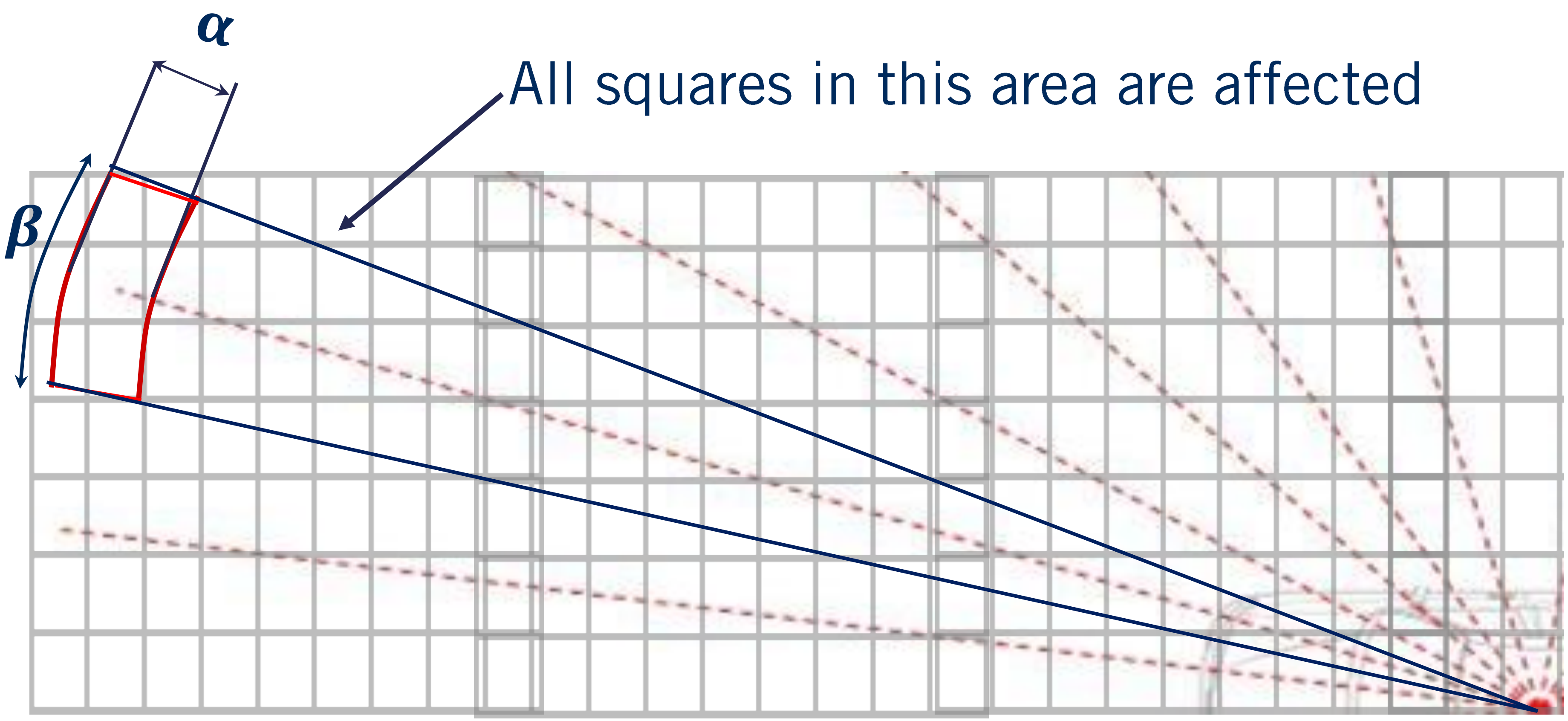
$$r^i = \sqrt{\left(m_x^i - x_{1,t}\right)^2 + \left(m_y^i - x_{2,t}\right)^2}$$

Relative bearing:

$$\phi^i = \tan^{-1}\left(\frac{m_y^i - x_{2,t}}{m_x^i - x_{1,t}}\right) - x_{3,t}$$

Inverse Measurement Module

- **α** - defines the affected range for high probability
- **β** - defines the affected angle for low and high probability



Inverse Measurement Module - Algorithm

- No Information

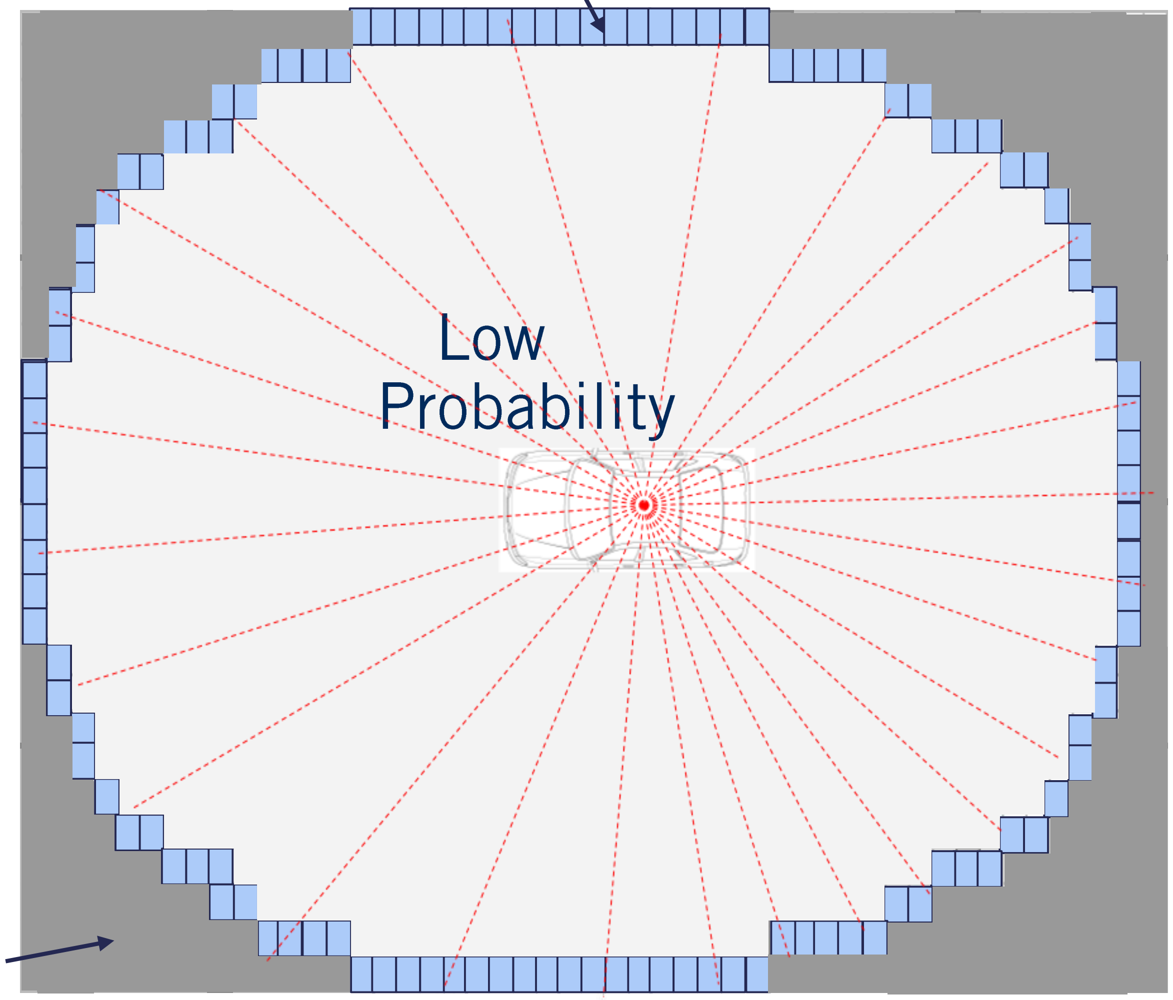
  if $r^i > \min(r^s_{max})$ or $|\phi^i - \phi^s_k| > \beta/2$

- High probability

  if $r^s_k < r^s_{max}$ and $|r^i - r^s_k| > \alpha/2$

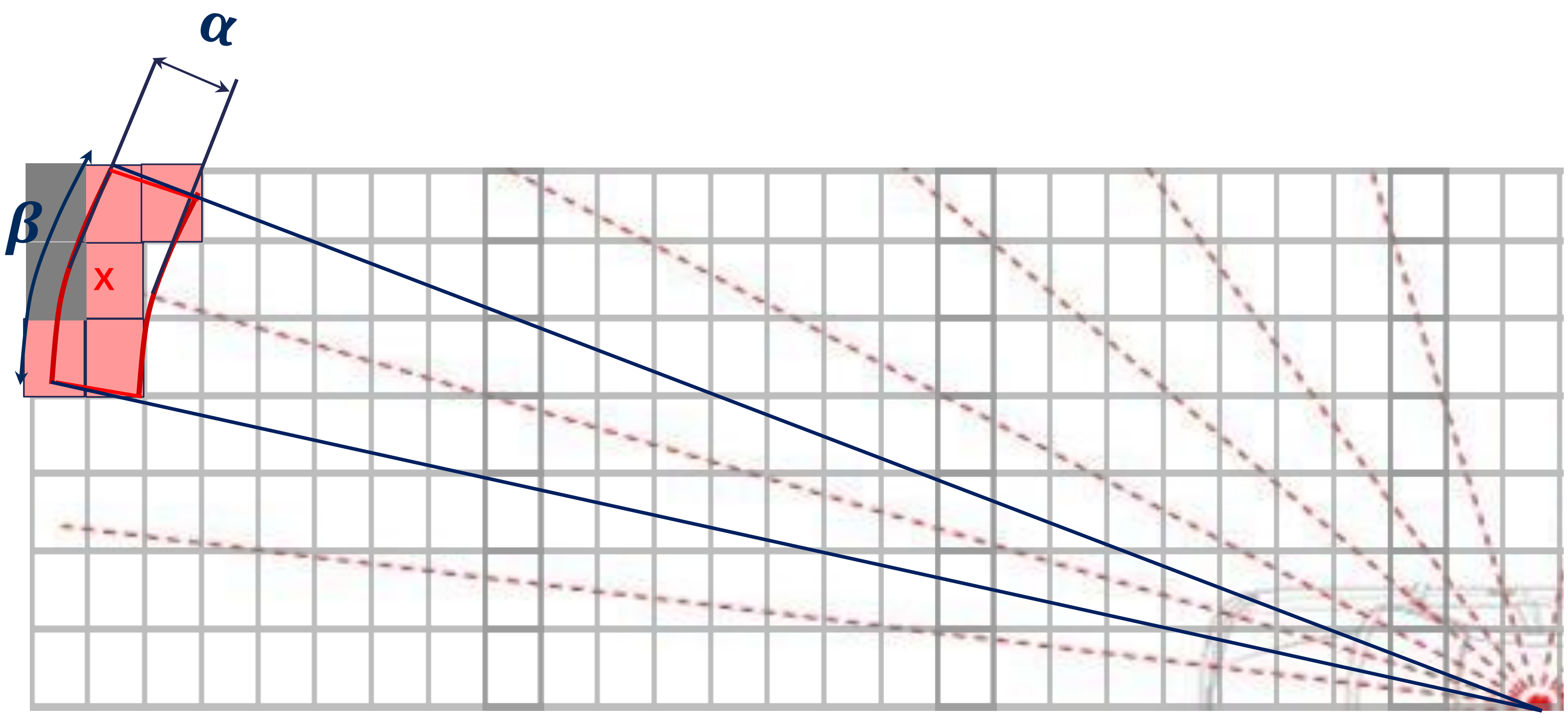
- Low probability

  if $r^i < r^s_k$



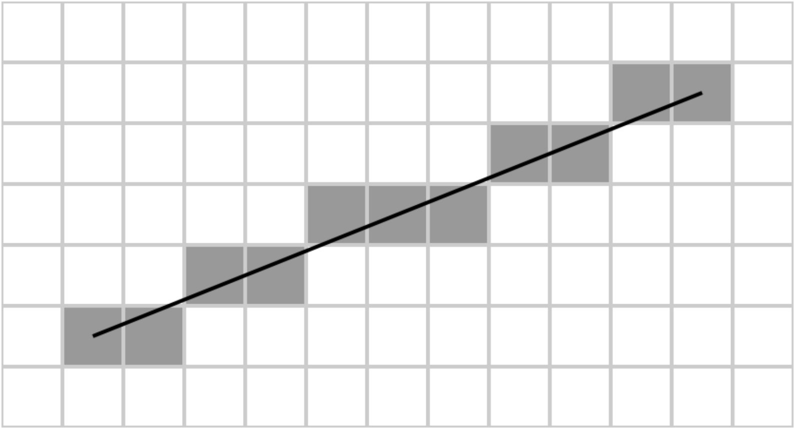
Inverse Measurement Module

- Example – red cells denote high probability of occupied, given measurement denoted by red x.



Inverse Measurement Module With Ray Tracing

- Ray tracing algorithm using Bresenham's line algorithm
  - Rasterized line algorithm
  - Uses very cheap fixed point operations for fast calculations
- Perform update on each beam from the LIDAR rather then each cell on the grid
  - Preforms far fewer updates (ignores no information zone)
  - Much cheaper per operation



Summary

- Create a simple Inverse Measurement Model
- Discuss an improvement using Bresenham line algorithm