Sonar Mapping And Object Detection

Faiyaz Chowdhury

Donald Innocent-Ike

Jordan Miller

Vishnu Perumal

ECE2031 Digital Design Laboratory L04

Georgia Institute of Technology

04/20/2016

Project Objectives

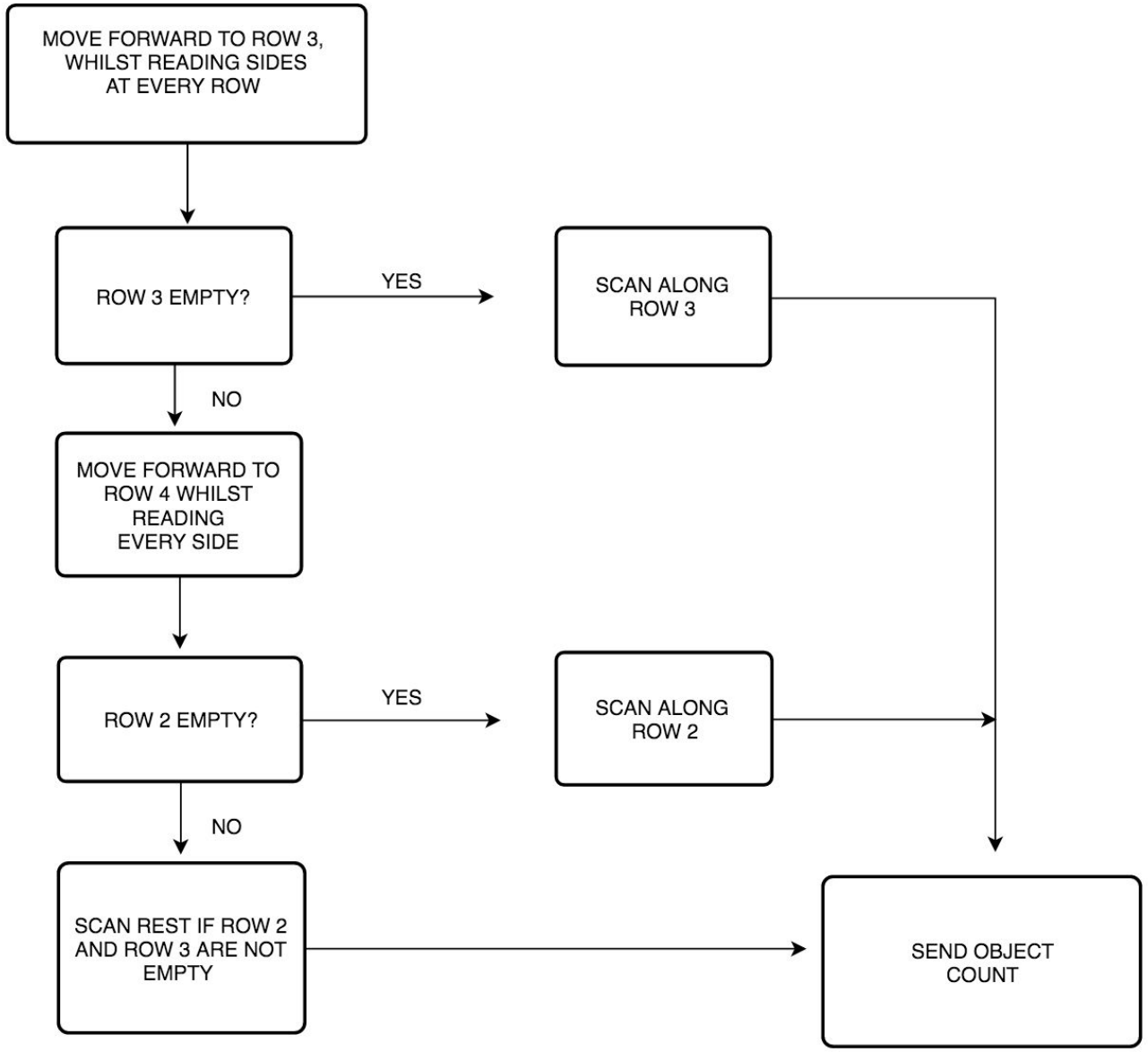
- Output coordinates of each object as X '___' Y '___'
- Output total number of objects as N '___'
- Output all results before the end of a 60 second run time
- Robot must not collide with anything

Overall Project Design Solution

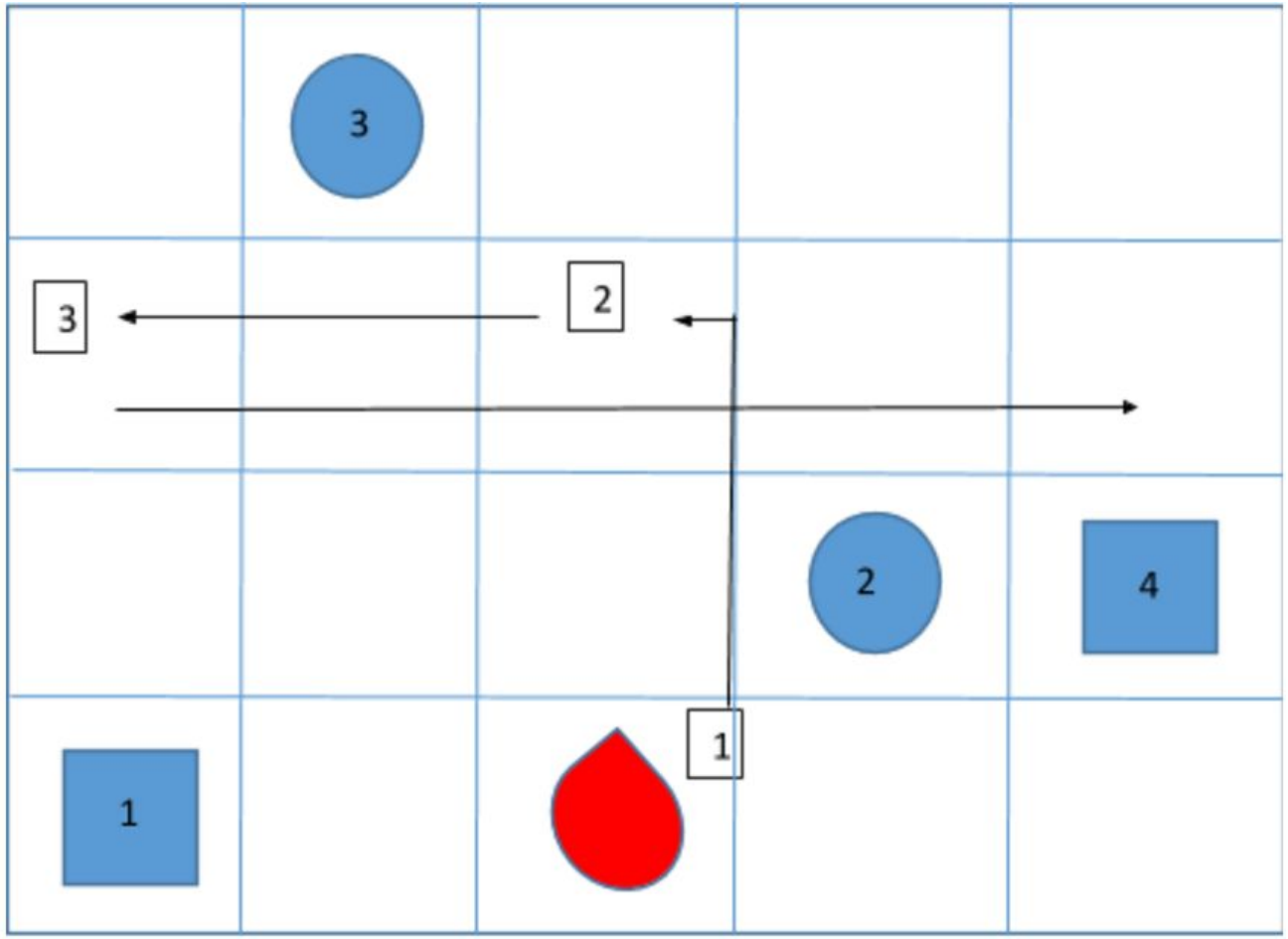
- Moves down empty/half empty aisles.
- Makes use of multiple states.
- Counts number of objects from 16 bit binary number.
- Coordinates stored in 16 bit binary number.

Design Interpretation

Design Proposal Synopsis



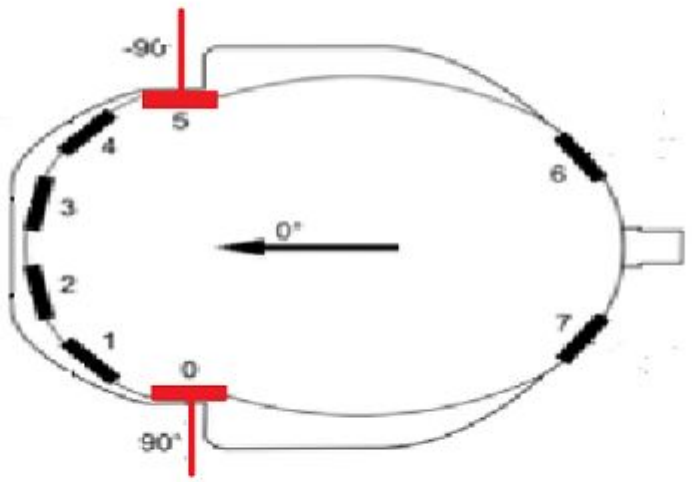
Object Path In Arena



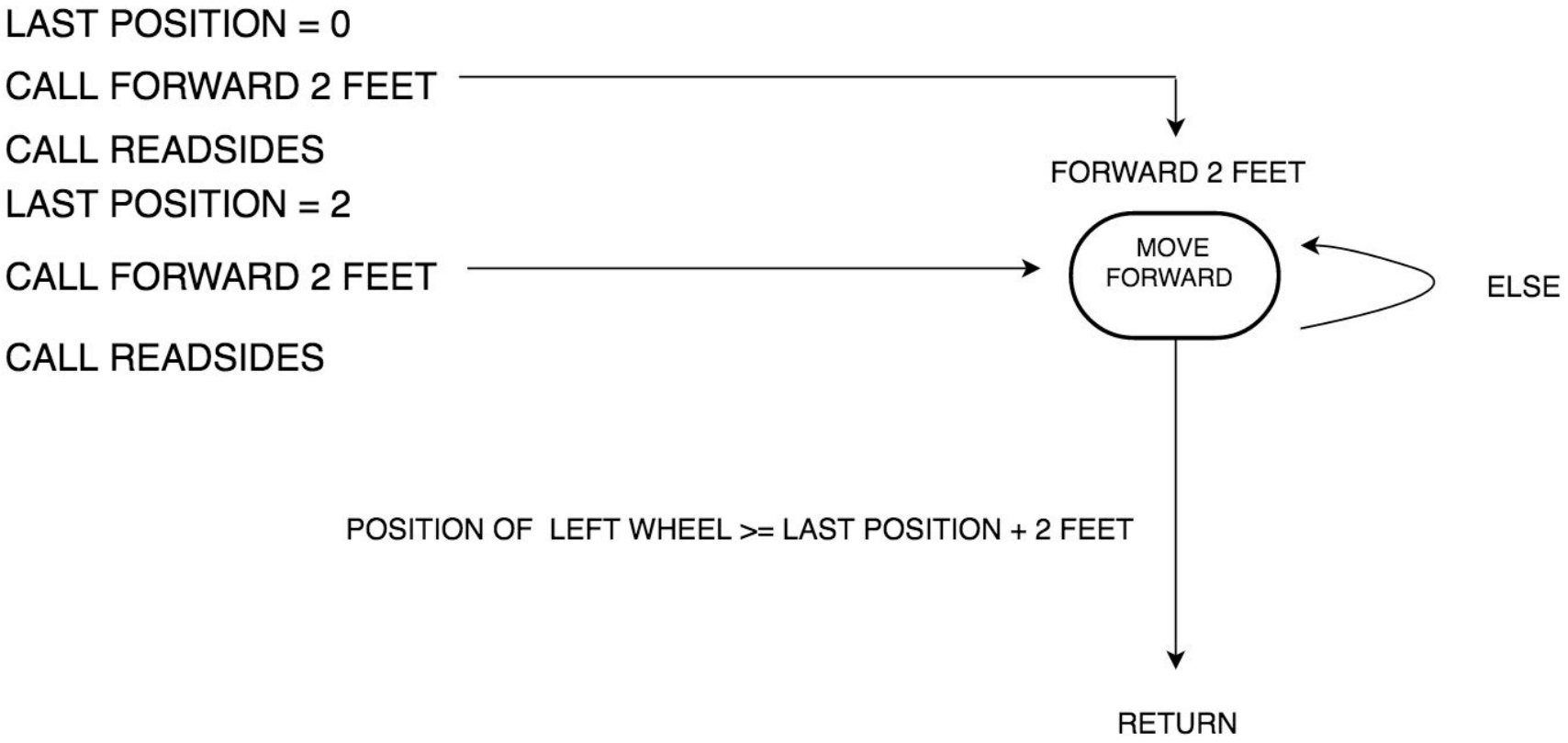
Sample movement of robot with random arrangements of objects.

Subsequent Movements And Object Detection

- Move in two foot increments → search either side of center aisle
- Move down any empty rows
- Search down farthest columns for hidden objects



Odometer Subroutines Prevent Error



Matrix of Coordinates

	X1	X2	X3	X4	X5
Y4	13	14		15	16
Y3	9	10		11	12
Y2	5	6		7	8
Y1	1	2		3	4

Y2 = 0B0000000011110000
X4 = 0B0100010001000100
LocMat = 0B0000000001000000
7th bit

Computation of Coordinates

EXAMPLE:

0B0000000011110000

0B0100010001000100

0B0000000001000000

ROWMAT: Y value of 2
COLMAT: X value of 4

LOCMAT: X04 Y02

Problems Encountered

Double Counting Objects

Problem: DE2Bot outputting incorrect number of objects

Solution:

- Created a 16 bit number
  - Each bit represents if an object exists within a coordinate
  - One means object exists in the respective coordinate
- Count number of objects at END of code
  - Check if MSB is true
  - Add one to the object count if MSB is true
  - Shift 16 bit number left and repeat 16 times

Row Searching

Problem: Half empty rows

- Bot will not enter half empty rows
- Misses objects hidden from the center aisle

Problem: Scraping or bumping walls

- Bot bumps the outer wall due to heading drifts
- Bot sometimes scrapes the walls in rows one and four

Future Improvements To Be Considered

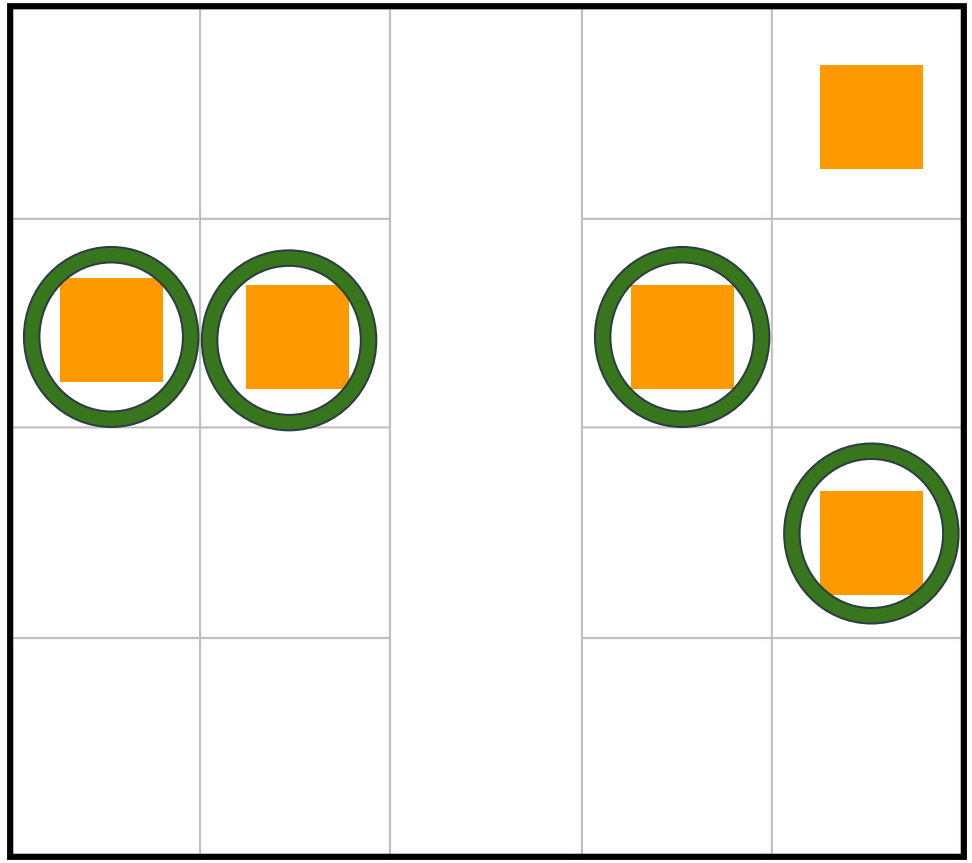
- Reduce the number of states → reuse if possible
- Implement better movement to avoid collision with walls
- Implement half empty first and fourth row scenarios
- Avoid sending the same coordinate twice

Strengths of Solution

- Odometer
- Row 3 and Row 2
  - Efficient
- Knows if row is empty without scanning it again
- Location Matrix
- No double count
  - Object Count < 7

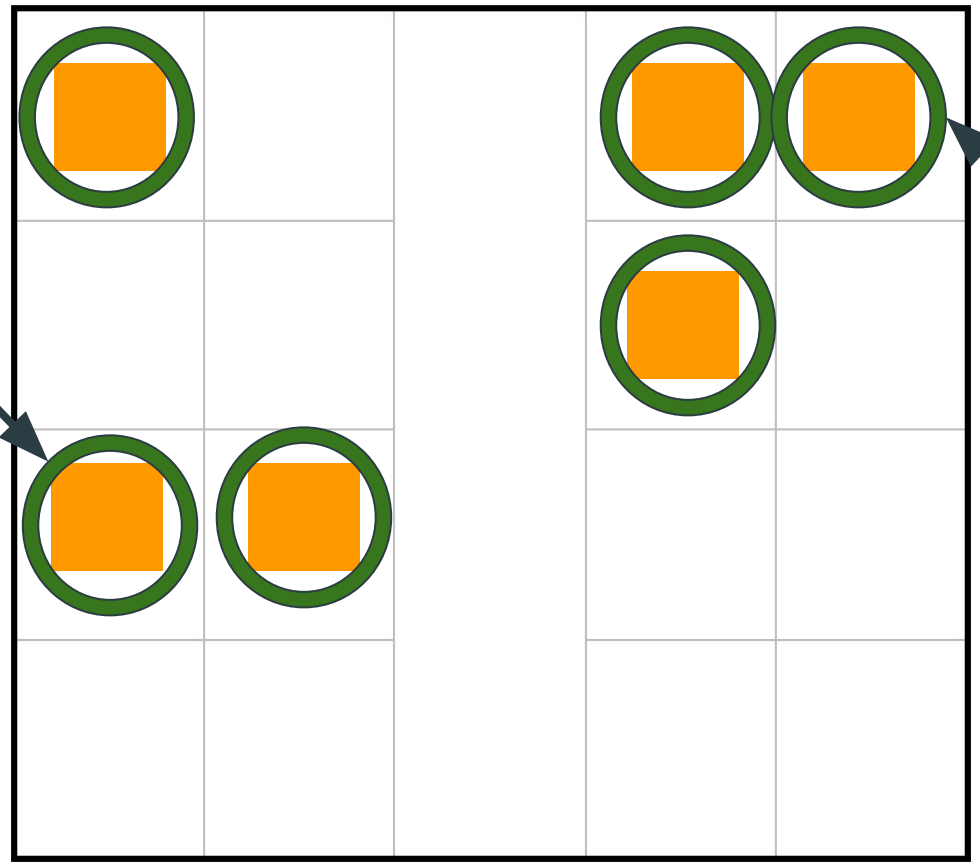
Results

Trial One



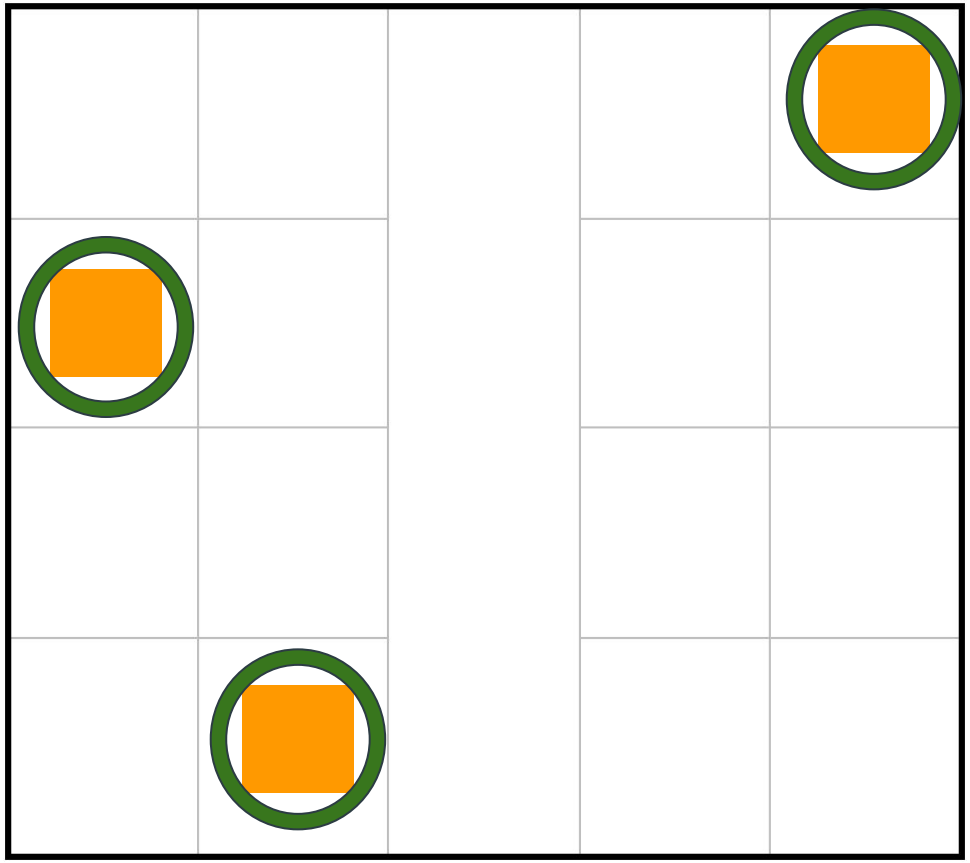
Trial Two

Partial credit for sending (2,2) twice



Partial credit for sending (5,4) twice

Trial Three



Questions?