CSCI 473: Project 2 Report

Léo Chely

***Abstract*— This report discusses the implementation details of the robot simulation trained by reinforcement learning to follow walls.**

I. INTRODUCTION

In this report, the parameters of the training and sensing will be presented. The Q-Table implementation will be detailed as well as the model parameters. Finally, the training and final result will be commented.

II. ALGORITHM DESIGN

A. Sensing

The sensing field has been divided in 4 zones:

- Left: 120°- 180°
- Front: 60°- 120°
- Right-Front: 60°- 30°
- Right: 60°- 0°

B. State, action and reward definitions

The sensor values have been discretized in 2 zones for the left and right-front regions, 4 zones for the front region and 5 zones for the right region.
A state is therefore defined as a combination of zones.
There are 3 actions:

- Turn left
- Turn right
- Go forward

The reward is either -1 if the robot is detecting a close wall on the left, right or front, or no wall at all on the right. Otherwise, the reward is 0.

C. Q-Table

The Q-Table is a numpy array of size 4x2x5x2x3 corresponding to all the combinations of states and actions. All the entries are initialized to 1.

D. Model Parameters

The parameters used for the model are the ones recommended:

- $\epsilon_0 = 0.9$
- $d = 0.985$
- $\alpha = 0.2$
- $\gamma = 0.8$

III. TRAINING

A. Episodes

The number of episodes needed varied but around 100 episodes were usually enough to get consistent good results.

B. Training Time

The training time of each episode varied from a few seconds to a couple minutes. The first episodes were very short because the robot gets stuck fast. The latest ones are longer due to the robot being closer from the expected behavior.

C. Experimental results

The trained robot performed as expected in all the scenarios. The robot follows the wall on its right and properly handles all types of corners (see fig 1, 2 and 3).

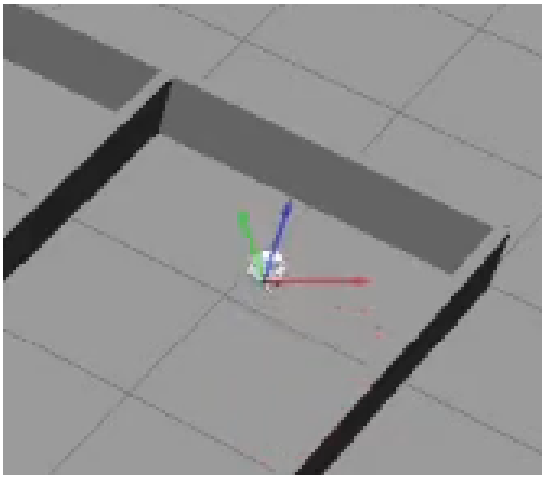

Fig. 1. Left turn

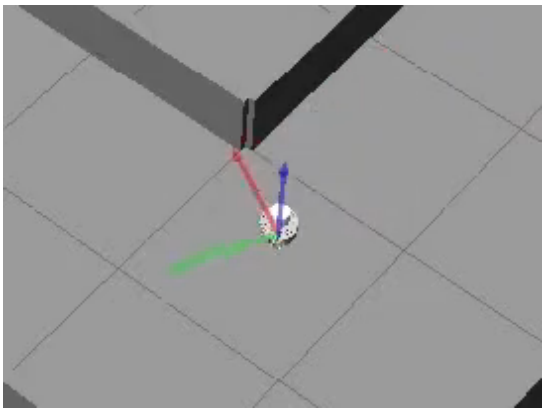

Fig. 2. Right turn

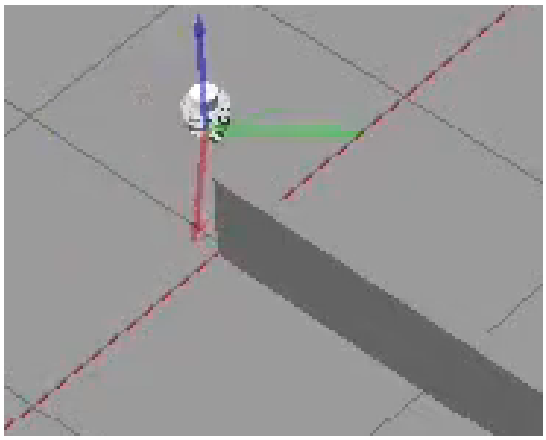

Fig. 3. U turn