

DeepRL Project

Reinaldo Ossuna

April 26, 2019

1 Introduction

The goal of this project is to create a DQN agent and define a reward functions to teach a robot arm to carry out two primary objectives:

- * Have any part of the robot arm touch the object of interest, with at least a 90% accuracy.
- * Have only the gripper base of the robot arm touch the object of interest, with at least a 80% accuracy.

2 Reward Functions

The arm joints were updated using the position control, since this resulted a better result in the simulation.

A positive reward was set for any any collision between the arm and the tube (in the second task only give the reward for collision between the gripper and the tube). And penalties for collision with the ground.

A interim reward function was added based on the distance from the gripper and the tube.

2.1 1. Objective

- REWARD_WIN +20
 - * Collision between the arm and the tube
- REWARD_LOSS -20
 - * Collision between the arm and the ground
- REWARD_INTERIM 2
 - ALPHA = 0.1
 - $\text{avgGoalDelta} = (\text{avgGoalDelta} * \text{ALPHA}) + (\text{distDelta} * (1.0f - \text{ALPHA})) ;$
 - $\text{rewardHistory} = \text{avgGoalDelta} * \text{REWARD_INTERIM} ;$

2.2 2. Objective

- REWARD_WIN +500
* Collision between the arm and the tube
- REWARD_LOSS -500
* Collision between the arm and the ground
- REWARD_INTERIM 10
ALPHA = 0.1

3 Hyperparameters

- Input_Width x Input_Height
Keeping the size of each image small helped in training. bigger images implies in more parameters to be trained.
define INPUT_WIDTH 64 define INPUT_HEIGHT 64
- Batch size
For the batch size 32 as a good number, a large number of batch require more memory.
- Optimizer
In this project Adam and RMSprop were tested and RMSprop showed be a better option.
- Learning_rate
The learning rate is a key parameters, a large value can result in the model to not converge. To the first task the 0.1 value was a good value, but in the second task the value 0.01 was required to complete the task.
- LSTM
We used the LSTM architecture to allow the network to use past frames in consideration instead of the only actual frame. A large number can cause memory problems, 256 was chosen.

4 Results

The first task the network was able to achieve really quick with a small effort. The second task the performance is not stable and sometimes the arm will not perform the task.

5 Future Work

The use of the Gazebo environment allow us to use a large amount of robots in different tasks. I want to try to recreate some previous projects using DQN.

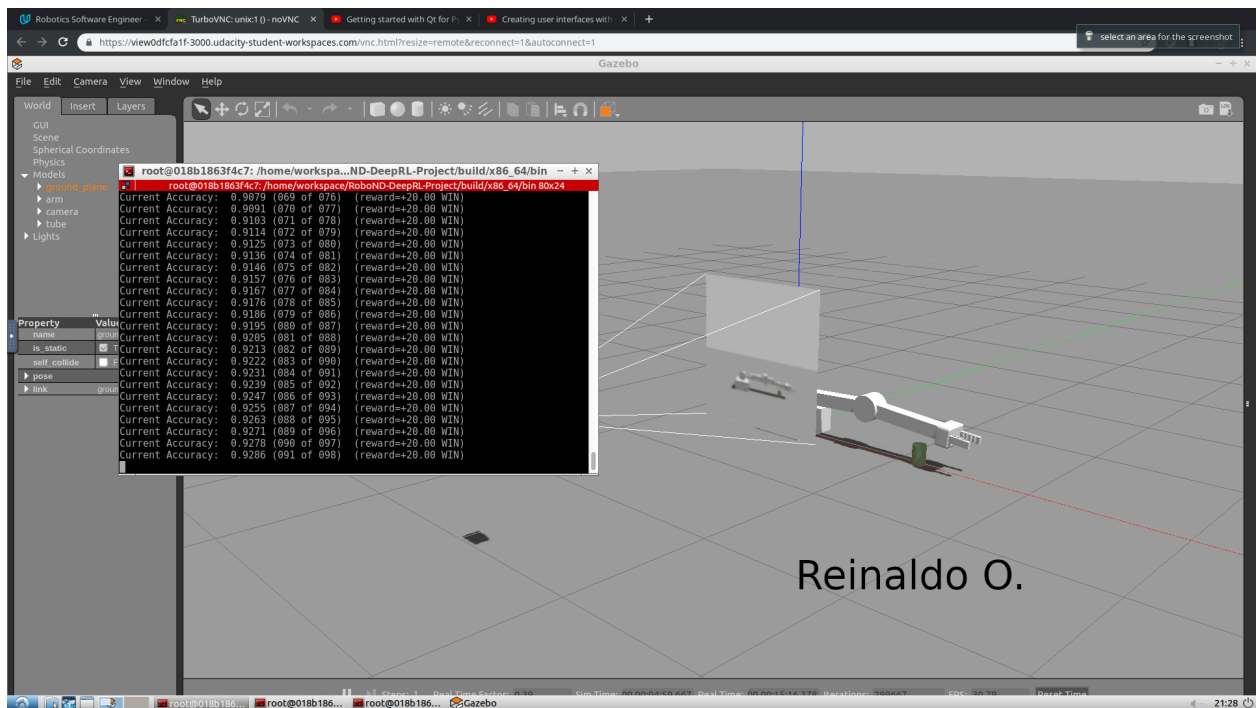


Figure 1: objective1

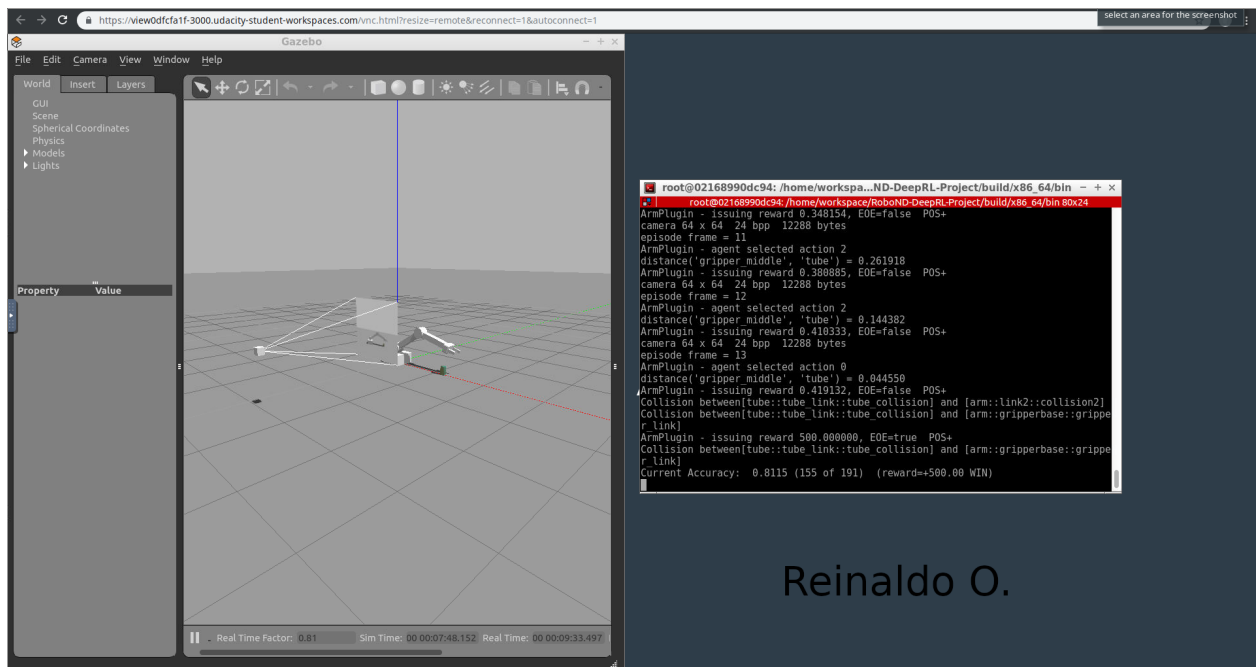


Figure 2: objective2