Deep RL Arm Manipulation

Jaeyoung Lee

Abstract—Deep RL

Index Terms—Robot, IEEEtran, Udacity, LaTeX, Localization.

1 Introduction

Deep reinforcement learning is the most dominant ML method for robot control AL these days [1]. This project shows one of the deep reinforcement learning examples with varying goals and tuning interim rewards. This document includes images of test environments and rewawrd formula and hyper-parameters table of accuracies of each test environments.

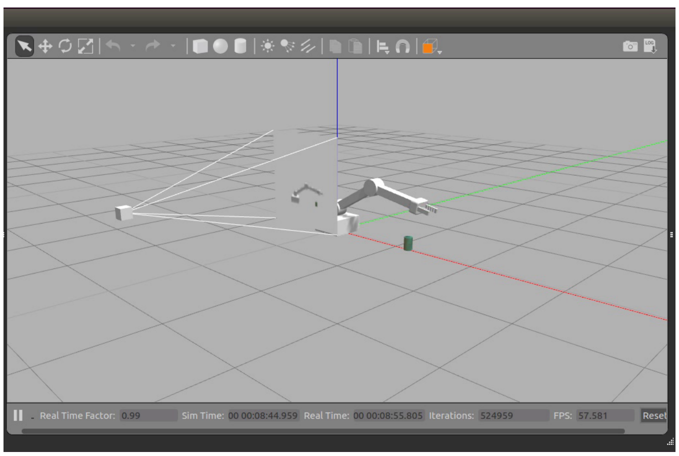


Fig. 1. Robot-Arm Touch Test Environment

2 Reward Functions

2.1 Choice of Joint Control

This test used Position-Control, not Velocity-Control. Position-Control is enough for these simple test. For the challenging test Velocity-Control might be needed.

2.2 Base Reward Design

- -1000 After 100 frames
- -1000 When touching the ground plane
- +1000 When touching the target with the gripper

2.2.1 *Robot Arm Touch Test*

- +1000 When touching the target with the robot arm

2.2.2 *Gripper-Only Touch Test*

- -1000 When touching the target with the robot arm

2.3 Interim Rewards

It was very hard to designing good interim reward policy because naive approach result in awkward joint movements and failures. After several trials and errors and survey on on-line articles and Slack threads one important strategy is not giving a positive rewards on the agent. Because the all episode should be finished after touching the target positive interim rewards can lead the agent greedy to short-sighted small rewards not chasing the final goal. Instead of it putting negative rewards only leads the reasonable joint movements and final goal.
Final interim rewards are designed as follows. avgGoalDelta means average distance difference between the gripper and the target. distGoal means distance difference between the gripper and the target.

$$\text{avgGoalDelta}_t = \text{avgGoalDelta}_{t-1} \times 0.8 + (1-0.8) \times \text{distGoal} \tag{1}$$

$$R_t = \begin{cases} -100 \times (1 - \exp(-\text{distGoal})), & \text{if avgGoalDelta} < 0.001 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$(1 - \exp(-\text{distGoal}))$ means bounded penalty from 0.0 to 1.0. The reason why the threshold 0.001 is not zero is for penalizing non-movement.

3 Hyperparameters

The hyper-parameters used in the tests are described in the following subsections.

3.1 Input Image Dimensions

Raw input images from Gazebo is 64x64(Refer to gazebo-arm.world). Larger dimension are useless for RL Networks.

- INPUT_WIDTH := 64
- INPUT_HEIGHT := 64

3.2 Optimizer

Adam Optimizer converges faster than RMSprop.

- OPTIMIZER := Adam
- LEARNING_RATE := 0.001f
- REPLAY_MEMORY := 20000

3.3 Neural Network

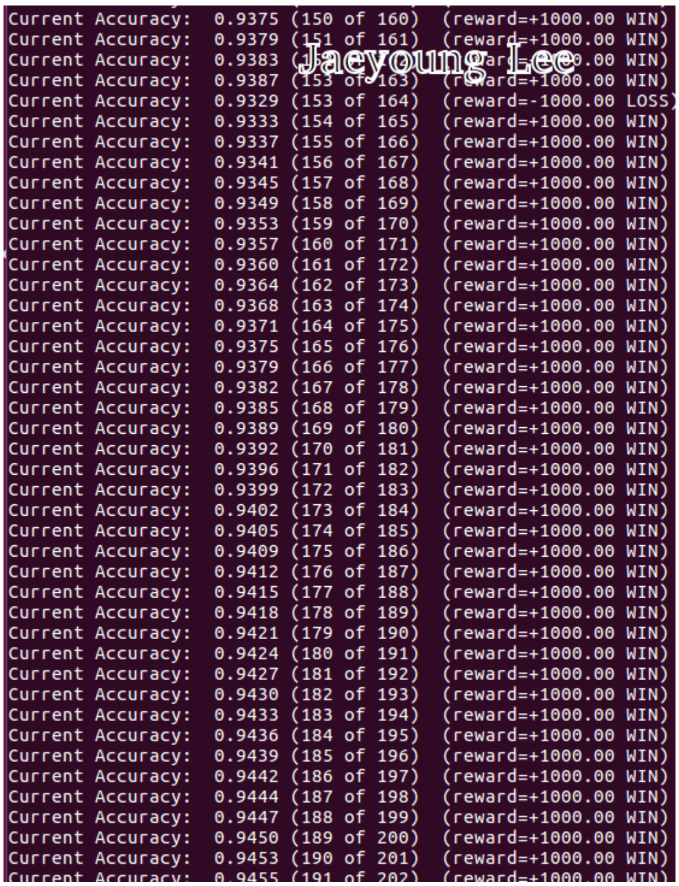
- BATCH_SIZE := 16
- USE_LSTM := true
- LSTM_SIZE := 128

4 Results

4.1 Robot Arm Touch Test

After 200 trials the system got 94% successes. This test was very easy even with more naive interim reward design (without final interim reward described in Sec 2).

The captured video clip is uploaded here: https://youtu.be/lHX6xdx9ag4



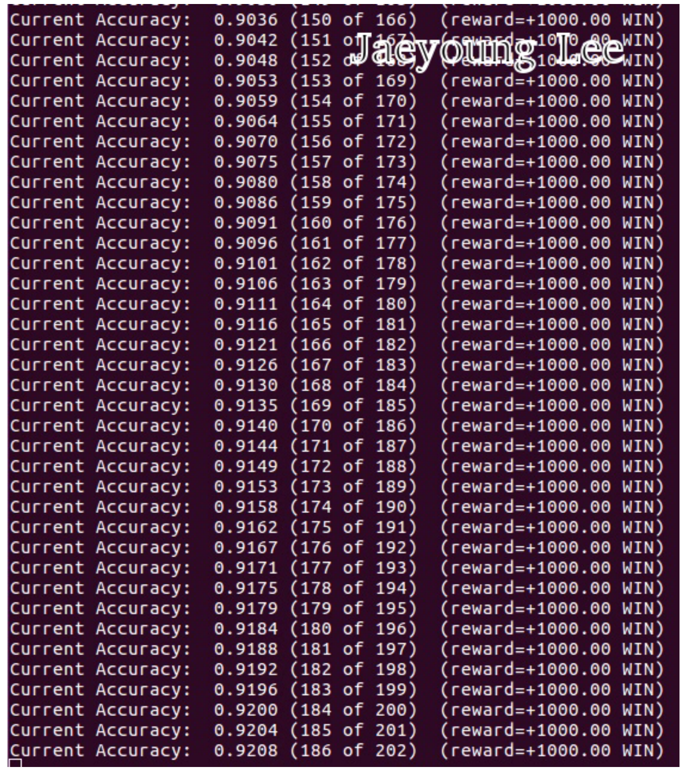
```
Current Accuracy:  0.9375 (150 of 160)  (reward=+1000.00 WIN)
Current Accuracy:  0.9379 (151 of 161)  (reward=+1000.00 WIN)
Current Accuracy:  0.9383 (                         .00 WIN)
Current Accuracy:  0.9387 (153 of 163)  (reward=+1000.00 WIN)
Current Accuracy:  0.9329 (153 of 164)  (reward=-1000.00 LOSS)
Current Accuracy:  0.9333 (154 of 165)  (reward=+1000.00 WIN)
Current Accuracy:  0.9337 (155 of 166)  (reward=+1000.00 WIN)
Current Accuracy:  0.9341 (156 of 167)  (reward=+1000.00 WIN)
Current Accuracy:  0.9345 (157 of 168)  (reward=+1000.00 WIN)
Current Accuracy:  0.9349 (158 of 169)  (reward=+1000.00 WIN)
Current Accuracy:  0.9353 (159 of 170)  (reward=+1000.00 WIN)
Current Accuracy:  0.9357 (160 of 171)  (reward=+1000.00 WIN)
Current Accuracy:  0.9360 (161 of 172)  (reward=+1000.00 WIN)
Current Accuracy:  0.9364 (162 of 173)  (reward=+1000.00 WIN)
Current Accuracy:  0.9368 (163 of 174)  (reward=+1000.00 WIN)
Current Accuracy:  0.9371 (164 of 175)  (reward=+1000.00 WIN)
Current Accuracy:  0.9375 (165 of 176)  (reward=+1000.00 WIN)
Current Accuracy:  0.9379 (166 of 177)  (reward=+1000.00 WIN)
Current Accuracy:  0.9382 (167 of 178)  (reward=+1000.00 WIN)
Current Accuracy:  0.9385 (168 of 179)  (reward=+1000.00 WIN)
Current Accuracy:  0.9389 (169 of 180)  (reward=+1000.00 WIN)
Current Accuracy:  0.9392 (170 of 181)  (reward=+1000.00 WIN)
Current Accuracy:  0.9396 (171 of 182)  (reward=+1000.00 WIN)
Current Accuracy:  0.9399 (172 of 183)  (reward=+1000.00 WIN)
Current Accuracy:  0.9402 (173 of 184)  (reward=+1000.00 WIN)
Current Accuracy:  0.9405 (174 of 185)  (reward=+1000.00 WIN)
Current Accuracy:  0.9409 (175 of 186)  (reward=+1000.00 WIN)
Current Accuracy:  0.9412 (176 of 187)  (reward=+1000.00 WIN)
Current Accuracy:  0.9415 (177 of 188)  (reward=+1000.00 WIN)
Current Accuracy:  0.9418 (178 of 189)  (reward=+1000.00 WIN)
Current Accuracy:  0.9421 (179 of 190)  (reward=+1000.00 WIN)
Current Accuracy:  0.9424 (180 of 191)  (reward=+1000.00 WIN)
Current Accuracy:  0.9427 (181 of 192)  (reward=+1000.00 WIN)
Current Accuracy:  0.9430 (182 of 193)  (reward=+1000.00 WIN)
Current Accuracy:  0.9433 (183 of 194)  (reward=+1000.00 WIN)
Current Accuracy:  0.9436 (184 of 195)  (reward=+1000.00 WIN)
Current Accuracy:  0.9439 (185 of 196)  (reward=+1000.00 WIN)
Current Accuracy:  0.9442 (186 of 197)  (reward=+1000.00 WIN)
Current Accuracy:  0.9444 (187 of 198)  (reward=+1000.00 WIN)
Current Accuracy:  0.9447 (188 of 199)  (reward=+1000.00 WIN)
Current Accuracy:  0.9450 (189 of 200)  (reward=+1000.00 WIN)
Current Accuracy:  0.9453 (190 of 201)  (reward=+1000.00 WIN)
Current Accuracy:  0.9455 (191 of 202)  (reward=+1000.00 WIN)
```


Fig. 2. Accuracy of Robot-Arm Touch Test

4.2 Gripper-Only Touch Test

After 200 trials the system got 92% successes. This test was succeeded after setting up the final interim reward policy described in Sec 2. But in some cases it failed to have 80% accuracy after 200 trials, for example when its first 10 trials all fail and when the joints vibrate too much by random noise. If the first few random trial succeed fast enough, it succeed to have 90% accuracy even before 200 trials. This kinds of tolerance looks reasonable in reinforcement learning.

The captured video clip is uploaded here: https://youtu.be/TkqwtaXrgfc



```
Current Accuracy:  0.9036 (150 of 166)  (reward=+1000.00 WIN)
Current Accuracy:  0.9042 (151 o                          WIN)
Current Accuracy:  0.9048 (152 o                          WIN)
Current Accuracy:  0.9053 (153 of 169)  (reward=+1000.00 WIN)
Current Accuracy:  0.9059 (154 of 170)  (reward=+1000.00 WIN)
Current Accuracy:  0.9064 (155 of 171)  (reward=+1000.00 WIN)
Current Accuracy:  0.9070 (156 of 172)  (reward=+1000.00 WIN)
Current Accuracy:  0.9075 (157 of 173)  (reward=+1000.00 WIN)
Current Accuracy:  0.9080 (158 of 174)  (reward=+1000.00 WIN)
Current Accuracy:  0.9086 (159 of 175)  (reward=+1000.00 WIN)
Current Accuracy:  0.9091 (160 of 176)  (reward=+1000.00 WIN)
Current Accuracy:  0.9096 (161 of 177)  (reward=+1000.00 WIN)
Current Accuracy:  0.9101 (162 of 178)  (reward=+1000.00 WIN)
Current Accuracy:  0.9106 (163 of 179)  (reward=+1000.00 WIN)
Current Accuracy:  0.9111 (164 of 180)  (reward=+1000.00 WIN)
Current Accuracy:  0.9116 (165 of 181)  (reward=+1000.00 WIN)
Current Accuracy:  0.9121 (166 of 182)  (reward=+1000.00 WIN)
Current Accuracy:  0.9126 (167 of 183)  (reward=+1000.00 WIN)
Current Accuracy:  0.9130 (168 of 184)  (reward=+1000.00 WIN)
Current Accuracy:  0.9135 (169 of 185)  (reward=+1000.00 WIN)
Current Accuracy:  0.9140 (170 of 186)  (reward=+1000.00 WIN)
Current Accuracy:  0.9144 (171 of 187)  (reward=+1000.00 WIN)
Current Accuracy:  0.9149 (172 of 188)  (reward=+1000.00 WIN)
Current Accuracy:  0.9153 (173 of 189)  (reward=+1000.00 WIN)
Current Accuracy:  0.9158 (174 of 190)  (reward=+1000.00 WIN)
Current Accuracy:  0.9162 (175 of 191)  (reward=+1000.00 WIN)
Current Accuracy:  0.9167 (176 of 192)  (reward=+1000.00 WIN)
Current Accuracy:  0.9171 (177 of 193)  (reward=+1000.00 WIN)
Current Accuracy:  0.9175 (178 of 194)  (reward=+1000.00 WIN)
Current Accuracy:  0.9179 (179 of 195)  (reward=+1000.00 WIN)
Current Accuracy:  0.9184 (180 of 196)  (reward=+1000.00 WIN)
Current Accuracy:  0.9188 (181 of 197)  (reward=+1000.00 WIN)
Current Accuracy:  0.9192 (182 of 198)  (reward=+1000.00 WIN)
Current Accuracy:  0.9196 (183 of 199)  (reward=+1000.00 WIN)
Current Accuracy:  0.9200 (184 of 200)  (reward=+1000.00 WIN)
Current Accuracy:  0.9204 (185 of 201)  (reward=+1000.00 WIN)
Current Accuracy:  0.9208 (186 of 202)  (reward=+1000.00 WIN)
```


Fig. 3. Accuracy of Gripper-Only Test

5 Future Work

All tests are done using interim rewards based on the distance between the gripper and the target. But it is very hard to get this distance information in the real environment. Even though it is succeeded using various sensors, the sensor data include noise. RL without the interim reward or simulating more realistic sensor data will be challenging as a future work. Student should discuss on what approaches they could take to improve their results.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.