

Project: Deep RL Arm Manipulation

Vladimir Kobal

October, 2018

1 Reward Function

The average of the delta of the distance to the goal is smoothed with the basic exponential smoothing: $\text{avgGoalDelta} = (\text{avgGoalDelta} * \text{ALPHA}) + (\text{distDelta} * (1 - \text{ALPHA}))$, where $\text{ALPHA} = 0.3$. Joints are controlled by setting positions.

1.1 First task: the arm touches the object

For the first task, reward function encourages the robot arm to move towards the goal. For ground collisions and episode timeouts it takes into account the distance to the goal at the last iteration. To encourage the agent speeding up the winning sequence the number of episode frames is discounted from the reward. In order to slow down the arm at the end of the winning sequence the last avgGoalDelta value is also discounted from the reward.

```
REWARD_WIN = 10
REWARD_LOSS = -30
```

Event	Reward
Collision between the arm and the object	$\text{REWARD_WIN} + \text{maxEpisodeLength} - \text{episodeFrames} - 1000 * \text{avgGoalDelta}$
Robot hit the ground	$\text{REWARD_LOSS} - 10 * \text{lastGoalDistance}$
Episode timeout	$\text{REWARD_LOSS} - 10 * \text{lastGoalDistance}$
Interim reward	$10 * \text{avgGoalDelta}$

Table 1: Rewards for the first task

1.2 Second task: the gripper middle touches the object

For the second task with the gripper middle touching the object, the reward function coefficients were changed. avgGoalDelta was excluded from the winning reward as far as the robot arm didn't tend to speed up too much. Square root function was applied to lastGoalDistance to make the influence of the distance to the goal more prominent when the arm is closer to the goal.

```
REWARD_WIN = 10
REWARD_LOSS = -1 -100 * sqrt(lastGoalDistance)
```

Event	Reward
Collision between the gripper and the object	$\text{REWARD_WIN} + \text{maxEpisodeLength} - \text{episodeFrames}$
Collision between the arm and the object	REWARD_LOSS
Robot hit the ground	$\text{REWARD_LOSS} - 10$
Episode timeout	REWARD_LOSS
Interim reward	$80 * \text{avgGoalDelta}$

Table 2: Rewards for the second task (gripper middle)

1.3 Second task: the gripper base touches the object

For the second task with the gripper base touching the object, the reward function was drastically simplified, and the range of its values was decreased.

REWARD_WIN = 0.5

REWARD_LOSS = -lastGoalDistance

Event	Reward
Collision between the gripper and the object	REWARD_WIN
Collision between the arm and the object	REWARD_LOSS
Robot hit the ground	REWARD_LOSS
Episode timeout	REWARD_LOSS
Interim reward	avgGoalDelta

Table 3: Rewards for the second task (gripper base)

2 Hyperparameters

Input width and height were decreased down to 64x64 (the size of images, taken by the camera). The number of actions was set to the number of joints multiplied by 2, as every joint can move in two directions. With small replay memory size, the agent can't remember sequences of motions, and large values didn't lead to better results, so this value was set to the default one. RMSprop optimizer does the job for all three cases.

2.1 First task: the arm touches the object

The task is simple, so we don't need exploration at all (ALLOW_RANDOM is set to "false"). LSTM and batch size were increased. When LSTM size is big, the agent tends to refine the winning sequence and remember it well. Experiments with Gamma and learning rate showed that these parameters do not play a crucial role in the agent behavior, so moderate values were chosen.

Hyperparameter	Value
INPUT_CHANNELS	3
ALLOW_RANDOM	false
GAMMA	0.7
EPS_START	-
EPS_END	-
EPS_DECAY	-
INPUT_WIDTH	64
INPUT_HEIGHT	64
NUM.ACTIONS	DOF * 2
OPTIMIZER	RMSprop
LEARNING_RATE	0.01
REPLAY_MEMORY	10000
BATCH_SIZE	32
USE_LSTM	true
LSTM_SIZE	128

Table 4: Hyperparameters for the first task

2.2 Second task: the gripper middle touches the object

For the second task with the gripper middle touching the object, exploration was enabled, and Epsilon end value was decreased to prevent agent exploring an action space after the winning sequence was found. Gamma was decreased in order to pay more attention to the most immediate reward. With slightly increased learning rate agent was giving more consistent results.

Hyperparameter	Value
INPUT_CHANNELS	3
ALLOW_RANDOM	true
GAMMA	0.4
EPS_START	0.9
EPS_END	0.01
EPS_DECAY	200
INPUT_WIDTH	64
INPUT_HEIGHT	64
NUM_ACTIONS	DOF * 2
OPTIMIZER	RMSprop
LEARNING_RATE	0.05
REPLAY_MEMORY	10000
BATCH_SIZE	32
USE_LSTM	true
LSTM_SIZE	128

Table 5: Hyperparameters for the second task (gripper middle)

2.3 Second task: the gripper base touches the object

For the second task with the gripper base touching the object, exploration was radically prolonged, but the probability of random actions was decreased. With LSTM disabled the successful task resolution became much more reproducible. When LSTM was enabled, the arm movements was quicker and smoother, but the agent was tending to repeat non-optimal action sequences.

Hyperparameter	Value
INPUT_CHANNELS	3
ALLOW_RANDOM	true
GAMMA	0.9
EPS_START	0.5
EPS_END	0.0001
EPS_DECAY	1000
INPUT_WIDTH	64
INPUT_HEIGHT	64
NUM.ACTIONS	DOF * 2
OPTIMIZER	RMSprop
LEARNING_RATE	0.01
REPLAY_MEMORY	10000
BATCH_SIZE	256
USE_LSTM	false
LSTM_SIZE	-

Table 6: Hyperparameters for the second task (gripper base)

3 Results

For both objectives, the DQN agent’s performance was quite good. The expected accuracy was achieved in less than 100 steps (Fig. 1, 3). The main problem was solving the second task with enabled LSTM. Not only the expected accuracy was achieved in more than 100 steps (Fig. 2), but successful runs were not steadily reproducible.

4 Future work

Further improvements could be made to the winning sequence speed and to the agent’s ability to reproduce successful runs. It can be achieved by fine-tuning reward function and parameters. It also worth trying to use velocity based control, as it can lead to quicker and more precise movements of the robot arm. Though, with such control, it will be more difficult for an agent to find a solution.

```
root@abc12e6301a7: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin
root@abc12e6301a7: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin 95x54
Current Accuracy: 0.9355 (087 of 093) (reward=+36.25 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0600887
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0600887
Current Accuracy: 0.9362 (088 of 094) (reward=+37.91 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0600105
Current Accuracy: 0.9368 (089 of 095) (reward=+37.99 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0603973
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0603973
Current Accuracy: 0.9375 (090 of 096) (reward=+37.60 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0603126
Current Accuracy: 0.9381 (091 of 097) (reward=+37.69 WIN)
GROUND CONTACT, EOE
delta = 0.0881348
Current Accuracy: 0.9286 (091 of 098) (reward=-31.81 LOSS)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0614016
Current Accuracy: 0.9293 (092 of 099) (reward=+36.60 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0602627
Current Accuracy: 0.9300 (093 of 100) (reward=+37.74 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0601822
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0601822
Current Accuracy: 0.9307 (094 of 101) (reward=+37.82 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0601007
Current Accuracy: 0.9314 (095 of 102) (reward=+37.90 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0604917
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0604917
Current Accuracy: 0.9320 (096 of 103) (reward=+37.51 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0604065
Current Accuracy: 0.9327 (097 of 104) (reward=+37.59 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0603239
Current Accuracy: 0.9333 (098 of 105) (reward=+37.68 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.060727
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.060727
Current Accuracy: 0.9340 (099 of 106) (reward=+37.27 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::link2::collision2]
delta = 0.0606388
Current Accuracy: 0.9346 (100 of 107) (reward=+37.36 WIN)
```

Figure 1: The accuracy of the arm touching the object

```
root@3ca4ade911be: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin - + x
root@3ca4ade911be: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin 90x54
Current Accuracy: 0.7889 (142 of 180) (reward=+66.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0502089
Current Accuracy: 0.7901 (143 of 181) (reward=+68.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0696199
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0696199
Current Accuracy: 0.7912 (144 of 182) (reward=+68.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.046841
Current Accuracy: 0.7923 (145 of 183) (reward=+67.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0740039
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0740039
Current Accuracy: 0.7935 (146 of 184) (reward=+70.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0736603
Current Accuracy: 0.7946 (147 of 185) (reward=+68.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0713922
Current Accuracy: 0.7957 (148 of 186) (reward=+69.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0563618
Current Accuracy: 0.7968 (149 of 187) (reward=+70.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0706116
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0706116
Current Accuracy: 0.7979 (150 of 188) (reward=+63.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0697944
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0697944
Current Accuracy: 0.7989 (151 of 189) (reward=+68.00 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripper_middle::middle_collis
ion]
delta = 0.0561739
Current Accuracy: 0.8000 (152 of 190) (reward=+68.00 WIN)
```

Figure 2: The accuracy of the gripper middle touching the object


```
root@83e11505e0e0: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin - + x
root@83e11505e0e0: /home/workspace/RoboND-DeepRL-Project/build/x86_64/bin 84x54
delta = 0.0616135
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0616135
Current Accuracy: 0.8679 (092 of 106) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0652513
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0652513
Current Accuracy: 0.8692 (093 of 107) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.104403
Current Accuracy: 0.8704 (094 of 108) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0229305
Current Accuracy: 0.8716 (095 of 109) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0247491
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0247491
Current Accuracy: 0.8727 (096 of 110) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0615465
Current Accuracy: 0.8739 (097 of 111) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0654031
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0654031
Current Accuracy: 0.8750 (098 of 112) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.0761451
Current Accuracy: 0.8761 (099 of 113) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.065438
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.065438
Current Accuracy: 0.8772 (100 of 114) (reward==+0.50 WIN)
Collision between[tube::tube_link::tube_collision] and [arm::gripperbase::gripper_li
nk]
delta = 0.119704
Current Accuracy: 0.8783 (101 of 115) (reward==+0.50 WIN)
```

Figure 3: The accuracy of the gripper base touching the object