Reinforcement Learning in Robotics

Boyuan Chen
PhD student, Computer Science
Creative Machines Lab
http://www.cs.columbia.edu/~bchen/
A great time for RL ...

Classic Control:
A great time for RL ...

Atari Games:
A great time for RL ...

Go:
A great time for RL ...
Outline

- How to set up RL problems in Robotics?
- Important topics
- How to get started?
RL in Robotics

- No labelled data;
- No access to real model;
- No fixed rule;
- Continuous space;
- Complex transition dynamics.
Problem Setting

- MDP process defined by: $s, a, p, r, \rho_0, \gamma, T$
- Policy: $\pi_\theta : s \times a \rightarrow \mathbb{R}_{\geq 0}$
- Expected Reward: $\mathbb{E}_\tau \left[ \sum_{t=0}^{T} \gamma^t r(s_t, a_t) \right]$
- Trajectories: $\tau = (s_0, a_0, \ldots)$
Real robot?
Topics: Manipulation

Topics: Manipulation

Contribution:

- DeepRL
  - Real complex robot system
  - Complex task
  - Asynchronous data collection
- Safety Exploration
Topics: Manipulation

Off-policy Deep Q-function based algorithms:

\[ Q^{\pi_n}(x_t, u_t) = \mathbb{E}_{r_i \geq t, x_i \sim E, u_i \sim \pi_n} \left[ R_t | x_t, u_t \right] \]

\[ \mu_{n+1}(x_t) = \arg \max_u Q^{\pi_n}(x_t, u_t) \]

- DDPG (Deep Deterministic Policy Gradient)  
- NAF (Normalized Advantage Function)  

\[ Q(x, u | \theta^Q) = A(x, u | \theta^A) + V(x | \theta^V) \]

\[ A(x, u | \theta^A) = -\frac{1}{2} (u - \mu(x | \theta^\mu))^T P(x | \theta^P) (u - \mu(x | \theta^\mu)) \]
Topics: Manipulation

- State Representation

<table>
<thead>
<tr>
<th>Joint Angles</th>
<th>End-effector Positions</th>
<th>Time Derivatives</th>
<th>Target Position to the State</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance from target location</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Handle position when door is closed + Quaternion of the door</td>
</tr>
</tbody>
</table>
Topics: Manipulation

- Asynchronous NAF
Topics: Manipulation

- Safety Constraints
  - Joint Position Limits:
    - Maximum commanded velocity allowed per joint
    - Strict position limits for each joint
  - Bounding sphere for end-effector position

Very important for training from scratch on real systems!
Topics: Manipulation
Topics: Manipulation

What have we learned from this?

- Efficiency is important for real robots
- Safety is important for real robots
- It is possible to apply DeepRL on real robots to accomplish complex tasks
Complex Task?
Topics: Locomotion

Topics: Locomotion

Topics: Locomotion

- Highlight:
  - Less prior knowledge
  - Hierarchical RL

![Diagram of FSM and QP]

\[
\begin{align*}
\text{FSM} & \\
\min_{\mathbf{q}_r,F} \left( \frac{1}{2} \mathbf{q}_r - \mathbf{g}_3(\mathbf{q}_r, F) \right)^2 \quad \text{subject to} \quad \mathbf{J}(\mathbf{q}_r)\mathbf{q}_r = \mathbf{S}(x_{x-1}, F, \mathbf{F})
\end{align*}
\]
Topics: Locomotion

Topics: Locomotion

Topics: Locomotion

LLC (Low Level Controller):

- State
- Action
- Goal
- Reward

\[ r_L = w_{\text{pose}} r_{\text{pose}} + w_{\text{vel}} r_{\text{vel}} + w_{\text{root}} r_{\text{root}} + w_{\text{com}} r_{\text{com}} + w_{\text{end}} r_{\text{end}} + w_{\text{heading}} r_{\text{heading}} \]
Topics: Locomotion

HLC (High Level Controller):
Topics: Locomotion

HLC (High Level Controller):

- State
- Training
Topics: Locomotion
Topics: Locomotion

What have we learned from this?

- Identify hierarchical structure is important.
- State representation for different hierarchies are important.

Can we do better?

- Current topic: identify the internal hierarchical structure automatically.
Are we ready?
Topics: Sim2Real

Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping: Bousmalis et al, 2017
Topics: Sim2Real

Why this is important?

- Grasping:

  Novel Object → Learning based algorithms → Hungry for labeled dataset → Model-based → Generate Synthetic Exp
Topics: Sim2Real

Approach
How to get started?

Algorithm

Environment
Algorithms

- Libraries:
  - Baselines (OpenAI): [https://github.com/openai/baselines](https://github.com/openai/baselines)
  - Rllab (OpenAI): [https://github.com/rll/rllab](https://github.com/rll/rllab)
  - ...

- Implement your own:
  - Go back to the original paper
  - Use open-source code as reference
  - Start testing from toy examples
Algorithms

How to select an algorithm for your problem?

- Action space: continuous? Discrete?

More often in Robotics

- Discrete
- Continuous

- Theta
- Velocity
Algorithms

How to select an algorithm for your problem?

- Reduce your problem to toy example
Algorithms

How to select an algorithm for your problem?

- Find similar task in “standard” problems

PR2 Robot (Huge robot, dual arm)  Fetch simulation in gym
Environment

- Hardware robot
- Simulation
Environment

- Hardware robot
- Simulation
What is the problem of starting from real robot?

- Expensive
- Safety
Simulator

● Simulation Environment
  ○ OpenAI Gym
  ○ MuJoCo-py
  ○ PyBullet
  ○ Gazebo
  ○ V-rep
  ○ Roboschool
  ○ Dart
  ○ ....

● Dynamics Engine
  ○ Box2D
  ○ Bullet
  ○ ODE
  ○ ...

...
RL in Robotics: problems

- Reward?
- Structure?
- Exploration?
- Stability?
Future Directions

- Efficient RL
- Long Horizon Reasoning
- Hierarchical RL
- Meta-RL
- Reward Function
- Multi-model
- Lifelong Learning
- Simulation to Real
- ...
References

[1] https://gym.openai.com/


References


Thank you!