Reinforcement Learning-Based Task Training: Simulation Training of Robotic Arm Gripping and Stacking Cube Tasks

Xi Lin, Dalian University of Technology
Mentor: Professor Hao Su
Contact: 20211044031@mail.dlut.edu.cn

Objectives and Challenges

- Basic Knowledge: Controlment and Reinforcement Learning

Understanding the theory of automatic control of mechanical devices and learning about reinforcement learning

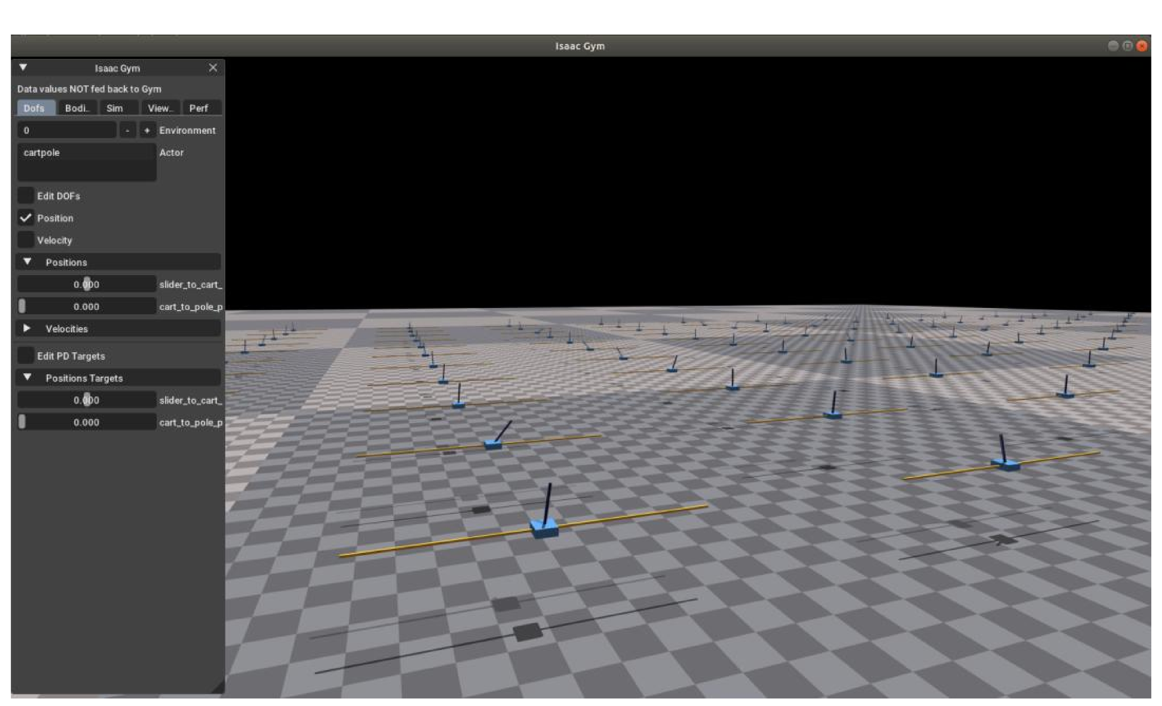
- Experiment: Building Environment and Simulating Training

Reinforcement learning is applied to the training and expansion of robotic arm functions by setting up a reinforcement learning experiment environment and importing the parameter model of the robotic arm to be controlled.

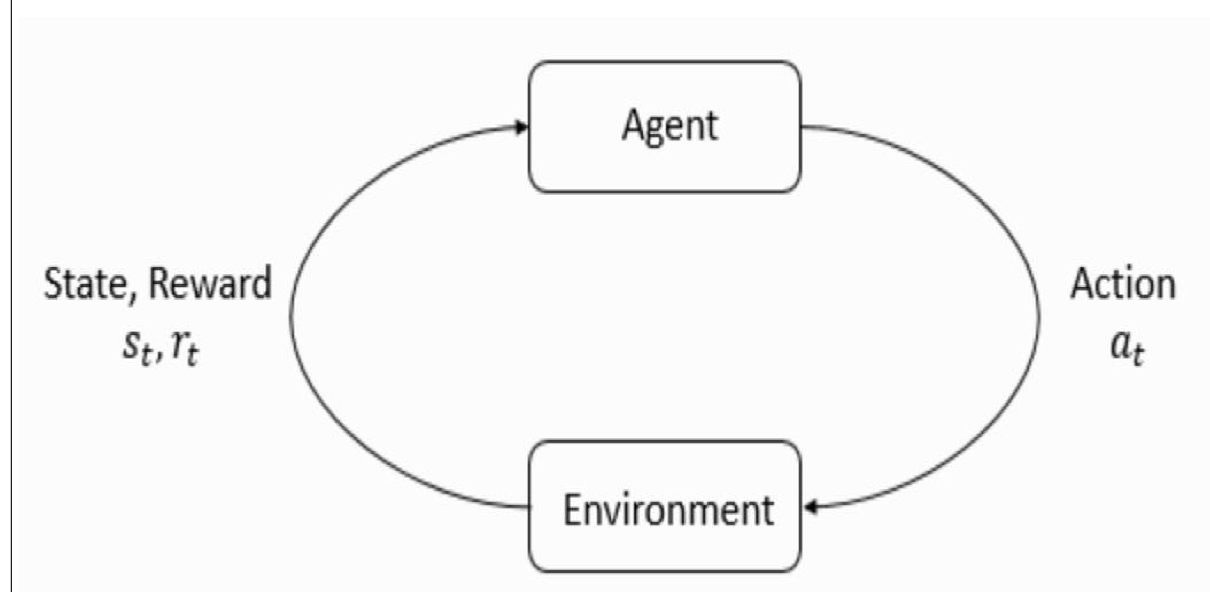
Training Environment: Isaac Gym

NVIDIA's Reinforcement Learning Environment, IsaacGym, offers a robust platform for developing and testing advanced robotic control algorithms. This comprehensive introduction outlines the key features and components of IsaacGym, highlighting its significance in the field of reinforcement learning for robotics:

- Simulation Capabilities:
- Realism and Immersion:
- Customizability:
- Robotics Toolkit Integration
- Multi-Agent Simulations
- Sensor Realism:
- Diverse Environments:
- Benchmarking and Evaluation:
- Community and Support



Background: Reinforcement Learning



Reinforcement Learning Introduction

Reinforcement Learning (RL) is a field in machine learning that focuses on learning how to take actions to maximize rewards without being explicitly told which actions to take. It involves an agent interacting with an environment, aiming to learn the best strategy to achieve its goals through trial and error.

- Core Concepts

The main components of RL are the agent and the environment. The agent observes the environment, takes actions, and receives rewards as feedback. Its objective is to maximize cumulative rewards over time.

- Policy

The agent's strategy, called a policy, determines which actions to take in different states. Policies can be deterministic or stochastic.

- Exploration and Exploitation

RL faces the exploration-exploitation dilemma, where the agent must balance trying new actions for potential rewards (exploration) and exploiting known actions for immediate rewards (exploitation).

- Reward

The reward function provides feedback to the agent about the goodness or badness of its actions in a given state.

Additional Introduction

- Markov Decision Process (MDP)

MDP is a theoretical model used to describe multi-step RL tasks. It includes states, actions, transition probabilities, and reward functions.

- Classification of Reinforcement Learning

RL can be classified as model-based or model-free learning. Model-based learning involves knowing the environment's model, while model-free learning does not rely on such knowledge.

Experiment: Environment and Training

1. Building Environment

- Ubuntu / Nvidia Driver

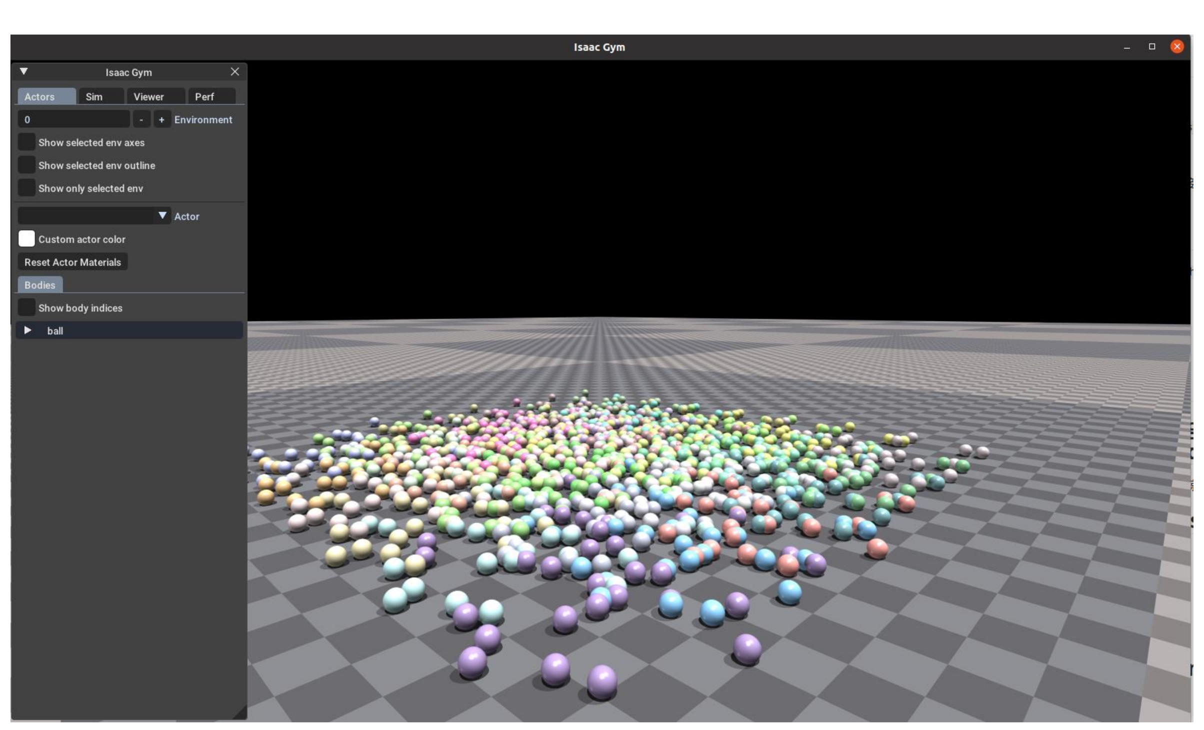
Since the whole experiment is deployed in Ubuntu environment and the computing power is supported by Nvidia GPUs, we need to find and install the drivers for different models of graphic cards and Ubuntu versions on the official website.

- Anaconda / PyTorch / CUDA

Since reinforcement learning is a branch of machine learning, and today's mainstream machine learning training environments simulate mirrored environments for training, install and configure the environment as above.

- Issac Gym / Legged Gym and Environment Testing

There are some examples under the official self-installation package. Among them, the small ball falling experiment (1080_balls of solitude) is selected for environment installation testing, and the simulation results can be obtained as follows, and can prove that the environment is built successfully.



2. Experimental Model Import

- Cube Stack Task

Code: python train.py task=FrankaCubeStack [options]

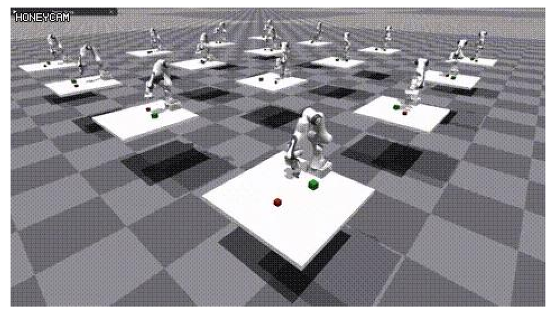
This is a task to stack boxes using the panda arm. 7-axis arm joint motions are learned step by step. The default setting is 10,000 epochs, but the arm motion can be learned in about 1,000 epochs.

Results: Robotic Arms Complete Task

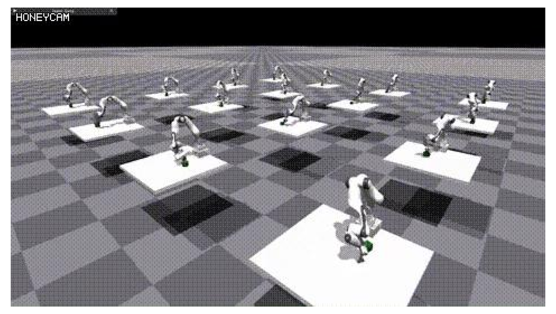
The action space consists of 7 dimensions of the arm joints, while the observation space has a total of 26 dimensions. After a step-by-step learning of the 7 degrees of freedom, the training was finalized and achieved the expected results.

- 7 dimensions for the position and orientation of the boxes being moved
- 3 dimensions for the vector from the stacked box to the moved box
- 7 dimensions for the gripper's grasping position and orientation
- 9 dimensions for arm joints and gripper fingers

- Before Training:



- After Training:



References

[1] Rudin, Nikita, David Hoeller, Marko Bjelonic, and Marco Hutter. "Advanced Skills by Learning Locomotion and Local Navigation End-to-End."
In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2497-2503. IEEE, 2022.

[2] Rudin, Nikita, David Hoeller, Philipp Reist, and Marco Hutter. "Learning to walk in minutes using massively parallel deep reinforcement learning."
In Conference on Robot Learning, pp. 91-100. PMLR, 2022.

[3] Issac Gym:
https://developer.nvidia.com/isaac-gym