Usefulness of using Nvidia IsaacSim and IsaacGym for AI robot manipulation training

Youngsung Son
Digital Convergence Research Laboratory
Electronics and Telecommunications Research Institute
Daejeon, Korea
ysson@etri.re.kr

Hyonyong Han
Digital Convergence Research Laboratory
Electronics and Telecommunications Research Institute
Daejeon, Korea
hyoyoung.han@etri.re.kr

Joonmyun Cho
Digital Convergence Research Laboratory
Electronics and Telecommunications Research Institute
Daejeon, Korea
jmcho@etri.re.kr

***Abstract*—AI technology is increasingly being used to learn robotic behavior. Various types of learning techniques are being applied to learn optimal robotics control or autonomous driving of driverless cars, and attempts are being made to use reinforcement learning techniques to learn robot behavior. When building data for robot behavior learning is not practical in terms of cost and time, imitation learning and reinforcement learning become realistic options for behavior learning using simulators. In this paper, we introduce a robot learning method using Nvidia's IsaacSim and IsaacGym, and examine the performance result required for learning to test its feasibility.**

Keywords*—*robot control, reinforcement learning, simulation, deep learning

I. Introduction

The rapid development of robot manipulation has been investigated in both academia and industry. Advanced learning models, such as reinforcement learning, have been used to solve complex control and movement learning problems[1]. These self-taught learning architectures typically use deep learning models to recognize their surroundings and accumulate knowledge. Deep reinforcement learning has been shown to outperform experts in many robot manipulation domains[2]. It can also be used to train robots in an end-to-end manner[3]. However, directly applying reinforcement learning algorithms to solve complex tasks in the real world can be inefficient, especially in robotic environments where interactions are long and costly. A common solution is to use simulated virtual environments to reconstruct the robot's dynamics and external variations, which can be used to replace time-consuming real-world experiments. In this paper, we are evaluating Nvidia's IsaacSim and IsaacGym for this purpose.

II. Simulation Environments for Robot Learning

In the early days of research on deep reinforcement learning techniques that combine reinforcement learning with deep learning, the classic control environment included in GymToolkit, Box2D, and 2D/3D robot control environments based on the Mujoco physics engine were used as benchmarks in many papers[3,9,10,11]. However, these existing benchmarks have become less and less popular due to the difficulty of topology and geometric deformation of objects in the virtual environment, the lack of a realistic dynamic environment model, and the lack of support for generating various types of manipulation tasks. Recently, new virtual environment benchmarks have been published that support a variety of real-world robotic agents, support for customization of environmental input and output parameters and object properties, support for different object models, provide task episodes that require more complex and sophisticated behavioral manipulation, and support authoring capabilities[4,5].

Table 1 compares nine recently published virtual environment benchmarks for behavioral model evaluation of robot manipulators [6,7,8,9].

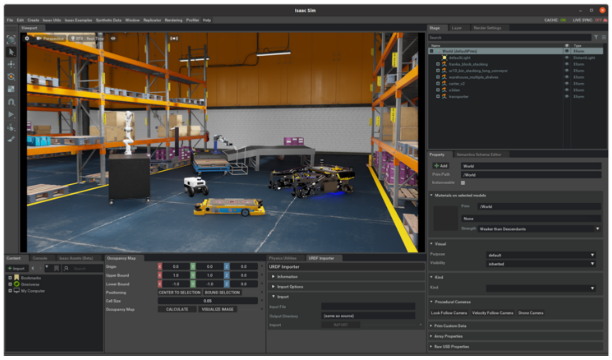
<Table.1> Comparisons among simulation environments

Benchmark	IsaacGym	RLBench	Robosuito
Organization	NVIDIA	Imperial College	Texas Univ. Stanford
Grasp	Physical	Abstract	Physical
MultiControl	No	Yes	Yes
#Objects	-	28	10
Ray tracing	-	-	NVISII
Randomization	Yes	Yes	Yes
Rigid body	PhysX5	CoppelliaSim	Mujoco

III. IsaacSim and IsaacGym

A. IsaacSim

NVIDIA Omniverse IsaacSim is a robotics simulation toolkit that enables researchers and practitioners to create robust, physically accurate simulations and synthetic datasets for navigation and manipulation applications. It supports ROS/ROS2 and simulates sensor data from sensors such as RGB-D, Lidar, and IMU for various computer vision techniques[6].



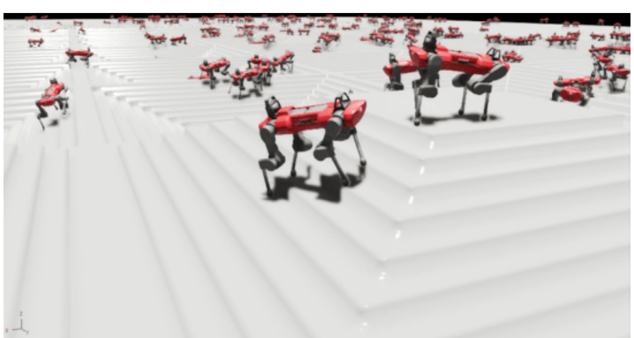
<Fig.1> Nvidia Omniverse IsaacSim

B. *IsaacGym*

NVIDIA IsaacGym is a GPU-accelerated reinforcement learning (RL) framework that can be used to train AI robots for manipulation tasks. It is built on top of NVIDIA Isaac Sim, which is a robotics simulation toolkit that provides a realistic and accurate environment for training robots[6].

IsaacGym offers a number of features that make it well-suited for AI robot manipulation training:

- GPU acceleration: IsaacGym is GPU-accelerated, which can significantly speed up the training process.
- Realistic physics: Isaac Sim uses realistic physics to simulate the environment, which helps to ensure that the robot learns to perform tasks in a way that is consistent with the real world.
- Modular design: IsaacGym is modular, which makes it easy to customize and extend to new applications.
- Support for ROS/ROS2: Isaac Gym supports ROS/ROS2, which makes it easy to integrate with other robotics software.
- Open source: IsaacGym is open source, which means that it is free to use and modify.



<Fig.2> Nvidia Omniverse IsaacGym

C. *Difference between IsaacSim and IsaacGym*

IsaacSim and IsaacGym are both robotics simulation tools developed by NVIDIA. However, they have different strengths and weaknesses. Isaac Sim is a general-purpose robotics simulation toolkit that can be used for a variety of tasks, including navigation, manipulation, and perception. It offers a wide range of features, including realistic physics, support for ROS/ROS2, and a variety of sensor models. However, Isaac Sim can be complex to use, and it can be difficult to customize for specific tasks. IsaacGym is a reinforcement learning (RL) framework that is built on top of Isaac Sim. It provides a simple and easy-to-use interface for training RL agents, and it supports a variety of RL algorithms. However, IsaacGym is not as feature-rich as Isaac Sim, and it is not as well-suited for tasks that do not involve RL.

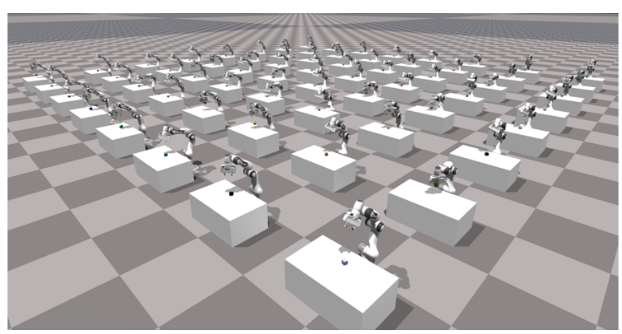
Here is a table that summarizes the key differences between Isaac Sim and Isaac Gym:

<Table.2> Comparisons among IsaacSim and IsaacGym

Features	IsaacSim	IsaacGym
General-purpose	Yes	No
Tasks	Navigation, manipulation, perception	Reinforcement learning
Features	Realistic physics, support for ROS/ROS2, variety of sensor models	Simple and easy-to-use interface for training RL agents, supports a variety of RL algorithms
Complexity	Complex	Simple
Customizability	Difficult to customize	Easy to customize
Ease of Use	Difficult to use	Easy to Use
Suitability for special tasks	Good for a variety of tasks	Good for tasks that involve RL

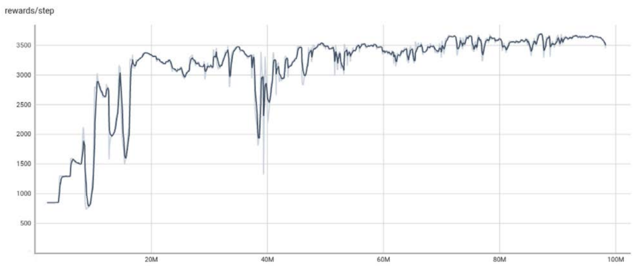
IV. ISAACSIM AND ISAACGYM EVALUATION

We built a simulation environment for learning pick and place behavior. We configured multi-Franka robots in Nvidia IsaacSim and IsaacGym and checked the learning performance by varying the batch size. The refinement learning algorithm used AAC to check the variation of reward and episode size with the number of Franka robots running simultaneous training. The learning results show that both Nvidia IsaacSim and IsaacGym perform similarly for learning robot control policies.

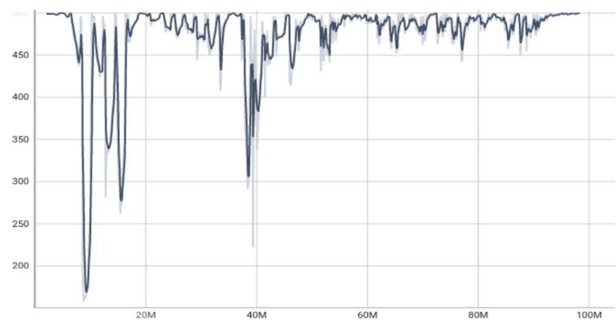


<Fig.3> Reinforcement Learning in IsaacSim and IsaacGym

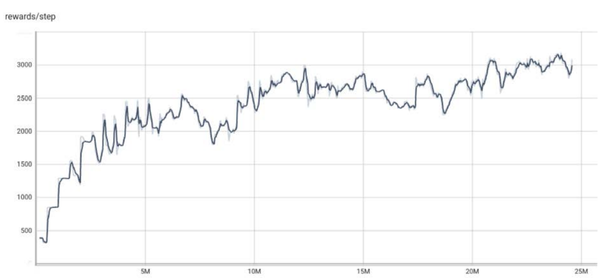
We didn't find any performance differences between IsaacSim and IsaacGym. Based on the test environment parameters, the performance and efficiency of reinforcement learning was as expected.



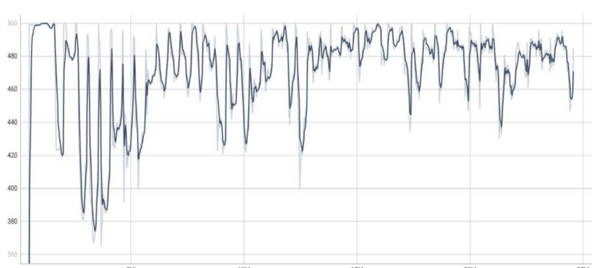
<Fig.4> Rewards with Steps, 4096 Robots(Best 3096)



<Fig.5> Episode lengths with Steps, 4096 Robots



<Fig.6> Rewards with Steps, 1024 Robots(Best 3160)



<Fig.7> Episode lengths with Steps, 1024 Robots

LIMITATIONS

Reinforcement learning has great potential for teaching robotics how to behave like humans. It can save time, money, and computing resources compared to other techniques that rely on trial-and-error optimization, and it provides clues that can lead to the acquisition of sophisticated manipulation skills. Nonetheless, there are still many challenges to be addressed in strong learning.

There are four challenges of reinforcement learning with simulation environment.

1. expert demonstration quality problem: If the quality of the demonstration is poor or does not capture important elements of the task processing, the learning algorithm may not be able to extract useful information.
2. The distributional shift problem: This can occur when expert demonstrations are collected in different environments or conditions than the agent.
3. The curse of dimensionality: This can occur when dealing with high-dimensional data with a very large number of variables.
4. the generalization problem: the problem of generalization, which is the ability to perform well on unexperienced conditions or untrained data.

CONCLUSION

Robot learning simulation technology is an excellent approach to learn behavioral policies by studying robots' control behaviors in detail under human expert instruction, and has shown great promise in various applications such as robotics and autonomous driving, manufacturing and healthcare, and finance. On the other hand, there are still several challenges that need to be overcome in robotic action learning, such as expert demonstration data quality issues, distributional equalization issues, dimensionality curse, generalization issues, and scalability.

To overcome these challenges, we used Nvidia IsaacSim and IsaacGym to build an environment with various conditions and showed that it is possible to build an environment that can learn complex and sophisticated tasks independently. As such, it is expected that the use of simulation environments will be represented in the progress of reinforcement learning for robot control through simulation as an important research theme in the field of artificial intelligence.

ACKNOWLEDGMENT

We thank Seonghyuk Jang, Joohwan Park for their helpful assistance. This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. [23ZR1100, A Study of Hyper-Connected Thinking Internet Technology by autonomous connecting, controlling and evolving ways] and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-0087, Development of an edge brain framework to make manufacturing equipment and robots intelligent).

REFERENCES

[1] R. Sutton and A. Barto, "Reinforcement Learning: An Introduction," MIT Press Cambridge, Second Ed., 2018.

[2] T. Osa, et al., "An Algorithmic Perspective on Imitation Learning," arXiv preprint, abs/1811.06711, 2018.

[3] X. Peng et al., "DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills," arXiv preprint, arXiv:1804.02717, 2018.

[4] J. Arthur et al., "Unity: A General Platform for Intelligent Agents," arXiv preprint, arxiv:1809.02627, 2018.

[5] N. G. Lopez et al., "Gym-Gazebo2, a Toolkit for Reinforcement Learning Using ROS 2 and Gazebo," arXiv preprint, arxiv:1903.06278, 2019

[6] V. Makoviychuk et al., "IsaacGym: High performance gpu-based physics simulation for robot learning," arXiv preprint, arXiv:2108.10470, 2021.

[7] S. James et al., "RLBench: The Robot Learning Benchmark & Learning Environment," arXiv preprint, arXiv:1909.12271, 2019.

[8] Y. Zhu, et al., "Robosuite: A Modular Simulation Framework and Benchmark for Robot Learning," arXiv preprint, arXiv:2009.12293, 2020.

[9] D. Han, et al., "A Survey on Deep Reinforcement Learning Algorithms for Robotic Manipulation," Sensors, 2023.

[10] T. Sun, et al, "RobotDrlSim: A Real Time Robot Simulation Platform for Reinforcement Learning and Human Interactive Demonstration Learning", Journal of Physics: Conference Sereis 2021.

[11] B. Tang et al., "IndustReal: Transferring Contact-Rich Assembly Tasks from Simulation to Reality," arXiv preprint, arXiv: 2305.17110, 2023