Scaling up Diffusion/Flow Matching Policy from simulation to the real world.

Keywords: Sim-to-real, reinforcement learning, imitation learning, flow matching, grasping and manipulation

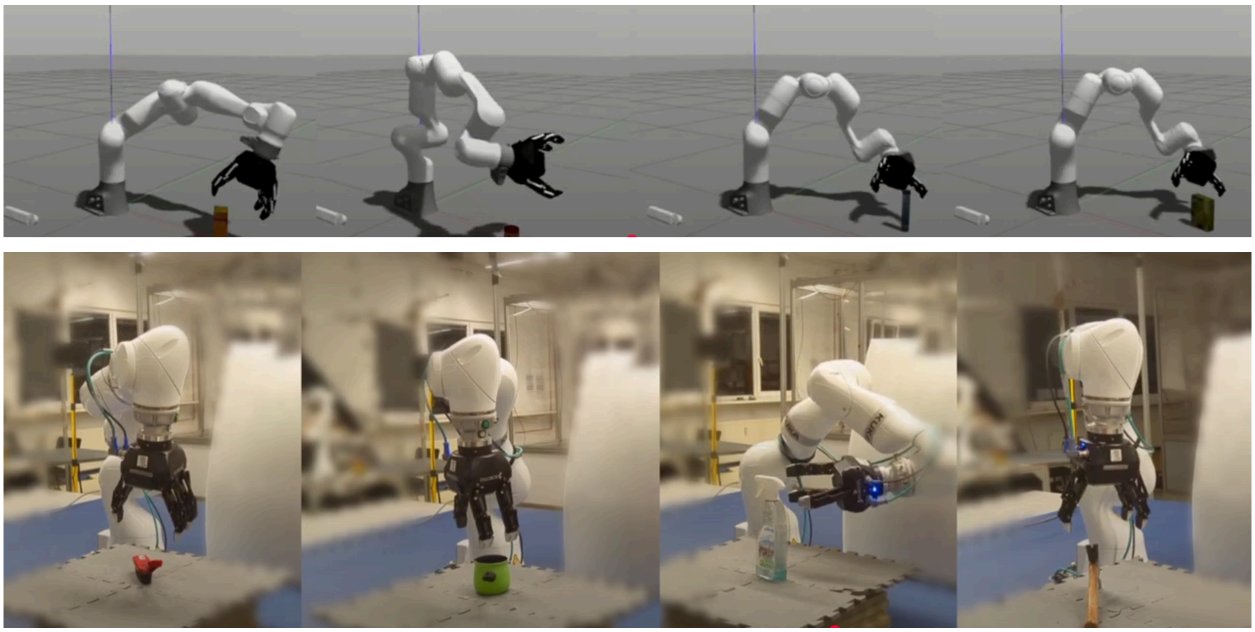

Fig. 1 Robot setup of KUKA iiwa and Robotiq 3F gripper, from simulation to real world ([video link]())

Training a policy with reinforcement learning (RL) in simulation instead of the real-world is an efficient approach to reduce the challenges associated with data collection in real-world environments. Still, the dynamics and fidelity are different to the real-world. Therefore, subsequent finetuning is a promising approach to leverage unlimited data in simulation and a few real-world trajectories to improve the policy performance.

This thesis investigates how to adapt a flow-matching policy, originally trained in simulation, to operate robustly in real-world environments. In particular, we seek to answer the following questions:

1. What is the best way to train the policy in simulation? Should we use a teacher/student framework? Can we combine RL and imitation learning?
2. Which fine-tuning methods are more sample-efficient when updating from limited real-world data? RL? Imitation learning?
3. What is the best way to adapt a flow-matching policy? Straightforward fine-tuning? Guidance? A residual velocity field?
4. How should we balance replaying simulated experiences versus incorporating new real-world trajectories during fine-tuning?

Relevant publications:

- Lipman et al., *Flow Matching for Generative Modeling*
- Zhao et al., *Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: a Survey*
- Lum et al., *DextrAH-G: Pixels-to-Action Dexterous Arm-Hand Grasping with Geometric Fabrics*
- Wang et al., *UniGraspTransformer: Simplified Policy Distillation for Scalable Dexterous Robotic Grasping (code)*

YOUR TASKS (2 students)

- Study the recent progress in flow matching policies.
- Implement a simulation of the real-world environment, e.g. using Mujoco or Isaac Sim.
- Train RL algorithms in simulation with a teacher/student framework
- Evaluate the result first in simulation.

- Get familiar with the real-world setup, how to control and teleoperate the robot etc.
- Design and try out different ways of fine-tuning the policy with simulation data plus real-world data.
- Evaluate the results in the real world.

YOUR QUALIFICATIONS

- Student (master) in natural sciences or engineering disciplines;
- Interest in machine learning, reinforcement learning, and robotics. Familiarity with diffusion, flow matching, and real-world robotic hardware is a plus;
- Very good knowledge of the Python programming language. Experience with PyTorch or JAX is a plus;
- Very good knowledge of numerical optimisation, probability theory, information theory, calculus and linear algebra;
- Self-motivated working;
- Proficient in English.

Contact: qian.feng@tum.de
Date: 13.05.2025