ETH zürich

Department of Mathematics
Computational Science Engineering

Robotic Seminar

Mingyuan Chi

14.11

Outline

1. Domain Randomization
2. High-Fidelity Simulation
3. Generalized and Robust Model
4. Limited Hardward/Interaction

Department of Mathematics
Computational Science Engineering

Outline

1. Domain Randomization

2. High-Fidelity Simulation

3. Generalized and Robust Model

4. Limited Hardware/Interaction

Domain Randomization

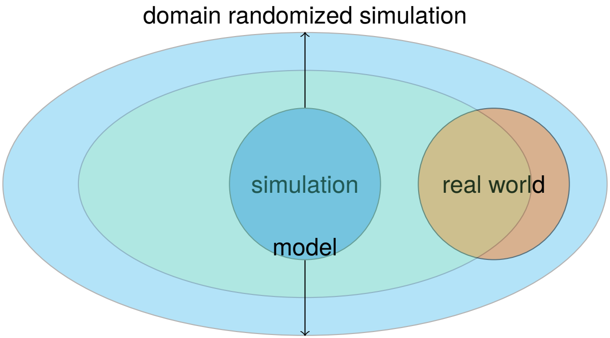


Figure: Domain Randomization

Domain Randomization

Reinforcement Learning with Adaptive Curriculum Dynamics Randomization for Fault-Tolerant Robot Control

Focus: Random Actuator Failures

- This research focuses on improving the fault tolerance of quadruped robots to actuator failure.
- It introduces an Adaptive Curriculum Reinforcement Learning algorithm with Dynamics Randomization (ACDR).
- The method trains robots to handle random actuator failures, developing a robust control policy.
- It eliminates the need for separate failure detection or policy switching mechanisms.

Domain Randomization

Crossing the Gap: A Deep Dive into Zero-Shot Sim-to-Real Transfer for Dynamics

Focus: Random Force

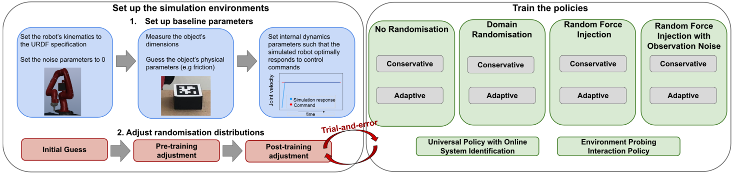


Figure: Random Force

- The paper explores the Random Force Injection (RFI) method.
- RFI involves injecting random noise into a simulated environment with fixed dynamics parameters.
- This method is simpler yet effective compared to complex domain randomisation.
- RFI requires less tuning and engineering effort, standing out for its ease of implementation.

Domain Randomization

Dynamics Randomization Revisited: A Case Study for Quadrupedal Locomotion

Focus: Proper Domain Random

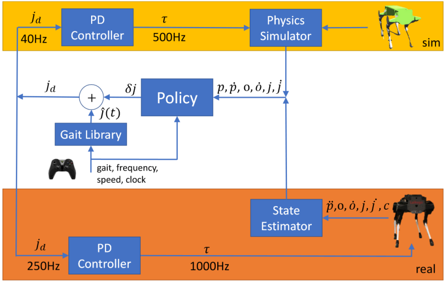


Figure: Dynamics Randomization Revisited: A Case Study for Quadrupedal Locomotion

- The study critiques blind application of dynamics randomization in sim-to-real transfer.
- It suggests that unnecessary randomization can lead to suboptimal and overly conservative policies.
- The authors emphasize using dynamics randomization only when significant modeling errors exist.
- It's advised to randomize or model only critical parameters to avoid detrimental effects on performance.

Domain Randomization

Analysis of Randomization Effects on Sim2Real Transfer in Reinforcement Learning for Robotic Manipulation Tasks

Focus: Proper Domain Random

- The research analyzes the impact of randomization on Sim2Real transfer in robotic manipulation.
- Increased randomization can facilitate Sim2Real transfer, but may hinder policy optimization in simulation.
- The study finds that a combination of full randomization and fine-tuning yields the best real-world performance.

Outline

1. Domain Randomization
2. High-Fidelity Simulation
3. Generalized and Robust Model
4. Limited Hardware/Interaction

High-Fidelity Simulation

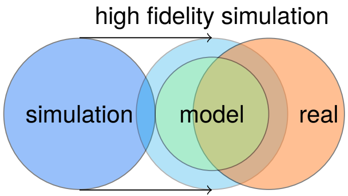


Figure: High Fidelity

Adapting Simulation Randomization with Real Experience

Closing the Sim-to-Real Loop

Focus: Simulation Parameters, Few World Trials

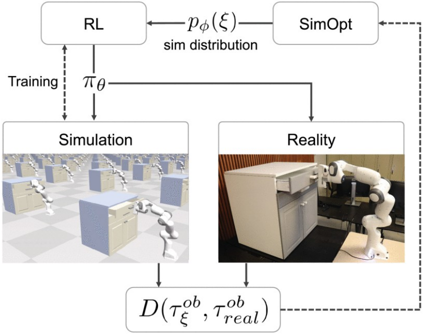


Figure: Adapting Simulation Randomization with Real Experience

- This study dynamically adjusts simulation parameters based on real-world trials.
- It aims to align simulated policy behavior more closely with actual performance.
- The method enhances the real-world transferability of trained robotic policies.

Learning Agile Locomotion for Quadruped Robots

Sim-to-real: Learning agile locomotion for quadruped robots

Focus: Traditional Method

- Enhances simulation fidelity with system identification, precise actuator modeling, and latency simulation.
- Introduces disturbances and utilizes a streamlined observation space for robust control.

Bridging the Gap in Flexible Object Manipulation

Sim2Real2Sim: Bridging the Gap Between Simulation and Real-World in Flexible Object Manipulation

Focus: Visual Feedback

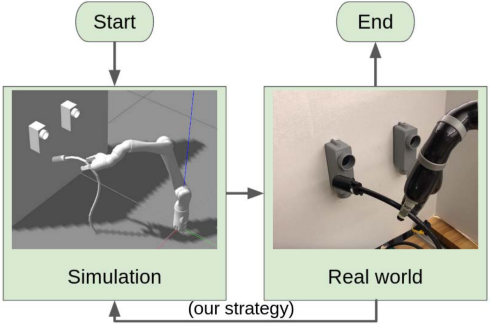


Fig. 2: Sim2Real2Sim flowchart representation.

- Sim2Real2Sim introduces a feedback and refinement step.
- Refines simulation models based on real-world data and experiences.
- Creates a feedback loop, evolving the simulation to be more accurate and representative.

Modeling Generalized Forces with Reinforcement Learning

Modelling generalized forces with reinforcement learning for sim-to-real transfer

Focus: Optimize GFM Model by Interacting with Environment

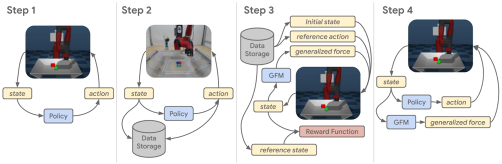


Figure: Training and Using GFM

- Train an agent in simulation with original model parameters.
- Collect real-world data using the trained agent.
- Learn the Generalized Force Model (GFM) to align simulation with real-world states.
- Retrain the agent with the updated hybrid model for the task of interest.

Learning Active Task-Oriented Exploration Policies

Learning active task-oriented exploration policies for bridging the sim-to-real gap

Focus: Learn SimOpt and Sim TrajOpt

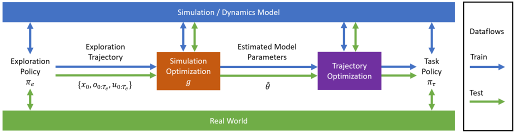


Figure: Exploration Policy Framework

- Framework for active, task-oriented exploration policies.
- Real-world trajectories generated by exploration policies.
- Optimizer (SimOpt) identifies dynamics parameters.
- Model-based TrajOpt finds a task policy for real-world performance.

Unsupervised Domain Adaptation with Dynamics-Aware Rewards

Unsupervised domain adaptation with dynamics-aware rewards in reinforcement learning

Focus: Dynamics-Aware Rewards in RL

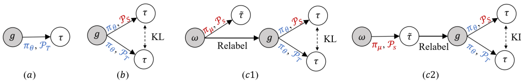


Figure: Dynamics-Aware Rewards Strategy

- max $\mathbb{E}_{\mathcal{B}_\mathcal{S}}[\text{log } q_\psi^{sas}(\text{source}|s_t, a_t, s_{t+1})] + \mathbb{E}_{\mathcal{B}_\mathcal{T}}[\text{log } q^{sas}(\text{target}|s_t, a_t, s_{t+1})]$
- max $\mathbb{E}_{\mathcal{B}_\mathcal{S}}[\text{log } q_\psi^{sa}(\text{source}|s_t, a_t)] + \mathbb{E}_{\mathcal{B}_\mathcal{T}}[\text{log } q_\psi^{sa}(\text{target}|s_t, a_t)]$
- $\Delta r(s_t, a_t, s_{t+1}) = \text{log} \frac{q_\psi^{sas}(\text{source}|s_t, a_t, s_{t+1})}{q_\psi^{sa}(\text{target}|s_t, a_t, s_{+1})} - \text{log} \frac{q_\psi^{sa}(\text{source}|s_t, a_t)}{q_\psi^{sa}(\text{target}|s_t, a_t)}$

- Introduces unsupervised domain adaptation with dynamics-aware rewards (DARS).
- Addresses dynamics shifts between different environments.
- Uses KL regularization to modify the reward function in unsupervised RL.
- Aligns trajectories in target environment with those in source environment.
- Encourages adaptable skills across different dynamics.

Sim2Real Predictivity

Sim2Real Predictivity: Does Evaluation in Simulation Predict Real-World Performance?

- The paper discusses the discrepancies between simulator performance and real-world outcomes.
- It highlights the issue of agents exploiting simulator flaws.
- Suggests that careful tuning of simulations can bridge the gap between simulation and real-world performance.

Learning Bipedal Walking for Humanoids With Current Feedback

Learning Bipedal Walking for Humanoids With Current Feedback

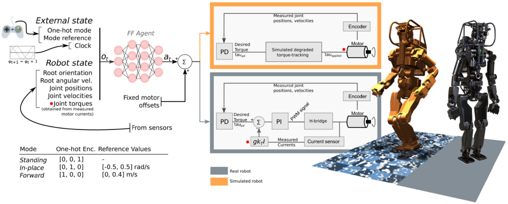


Figure: Bipedal Walking with Current Feedback

- Introduces simulating back-electromotive force (back-EMF) and current feedback.
- Enhances sim2real transfer for dynamic walking, stepping, and turning movements in robots.

Nerf2real: Sim2real Transfer Using Neural Radiance Fields

Nerf2real: Sim2real transfer of vision-guided bipedal motion skills using neural radiance fields

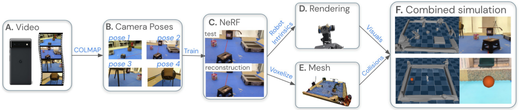


Figure: Nerf2real Methodology

- Utilizes neural radiance fields (Nerf) to create photorealistic simulation environments.
- Demonstrates training and transferring vision-based navigation and interaction policies for humanoid robots.

Safety-Critical Controller Verification

Safety-Critical Controller Verification via Sim2Real Gap Quantification

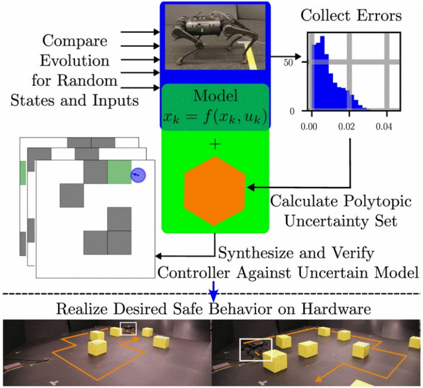


Figure: Sim2Real Gap Quantification

- Develops a method to detect and measure sim2real gap discrepancies.
- Uses "uncertain model" to reflect reality more closely in simulations.
- Designs and tests controllers in simulation with probabilistic approach.
- Ensures effectiveness and reliability of controllers in real-world application.

Auto-tuned Sim-to-Real Transfer

Auto-tuned sim-to-real transfer

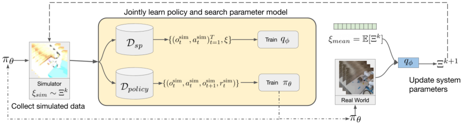


Figure: Auto-tuned Sim2Real Approach

- Introduces an approach for automatically tuning simulation parameters.
- Focuses on closely matching simulation parameters with real-world conditions.
- Presents the Search Param Model (SPM) for effective adjustment based on real-world observations.

LiDAR Sensor Modeling with GANs

LiDAR Sensor modeling and Data augmentation with GANs for Autonomous driving

$$\text{Realistic LiDAR Data} = \text{CycleGAN}(\text{Simulated LiDAR Data}, \text{Real-world LiDAR Features}) \tag{1}$$

- Uses CycleGANs for image-to-image translation from unpaired data.
- Formulates sensor modeling problem for LiDAR.
- Produces realistic LiDAR data from simulated inputs.
- Enables real2real translation to enhance LiDAR data resolution.

UniSim: A Neural Closed-Loop Sensor Simulator

UniSim: A Neural Closed-Loop Sensor Simulator

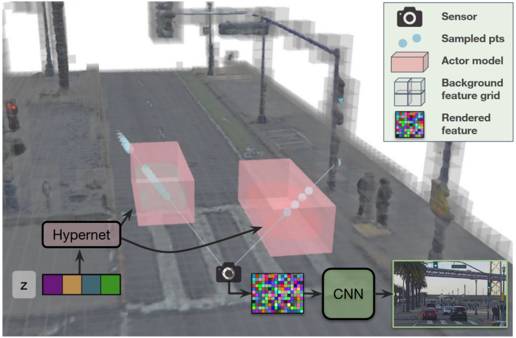


Figure: UniSim Simulation Process

The approach divides a 3D scene into a static background and dynamic actors. It separately queries neural feature fields for both, using volume rendering for neural feature descriptors. The static scene is modeled with a sparse feature-grid, while dynamic actors are represented via a hypernetwork and learnable latent. A convolutional network decodes feature patches into an image.

Outline

Generalized and Robust Model

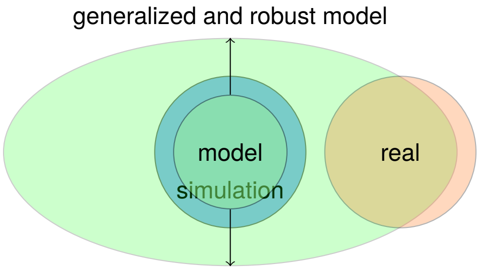


Figure: Generalized and Robust Model

Reinforcement Learning with Perturbed Rewards

Reinforcement learning with perturbed rewards

Algorithm 1 Reward Robust RL (sketch)

Input: $\tilde{\mathcal{M}}$, $\tilde{R}(s,a)$, η
Output: $Q(s,a)$, $\pi(s)$
Initialize value function $Q(s,a)$ arbitrarily.
while Q is not converged **do**
  Initialize state $s \in \mathcal{S}$, observed reward set $\tilde{R}(s,a)$
  Set confusion matrix $\tilde{\mathbf{C}}$ as identity matrix $\mathbf{I}$
  while s is not terminal **do**
    Choose a from s using policy derived from Q
    Take action a, observe s' and noisy reward $\tilde{r}$
    if collecting enough $\tilde{r}$ for all $\mathcal{S} \times \mathcal{A}$ pairs **then**
      Get predicted true reward $\bar{r}$ using majority voting
      Re-estimate $\tilde{\mathbf{C}}$ based on $\tilde{r}$ and $\bar{r}$ (using Eqn. 5)
    end if
    Obtain surrogate reward $\dot{r}$ ($\hat{\mathbf{R}} = (1-\eta)\cdot\mathbf{R} + \eta\cdot\tilde{\mathbf{C}}^{-1}\mathbf{R}$)
    Update Q using surrogate reward
    $s \leftarrow s'$
  end while
end while
return $Q(s,a)$ and $\pi(s)$

Figure: Perturbed Rewards

This paper introduces a robust framework to address noisy reward signals in robotics. The method compensates for perturbed rewards without assuming noise distribution, enhancing real-world robotic system performance and improving convergence in noisy environments.

Sim-to-Real via Sim-to-Sim

Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks

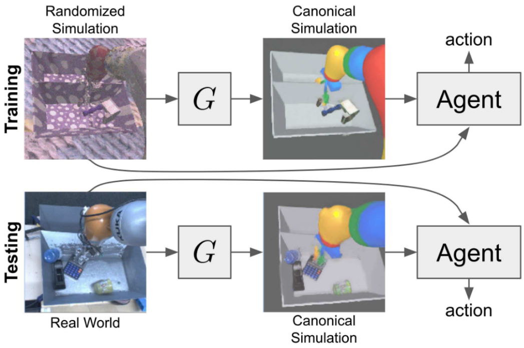


Figure: Randomized-to-Canonical Adaptation Networks (RCAN)

This study introduces the RCAN approach for sim-to-real transfer in robotics. It transforms randomized simulation images into canonical versions, enabling data-efficient training of robotic grasping algorithms. The method outperforms direct domain randomization and requires less real-world data, offering the additional benefit of an interpretable intermediate output.

DiAReL: Disturbance Awareness in RL for Robust Sim2Real Transfer

DiAReL: Reinforcement Learning with Disturbance Awareness for Robust Sim2Real Policy Transfer in Robot Control

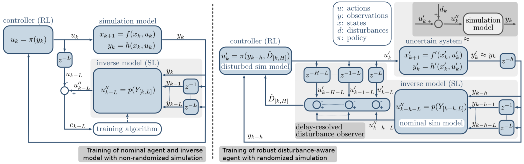


Figure: Disturbance-Augmented MDP in DiAReL

DiAReL introduces a Disturbance-Augmented MDP (DAMDP) to enhance RL agent robustness. It augments the state space with estimated disturbances, addressing challenges in environments with observation or action delays. The approach integrates data-driven disturbance estimation into the agent's observation space for effective training in simulated environments, aiding sim2real transfer in robot control.

Sim2Real with Gaussian Processes

How to sim2real with gaussian processes: Prior mean versus kernels as priors

- Discusses the use of Gaussian Processes (GPs) in sim2real applications.
- Advocates for embedding prior knowledge into GP kernels.
- The kernel-centric approach is more adaptable for simulation-based information.
- Offers robustness against sim2real mismatches, enhancing flexibility.

Vision-Guided Quadrupedal Locomotion with Cross-Modal Transformers

Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers

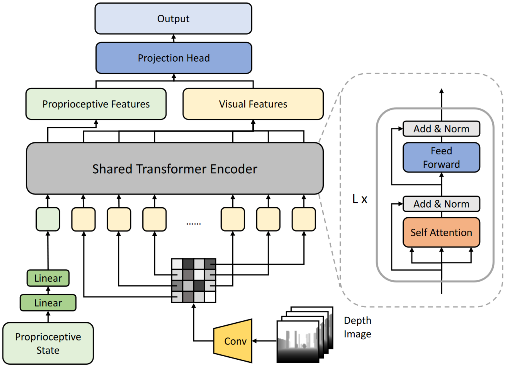


Figure: LocoTransformer Model

The paper introduces LocoTransformer, a novel RL model combining proprioceptive data with visual information. This multi-modal approach, using depth camera data and proprioceptive inputs, enables quadrupedal robots to navigate complex terrains and avoid obstacles more effectively.

Trustworthy Reinforcement Learning for Robustness, Safety, and Generalizability

Trustworthy Reinforcement Learning Against Intrinsic Vulnerabilities: Robustness, Safety, and Generalizability

The study focuses on enhancing the robustness of RL algorithms, particularly in scenarios with significant training and testing condition differences. It emphasizes the importance of handling mismatches between observed and actual states due to sensor errors or limitations. The paper also explores training methodologies, such as adversarial training, to improve reliability and avoid overly conservative policy outcomes in diverse and challenging real-world applications.

Discovering Blind Spots in Reinforcement Learning

Discovering blind spots in reinforcement learning

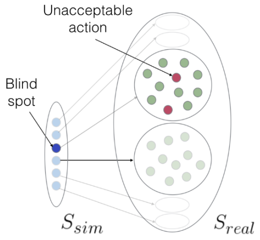


Figure: Blind spots in RL

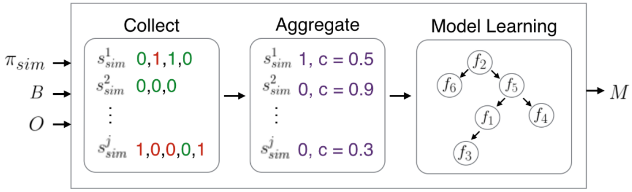


Figure: Oracle feedback in RL

This paper introduces a method to identify 'blind spots' in RL agents due to insufficient state representations. It leverages oracle feedback to learn about these blind spots, which are crucial for ensuring the safe and effective deployment of RL agents in real-world scenarios.

TuneNet: Residual Tuning for Sim-to-Real Transfer

Tunenet: One-shot residual tuning for system identification and sim-to-real robot task transfer

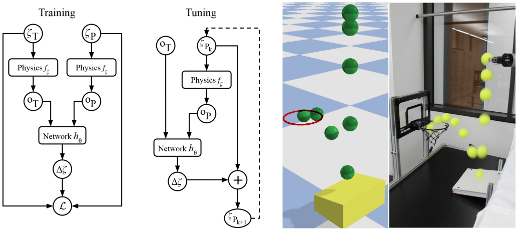


Figure: TuneNet Approach

TuneNet is a machine-learning-based approach for bridging the sim-to-real gap in robot training. It utilizes a novel residual tuning technique, enabling simulation parameter adjustment with minimal effort and a single observation from the target environment, thereby enhancing the accuracy and applicability of simulation models in real-world scenarios.

Sim2Real Transfer without Dynamics Randomization

Sim2Real Transfer for Reinforcement Learning without Dynamics Randomization

This paper presents a framework that utilizes Operational Space Control (OSC) for reinforcement learning in Cartesian space. The approach allows for fast learning with adjustable degrees of freedom, enabling safe learning on real robots or flexible goal-conditioned policies that can be easily transferred from simulation to real robots.

Universal Policy with Online System Identification

Preparing for the unknown: Learning a universal policy with online system identification

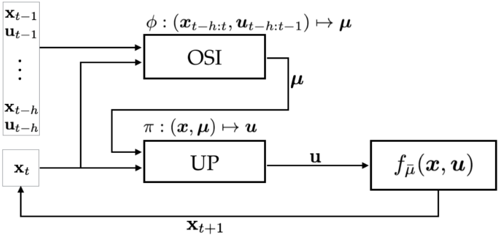


Figure: UP - OSI Approach

The method introduces a novel approach for learning control policies under unknown dynamic models. It includes two key components:

- **Universal Policy (UP)**: Trained over a wide range of dynamic models for adaptability in various conditions.
- **Online System Identification (OSI)**: Predicts dynamic model parameters from recent states and actions, enhancing the UP's adaptability and responsiveness.

Deep Whole-Body Control

Deep Whole-Body Control: Learning a Unified Policy for Manipulation and Locomotion

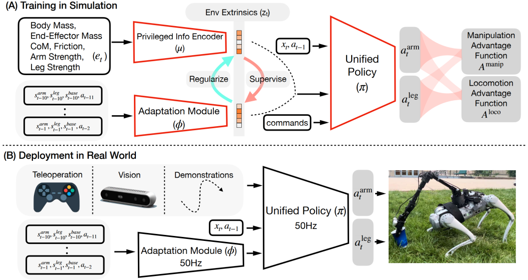


Figure: Advantage Mixing and Regularized Online Adaptation

This paper introduces Advantage Mixing to train robust policies for high-DoF robots, linking arm actions with manipulation tasks and leg actions with locomotion. It also presents Regularized Online Adaptation for Sim-to-Real transfer, which uses an adaptation module to estimate environment extrinsics from sensory observations, aiding in real-world application.

Quadruped Robot Control

Sim-to-real: Quadruped Robot Control with Deep Reinforcement Learning and Parallel Training

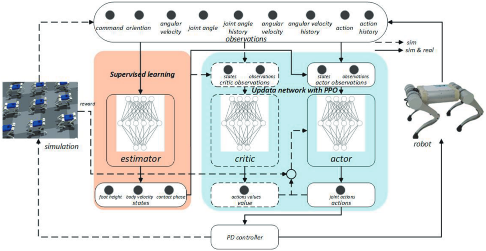


Figure: Neural Network Framework for Quadruped Robot Control

The study proposes an end-to-end neural network framework for quadruped robot control, integrating an estimator network for ontology states with critic and actor networks. This robust control system enhances the adaptability and stability of the robot in various environments.

Iterative Learning in Human-Robot Interaction

i-Sim2Real: Reinforcement Learning of Robotic Policies in Tight Human-Robot Interaction Loops

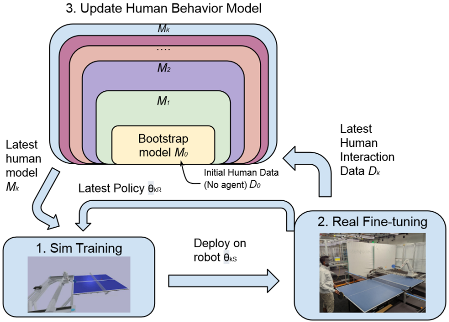


Figure: i-Sim2Real Framework

https://proceedings.mlr.press/v205/abeyruwan23a.html
The i-S2R framework is an iterative learning process that alternates between simulation and real-world deployment. It starts with a coarse model of human behavior and refines both the human behavior model and the robot's policy iteratively through successive rounds of simulation and real-world interaction.

Stabilizing Sim2Real RL via Domain Decomposition

Adaptability preserving domain decomposition for stabilizing sim2real reinforcement learning

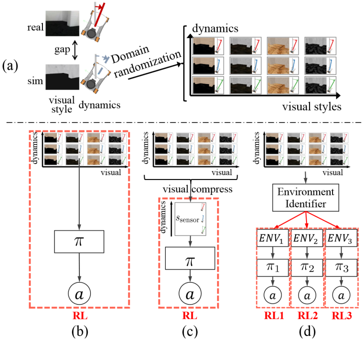


Figure: Domain Decomposition in RL

https://ieeexplore.ieee.org/abstract/document/9341124
Presents an innovative Domain Decomposition (DD) algorithm that stabilizes sim2real reinforcement learning by training separate policies for different simulated environments. This method maintains the adaptability of the overall policy while enhancing training stability. The paper introduces an environment identifier to classify simulation environments for targeted training and theoretically proves the preservation of adaptability in RL policies.

Outline

1. Domain Randomization
2. High-Fidelity Simulation
3. Generalized and Robust Model
4. Limited Hardward/Interaction

Collaborative Multi-Robot Learning
Towards Closing the Sim-to-Real Gap in Collaborative Multi-Robot Deep Reinforcement Learning

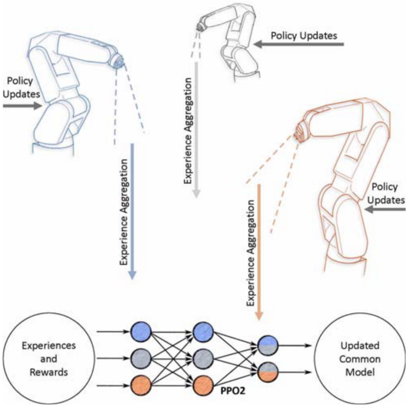


Figure: Multi-Robot Learning

https://ieeexplore.ieee.org/abstract/document/9310796
This study explores the use of multiple agents to learn the same task while sharing a policy, aiming to close the sim-to-real gap in collaborative multi-robot deep reinforcement learning.

Addressing Stochastic State Transition Delays

Sim2Real Transfer for Deep Reinforcement Learning with Stochastic State Transition Delays

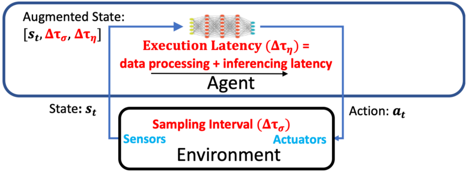


Figure: Time Delay and Sampling Rate

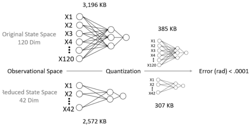
https://proceedings.mlr.press/v155/sandha21a.html
Discussing the challenges of Sim2Real transfer in Deep RL due to sensor sampling rates and actuation delays, this paper introduces the Time-in-State RL approach, which integrates timing delays into training observations, enhancing the robustness of Deep RL policies.

Imitation Learning for Ultra-Low-Cost Robots

Closing the Sim-to-Real Gap for Ultra-Low-Cost, Resource-Constrained, Quadruped Robot Platforms

	Unitree A1	Petoi Bittle	Ratio
Cost	$10,000 USD	$299 USD	33x
Weight	12 kg	.29 kg	41x
Dimensions	.5 x .3 x .4 m	.2 x .11 x .11 m	2.5x
Degrees of Freedom (DoF)	12 (Leg: 3)	8 (Leg: 2)	1.5x
Battery Capacity	25.2V 4200mAh	7.4V 1000mAh	3x
Motor Resolution	.022°	1°	45x
IMU	Yes	Yes	NA
Motor Feedback	Yes	No	NA
Foot Pressure Sensor	Yes	No	NA
LiDAR	Yes	No	NA
Computing	ARM Cortex-A72 2.5GHz	Nyboard V1 ATMega328P 20MHz	125x
Optional Additional Computing	NVIDIA TX2 1.3GHz	Raspberry Pi Zero 2W 1GHz	1.3x

http://jabbourjason.com/pubs/Closing_the_sim-to-real_gap_for_ultra-low-cost.pdf



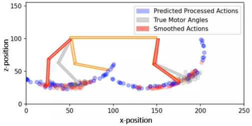


Figure: Quadruped Robot Platforms

This research adapts imitation learning pipelines to ultra-low-cost robots with poor actuation, limited computing resources, and limited sensors. It proposes practical solutions to overcome these difficulties, laying the groundwork for a future with globally-accessible, capable, ultra-low-cost robots.

Thanks for your attention

Mingyuan Chi
minchi@student.ethz.ch