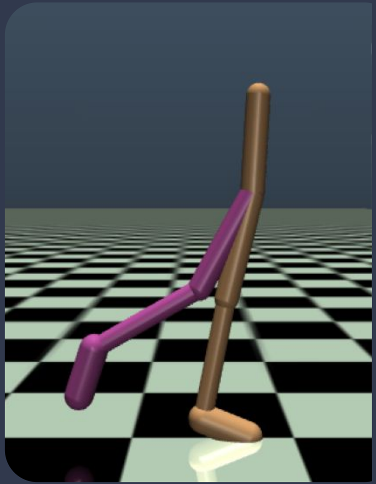
Domain Randomization Techniques for Reinforcement Learning: Bridging the Reality Gap

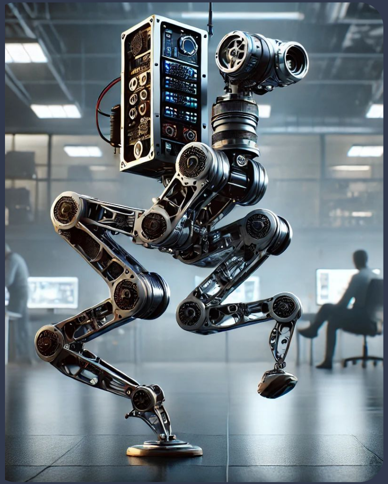
Ivan Necerini
s345147

Giorgia Modi
s330519

Andrea Delli
s331998

The Sim-to-Real Gap





- **Challenge:** Transferring RL policies from simulation to real-world applications
- **Limitation:** Simulated environments enable efficient and safe RL training but fail to capture real-world dynamics
- **Issue:** Policies trained in simulation often perform poorly when deployed on actual robotic hardware
- **Approach:** To address the sim-to-real gap, we explored Domain Randomization (DR)
- **Method:** We implement a sim-to-sim setup by introducing a controlled domain gap

Environments

Two environments provided by OpenAI Gym: **Hopper, Walker2d**

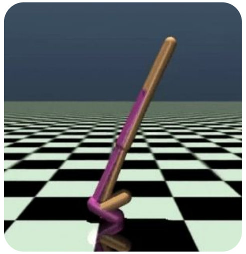
- Continuous state and action spaces
- Reward function
- Target: default env
  Source: -1kg torso mass

DEFAULT HOPPER

Body Name	Mass
torso	3.534
thigh	3.927
leg	2.714
foot	5.089

DEFAULT WALKER2D

Body Name	Mass
torso	3.534
thigh	3.927
leg	2.714
foot	2.941
thigh_left	3.927
leg_left	2.714
foot_left	2.941



Key Contributions of Our Study

Comparative analysis of UDR and TNR across **single and multi-mass** randomization setups

Insights into the **effect of dynamic parameters** (e.g., leg and foot masses)

Evaluation of **increasing** the source-to-target **domain gap**

RL Training Pipeline

📥 Input

🧠 PPO Training

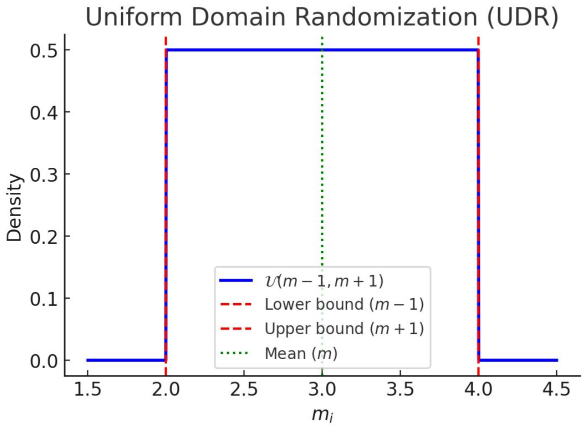
🎯 Trained Policy

🏁 Evaluation

Domain Randomization

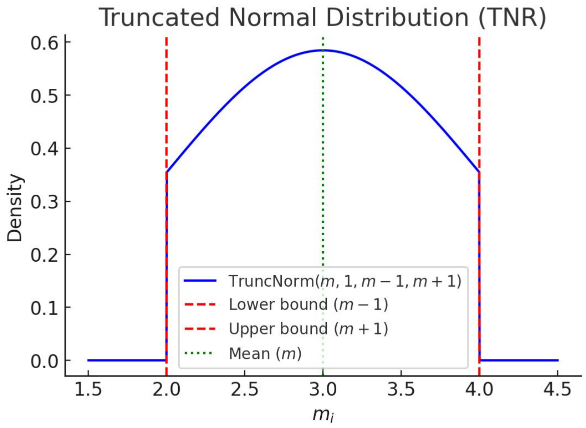
Uniform Domain Randomization (UDR)

$$m_i^{\text{new}} \sim \text{Uniform}(m_i - 1, m_i + 1)$$



Truncated Normal Domain Randomization (TNR)

$$m_i^{new} \sim \text{TruncNorm}(m_i, 1, m_i - 1, m_i + 1)$$



Preliminary Experiments

- Learning Rate Schedules
- Grid Search for Hyperparameters Tuning

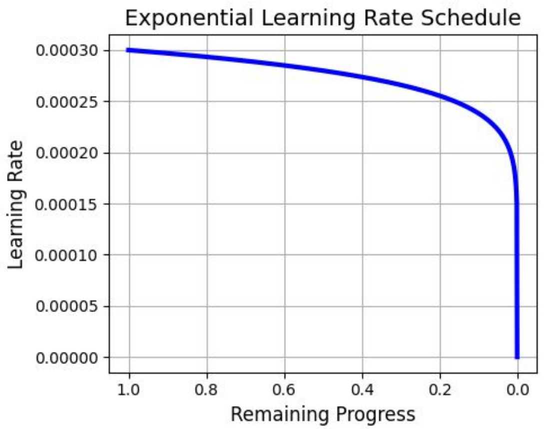
Learning Rate Schedules

Constant LR $$\mathrm{lr} = v_{\text{initial}}$$

Linear LR $$\mathrm{lr}(p) = p \cdot v_{\text{initial}}$$

Exponential LR $$\mathrm{lr}(p) = v_{\text{initial}} \cdot p^r$$

v_{initial}	initial LR value (0.0003)
p	remaining progress (1→0)
r	decay rate (0.1)



LR Schedule	Test Reward (avg ± std)
Constant	1202.06 ± 324.22
Linear	1194.65 ± 237.26
Exponential	1263.31 ± 215.91

- Exponential LR schedule outperformed others
- Gradual reduction ⇒ balances exploration and exploitation

Grid Search for Hyperparameters Tuning

- Tested multiple **hyperparameters** combinations on Hopper source domain (100k timesteps)
- **Best configuration** was further trained for 2M timesteps
- **Default PPO hyperparameters** performed best after a longer training!
- **Conclusion**: We adopted standard PPO hyperparameters for the rest of the study

Tested ranges

```
n_epochs ∈ {5, 10, 20}
clip_range ∈ {0.1, 0.2, 0.3}
gae_lambda ∈ {0.9, 0.95, 0.99}
gamma ∈ {0.95, 0.99, 0.999}
batch_size ∈ {32, 64, 128}
```

Best during grid search

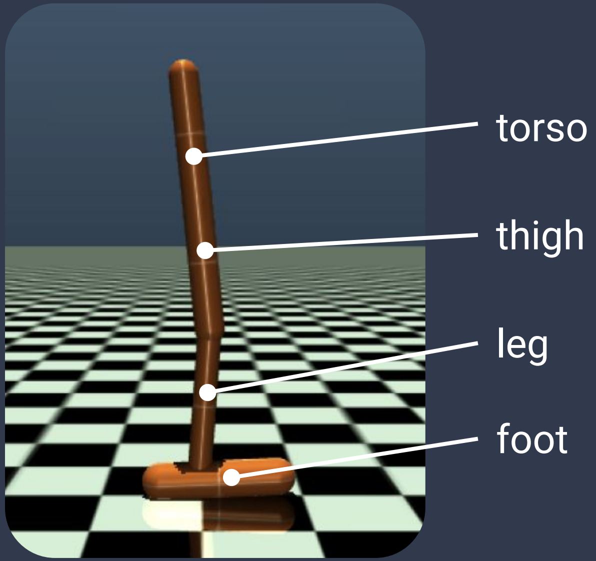
```
n_epochs = 20
clip_range = 0.3
gae_lambda = 0.99
gamma = 0.999
batch_size = 128
```

Default PPO params

```
n_epochs = 10
clip_range = 0.2
gae_lambda = 0.95
gamma = 0.99
batch_size = 64
```

Experiments on the Hopper Environment

- Baseline Models Evaluation
- Domain Randomization
- Single Mass Domain Randomization
- Increase Source-Target Gap



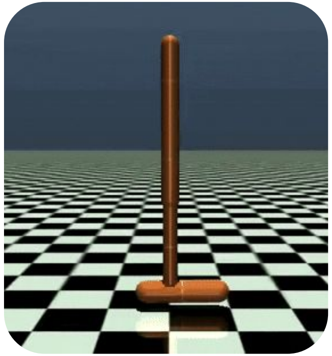
Baseline models evaluation

PPO with exponential LR schedule
2M training timesteps
1000 test episodes

Target: default Hopper

Source: Hopper with -1kg for torso mass

Three setups
Source→Source
Source→Target
Target→Target



Source→Target

Model Setup	Test Reward (avg ± std)
Source → Source	1471.47 ± 311.72
Source → Target	1495.05 ± 253.62
Target → Target	1696.48 ± 93.64

- Source→Source and Target→Target perform well
- Source→Target performed comparably to Source→Source, despite domain gap!
  ⇒ simpler dynamics of the source environment
- Source→Target didn't perform comparably to Target→Target
- Target→Target is ideal but impractical

Domain Randomization

- **Goal:** Improve policy robustness and generalization
- **Strategies tested:**
  Uniform Domain Randomization (UDR)
  Truncated Normal Domain Randomization (TNR)
- Randomization applied to **link masses** (thigh, leg, foot), excluding the torso

Configuration	**UDR** (avg ± std)	**TNR** (avg ± std)
Source $\rightarrow$ Source	1719.87 ± 8.32	1286.19 ± 366.47
Source $\rightarrow$ Target	1721.65 ± 9.58	1351.57 ± 133.70

- UDR ⇒ significantly improved performance
- TNR ⇒ higher variance and lower rewards
- Source→Target with UDR outperform baseline Target→Target

UDR

TDR

Single Mass Randomization

- Randomizing **one** mass at a time (thigh, leg, foot)

- **Goal**: isolate the impact of each individual mass in DR

- UDR ⇒ leg mass randomization achieved the best results (vs thing and foot)

- TNR ⇒ Single mass randomizations outperformed full randomization, with leg mass achieving the best results
  ⇒ Randomizing one mass at a time reduces instability

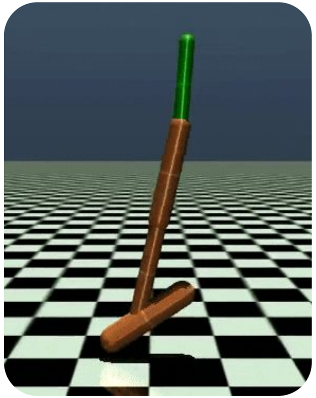
- From now on: UDR for more stable results

Configuration	UDR (avg ± std)			TNR (avg ± std)		
	Thigh	Leg	Foot	Thigh	Leg	Foot
Source → Source	1459.02 ± 101.61	1721.99 ± 104.94	1320.71 ± 268.69	1725.21 ± 17.97	1791.63 ± 246.64	1768.91 ± 53.58
Source → Target	1466.23 ± 112.99	1723.13 ± 103.06	1321.97 ± 271.90	1725.12 ± 17.60	1786.45 ± 261.22	1769.61 ± 52.63

Increasing the Domain Gap: Thin Hopper

More challenging source-to-target gap: Decreased torso mass in source domain (-2 kg) vs previous torso mass (-1 kg)

Thin Hopper



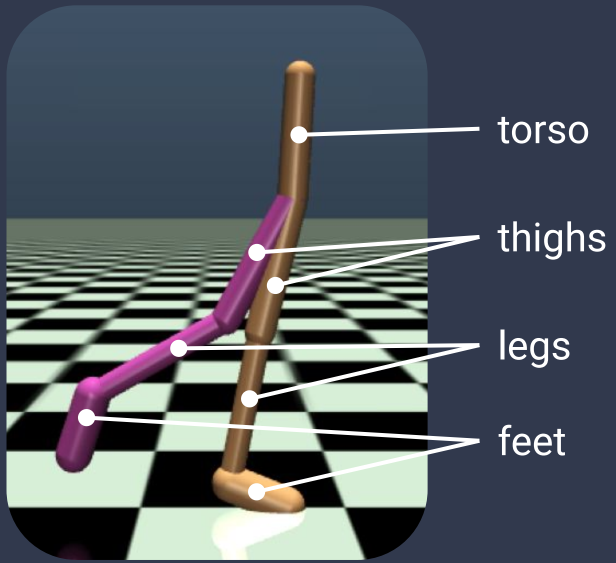
Source→Target (no UDR)

Model Setup	Test Reward (avg $\pm$ std)
Source $\rightarrow$ Source (no DR)	1190.83 $\pm$ 154.54
Source $\rightarrow$ Target (no DR)	1190.58 $\pm$ 160.12
Source $\rightarrow$ Source (UDR)	1412.48 $\pm$ 249.39
Source $\rightarrow$ Target (UDR)	1416.21 $\pm$ 256.35
Target $\rightarrow$ Target	1696.48 $\pm$ 93.64

- General decrease in performance
- UDR remained highly effective
- Not reached comparable values of Target→Target

Experiments on Walker2d Environment

- Baseline Models Evaluation
- Domain Randomization
- Single Mass Domain Randomization
- Increase Source-Target Gap



Baseline models evaluation

PPO with exponential LR schedule
2M training timesteps
1000 test episodes

Target: default Walker2D

Source: Walker2D with -1kg for torso mass

Three setups

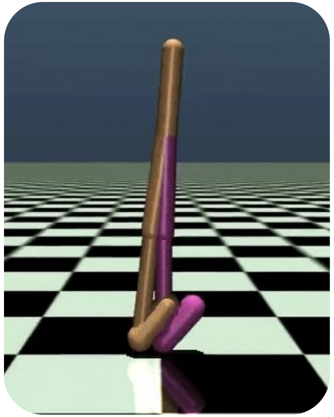
Source→Source

Source→Target

Target→Target

Configuration	Test Reward (avg ± std)
Source → Source	2376.83 ± 765.40
Source → Target	2039.69 ± 940.58
Target → Target	2293.77 ± 453.39

- Source→Source and Target→Target perform well
- Source→Target performed worse than Source→Source, with high variance ⇒ visible domain gap!

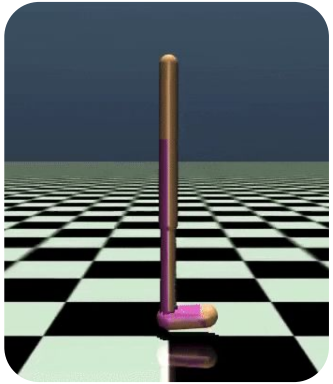


Source→Target

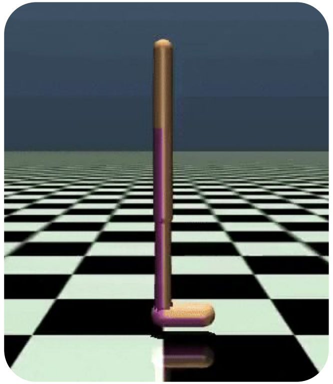
Domain Randomization

Strategies tested:
Uniform Domain Randomization (UDR)
Truncated Normal Randomization (TNR)

Randomization applied to all **body segment masses** (thigh, leg, foot), excluding the torso



UDR
Source→Target



TNR
Source→Target

Configuration	UDR (avg ± std)	TNR (avg ± std)
Source → Source	2854.67 ± 583.67	1605.18 ± 761.54
Source → Target	2861.97 ± 576.78	1589.14 ± 749.07

- UDR ⇒ significantly improved performance, outperforming baselines.

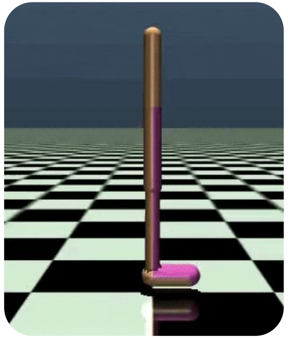
  Effectively filled the domain gap!

- TNR ⇒ higher variance and lower rewards

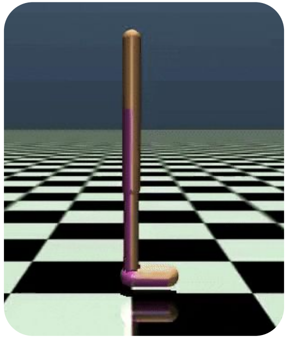
Single Mass Randomization

Randomizing **one pair** of masses at a time (both thighs, legs, or feet), to ensure consistency and symmetry

- Significantly more effective than all-masses DR
- UDR ⇒ leg mass randomization achieved the best results
- TNR ⇒ Worse results than UDR and higher variance



UDR (leg)
Source→Target



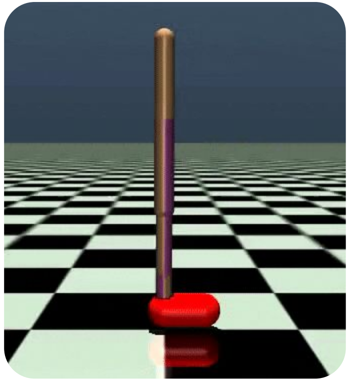
TNR (leg)
Source→Target

Configuration	UDR (avg ± std) Thigh	UDR Leg	UDR Foot	TNR (avg ± std) Thigh	TNR Leg	TNR Foot
Source → Source	3360.91 ± 785.10	4228.56 ± 845.09	3919.82 ± 975.15	3109.39 ± 1116.03	1221.81 ± 725.5	2656.49 ± 779.30
Source → Target	3345.01 ± 812.06	4138.72 ± 826.53	3939.74 ± 956.36	3201.08 ± 1176.75	1260.49 ± 740.97	2689.84 ± 772.25

Increasing the Domain Gap: BigFoot Walker2d

More challenging source-to-target gap:
Triple the feet masses

3kg ➔ 9kg per foot!

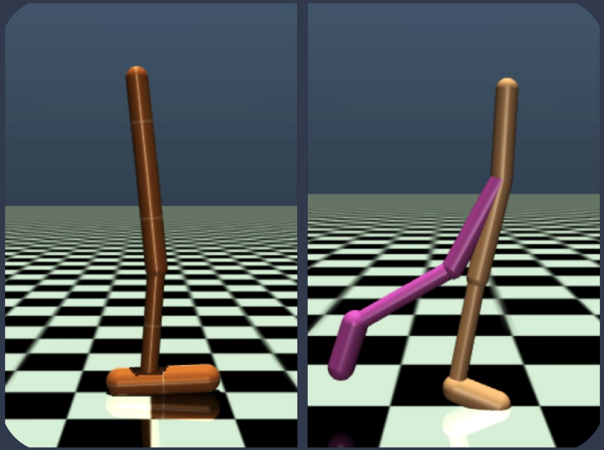


BigFoot Walker2d (visualization)
UDR Source→Target

Model setup	Test Reward (avg $\pm$ std)
Source $\to$ Source (no DR)	2957.91 ± 894.13
Source $\to$ Target (no DR)	2894.83 ± 896.59
Source $\to$ Source (UDR)	4158.03 ± 757.54
Source $\to$ Target (UDR)	4140.05 ± 784.25
Source $\to$ Source (TNR)	2314.29 ± 1770.93
Source $\to$ Target (TNR)	2353.83 ± 1781.62
Target $\to$ Target	2293.77 ± 453.39

- Higher rewards wrt baseline, no domain gap
- UDR outperforms all other methods
- TNR has low reward and higher variance

Conclusions



- **LR Schedule** and **grid search** allowed to choose the best training setup
- **Hopper and Walker2d** ⇒ two specular environments
- **Domain gap** is not always visible
- **UDR** achieved higher rewards and lower variance (robustness), outperforming TNR
- **Single mass randomization** ⇒ importance of certain masses (*legs*) during training process
- **Increasing domain gap** highlighted the necessity of domain randomization

Future works: different DR approaches, randomization ranges, and mass combinations

THANK YOU
FOR YOUR ATTENTION!

Ivan Necerini
s345147

Giorgia Modi
s330519

Andrea Delli
s331998