RF-DROPO: Data-Efficient Adaptive Domain Randomization for Zero-Shot Sim-to-Real Transfer in Soft Robotics

Andrea Protopapa[1], Gabriele Tiboni[1], Julien Vandaele[2],
Tatiana Tommasi[1], Raffaello Camoriano[1,3], Christian Duriez[2], and Giuseppe Averta[1]

***Abstract*— Simulation is a practical tool for training control policies for soft robots, but transferring these policies to real systems remains difficult because soft-body dynamics are hard to model accurately and only partially observable in practice. We present RF-DROPO, an offline adaptive domain randomization method that uses a small amount of real-world trajectory data to infer a distribution over simulator parameters, which is then used to train reinforcement learning policies in simulation. Unlike prior approaches that rely on full-state observability or simulator resets to arbitrary intermediate states, RF-DROPO performs reset-free trajectory matching from a shared initial condition and progressively increases the rollout horizon during inference. This makes the method well suited to deformable robotic systems, where accurate state reconstruction is often unavailable and data collection is expensive. We evaluate the approach on soft-robot control tasks including reaching and pushing, and show that the inferred parameter distributions support more reliable transfer than static domain randomization and existing adaptive baselines. We also report zero-shot deployment on a physical soft robot. Overall, the results suggest that lightweight simulator adaptation can substantially improve the practicality of sim-to-real policy learning for deformable robots.**

I. Introduction

Soft robots are promising for safe interaction and adaptable manipulation, but their compliance also makes control difficult. Their dynamics are strongly nonlinear, sensitive to material properties, and only partially observable through practical sensing. As a result, Reinforcement Learning (RL) policies trained in simulation often face a substantial reality gap when deployed on hardware.

A common way to improve transfer is to randomize simulator parameters during RL training, with an approach referred to as Domain Randomization (DR) [1]. However, manually and statically choosing randomization ranges (a.k.a. Uniform Domain Randomization – UDR) is difficult for soft robots, where even small changes in elasticity or contact parameters can lead to large changes in behavior. This motivates methods that use a small amount of real data to adapt the DR parameters distribution before policy learning.

This study was carried out within the project FAIR - Future Artificial Intelligence Research - and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

[1]Politecnico di Torino, Turin, Italy, [firstname].[lastname]@polito.it
[2]Inria, Lille, France
[3]Istituto Italiano di Tecnologia, Genoa, Italy

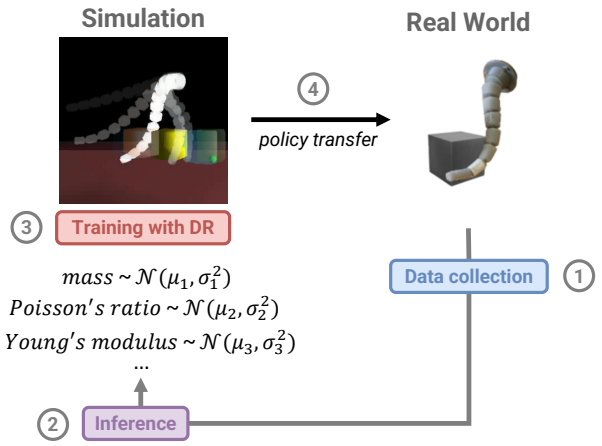


Fig. 1. Overview of RF-DROPO. A small amount of target-domain trajectory data is collected offline, used to infer a distribution over simulator parameters through reset-free trajectory matching, and then employed for domain-randomized policy learning in simulation.

In this work, we revisit this problem through the lens of offline Adaptive Domain Randomization (ADR). We consider the setting where a limited number of trajectories are collected in the target domain, after which all policy learning is performed in simulation. Our focus is on soft robots, where two practical issues are especially important: full state reconstruction is generally unavailable, and resetting a simulator to arbitrary states along a recorded trajectory is not realistic.

We address these challenges with RF-DROPO, a reset-free inference method that estimates a distribution over simulator dynamics parameters directly from recorded trajectories [2]. The key idea is to replay the action sequence in simulation from a shared initial state, match the resulting rollouts to the observed trajectory, and progressively increase the rollout horizon during optimization. The inferred parameter distribution is then used for RL with Domain Randomization.

An overview of the proposed framework is shown in Fig. 1.

II. Method

Let ξ denote the vector of simulator dynamics parameters, including quantities such as masses, friction coefficients, or elastic properties, and let $p_\phi(\xi)$ be the corresponding parameter distribution. Instead of training on a single nominal simulator, we seek a distribution over dynamics that is

consistent with the target system and can be used for robust policy learning.

Assume we collect an offline trajectory dataset

$$\mathcal{D} = \{(s_t, a_t, s_{t+1})\}_{t=0}^{T-1}, \tag{1}$$

in the target domain. Prior offline adaptation methods often assume that the simulator can be reset to every observed state s_t, which requires accurate full-state reconstruction. This assumption is problematic for soft robots.

RF-DROPO removes this requirement by replaying the recorded action sequence from a shared initial state s_0. The parameter distribution is estimated by maximizing the likelihood of the observed trajectory under the simulator rollouts:

$$\phi^\star = \arg\max_{\phi} \sum_{t=0}^{T-1} \log P_\phi(s_{t+1} \mid s_0, a_{0:t}), \tag{2}$$

where $a_{0:t} = (a_0, \ldots, a_t)$ and $P_\phi(s_{t+1} \mid s_0, a_{0:t})$ denotes the probability of reaching the observed next state after replaying the $t+1$ recorded actions from the initial state in simulation under parameters sampled from $p_\phi(\xi)$. In practice, this likelihood is approximated by Monte Carlo rollouts in a black-box simulator and optimized with a gradient-free method.

A practical challenge is that open-loop rollouts accumulate simulation errors over time. To stabilize inference, RF-DROPO uses an *horizon scheduling* mechanism: rather than optimizing the full trajectory objective from the beginning, it starts with short trajectory prefixes and progressively increases the rollout horizon,

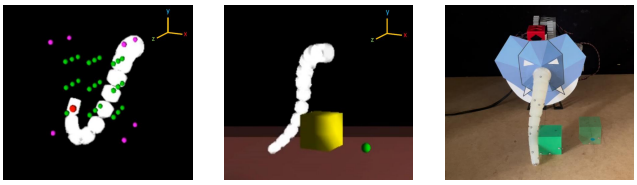
$$\mathcal{L}^{(\tau)}(\phi) = \sum_{t=0}^{\tau-1} \log P_\phi(s_{t+1} \mid s_0, a_{0:t}), \qquad \tau \leq T. \tag{3}$$

This provides a curriculum over trajectory length: early iterations focus on short-horizon matching, while later iterations incorporate longer-horizon effects once the parameter estimate has become more reasonable.

After inference, the optimized distribution $p_{\phi^\star}(\xi)$ is used for Domain Randomization during Reinforcement Learning. At the beginning of each episode, simulator parameters are sampled from the inferred distribution, and the policy is trained entirely in simulation before zero-shot transfer to the target domain.

III. Experimental Setup

We evaluate RF-DROPO on a cable-driven soft continuum manipulator, also known as *Trunk* [3], in both simulation and real-world deployment. The robot is controlled through cable actuation and exhibits highly nonlinear, partially observable soft-body dynamics. In our experiments, we use SOFA [4] as FEM-based simulation framework. We consider two manipulation tasks, depicted in Fig. 2. In *TrunkReach*, the robot tip is driven toward a Cartesian target, providing a relatively clean setting to assess adaptation quality. In *TrunkPush*, the robot must push an object toward a goal location, introducing contact-rich interactions and making transfer substantially more sensitive to simulator mismatch. The latter has been also deployed on real hardware to evaluate a zero-shot sim-to-real transfer on a real-world robotic platform.



(a) (b) (c)

Fig. 2. Tasks and platform overview. (a) reaching task *TrunkReach*; (b) pushing task *TrunkPush* in simulation; (c) physical soft-robot platform used for zero-shot sim-to-real evaluation for the *TrunkPush* task.

TABLE I

Summary of parameter inference quality on the target domains. Lower is better.

Method	*TrunkReach* MSE	*TrunkPush* MSE
BayesSim	16.60 ± 0.30	25.50 ± 5.14
NPDR	8.38 ± 3.22	1.71 ± 0.17
RF-DROPO	$\mathbf{1.70 \pm 1.93}$	$\mathbf{0.62 \pm 0.34}$

For all experiments, a small amount of target-domain trajectory data is collected offline and used only for simulator adaptation. Policy learning is then performed entirely in simulation, and the resulting policy is transferred zero-shot to the target domain.

IV. Results

We focus on two questions: (*i*) whether the inferred parameter distributions are consistent with the target dynamics, and (*ii*) whether this leads to stronger downstream transfer, especially in the real-world zero-shot setting.

A. Parameter inference quality

Before considering policy performance, we verify in simulation that RF-DROPO produces parameter distributions that are consistent with a target system domain. As shown in Tab. I, on the simulated benchmarks RF-DROPO yields lower parameter-estimation error than the compared adaptive baselines, namely BayesSim [5] and NPDR [6]. Specifically, we report the mean squared error (MSE) between the ground-truth target parameters and parameters sampled from the inferred distribution; lower values therefore indicate that the inferred distribution is closer to the target dynamics. These results suggest that reset-free trajectory matching can recover informative simulator distributions even under partial observability. This point is important because the method is not only producing better policies downstream, as shown in Sec. IV-B, but is also adapting the simulator in a meaningful way. These results serve mainly as supporting evidence: they show that the inferred distributions are not arbitrary, but capture properties of the target dynamics that are relevant for transfer.

TABLE II

ZERO-SHOT SIM-TO-REAL PERFORMANCE ON THE REAL TRUNKPUSH TASK. A ROLLOUT IS SUCCESSFUL IF THE FINAL CUBE POSITION LIES WITHIN THE GIVEN DISTANCE FROM THE GOAL. WE REPORT SUCCESS RATES (%) OVER 25 REAL-WORLD ROLLOUTS PER METHOD.

Method	15.25 mm	21.57 mm	30 mm
UDR	0	0	0
BayesSim	12	16	28
NPDR	0	4	36
RF-DROPO	**32**	**56**	**60**

B. Zero-shot sim-to-real transfer

Our main result is the real-world zero-shot deployment on the physical soft robot. Tab. II reports success rates at three representative thresholds, including two geometrically meaningful ones: 15.25 mm (half the cube side) and 21.57 mm (half the cube diagonal). RF-DROPO consistently outperforms all baselines across all thresholds, and is the only method that remains reliable under strict success criteria. In particular, it achieves 32% success at 15.25 mm and 56% at 21.57 mm, whereas the baselines remain near zero in the same regime. Even under the largest threshold of 30 mm, RF-DROPO reaches 60% success, substantially above the competing approaches.

These results indicate that the benefit of RF-DROPO is not only visible in simulation, but translates to substantially stronger zero-shot transfer on the physical platform. BayesSim shows limited but non-negligible transfer, while NPDR improves over the static baselines only at larger thresholds, suggesting less consistent real-world behavior. Overall, the results support the claim that offline simulator adaptation is particularly valuable for soft robots, where transfer is highly sensitive to mismatch in deformation and contact dynamics.

V. DISCUSSION

The main takeaway of this work is that simulator adaptation through Domain Randomization can be effective even in the difficult setting of soft robotics, where full-state reconstruction is unavailable and only a limited amount of target-domain data can be collected. In this regime, RF-DROPO provides a simple offline way to tune the DR parameters distribution before policy learning, rather than relying on hand-designed randomization ranges or repeated online adaptation on hardware.

The strongest evidence comes from the zero-shot sim-to-real experiment on the physical Trunk robot. The improvement over the baselines shows that better simulator adaptation translates into more reliable real-world behavior, especially in a contact-rich task where transfer is highly sensitive to model mismatch.

A current limitation is that the quality of the inferred distribution depends on how informative the offline trajectories are. Extending the approach to richer posterior families and to more diverse deformable manipulation settings is a promising direction for future work.

VI. CONCLUSION

We presented RF-DROPO, an offline Adaptive Domain Randomization method for soft-robot sim-to-real transfer under partial observability. By inferring a distribution over simulator parameters from a small amount of trajectory data without requiring arbitrary state resets, the method makes simulator adaptation feasible for deformable robotic systems. Experiments in simulation and on hardware indicate that this leads to more reliable transfer than static domain randomization and existing adaptive baselines.

REFERENCES

[1] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017.

[2] G. Tiboni, A. Protopapa, T. Tommasi, and G. Averta, "Domain randomization for robust, affordable and effective closed-loop control of soft robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 612–619, 2023.

[3] E. Coevoet, A. Escande, and C. Duriez, "Optimization-based inverse model of soft robots with contact handling," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1413–1419, 2017.

[4] E. Coevoet, T. Morales-Bieze, F. Largilliere, Z. Zhang, M. Thieffry, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, J. Dequidt, *et al.*, "Software toolkit for modeling, simulation, and control of soft robots," *Advanced Robotics*, vol. 31, no. 22, pp. 1208–1224, 2017.

[5] F. Ramos, R. C. Possas, and D. Fox, "BayesSim: Adaptive domain randomization via probabilistic inference for robotics simulators," in *Proceedings of Robotics: Science and Systems (RSS)*, Freiburg im Breisgau, Germany, 2019.

[6] F. Muratore, T. Gruner, F. Wiese, B. Belousov, M. Gienger, and J. Peters, "Neural posterior domain randomization," in *Proceedings of the Conference on Robot Learning (CoRL)*, pp. 1532–1542, 2022.