Improving Generalization of Differentiable Simulator Policies with Sharpness-Aware Optimization

Severin R. Bochem
ETH Zurich
sbochem@ethz.ch

Eduardo Gonzalez
ETH Zurich
gonzalez@iwf.mavt.ethz.ch

Yves Bicker
University of Zurich
yves.bicker@uzh.ch

Gabriele Fadini
ETH Zurich
gabriele.fadini@laas.fr

Abstract

This work contributes to the ongoing discussion on the trade-off between performance and generalization in reinforcement learning, particularly in the context of sim-to-real transfer in robotics. We investigate the generalization capabilities of policies learned using differentiable simulators in contact-rich robotic scenarios. While first-order optimization achieves a higher sample efficiency, it has been empirically shown to be unstable in loco-manipulation problems. We demonstrate that, while first-order methods achieve superior performance and sample efficiency in training, they exhibit less robustness to environmental variations. To address this limitation, we propose augmenting them with sharpness-aware optimization. Our simulation results show that this approach improves the generalization of learned policies over a larger magnitude of perturbation noise.

1 Introduction

Transfer learning from simulation to unknown environments has gained relevance in robotics as it allows for policy learning in simulation and subsequent transfer to real robots. This method bypasses the impractical process of collecting real-world demonstrations. However, the main challenges in sim-to-real transfer learning are the discrepancies between computer simulation and real-world robot dynamics [1, 2].

Reinforcement Learning (RL) has been successfully employed to learn robust control policies for robotic environments from simulation data [3, 4]. A major downside of RL is the large amount of training data it needs to approximate the policy gradient. In response to this challenge, differentiable simulators have emerged as a powerful tool for sample-efficient policy optimization in robotics. By providing analytic gradients of a policy's value function, they enable the use of first-order policy gradient (FoPG) methods [5]. These methods leverage the gradient information to update the policy parameters in the direction of the steepest ascent, leading to faster convergence and improved sample efficiency compared to zeroth-order methods and potentially lower computational cost.

However, the effectiveness of FoPG methods heavily relies on the quality and accuracy of the simulator gradients. In real-world robotics applications, such as locomotion and manipulation, the dynamics often involve complex contact interactions between objects. These contact dynamics are inherently non-differentiable [6, 7], posing challenges for differentiable simulators.

In this ongoing work, we demonstrate that while the first-order policy method Short Horizon Actor-Critic (SHAC) [8] is more sample-efficient and achieves better rewards than the zeroth-order method Proximal Policy Optimization (PPO) [9], it lacks generalization capabilities in noisy environments. By

D3S3: Data-driven and Differentiable Simulations, Surrogates, and Solvers @ NeurIPS 2024.

integrating SHAC with the Adaptive Sharpness-Aware Minimization (ASAM) [10], we significantly enhance its generalization capabilities while retaining its sample efficiency. Our findings suggest that this approach offers a robust solution for simulation-to-reality transfer in robotics, effectively addressing the limitations of existing methods.

2 Related Work

Differentiable simulators [11–14] have emerged as a promising tool in model-based reinforcement learning for sample-efficient policy optimization in robotics. Policy Optimization with Differentiable Simulation (PODS) [5] and SHAC leverage the analytical gradients provided by such simulators to improve sample efficiency compared to model-free reinforcement learning algorithms like PPO. SHAC, in particular, addresses the challenges of contact-rich dynamics by employing a truncated learning window and a critic function smoothing technique. Adaptive Horizon Actor-Critic (AHAC) [15] extends SHAC by dynamically truncating the optimization horizon based on contact information.

To enhance the generalization capabilities of learned policies, previous work relied on domain randomization [16] and domain adaption [17]. Independent of these methods, techniques such as Sharpness-Aware Minimization (SAM) [18] and its adaptive variant Adaptive Sharpness-Aware Minimization (ASAM) [10], which seek flatter minima during training, have shown promise in improving the generalization of deep learning models. However, to the best of our knowledge, none of the existing algorithms that leverage differentiable simulation gradients actively search for flat local minima, which have been shown to improve generalization in deep learning.

3 Motivation

A fundamental concept in deep learning is the relationship between the sharpness of local minima in the loss landscape and the generalization capabilities of neural networks [19]. Previous work in deep learning shows that noise in gradient updates can improve generalization [20]. It has also been demonstrated that in dynamic environments with highly deformable objects and fluids, differentiable simulators produce rugged loss landscapes [21]. These sharp local optima are especially prevalent in contact-rich environments which need to employ simplifications to ensure differentiability [22]. Policies learned from FoPG might get trapped into these sharp local optima rather than learning more robust and transferable strategies. In contrast, zeroth-order methods like PPO, which rely on noisy gradient estimates, may naturally avoid such sharp minima and avoid overfitting, hence converging to more generalizable solutions [23]. We hypothesize that the gradient information provided by differentiable simulators may inadvertently lead optimization algorithms toward sharp local minima in the policy space. This hypothesis is supported by the results shown in Fig. 1. We evaluate policy robustness by varying key simulation parameters: contact stiffness and damping . This creates environments that challenge policies beyond their training distribution. While SHAC achieves higher rewards in the original settings, PPO demonstrates a broader range of validity across varied parameters, suggesting greater robustness to environmental changes.

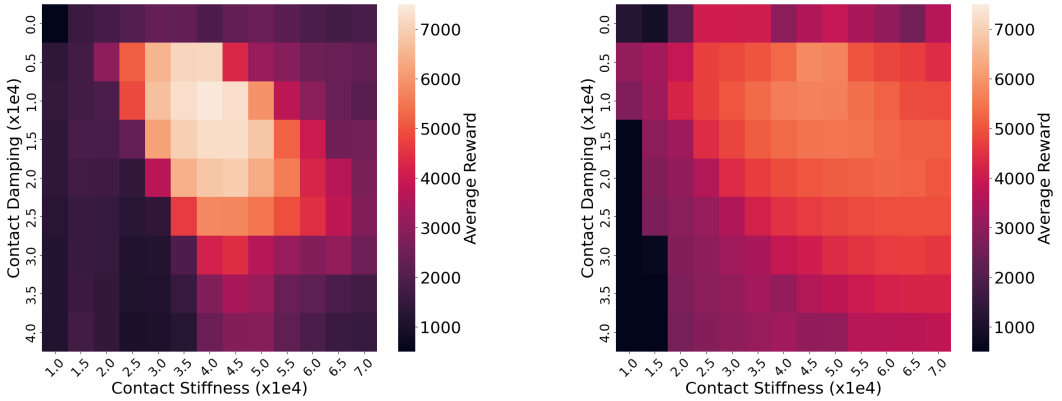

Figure 1: Reward heatmaps for SHAC (left) and PPO (right) policies under varying contact stiffness and damping in the Ant environment.

4 Method

In this work, we propose to incorporate the adaptive sharpness-aware optimization (ASAM) technique [10] into the short-horizon actor-critic (SHAC) algorithm [8] to improve the robustness of learned policies against local sharp minima in noisy environments. We focus on SHAC because of its well-documented, pytorch-based open-source implementations and promising performance results.

SHAC leverages analytical gradients from dFlex a differentiable simulator engine and addresses the challenges of contact-rich dynamics, long horizons, and sample inefficiency. In each learning episode, the algorithm samples N trajectories $\{\tau_i\}_{i=1}^{N}$ of short-horizons of length $h \ll H$ in parallel from the simulator, where H is the full task horizon. The SHAC policy loss is defined as:

$$L_{\mathcal{S}}(\theta) = -\frac{1}{Nh}\sum_{i=1}^{N}\left[\Big(\sum_{t=t_0}^{t_0+h-1}\gamma^{t-t_0}\mathcal{R}(\mathbf{s}_t^i,\mathbf{a}_t^i)\Big) + \gamma^h V_\phi(\mathbf{s}_{t_0+h}^i)\right], \tag{1}$$

where $\mathbf{s}_t^i$ and $\mathbf{a}_t^i \sim \mathcal{N}(\mu_\theta(\mathbf{s}_t^i), \sigma_\theta(\mathbf{s}_t^i))$, are the state and action at step t of the i-th trajectory, γ is the discount factor, and V_ϕ is the critic function. The mini-max optimization of ASAM is defined by:

$$\min_\theta L_{\mathcal{S}}^{ASAM}(\theta) + \lambda\|\theta\|_2^2 \quad \text{where} \quad L_{\mathcal{S}}^{ASAM}(\theta) \triangleq \max_{\|\mathbf{T}_\theta^{-1}\epsilon\|_p \le \rho} L_{\mathcal{S}}(\theta+\epsilon), \tag{2}$$

where $L_{\mathcal{S}}(\theta)$ is the SHAC policy loss. The inner maximization problem seeks a scale-invariant perturbation ϵ within a norm ball of radius ρ that maximizes the training loss. This identifies the worst-case loss within the local neighborhood, while the outer minimization adjusts the model parameters θ to minimize this worst-case loss, steering the optimization towards flatter minima.

The proposed SHAC-ASAM algorithm combines the sample efficiency of SHAC, with improved robustness provided by ASAM. This results in more stable policies in noisy environments, making ASAM-SHAC well-suited for learning robust policies in contact-rich, long-horizon tasks with limited sample budgets while potentially helping in improving sim-to-real transfer.

Algorithm 1 ASAM-SHAC Policy Learning

1: Initialize policy π_θ, value function V_ϕ, and target value function $V_{\phi_0} \leftarrow V_\phi$.
2: **for** learning episode $= 1, 2, \ldots, M$ **do**
3: Sample N short-horizon trajectories of length h by the parallel differentiable simulation from the final states of the previous trajectories.
4: Compute the SHAC policy loss L_{θ_t} defined in 1 from the sampled trajectories and V_{ϕ_0}.
5: Compute the analytical gradient $\nabla L(\theta_t)$.
6: Update the policy π_{θ_t} using ASAM:
   1. Determine the normalization operator T_θ^{-1} based on the model parameters, which adapts the size of the neighborhood during the optimization.
   2. Compute the perturbation ϵ_t and update the model parameters temporarily:

$$\epsilon_t = \rho\frac{T^2\theta_t\nabla L(\theta_t)}{\|T_{\theta_t}\nabla L(\theta_t)\|_2} \qquad \tilde{\theta}_t = \theta_t + \epsilon_t \tag{3}$$

   3. Update the model parameters using the gradient at the perturbed point:

$$\theta_{t+1} = \theta_t - \alpha_t(\nabla L(\tilde{\theta}_t) + \lambda\cdot\theta_{\mathbf{t}}) \tag{4}$$

7: Compute estimated values for all the states in sampled trajectories as in Eq. 7 from [8].
8: Fit the value function V_ϕ using the critic loss defined in Eq. 6 from [8].
9: Update target value function: $V_{\phi_0} \leftarrow \alpha V_{\phi_0} + (1-\alpha)V_\phi$.
10: **end for**

5 Experiments & Results

5.1 Noise Injection on Action Space

Figure 2 illustrates the average episode reward as a function of the noise strength injected into policy actions. Each algorithm is evaluated using three policies. To assess robustness, we add noise to each component of the action vector $\mathbf{a} = (a_1, ..., a_n) \in [-1, 1]^n$, with n number of actions. The noise injection process is described as follows:

- Clip each action component: $a_i = \text{clip}(a_i, -1, 1)$
- Generate noise vector: $\mathbf{n} = (n_1, ..., n_n)$, where $n_i \sim \mathcal{U}(-1, 1)$
- Combine action and noise: $\mathbf{a}' = (1 - \lambda)\mathbf{a} + \lambda\mathbf{n}$

Here, $\lambda \in [0, 0.5]$ controls the noise strength. We tested λ values ranging from 0 (no noise) to 0.5 (equal weight to the original action and noise). The results indicate that SHAC-ASAM significantly enhances robustness, outperforming the baseline SHAC and maintaining higher rewards than PPO up to $\lambda = 0.15$. Notably, SHAC-ASAM avoids the performance decline observed in SHAC at $\lambda \approx 0.05$.

The radius parameter ρ in SHAC-ASAM is crucial for balancing performance and generalization by adjusting the characteristics of the local optima. It determines the neighborhood size for the worst-case loss in the sharpness-aware optimization algorithm. Larger ρ values lead to flatter minima, enhancing generalization by reducing sensitivity to noise. Conversely, smaller ρ values revert to the original SHAC behavior, achieving higher peak performance but with reduced noise robustness. This relationship is illustrated in Figure 3. The results confirm our hypothesis that flatter minima improve generalization and demonstrate the trade-off between generalization and performance.

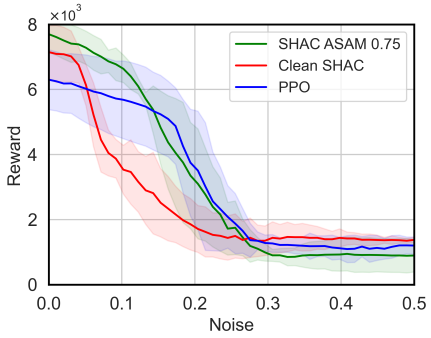

Figure 2: Evaluation of 3 SHAC, SHAC-ASAM ($\rho = 0.75$), and PPO policies in Ant with increasing action noise.

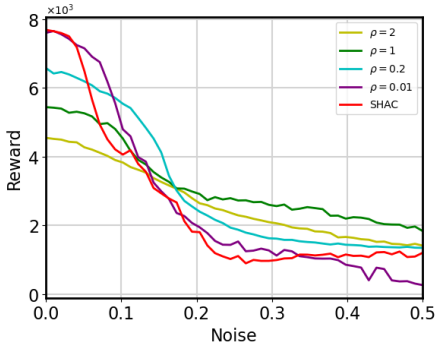

Figure 3: Reward vs Action Noise for policies trained with SHAC-SAM for different ρ values

6 Conclusion & Future Work

While first-order methods are more sample efficient, we demonstrated that they lead to policies that struggle to generalize to unseen environments. To address this, we integrated a sharpness-aware optimizer into SHAC. Our initial results indicate that this novel approach, SHAC-ASAM, can indeed enhance the generalization capabilities of the trained policy. To validate our preliminary findings, we plan to train policies across a broader range of environments and incorporate more diverse perturbation mechanisms. The key idea of applying ASAM to a differentiable simulator is generalizable, and similar improvements are expected with other first-order algorithms like AHAC or PODS.

In our ongoing work, we plan to implement ASAM for AHAC and benchmark our findings. Additionally, we aim to include a dynamic adjustment of the parameter ρ during training to automatically balance the trade-off between generalization and sample efficiency. To mitigate the additional computational cost of SHAC-ASAM, further investigation is needed, potentially leading to the development of more efficient and tailored methods for sharpness-aware optimization.

References

[1] Josh Tobin, Rachel Fong, Alex Ray, et al. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World, March 2017. URL http://arxiv.org/abs/1703.06907.

[2] Sehoon Ha, Joonho Lee, Michiel van de Panne, et al. Learning-based legged locomotion; state of the art and future perspectives, 2024. URL https://arxiv.org/abs/2406.01152.

[3] Jemin Hwangbo, Joonho Lee, Dosovitskiy, et al. Learning agile and dynamic motor skills for legged robots. *CoRR*, 2019. URL http://arxiv.org/abs/1901.08652.

[4] Marcin Andrychowicz, Bowen Baker, Maciek Chociej, et al. Learning dexterous in-hand manipulation, 2019. URL https://arxiv.org/abs/1808.00177.

[5] Miguel Angel Zamora Mora, Momchil Peychev, Sehoon Ha, et al. Pods: Policy optimization via differentiable simulation. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*. PMLR, 2021. URL https://proceedings.mlr.press/v139/mora21a.html.

[6] Quentin Le Lidec, Wilson Jallet, Louis Montaut, et al. Contact models in robotics: a comparative analysis, 2024. URL https://arxiv.org/abs/2304.06372.

[7] Aykut Ozgun Onol, Philip Long, and Taskin Padlr. A comparative analysis of contact models in trajectory optimization for manipulation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[8] Jie Xu, Viktor Makoviychuk, Yashraj Narang, et al. Accelerated policy learning with parallel differentiable simulation, 2022. URL https://arxiv.org/abs/2204.07137.

[9] John Schulman, Filip Wolski, Prafulla Dhariwal, et al. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

[10] Jungmin Kwon, Jeongseop Kim, Hyunseo Park, et al. ASAM: Adaptive Sharpness-Aware Minimization for Scale-Invariant Learning of Deep Neural Networks, 2021. URL http://arxiv.org/abs/2102.11600.

[11] C. Daniel Freeman, Erik Frey, et al. Brax – A Differentiable Physics Engine for Large Scale Rigid Body Simulation, 2021. URL http://arxiv.org/abs/2106.13281.

[12] Tao Du, Kui Wu, Pingchuan Ma, et al. Diffpd: Differentiable projective dynamics. *ACM Trans. Graph.*, 2021. URL https://doi.org/10.1145/3490168.

[13] Taylor A. Howell, Simon Le Cleac'h, Jan Brüdigam, et al. Dojo: A differentiable physics engine for robotics, 2023. URL https://arxiv.org/abs/2203.00806.

[14] Moritz Geilinger, David Hahn, Jonas Zehnder, Bächer, et al. ADD: Analytically differentiable dynamics for multi-body systems with frictional contact, 2020. URL http://arxiv.org/abs/2007.00987.

[15] Ignat Georgiev, Krishnan Srinivasan, Jie Xu, et al. Adaptive horizon actor-critic for policy learning in contact-rich differentiable simulation, 2024. URL https://arxiv.org/abs/2405.17784.

[16] Ken Caluwaerts, Atil Iscen, J. Chase Kew, et al. Barkour: Benchmarking animal-level agility with quadruped robots, 2023. URL https://arxiv.org/abs/2305.14654.

[17] Fabio Muratore, Christian Eilers, Michael Gienger, et al. Data-efficient domain randomization with bayesian optimization. *IEEE Robotics and Automation Letters*, 2021.

[18] Pierre Foret, Ariel Kleiner, Hossein Mobahi, et al. Sharpness-Aware Minimization for Efficiently Improving Generalization, 2021. URL http://arxiv.org/abs/2010.01412.

[19] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 1997. URL https://doi.org/10.1162/neco.1997.9.8.1735.

[20] Sam Smith and Quoc V. Le. A bayesian perspective on generalization and stochastic gradient descent. 2018. URL https://openreview.net/pdf?id=BJij4yg0Z.

[21] Rika Antonova, Jingyun Yang, Krishna Murthy, et al. Rethinking optimization with differentiable simulation from a global perspective, 2022. URL https://arxiv.org/abs/2207.00167.

[22] H. J. Terry Suh, Max Simchowitz, Kaiqing Zhang, et al. Do differentiable simulators give better policy gradients?, 2022. URL https://arxiv.org/abs/2202.00817.

[23] Yurii Nesterov and Vladimir G. Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 2015. URL https://api.semanticscholar.org/CorpusID:2147817.