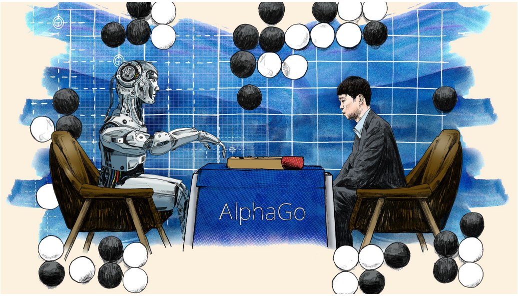
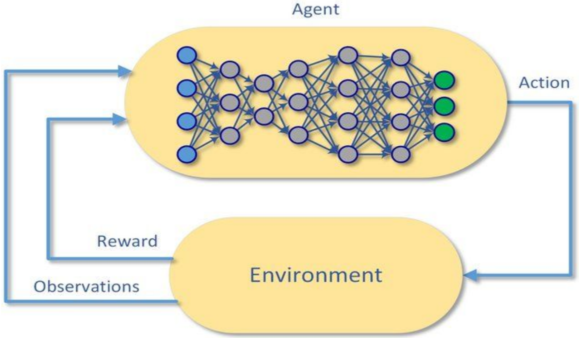
COUNTEREXAMPLE GUIDED RL POLICY REFINEMENT USING BAYESIAN OPTIMIZATION

AUTHORS: BRITI GANGOPADHYAY, Prof. PALLAB DASGUPTA
DEPT. OF COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

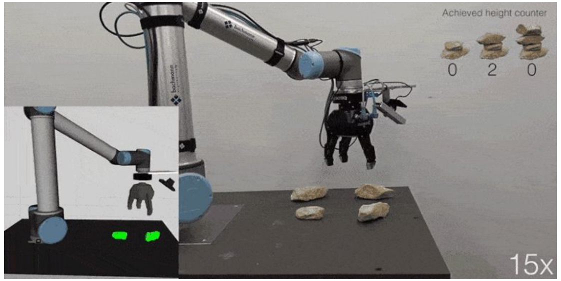
INTRODUCTION



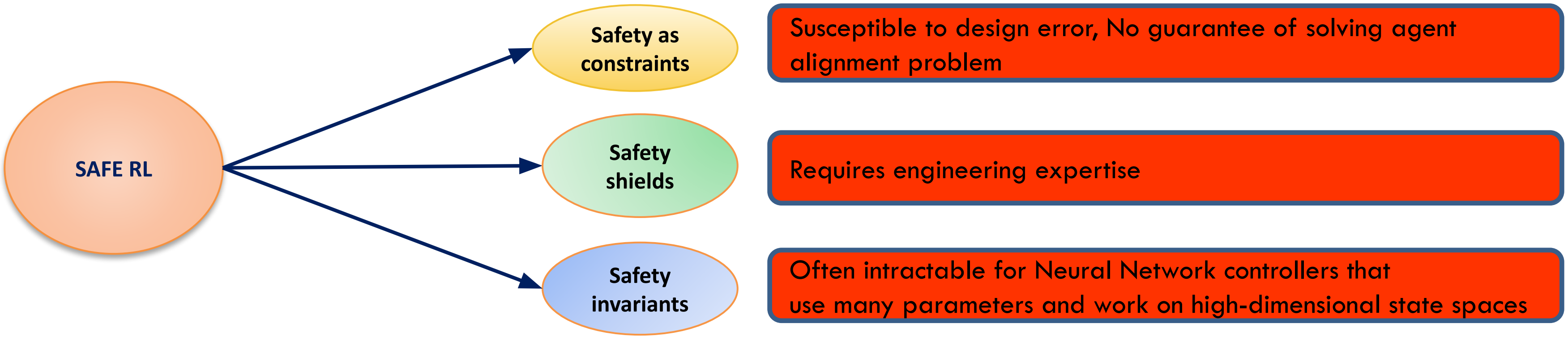


Non interpretable, Reward may not induce safe behaviour, Failures are rare and do not contribute significantly to reduce the reward

Uncertain, Parameters may change after deployment



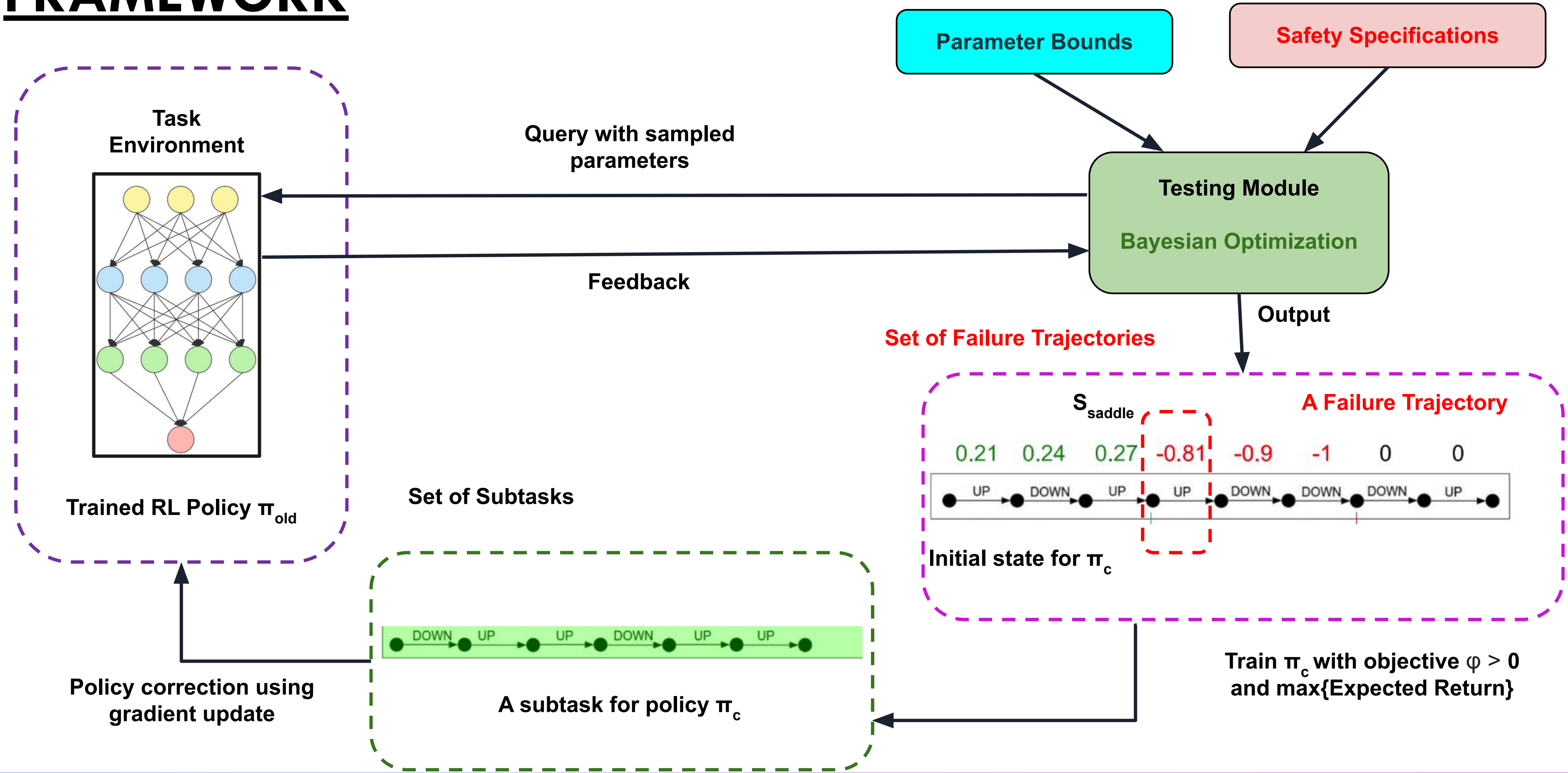
PROBLEM STATEMENT



1. Given a policy π_{old}, learnt from optimizing reward in a given environment, test it against parameters with uncertainties and a set of objective functions φ derived from the negation of the given safety criteria.

2. Using the failure trajectories selectively do a gradient update on π_{old} to construct a new policy π_{new}, that excludes the counterexample traces under the given domain uncertainties

FRAMEWORK



Parameter Bounds
Safety Specifications
Task Environment
Query with sampled parameters
Testing Module
Bayesian Optimization
Feedback
Output
Set of Failure Trajectories
Trained RL Policy π_{old}
S_{saddle}
A Failure Trajectory
0.21 0.24 0.27 -0.81 -0.9 -1 0 0
UP DOWN UP UP DOWN DOWN DOWN UP
Initial state for π_c
Set of Subtasks
DOWN UP UP DOWN UP UP
A subtask for policy π_c
Policy correction using gradient update
Train π_c with objective $\varphi > 0$ and max{Expected Return}

FINDING COUNTER-EXAMPLE TRACES

Safety Specification : The lander cannot be tilted at an angle while being close to the ground.

Coordinates : (l_x, l_y) $0 \leq (l_x, l_y) \leq 10$ Angle : (l_{angle}) $-1 \leq l_{angle} \leq 0$

$$l_y < 5 \rightarrow l_{angle} \geq -0.5$$

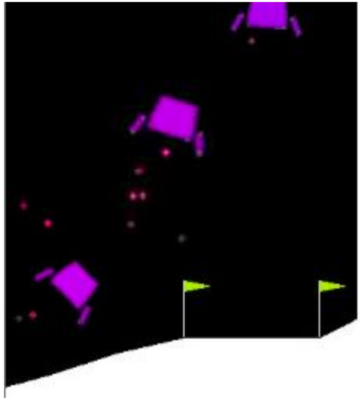
Negation of the Specification φ : $\neg(l_y < 5 \rightarrow l_{angle} \geq -0.5) \equiv \neg(\neg(l_y < 5) \vee (l_{angle} \geq -0.5))$

$$\equiv (l_y < 5) \wedge (l_{angle} < -0.5)$$

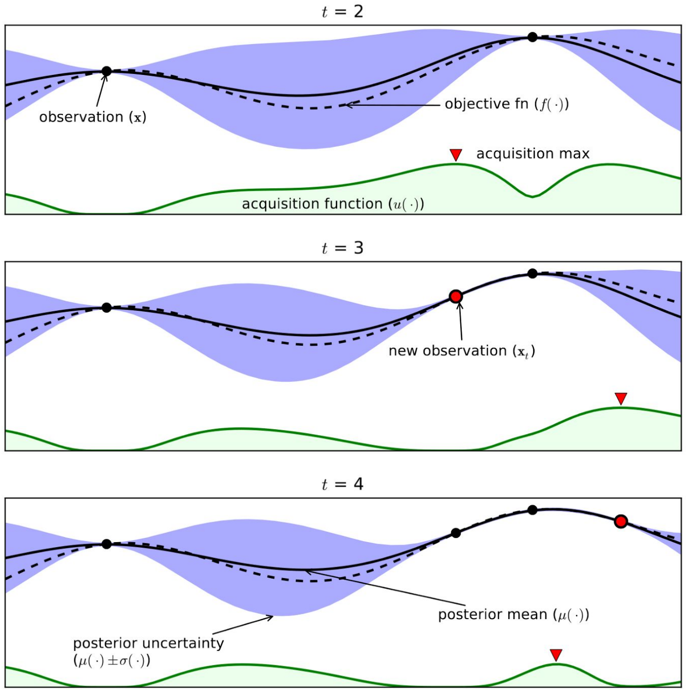
$$\mu_1 : l_y - 5 < 0 \qquad \mu_2 : l_{angle} + 0.5 < 0$$

Optimization Objective : $\min(\mu_1 + \mu_2)$

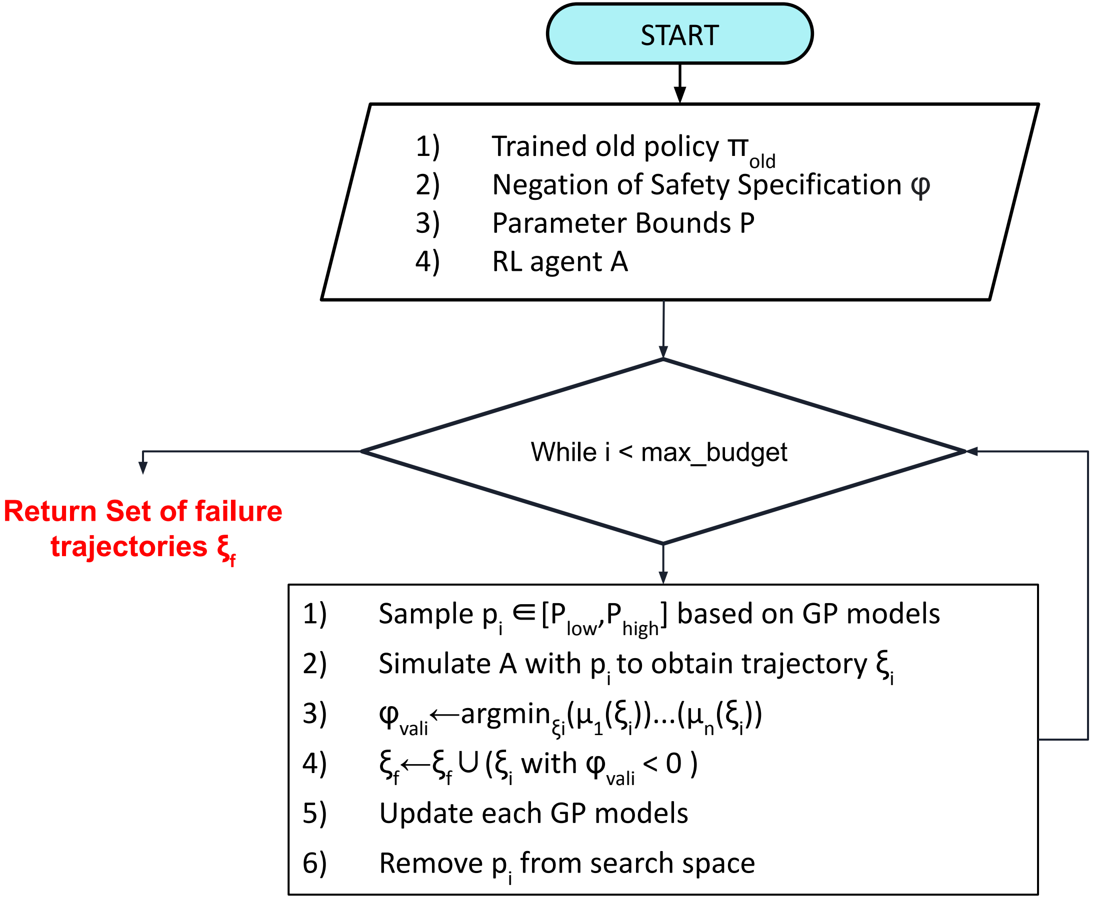
Counterexample : $l_y = 1$ and $l_{angle} = -0.8$



FINDING COUNTER-EXAMPLE TRACES



Bayesian Optimization



PROXIMAL POLICY OPTIMIZATION OVERVIEW

Current policy that we want to refine : $\pi_\theta(a_t|s_t)$

Policy that we last used to collect samples : $\pi_{\theta_{old}}(a_t|s_t)$

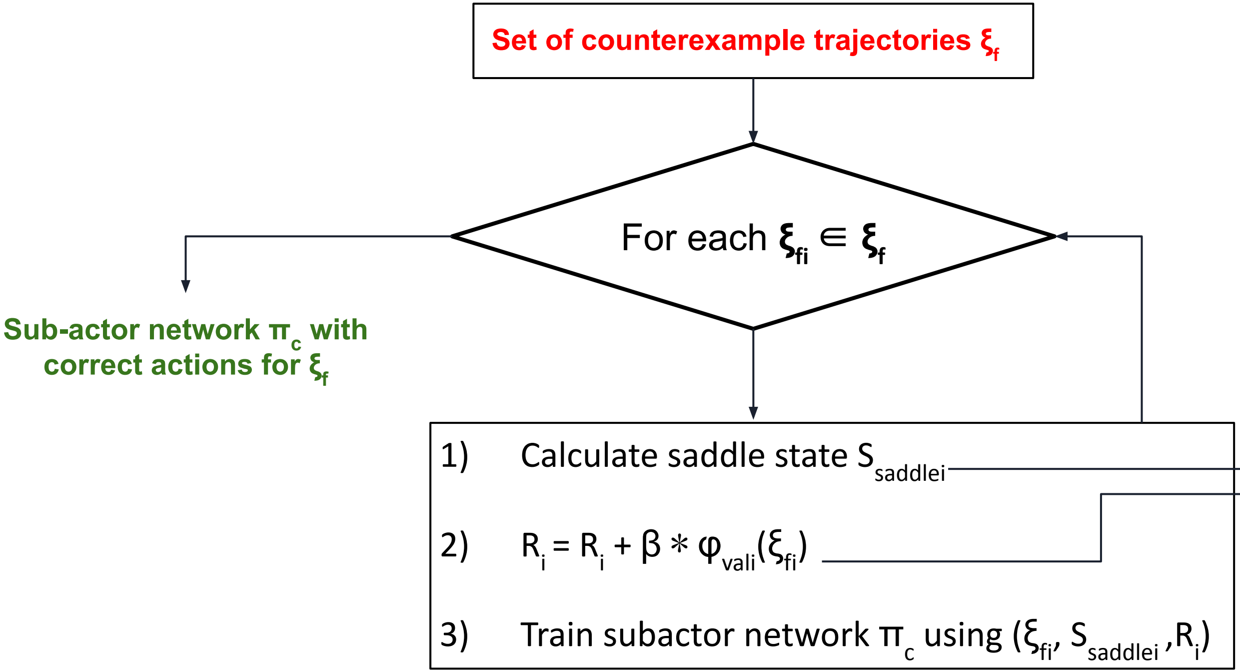
Evaluate a new policy with samples collected from an older policy : $\underset{\theta}{\text{maximize}} \quad \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}\hat{A}_t\right]$

+ve advantage will make that action more likely in the future, for that state.
-ve advantage will make that action less likely in the future, for that state.

Objective Ratio : $r_t(\theta) = \dfrac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$

Clipped Objective : $\mathcal{L}^{CLIP}_{\theta_k}(\theta) = \underset{\tau\sim\pi_k}{\mathrm{E}}\left[\sum_{t=0}^{T}\left[\min(r_t(\theta)\hat{A}_t^{\pi_k}, \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t^{\pi_k})\right]\right]$

POLICY REFINEMENT METHODOLOGY



The first state on ξ_{fi} where the old actor network violates the safety specification

Updated Reward with negative penalty. Penalty depends on margin of violation.

For example for lunar lander $-0.4 < l_x < 0.4$. If the lander lands at 0.8 then the reward for the trajectory is

$200 + 50*(0.4 - 0.8) = 180$,

Where 200 is the original reward and $\beta = 50$ (penalty per unit deviation)

POLICY REFINEMENT METHODOLOGY

Clipped Objective Ratio : $r_t(\theta) = \dfrac{\pi_{old}(a_t|s_t)}{\pi_c(a_t|s_t)}$

Advantage A_t = 1

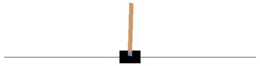
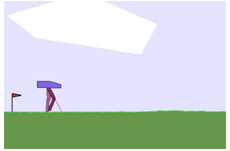
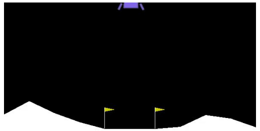
Since, these corrected trajectories are to be enforced into π_{old} we set the advantage factor A_t to be 1.

Update π_{old} to π_{new} by maximizing the PPO clip objective using π_c

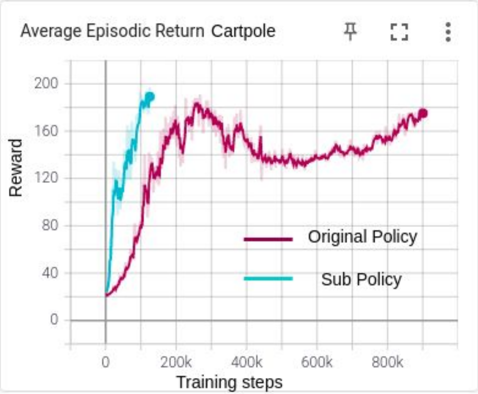
Variation Distance between two policies:

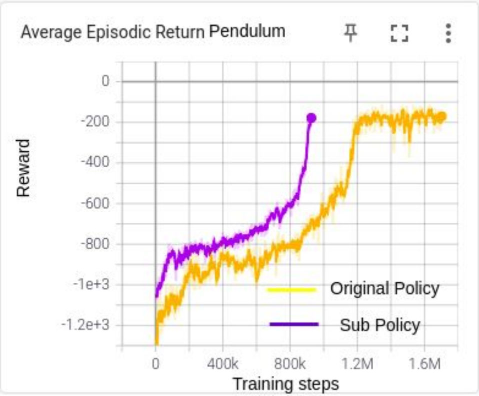
$$D_v(\pi_{old}\,||\,\pi_{new}) = \frac{1}{n}\sum_{\xi_i,\xi_i'\in\xi}\sqrt{\sum_{s_i\in\xi_i,s_i'\in\xi_i'}|(s_i)_{\pi_{old}} - (s_i')_{\pi_{new}}|^2}$$

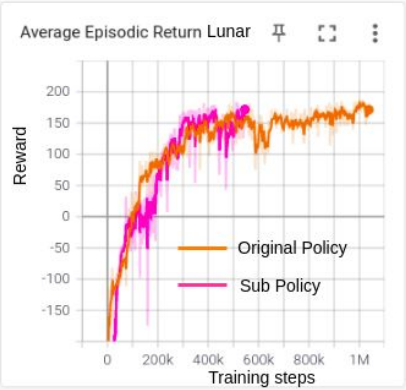
EMPIRICAL STUDIES

Environment	Safety Criteria	Parameter Bounds	Failures	Distance
Cart-pole-v0 	1. -2.4 < position < 2.4 2. -2.0 < momentum < 2.0 3. angle > 0.2	State : [(-0.05, 0.05)] * 4 Mass : (0.05, 0.15) Length of pole : (0.4, 0.6) force magnitude: (8.00, 12.00)	174.4 ± 0.51	1.255 ± 0.195
Pendulum-v0 	1. Reward > -300	θ: $(-\pi,\pi)$ $\dot{\theta}$: (0,1) speed: (-1,1)	80.1 ± 1.85	10.866 ± 1.379
BipedalWalker-v3 	1. Hull Position > 0 2. -0.8 < Hull Angle < 2	Hull angle : $(0,2*\pi)$ Velocity x: (-1,1) Velocity y: (-1,1)	40.6 ± 4.08	11.189 ± 1.375
LunarLanderContinuous-v2 	1. $-0.4 < Landing_{Positionx} < 0.4$ 2. $Pos_y < 0.1 \rightarrow (angle > -1 \vee angle < 1)$ 3. Reward > 0	$x\delta$: (0,10) $y\delta$: (0,20) $velx\delta$: (0,3) $vely\delta$: (0,3)	40.85 ± 5.14	2.215 ± 0.282

REWARD PLOTS

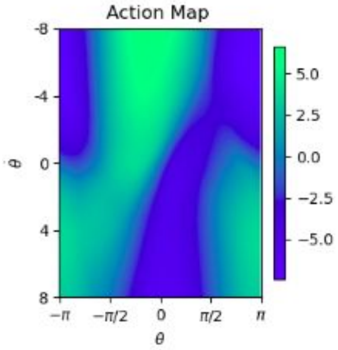




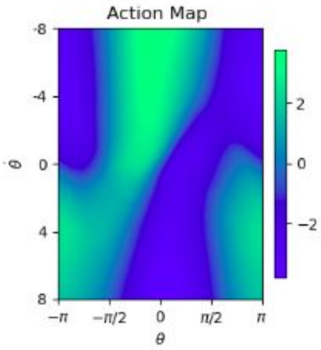


Random Observations

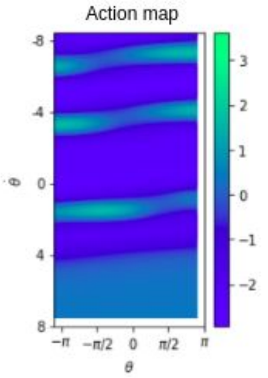
Failure Trajectory Observations



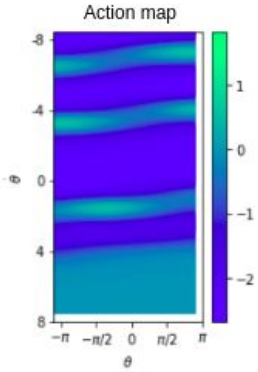
Original Pendulum Policy



Updated Pendulum Policy



Original Pendulum Policy



Updated Pendulum Policy

COMPARISON WITH BASELINES

A) PPO policy trained from scratch with negative penalty for property violation,

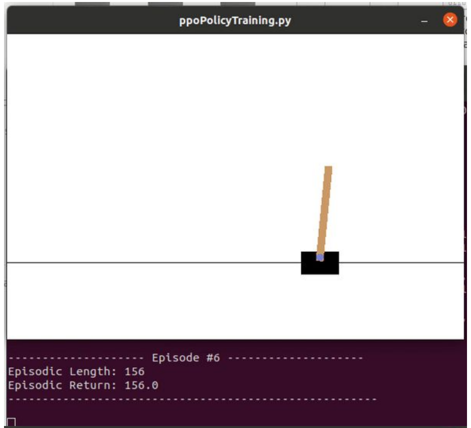
B) PPO policy trained from scratch with only counterexample traces and negative penalty after one iteration of testing with BO same as π_c

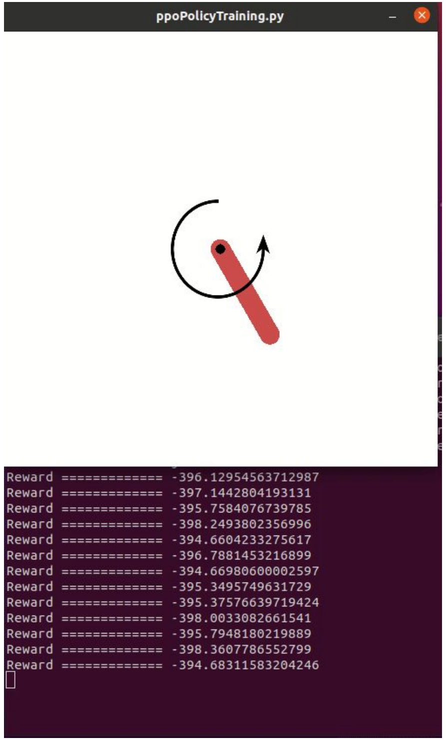
C) PPO policy trained from scratch with original training traces, counterexample traces and negative penalty after testing with one iteration of BO, and

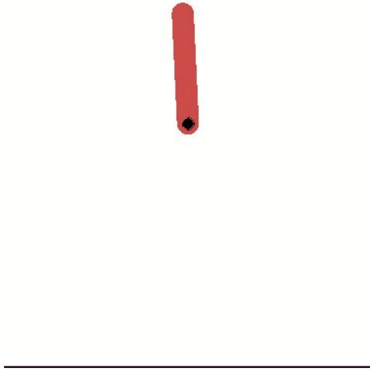
D) The refined policy π_{new}

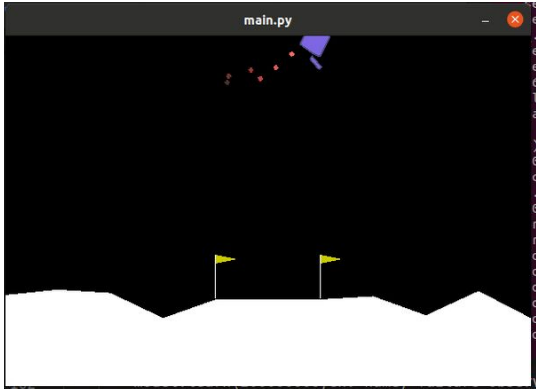
Environment	Policy A	Policy B	Policy C	Policy D
Cart-pole-v0	Failures: 179 Training Steps: 900K	Failures: 52 Training Steps: 150K	Failures: 0 Training Steps: 1M	Failures: 0 Training Steps: 150K+ 80K (Update)
Pendulum-v0	Failures: 89 Training Steps: 1.6M	Failures: 102 Training Steps: 850K	Failures: 0 Training Steps: 1.8M	Failures: 0 Training Steps: 850K+ 20K (Update)
BipedalWalker-v3	Failures: 45 Training Steps: 7.5M	Failures: 145 Training Steps: 2.8M	Failures: 41 Training Steps: 8M	Failures: 0 Training Steps: 2.8M+ 20K (Update)
LunarLanderContinuous-v2	Failures: 42 Training Steps: 1.1M	Failures: 18 Training Steps: 400K	Failures: 5 Training Steps: 1.2M	Failures: 0 Training Steps: 400K+ 20K (Update)

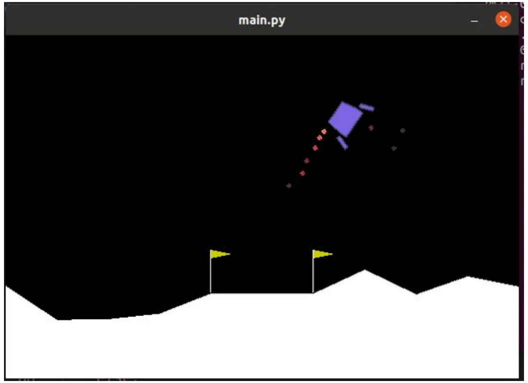
EXAMPLES OF FAILURES AND CORRECTIONS











REFERENCES

1. Javier García, Fern, and o Fernández. A comprehensive survey on safe reinforcement learning. Journal of Machine Learning Research, 16(42):1437–1480, 2015. URL http://jmlr.org/papers/ v16/garcia15a.html.
2. Rajeev Alur, Thao Dang, and Franjo Ivanciˇ c. Counterexample-guided predicate abstraction of ´ hybrid systems. Theoretical Computer Science, 354(2):250–271, 2006. ISSN 0304-3975. Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2003).
3. Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. arXiv preprint arXiv:1606.01540, 2016.
4. Steven Carr, Nils Jansen, Ralf Wimmer, Alexandru Serban, Bernd Becker, and Ufuk Topcu. Counterexample-guided strategy improvement for pomdps using recurrent neural networks. In IJCAI, pages 5532–5539, 08 2019. doi: 10.24963/ijcai.2019/768.
5. Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerík, T. Hester, Cosmin Paduraru, and Y. Tassa. Safe exploration in continuous action spaces. ArXiv, abs/1801.08757, 2018.
6. B. Gangopadhyay, S. Khastgir, S. Dey, P. Dasgupta, G. Montana, and P. Jennings. Identification of test cases for automated driving systems using bayesian optimization. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 1961–1967, 2019. doi: 10.1109/ITSC.2019. 8917103.
7. John Schulman, F. Wolski, Prafulla Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. ArXiv, abs/1707.06347, 2017.
8. Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2, NIPS'12, page 2951–2959, Red Hook, NY, USA, 2012. Curran Associates Inc.
9. Weichao Zhou and Wenchao Li. Safety-aware apprenticeship learning. In Hana Chockler and Georg Weissenbacher, editors, Computer Aided Verification, pages 662–680, Cham, 2018. Springer International Publishing
10. S. Ghosh, F. Berkenkamp, G. Ranade, S. Qadeer, and A. Kapoor. Verifying controllers against adversarial examples with bayesian optimization. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 7306–7313, 2018. doi: 10.1109/ICRA.2018.8460635.
11. Briti Gangopadhyay, Harshit Soora, and Pallab Dasgupta. Hierarchical program-triggered reinforcement learning agents for automated driving. IEEE Transactions on Intelligent Transportation Systems, pages 1–10, 2021. doi: 10.1109/TITS.2021.3096998.


THANK YOU!