Harnessing Adversarial Attacks to Improve Robustness of Deep Reinforcement Learning

Anay Pattanaik, Zhenyi Tang*, Shuijing Liu*, and Girish Chowdhary
* Equal contribution
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Objectives

Robustness of Reinforcement Learning (RL) is critical for real world applications. We first design adversarial attacks on Deep Reinforcement algorithms (DRL) and then harness them to improve robustness of DRL.

Background and State of Art

- **Q Learning (Q):** Q learning is a value function based algorithm.
  - The learning agent updates the Q value using the temporal difference error and simultaneously acts to maximize its long run return.
  - In the deep Q learning algorithm, the agent uses a Deep Neural Network (DNN) to approximate this Q function, while in Radial Basis Function(RBF)-Based Q learning, RBF approximators are used . For deep learning, the relevant equations are [1]

$$Q^*(s,a) = \mathbb{E}_{s'\sim\xi}\left[r + \gamma \max_{a'} Q^*(s',a')|s,a\right]$$

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{s,a\sim p(.);s'\sim\xi}\left[\left(r + \gamma \max_{a'} Q(s',a';\theta_{i-1}) - Q(s',a';\theta_i)\right)\nabla_{\theta_i} Q(s,a;\theta_i)\right]$$

- **Actor-critic methods:** It uses both policy gradient method as well as value function network [2]
  - The action is explicitly expressed by an actor network, and a critic network is used to evaluate the value function. The agent simultaneously updates the actor and critic as it acts in real world.
  - Underlying function approximator can be DNN or (RBF). For Deep Deterministic Policy Gradient (DDPG) [3], the update for actor is

$$\nabla_{\theta^\mu} J \approx E_{s_t\sim\rho^\beta}\left[\nabla_a Q\left(s,a|\theta^Q\right)\Big|_{s=s_t,a=\mu(s_t)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s=s_t}\right]$$

- **Adversarial Examples:**
  - [4] have fooled Neural Networks into predicting incorrect labesl with high confidence through small perturbations of the input images.
  - [5] have used strategy similar to [4] for attacking RL algorithms that use image input as observation.
  - [6] Inject noise only when value function is above a certain threshold.
- **Robust Reinforcement Learning:**
  - [7] Sample an ensemble of different models and train on them. Needs runs outside the "simulator".
  - [8] Used a heuristic $||u||_2$ as objective for adversary
  - [9] Both adversary and RL agent learn alternatively
  - [10] Deep version of [9] using TRPO

Methodology

- **Objective function for adversarial attack:**
  - *DDQN:* The cross entropy loss between the adversarial probability distribution and optimal policy generated by the RL agent

$$J(s,\pi^*) = -\sum_{i=1}^{n} p_i log \pi_i^*$$

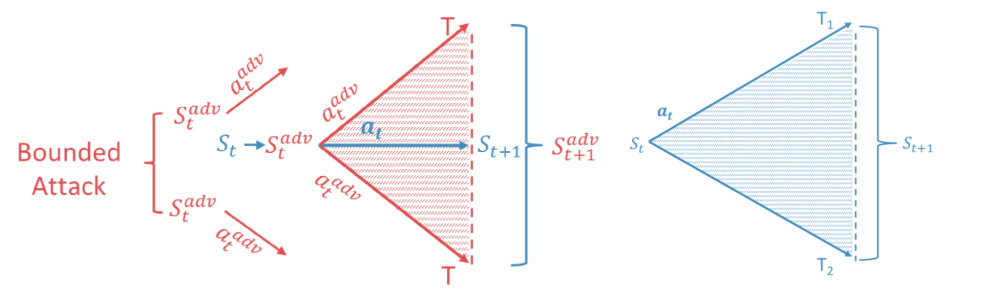
where $\pi_i^* = \pi^*(a_i|s)$, $p_i = P(a_i)$, the adversarial probability distribution P is given by

$$P(a_i) = \begin{cases} 1, & \text{if } a_w = 1 \\ 0, & \text{otherwise} \end{cases} \qquad (2)$$

  - *DDPG:*

$$\nabla_s Q^*(s,a) = \frac{\partial Q^*}{\partial s} + \frac{\partial Q^*}{\partial U^*}\frac{\partial U^*}{\partial s}$$

- **Adversarial Attack:**
  - Naive sampling: Sample noise from nearby states and use the worst possible noise
  - Gradient based: Sample noise along gradient of the objective function proposed above and return the worst possible noise
  - Stochastic gradient descent based: Use stochastic gradient descent (SGD) for the proposed objective function
- **Robust RL through adversarial training:**
  - Take pre-trained network
  - Train again, fool the agent through corrupted state that forces it to take "bad" action



Adversarial training and robustness over transitions

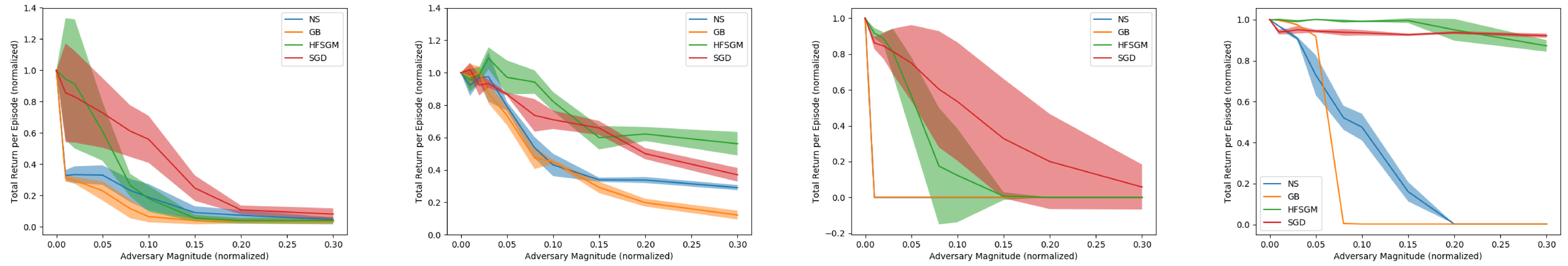
$$\eta(\pi,T) = \mathbb{E}_\tau[\sum_{t=0}^{T}\gamma^t r(s_t,a_t)|s_0,\pi,T]$$

$$\eta(\pi) = \mathbb{E}_T[\eta(\pi,T)]$$

$$\eta_{RC}(\pi) = \mathbb{E}_T[\eta(\pi,T)|\mathbb{P}(\eta(\pi,T)\leq\beta)=\alpha] \ [11]$$

Results

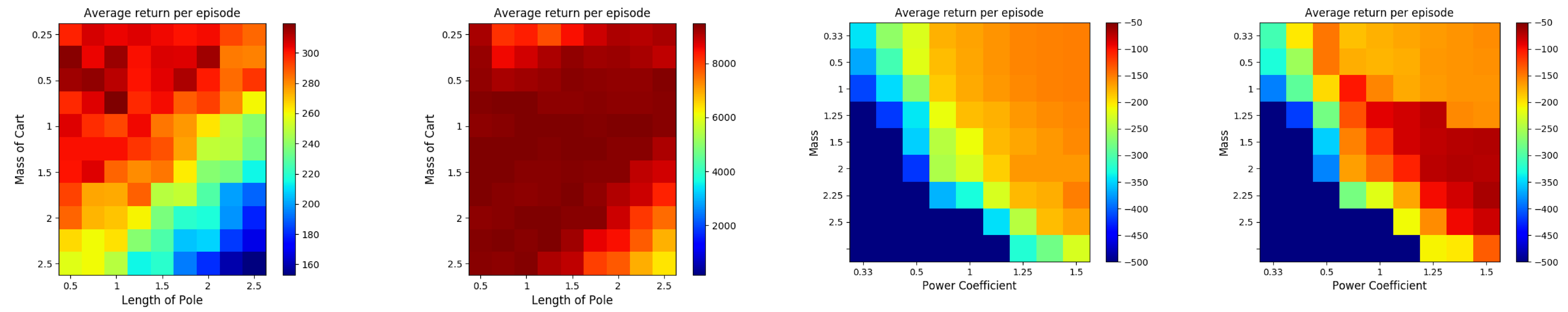
- **Adversarial Attack**



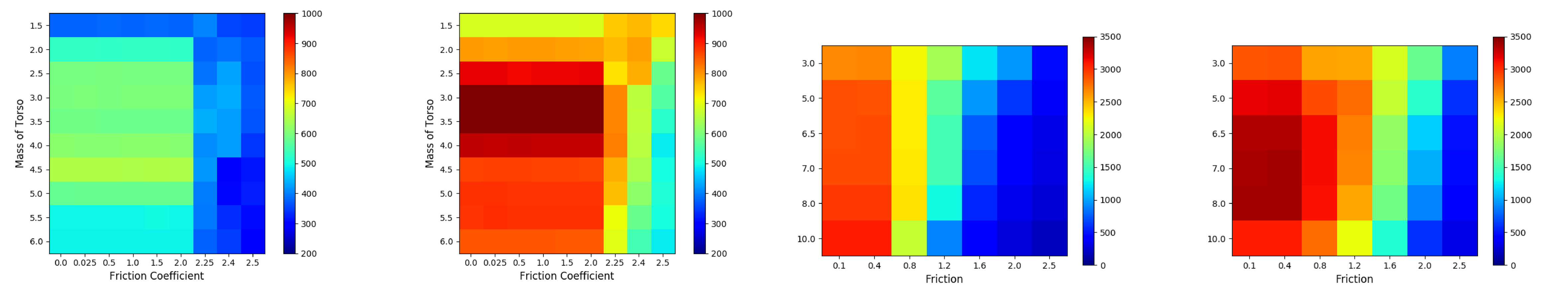
(a) DDQN Cart Pole (b) RBF Q Cart Pole (c)DDQN Mountain Car (d) RBF Q Mountain Car

Gradient based (GB) is better than naive sampling (NS) which in turn is better than HSFGM ([5]) and SGD

- **Robust RL**



(a) DDQN Cartpole (b) Robust DDQN Cartpole (a) DDQN Mountain Car (b) Robust DDQN Mountain-Car



(a) DDPG Hopper (b) Robust DDPG hopper (a) DDPG Half-Cheetah (b) Robust DDPG Half-Cheetah

Significant improvement of robustness over a range of model parameters because of robust training.

Conclusion and Acknowledgement

+ Proposed adversarial attacks and harnessed them for robust RL
+ Deep Neural Network based RL is more susceptible adversarial attacks as compared to RBF based RL
+ Future work involves establishing theoretical relationship between robustness and adversarial training.
+ This work was sponsored in part by AFOSR FA9550-15-1-0146 and AFOSR FA9550-14-1-0399

References

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
[2] Ivo Grondman, Lucian Busoniu, Gabriel AD Lopes, and Robert Babuska. A survey of actor-critic reinforcement learning: Standard and natural policy gradients. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1291–1307, 2012.
[3] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.
[4] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
[5] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017.
[6] Jernej Kos and Dawn Song. Delving into adversarial attacks on deep policies. *arXiv preprint arXiv:1705.06452*, 2017.
[7] Aravind Rajeswaran, Sarvjeet Ghotra, Balaraman Ravindran, and Sergey Levine. Epopt: Learning robust neural network policies using model ensembles. *arXiv preprint arXiv:1610.01283*, 2016.
[8] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Fei-Fei Li, and Silvio Savarese. Adversarially robust policy learning: Active construction of physically-plausible perturbations. *IEEE International Conference on Intelligent Robots and Systems (to appear)*, 2017.
[9] Jun Morimoto and Kenji Doya. Robust reinforcement learning. *Neural computation*, 17(2):335–359, 2005.
[10] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. *arXiv preprint arXiv:1703.02702*, 2017.
[11] Aviv Tamar, Yonatan Glassner, and Shie Mannor. Optimizing the cvar via sampling. In *AAAI*, pages 2993–2999, 2015.