Re-Evolutionary Algorithms

Devang Agrawal (devang18) , Kaushik Ram Sadagopan (kaushik7), Fatma Tlili (ftlili)

Introduction

Motivation:
Policy gradient methods in reinforcement learning face the issue of lack of exploration. Evolutionary strategies is a black-box optimization algorithm to overcome local optima which suffer from low exploitation of the environment feedback signals.

Objective:
Our project is aimed at developing a hybrid evolutionary reinforcement learning algorithm (ERL) and apply it to a classic control problem to prove its superiority over the standalone algorithms. We implement a policy gradient algorithm (Advantage Actor Critic - A2C) and an evolutionary algorithm (ES) for the cartpole problem on OpenAI gym. Subsequently, we combine A2C with ES for the cartpole problem to show that it performs better than the standalone algorithms.

Environment:
A pole is attached to a cart which moves along a frictionless track. The system is controlled by moving the cart right or left. The pole starts upright and the goal is to prevent it from falling over. The objective of this task is to keep the cartpole upright continuously for 200 timesteps which corresponds to a reward of 200.

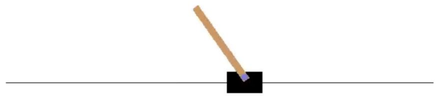
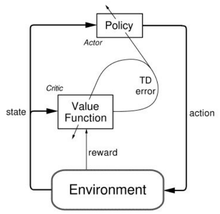
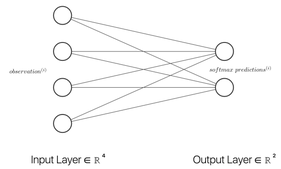


Figure 1: Cartpole Problem on OpenAI Gym

The Actor-Critic Architecture

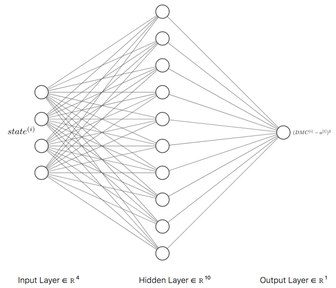


Schematic of the actor-critic setup



Policy Gradient architecture which outputs predictions of each action

The neural network architecture for the policy gradient and the value gradient functions are described. The policy gradient outputs a probability distribution for the policy from which actions are sampled and hence, the **actor**. The value gradient computes the advantage of taking a particular action given an observed state and hence, the **critic**.



Value Gradient architecture using a single hidden layer

Results & Conclusions

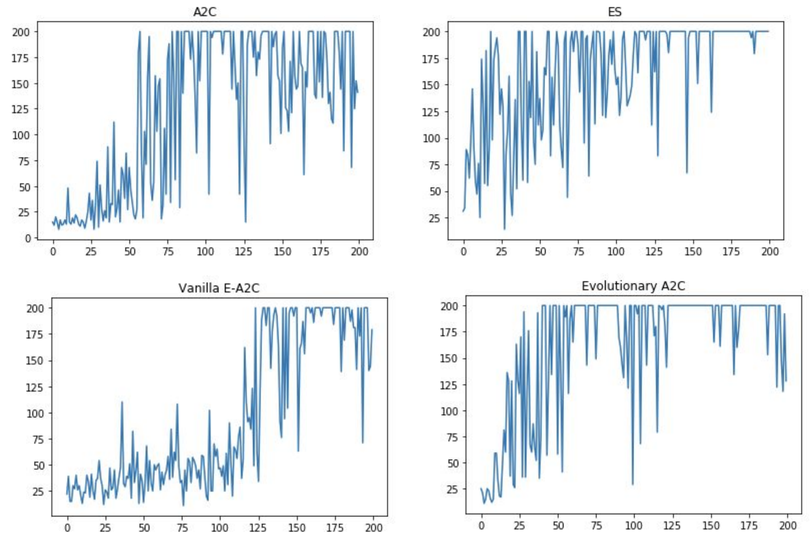


Figure 6: Plots of Rewards obtained in each epoch for A2C, ES, Vanilla E-A2C and Evolutionary A2C algorithms respectively

Observations: The training converges when the average reward reaches 200 consistently. A2C reaches this state at around 75 epochs (5 episodes each) but it has a lot of variation due to the stochasticity in the selection of actions. ES has a lot of variation at the beginning but stabilizes after 150 epochs. Vanilla E-A2C reaches this state after 125 epochs. The evolutionary A2C is clearly superior to the other three algorithms in the sense that it is the quickest to converge and the variations in the reward are minimal after reaching this state.

Evolutionary Strategies

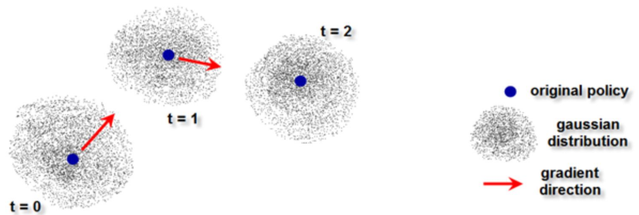
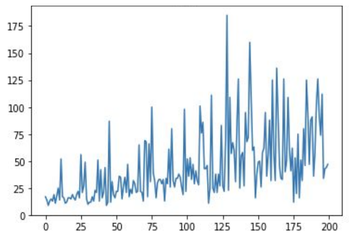


Figure 2: Evolutionary Strategies
ES spawns moving the parameters to a global optimum.

Weighted Combination of the Candidate Parameters



Maximum Candidate Parameters

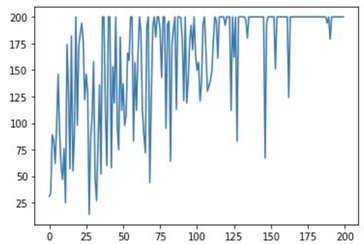
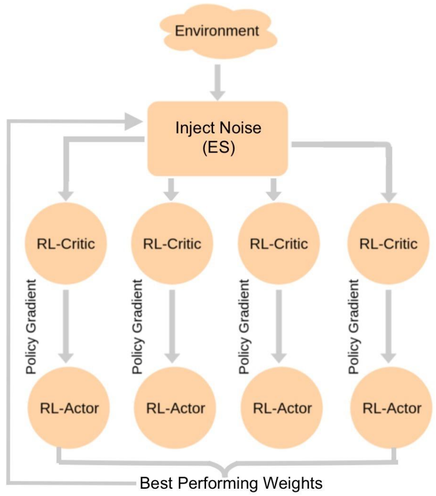


Figure 5: In the first plot the candidate parameters of the population are weighted by their corresponding rewards. In the second plot only the candidate which corresponds to the maximum reward is chosen. The plots shows that taking the best candidate parameters performs better than the weighted combination of the candidates.

Evolutionary Actor-Critic

Figure 3: Vanilla E-A2C
We combine ES and A2C iteratively in a sequence. Each iteration of algorithm spawns parameters and makes an update by choosing the best candidate. The weights updated by the ES is passed on to the policy gradient function of the A2C algorithm which performs a gradient descent update.



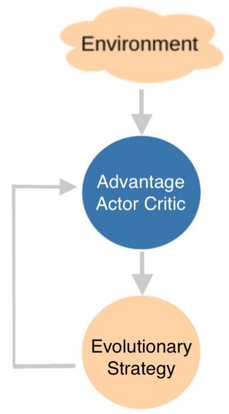
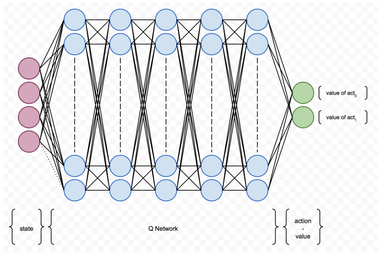
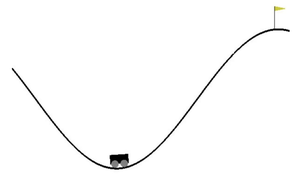


Figure 4: Evolutionary A2C
ES spawns a population of parameters and A2C updates each member of the population by performing a series of gradient descent updates. Finally ES chooses the best parameter vector (based on the rewards obtained) in the population, and injects noise onto this parameter vector to generate the new population.

Future Work





Implement evolutionary DQN for the mountain car environment to show the exploration capability enhanced by ES

References

[1] Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. 1998.
[2] Khadka, Shauharda, and Kagan Tumer. "Evolutionary Reinforcement Learning." *arXiv preprint arXiv:1805.07917*(2018)
[3] Houthooft, Rein, et al. "Evolved policy gradients." *arXiv preprint arXiv:1802.04821* (2018).