Novelty-Guided Proximal Curriculum Learning

Jan Malte Töpperwien

16.09.2024

- need lots of exploration
- sparse rewards -> infrequent learning signal
- curriculum learning for appriorate task difficulty

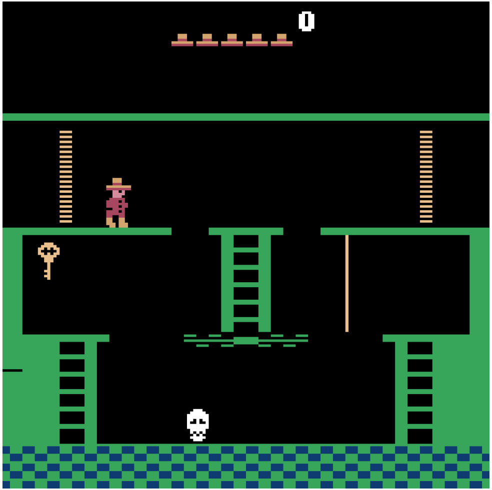


Figure: https://images.openai.com/blob/2c736f64-38dc-4c65-a1dd-aabe2ceb8ddf/learning-montezumas-revenge-from-a-single-demonstration.png

- need lots of exploration
- sparse rewards -> infrequent learning signal
- curriculum learning for appriориate task difficulty
- *How to set proper tasks?*
- current approaches often rigid and demonstration-based

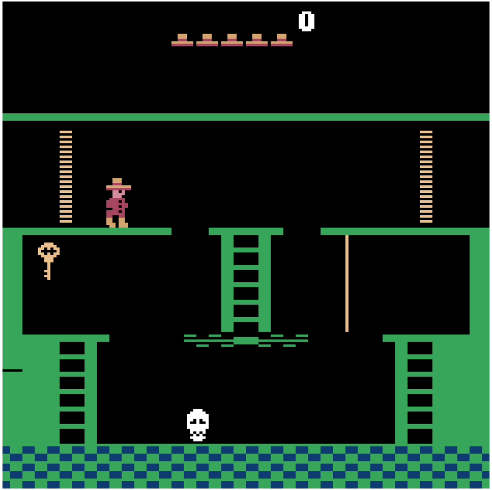


Figure: https://images.openai.com/blob/2c736f64-38dc-4c65-a1dd-aabe2ceb8ddf/learning-montezumas-revenge-from-a-single-demonstration.png

Proximal Curriculum Learning (PCL) [4]

- starting state based on *probability of success* PoS
- $PoS \approx 0.5 \rightarrow$
- distribution over S_{init}
- Approximate PoS via agents value function V (scaled to $[0, 1]$)

Proximal Curriculum Learning (PCL) [4]

- starting state based on *probability of success* PoS
- $PoS \approx 0.5 \rightarrow$
- distribution over S_{init}
- Approximate PoS via agents value function V (scaled to $[0, 1]$)

Problems:

- V initialized randomly
- V inaccurate for seldomly seen states
- states may get PoS of 0 or 1 and never get chosen

State Novelty

Exploration may help us fix the problem.

- state novelty $\sim$ V inaccuracy

$\rightarrow$ explore seldomly seen states

- set starting state based on state novelty

Novelty-Guided Proximal Curriculum Learning NGPCL

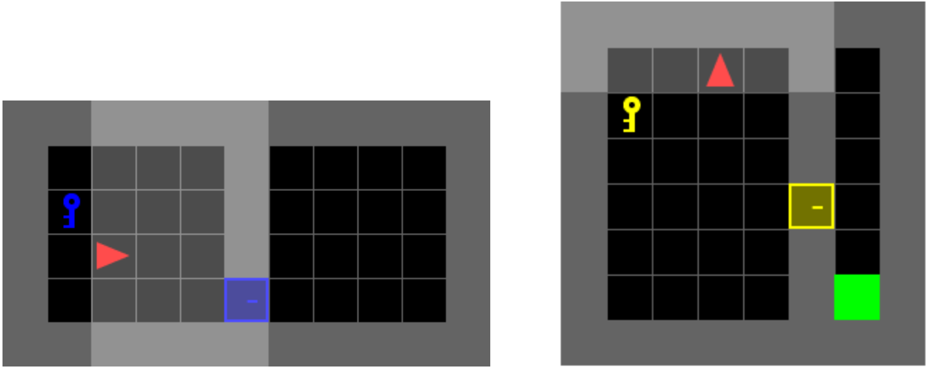
- incorporate novelty by creating distribution over S_{init}

$\rightarrow$ faster V convergence

$\rightarrow$ skip environment steps needed for intrinsic reward

- overlay both distributions by using weighted sum

Environments

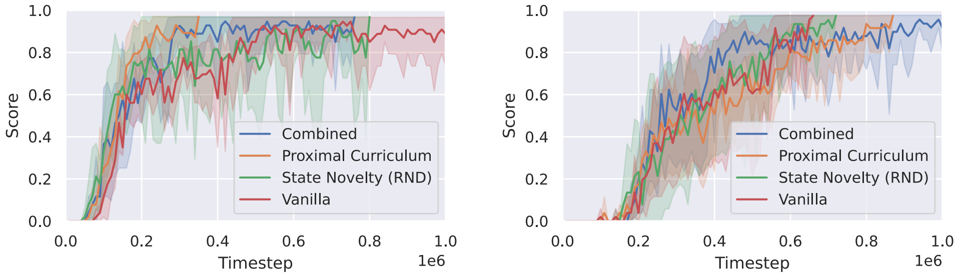


(a) Unlock (b) DoorKey-8x8

Figure: Environments used for experiments [2]. Key was added to observation.

Setup

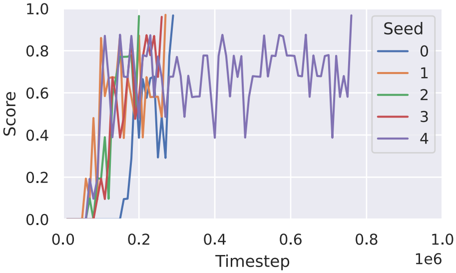
- S_{init} over all states, not evolving
- Random Network Distillation (RND) [1] as state novelty implementation
- Hyperparameters and architectures optimized by bayesian optimization using SMAC [3]



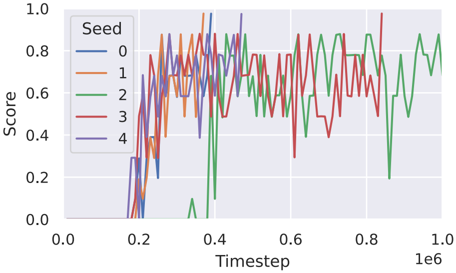
(a) Unlock

(b) Doorkey-8x8

Figure: Rewards over training steps starting from usual starting state (NGPCL). 5 seeds with 95% CI.



(a) Unlock



(b) Doorkey-8x8

Figure: Rewards over the training steps given seed (NGPCL).

Discussion

Results:

- some seeds showed fast learning
- ... others failed to solve the environment
- performance differs per environment

Possible causes:

- distribution overlaying may be destructive
- Hyperparameters could be set poorly (especially for RND)

Future Work

Things to try:

- schedule for overlay parameter
- interleave PCL with RND
- try other state novelty approaches
- (dynamic) S_{init} determination

Environments to evaluate:

- dense rewards
- big/continuous state- and/or action-spaces
- reward space not being convex