A Proactive Approach to Policy Learning

Chaz Chang
Dept. of Computer Science
San Jose State University
San Jose, CA
chaz.chang@sjsu.edu

Divyashree Jayaram
Dept. of Computer Science
San Jose State University
San Jose, CA
divyashree.jayaram@sjsu.edu

Yosha Mundhra
Dept. of Computer Science
San Jose State University
San Jose, CA
yosha.mundhra@sjsu.edu

***Abstract*—Reinforcement learning is an effective method for learning behavior through interactions with one's surroundings. However, most behaviors are learned in a completely reactive manner, in which a suitable action is chosen in response to an observation. It is difficult to develop new abilities when new decisions must be made. This makes learning inefficient, especially in situations where fine and coarse control are required. To solve this, the new approach offer a proactive setting where the agent chooses an action in a state and decides how long to repeat it. The proposed method establishes skip connections between states and develops a skip-policy for repeating the particular action over these skips. The findings of this experiment suggest that the new method can learn to reach the goal faster than traditional Q-learning.**

***Index Terms*—Better exploration, improved learning speed, sample efficient, skip-connections**

I. Introduction

a) Motivation: An approach to learn behaviour by interacting with the environment is called Reinforcement learning. Despite the fact that reinforcement learning (RL) has had a lot of success in recent years [1, 2], policies are still learned in a primarily reactive manner, that is, observing a condition and reacting to it with an action. Temporal abstractions are a typical method to make policies with potentially extensive action sequences easier to learn [3, 4]. Temporal abstraction is learned at the top of the hierarchy, while needed behavior is learned at the bottom. A goal policy on high level learns important states that must be visited and it learns how to reach the goal on the lower level.

b) Impact of the Research: Improved learning speed, Better exploration.

c) Problem Statement: With traditional RL, learning a policy can be slow. Also, it is difficult to learn when new decisions must be made. This makes learning inefficient, especially in situations where fine and coarse control are required.

d) New Approach: Using skip connections [5] allows us to learn the skip-policy to repeat a particular action for certain number of steps, making new approach sample efficient.

1) In comparison to standard one-step exploration, this could potentially provide better results.
2) By requiring fewer decisions, proactive policies enable faster learning.
3) Explainability is the ability of learning agents to indicate when new decisions are expected.

II. Formal Problem and Evaluation Criteria

A. Formal Problem

Learning a policy with standard reinforcement learning is slow and inefficient in terms of sample size. It is also difficult to develop new skills when new decisions must be made.

- Markov Decision Process M = ⟨ S, A, P, R ⟩, S is a finite set of states, A is a finite set of actions, P is a state transition probability matrix, R is a reward function.
- π: policy - behaviour policy and skip policy.
  1) Q function to learn behaviour policy.
  2) n-step Q function to learn skip policy.

$$Q^{\pi}(s,a) := \mathbb{E}\left[r_t + \gamma Q^{\pi}(s_{t+1}, a_{t+1}) | s = s_t, a\right]$$

$$Q^{\pi_J}(s,j|a) := \mathbb{E}\left[\sum_{k=0}^{j-1} \gamma^k r_{t+k} + \gamma^j Q^{\pi}(s_{t+j}, a_{t+j}) | s = s_t, a, j\right]$$

where a is an action, s is a state, γ is the discount factor, j is the skip length, and t is the time.

B. Evaluation Criteria

- Objective: Shorten the policy learning time by introducing skip connections in MDP.
- Measure for learning speed:
  1) No. of episodes to learn how to reach the goal.
  2) No. of steps per episode.

III. Methodology

Change MDP (Markov Decision Process) by introducing skip connections:

$$M = \langle S, A, P_c, R_c \rangle$$

where S is state space, A is action space, P_c is transition probabilities, R_c is reward function.

$$c = \langle s, a, j \rangle$$

where s is starting state, a is an action that is repeated, and j is skip-length. A skip connection connects two states s and s' (if reachable) by repeating action a j times. Skip MDPs are similar to skip connections in neural networks, because they can propagate information about future rewards more quickly. They also can learn when changing behaviors gives better rewards. Fig. 1 shows the length of the skip connections j = 1, 2, and 3 for which we repeat action a 1, 2, and 3 times respectively.

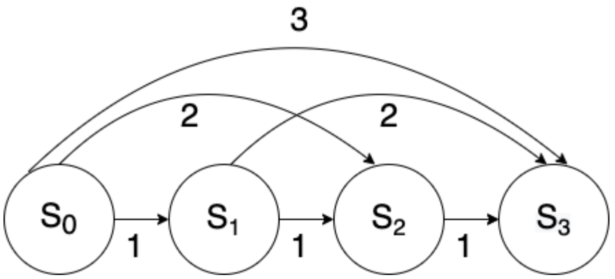
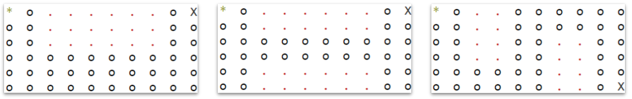


Fig. 1: Skip connections 1, 2, and 3.



```
* o . . . . . . o X
o o . . . . . . o o
o o . . . . . . o o
o o o o o o o o o o
o o o o o o o o o o
o o o o o o o o o o
```

(a) Cliff

```
* o . . . . . . o X
o o . . . . . . o o
o o o o o o o o o o
o o o o o o o o o o
o o . . . . . . o o
o o . . . . . . o o
```

(b) Bridge

```
* o . . o o o o o o
o o . . o o o o o o
o o . . o o . . o o
o o . . o o . . o o
o o o o o o . . o o
o o o o o o . . o X
```

(c) ZigZag

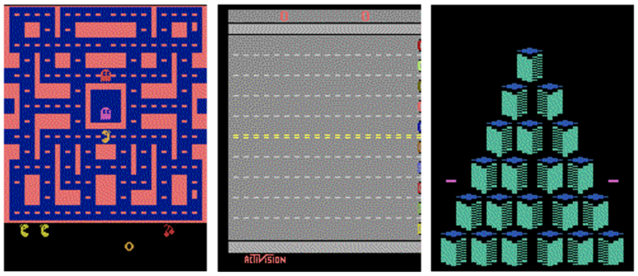
Fig. 2: Gridworlds

IV. Evaluation Settings

- Use an existing environment such as:
  1) Tabular Skip Learning: Gridworlds (Cliff, Bridge, ZigZag) shown in Fig. 2
     - Rewards
       * Goal (X): +1
       * Lava (.): -1
       * Other (o): -0.01
     - 5000 episodes
     - Max skips = 7
     - 100 steps before termination
     - Tabular Q-Learning or Tabular Skip Q-Learning
  2) Skip DQN Learning: Atari (MsPacman, QBert, Freeway) shown in Fig. 3
     - Episodes
       * DQN Freeway: 1229, QBert: 18861, MsPacman: 10410
       * Skip DQN Freeway: 1230, QBert: 19392, MsPacman: 10737
     - Max skips = 10
     - 10000 steps before termination
     - DQN or Skip DQN
- Run traditional RL methods on the selected environments.
- Run the new approach on the selected environments.
- Compare how fast the Skip Q-Learning learns to reach the goal compared to the traditional Q-Learning by measuring reward vs. episodes and no. of steps vs episodes.

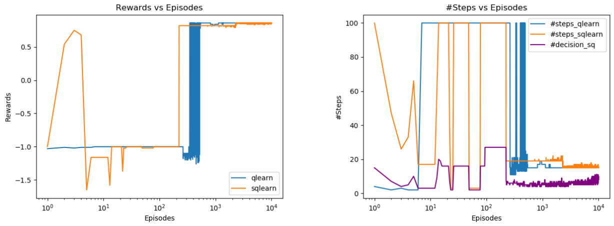
V. Evaluation Results

Introduced skip-MDPs is a method that uses established and popular learning methods. The proposed technique is empirically evaluated in an adversarial environment utilizing tabular and deep function approximation of learning behavior.



(a) MsPacman (b) Freeway (c) QBert

Fig. 3: Atari



(a) Rewards Vs Episodes (b) Steps Vs Episodes

Fig. 4: Gridworlds (Cliff)

A. Tabular Skip Learning Results

- Qlearn: Q-Learning
- Sqlearn : Skip Q-Learning

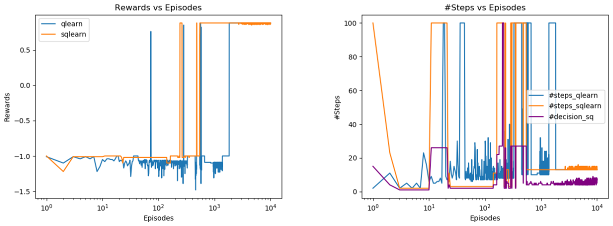
Skip Q-Learning reaches the goal quickly. Skip Q-Learning discovers a policy that reaches the goal with fewer decisions making it easy to learn than Q-Learning as shown in figures Fig. 4, 5, 6.

B. Skip DQN Results

- DQNlearn: DQN-Learning
- sDQNlearn : Skip DQN-Learning
  - Skip DQN in comparison to DQNs, starts learning faster and attains a higher final reward for Atari (Freeway) Fig. 7.
  - Skip DQN continues to outperform DQN, having learned to balance coarse and fine control levels for Atari (QBert) Fig. 8.
  - It also attains the same performance as DQN, Skip DQN on Atari (MsPacman) Fig. 9 and learns to apply various degrees of fine and coarse control. A proactive Skip DQN, on the other hand, requires around 33% fewer decisions.

C. Non-trivial Evaluation

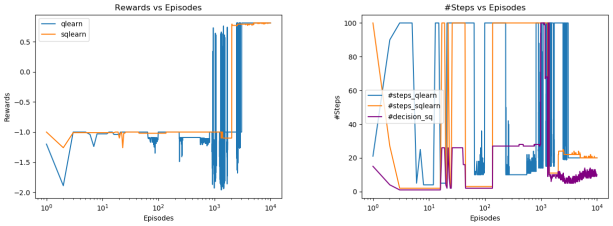
Increasing max-skips too much decreases performance when there is an equal chance of picking all possible skips lengths as shown in Fig. 10 (a).



(a) Rewards Vs Episodes

(b) Steps Vs Episodes

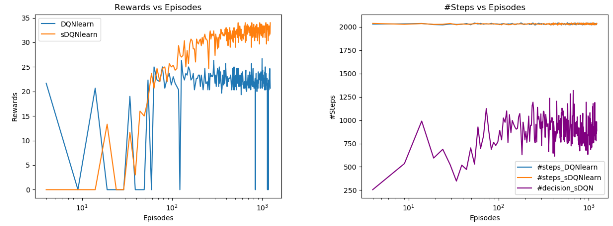
Fig. 5: Gridworlds (Bridge)



(a) Rewards Vs Episodes

(b) Steps Vs Episodes

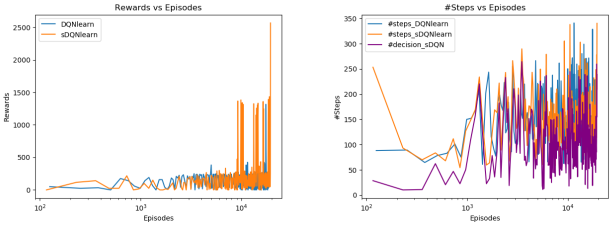
Fig. 6: Gridworlds (ZigZag)



(a) Rewards Vs Episodes

(b) Steps Vs Episodes

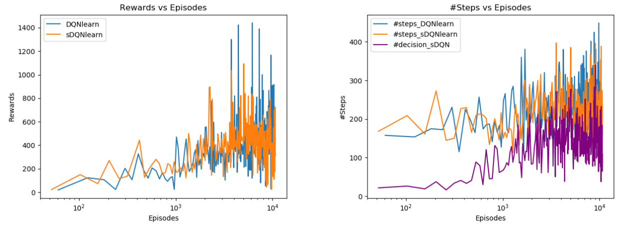
Fig. 7: Atari (Freeway)



(a) Rewards Vs Episodes

(b) Steps Vs Episodes

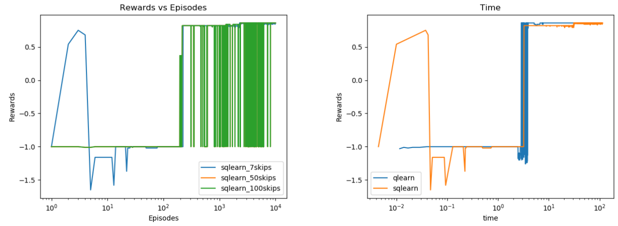
Fig. 8: Atari (QBert)



(a) Rewards Vs Episodes

(b) Steps Vs Episodes

Fig. 9: Atari (MsPacman)



(a) Different Skip Lengths

(b) Time

Fig. 10: Non-trivial Results

VI. Discussion

It is discovered that the Skip Q-Learning is able to learn how to respond in a given state and when to change to a new action. It also reveals that the ability to repeat activities boosts learning speed and understand which repeats are beneficial for understanding when to change actions. Both tabular and deep RL were found to benefit greatly from the proposed technique. Skip Q-Learning requires fewer decisions, which enables faster learning by repeating actions and deciding when it is beneficial to change them. It can provide better results in comparison to standard one-step exploration.

A. *Non-trivial Analysis*

- If max-skips = 7, there is 1/7 chance of choosing the correct 1st optimal skip-length. If max-skips = 1000, there is 1/1000 chance of choosing the correct 1st optimal skip-length as shown in Fig. 10 (a).
- Even though Skip Q-learning has extra logic, it learns at the same speed as Q-learning in terms of execution time as shown in Fig. 10 (b).

VII. Related Works

- Schoknecht Riedmiller (2002; 2003) [6] shows that learning with multiple step actions can greatly speed up learning. Lakshminarayanan et al. (2017) [7] proposed DAR, a Q-network with many outputs per action to include different repetition lengths, greatly increasing the action space but improving learning. Sharma et al. (2017) [8] proposed FiGAR, a framework that simultaneously learns an action policy and repetition policy that determines repetition length.

- However, the repetition policy only learns one repetition length for all actions that is good on average. Further, FiGAR needs changes to the training of the agent to include the repetition policy.
- The proposed approach allows to learn policy efficiently by learning a skip policy (executing the same action) and to learn when another decision has to be chosen on the behavioural level.

References

[1] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". In: *nature* 518.7540 (2015), pp. 529–533.

[2] Bowen Baker et al. "Emergent tool use from multi-agent autocurricula". In: *arXiv preprint arXiv:1909.07528* (2019).

[3] Richard S Sutton, Doina Precup, and Satinder Singh. "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning". In: *Artificial intelligence* 112.1-2 (1999), pp. 181–211.

[4] Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. "Search on the replay buffer: Bridging planning and reinforcement learning". In: *Advances in Neural Information Processing Systems* 32 (2019).

[5] André Biedenkapp et al. "TempoRL: Learning when to act". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 914–924.

[6] Ralf Schoknecht and Martin Riedmiller. "Speeding-up reinforcement learning with multi-step actions". In: *International Conference on Artificial Neural Networks*. Springer. 2002, pp. 813–818.

[7] Aravind Lakshminarayanan, Sahil Sharma, and Balaraman Ravindran. "Dynamic action repetition for deep reinforcement learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017.

[8] Sahil Sharma, Aravind Srinivas, and Balaraman Ravindran. "Learning to repeat: Fine grained action repetition for deep reinforcement learning". In: *arXiv preprint arXiv:1702.06054* (2017).