Adaptive Discount Factor in RL

Dikshant
University of Alberta
ddikshan@ualberta.ca

Pranaya Jajoo
University of Alberta
pranayajajoo@ualberta.ca

Siddarth Chandrasekar
University of Alberta
siddart2@ualberta.ca

Abstract

The discount factor (γ) is a pivotal parameter in Reinforcement Learning (RL), shaping an agent's temporal decision-making and value estimation. This research explores an adaptive discount factor (γ_a) that increases with an agent's interactions. By dynamically adjusting the discount factor during training, this study investigates whether this approach can improve learning efficiency and policy performance across grid-world environments. The experimental results demonstrate the potential of adaptive discounting to accelerate convergence and mitigate initialization bias.

1 Introduction

Credit assignment is a key challenge in developing artificial learning agents, particularly RL. The discount factor (γ) determines the prioritization of immediate versus future rewards, fundamentally shaping an agent's learning trajectory by determining how far into the future consequences are considered. Choosing the right γ is critical as it impacts both the learning efficiency and robustness.

A higher γ (closer to 1) prioritizes long-term rewards, which can increase the computational complexity of value estimation, requiring a lot of environmental interactions. A lower γ (closer to 0) can accelerate learning by focusing on short-term rewards but risks converging to suboptimal policies in long-horizon tasks. To address this trade-off, we investigate whether an adaptive γ can improve sample efficiency while learning the optimal policy. Specifically, we ask:

Can an agent achieve the optimal return more quickly with an adaptive discount factor?

We show that gradually increasing γ during training benefits learning in tabular Markov Decision Processes (MDPs). Our research draws inspiration from the work of Laidlaw et al. [1], which demonstrates that in current Deep-RL benchmark environments, agents do not require extensive future planning to identify optimal policies. This insight is encapsulated by the concept of *effective horizon*, which quantifies the depth of lookahead an agent needs to make *optimal* decisions. In most environments, it is substantially shorter than the optimal episode length, indicating that agents can effectively focus on near-term strategies rather than being overly concerned with distant consequences. Further, Lavet et al. [2] showed that progressively increasing the discount factor during training reduces instability and accelerates learning, particularly in non-linear function approximation settings. By starting with a slighty lower γ and gradually increasing it, agents achieve faster policy improvement and robustness to initialization issues.

2 Background

RL problems are typically framed as MDPs [3, 4], formally defined as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma_e)$ where $\mathcal{S}$ is the set of all possible states, $\mathcal{A}$ is the set of all possible actions, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition probability function, and $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function. The future discounted return at time t is defined as $R_t = \sum_{t'=t}^{T} \gamma_e^{t'-t} r(s_{t'}, a_{t'})$, where T is the time step when the episode

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

terminates and $\gamma_e \in [0, 1]$ is the environment discount factor. The optimal action-value function $Q^*_{\gamma_e}(s, a)$ is defined as the maximum expected return achievable by following a policy π, after taking action a in state s: $Q^*_{\gamma_e}(s, a) = \max_\pi \mathbb{E}[R_t | s_t = s, a_t = a, \pi]$. Here, policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ is a *strategy* that controls agent behavior, and the goal is to find the optimal *strategy* π^* that yields the maximum return. Q-Learning, as introduced by Watkins et al. [5], is an off-policy algorithm that learns the optimal action-value functions by recursively updating reward estimations across state-action sequences: $Q_\gamma(s, a) = \max_\pi \mathbb{E}_\mathcal{P} [r(s, a) + \gamma \max_{a'} Q_\gamma(s', a')]$

Adaptive discounting: Recent research highlights the advantages of employing adaptive approaches to the discount factor in RL. Studies have explored strategies such as using agent-specific discount factors [6, 7] and incrementally increasing γ during training [2, 8]. These approaches can improve learning efficiency while maintaining policy's learning stability.

Average-reward RL: In the average-reward RL paradigm, the goal is to maximize the agent's long-term average reward over an infinite horizon. This average reward is defined as:

$$g^\pi_s = \lim_{T\to\infty} \frac{1}{T+1} \mathbb{E}_{\mathcal{P},\pi} \left[\sum_{t=0}^{T} r(s_t, a_t) | s_0 = s \right] \forall s \in S \tag{1}$$

There exists a critical discount factor, $\gamma^* \in [0, 1)$, such that $\forall \gamma \in [\gamma^*, 1)$, the γ-optimal policy in episodic RL settings aligns with the optimal policy for the average-reward objective [9]. This connection suggests that adapting γ can unify episodic and average-reward formulations.

3 Adaptive Discount Factor

We define the adaptive discount factor at episode n as $\gamma^n_a = \min\left(\gamma_e, \gamma^{n-1}_a + \gamma_e \cdot \frac{n}{T}\right)$, where T is the maximum number of steps and γ_e is the environment's discount factor. We initialize γ_a to 0.5, and increase it linearly with environmental interactions, facilitating a gradual adjustment in credit assignment during training. This linear increase aims to balance learning efficiency and long-term planning by addressing two key hypotheses:

- **H1:** A smaller initial discount factor reduces the burden of accurate long-horizon value predictions, enabling the agent to learn a stable value function more quickly.
- **H2:** Adaptive discount factor can mitigate the effects of biases in initial value estimates, as bootstrapping methods propagate errors less aggressively when future rewards are down-weighted.

This adaptive approach helps the agent learn a policy optimized for the MDP's original discount factor, γ_e, more effectively by aligning initial training dynamics with shorter planning horizons and gradually extending the focus to long-term rewards. While prior studies, such as Lavet et al. [2], have investigated adaptive discount strategies using deep neural networks, the specific mechanisms and benefits of these methods have not been thoroughly examined. This work seeks to address this gap by systematically analyzing how linearly increasing discount factor impacts sample efficiency, convergence stability, and overall policy performance.

4 Experimental Setup

We test our hypothesis using tabular Q-learning across a suite of grid-world environments [10][1]. Each environment requires the agent to learn the optimal policy for the environment discount factor of $\gamma_e = 0.99$. The learning rate was set to 0.5, determined through hyperparameter tuning for this fixed γ_e. To evaluate the learned policies, we measure the expected return every 100 steps over a total of 60,000 steps. During each evaluation step, the agent is evaluated over 15 episodes, and the results are averaged across 50 independent seeds to ensure statistical robustness. Further, we test for different initialization values of the action-value function in Q-learning, specifically $Q_{init} \in \{0, 5, 10\}$. We perform and compare these experiments for 3 different discount factors: γ_0(=0.5), γ_e(=0.99), and γ_a

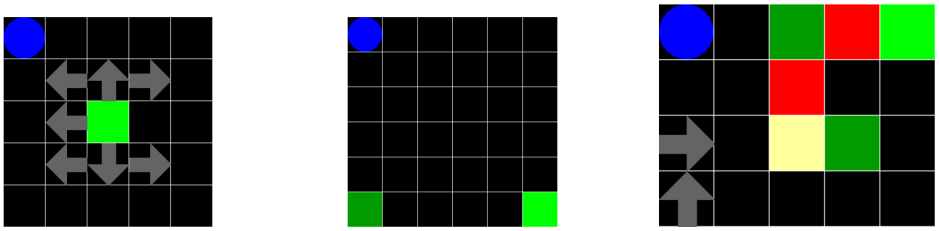
Our experiments focus on three representative environments (Figure 1): `Barrier 5x5-v0`, `Empty-Distract-6x6-v0`, and `Full 4x5-v0`, as they encompass diverse reward structures, stochasticity, and environmental constraints.

[1] Code can be accessed on github from here: Adaptive Discount Factor in RL

The characteristics of these environments are as follows:

- **Blue circle**: Represents the agent,
- **Black tiles**: Empty spaces,
- **Red tiles**: Negative reward tiles, -10
- **Yellow tiles**: Quicksand tiles, where actions fail with a 90% probability - sticky actions,
- **Green tiles**: Positive reward tiles; brighter shades indicate higher rewards, +0.1 for distraction and +1 for reaching the goal; episode terminates if agent performs stay action here,
- **Black tiles with gray arrows**: Restrictive tiles where the agent can move only in one specific direction; other actions fail.

Table 1 presents the average discounted (w.r.t. γ_e) episodic return, across all evaluation episodes and random seeds. Table 2 displays the percentage of evaluation trajectories that obtain the optimal return, averaged across all seeds. A higher percentage indicates that the agent learned the optimal policy quicker. For a more thorough analysis, additional results for other relevant environments are presented in Appendix A.



`Barrier 5x5-v0` `Empty-Distract-6x6-v0` `Full 4x5-v0`

Figure 1: Three representative environments

$\gamma \setminus Q_{init}$	0	5	10
`Barrier-5x5-v0`			
γ_0	**0.679±0.057**	**0.788±0.046**	**0.771±0.048**
γ_e	0.669±0.058	0.009±0.011	0.009±0.011
γ_a	0.676±0.057	**0.788±0.046**	0.754±0.050
`Empty-Distract-6x6-v0`			
γ_0	0.099±0.009	0.090±0.012	0.090±0.012
γ_e	**0.871±0.023**	0.000±0.001	0.000±0.000
γ_a	0.828±0.033	**0.792±0.042**	**0.778±0.044**
`Full-4x5-v0`			
γ_0	0.031±0.318	-1.089±2.618	-0.813±2.284
γ_e	**0.786±0.279**	-0.303±1.582	-1.071±2.481
γ_a	0.773±0.159	**0.218±1.756**	**-0.326±2.519**

Table 1: Discounted episodic return during evaluation, averaged across all evaluation episodes and across different seeds

$\gamma \setminus Q_{init}$	0	5	10
`Barrier-5x5-v0`			
γ_0	**73.0±2.8**	84.8±0.3	**82.9±0.3**
γ_e	72.0±3.0	0.3±0.1	0.2±0.1
γ_a	72.8±2.0	**85.0±0.3**	81.2±0.6
`Empty-Distract-6x6-v0`			
γ_0	0.5±0.2	0.8±0.1	1.0±0.1
γ_e	**95.8±0.3**	0.0±0.0	0.0±0.0
γ_a	90.6±0.2	**87.5±0.1**	**86.0±0.1**
`Full-4x5-v0`			
γ_0	0.0±0.0	0.0±0.0	0.0±0.0
γ_e	**88.7±0.4**	16.3±0.4	0.0±0.0
γ_a	84.2±0.4	**80.7±0.3**	**80.9±0.4**

Table 2: Percentage of optimal trajectories during evaluation, averaged across all evaluation episodes and across different seeds (%)

5 Discussion

From Tables 1 and 2, it is clear that for non-zero Q_{init} values, the γ_a agent performs better than both γ_e and γ_0 agents, supporting our second hypothesis (**H2**). This is because the γ_a agent reduces noise based on its initial γ value, which lowers bootstrapping errors while maintaining sufficient exploration. This helps reduce initialization bias and provides more accurate value function estimates.

For $Q_{init} = 0$, the performance of γ_a is similar to γ_e, which does not support our first hypothesis (**H1**). This is likely due to the simplicity of the MDPs used, where the reward structures are simple and mostly binary. These environments' effective planning horizons are same as the optimal episode length, demanding a higher discount factor to efficiently propagate final rewards back to all states. As a result, γ_e agents outperform γ_a agents in sparse environments like `Empty-Distract-6x6-v0` and lower γ fails horribly across all initializations.

The adaptive discount factor shows advantages in environments with complex rewards and restricted actions. Structural constraints, such as directional movement restrictions in `Barrier 5x5-v0`, do not affect its performance. In environments like `Full-4x5-v0`, where rewards range from -10 penalties to +1, static high discount factors struggle under high Q_{init} values, leading to instability and slow convergence during early exploration. This is the reason we observe a higher variance in all the environments which have these red tiles (1, A). In contrast, γ_a provides stable performance and effectively optimizes returns in all challenging scenarios demonstrating its robustness.

5.1 Implications

Our results demonstrate that an adaptive discount factor γ_a facilitates efficient learning by mitigating the adverse effects of poor initial value estimates and balancing short-term and long-term rewards. In majority of the environments, we found the performance of γ_a to be similar or to the MDP's discount factor γ_e. An adaptive discount factor could also assist greatly in learning the value function with function approximators, by reducing early noise caused by random initializations [2].

Further, while many RL methods predominantly utilize the ϵ-greedy approach for exploration, optimistic initialization represents an alternative strategy that helps in the initial exploration phase [4]. However, optimistic initialization exhibits significant limitations, requiring meticulous tuning of the agent's learning rate to enable quicker convergence. The adaptive discount factor γ_a offers a more robust alternative, enabling learning without any extensive hyperparameter tuning while simultaneously incorporating advantages of optimistic initialization, as demonstrated in Table 2.

To summarize, adaptive discounting *could* demonstrate particular efficacy when the agent's planning horizon is shorter than the optimal episodic length. Moreover, this approach mitigates the inherent initialization bias present in value function estimations, thereby enhancing the robustness of the learning process. This flexibility makes adaptive discount factors a promising tool for addressing challenges in diverse RL settings, including those with large state spaces.

5.2 Future Work

Limitations: As discussed earlier, the grid-world environments employed in this study present inherent limitations in their complexity and behavioral diversity. This constraint significantly impedes the validation of our hypothesis (**H1**) regarding the potential for reduced planning horizons.

Hence, we plan to expand our experiments to more stochastic and complex MDPs like `BRIDGE` [1], where the effective horizon is much smaller than the episodic length. We will extend our study to larger state spaces using linear and non-linear function approximators, and investigate the adaptive discount factor's performance in classical control environments.

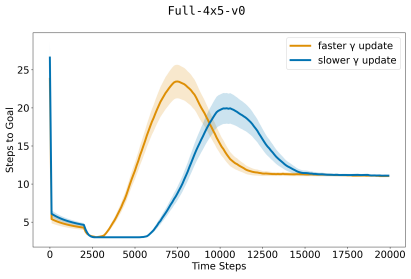


Figure 2: Different γ_a

Further, our preliminary experiments highlighted the significance of the rate of change in γ_a (Figure 2). Our future research aims to investigate more sophisticated adaptive approaches for γ_a that can integrate diverse learning feedback mechanisms. For example, modulating γ_a based on state-visitation frequency or TD-error could provide a mechanism for quantifying the agent's epistemic certainty. This approach would enable the agent to leverage a higher γ_a when bootstrapping value estimates for states with greater confidence. Additionally, in model-based RL, implementing a dynamically decaying γ_a across individual roll-outs may potentially mitigate the impact of imprecise model simulations.

6 Acknowledgment

We want to thank Jacob for his helpful suggestions on how to set up the experiment. We also want to thank Mike and Simone for their feedback on our presentation.

References

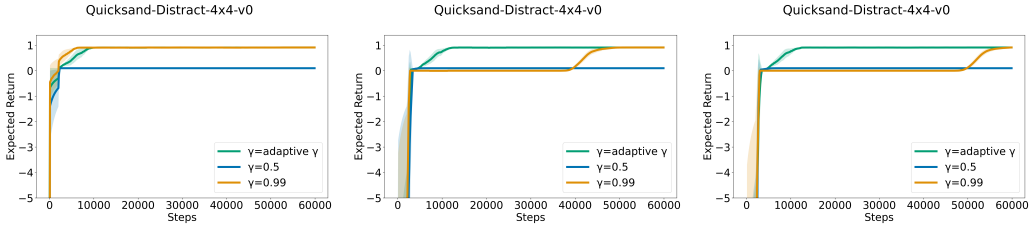
1. Laidlaw C, Russell S, and Dragan A. Bridging RL Theory and Practice with the Effective Horizon. 2024. arXiv: 2304.09853 [cs.LG]. URL: https://arxiv.org/abs/2304.09853.
2. François-Lavet V, Fonteneau R, and Ernst D. How to Discount Deep Reinforcement Learning: Towards New Dynamic Strategies. 2016. arXiv: 1512.02011 [cs.LG]. URL: https://arxiv.org/abs/1512.02011.
3. Puterman ML. Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons, 2014.
4. Sutton RS and Barto AG. Reinforcement learning: An introduction. MIT press, 2018.
5. Watkins CJCH and Dayan P. Q-learning. Machine Learning 1992;8:279–92.
6. Obando-Ceron J, Ara'ujo JG, Courville AC, and Castro PS. On the consistency of hyper-parameter selection in value-based deep reinforcement learning. ArXiv 2024;abs/2406.17523.
7. Araújo JGM, Obando-Ceron J, and Castro PS. Lifting the Veil on Hyper-parameters for Value-based Deep Reinforcement Learning. In: 2021. URL: https://api.semanticscholar.org/CorpusID:249118957.
8. Schwarzer M, Obando-Ceron J, Courville AC, Bellemare MG, Agarwal R, and Castro PS. Bigger, Better, Faster: Human-level Atari with human-level efficiency. In: *International Conference on Machine Learning*. 2023. URL: https://api.semanticscholar.org/CorpusID:258987895.
9. Grand-Clément J and Petrik M. Reducing Blackwell and Average Optimality to Discounted MDPs via the Blackwell Discount Factor. ArXiv 2023;abs/2302.00036.
10. Parisi S. Gym Gridworlds. https://github.com/sparisi/gym_gridworlds. 2024.

A Appendix

$\gamma \setminus Q_{init}$	**0**	**5**	**10**
`Empty-10x10-v0`			
γ_0	0.630±0.049	0.453±0.059	0.428±0.059
γ_e	**0.649±0.047**	0.000±0.001	0.000±0.000
γ_a	0.637±0.048	**0.553±0.056**	**0.537±0.057**
`Penalty-3x3-v0`			
γ_0	0.702±1.059	**0.768±0.975**	0.683±1.264
γ_e	**0.707±1.035**	0.467±0.855	0.317±1.132
γ_a	0.640±1.211	0.686±1.192	**0.739±1.101**
`Quicksand-Distract-4x4-v0`			
γ_0	0.047±0.373	-0.560±2.015	-0.650±2.087
γ_e	**0.847±0.208**	-0.125±1.436	-0.419±1.754
γ_a	0.808±0.239	**0.082±2.058**	**-0.030±2.148**
`TwoRoom-Quicksand-3x5-v0`			
γ_0	0.855±0.032	0.792±0.045	0.778±0.047
γ_e	0.875±0.028	0.201±0.054	0.027±0.022
γ_a	**0.880±0.026**	**0.819±0.041**	**0.802±0.044**
`TwoRoom-Distract-Middle-2x11-v0`			
γ_0	0.916±0.020	0.854±0.039	**0.842±0.041**
γ_e	0.918±0.019	0.120±0.044	0.001±0.003
γ_a	**0.919±0.019**	**0.855±0.038**	0.841±0.041

$\gamma \setminus Q_{init}$	**0**	**5**	**10**
`Empty-10x10-v0`			
γ_0	72.9±1.7	54.1±0.2	51.2±0.1
γ_e	**75.3±1.5**	0.0±0.0	0.0±0.0
γ_a	73.8±1.5	**66.2±0.1**	**64.3±0.1**
`Penalty-3x3-v0`			
γ_0	99.4±0.0	97.9±0.1	**97.5±0.1**
γ_e	**99.5±0.0**	64.6±0.1	54.2±0.0
γ_a	**99.5±0.0**	**97.8±0.1**	97.5±0.0
`Quicksand-Distract-4x4-v0`			
γ_0	0.0±0.0	0.0±0.0	0.0±0.0
γ_e	**92.3±0.5**	29.2±0.5	12.6±0.4
γ_a	87.8±0.4	**84.1±0.4**	**83.8±0.5**
`TwoRoom-Quicksand-3x5-v0`			
γ_0	79.7±0.9	75.5±0.8	74.5±0.8
γ_e	90.6±0.4	21.3±0.4	2.8±0.2
γ_a	**90.7±0.4**	**86.4±0.4**	**84.6±0.4**
`TwoRoom-Distract-Middle-2x11-v0`			
γ_0	**98.2±0.1**	90.6±0.1	89.3±0.1
γ_e	**98.2±0.1**	12.7±0.1	0.0±0.0
γ_a	**98.2±0.1**	**90.7±0.1**	**89.3±0.2**

Table 3: (L) Discounted episodic return during evaluation, averaged across all evaluation episodes and across different seeds; (R) Percentage of optimal trajectories during evaluation, averaged across all evaluation episodes and across different seeds (%)



$Q_{init} = 0$ $Q_{init} = 5$ $Q_{init} = 10$

Figure 3: Learning curve of the agent across different initializations with different agent discount factor, $\gamma_0 = 0.5$, γ_e = 0.99, and γ_a for `Quicksand-Distract-4x4-v0` environment.