Reinforcement Learning

Lecture 7

TD(λ)

Eligibility Traces

Robert Peharz

Institute of Theoretical Computer Science
Graz University of Technology
Winter Term 2023/24

Recap: Model-free Control with Monte Carlo

$$q_\pi(s,a)$$

$$= \underbrace{r(s,a) + \gamma \sum_{s'} p(s' \mid s,a)}_{\text{MDP, now unknown}} \, v_\pi(s')$$

Exploration  Exploitation

On-policy methods
need to find a compromise

Off-policy methods
separate the problem

Behavior policy b
generates behavior, data

Target policy π
learns from b's data

Monte Carlo Control with Exploring Starts

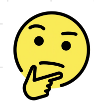


Image: pinterest.uk

ε-greedy Policy: With probability ε, take random action (uniformly), otherwise take greedy action

?
ε → random
1-ε → greedy

$$\pi(a \mid s) = \begin{cases} 1-\varepsilon + \frac{\varepsilon}{|\mathcal{A}(s)|} & \text{if } a = \arg\max_{a'} q(s,a') \\ \frac{\varepsilon}{|\mathcal{A}(s)|} & \text{o.w.} \end{cases}$$

GLIE

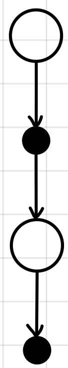
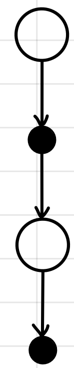
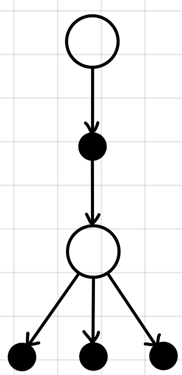
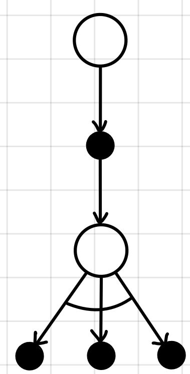
Greedy in the Limit with Infinite Exploration

Off-policy Monte Carlo via Importance Sampling

$$W \leftarrow W \frac{\pi(A_t \mid S_t)}{b(A_t \mid S_t)}$$

Recap: TD-based Control

$$q(s,a) \leftarrow q(s,a) + \alpha\left(\text{Goal} - q(s,a)\right)$$

	on-policy	off-policy
sample A	**Sarsa** $(\boxed{S}_t, \boxed{A}_t, \boxed{R}_{t+1}, \boxed{S}_{t+1}, \boxed{A}_{t+1})$ SARSA  Goal = $R_{t+1} + \gamma\, q(S_{t+1}, A_{t+1})$	**Importance-weighted Sarsa**  Goal = $R_{t+1} + \gamma \frac{\pi(A_{t+1} \mid S_{t+1})}{b(A_{t+1} \mid S_{t+1})} q(S_{t+1}, A_{t+1})$
$\mathbb{E}_A$	**Expected Sarsa**  Goal = $R_{t+1} + \gamma \sum_{a'} \pi(a' \mid S_{t+1})\, q(S_{t+1}, a')$	**Q-learning (Sarsa Max)**  Goal = $R_{t+1} + \gamma \max_{a'} q(S_{t+1}, a')$

TD(λ) and Eligibility Traces

Recall: n-step TD Updates

General update: $v(s) \leftarrow v(s) + \alpha\,(\hat{G} - v(s))$

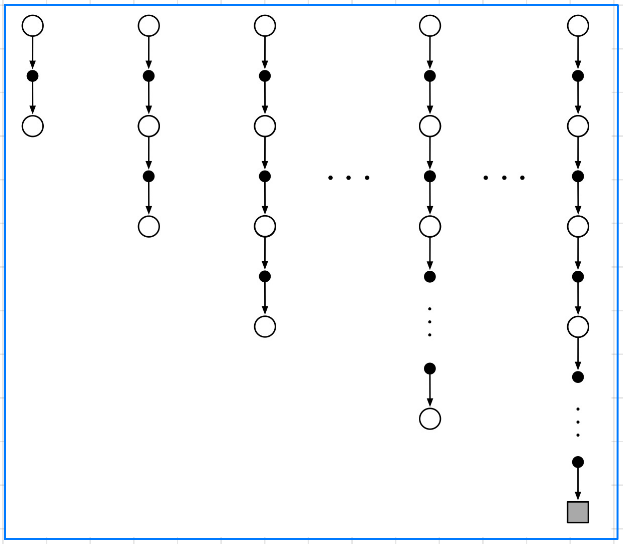
(1-step) TD: $\hat{G} = R_{t+1} + \gamma\, v(S_{t+1})$

2-step TD: $\hat{G} = R_{t+1} + \gamma R_{t+2} + \gamma^2\, v(S_{t+2})$

3-step TD: $\hat{G} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3\, v(S_{t+3})$

$\vdots$

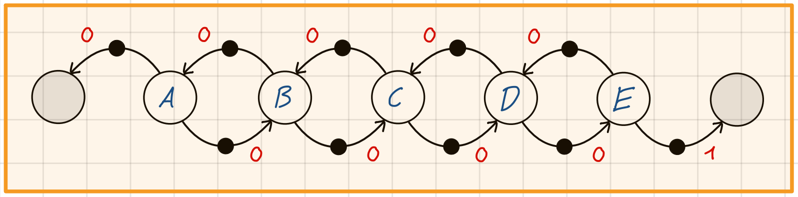
∞-step TD (MC) $\hat{G} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + \ldots = G_t$



Which is the best n?

Which converges fastest?

Random Walk



Consider the random walk example again, but with 19 states.
Difference to true v_π for various n and α:

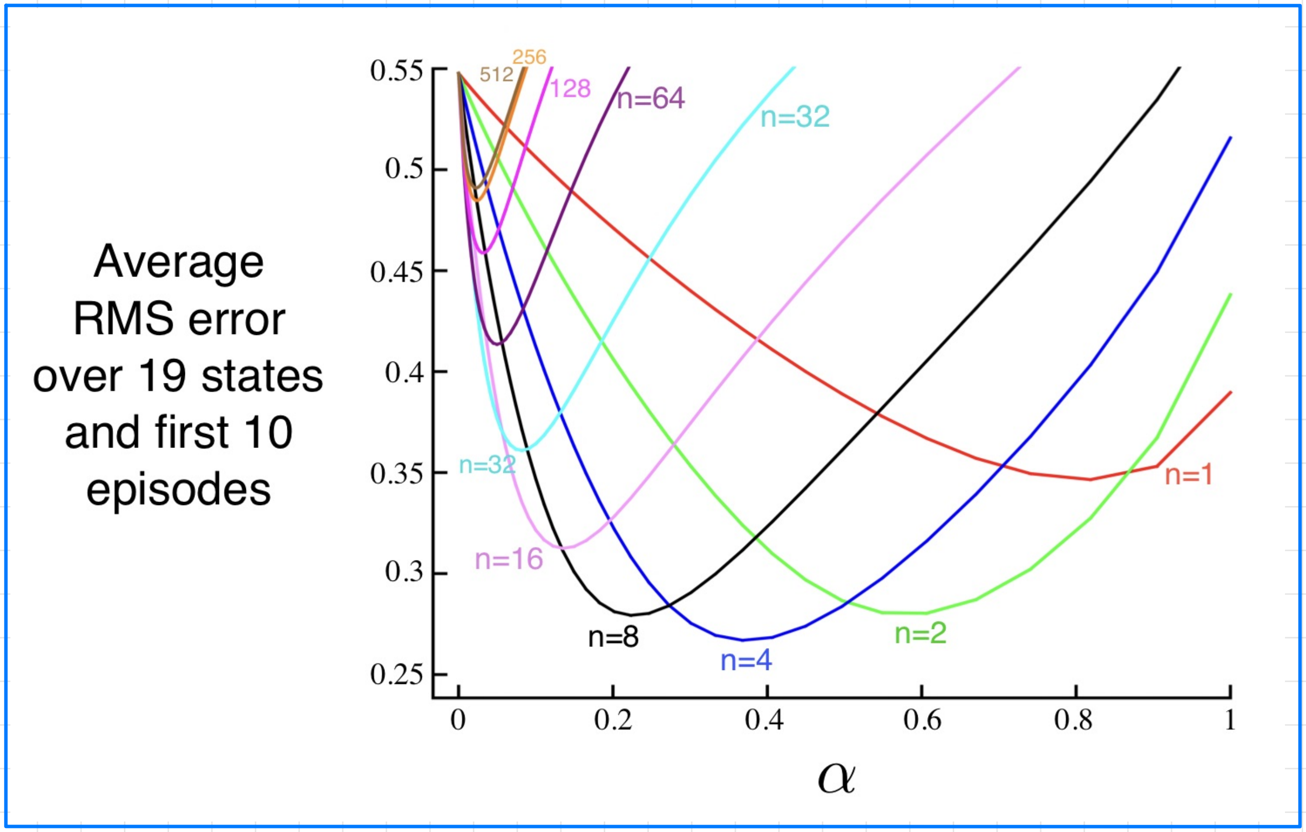


Image: Sutton & Barto

Sweet spot at $n=4$ and $\alpha = 0.39$. This is highly problem-dependent.

Since optimal n is hard to pick, why not take all of them?

N-step Return

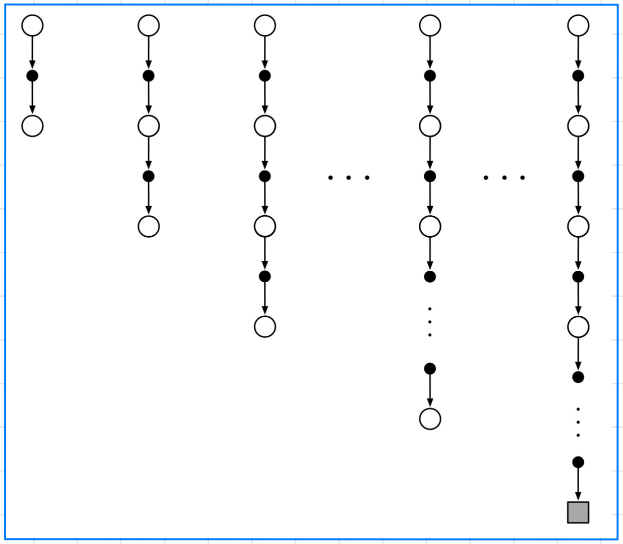
(1-step) TD: $\hat{g} = R_{t+1} + \gamma v(S_{t+1})$ $=: \hat{G}_t^1$

2-step TD: $\hat{g} = R_{t+1} + \gamma R_{t+2} + \gamma^2 v(S_{t+2})$ $=: \hat{G}_t^2$

3-step TD: $\hat{g} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 v(S_{t+3})$ $=: \hat{G}_t^3$

$\vdots$

∞-step TD (MC) $\hat{g} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + \ldots = G_t$ $=: \hat{G}_t^\infty$



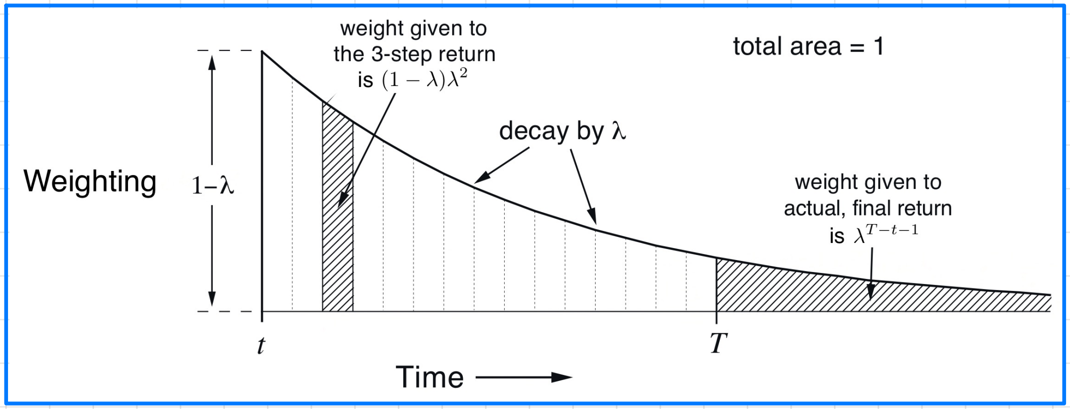
n-step return

$$\hat{G}_t^n := \sum_{k=0}^{n-1} \gamma^k R_{t+k+1} + \gamma^n v(S_{t+n})$$

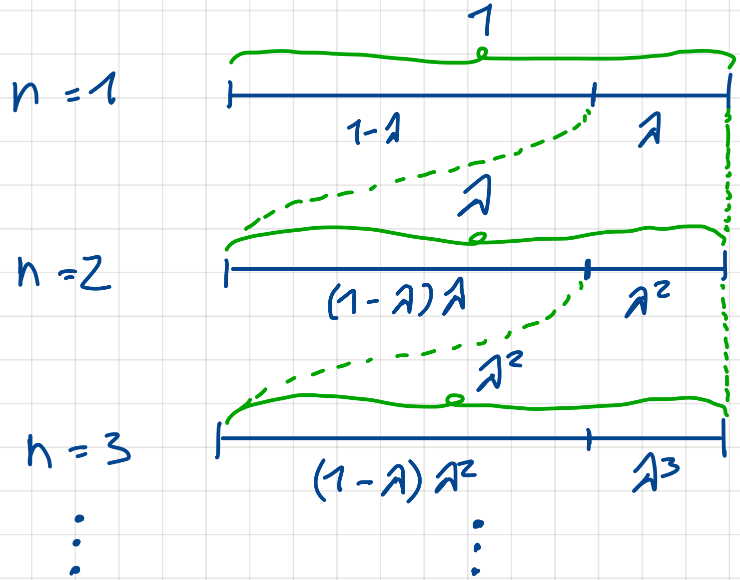
- any $\hat{G}_t^n$ is a legit goal
- any weighted average of $\hat{G}_t^n$'s is also a legit goal!
- e.g. we could use $\frac{1}{2}\hat{G}_t^2 + \frac{1}{2}\hat{G}_t^4$, but also average all $\hat{G}_t^n$'s

λ-Return

- for $\lambda \in [0,1]$ consider weights $(1-\lambda)\lambda^{n-1}$
- note that $\sum_{n=1}^{\infty} (1-\lambda)\lambda^{n-1} = 1$



"stick breaking weights"



Average all $\hat{G}_t^n$ with these weights: λ-Return

$$G_t^{\lambda} = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} \hat{G}_t^n$$

For finite episode length T:

$$G_t^{\lambda} = (1-\lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} \hat{G}_t^n + \lambda^{T-t-1} \hat{G}_t^{\infty}$$

Note:

$$G_t^0 = \hat{G}_t^1$$

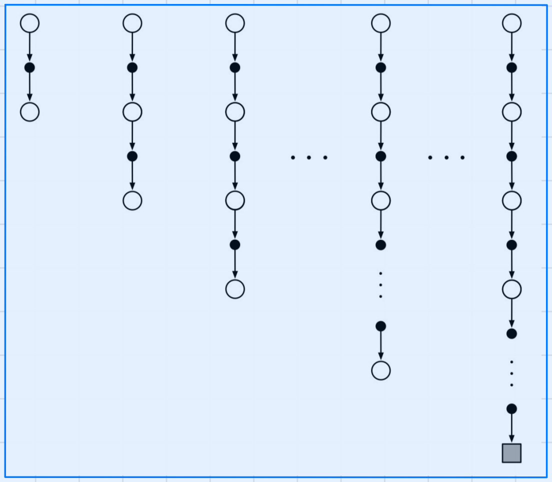
$$G_t^1 = \hat{G}_t^{\infty}$$

λ-Return for Prediction

n-step TD

$$\hat{G}_t^n := \sum_{k=0}^{n-1} \gamma^k R_{t+k+1} + \gamma^n V(S_{t+n})$$

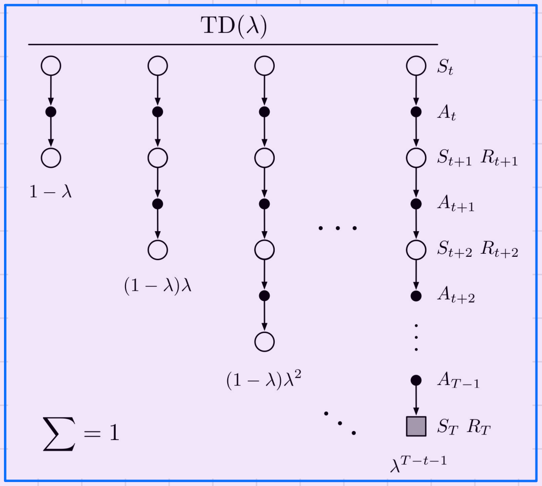
$$V(s) \leftarrow V(s) + \alpha\left(\hat{G}_t^n - V(s)\right)$$



λ-return algorithm

$$G_t^{\hat{\lambda}} = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} \hat{G}_t^n$$

$$V(s) \leftarrow V(s) + \alpha\left(G_t^{\lambda} - V(s)\right)$$



Prediction with λ-Return

Let $\alpha \in (0,1]$, $\lambda \in [0,1]$

Initialize $V(s)$ arbitrarily, except $V(s) = 0$ for terminal s

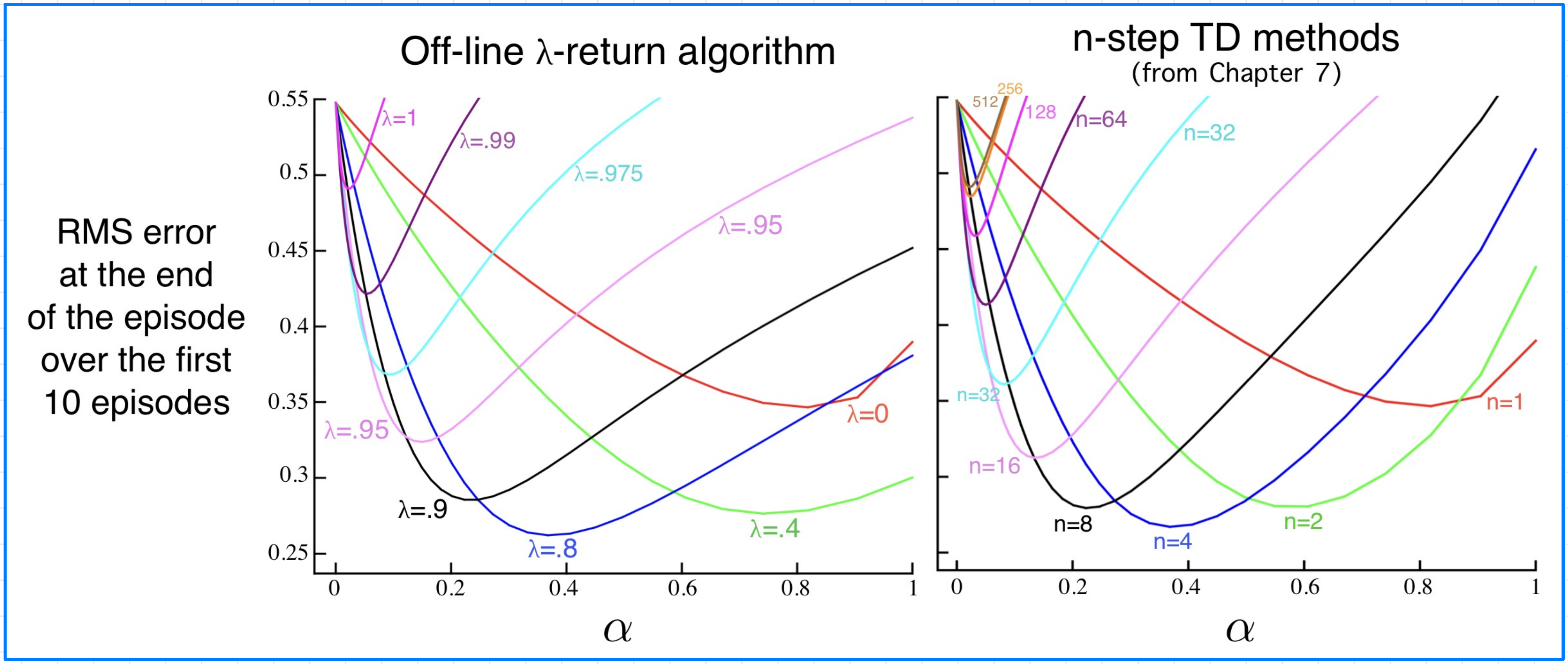
- repeat
  - using π generate episode
    $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, \ldots, S_{T-1}, A_{T-1}, R_T, S_T$ terminal
  - for $t = T-1 \ldots 0$
    - for $n = 1 \ldots T-t$ $\quad \hat{G}_t^n = \begin{cases} R_{t+1} + \gamma V(S_{t+1}) & \text{if } n = 1 \\ R_{t+1} + \gamma \hat{G}_{t+1}^{n-1} & \text{if } 2 \leq n \leq T-t \end{cases}$ (*)
    - $G_t^\lambda = (1-\lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} \hat{G}_t^n + \lambda^{T-t-1} \hat{G}_t^{T-t}$
    - $V(S_t) \leftarrow V(S_t) + \alpha \left(G_t^\lambda - V(S_t) \right)$

(*) $$\hat{G}_t^n := \sum_{k=0}^{n-1} \gamma^k R_{t+k+1} + \gamma^n V(S_{t+n}) = R_{t+1} + \sum_{k=1}^{n-1} \gamma^k R_{t+k+1} + \gamma^n V(S_{t+n})$$

$$= R_{t+1} + \gamma \left(\sum_{k=0}^{n-2} \gamma^k R_{t+k+2} + \gamma^{n-1} V(S_{t+1+n-1}) \right) = R_{t+1} + \gamma \hat{G}_{t+1}^{n-1}$$

N-step TD vs. λ-Return

Consider the random walk example with 19 states.
Difference to true v_π for various n (λ) and α



Some intermediate value of $\lambda = 0.5$ "will usually do."

Control with λ-Return

Let $\varepsilon > 0$, $\alpha \in (0,1]$, $\lambda \in [0,1]$

Initialize $q(s,a)$ arbitrarily, except $q(s,a) = 0$ for terminal s

- repeat
  - using ε-greedy(q), generate episode
    $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, \ldots, S_{T-1}, A_{T-1}, R_T, \underbrace{S_T}_{\text{terminal}}$
  - for $t = T-1 \ldots 0$
    - $$\hat{G}_t^n = \begin{cases} R_{t+1} + \gamma v(S_{t+1}) & \text{if } n = 1 \\ R_{t+1} + \gamma \hat{G}_{t+1}^{n-1} & \text{if } 2 \le n \le T-t \end{cases}$$
    - $$G_t^\lambda = (1-\lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} \hat{G}_t^n + \lambda^{T-t-1} \hat{G}_t^{T-t}$$
    - $$q(S^t, A^t) \leftarrow q(S^t, A^t) + \alpha\left(G_t^\lambda - q(S^t, A^t)\right)$$

What is the drawback of this algorithm?

λ-Return as Forward View Algorithm

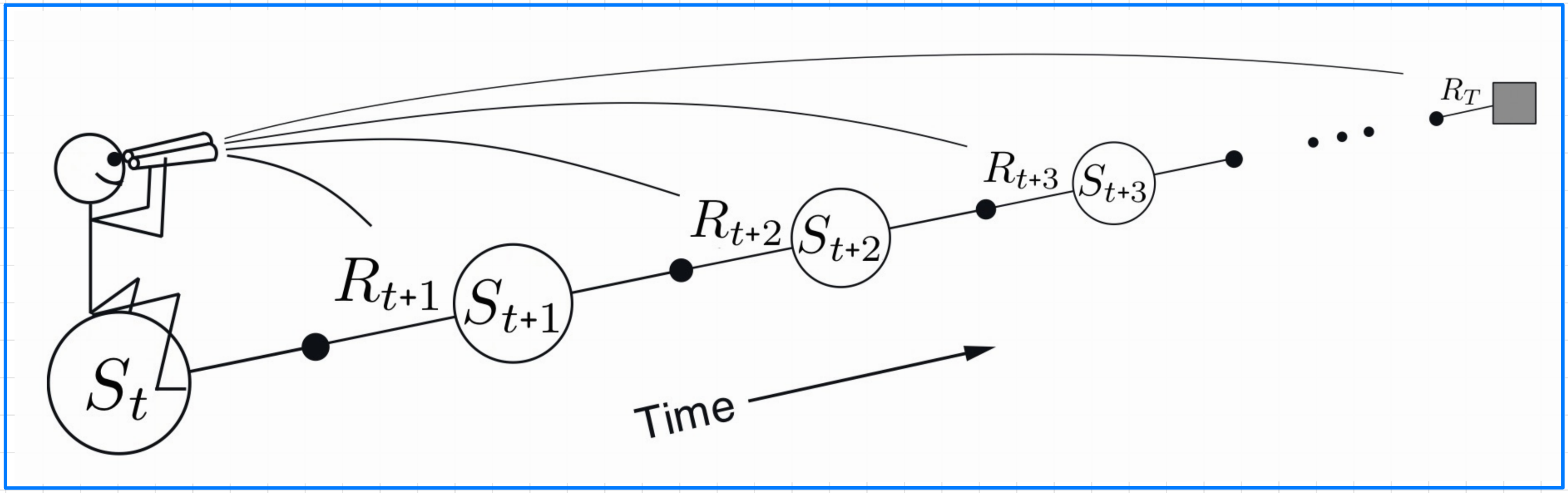


Image: Sutton & Barto

- λ-return is a "forward view" algorithm
- we need to wait until the end of the episode
- same disadvantage as MC (since G_t^λ contains G_t^∞)

Is there a Backward View?

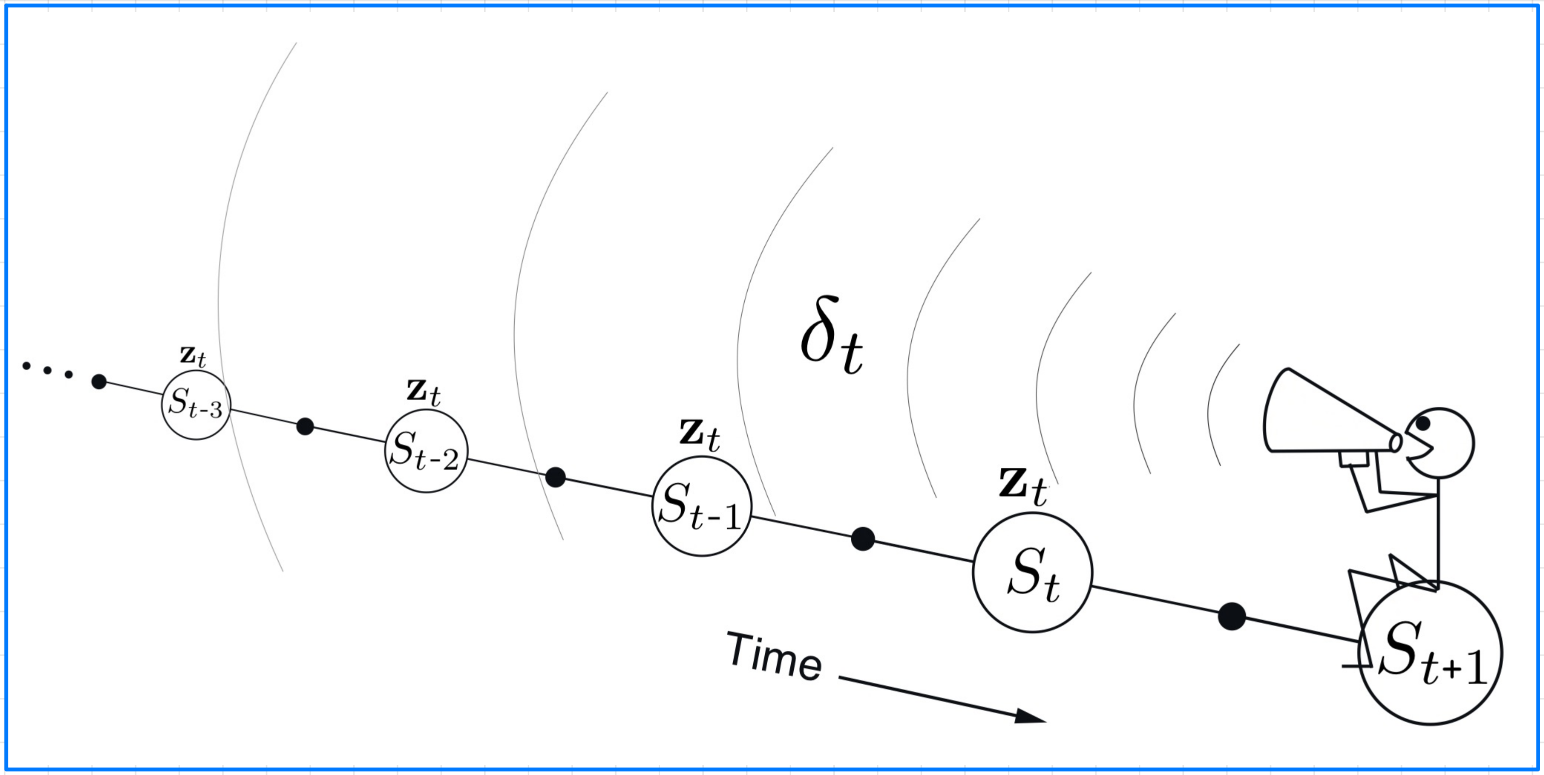
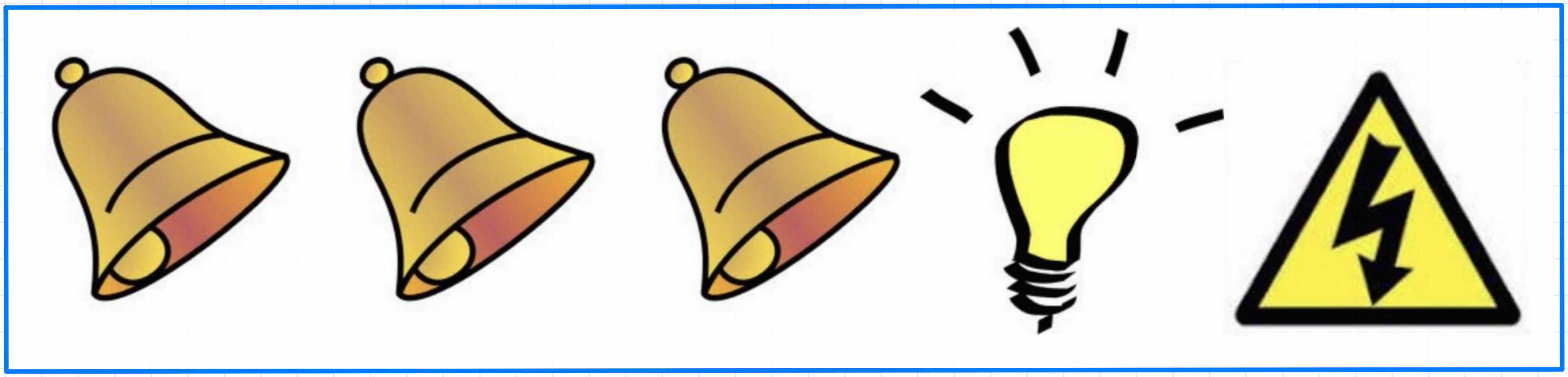


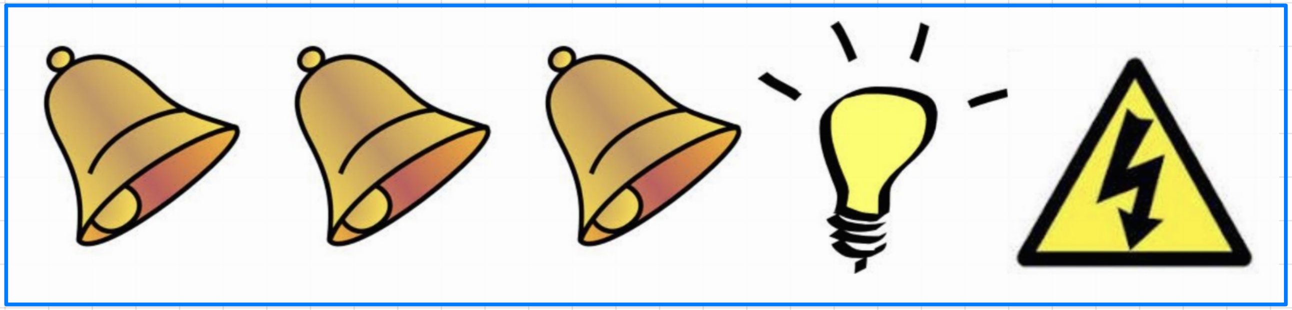
Image: Sutton & Barto

Eligibility Traces

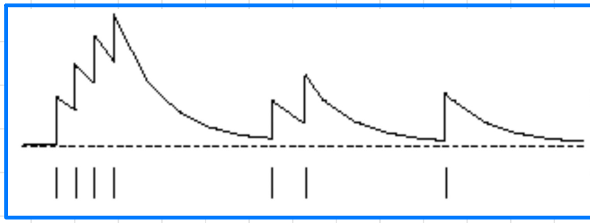
Do the bells or the light cause the shock?



Eligibility Traces



- did the bells or the light cause the electric shock?
- bells: more frequent
  light: more recent
- heuristic credit assignment
- eligibility traces combine both heuristics



$t=0$: $\quad e(s) \leftarrow 0$

$t>0$: $\quad e(s) \leftarrow \lambda\gamma e(s) + \mathbb{1}[S_t = s]$

exponential decay

"pumps up by 1" if state is visited

Images: D. Silver

TD(λ)

"One the oldest and most widely used algorithms in RL"
Sutton & Barto, p. 292

Given: π // TD(λ) is a prediction algorithm
α, λ, γ
initialize $v(s)$ arbitrarily $\forall s$

repeat until v has converged
$e(s) \leftarrow 0 \quad \forall s$
initialize S
repeat until S is terminal or v has converged
$A \sim \pi(a|S)$
observe S', R // $\sim p(s', a | S, A)$
$e(s) \leftarrow \lambda \gamma e(s) + \mathbb{1}[S = s] \quad \forall s$
$\delta \leftarrow R + \gamma v(S') - v(S)$ // TD error
$v(s) \leftarrow v(s) + \alpha \delta e(s) \quad \forall s$
$S \leftarrow S'$

Forward λ-Return vs TD(λ)

Update using λ-return:

$$V(S_t) \leftarrow V(S_t) + \alpha\big(\underbrace{G_t^\lambda - V(S_t)}_{\text{error term}}\big)$$

It can be shown that:

$$G_t^\lambda - V(S_t) = \sum_{t=1}^{\infty} \underbrace{(\lambda\gamma)^{t-1}}_{\text{eligibility trace } e} \underbrace{\big(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\big)}_{\text{TD error } \delta}$$

Hence, λ-return and TD(λ) are almost the same algorithm!
TD(λ) updates immediately, while λ-return waits until end of episode. Accumulating δe in TD(λ) until end of episode before updating would make the two equivalent

- TD(0) is TD
- TD(1) is akin to MC, but with immediate updates

Forward λ-Return vs TD(λ)

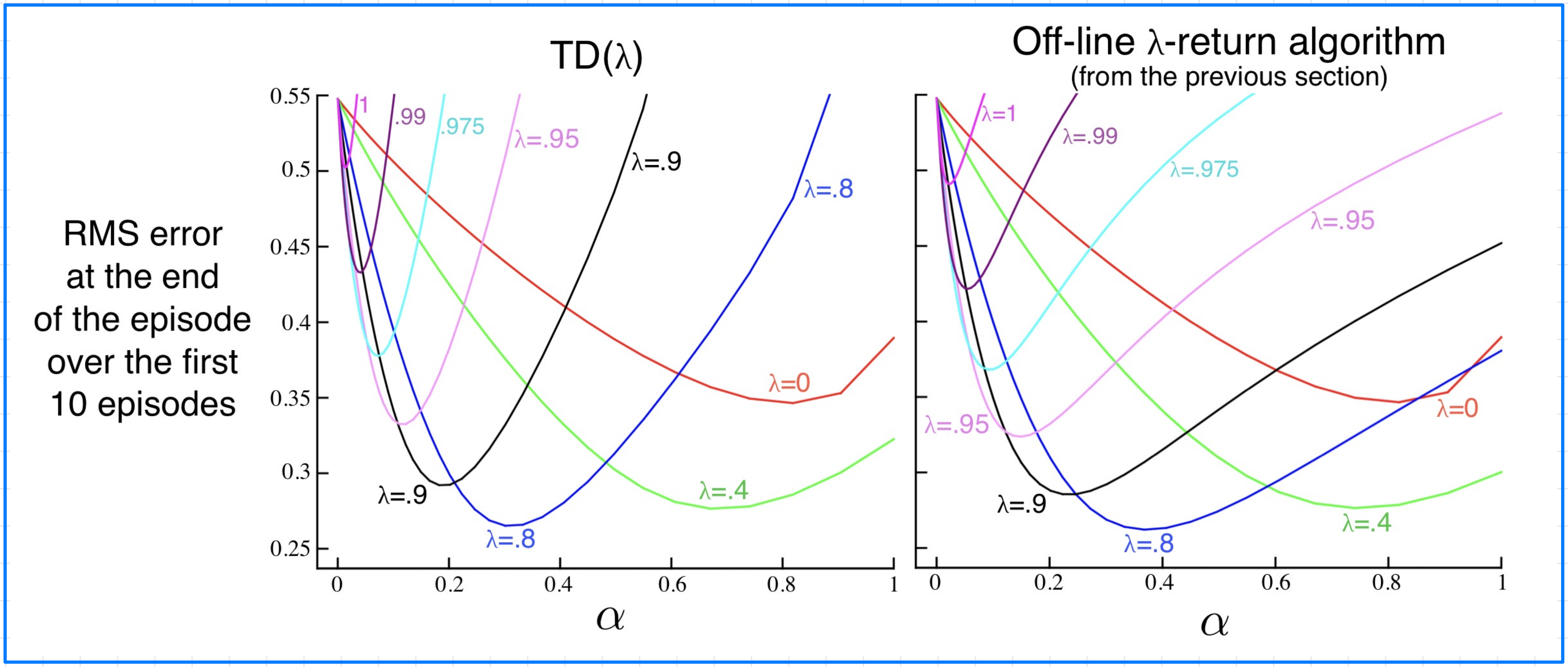


Image: Sutton & Barto

Sarsa(λ)

Given: α, λ

initialize $q(s,a)$ arbitrarily $\quad \forall s,a$

repeat until q has converged

$\quad e(s,a) \leftarrow 0 \quad \forall s,a$

$\quad$ initialize S

$\quad A \sim \varepsilon\text{-greedy}(q(S,\cdot))$

$\quad$ repeat until S is terminal or q has converged

$\quad\quad$ observe $S', R \quad // \sim p(s', r \mid S, A)$

$\quad\quad A' \sim \varepsilon\text{-greedy}(q(S',\cdot))$

$\quad\quad e(s,a) \leftarrow \lambda\gamma e(s,a) + \mathbb{1}[S=s \wedge A=a] \quad \forall s,a$

$\quad\quad \delta \leftarrow R + \gamma q(S',A') - q(S,A) \quad$ // TD error

$\quad\quad q(s,a) \leftarrow q(s,a) + \alpha\delta e(s,a) \quad \forall s,a$

$\quad\quad S, A \leftarrow S', A'$

Eligibility Trace of Sarsa(λ)

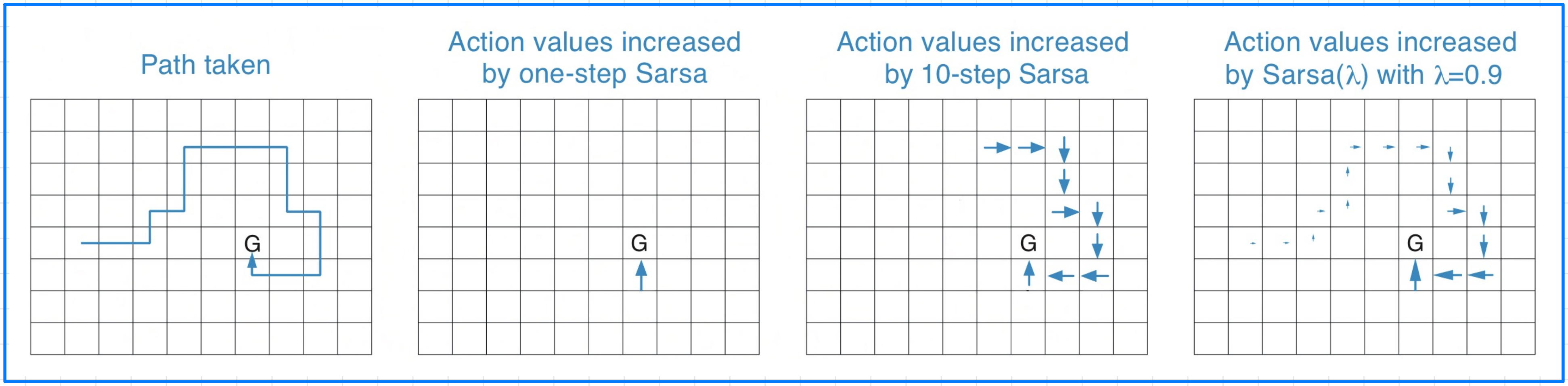
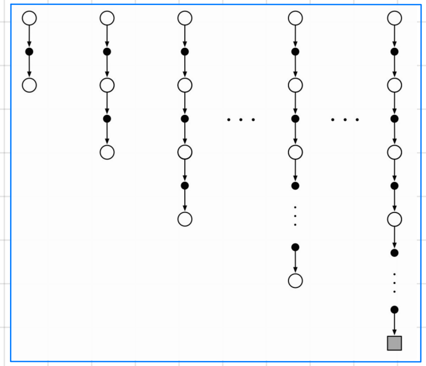


Image: Sutton & Barto

Summary

n-step return

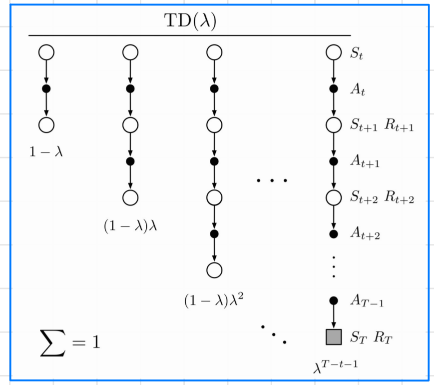
$$\hat{G}_t^n := \sum_{k=0}^{n-1} \gamma^k R_{t+k+1} + \gamma^n v(S_{t+n})$$

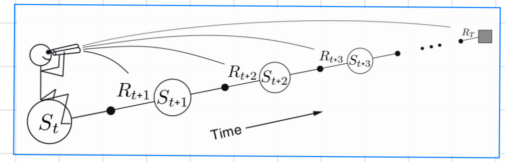


λ-return algorithm

$$G_t^\lambda = (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} \hat{G}_t^n$$

$$v(s) \leftarrow v(s) + \alpha \left(G_t^\lambda - v(s) \right)$$

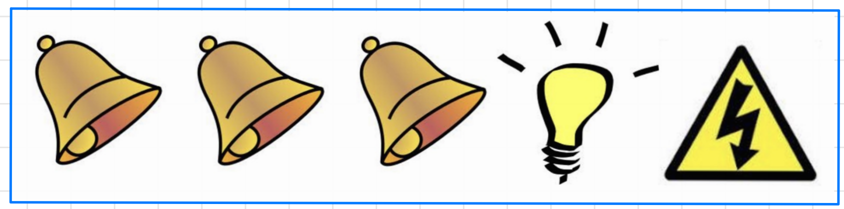


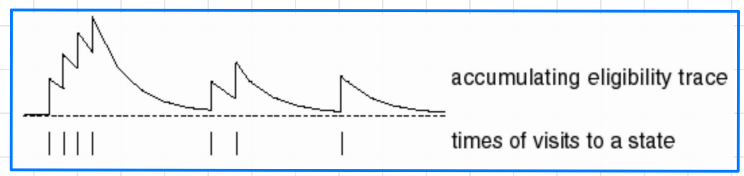


eligibility traces

$t = 0: \quad e(s) \leftarrow 0$

$t > 0: \quad e(s) \leftarrow \lambda \gamma e(s) + \mathbb{1}[S_t = s]$





$$e(s) \leftarrow \lambda \gamma e(s) + \mathbb{1}[S = s] \quad \forall s$$

$$\delta \leftarrow R + \gamma v(S') - v(S)$$

$$v(s) \leftarrow v(s) + \alpha \delta e(s)$$