Eligibility Traces

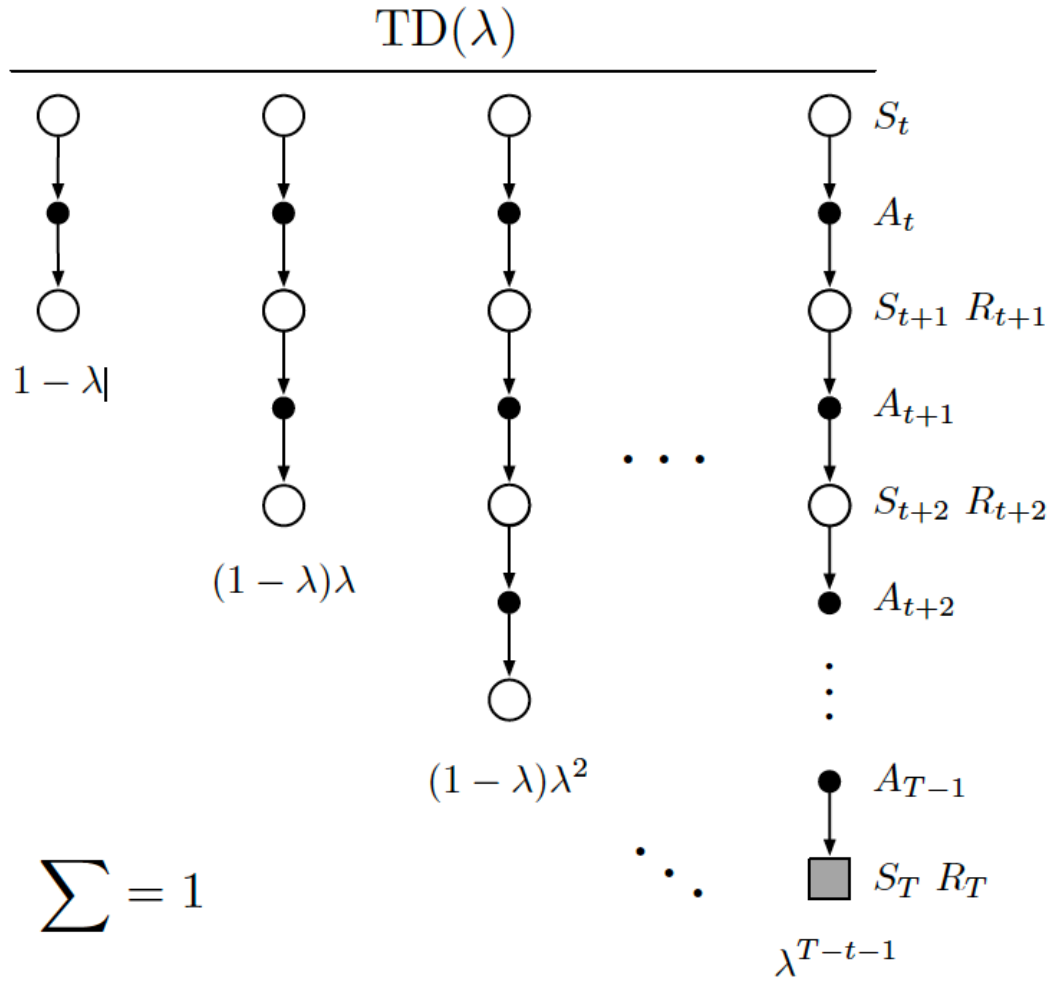
- Unify and generalize TD (λ=0) and Monte Carlo (λ=1) methods
- Involved a short-term memory vector, the eligibility trace z_t and long-term weight vector w_t and trace-decay parameter λ.
- When a component of w_t participates in producing an estimated value, then the corresponding component of z_t is bumped up and then begins to fade away determined by λ.
- Learning will then occur in that component of w_t if a nonzero TD error occurs before the trace falls back to zero.
- Computational advantage over n-step TD methods as only a single trace vector is required rather than a store of the last n feature vectors.

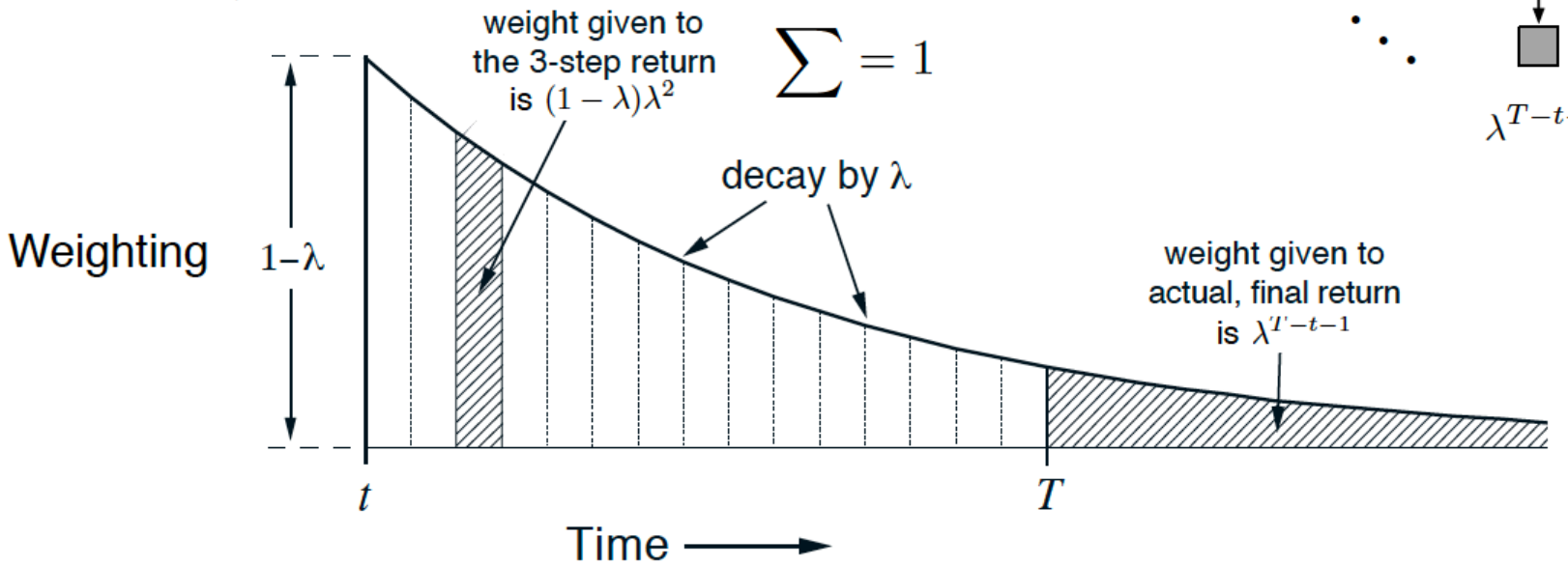
The λ-return

- TD(λ) using an λ-return, an average of n-step returns:

$$G_t^\lambda \doteq (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}G_{t:t+n}.$$

$$G_t^\lambda = (1-\lambda)\sum_{n=1}^{T-t-1}\lambda^{n-1}G_{t:t+n} + \lambda^{T-t-1}G_t,$$



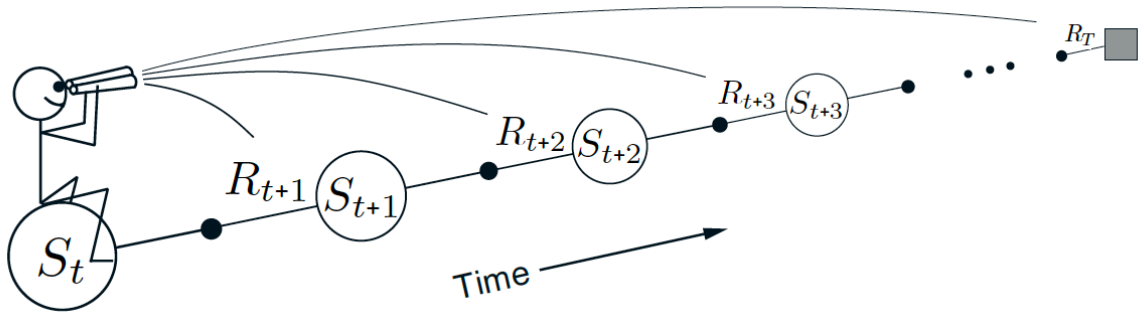


Offline λ-return algorithm

- Updates are done at the end of an episode

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t + \alpha \Big[G_t^\lambda - \hat{v}(S_t, \mathbf{w}_t) \Big] \nabla \hat{v}(S_t, \mathbf{w}_t), \quad t = 0, \ldots, T-1.$$

- Theoretical or forward view
  - We look forward in time to all the future rewards and decide how best to combine them.

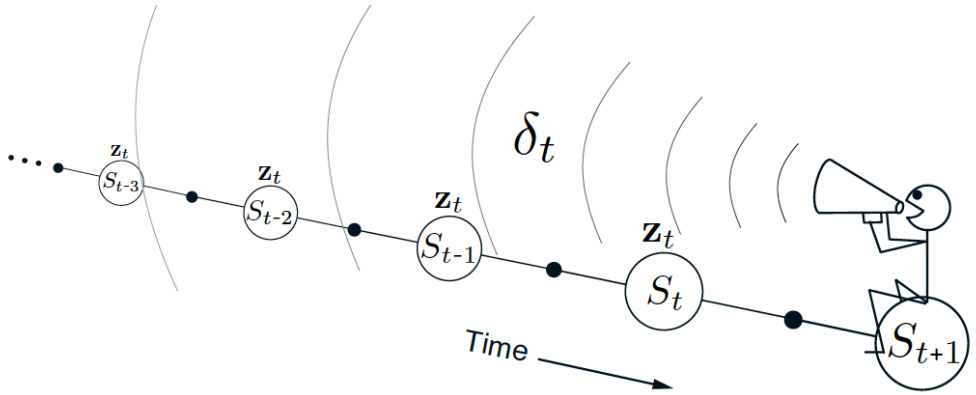


TD(λ)

- Update weight vector every step of an episode rather than only at the end so can be used online/continuous tasks with incremental updates.
- The eligibility trace vector is initialized to zero at the beginning of the episode, is incremented on each time step by the value gradient, and then fades away:

$$\mathbf{z}_{-1} \doteq \mathbf{0},$$
$$\mathbf{z}_t \doteq \gamma\lambda\mathbf{z}_{t-1} + \nabla\hat{v}(S_t,\mathbf{w}_t), \quad 0 \leq t \leq T,$$

- Backward view:
  - At each moment we look at the current TD error and assign it backward to each prior state according to how much that state contributed to the current eligibility trace at that time.



Semi-gradient TD(λ) for estimating $\hat{v} \approx v_\pi$

Input: the policy π to be evaluated
Input: a differentiable function $\hat{v} : \mathcal{S}^+ \times \mathbb{R}^d \to \mathbb{R}$ such that $\hat{v}(\text{terminal},\cdot) = 0$
Algorithm parameters: step size $\alpha > 0$, trace decay rate $\lambda \in [0, 1]$
Initialize value-function weights $\mathbf{w}$ arbitrarily (e.g., $\mathbf{w} = \mathbf{0}$)

Loop for each episode:
    Initialize S
    $\mathbf{z} \leftarrow \mathbf{0}$ (a d-dimensional vector)
    Loop for each step of episode:
    | Choose $A \sim \pi(\cdot|S)$
    | Take action A, observe R, S'
    | $\mathbf{z} \leftarrow \gamma\lambda\mathbf{z} + \nabla\hat{v}(S,\mathbf{w})$
    | $\delta \leftarrow R + \gamma\hat{v}(S',\mathbf{w}) - \hat{v}(S,\mathbf{w})$
    | $\mathbf{w} \leftarrow \mathbf{w} + \alpha\delta\mathbf{z}$
    | $S \leftarrow S'$
    until S' is terminal