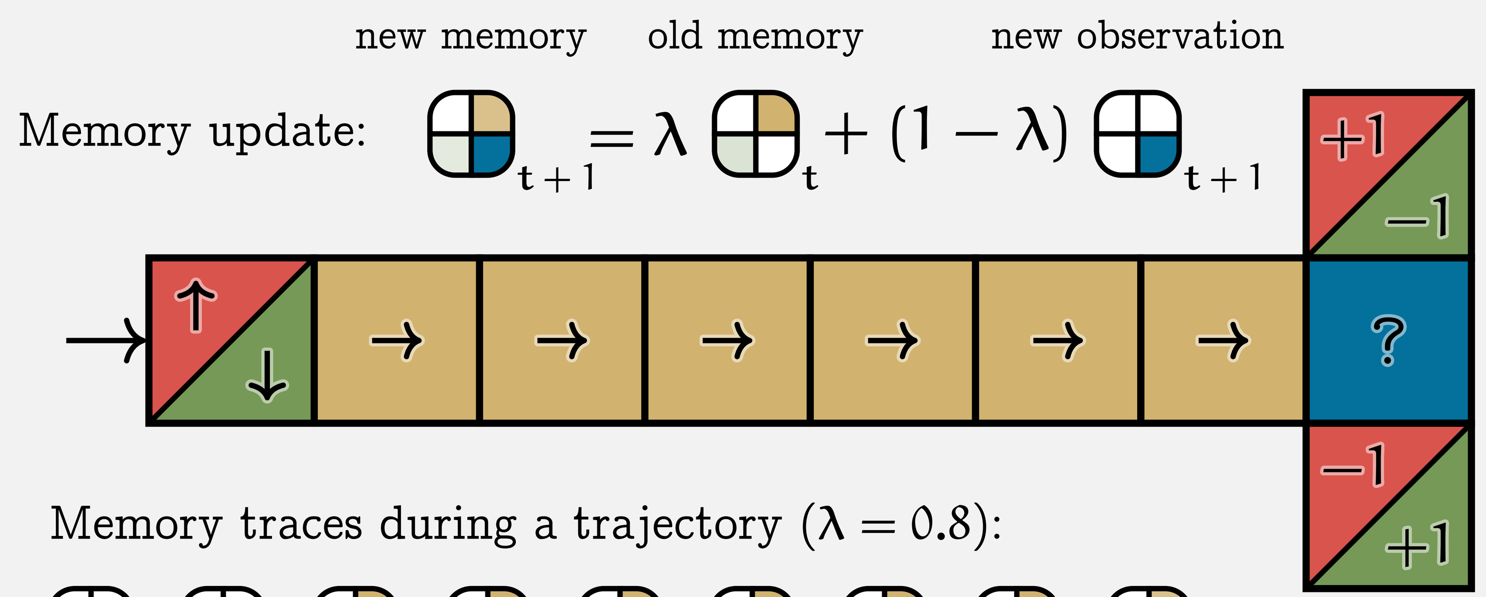
# Partially Observable Reinforcement Learning with Memory Traces

Onno Eberhard[1,2] Michael Muehlebach[1] Claire Vernade[2]

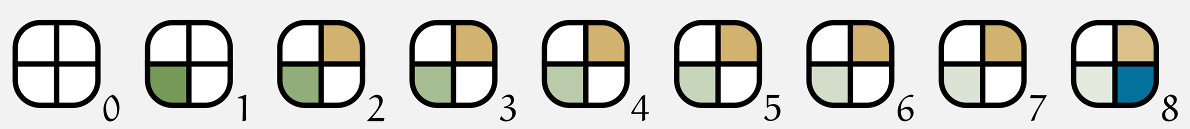
[1]Max Planck Institute for Intelligent Systems [2]University of Tübingen

## Eligibility traces are more effective than sliding windows as a memory mechanism for RL in POMDPs.

## Motivation & memory




Memory traces during a trajectory ($\lambda = 0.8$):




- Memory is necessary in many partially observable environments
- Length-$m$ window: $\text{win}_m(y_t, y_{t-1}, \dots) \doteq (y_t, y_{t-1}, \dots, y_{t-m+1})$
- *Memory trace* with forgetting factor $\lambda \in [0, 1)$:

$$z_\lambda(y_t, y_{t-1}, \dots) = \lambda z_\lambda(y_{t-1}, y_{t-2}, \dots) + (1-\lambda) y_t$$

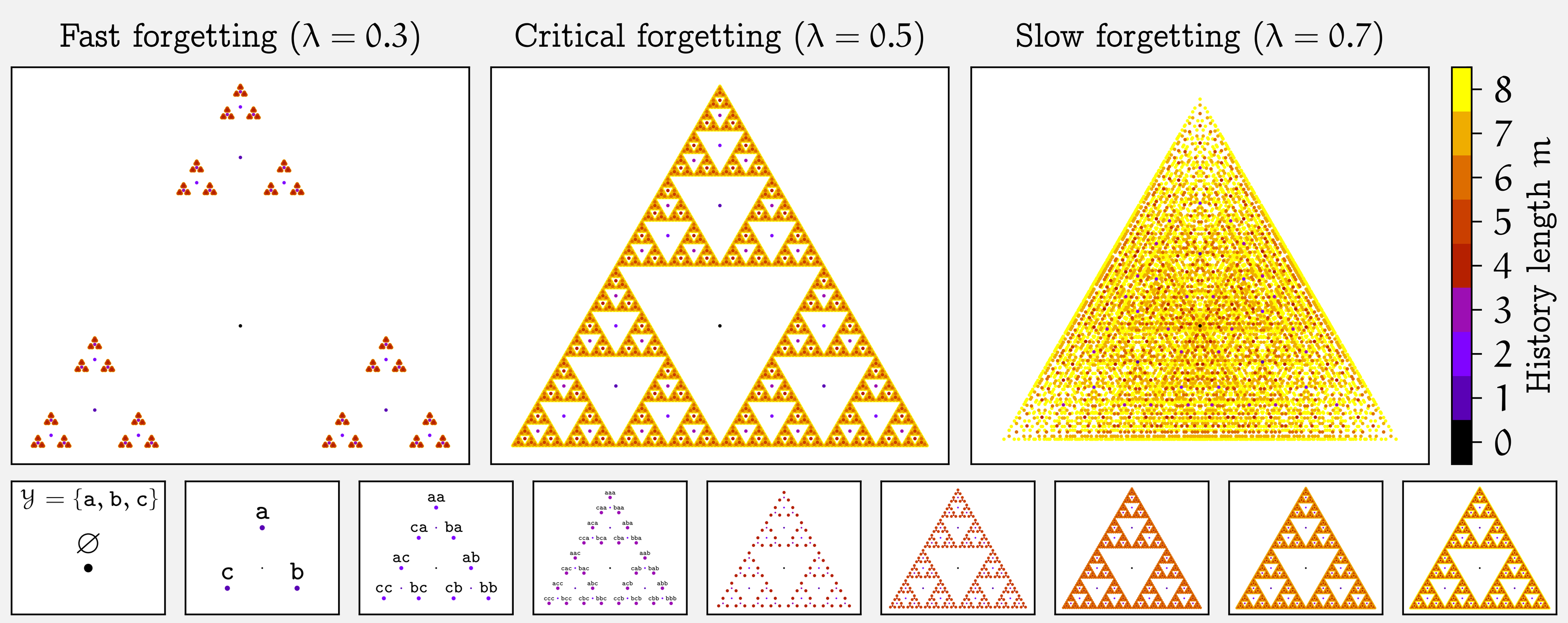
## POMDPs & value functions

- We consider the problem of *policy evaluation* with offline data
  - $\rightarrow$ Environment $\mathcal{E}$ is a hidden Markov model, observation space $\mathcal{Y}$ is one-hot
- Q: How much data do we need to accurately estimate the value function?
- Goal: given a function class $\mathcal{F} \subset \{\mathcal{Y}^\infty \to [\underline{v}, \overline{v}]\}$, find $f \in \mathcal{F}$ that minimizes

$$\mathcal{R}_\mathcal{E}(f) \doteq \mathbb{E}_\mathcal{E}\left[\left\{f(y_0, y_{-1}, \dots) - \sum_{t=0}^{\infty} \gamma^t r(y_{t+1})\right\}^2\right].$$

- Length-$m$ window: $\mathcal{F}_m \doteq \{f \circ \text{win}_m \mid f : \mathcal{Y}^m \to [\underline{v}, \overline{v}]\}$
- Memory traces: $\mathcal{F}_\lambda \doteq \{f \circ z_\lambda \mid f : \mathcal{Z}_\lambda \to [\underline{v}, \overline{v}]\}$, where $\mathcal{Z}_\lambda \doteq \{z_\lambda(h) \mid h \in \mathcal{Y}^\infty\}$
- Learning theory: learning is easier if the *metric entropy* $H_\epsilon(\mathcal{F})$ is small
- For windows, we have $\boxed{H_\epsilon(\mathcal{F}_m) \in \Theta(|\mathcal{Y}|^m)}$ $\rightarrow$ long windows are expensive!

## The geometry of *trace space*




- Memory traces with forgetting factor $\lambda < \frac{1}{2}$ remember everything
  - $\rightarrow z_\lambda$ is invertible, and therefore $\boxed{H_\epsilon(\mathcal{F}_\lambda) = \infty}$
- Need to "zoom in" to differentiate histories that only differ far in the past
- The "resolution" of a function class is given by its Lipschitz constant
- We consider the class $\mathcal{F}_{\lambda,L} \doteq \{f \circ z_\lambda \mid f : \mathcal{Z}_\lambda \to [\underline{v}, \overline{v}], f \text{ is } L\text{-Lipschitz}\}$
- We have $\boxed{H_\epsilon(\mathcal{F}_{\lambda,L}) \in \mathcal{O}(L^{\min\{d_\lambda, |\mathcal{Y}|-1\}}),}$ where $d_\lambda \doteq \frac{\log|\mathcal{Y}|}{\log(1/\lambda)}$

## Fast forgetting: $\lambda < 1/2$

### Theorem (window $\rightarrow$ trace)

Windows are not more efficient than memory traces.

Let $m \in \mathbb{N}$ be a window length, $0 < \lambda < \frac{1}{2}$ a forgetting factor, and define

$$L(m) = \frac{\overline{v} - \underline{v}}{\sqrt{2}(1-2\lambda)\lambda^{m-1}}.$$

Then, for every $\epsilon > 0$ and every environment $\mathcal{E}$,

$$\mathcal{R}_\mathcal{E}(\mathcal{F}_{\lambda,L(m)}) \leqslant \mathcal{R}_\mathcal{E}(\mathcal{F}_m) \quad \text{and} \quad H_\epsilon(\mathcal{F}_{\lambda,L(m)}) \in \mathcal{O}(|\mathcal{Y}|^m) = \mathcal{O}(H_\epsilon(\mathcal{F}_m)).$$

### Theorem (trace $\rightarrow$ window)

Memory traces with $\lambda < \frac{1}{2}$ seem no more efficient than windows.

Let $\lambda \in [0, 1)$ be a forgetting factor, $L > 0$ a Lipschitz constant, $\epsilon \in (0, L)$, and define

$$m(\lambda, L) = \left\lceil \frac{\log(L/\epsilon)}{\log(1/\lambda)} \right\rceil.$$

Then, for every environment $\mathcal{E}$,

$$\mathcal{R}_\mathcal{E}(\mathcal{F}_{m(\lambda,L)}) \leqslant \mathcal{R}_\mathcal{E}(\mathcal{F}_{\lambda,L}) + \mathcal{O}(\epsilon) \quad \text{and} \quad H_\epsilon(\mathcal{F}_{m(\lambda,L)}) \in \mathcal{O}(L^{d_\lambda}).$$

If $\lambda < \frac{1}{2}$, then $d_\lambda < |\mathcal{Y}| - 1$.

- Learning with windows and memory traces ($\lambda < \frac{1}{2}$) seems equivalent!

## Slow forgetting: $\lambda \geqslant 1/2$

### Theorem (T-maze)

Memory traces ($\lambda \geqslant \frac{1}{2}$) can be significantly more efficient than windows.

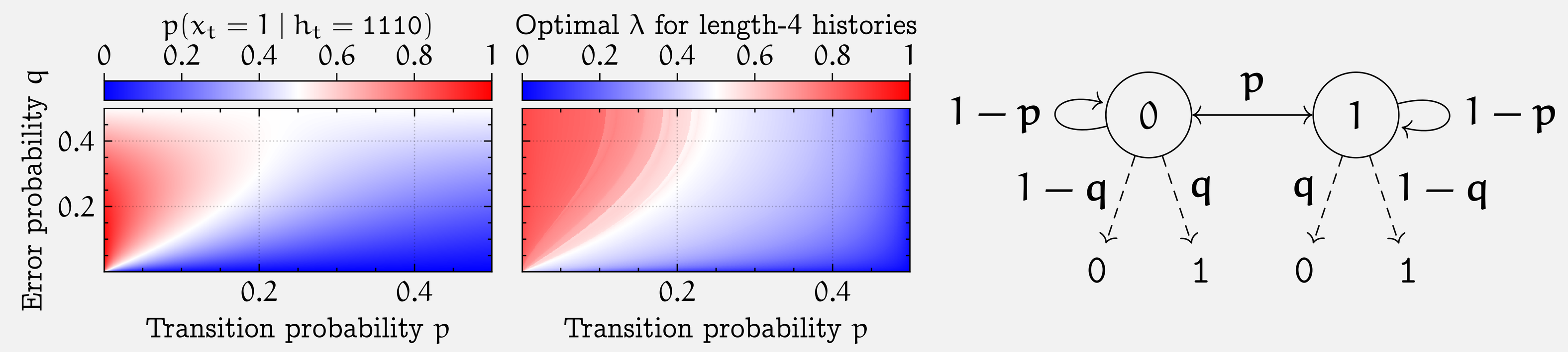
There exists a sequence $(\mathcal{E}_k)$ of environments (with constant observation space $\mathcal{Y}$) with the property that, for every $\epsilon > 0$,

$$\min_{m \in \mathbb{N}} \{H_\epsilon(\mathcal{F}_m) \mid \mathcal{R}_{\mathcal{E}_k}(\mathcal{F}_m) = 0\} \in \Omega(|\mathcal{Y}|^k), \text{ and}$$

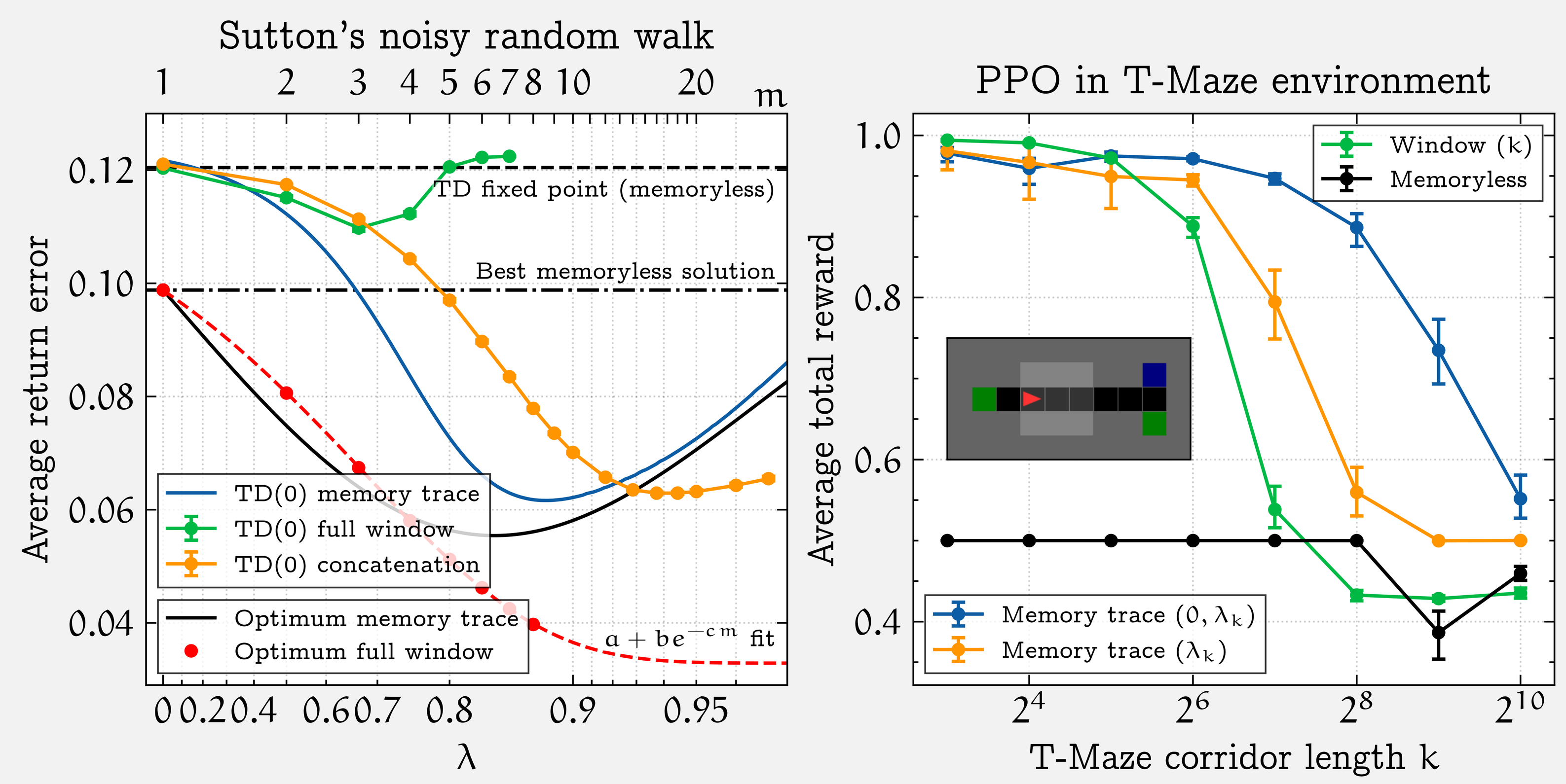
$$\min_{\lambda \in [0,1)} \min_{L \geqslant 0} \{H_\epsilon(\mathcal{F}_{\lambda,L}) \mid \mathcal{R}_{\mathcal{E}_k}(\mathcal{F}_{\lambda,L}) = 0\} \in \mathcal{O}(k^{|\mathcal{Y}|-1}).$$

In particular, the *T-maze* with corridor length $k$ is such a sequence. In this case, the minima are attained at $m_k = k$, $\lambda_k = \frac{k-1}{k}$, and $L_k \leqslant \sqrt{2}ek$.

- In the T-maze, most of the $|\mathcal{Y}|^k$ histories are irrelevant
  - $\rightarrow$ Can map these to arbitrary values, allows for larger Lipschitz constant
- In other environments, memory traces can effectively smooth out noise




## Experiments




- Memory traces are an effective drop-in replacement for frame stacking

Paper
Code
More