ICML 2020

Discount Factor as a Regularizer in RL

Ron Amit , Ron Meir (Technion) , Kamil Ciosek (MSR)

TECHNION
Israel Institute of Technology

(RL)²
Reinforcement Learning Research Labs

Microsoft Research, Cambridge UK

Microsoft

RL problems objectives

- The expected γ_e-discounted return (value function)

$$V^{\pi}_{\gamma_e}(s) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma_e{}^t r_t | s_0 = s, \pi] \quad \gamma_e \in (0,1]$$

Evaluation discount factor

- Policy Evaluation $\min_{\hat{V}} \|V^{\pi}_{\gamma_e} - \hat{V}\|$

- Policy Optimization $\max_{\pi} V^{\pi}_{\gamma_e}(s)$

How can we improve perfomance in the limited data regime?

Discount regularization

- *Discount regularization*: $0 \leq \gamma \leq \gamma_e$ *"guidance discount factor"* (Jiang '15)

  Algorithm hyperparameter
- Theoretical analysis:
  - Petrik and Scherrer '09 – Approx. DP
  - Jiang '15 – model based

  Better performance for limited data
- Regularization effect:
  - ↑ Bias $\|V_\gamma - V_{\gamma_e}\|$
  - ↓ Variance $\|\hat{V} - V_\gamma\|$
- Our work:
  - **In TD learning, discount regularization == explicit added regularizer**
  - ***When is discount regularization effective?***

Temporal Difference (TD) Learning

- Policy evaluation with value-function model $\hat{V}_\theta(s)$
- Batch TD(0)

Input: D data batch
for $i = 0, 1, \ldots, N_{\text{iter}} - 1$ **do**
  Pick at random (s, a, r, s') from D
  $\theta_{i+1} := \theta_i + \alpha_i \left(r + \gamma \hat{V}_{\theta_i}(s') - \hat{V}_{\theta_i}(s) \right) \nabla \hat{V}_{\theta_i}(s)$
end for

Discount factor hyperparameter

Aim to minimize $\mathbb{E}_{s \sim \hat{D}} \left(r + \gamma \hat{V}_\theta(s') - \hat{V}_\theta(s) \right)^2$

Equivalent Form

- Equivalent update steps

$$0 \le \gamma \le \gamma_e$$

Discount regularization (using $\gamma < \gamma_e$)

$$\theta \leftarrow \theta + \alpha\Big(r + \gamma\hat{V}_\theta(s') - \hat{V}_\theta(s)\Big)\nabla\hat{V}_\theta(s)$$

$$\Updownarrow$$

$$\alpha' = \gamma/\gamma_e \cdot \alpha \qquad \lambda = \frac{\gamma_e - \gamma}{2\gamma}$$

Using γ_e + regularization term

$$\theta \leftarrow \theta + \alpha'\Big(\xi r + \gamma_e\hat{V}_\theta(s') - \hat{V}_\theta(s)\Big)\nabla\hat{V}_\theta(s) - \alpha'\nabla\Big(\lambda\hat{V}_\theta(s)\Big)^2$$

$$\xi = \gamma_e/\gamma$$

Regularization term gradient

Similar Equivalence
- (expected) SARSA
- LSTD

$$\mathbb{E}_{s\sim\hat{D}}\Big[\Big(\xi r + \gamma_e\hat{V}_\theta(s') - \hat{V}_\theta(s)\Big)^2 + \Big(\lambda\hat{V}_\theta(s)\Big)^2\Big]$$

Activation regularization

The Equivalent Regularizer

- *Activation regularization* $\mathbb{E}_{s\sim\hat{D}}\left(\hat{V}_\theta(s)\right)^2$

  L_2 *regularization* $||\theta||^2$

- Tabular case:

$$\hat{V}_\theta(s) := \theta_s$$

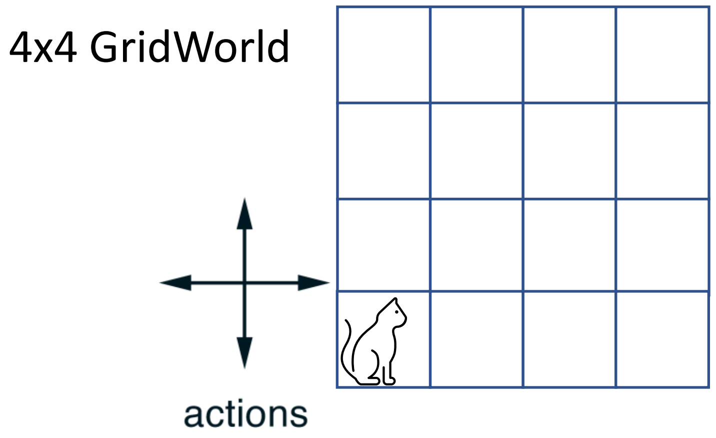
$$\mathbb{E}_{s\sim\hat{D}}\left(\hat{V}_\theta(s)\right)^2 = \sum_{s\in S}\hat{D}(s)\theta_s^2.$$

Discount regularization is sensitive to the empirical distribution

Tabular Experiments

- **Policy evaluation,** $\pi(a|s)$ uniform.
- **Goal**: find $\hat{V}$ that estimates $V^{\pi}_{\gamma_e}$ $(\gamma_e = 0.99)$
- **Loss measures:**
  - $\boldsymbol{L_2}$ **loss:** $\|\hat{V} - V^{\pi}_{\gamma_e}\|_2^2 = \sum_{s\in S}|\hat{V} - V^{\pi}_{\gamma_e}|^2$
  - **Ranking Loss:** $-\mathrm{Kandal`s\_Tau}(\hat{V}, V^{\pi}_{\gamma_e})$
    (~ number of order switches between state ranks)

- Average over 1000 MDP instances
- Data: trajectories of 50 time-steps



In each MDP Instance:

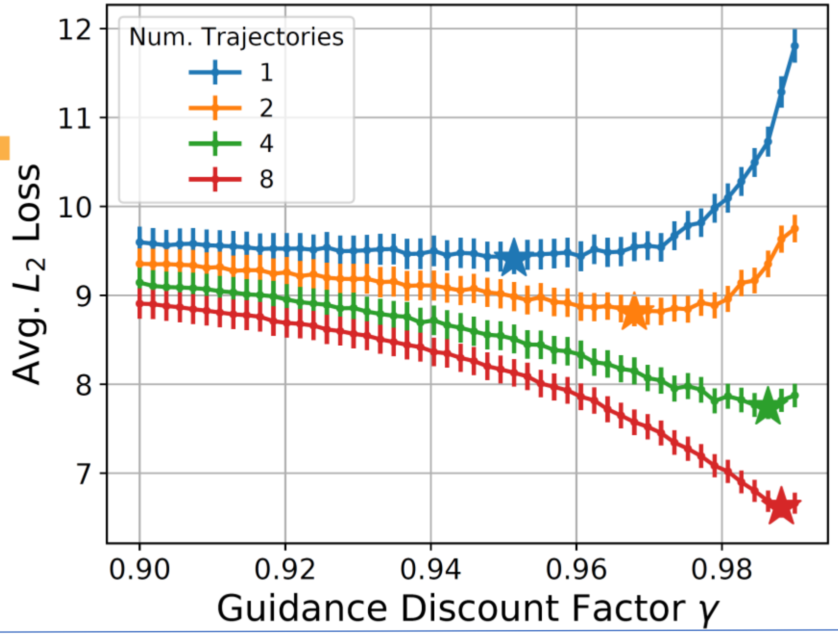
- Draw $\mathbb{E}R(s)$
- Draw $P(.|s,a)$

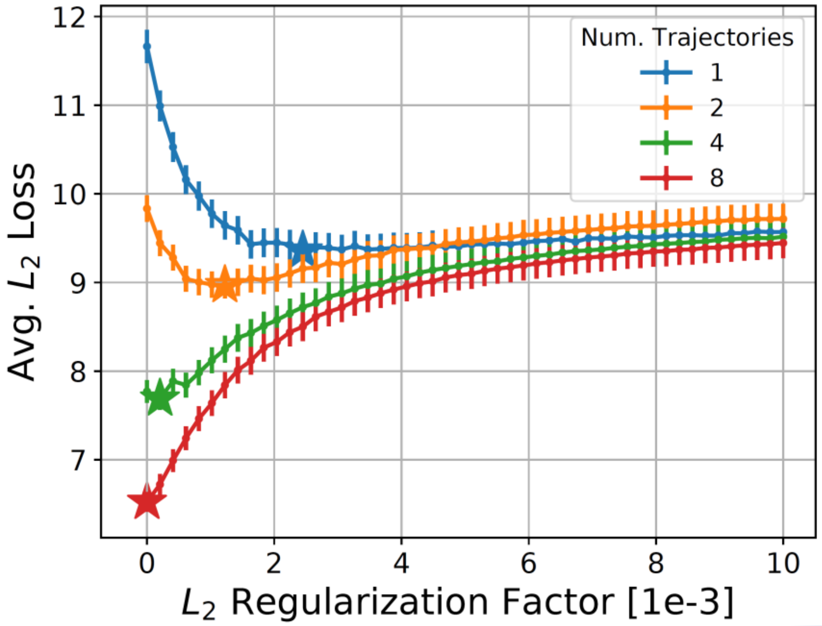
TD(0) Results

Discount Regularization

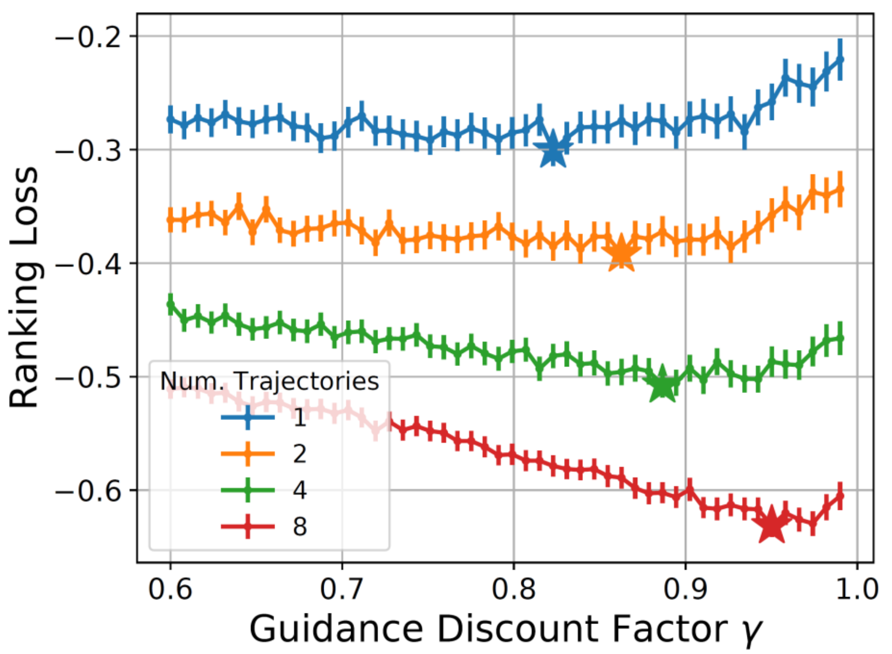
L_2 Regularization

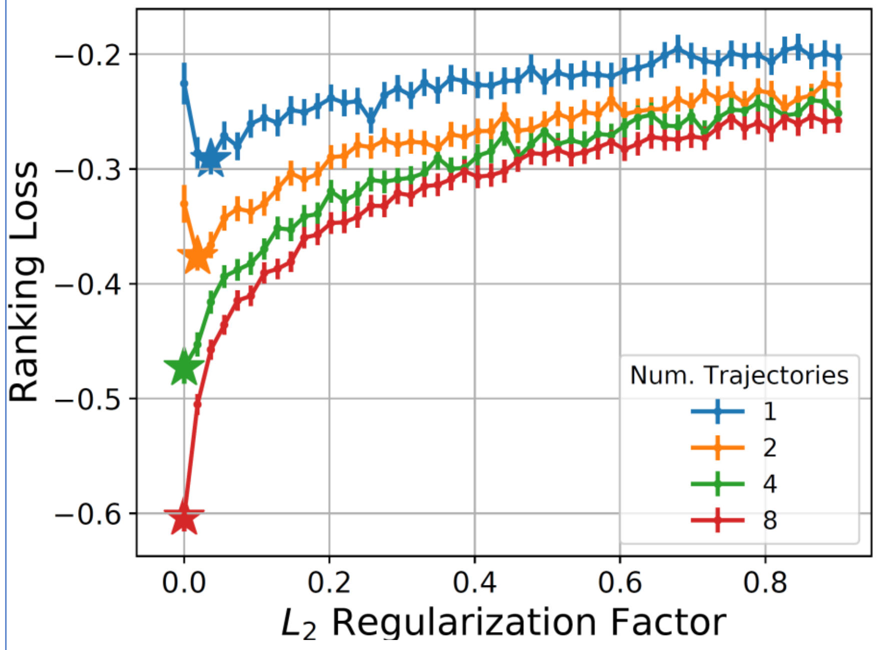
L_2 loss





Ranking Loss



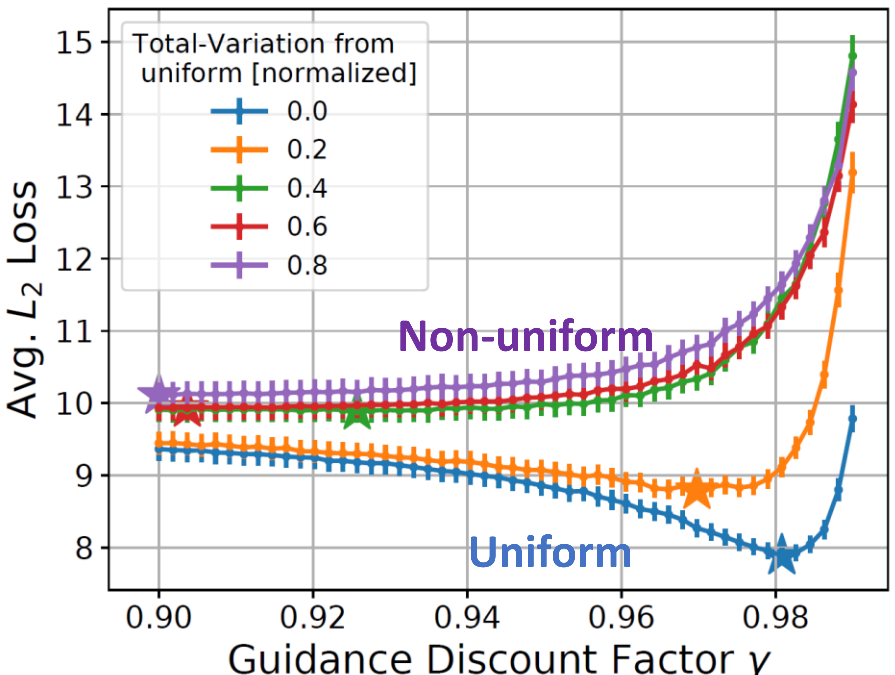


($\gamma_e = 0.99$)

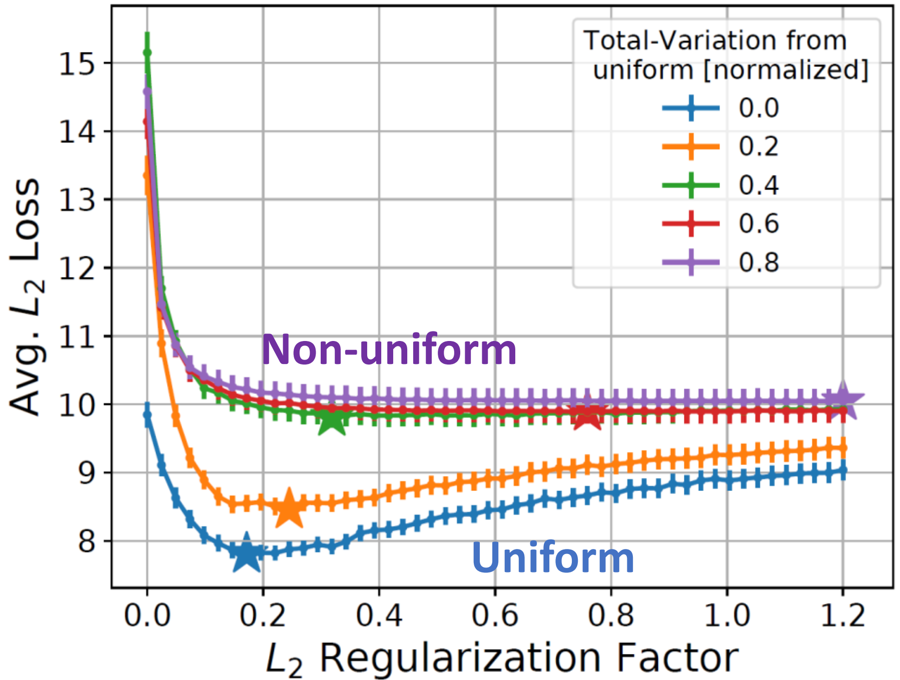
Effect of the Empirical Distribution

- Equivalent regularizer: $\mathbb{E}_{s\sim\hat{D}}\left(\hat{V}_\theta(s)\right)^2 = \sum_{s\in S}\hat{D}(s)\theta_s^2.$
- Tuples (s, s', r) generation: $s \sim g(s)$, $s' \sim P^\pi(s'|s), r \sim R^\pi(s)$
- For each MDP - draw distribution $g(s)$ at d_{TV} from uniform

Discount regularization



L_2 regularization

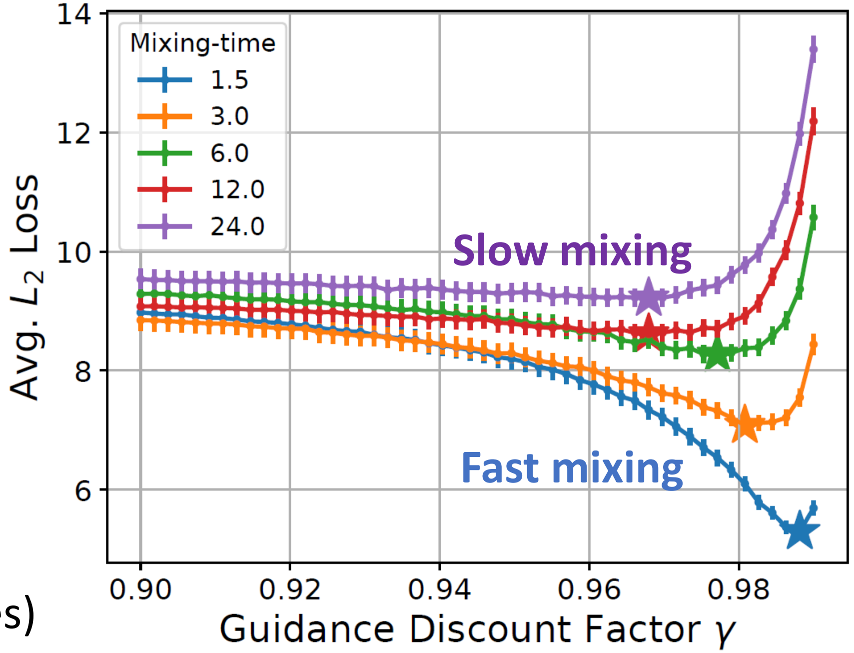


$(\gamma_e = 0.99)$

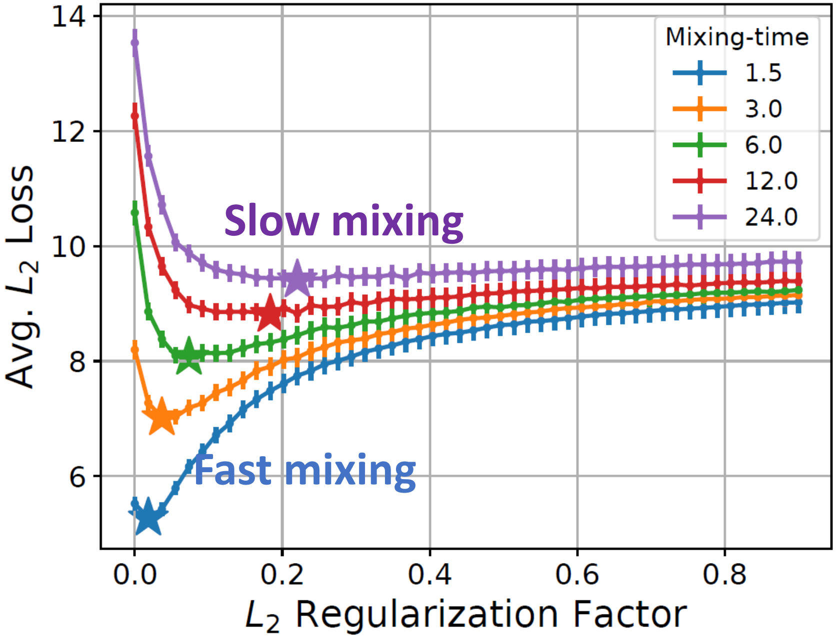
Effect of the Mixing Time

- **Lower mixing time (slow mixing) → Higher estimation variance → more regularization is needed**

Discount regularization



L_2 regularization



$(\gamma_e = 0.99)$
(LSTD, 2 trajectories)

Policy Optimization

Goal: $\min_{\pi} \left\| V_{\gamma_e}^{\pi} - V_{\gamma_e}^{\pi^*} \right\|_1$

Policy-iteration:

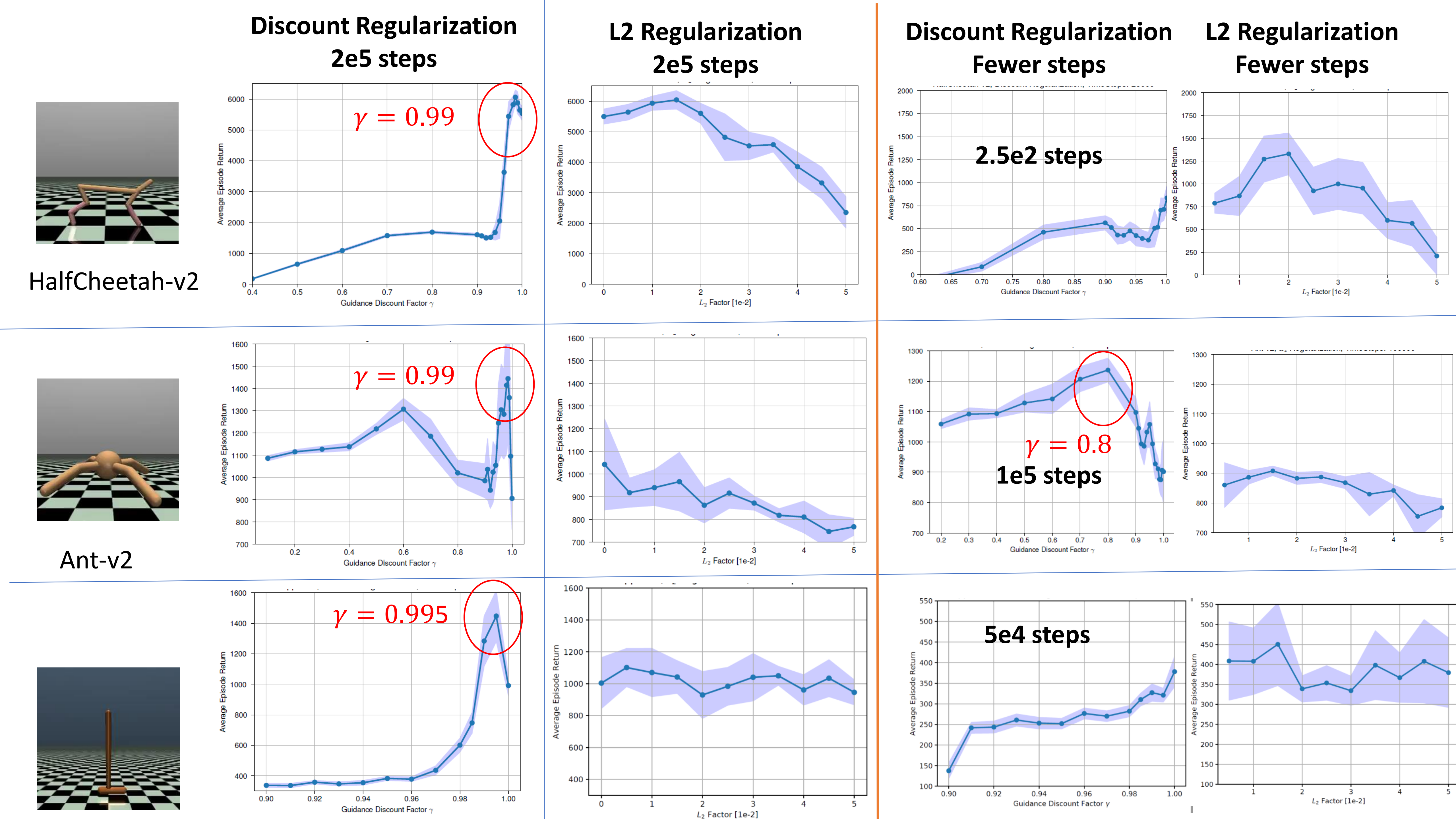
- For episodes:
  - Get data
  - $\hat{Q} \leftarrow$ Policy evaluation (e.g, SARSA)
  - Improvement step (e.g., ε-epsilon-greedy)

Activation regularization term:

$$\lambda \mathbb{E}_{(s,a)} \left(\hat{Q}_{\theta}(s,a) \right)^2$$

Deep RL Experiments

- Actor-critic algorithms: DDPG (Lillicrap '15), TD3 (Fujimoto '18)
- Mujoco continuous control (Todorov '12)
- **Goal:** undiscounted sum of rewards ($\gamma_e = 1$)
- Limited number of time-steps (2e5 or less)
- Tested cases:
  - Discount regularization (and no L_2)
  - L_2 regularization (and $\gamma = 0.999$)



Discount Regularization 2e5 steps
L2 Regularization 2e5 steps
Discount Regularization Fewer steps
L2 Regularization Fewer steps
HalfCheetah-v2
Ant-v2
γ = 0.99
γ = 0.99
γ = 0.995
γ = 0.8
2.5e2 steps
1e5 steps
5e4 steps
Average Episode Return
Guidance Discount Factor γ
L2 Factor [1e-2]

Conclusions

- Discount regularization in TD is equivalent to adding a regularizer term
- Regularization effectiveness is closely related to the data distribution and mixing rate.
- Generalization in deep RL is strongly affected by regularization
- Future work – theory needed

Thanks for listening