Stabilizing Temporal Difference Learning via Implicit Stochastic Recursion

Hwanwoo Kim

Department of Statistical Science, Duke University

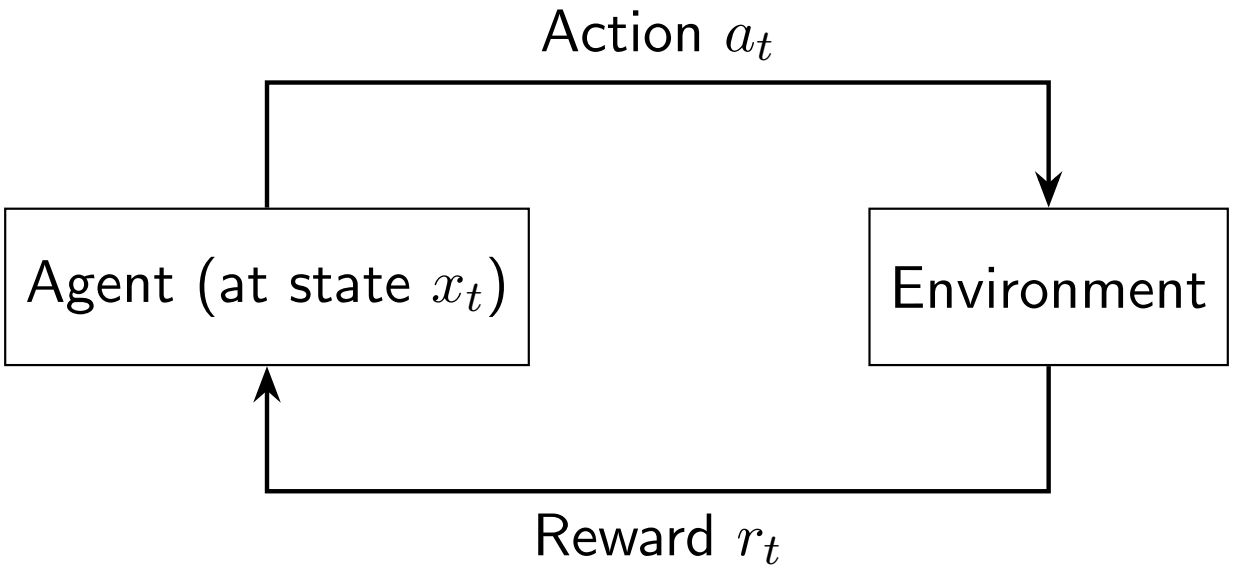
Joint work with Panos Toulis (U.Chicago), Eric Laber (Duke)

2025 IMS International Conference on Statistics and Data Science

Outline

Interactive Learning

Learning from interaction to achieve goals



Examples: Robot control, game playing and etc

Markov Decision Process (MDP)

Mathematical framework for sequential decision making

Components:

- **States** x_t: status at time t
- **Actions** $a_t = \pi(x_t)$: decision at time t
- **Rewards** $r_t = r(x_t, a_t)$: feedback from environment at time t
- **Transition kernel** $\mathrm{P}(x_{t+1}|x_t, a_t)$: governing rule of dynamics
- **Policy** π: a mapping from $\mathcal{X}$ to $\mathcal{A}$

Trajectory of Data: $(x_1, a_1, r_1, x_2, a_2, r_2, \ldots)$

Value Function

How to quantify goodness of the policy π?

$$V^{\pi}(x) = \mathbb{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, x_0 = x\right]$$

Expected total discounted reward when starting from state x and following policy π.

Why important?

- Summarizes long-term prospects of a policy from any state
- Enables comparison between different policies
- Foundation for optimal decision making

Policy Evaluation Problem

Task: Given a fixed policy π, estimate $V^\pi(x)$ for all states x

Two fundamental difficulties:

Difficulty 1: Computational Challenge

The expectation $\mathbb{E}^\pi[\sum_{t=0}^{\infty} \gamma^t r_t]$ requires knowing:

- Transition probabilities $\mathrm{P}(x_{t+1}|x_t, a_t)$ — typically unknown
- Summation over infinitely many future time steps

Difficulty 2: Scalability Challenge

Need to compute and store $V^\pi(x)$ for every state x:

- Infeasible when state space is large or continuous

Stochastic Approximation for Mean Estimation

Classical problem: Estimate $\mathbb{E}[X]$ from samples $x_0, x_1, x_2, \ldots$
Robbins-Monro stochastic approximation [Robbins and Monro, 1951]:

$$\mu_{m+1} = (1 - \alpha_m)\mu_m + \alpha_m x_m = \mu_m + \alpha_m(x_m - \mu_m)$$

Convergence theorem: If $\sum_{m \geq 0} \alpha_m = +\infty$ and $\sum_{m \geq 0} \alpha_m^2 < +\infty$, then

$$\mu_m \xrightarrow{a.s.} \mathbb{E}[X]$$

Intuition:

- Update current estimate toward new observation
- $(x_m - \mu_m)$ is the error between observation and current estimate
- Step size α_m controls influence of new data

Tabular TD Learning

Value functions satisfy a recursive relationship

$$V^\pi(x) = \mathbb{E}^\pi[r(x,a) + \gamma V^\pi(X') \mid x]$$

"Value at state x = immediate reward + discounted value of next state"

- Want to estimate $V^\pi(x)$ for each state $x \in \mathcal{X}$
- Each transition (x, a, x') provides: $r(x,a) + \gamma V^\pi(x')$
- This is a noisy sample of $V^\pi(x)$

Natural stochastic approximation update (tabular case):

$$V_{t+1}(x_t) = V_t(x_t) + \alpha_t \left[r_t + \gamma V_t(x_{t+1}) - V_t(x_t)\right]$$

$\Longrightarrow$ **Temporal Difference (TD) learning** [Sutton and Barto, 2018]

TD with Function Approximation

Motivation: Tabular TD doesn't scale: take parametric approximation

$$V^{\pi}(x) \approx f_{\mathbf{w}}(x) := \phi(x)^T \mathbf{w}, \quad \phi(x) = [\phi_1(x), \cdots, \phi_d(x)]^\top$$

for some basis functions $\{\phi_1(x), \cdots, \phi_d(x)\}$.
Semi-gradient descent on squared loss:

$$\begin{aligned}\mathbf{w}_{t+1} &= \mathbf{w}_t - \alpha \nabla_{\mathbf{w}_t} \frac{1}{2} \left[V^{\pi}(x_t) - f_{\mathbf{w}_t}(x_t)\right]^2 \\ &= \mathbf{w}_t - \alpha \left[V^{\pi}(x_t) - f_{\mathbf{w}_t}(x_t)\right] \nabla_{\mathbf{w}_t} f_{\mathbf{w}_t}(x_t)\end{aligned}$$

Replace unknown $V_{\pi}(x_t)$ with TD target $r_t + \gamma f_{\mathbf{w}_t}(x_{t+1})$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha \left[r_t + \gamma f_{\mathbf{w}_t}(x_{t+1}) - f_{\mathbf{w}_t}(x_t)\right] \nabla_{\mathbf{w}_t} f_{\mathbf{w}_t}(x_t)$$

- **Benefit:** Store $|\mathbf{w}|$ parameters instead of $|\mathcal{X}|$ values

TD(0) with Linear Approximation

Substituting $\phi_t = \phi(x_t)$ into the gradient descent update:

$$w_{t+1} = w_t + \alpha_t \underbrace{\left[r_t + \gamma \phi_{t+1}^T w_t - \phi_t^T w_t\right]}_{\text{TD error}} \phi_t$$

Algorithm:

1. Observe transition (x_t, r_t, x_{t+1})
2. Compute TD error: $\delta_t = r_t + \gamma \phi_{t+1}^T w_t - \phi_t^T w_t$
3. Update weights: $w_t \leftarrow w_t + \alpha_t \delta_t \phi_t$

Critical Issue

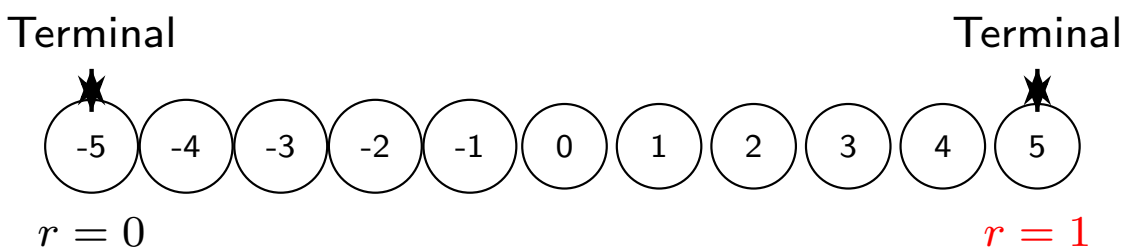
Highly sensitive to step size α_t

11-State Random Walk Example

A simple illustrative environment

Setup:

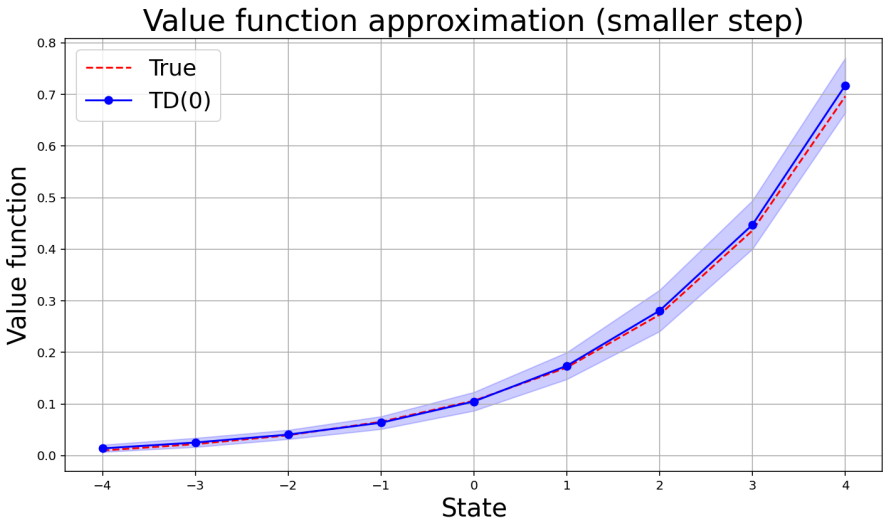
- States: $\{-5, -4, \ldots, 0, \ldots, 4, 5\}$
- Actions: Move left or right with equal probability (0.5 each)
- Rewards:
  - $r = 1$ when reaching rightmost state (5)
  - $r = 0$ everywhere else
- Episodes terminate at boundaries (± 5)
- Discount factor: $\gamma = 0.99$



Random Walk: Impact of Step Size

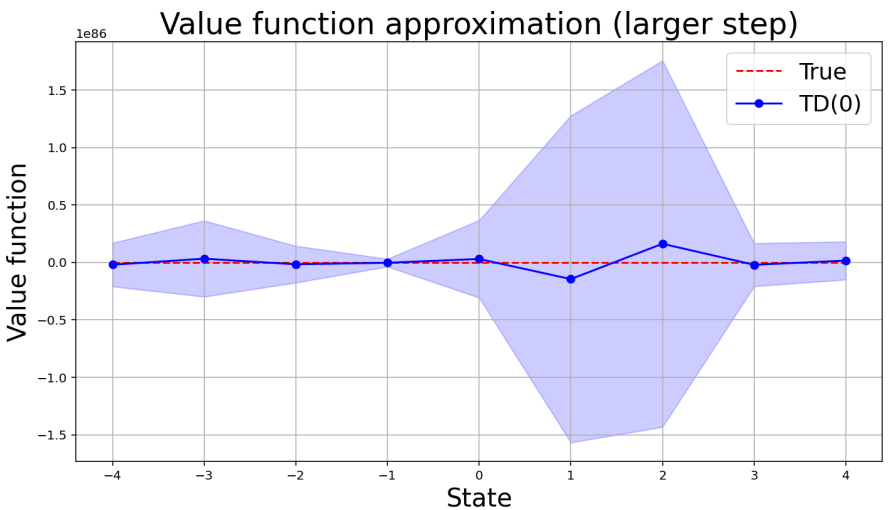
Experiment: TD(0) with two different constant step sizes

Small step size: $\alpha = 0.05$



- ✓ Stable convergence
- ✓ Matches true values
- × Very slow learning

Large step size: $\alpha = 1.5$



- ✓ Fast initial exploration
- × Severe divergence
- × Unstable estimates

Challenge: Finding the right α is critical but difficult in practice

Inspiration: Implicit SGD

Standard SGD:

$$w_{t+1} = w_t + \alpha_t \nabla f(w_t)$$

Implicit SGD [Toulis and Airoldi, 2017]**:**

$$w_{t+1}^{im} = w_t^{im} + \alpha_t \nabla f(w_{t+1}^{im})$$

Key Idea

By using the next iterate w_{t+1} in the gradient evaluation, we create a fixed-point equation that introduces natural stabilization

Benefits:

- Automatic step size adaptation
- Improved stability with minimal overhead

Implicit TD(0) Update

Standard TD(0):

$$w_{t+1} = w_t + \alpha_t(r_t + \gamma\phi_{t+1}^T w_t - \phi_t^T w_t)\phi_t$$

Implicit TD(0):

$$w_{t+1}^{im} = w_t^{im} + \alpha_t(r_t + \gamma\phi_{t+1}^T w_t^{im} - \phi_t^T w_{t+1}^{im})\phi_t$$

Key difference: Use w_{t+1}^{im} instead of w_t^{im} in the current state's value

Challenge: This creates an implicit equation — how to solve it efficiently?

Closed-Form Update

Rearrange the implicit equation:

$$(I + \alpha_t \phi_t \phi_t^T) w_{t+1}^{im} = w_t^{im} + \alpha_t (r_t + \gamma \phi_{t+1}^T w_t^{im}) \phi_t$$

Implicit TD(0) — Closed Form

$$w_{t+1}^{im} = w_t^{im} + \frac{\alpha_t}{1 + \alpha_t \|\phi_t\|^2} \delta_t^{im} \phi_t$$

where $\delta_t^{im} = r_t + \gamma \phi_{t+1}^T w_t^{im} - \phi_t^T w_t^{im}$

Cost: Same as standard TD(0) — only $O(d)$ operations per step

Why Implicit TD Stabilizes Learning

Effective step size:

$$\tilde{\alpha}_t = \frac{\alpha_t}{1 + \alpha_t \|\phi_t\|^2}$$

Key properties:

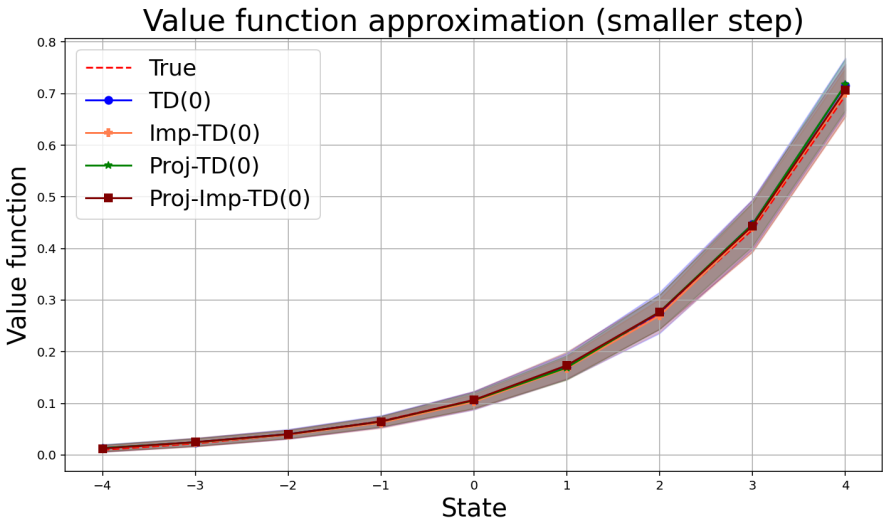
1. $\tilde{\alpha}_t \leq \alpha_t$ always holds
2. When $\|\phi_t\|$ is large:
   - Standard TD: takes full step α_t
   - Implicit TD: adaptively shrinks α_t
3. When $\|\phi_t\|$ is small:
   - Both methods behave similarly

Result: Implicit TD automatically adjusts updates when feature norm is large, preventing instability

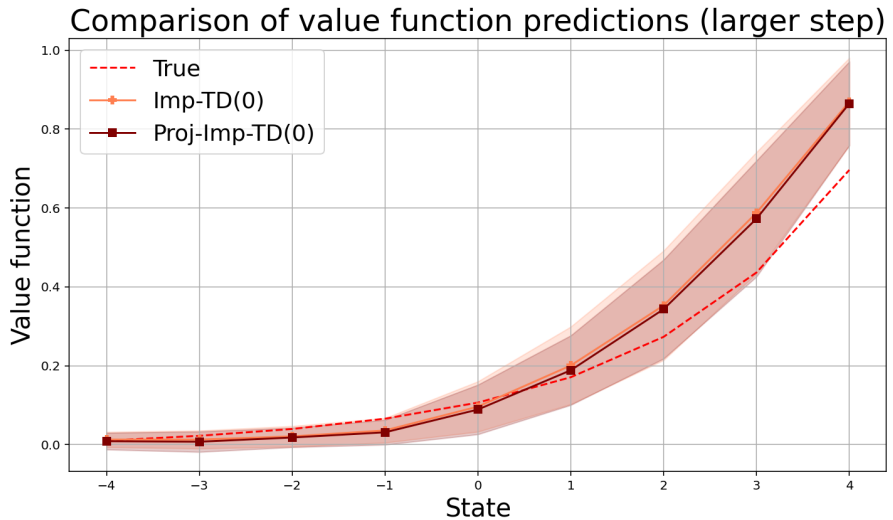
Random Walk: Implicit TD Performance

Experiment: Implicit TD(0) with two different constant step sizes

Small step size: $\alpha = 0.05$



Large step size: $\alpha = 1.5$



Result: Implicit TD remains stable with large step size

Finite-Time Analysis of Implicit TD

Theorem (Kim, Toulis, and Laber, 2025)

Under suitable assumptions with constant step size α, if

$$2\alpha(1-\gamma)\lambda_{\min} < 1+\alpha$$

then projected implicit TD(0) satisfies

$$\mathbb{E}^{\mu}\|w_n^{im} - w_\star\|^2 = e^{-cn}\|w_1 - w_\star\|^2 + O(\tau_\alpha \alpha)$$

for some constant $c > 0$ with a mixing time $\tau_\alpha := \min\{i \in \mathbb{N} : m\rho^i \leq \alpha\}$

Notations

- μ: stationary distribution of ergodic Markov Chain $\{x_n\}_{n\in\mathbb{N}}$
- $\lambda_{\min}$: smallest eigenvalue of $\Sigma = \mathbb{E}^{\mu}[\phi_t \phi_t^T]$
- $w_\star$: solution to $\mathbb{E}^{\mu}[\phi_t(\phi_t - \gamma\phi_{t+1})^T]w_\star = \mathbb{E}^{\mu}[\phi_t r_t]$

Stability Region Comparison

Convergence conditions [Bhandari, Russo, and Singal, 2018]:

Standard TD(0):

$$\boxed{2\alpha(1-\gamma)\lambda_{\min} < 1}$$

Implicit TD(0):

$$\boxed{2\alpha(1-\gamma)\lambda_{\min} < 1+\alpha}$$

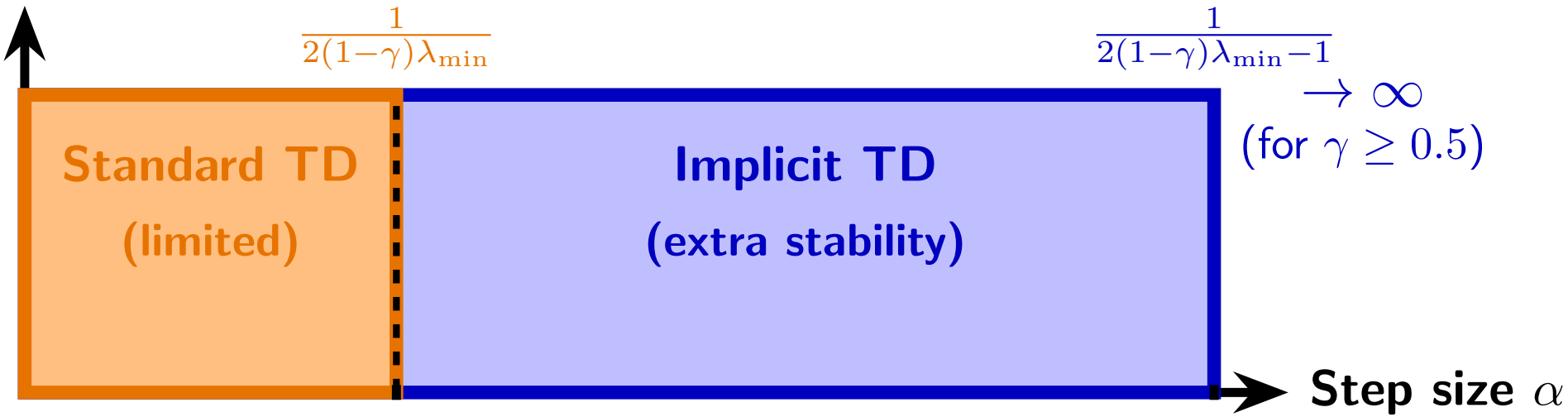
Key Insight

For $\gamma \geq 0.5$: we have $2(1-\gamma)\lambda_{\min} - 1 < 0$, meaning

$$\alpha_{\max}^{im} = +\infty$$

Implicit TD(0) is stable for ANY positive step size $\alpha > 0$!

Visualizing Stability Regions



Mountain Car Task

Environment:

- Underpowered car in valley between two hills
- State: position and velocity
- Actions: accelerate left, right, or neutral

Goal: Reach the flag at the top of the right hill

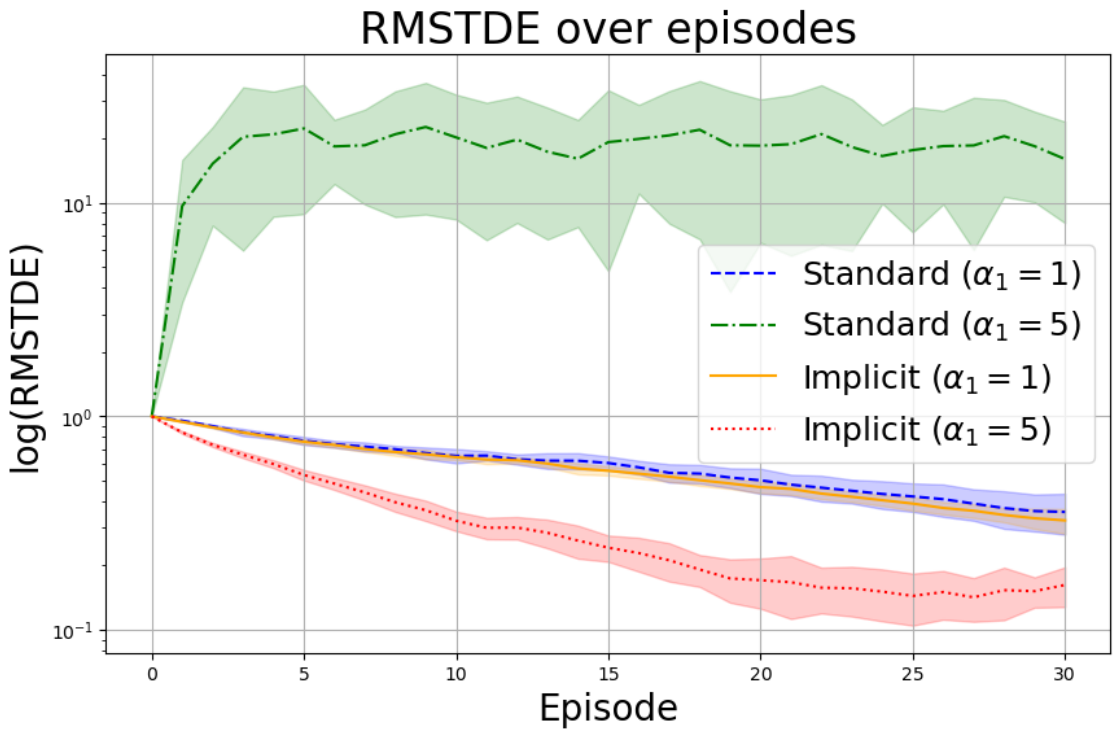
Challenge: Direct approach fails — must build momentum by oscillating

Setup:

- 100 RBF features
- Decreasing step size: $\alpha_t = \alpha_1/t$
- Test: $\alpha_1 \in \{1, 5\}$

Mountain Car: Learning Curves

Experiment: Compare Standard vs Implicit TD(0)



$\alpha_1 = 1$ **(moderate)**

- Both methods converge

$\alpha_1 = 5$ **(aggressive)**

- Standard: diverges
- Implicit: faster convergence

Extensions

Our implicit approach extends to:

- **Implicit TD(λ):** Multi-step temporal difference learning with eligibility traces [Kim, Toulis, and Laber, 2025]
- **Implicit TDC:** Off-policy evaluation with gradient correction [Kim, Toulis, and Laber, 2025]
- **Average-reward implicit TD(λ):** Continuing tasks without discounting [Kim, Cho, and Laber, 2025]
- **Implicit Q-learning/SARSA:** Policy control and optimal decision-making [Kim and Laber, 2026]

Key benefit: All variants inherit improved stability and relaxed step size requirements

Thank You!

Questions?

hwanwoo.kim@duke.edu

arXiv:2505.01361v2

References I

Bhandari, J., Russo, D., & Singal, R. (2018). A finite time analysis of temporal difference learning with linear function approximation. *Conference on learning theory*, 1691–1692.

Kim, H., Cho, D. D., & Laber, E. (2025). Implicit Updates for Average-Reward Temporal Difference Learning. *arXiv preprint arXiv:2510.06149*.

Kim, H., & Laber, E. (2026). Implicit Q-Learning and SARSA: Liberating Policy Control from Step-Size Calibration. *arXiv preprint arXiv:2601.18907*.

Kim, H., Toulis, P., & Laber, E. (2025). Stabilizing Temporal Difference Learning via Implicit Stochastic Recursion. *arXiv preprint arXiv:2505.01361*.

References II

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 400–407.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Toulis, P., & Airoldi, E. M. (2017). Asymptotic and finite-sample properties of estimators based on stochastic gradients. *The Annals of Statistics*, *45*(4), 1694–1727.