Understanding the Convergence of Reinforcement Learning Algorithms from Dynamical Systems Perspectives

Niao He

Assistant Professor

University of Illinois at Urbana-Champaign

RL Theory Virtual Seminar

June 16, 2020

Acknowledgement

- A Unified Switching System Perspective and O.D.E. Analysis of Q-Learning Algorithms. arXiv:1912.02270.
  with Donghwan Lee.

Donghwan Lee
(former Postdoc, now Assistant Professor at KAIST)

Wentao Weng
(Undergrad student, Tsinghua)

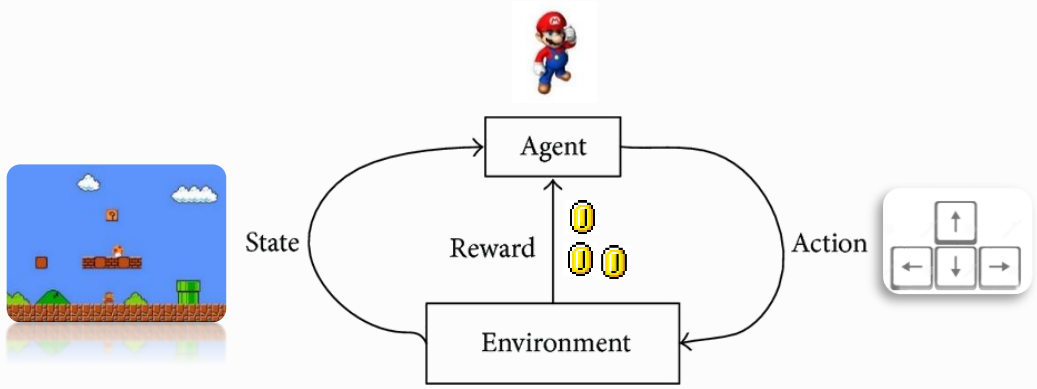
Harsh Gupta
(PhD student, UIUC)

R. Srikant
(Professor, UIUC)

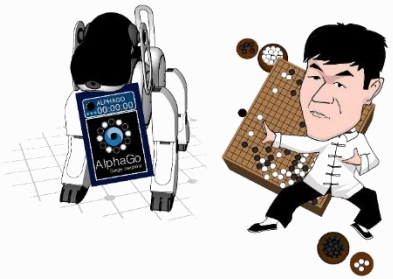
- The Mean-squared Error of Double Q-learning, working paper, 2020.
  with Wentao Weng, Harsh Gupta, Ying Lei, and R. Srikant.

Reinforcement Learning

- An agent selects actions to maximize long-term reward

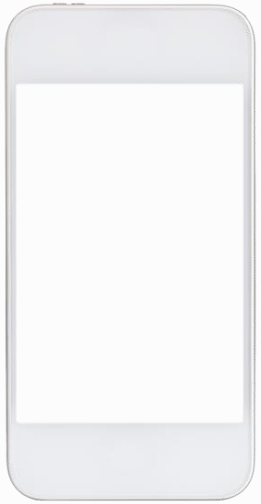


- A wide spectrum of applications



Games

Self-driving vehicle

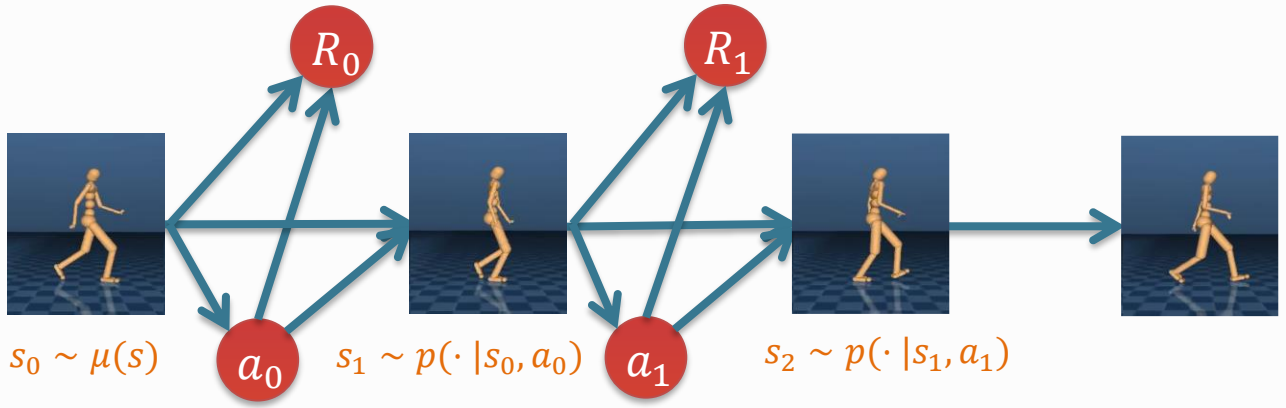


Chatbot

Reinforcement Learning

- **Markov decision processes :** describe the environment for RL:
  - State (S): a (finite) set of states
  - Action ($\mathcal{A}$): a (finite) set of actions
  - Probability transition matrix ($P_{ss'}^{a}$): $P(s_{t+1} = s' | s_t = s, a_t = a)$, unknown
  - Reward function (R): $R(s, a) = E[R_{t+1} | s_t = s, a_t = a]$
  - Discount factor (γ): $\gamma \in [0,1]$

- **Policy (π):** the agent's behavior strategy, $a \sim \pi(\cdot | s)$



Reinforcement Learning

- Two fundamental tasks in RL:

Policy Evaluation

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}R(s_t, a_t)|s_0 = s\right]$$

Bellman Equation

$$V^{\pi}(s) = \mathbb{E}_{s'|s,a\sim\pi(s)}\left[R(s,a) + \gamma V^{\pi}(s')\right]$$

Policy Optimization

$$\max_{\pi}\mathbb{E}_{s}[V^{\pi}(s)] = \mathbb{E}_{s}\left[\mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}R(s_t, a_t)|s_0 = s\right]\right]$$

Bellman Optimality Equation

$$Q^{*}(s,a) = \mathbb{E}_{s'|s,a}\left[R(s,a) + \gamma\max_{a'\in\mathcal{A}}Q^{*}(s',a')\right]$$

$$\pi^{*}(s) = \operatorname*{argmax}_{a\in\mathcal{A}}Q(s,a)$$

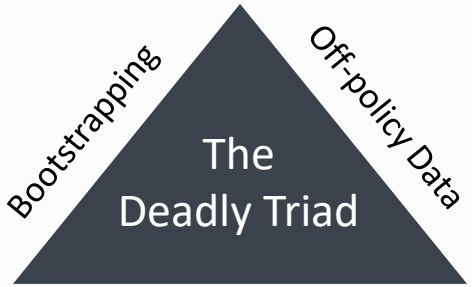
Reinforcement Learning

Challenges of solving Bellman equations:

- Unknown MDP: $P(s'|s,a), R(s,a)$ are unknown
  - We only observe samples $\{(s_k, a_k, s_{k+1})\}_{k=1}^{n}$ from some behavior policy
  - Resort to bootstrapping or stochastic approximation schemes
  - Example: standard Q-learning

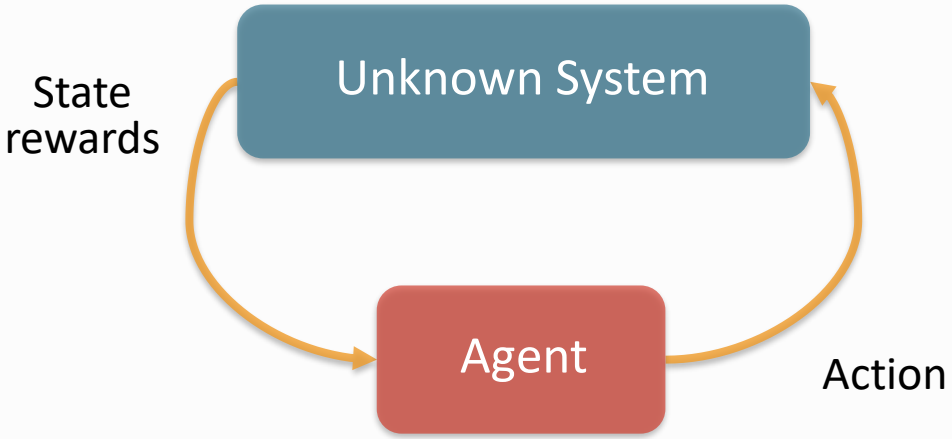
$$Q(s_k, a_k) = Q(s_k, a_k) + \alpha_k(R(s_k, a_k) + \gamma \cdot \max_{a' \in \mathcal{A}} Q(s_{k+1}, a') - Q(s_k, a_k))$$

- Large State and Action Spaces:
  - $S, \mathcal{A}$ can be extremely large, even infinite
  - Resort to function approximation techniques
  - Classical algorithms can diverge when function approximation is used.



The Interplay of Dynamical Systems and Reinforcement Learning

- RL is all about learning unknown dynamical systems.



- Many RL algorithms can be viewed as discrete dynamical systems.

Stochastic dynamic programming

$$Q_{k+1} = \hat{F} Q_k$$

$$k \leftarrow k + 1$$

Recipe for Analyzing Classical RL Algorithms

Stochastic Approximation

$$x_{k+1} = x_k + \alpha_k(f(x_k) + \epsilon_{k+1})$$

Dynamical Systems

$$\frac{d}{dt}x_t = f(x_t)$$

Asymptotic Convergence

- **TD-learning with LFA**
  [Tsitsiklis & Van Roy, 1997]
- **Double TD-learning with LFA**
  [Lee & He, 2019]
- **Synchronous Q-learning**
  [Borkar & Meyn, 2000]
- **Asynchronous Q-learning**
  [Jaakkola et al.,1994][Tsitsiklis, 1994] [Lee & He, 2020]
- **Q-learning with LFA**
  [Melo, Meyn, & Ribeiro, 2008] [Lee & He, 2020]
- **Greedy-GQ algorithm**
  [Maei et al., 2010]

Finite-time Convergence

- **TD-learning with LFA**
  [Srikant & Ying, 2019]
  [Dalal et al., 2018] [Bhandari et al., 2019]
  [Lakshminarayanan & Szepesvári, 2018]
- **Synchronous Q-learning**
  [Wainwright, 2019]
- **Asynchronous Q-learning**
  [Szepesvári, 1998][Even-Dar & Mansour, 2003]
  [Qu & Wierman, 2020] [Li et al., 2020]
- **Q-learning with LFA**
  [Chen et al., 2019] [Wang & Giannakis,2020]

Tight Error Bound

- **TD-learning with LFA**
  [Hu & Syed, 2019]
  [Devraj & Meyn, 2017]
  [Chen et al., 2020]
- **Q-learning & Relative Q-learning**
  [Devraj & Meyn, 2020]
- **Double Q-learning**
  [Weng et al., 2020]

Recipe for Analyzing Modern Optimization-based RL Algorithms

(Non-)Convex Optimization

$$\min_x L(x) \quad or \quad \min_x \max_y L(x, y)$$

First-Order Methods

$e.g.$, GD, SGD, MD, SVRG, Primal-Dual, etc.

Policy Evaluation

- Residual gradient algorithm
- GTD, GTD-2, and its cousins
- SVRG/SAGA
- ……

Policy Optimization

- Policy Gradient
- TRPO
- PPO
- SPD-RL
- SBEED
- ……

Asymptotic Convergence

Finite-time Convergence

Lower Bound

IEEE Magazine Article : *Optimization for Reinforcement Learning: from a single agent to cooperative agents*. 2020. (with Lee, Kamal, and Cevher)

Dynamical Systems Perspectives for RL

- **Opportunities:**
  - Unification
  - No need for objective/gradients/regularizations
  - Characterization of exact behavior
  - Theoretical insights and practical guidelines

- **Challenges:**
  - Possibly nonlinear systems (e.g., even tabular Q-learning)
  - Hard to cope with nonlinear function approximation

Question: How to systematically analyze the nonlinear dynamics of the large family of Q-learning algorithms?

A Quick Tour of the O.D.E. Analysis

The ODE Method

SIAM J. CONTROL OPTIM.
Vol. 38, No. 2, pp. 447–469

THE O.D.E. METHOD FOR CONVERGENCE OF STOCHASTIC APPROXIMATION AND REINFORCEMENT LEARNING*

V. S. BORKAR† AND S. P. MEYN‡

Abstract. It is shown here that stability of the stochastic approximation algorithm is implied by the asymptotic stability of the origin for an associated ODE. This in turn implies convergence of the algorithm. Several specific classes of algorithms are considered as applications. It is found that the results provide (i) a simpler derivation of known results for reinforcement learning algorithms; (ii) a proof for the first time that a class of asynchronous stochastic approximation algorithms are convergent without using any a priori assumption of stability; (iii) a proof for the first time that asynchronous adaptive critic and Q-learning algorithms are convergent for the average cost optimal control problem.

Key words. stochastic approximation, ODE method, stability, asynchronous algorithms, reinforcement learning

AMS subject classifications. 62L20, 93E25, 93E15

PII. S0363012997331639

1. Introduction. The stochastic approximation algorithm considered in this paper is described by the d-dimensional recursion

$$X(n+1) = X(n) + a(n)\big[h\big(X(n)\big) + M(n+1)\big], \quad n \ge 0, \tag{1.1}$$

where $X(n) = [X_1(n), \dots, X_d(n)]^T \in \mathbb{R}^d$, $h : \mathbb{R}^d \to \mathbb{R}^d$, and $\{a(n)\}$ is a sequence of positive numbers. The sequence $\{M(n) : n \ge 0\}$ is uncorrelated with zero mean.

Though more than four decades old, the stochastic approximation algorithm is now of renewed interest due to novel applications to reinforcement learning [20] and as a model of learning by boundedly rational economic agents [19]. Traditional convergence analysis usually shows that the recursion (1.1) will have the desired asymptotic behavior provided that the iterates remain bounded with probability one, or that they visit a prescribed bounded set infinitely often with probability one [3, 14]. Under such stability or recurrence conditions one can then approximate the sequence $\boldsymbol{X} = \{X(n) : n \ge 0\}$ with the solution to the ordinary differential equation (ODE)

$$\dot{x}(t) = h\big(x(t)\big) \tag{1.2}$$

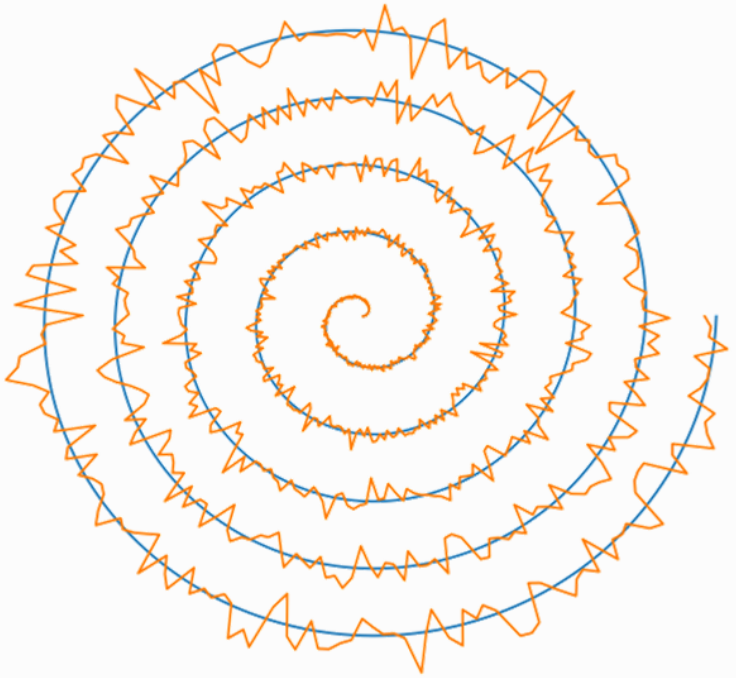
with identical initial conditions $x(0) = X(0)$.

The ODE Method: Key Idea

$$X_{k+1} = X_k + \alpha_k(f(X_k) + \epsilon_{k+1})$$

Dynamical system: $\frac{d}{dt}x_t = f(x_t)$

Asymptotic stability



Dynamical system, $\frac{d}{dt}x_t = f(x_t)$, is *globally asymptotically stable* if $x_t \to x^*$ for any x_0.

The ODE Method: Borkar and Meyn Theorem

SA: $X_{k+1} = X_k + \alpha_k(f(X_k) + \epsilon_{k+1})$

[Borkar and Meyn Theorem, 2000]

Under the following conditions:

a) Global Lipschitz continuity of the mapping f

b) Robbins-Monro stepsize: $\sum \alpha_k = \infty, \sum \alpha_k^2 < \infty$

c) Bounded noise of martingale difference: $E[\|\epsilon_{k+1}\|^2 | G_k] \leq C_0(1 + \|X_k\|^2), \forall k \geq 0$

d) *Asymptotic stability of the limiting ODE* : $\dot{x}_t = f_\infty(x_t) \coloneqq \lim_{c \to \infty} \frac{f(cx)}{c}$

e) *Global asymptotic stability of the original ODE*: $\dot{x}_t = f(x_t)$

we have $X_k \to x^*$ as $k \to \infty$.

Stability of Linear Systems and Applications in TD-learning

- **Linear System:**

$$\frac{d}{dt}x_t = Ax_t$$

The origin is an asymptotically stable equilibrium point if and only if A is Hurwitz.

- TD-learning with Linear Function Approximation

$$\theta_{k+1} = \theta_k + \alpha_k \phi(s_k)[r(s_k, a_k) + \gamma \phi(s_{k+1})^T \theta_k - \phi(s_k)^T \theta_k]$$

$$\frac{d}{dt}(\theta_t - \theta^*) = A(\theta_t - \theta^*), \qquad A = \Phi^T D(\gamma P^\pi - I)\Phi \text{ is } Hurwitz$$

- Applications to other TD-learning variants:
  - TD(0), TD(λ) [Tsitsiklis & Van Roy, 1997]
  - GTD, TDC [Sutton et al., 2009]
  - A-TD, D-TD [Lee and He, 2019]

* A matrix is Hurwitz if all eigenvalues have strictly negative real parts.

Stability of Nonlinear Systems and Applications in Q-learning

- **Nonlinear System:**

$$\frac{d}{dt}x_t = f(x_t)$$

[Khalil, 2002]

The origin is unique, globally asymptotically stable if there exists a twice differentiable Lyapunov function $V(x)$ such that

$$k_1\|x\|^\alpha \le V(x) \le k_2\|x\|^\alpha$$
$$\frac{dV}{dx}f(x) \le -k_3\|x\|^\alpha$$

for some positive constants α, k_1, k_2, k_3.

- **Applications:**
  - Tabular Q-learning [Borkar & Meyn, 2000]
  - Q-learning with linear function approximation [Melo et al., 2008] [Wang & Giannakis, 2020]

Stability of Linear Switching Systems

- **Linear switching system**:

$$\frac{d}{dt}x_t = A_{\sigma_t}x_t$$

  - coupling between continuous dynamics and discrete events (switching)
  - σ_t: switching signal $\in \{1,2,\cdots,M\}$; $\{A_1,\dots,A_M\}$ subsystem matrices
  - $\sigma_t = \sigma(x_t)$: state-feedback switching signal

[Lin and Antsaklis, 2009]

The origin is the unique globally asymptotically stable equilibrium point if and only if there exists a full column rank matrix L and a family of NRD matrices $\{\bar{A}_1,\dots,\bar{A}_M\}$ such that

$$LA_\sigma = \bar{A}_\sigma L, \forall \sigma \in \{1,2,\dots,M\}.$$

Negative Row Dominant Diagonal (NRD) matrix A: $a_{ii} + \sum_{j\neq i}|a_{ij}| < 0, \forall i$

Switching System Perspective of Q-learning Algorithms

with Donghwan Lee (2020)

(Asynchronous) Q-learning

- **Q-learning** (Watkins, 1992)

$$Q_{k+1}(s_k, a_k) = Q_k(s_k, a_k) + \alpha_k(r(s_k, a_k) + \gamma \max_{a'} Q_k(s_{k+1}, a') - Q_k(s_k, a_k))$$

  - Use a single trajectory of samples $\{(s_k, a_k, s_{k+1})\}$ from behavior policy
  - If every state-action pair is visited infinitely often, $Q_k \to Q^*$ with probability one

- **Convergence Analysis**
  - The original proof [Watkins and Dayan, 1992]
  - Stochastic-approximation-based approach [Jaakkola et al., 1994] [Tsitsiklis, 1994]
  - Finite-time analysis: [Szepesvári, 1998][Even-Dar & Mansour, 2003]
  - Recent work: [Qu & Wierman, 2020] [Li et al., 2020]

Switching System Model of Q-Learning

- **Q-learning**

$$Q_{k+1}(s_k, a_k) = Q_k(s_k, a_k) + \alpha_k(r(s_k, a_k) + \gamma \max_{a'} Q_k(s_{k+1}, a') - Q_k(s_k, a_k))$$

- **Dynamical system**

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q_t}} - D\right)(Q_t - Q^*) + \gamma DP(\Pi_{\pi_{Q_t}} - \Pi_{\pi^*})Q^*$$

  - Greedy policy: $\pi_{Q_t}(s) = \text{argmax}_a Q_t(s, a)$
  - Diagonal elements of D : state-action distribution
  - $P = \begin{bmatrix} \vdots \\ P_a \\ \vdots \end{bmatrix}$, P_a=transition probability matrix for taking action a
  - $\Pi_\pi \coloneqq [\cdots \quad \Gamma_a \quad \cdots], [\Gamma_a]_{(s,a')} = 1 \; if \;\; \pi(s) = a'$ and 0 otherwise

Switching System Model of Q-Learning

- **Q-learning**

$$Q_{k+1}(s_k, a_k) = Q_k(s_k, a_k) + \alpha_k(r(s_k, a_k) + \gamma \max_{a'} Q_k(s_{k+1}, a') - Q_k(s_k, a_k))$$

- **Dynamical system**

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q_t}} - D\right)(Q_t - Q^*) + \gamma DP(\Pi_{\pi_{Q_t}} - \Pi_{\pi^*})Q^*$$

Affine switching system

$$\frac{d}{dt}x_t = A_{\sigma(x_t)}x_t + b_{\sigma(x_t)}$$

- $x_t = Q_t - Q^*, \sigma(x_t) = \psi(\pi_{Q_t}), \pi_{Q_t}(s) = \text{argmax}_a Q_t(s, a)$
- ψ: deterministic policy → integer

Stability Analysis: Upper and Lower Comparison Systems

- **Upper comparison system (linear switching system)**

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q_t}} - D\right)(Q_t - Q^*)$$

$$\geq$$

- **Original affine switching system**

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q_t}} - D\right)(Q_t - Q^*) + \gamma DP(\Pi_{\pi_{Q_t}} - \Pi_{\pi^*})Q^*$$

$$\geq$$

- **Lower comparison system (linear system)**

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q^*}} - D\right)(Q_t - Q^*)$$

Stability Analysis: Vector Comparison Principle

$$\underline{\frac{d}{dt}\underline{x}_t = \underline{f}(\underline{x}_t)} \quad \leq \quad \frac{d}{dt}x_t = f(x_t) \quad \leq \quad \underline{\frac{d}{dt}\overline{x}_t = \overline{f}(\overline{x}_t)}$$

Linear system
Easy analysis

$\downarrow t \to \infty$

$\underline{x}_t \to 0$

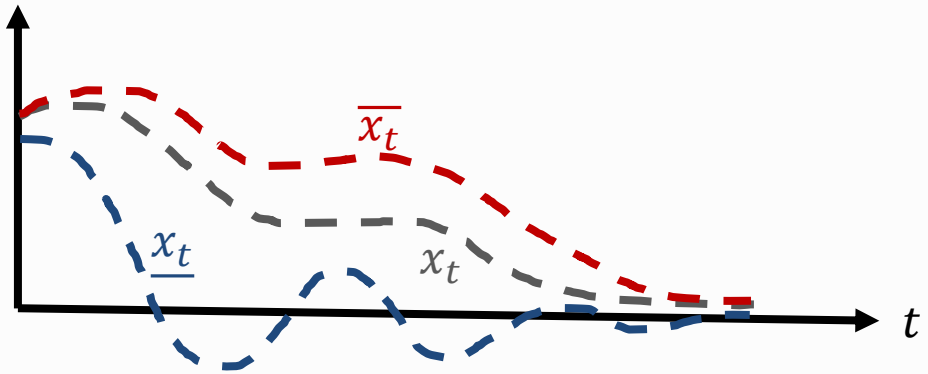
Linear switching system
Easy analysis

$\downarrow t \to \infty$

$\overline{x}_t \to 0$

$x_t \to 0$

A_σ *is NRD*



[Vector Comparison Principle]

If $\bar{f}$ and f are globally Lipschitz continuous, $\bar{f}$ is quasi-monotone increasing, then

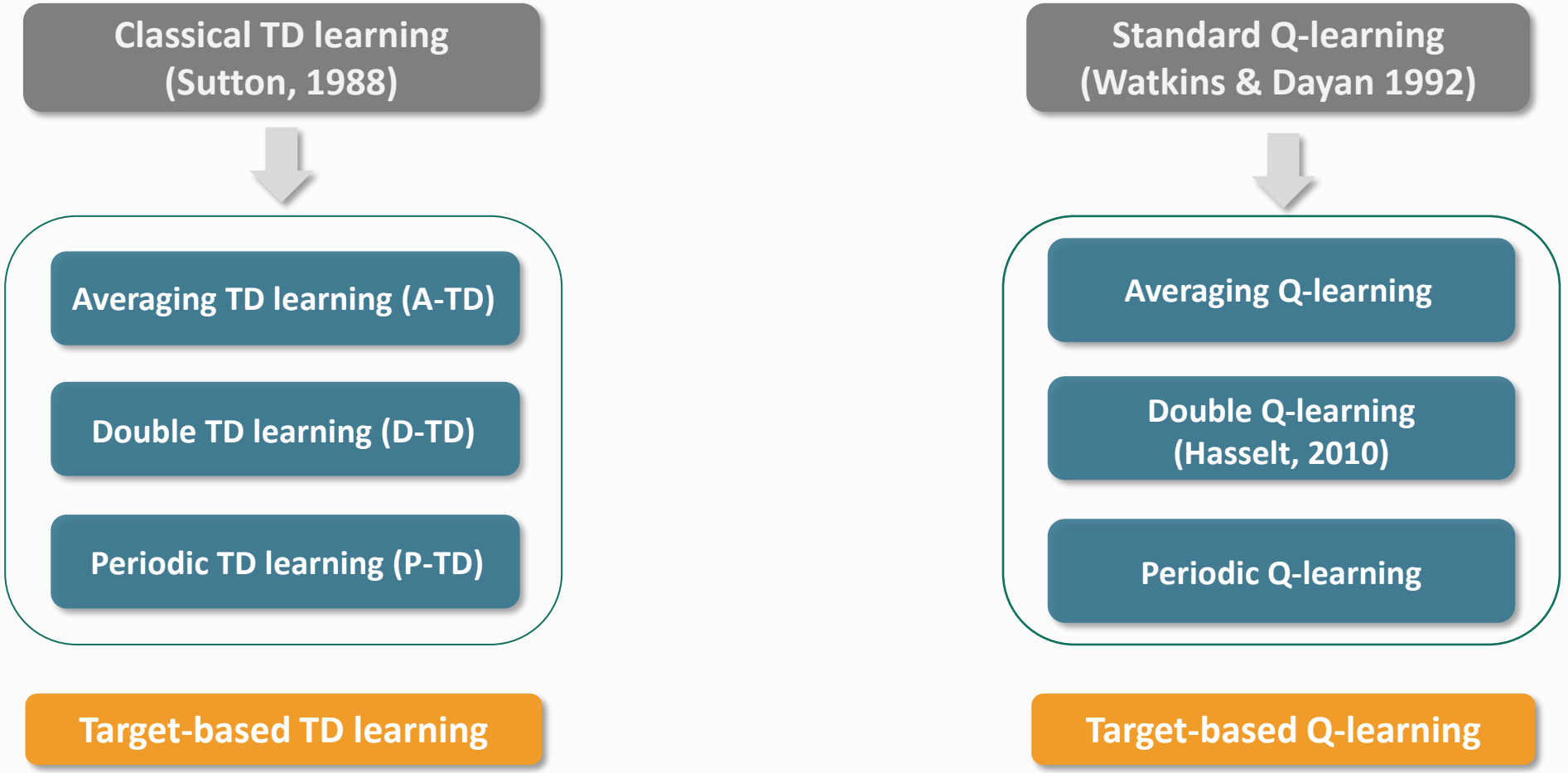
$$f \leq \bar{f}, x_0 \leq \bar{x}_0 \Rightarrow x_t \leq \bar{x}_t, \forall t \geq 0.$$

Stability and Convergence

- Immediate result:
  - $\underline{Q}_t - Q^* \leq Q_t - Q^* \leq \overline{Q}_t - Q^*, \ \forall t \geq 0$
  - The origin is the unique globally asymptotically stable equilibrium point of the three systems.
  - Under the Robbins-Monro stepsize, $Q_k \to Q^*$, as $k \to \infty$.

- Extensions:
  - Target-based Q-learning algorithms
  - Q-learning with linear function approximation

Target-based Q-learning



Lee and **H.**, "Target-based temporal difference learning," ICML, 2019.
Lee and **H.**, "Periodic Q-learning," L4DC, 2020.

Averaging Q-learning

- **Algorithm:**

$$Q_{k+1}^{A}(s_k, a_k) = Q_k^{A}(s_k, a_k) + \alpha_k(r(s_k, a_k) + \gamma \max_{a'} Q_k^{B}(s_{k+1}, a') - Q_k^{A}(s_k, a_k))$$

$$Q_{k+1}^{B}(s_k, a_k) = Q_k^{B}(s_k, a_k) + \alpha_k \delta(Q_k^{A}(s_k, a_k) - Q_k^{B}(s_k, a_k))$$

  - This can also be formulated as a switching system.
  - Similarly, we can easily show that for any $\delta > 0$, $Q_k^A \to Q^*$ and $Q_k^B \to Q^*$, as $k \to \infty$.

Q-learning with Linear Function Approximation

- **Algorithm:**

$$\theta_{k+1} = \theta_k + \alpha_k \phi(s_k, a_k)[r(s_k, a_k) + \gamma \max_{a'}(\Phi\theta_k)(s_{k+1}, a') - (\Phi\theta_k)(s_k, a_k)]$$

- **Switching system:**

$$\frac{d}{dt}\theta_t = \left(\gamma \Phi^T D P \Pi_{\pi(\theta_t)} \Phi - \Phi^T D \Phi\right)\theta_t + \Phi^T D R$$

- **New sufficient condition:**

$$-\phi_i^T D \phi_i + \gamma \phi_i^T D P \Pi_\pi \sum \phi_j < 0, \forall \text{ admissible } \pi$$

  - Under the above condition, we can easily show that for $\theta_k \to \theta^*$ as $k \to \infty$.
  - Less conservative than the Melo's condition.

The Roadmap

Q-Learning

$$Q_{k+1}(s,a) = Q_k(s,a) - \alpha_k(r + \gamma \max_{a'} Q_k(s',a') - Q_k(s,a))$$

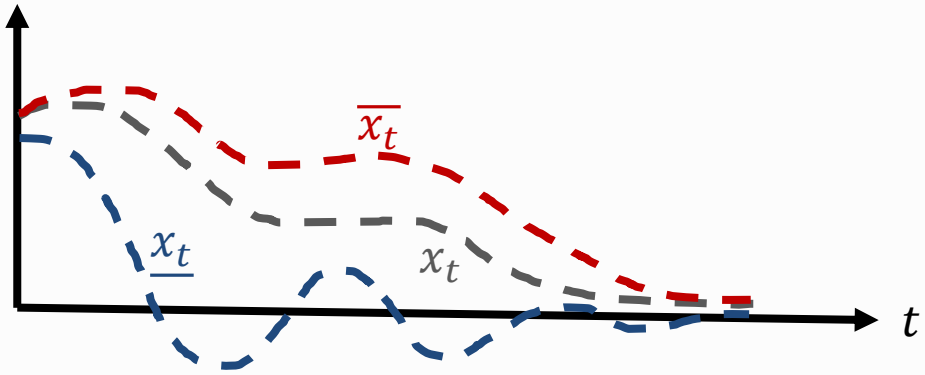
Nonlinear ODE

$$\frac{d}{dt}(Q_t - Q^*) = \left(\gamma DP\Pi_{\pi_{Q_t}} - D\right)(Q_t - Q^*) + \gamma DP(\Pi_{\pi_{Q_t}} - \Pi_{\pi^*})Q^*$$

Affine switching system

$$\frac{d}{dt}x_t = A_{\sigma(x_t)}x_t + b_{\sigma(x_t)}$$

Asymptotic stability of upper and lower comparison systems



Comparison principle

Convergence

Highlights

- **First connection** between reinforcement learning and switching systems
- **Simple and intuitive analysis** of asynchronous Q-learning based on existing control theory
- **Unified framework** for analyzing the convergence of a family of Q-learning algorithms
- **Tight conditions** and weak assumptions

- Future Work
  - Continuous-time vs. discrete-time dynamics of switching systems
  - Finite-time convergence rate and tight error bounds
  - Other Q-learning variants: deep Q-learning
  - Efficient and robust RL algorithms from the control perspective

Is Double Q-learning Provably More Efficient than Q-learning?

with Weng, Gupta, and Srikant (2020)

Double Q-learning

- **Q-learning with LFA:**

$$\theta_{k+1} = \theta_k + \alpha_k \phi(s_k, a_k)[r(s_k, a_k) + \gamma H(\theta_k, \theta_k, s_{k+1}) - \phi(s_k, a_k)^T \theta_k]$$

- **Double Q-learning with LFA:**

$$\theta^A_{k+1} = \theta^A_k + \beta_k \delta_k \phi(s_k, a_k)[r(s_k, a_k) + \gamma H(\theta^A_k, \theta^B_k, s_{k+1}) - \phi(s_k, a_k)^T \theta^A_k$$

$$\theta^B_{k+1} = \theta^B_k + (1 - \beta_k)\delta_k \phi(s_k, a_k)[r(s_k, a_k) + \gamma H(\theta^B_k, \theta^A_k, s_{k+1}) - \phi(s_k, a_k)^T \theta^B_k$$

$$H(\theta_1, \theta_2, s) = \phi\left(s, \arg\max_a \phi(s, a)^T \theta_1\right)^T \theta_2, \ \beta_k \sim \text{Bernoulli}\left(\frac{1}{2}\right) \text{ i.i.d.}$$

The Mean-Squared Error

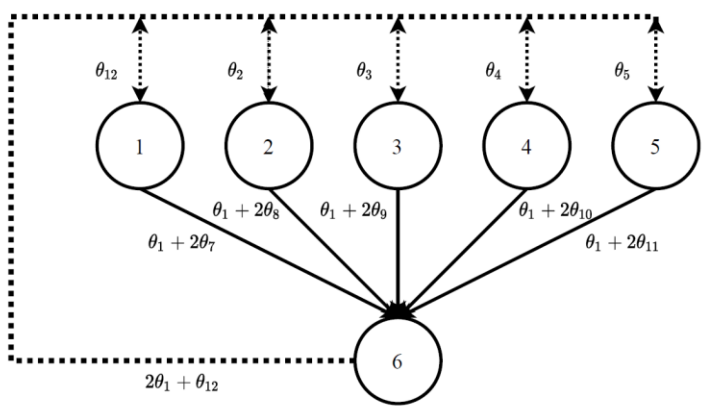
- Asymptotic mean-squared errors
  - Q-learning: $AMSE(\theta) \coloneqq \lim_{k\to\infty} kE\|\theta_k - \theta^*\|^2$
  - Double Q-learning: $AMSE(\theta^A) \coloneqq \lim_{k\to\infty} kE\left\|\theta_k^A - \theta^*\right\|^2$
  - Double Q-learning with average estimator: $AMSE\left(\frac{\theta^A+\theta^B}{2}\right) \coloneqq \lim_{k\to\infty} kE\left\|\frac{\theta_k^A+\theta_k^B}{2} - \theta^*\right\|^2$

Theorem (informal): Set $\alpha_k = \frac{g}{k}, \delta_k = \frac{2g}{k}$ and assume both Q-learning and Double Q-learning converge. Under mild conditions, we have

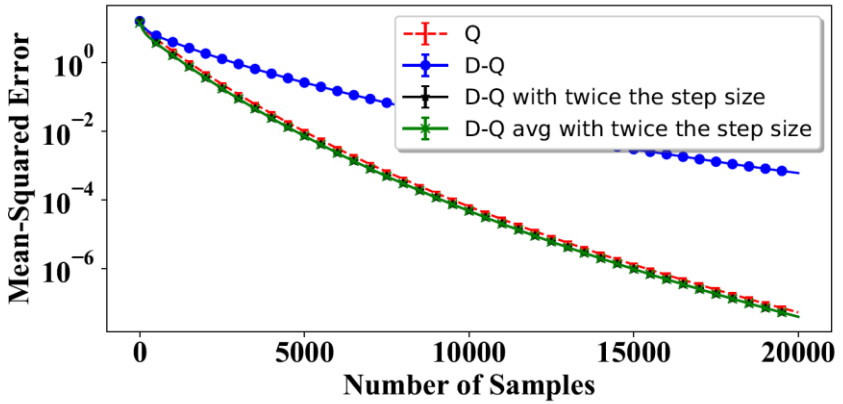
$$AMSE(\theta^A) = AMSE(\theta^B) \geq AMSE(\theta) + c_0 g \qquad (c_0 > 0, g > 0)$$

$$AMSE\left(\frac{\theta^A + \theta^B}{2}\right) = \mathrm{AMSE}(\theta)$$

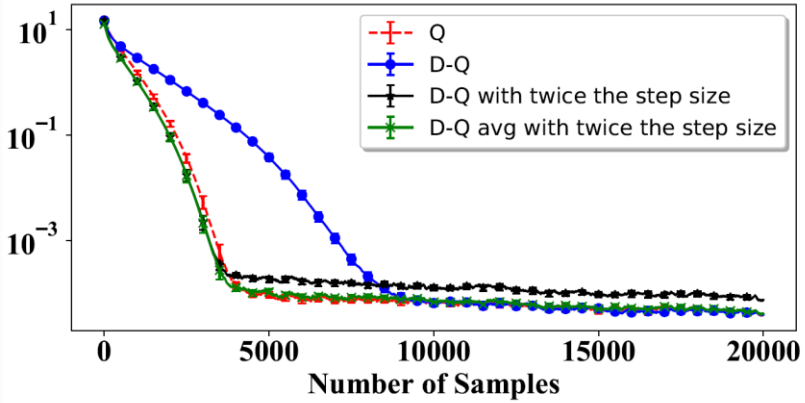
Baird's Example



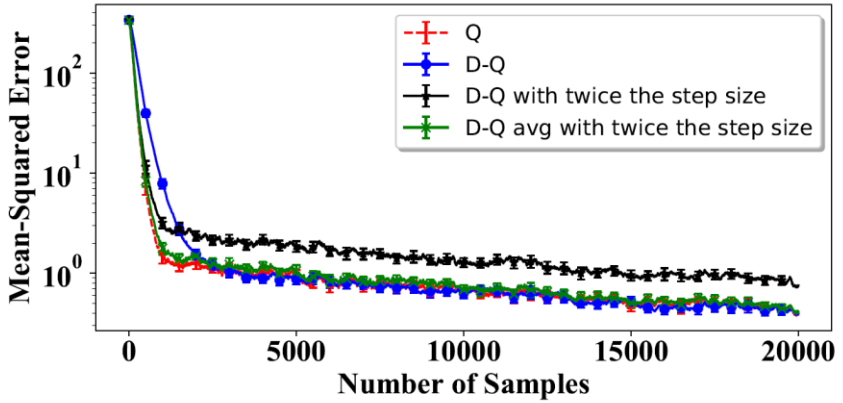
(a) Baird's Example [1]



(b) Zero Reward



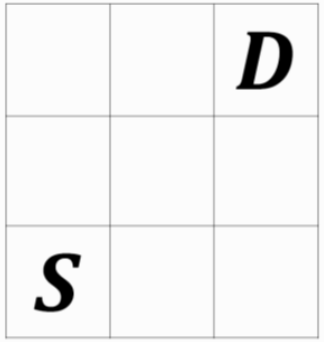
(c) Small Random Reward



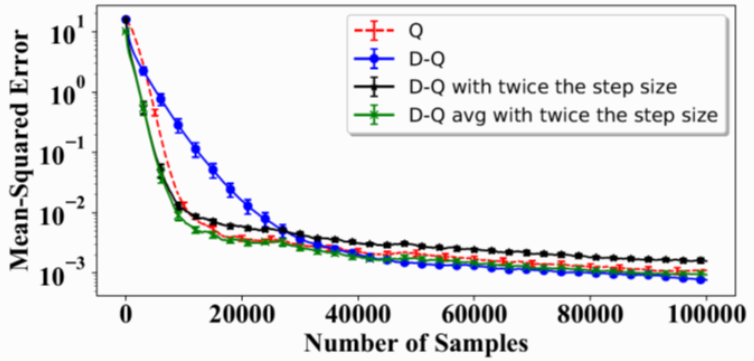
(d) Large Random Reward

Figure 1: Simulation results for Baird's example. The y-axis is in log scale.

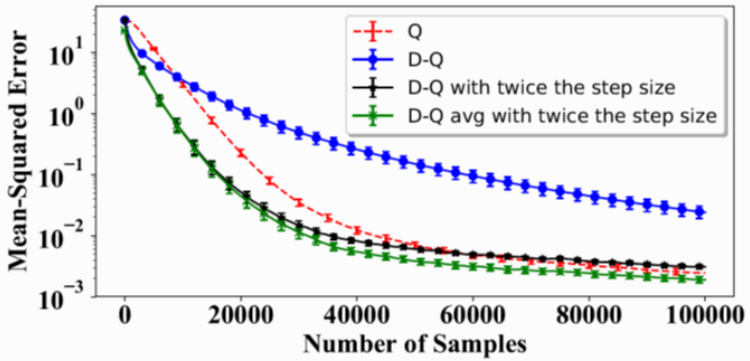
GridWorld



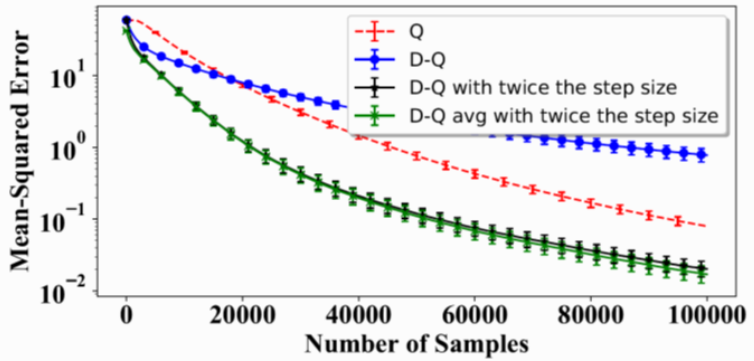
(a) An Example of 3×3 GridWorld



(b) 3×3 GridWorld



(c) 4×4 GridWorld



(d) 5×5 GridWorld

Figure 2: Simulation results for GridWorld with dimensions $3, 4, 5$. In all the three simulations, Double Q-learning with twice the step-size and averaged output outperforms Q-learning.

Observations

- Both from theoretical and numerical results:
  - Double Q-learning with the same stepsize converges slower than Q-learning;
  - Double Q-learning with twice stepsize can converge as fast as and even faster than Q-learning, but suffers from larger variance;
  - When using average estimator as the output, Double Q-learning with twice stepsize obtains both faster convergence rate and smaller mean-squared error.

Concluding Remarks

Existing optimization and control theory can help

- Build better understanding of common RL techniques
- Provide unified framework, finite sample analysis, and tight bounds
- Design principled, data-efficient, robust, and extensible RL algorithms