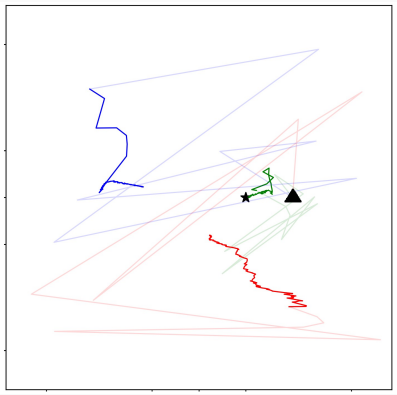
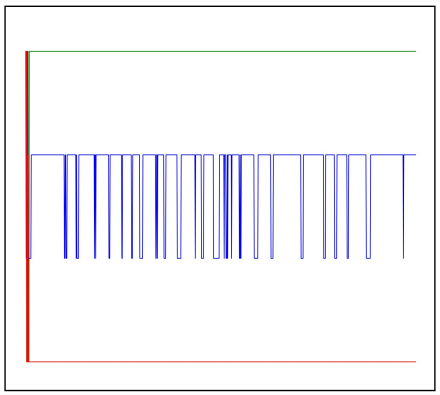
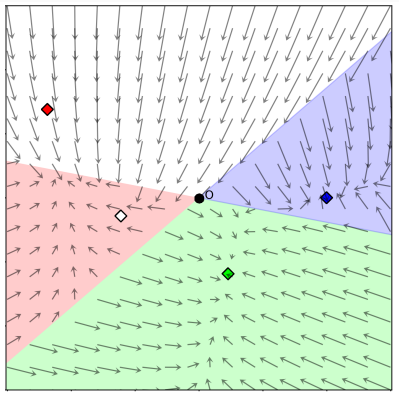
Does DQN Learn?

Gugan Thoppe | Aditya Gopalan

Indian Institute of Science, Bengaluru

Background Story

- Mnih et al. (2015):

  - DQN can play Atari games

  - Deep neural networks, ϵ-greedy, experience replay, target n/w

- Patterson et al. (2024):

  - " ... we observed (rare) catastrophic failure events for DQN across nearly every tested domain ... In Lunar Lander, some agents would simply fly off into oblivion, obtaining incredible amounts of negative reward until the episode was mercifully terminated ... In Cliff World, DQN would get stuck in a corner perpetually ... some agents would learn to jump into the cliff immediately to obtain massive negative rewards."

Natural Questions about DQN

- Is there monotonic improvement in the policies learned?
- Do the DQN iterates converge to a locally optimal policy?
- Does DQN at least learn a policy better than the initial one?

DQN Algorithm (Mnih et al., 2015)

Algorithm 1: deep Q-learning with experience replay.

Initialize replay memory D to capacity N

Initialize action-value function Q with random weights θ

Initialize target action-value function $\hat{Q}$ with weights $\theta^{-} = \theta$

For episode $= 1, M$ **do**

Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$

For $t = 1$,T **do**

With probability ε select a random action a_t

otherwise select $a_t = \text{argmax}_a Q(\phi(s_t), a; \theta)$

Execute action a_t in emulator and observe reward r_t and image x_{t+1}

Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$

Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D

Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D

Set $y_j = \begin{cases} r_j & \text{if episode terminates at step j+1} \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^{-}) & \text{otherwise} \end{cases}$

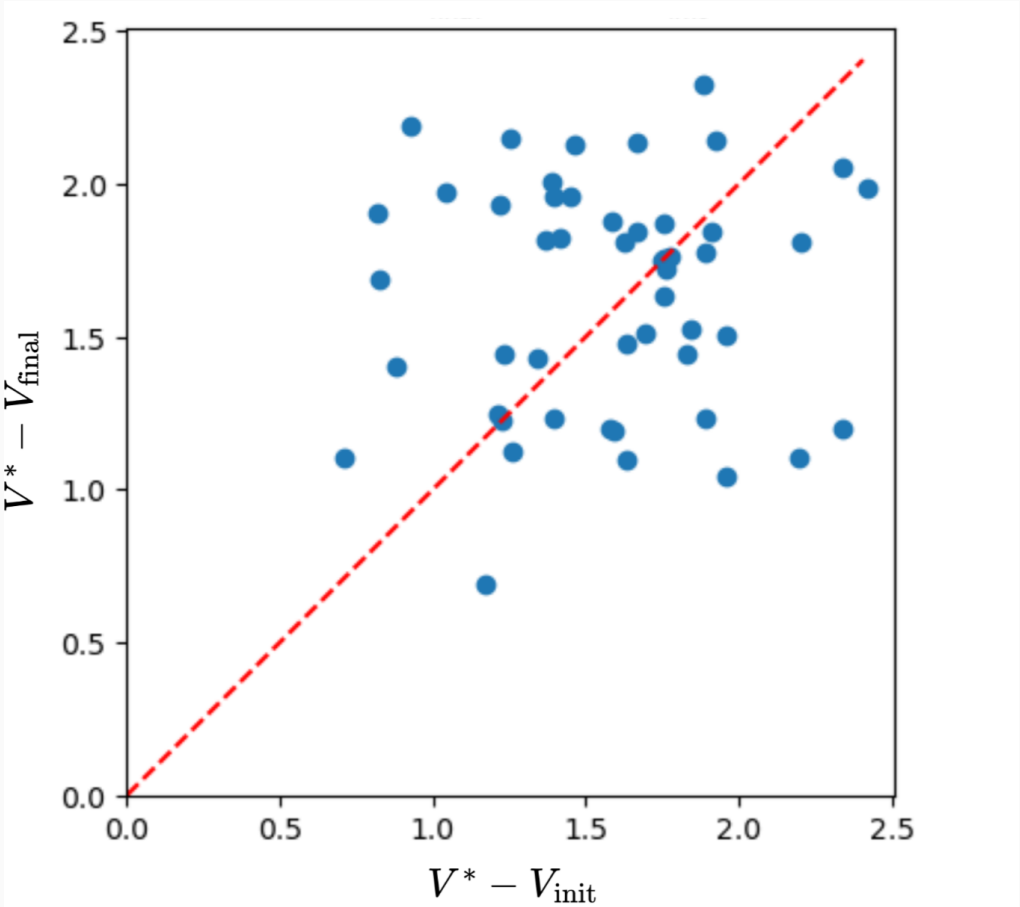
Perform a gradient descent step on $\left(y_j - Q(\phi_j, a_j; \theta)\right)^2$ with respect to the network parameters θ

Every C steps reset $\hat{Q} = Q$

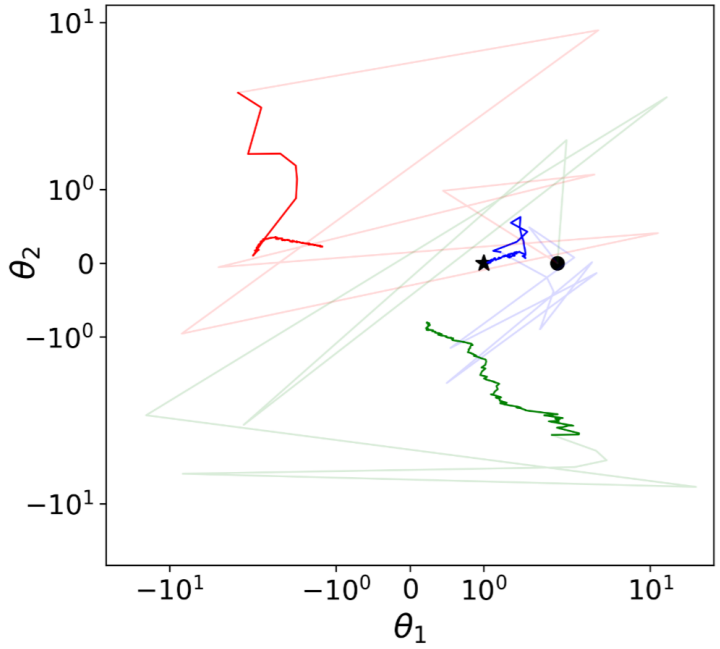
End For

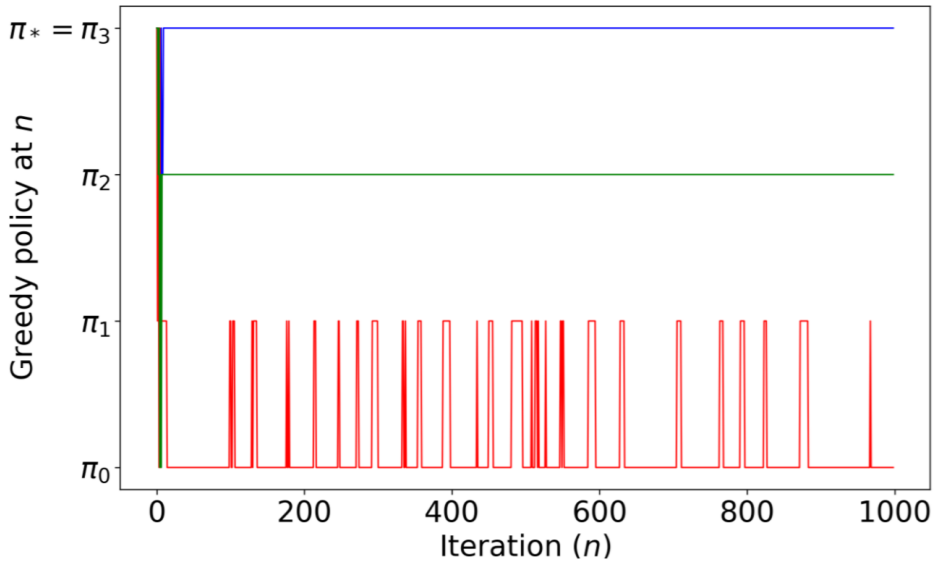
End For

Unreliability of DQN



Unrealiability of Linear DQN





Limitations of Existing ODE-based Analyses

- Any iterative RL method can be typically expressed as

$$\theta_{n+1} = \theta_n + \alpha_n[f(x_n) + M_{n+1}]$$

- For **nice** f, e.g., globally Lipschitz continuous, the ODE method shows that (θ_n) converges to an invariant set of $\dot{\theta}(t) = f(\theta(t))$.

- Such niceness holds in policy evaluation, but not in RL control

- The update rules in DQN-type methods are complex:

  1. non-linear even with linear-function approximation

  2. involve sampling from distributions that change with θ_n.

Limitations of Existing ODE-based Analyses

- ODE method has been made to work for DQN-type methods by viewing them as **general non-linear schemes** and making **restrictive assumptions** on the behavior policy:

    1. Fixed (Carvalo, 2020)
    2. Near-optimal (Melo et al., 2008; Chen et al., 2022)
    3. Smooth soft-max policy (Melo et al., 2008; Zou et al., 2019)

- With ϵ-greedy exploration, the situation is worse since f then is discontinuous and no analysis exists for it

Summary of Key Contributions

- Novel **Differential Inclusion** (DI) based analysis framework
- **Main Result**: Explanation of **all asymptotic behaviors** of linear Q-learning and linear SARSA employing ϵ-greedy exploration, (idealized) experience replay, and a target network
- Discovery of **traps** that impede learning
- Explanation of policy oscillation via a **sliding-mode attractor**

Synchronous Q-learning algorithm

- $Q_*(s,a) = \mathbb{E}_{s'}\left[r(s,a,s') + \gamma \max_a Q_*(s',a)\right]$

- **Update Rule**: For all s, a,

$$Q_{n+1}(s,a) = (1-\alpha_n)Q_n(s,a) + \alpha_n \left[r(s,a,s') + \gamma \max_{a'} Q_n(s',a')\right]$$
$$= Q_n(s,a) + \alpha_n \delta_n(s,a),$$

  where $s' \sim \mathbb{P}(\cdot|s,a)$ and

$$\delta_n(s,a) = r(s,a,s') + \gamma \max_{a'} Q_n(s',a') - Q_n(s,a)$$

  is the Temporal Difference (TD) error

- [Watkins and Dayan '92]: $Q_n \xrightarrow{a.s.} Q_*$ as $n \to \infty$

Asynchronous Q-learning

- Interact with the environment using a fixed or adaptive policy
- At time n, the interaction results in $(s_n, a_n, r(s_n, a_n, s_{n+1}), s_{n+1})$
- **Update rule**:

$$Q_{n+1}(s_n, a_n) = Q_n(s_n, a_n) + \alpha_n \delta_n$$

  where $\delta_n = r(s_n, a_n, s_{n+1}) + \gamma \max_{a'} Q_n(s_{n+1}, a') - Q_n(s_n, a_n)$

- [Borkar '08]: $Q_n \xrightarrow{a.s.} Q_*$, if all state-actions are visited frequently

Q-learning with Linear Function Approximation

- **Input**: 'Tall' feature matrix Φ of size $|\mathcal{S}||\mathcal{A}| \times d$, where $\phi(s_n, a_n)$ is the feature vector associated with (s_n, a_n)

- **Goal**: Find θ_* such that the optimal Q-function $Q_* \approx \Phi\theta_*$

- **Update rule**:

$$\theta_{n+1} = \theta_n + \alpha_n \, \delta_n \, \phi(s_n, a_n),$$

  where $\delta_n = r(s_n, a_n, s'_n) + \gamma \max_{a'} \phi^\top(s_{n+1}, a')\theta_n - \phi^\top(s_n, a_n)\theta_n$

- The update rule is nonlinear because of max and the asymptotic behavior depends on the interaction strategy

Different Interaction Strategies

- **Fixed-behavior**: $a_n \sim \pi(\cdot|s_n)$ and $s_{n+1} \sim \mathbb{P}(\cdot|s_n, a_n)$ for fixed π

  [Melo et al. '08, Chen et al. '19, Carvalo et al. '20]: Convergence guaranteed, but the quality of the limit depends on π

- ϵ-**greedy exploration**:

$$a_n = \begin{cases} \arg\max \phi^\top(s_n, a)\theta_n & \text{with probability } 1-\epsilon, \\ \text{random action} & \text{with probability } \epsilon, \end{cases}$$

  and $s_{n+1} \sim \mathbb{P}(\cdot|s_n, a_n)$

  Widely used in practice, but no theory

Main Result

- Let $b_{\bar{a}} = \mathbb{E}[\phi(s_n, a_n) r(s_n, a_n, s'_n)]$ and

$$A_{\bar{a}} = \mathbb{E}[\phi(s_n, a_n)\phi^\top(s_n, a_n) - \gamma\phi(s_n, a_n)\phi^\top(s'_n, a'_n)]$$

- Let $f(\theta)$ denote the **expected update direction**, i.e.,

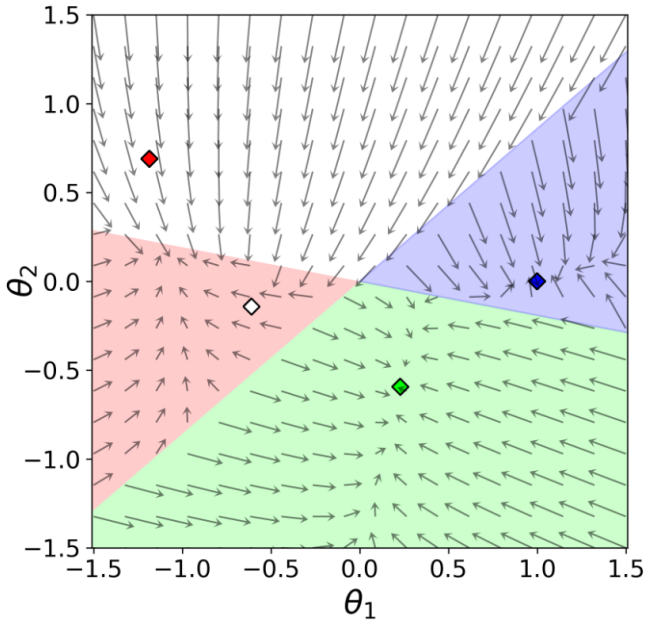
$$f(\theta) := \sum_{\bar{a} \in \mathcal{A}^S} (b_{\bar{a}} - A_{\bar{a}}\theta)\mathbb{1}[\theta \in \mathcal{P}_{\bar{a}}]$$

  Thus, $\theta_{n+1} = \theta_n + \alpha_n[f(\theta_n) + M_{n+1}]$

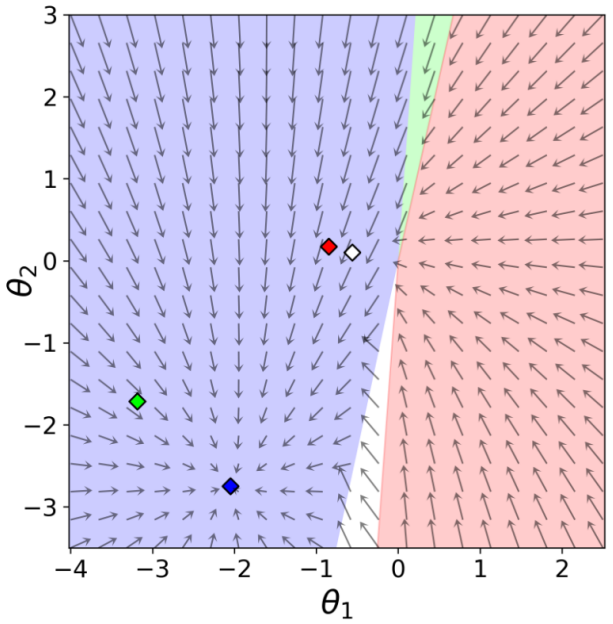
- Let $h(\theta) := \bigcap_{\delta > 0} \overline{\text{co}}(f(B(\theta, \delta)))$ be **all update directions near θ**

- The limiting behavior of (θ_n) is governed by $\dot{\theta}(t) \in h(\theta(t))$

Illustration of f and h maps



(a)



(b)

Assumptions

1. Φ has full column rank, and rewards are bounded
2. Markov chain under $\bar{a}^\epsilon$ has a unique stationary distribution
3. $\sum_{n \geq 0} \alpha_n = \infty$ and $\sum_{n \geq 0} \alpha_n^2 < \infty$

Unreliability of Linear DQN with ϵ-greedy exploration

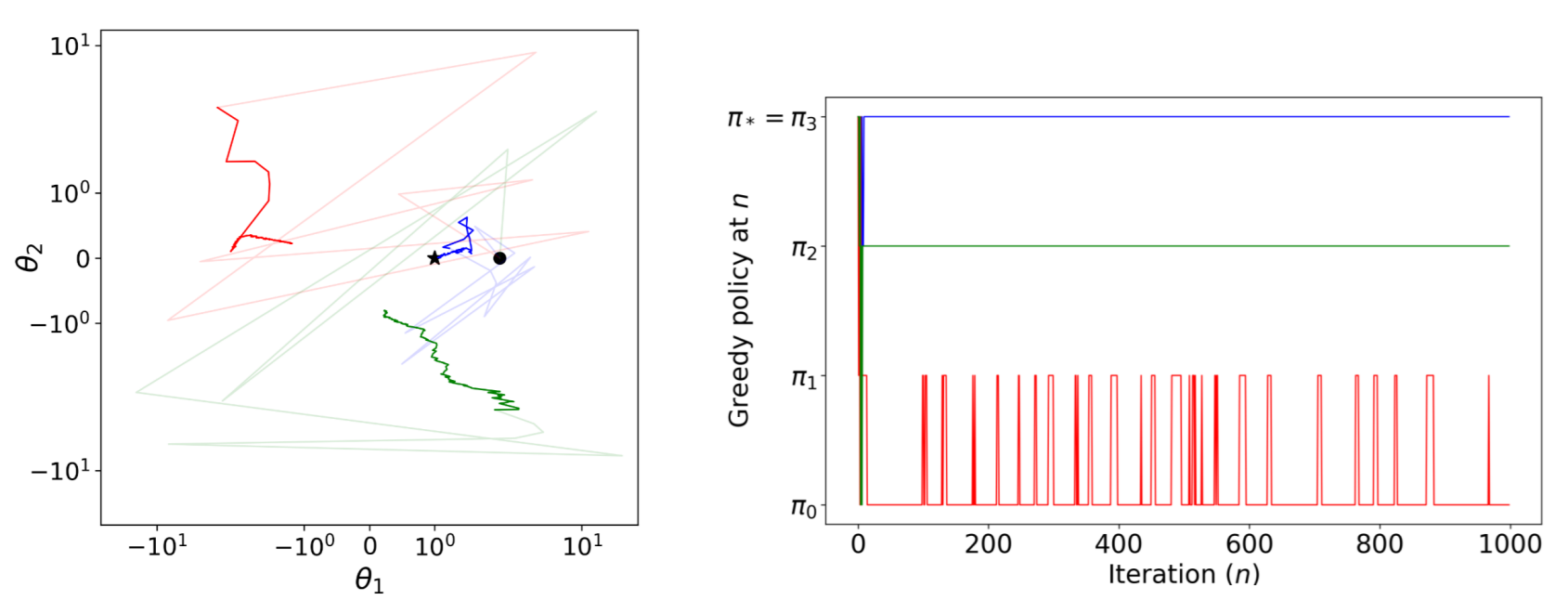
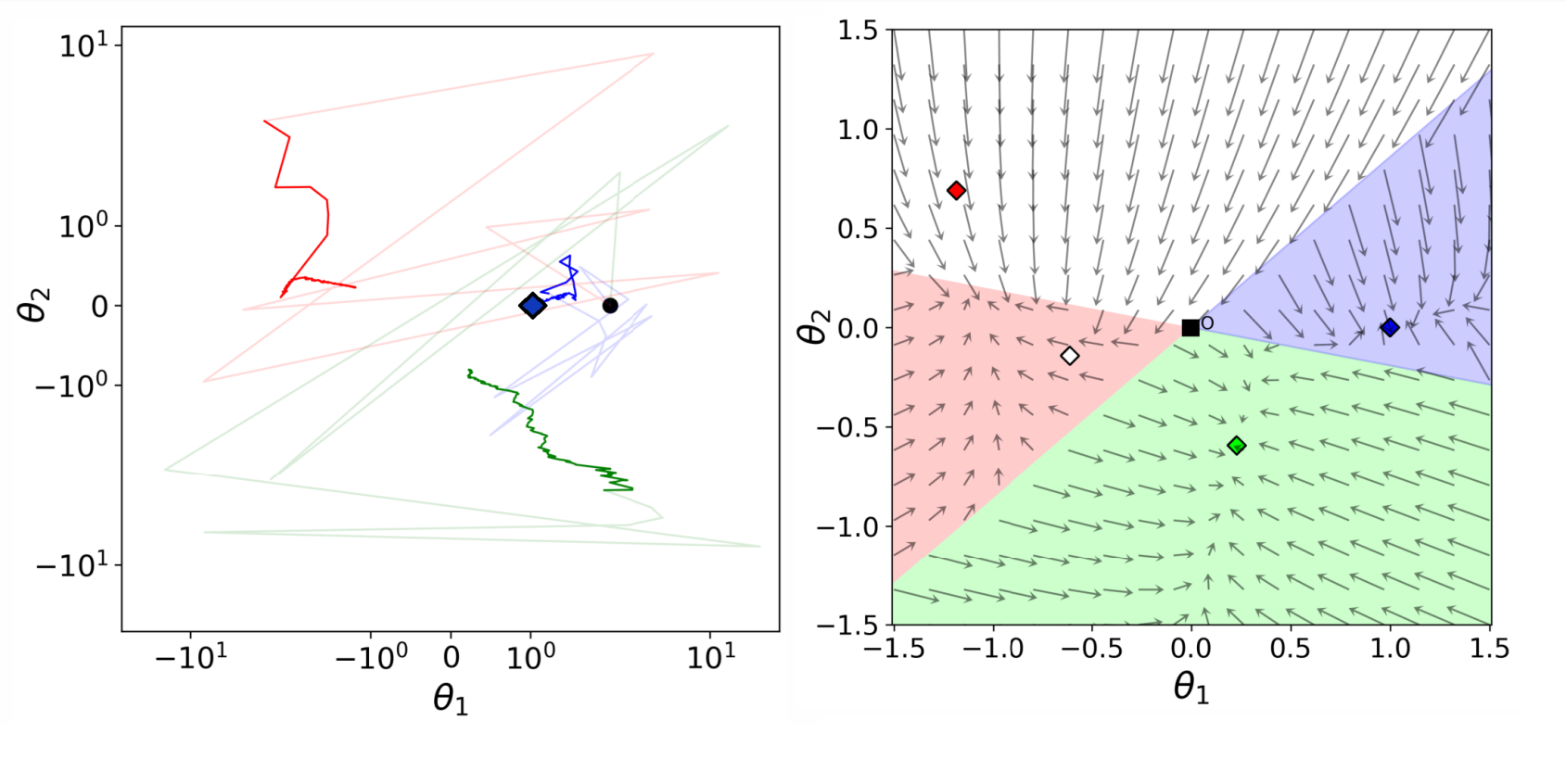
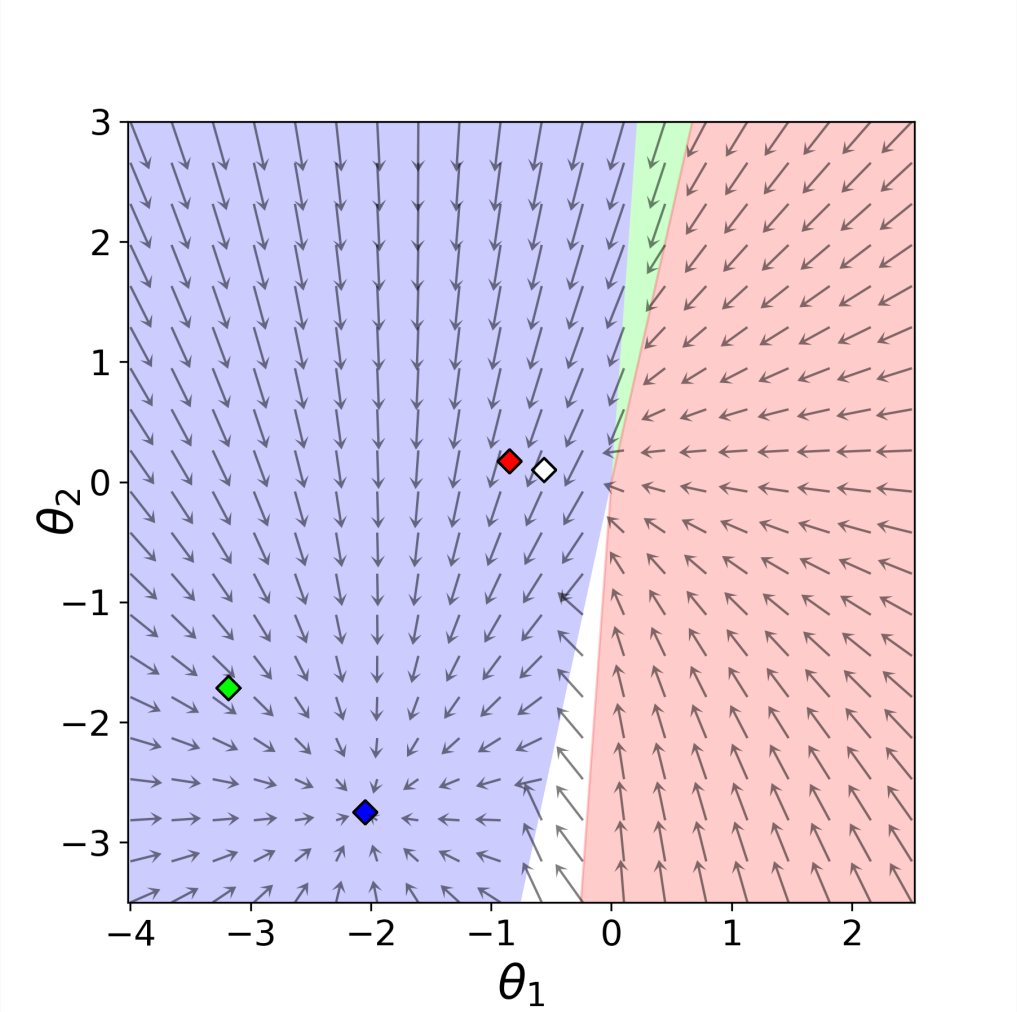


Figure 1: Linear DQN with realizable Q^* on a 2-state 2-action MDP

Resolving the Linear DQN behaviors



Traps and Worst-case Convergence



Conclusion

- **Good News**: Systematic way to study limiting behaviors of approximate value-based RL methods with greedy exploration
- **Bad News**: Limiting DI typically shows complex phenomena
- **Road Ahead**: Adopt a principled approach to fix it