Provably Efficient Model-free Constrained Reinforcement Learning Algorithm with Function Approximation

(Joint work with
Xingyu Zhou, Wayne State University,
Ness Shroff, The Ohio State University)

Arnob Ghosh,
The Ohio State University, Dept. of Electrical and Computer Engineering,
Research Scientist at the NSF AI-Edge Institute

Constrained MDP

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
    - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
- Modeled by Constrained MDP (CMDP); maximize ${}_{\pi}V^{\pi}_{r,1}(x_1)$

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
- Modeled by Constrained MDP (CMDP); maximize $_{\pi} V^{\pi}_{r,1}(x_1)$
- subject to $V^{\pi}_{g,1}(x_1) \geq b$

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
- Modeled by Constrained MDP (CMDP); maximize $_{\pi} V^{\pi}_{r,1}(x_1)$
- subject to $V^{\pi}_{g,1}(x_1) \geq b$
- $V^{\pi}_{r,1}$: value function for reward, $V^{\pi}_{g,1}$: Value function for utility (cost), agent receives $r_h(x_h, a_h)$ reward and $g_h(x_h, a_h)$ utility.

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
- Modeled by Constrained MDP (CMDP); maximize $_{\pi} V_{r,1}^{\pi}(x_1)$
- subject to $V_{g,1}^{\pi}(x_1) \geq b$
- $V_{r,1}^{\pi}$: value function for reward, $V_{g,1}^{\pi}$: Value function for utility (cost), agent receives $r_h(x_h, a_h)$ reward and $g_h(x_h, a_h)$ utility.
-

Constrained MDP

- Unconstrained MDP: Provably efficient RL algorithms exist **even** for linear-function approximation.
  - LSVI-UCB [Jin et al'20]: Regret bound is $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ where d is the dimension of the feature space, H is the no. of steps in episode, K is the number of episodes.
- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
- Modeled by Constrained MDP (CMDP); maximize $_{\pi} V^{\pi}_{r,1}(x_1)$
- subject to $V^{\pi}_{g,1}(x_1) \geq b$
- $V^{\pi}_{r,1}$: value function for reward, $V^{\pi}_{g,1}$: Value function for utility (cost), agent receives $r_h(x_h, a_h)$ reward and $g_h(x_h, a_h)$ utility.
- How to sequentially learn policies which will be close to optimal while also satisfying the constraint?

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
  - Learn a policy which is close to the optimality while also satisfying the constraints.

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
  - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
    - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**
    - Not valuable for the large-state space (potentially infinite).

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
    - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**
    - Not valuable for the large-state space (potentially infinite).
    - **Model-free is computationally more efficient (for large-state).**

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
  - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**
  - Not valuable for the large-state space (potentially infinite).
  - **Model-free is computationally more efficient (for large-state).**
- Only model-free result for **tabular-case**: sub-optimal regret $\tilde{\mathcal{O}}(K^{0.8})$ with zero violation [Wei et al.'22]

State-of-the Art Results

- Metric: $\mathrm{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \mathrm{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
    - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**
    - Not valuable for the large-state space (potentially infinite).
    - **Model-free is computationally more efficient (for large-state).**
- Only model-free result for **tabular-case**: sub-optimal regret $\tilde{\mathcal{O}}(K^{0.8})$ with zero violation [Wei et al.'22]
    - Not valuable for the large-state space, as regret bound depends on the state space.

State-of-the Art Results

- Metric: $\text{Regret}(K) = \sum_{k=1}^{K} (V_{r,1}^{\pi^*}(x_1) - V_{r,1}^{\pi_k}(x_1)), \quad \text{Violation}(K) = \sum_{k=1}^{K} (b - V_{g,1}^{\pi_k}(x_1)),$
- **Goal: Select policies over K episodes to minimize the regret and violation**
  - Learn a policy which is close to the optimality while also satisfying the constraints.
- Model-based: Tabular Case, **regret scales with state-action space.**
  - Not valuable for the large-state space (potentially infinite).
  - **Model-free is computationally more efficient (for large-state).**
- Only model-free result for **tabular-case**: sub-optimal regret $\tilde{\mathcal{O}}(K^{0.8})$ with zero violation [Wei et al.'22]
  - Not valuable for the large-state space, as regret bound depends on the state space.
- **Can we develop a provably-efficient model-free RL algorithm for constrained MDP with function approximation?**

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**

Our Contribution

- Can we develop a model-free provably RL for constrained MDP with function approximation?
  - Yes!! We can for linear CMDP.

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
  - **Yes!! We can for linear CMDP.**
    - Reward, utilities, and transition probabilities are linear in feature space.

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
    - **Yes!! We can for linear CMDP.**
        - Reward, utilities, and transition probabilities are linear in feature space.
    - **Our regret and violation upper bound $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ with high probability (informal).**

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
  - **Yes!! We can for linear CMDP.**
    - Reward, utilities, and transition probabilities are linear in feature space.
  - **Our regret and violation upper bound $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ with high probability (informal).**
    - **Improves the regret bound of the tabular case as linear MDP contains tabular.**

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
  - **Yes!! We can for linear CMDP.**
    - Reward, utilities, and transition probabilities are linear in feature space.
  - **Our regret and violation upper bound $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ with high probability (informal).**
    - **Improves the regret bound of the tabular case as linear MDP contains tabular.**
  - It is possible to achieve zero violation for large enough K with high probability while maintaining $\tilde{\mathcal{O}}(\sqrt{K})$ regret.

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
  - **Yes!! We can for linear CMDP.**
    - Reward, utilities, and transition probabilities are linear in feature space.
  - **Our regret and violation upper bound $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ with high probability (informal).**
    - **Improves the regret bound of the tabular case as linear MDP contains tabular.**
  - It is possible to achieve zero violation for large enough K with high probability while maintaining $\tilde{\mathcal{O}}(\sqrt{K})$ regret.
    - **Idea:** Consider an ϵ-tighter problem (consider $b+\epsilon$ instead of b in constraint).

Our Contribution

- **Can we develop a model-free provably RL for constrained MDP with function approximation?**
  - **Yes!! We can for linear CMDP.**
    - Reward, utilities, and transition probabilities are linear in feature space.
  - **Our regret and violation upper bound $\tilde{\mathcal{O}}(\sqrt{d^3H^4K})$ with high probability (informal).**
    - **Improves the regret bound of the tabular case as linear MDP contains tabular.**
  - It is possible to achieve zero violation for large enough K with high probability while maintaining $\tilde{\mathcal{O}}(\sqrt{K})$ regret.
    - **Idea:** Consider an ϵ-tighter problem (consider $b+\epsilon$ instead of b in constraint).
    - **However, scales the regret by an additional H factor.**

Algorithm (High-level idea)

Algorithm (High-level idea)

- Primal-Dual Adaptation of LSVI-UCB

Algorithm (High-level idea)

- Primal-Dual Adaptation of LSVI-UCB
- Natural idea: Take Lagrangian: $V_{r,1}^{\pi} + YV_{g,1}^{\pi}$, and solve it like an unconstrained version;.
  - Estimate *optimistic versions* $V_{r,1}^{k}, V_{g,1}^{k}$; set policy as **greedily** with respect to the composite $Q_{r,1}^{k}(x,a) + YQ_{g,1}^{k}(x,a)$; update the dual based on these estimated function.

Algorithm (High-level idea)

- Primal-Dual Adaptation of LSVI-UCB
- Natural idea: Take Lagrangian: $V_{r,1}^{\pi} + YV_{g,1}^{\pi}$, and solve it like an unconstrained version;.
    - Estimate *optimistic versions* $V_{r,1}^{k}, V_{g,1}^{k}$; set policy as **greedily** with respect to the composite $Q_{r,1}^{k}(x,a) + YQ_{g,1}^{k}(x,a)$; update the dual based on these estimated function.
- **However, it does not work!!**
    - Need to show uniform concentration bound for individual value-function— can not get ϵ -covering number for individual value function class which scales $O(K)$ for greedy-policy.

Algorithm (High-level idea)

- Primal-Dual Adaptation of LSVI-UCB
- Natural idea: Take Lagrangian: $V^{\pi}_{r,1} + YV^{\pi}_{g,1}$, and solve it like an unconstrained version;.
  - Estimate *optimistic versions* $V^k_{r,1}, V^k_{g,1}$; set policy as **greedily** with respect to the composite $Q^k_{r,1}(x,a) + YQ^k_{g,1}(x,a)$; update the dual based on these estimated function.
- **However, it does not work!!**
  - Need to show uniform concentration bound for individual value-function— can not get ϵ -covering number for individual value function class which scales $O(K)$ for greedy-policy.
- **Our solution: Use soft-max policy instead of greedy policy.**
  - Optimism result does not hold, but can bound the gap by controlling the temp. co-efficient.

Future Research Direction

- Multi-agent Domain

Future Research Direction

- Multi-agent Domain
- Non-linear Function Approximation.

Future Research Direction

- Multi-agent Domain
- Non-linear Function Approximation.
- Will it be possible to reduce the dependence on H or d? (The Lower bound for *unconstrained* case is $\Omega(d\sqrt{H^3K})$)

Future Research Direction

- Multi-agent Domain
- Non-linear Function Approximation.
- Will it be possible to reduce the dependence on H or d? (The Lower bound for *unconstrained* case is $\Omega(d\sqrt{H^3K})$)
- Check our paper, arXiv:2206.11889

References

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. *Provably efficient reinforcement learning with linear function approximation*. In Conference on Learning Theory, pages 2137– 2143. PMLR, 2020.

Honghao Wei, Xin Liu, and Lei Ying. *A provably-efficient model-free algorithm for constrained markov decision processes*. arXiv preprint arXiv:2106.01577

Yonathan Efroni, Shie Mannor, and Matteo Pirotta. *Exploration-exploitation in constrained mdps.* arXiv preprint arXiv:2003.02189.