Efficient and Resilient Algorithms for Stochastic Optimization & Reinforcement Learning

Shaocong Ma

April 14, 2024

University of Utah
s.ma@utah.edu

Overview

1. Review of Ph.D. Work
2. Resilient Stochastic Optimization over Dependent Data
3. Robust V-Learning for Markov Games with Model Uncertainty
4. Variance-Reduced Greedy-GQ Algorithm for Optimal Control
5. Future Directions

My Ph.D. Work

- Efficient and Resilient Algorithms for Stochastic Optimization
  (1) Efficient Stochastic Optimization with Random Reshuffling
  (2) Resilient Stochastic Optimization over Dependent Data

- Efficient and Resilient Algorithms for Reinforcement Learning
  (3) Variance-Reduced Off-Policy Algorithms
  (4) Robust Reinforcement Learning with Model Uncertainty

(1) Motivation of Studying Random Reshuffling

"Although the theory calls for picking examples randomly, it is usually faster to zip sequentially through the training set." [1]

```
from torch.utils.data import DataLoader, RandomSampler

# Random Reshuffling (by default)
dataloader = DataLoader(dataset, batch_size=1)

# Uniform Sampling
dataloader = DataLoader(dataset, batch_size=1,
                        shuffle=RandomSampler, shuffle=False)
```

Shuffling has been widely implemented in Tensorflow and Pytorch!

[1] L. Bottou. Stochastic Gradient Descent Tricks. In Neural networks: Tricks of the trade, pages 421–436. Springer, 2012.

(1) Efficient Stochastic Optimization with Random Reshuffling

- Previous work on SGD with random reshuffling:
  - Only in-expectation convergence guarantees.
  - Cannot cover non-convex scenarios.

(1) Efficient Stochastic Optimization with Random Reshuffling

- Previous work on SGD with random reshuffling:
  - Only in-expectation convergence guarantees.
  - Cannot cover non-convex scenarios.
- Empirical loss optimization:

$$\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(\theta).$$

(1) Efficient Stochastic Optimization with Random Reshuffling

- Previous work on SGD with random reshuffling:
  - Only in-expectation convergence guarantees.
  - Cannot cover non-convex scenarios.
- Empirical loss optimization:

$$\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(\theta).$$

- SGD with random reshuffling: Let $(1, 2, \ldots, n) \mapsto (\sigma_1, \sigma_2, \ldots, \sigma_n)$ be a random permutation.

$$\theta_{k+1} = \theta_k - \eta \nabla \ell_{\sigma_k}(\theta_k).$$

(1) Efficient Stochastic Optimization with Random Reshuffling

- Previous work on SGD with random reshuffling:
  - Only in-expectation convergence guarantees.
  - Cannot cover non-convex scenarios.
- Empirical loss optimization:

$$\min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell_i(\theta).$$

- SGD with random reshuffling: Let $(1, 2, \ldots, n) \mapsto (\sigma_1, \sigma_2, \ldots, \sigma_n)$ be a random permutation.

$$\theta_{k+1} = \theta_k - \eta \nabla \ell_{\sigma_k}(\theta_k).$$

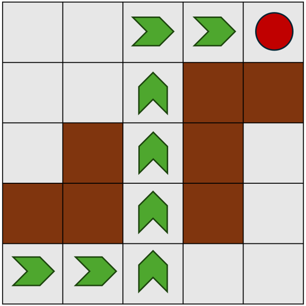
- Our contributions[2]:
  - $\{\theta_k\}$ has a unique limit point under over-parameterization.
  - Theoretical frameworks for non-convex objectives (quasi-strongly convex).
  - Theoretically explain why Random Reshuffling is better.

[2] Understanding the Impact of Model Incoherence on Convergence of Incremental SGD with Random Reshuffle. ICML 2020.

(2) Resilient Stochastic Optimization over Dependent Data

Many real-world applications need to handle the dependent data.

- Asset price (e.g. stock price, defaultable bond, ...)
- Reinforcement learning
- Online recommendation system
- ...



Example from RL:
The trajectory of RL usually forms a Markov chain in the left figure. Each green arrow represents the agent's state at a given time. Traditional optimization theory usually cannot explicitly characterize the impact of data dependence in the upper bound.

(2) Resilient Stochastic Optimization over Dependent Data

- Expected loss minimization:

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{\xi \sim \mu} f(\theta; \xi).$$

Data is generated from a stochastic process: $\{\xi_k\}; \mathbb{P}(\xi_k \in \cdot) \to \mu$.

(2) Resilient Stochastic Optimization over Dependent Data

- Expected loss minimization:

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{\xi \sim \mu} f(\theta; \xi).$$

  Data is generated from a stochastic process: $\{\xi_k\}$; $\mathbb{P}(\xi_k \in \cdot) \to \mu$.
- Previous work on online SGD:
  - Geometric mixing data.
  - Naive SGD. Unstable due to data dependency.

(2) Resilient Stochastic Optimization over Dependent Data

- Expected loss minimization:

$$\min_{\theta \in \mathbb{R}^d} \mathbb{E}_{\xi \sim \mu} f(\theta; \xi).$$

  Data is generated from a stochastic process: $\{\xi_k\}$; $\mathbb{P}(\xi_k \in \cdot) \to \mu$.

- Previous work on online SGD:
  - Geometric mixing data.
  - Naive SGD. Unstable due to data dependency.

- Our contributions[3]:

  SGD with sub-sampling: $\theta_{k+1} = \theta_k - \eta \nabla f(\theta_k; \xi_{\tau k})$.

  Mini-batch SGD: $\theta_{k+1} = \theta_k - \eta \sum_{i=0}^{B-1} \nabla f(\theta_k; \xi_{kB+i})$.

  - Arbitrary mixing data.
  - More robust w.r.t. dependent data.

[3] Data Sampling Affects the Complexity of Online SGD over Dependent Data. UAI 2022.

My Ph.D. Work

- Efficient and Resilient Algorithms for Stochastic Optimization
  (1) Efficient Stochastic Optimization with Random Reshuffling
  (2) Resilient Stochastic Optimization over Dependent Data

- Efficient and Resilient Algorithms for Reinforcement Learning
  (3) Variance-Reduced Off-Policy Algorithms
  (4) Robust Reinforcement Learning with Model Uncertainty

(3) Motivation of Using Variance Reduction

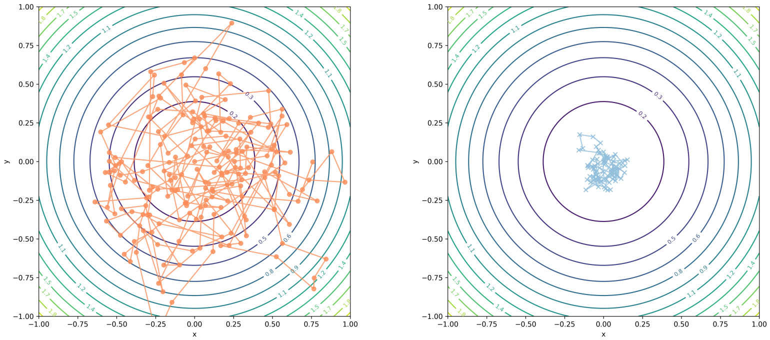


Figure 1: Illustration of trajectories of SGD algorithm (left) and SVRG algorithm (right) for minimizing $f(x, y) = E_{\xi,\zeta \sim N(0,1)}[(x-\xi)^2 + (y-\zeta)^2]$. The low-variance algorithm has much smaller variance near the optimal point $(0, 0)$ and performs much more stable.

(3) Variance-Reduced Off-Policy Algorithm for Policy Evaluation

- Off-Policy Policy Evaluation:

$$\mathrm{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \|\hat{V}_\theta - \Pi_{R_\theta} T^\pi \hat{V}_\theta\|^2.$$

$$(\text{Off-Policy TDC}) \begin{cases} \theta_{t+1} & = \theta_t + \alpha(A_t\theta_t + b_t + B_t\omega_t), \\ \omega_{t+1} & = \omega_t + \beta(A_t\theta_t + b_t + C_t\omega_t). \end{cases}$$

(3) Variance-Reduced Off-Policy Algorithm for Policy Evaluation

- Off-Policy Policy Evaluation:

$$\mathrm{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \|\hat{V}_\theta - \Pi_{R_\theta} T^\pi \hat{V}_\theta\|^2.$$

$$(\text{Off-Policy TDC}) \begin{cases} \theta_{t+1} & = \theta_t + \alpha(A_t\theta_t + b_t + B_t\omega_t), \\ \omega_{t+1} & = \omega_t + \beta(A_t\theta_t + b_t + C_t\omega_t). \end{cases}$$

- Previous work on TDC:
  - Convergence suffers from a large variance.
  - Two-time scale + Markovian sample: no appropriate solution

(3) Variance-Reduced Off-Policy Algorithm for Policy Evaluation

- Off-Policy Policy Evaluation:

$$\mathrm{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \| \hat{V}_\theta - \Pi_{R_\theta} T^\pi \hat{V}_\theta \|^2.$$

$$(\text{Off-Policy TDC}) \begin{cases} \theta_{t+1} & = \theta_t + \alpha(A_t\theta_t + b_t + B_t\omega_t), \\ \omega_{t+1} & = \omega_t + \beta(A_t\theta_t + b_t + C_t\omega_t). \end{cases}$$

- Previous work on TDC:
  - Convergence suffers from a large variance.
  - Two-time scale + Markovian sample: no appropriate solution
- Our contributions[4]:
  - Variance reduction for two-time scale algorithm over Markovian samples.
  - Best-known sample complexity.

[4] Variance-Reduced Off-Policy TDC Learning: Non-Asymptotic Convergence Analysis. NeurIPS 2020.

(3) Variance-Reduced Algorithm for Optimal Control

- Off-Policy Optimal Control:

$$\text{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \| Q_\theta - \Pi T^{\pi_\theta} Q_\theta \|^2.$$

$$(\text{Greedy-GQ}) \begin{cases} \theta_{t+1} & = \theta_t - \eta_\theta \left(-\delta_{t+1}(\theta_t)\phi_t + \gamma \left(\omega_t^\top \phi_t\right) \widehat{\phi}_{t+1}(\theta_t) \right), \\ \omega_{t+1} & = \omega_t - \eta_\omega \left(\phi_t^\top \omega_t - \delta_{t+1}(\theta_t) \right) \phi_t, \\ \pi_{\theta_{t+1}} & = \mathcal{P}(\phi^\top \theta_{t+1}). \end{cases}$$

(3) Variance-Reduced Algorithm for Optimal Control

- Off-Policy Optimal Control:

$$\mathrm{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \| Q_\theta - \Pi T^{\pi_\theta} Q_\theta \|^2.$$

$$(\text{Greedy-GQ}) \begin{cases} \theta_{t+1} & = \theta_t - \eta_\theta \left(-\delta_{t+1}(\theta_t)\, \phi_t + \gamma \left(\omega_t^\top \phi_t \right) \widehat{\phi}_{t+1}(\theta_t) \right), \\ \omega_{t+1} & = \omega_t - \eta_\omega \left(\phi_t^\top \omega_t - \delta_{t+1}(\theta_t) \right) \phi_t, \\ \pi_{\theta_{t+1}} & = \mathcal{P}(\phi^\top \theta_{t+1}). \end{cases}$$

- Previous work on Greedy-GQ:
    - Convergence suffers from a large variance.
    - Two-time scale + Markovian sample + Non-convex objectives

(3) Variance-Reduced Algorithm for Optimal Control

- Off-Policy Optimal Control:

$$\mathrm{MSPBE}(\theta) = \mathbb{E}_{\mu_b} \| Q_\theta - \Pi T^{\pi_\theta} Q_\theta \|^2.$$

$$(\text{Greedy-GQ}) \begin{cases} \theta_{t+1} & = \theta_t - \eta_\theta \left(-\delta_{t+1}(\theta_t)\, \phi_t + \gamma \left(\omega_t^\top \phi_t \right) \widehat{\phi}_{t+1}(\theta_t) \right), \\ \omega_{t+1} & = \omega_t - \eta_\omega \left(\phi_t^\top \omega_t - \delta_{t+1}(\theta_t) \right) \phi_t, \\ \pi_{\theta_{t+1}} & = \mathcal{P}(\phi^\top \theta_{t+1}). \end{cases}$$

- Previous work on Greedy-GQ:
  - Convergence suffers from a large variance.
  - Two-time scale + Markovian sample + Non-convex objectives
- Our contributions[5]:
  - Improved sample complexity from $\mathcal{O}(\epsilon^{-3})$ to $\mathcal{O}(\epsilon^{-2})$.

[5] Greedy-GQ with Variance Reduction: Finite-time Analysis and Improved Complexity. ICLR 2021.

(4) Motivation of Considering Model Uncertainty

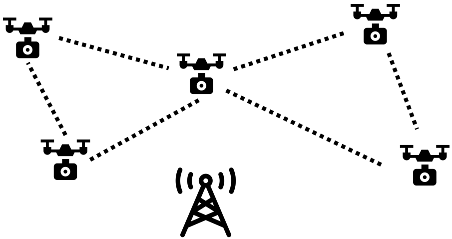


Figure 2: An UAV system with 5 drones.

What kinds of RL algorithms do we need?

- Noises from environments: resilience.
- Cannot communicate with the ground station: decentralization.
- Different tasks for different UAV: a general-sum stochastic game.

(4) Robust V-Learning for Markov Games with Model Uncertainty

- Previous work:
  - Aim to find robust NE, a PPAD-complete problem. (open)

(4) Robust V-Learning for Markov Games with Model Uncertainty

- Previous work:
  - Aim to find robust NE, a PPAD-complete problem. (open)
- Robust CE: for any player j, any stochastic modification $\phi^{(j)}$, any $s \in \mathcal{S}$,

$$V^{(j)}_{\pi,1}(s) \geq V^{(j)}_{\widetilde{\pi}^{(j)} \times \pi^{(\setminus j)},1}(s)$$

(4) Robust V-Learning for Markov Games with Model Uncertainty

- Previous work:
  - Aim to find robust NE, a PPAD-complete problem. (open)
- Robust CE: for any player j, any stochastic modification $\phi^{(j)}$, any $s \in \mathcal{S}$,

$$V^{(j)}_{\pi,1}(s) \geq V^{(j)}_{\widetilde{\pi}^{(j)} \times \pi^{(\setminus j)},1}(s)$$

- Our contributions[6]:
  - Propose Robust Correlated Equilibrium for robust Markov games.
  - Propose Robust V-Learning to find Robust Correlated Equilibrium.

[6] Decentralized Robust V-Learning for Solving Markov Games with Model Uncertainty. JMLR 2023.

(4) Robust Policy Optimization with Model Uncertainty

Real-world applications require resilient algorithms:

- Autonomous vehicle
- Robotics
- Healthcare
- Trading algorithm
- ...

(4) Robust Policy Optimization with Model Uncertainty

Worst-case value function:

$$V_u^\pi(s) := E\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t) \mid s_0 = s, P_u, \pi\Big],$$

$$V^\pi(s) := \min_u V_u^\pi(s).$$

Goal to find the optimal policy in the worst-case senario:

$$\pi^* = \arg\max V^\pi(s)$$

- Our contributions:
  - Monotonic policy improvement in the worst-case scenario.
  - Theoretical convergence guarantees to an optimal policy.

Resilient Stochastic Optimization over Dependent Data
(UAI 2022)

Expected Loss Optimization

- Expected loss optimization:

$$\min_{w \in \mathcal{W}} f(w) := \mathbb{E}_{\xi \sim \mu}[F(w; \xi)].$$

In practice, the data often cannot be directly sampled from the distribution μ. Instead, it comes from a stochastic process which limiting distribution is μ.

- Broad applications in machine learning:
  - Optimization theory
  - Portfolio optimization
  - Reinforcement learning
  - Quantitative trading
  - ...

Example from Real World: Reinforcement Learning

Example (Reinforcement Learning)

The data point (s_t, a_t, r_t, s_{t+1}) in RL comes from a trajectory:

$$s_1, a_1, r_1, s_2, a_2, r_2, \ldots$$

Not ind. + Non-identical distribution.

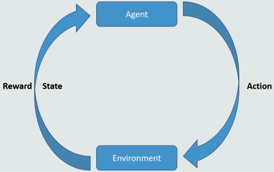


Figure 3: Agent-Environment Interaction

Example from Real World: Portfolio optimization

Example (Portfolio optimization)

Given n assets. Build a long-term portfolio w such that

$$\mathrm{Var} := w^T \Sigma w$$

is minimized (Σ is the covariance matrix of asset prices).

Data process: Daily estimated covariance matrix based on pre-processed daily asset returns: $\{\Sigma_1, \Sigma_2, \dots\}$

SGD update: $w \leftarrow w - \eta \cdot (\Sigma_i + \Sigma_i^T) w$.

- Biased gradient: $\mathbb{E}\Sigma_i \neq \mathbb{E}_{\Sigma \sim \Xi}\Sigma$.
- Data dependence: $\mathbb{E}\Sigma_i \Sigma_j \neq \mathbb{E}\Sigma_i \mathbb{E}\Sigma_j$.

Motivation: Robust Algorithm for Dependent Data Processes

- Previous work on dependent data:
    - Strong geometric mixing assumption.
    - Weak in-expectation convergence guarantees.
    - The performance of SGD is significantly affected by data dependency.
- Our analysis for SGD algorithm:
    - Arbitrary mixing assumption.
    - Strong high-probability convergence guarantees.
    - Propose multiple methods to reduce the impact of data dependency.

Characterization of Data Dependency:

We use the mixing coefficient to measure the data dependency.

Definition

- $\{\xi_t\}_t$: a process with a stationary distribution μ.
- $\mathbb{P}(\xi_{t+k} \in \cdot | \mathcal{F}_t)$: the dist. of ξ_{t+k} cond. on $\mathcal{F}_t$.
- d_{TV}: the total variation distance.

The process $\{\xi_t\}_t$ is called ϕ-*mixing* if

$$\underbrace{\phi(k)}_{\text{mixing coef.}} := \sup_{t \in \mathbb{N}, A \in \mathcal{F}_t} 2 d_{\mathrm{TV}}\big(\mathbb{P}(\xi_{t+k} \in \cdot | A), \mu\big) \to 0,$$

as $k \to \infty$.

Assumptions

$$\min_{w \in \mathcal{W}} f(w) := \mathbb{E}_{\xi \sim \mu}[F(w; \xi)].$$

Assumptions

- For every ξ, function $F(\cdot, \xi)$ is G-Lipschitz continuous over the domain $\mathcal{W}$.
- Function $f(\cdot)$ is convex and bounded below, i.e. $f(w^*) := \inf_{w \in \mathcal{W}} f(w) > -\infty$.
- $\mathcal{W}$ is convex and compact with bounded diameter R.
- There is a non-increasing sequence $\{\kappa(t)\}_t$ such that $\|w(t+1) - w(t)\| \leq \kappa(t)$.

Online SGD Algorithms

- Naive SGD:

$$w(t+1) = w(t) - \eta_t \nabla F(w(t); \xi_t).$$

- Sub-sampling SGD:

$$w(t+1) = w(t) - \eta_t \nabla F(w(t); \xi_{tr+1}).$$

- Mini-batch SGD:

$$w(t+1) = w(t) - \frac{\eta_t}{B} \sum_{\xi \in x_t} \nabla F(w(t); \xi).$$

Data dependence model	$\phi_\xi(k)$	SGD	SGD w/ subsampling	Mini-batch SGD
Geometric ϕ-mixing (Weakly dependent)	$\exp(-k^\theta)$, $\theta > 0$	$\mathcal{O}(\epsilon^{-2}(\log \epsilon^{-1})^{\frac{2}{\theta}})$	$\mathcal{O}(\epsilon^{-2}(\log \epsilon^{-1})^{\frac{1}{\theta}})$	$\mathcal{O}(\epsilon^{-2})$
Fast algebraic ϕ-mixing (Medium dependent)	$k^{-\theta}$, $\theta \geq 1$	$\mathcal{O}(\epsilon^{-2-\frac{2}{\theta}})$	$\mathcal{O}(\epsilon^{-2-\frac{1}{\theta}})$	$\widetilde{\mathcal{O}}(\epsilon^{-2})$
Slow algebraic ϕ-mixing (Highly dependent)	$k^{-\theta}$, $0 < \theta < 1$	$\mathcal{O}(\epsilon^{-2-\frac{2}{\theta}})$	$\mathcal{O}(\epsilon^{-2-\frac{1}{\theta}})$	$\mathcal{O}(\epsilon^{-1-\frac{1}{\theta}})$

Convergence of Sub-Sampling SGD

Theorem

$$f(\widehat{w}_n) - f(w^*) \le \mathcal{O}\Big(\frac{1}{\sqrt{n}} + \underbrace{\inf_{\tau\in\mathbb{N}}\Big\{\frac{(\tau-1)}{\sqrt{n}} + \sqrt{\frac{\tau}{n}\log\frac{\tau}{\delta}} + \phi(r\tau)\Big\}}_{\textit{Err. caused by data dependence}}\Big).$$

- Geometric ϕ-mixing data:

  Sample complexity is $rn = \mathcal{O}\big(\epsilon^{-2}(\log\frac{1}{\epsilon})^{\frac{1}{\theta}}\big)$.

- Algebraic ϕ-mixing data:

  Sample complexity is $rn = \mathcal{O}\big(\epsilon^{-2-\frac{1}{\theta}}\big)$.

Convergence of Mini-Batch SGD

Theorem

$$f(\widehat{w}_n) - f(w^*) \le \widetilde{\mathcal{O}}\Bigg(\sqrt{\frac{\sum_{j=1}^{n}\phi(j)}{nB}} + \frac{GR(\tau-1)}{n} + \frac{1}{nB}\sum_{i=1}^{B}\phi(\tau B + i) + \sqrt{\frac{\tau}{nB}}\Big(B^{-\frac{1}{4}} + \Big[\sum_{i=1}^{B}\phi(i)\Big]^{\frac{1}{4}}\Big)\Bigg).$$

- Geometric ϕ-mixing data:

  Sample complexity is $nB = \mathcal{O}\big(\epsilon^{-2}(\log\frac{1}{\epsilon})^{\frac{1}{\theta}}\big)$.
- Fast algebraic ϕ-mixing data:

  Sample complexity is $nB = \tilde{\mathcal{O}}\big(\epsilon^{-2}\big)$.
- Slow algebraic ϕ-mixing data:

  Sample complexity is $nB = \mathcal{O}\big(\epsilon^{-1-\frac{1}{\theta}}\big)$.

Robust V-Learning for Markov Games with Model Uncertainty (JMLR 2023)

Robust Markov Games

- Broad applications in machine learning:
  - Game theory
  - Insurance
  - Portfolio optimization
  - Multi-UAV systems
  - ...
- Robust Value Function

$$V_{\pi,h}^{(j)}(s) := \inf_{\widetilde{\mathbb{P}} \in \mathcal{P}} \mathbb{E}\Big[\sum_{\ell=h}^{H} r_\ell^{(j)}(s_\ell, a_\ell)\Big| s_h = s, \pi, \widetilde{\mathbb{P}}\Big].$$

Real-World Example: Insurance

Example

A specific insurance policy costs \$1.

- If a covered event occurs (e.g., a disease outbreak, death, etc.), the policy pays out \$2.
- If the event does not occur, the \$1 spent on buying the policy is lost.

Question: Given that the probability of the insured event happening is extremely low, is it rational to purchase this insurance policy?

The worst case: the covered event occurs.

Robust NE

Definition (Robust Nash Equilibrium)

A joint policy π is called a robust NE if

(i) for all h, π_h is a product policy;

(ii) for any player j with any policy $\widetilde{\pi}^{(j)}$, we have $V^{(j)}_{\pi,1}(s) \geq V^{(j)}_{\widetilde{\pi}^{(j)} \times \pi^{(\setminus j)},1}(s)$ for all $s \in \mathcal{S}$.

Solving the NE of a general-sum multi-player game is PPAD-complete.

Robust CE

Definition (Robust Correlated Equilibrium)

A joint policy π is called a robust CE if for any player j and any stochastic modification $\phi^{(j)}$, it holds that $V^{(j)}_{\pi,1}(s) \geq V^{(j)}_{\phi^{(j)}\circ\pi,1}(s)$ for all states $s \in \mathcal{S}$.

There are many algorithms solving the CE of a general-sum multi-player game in polynomial time.

Fundamental Properties of Robust CE

Propositions

1. Any robust NE is a robust CE.
2. There exists a robust CE which is not a robust NE.

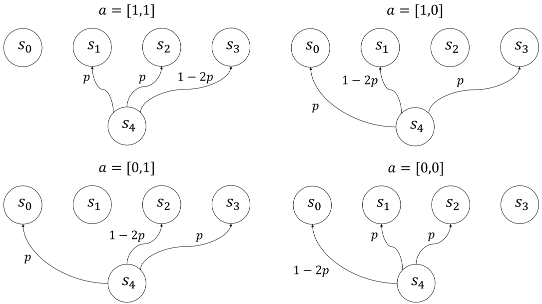


Figure 4: For any $p \in (\frac{10}{29}, \frac{1}{2})$, there are two robust NE: $\pi_1(a = [0,1] | s = s_4) = 1$ and $\pi_1(a = [1,0] | s = s_4) = 1$ (π_2 can be arbitrary). Any convex combination of these two policies is a robust CE but not robust NE.

Decentralized Robust V-Learning

At the h-step of an episode:

- Each agent takes its action $a_h^{(j)}$. Transfer to the next state to s_{h+1}.
- Receive reward $r_h^{(j)}$ and set $t := N_{k+1,h}^{(j)}(s_h) \leftarrow N_{k,h}^{(j)}(s_h) + 1$
- Let $\widetilde{V}_{k+1,h}^{(j)} \leftarrow \widetilde{V}_{k,h}^{(j)}$, $V_{k+1,h}^{(j)} \leftarrow V_{k,h}^{(j)}$, $\pi_{k+1,h}^{(j)} \leftarrow \pi_{k,h}^{(j)}$.

$$\begin{aligned}
\widetilde{V}_{k+1,h}^{(j)}(s_h) &= (1-\alpha_t)\widetilde{V}_{k,h}^{(j)}(s_h) + \alpha_t \left(r_h^{(j)} + \widehat{\sigma}_{\mathcal{P}_h(s_h,a_h)}(V_{k,h+1}^{(j)}) + \beta_t^{(j)} \right) \\
V_{k+1,h}^{(j)}(s_h) &= \min\{H+1-h, \widetilde{V}_{k+1,h}^{(j)}(s_h)\} \\
\pi_{k+1,h}^{(j)}(\cdot|s_h) &= \mathtt{ADV_BANDIT}\left(t, a_h, 1 - \frac{r_h^{(j)} + \widehat{\sigma}_{\mathcal{P}_h(s_h,a_h)}(V_{k,h+1}^{(j)})}{H}, \pi_{k,h}^{(j)}(\cdot|s_h)\right)
\end{aligned}$$

Definition and Assumptions

- *Uncertainty diameter:*

$$D := \max_{h,s,a,a'} \max_{\mathbb{P}\in\mathcal{P}_h(s,a),\widetilde{\mathbb{P}}\in\mathcal{P}_h(s,a')} \|\mathbb{P}(\cdot) - \widetilde{\mathbb{P}}(\cdot)\|_\infty.$$

- *Estimation error:*

$$\mathfrak{e} := \sup_{h,s,a,V} \left|\sigma_{\mathcal{P}_h(s,a)}(V) - \widehat{\sigma}_{\mathcal{P}_h(s,a)}(V)\right|,$$

  where the supremum is taken over all bounded value tables that satisfy $0 \le V(s) \le H+1$ for all s.
- *State exploration:*

$$p_{\min} := \min_{s,h,k} \mathbb{P}(s_{k,h} = s),$$

  which denotes the minimum probability of visiting an arbitrary state s at any step h of any episode k.

Decentralized Robust V-Learning (Small Uncertainty Set)

Theorem

For any $D \geq 0$,

$$\max_{j \in [J]} \max_{s \in \mathcal{S}} \left(V^{(j)}_{\phi^* \circ \hat{\pi},1}(s) - V^{(j)}_{\hat{\pi},1}(s)\right) \leq 5DSH^2 + \mathcal{O}\left(H\left(A\sqrt{\frac{H^3 S}{K} \ln \frac{mKHSA^2}{\delta}} + \mathfrak{e}\right)\right).$$

If the uncertainty diameter $D \leq \frac{\epsilon}{SH^2}$ and the approximation error $\mathfrak{e} = \mathcal{O}(\frac{\epsilon}{H})$,

the ϵ-accuracy is guaranteed with $K = \widetilde{\mathcal{O}}(SA^2 H^5 \epsilon^{-2})$ episodes.

Decentralized Robust V-Learning (Sufficient Exploration)

Theorem

For any D and $p_{\min}$ satisfying $\frac{\epsilon}{SH^2} \le D < \frac{p_{\min}}{H}$,

$$\max_{j\in[J]} \max_{s\in\mathcal{S}} \left(V^{(j)}_{\phi^*\circ\hat{\pi},1}(s) - V^{(j)}_{\hat{\pi},1}(s)\right) \le \mathcal{O}\left(\frac{H}{p_{\min} - DH}\left(A\sqrt{\frac{H^3S}{K}\ln\frac{mKHSA^2}{\delta}} + \mathfrak{e}\right)\right).$$

If the state exploration $p_{\min} > \frac{\epsilon}{SH}$ and the approximation error $\mathfrak{e} = \mathcal{O}(\frac{\epsilon p_{\min}}{H})$,

the ϵ-accuracy is guaranteed with $K = \widetilde{\mathcal{O}}(SA^2H^5p_{\min}^{-2}\epsilon^{-2})$ episodes.

Variance-Reduced Greedy-GQ Algorithm for Optimal Control (ICLR 2021)

Optimal Control

- Broad applications in machine learning:
  - Robotic control
  - Recommendation systems
  - Large language model
  - ...
- V-function (the state value function):

$$V^\pi(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s\Big].$$

- Q-function (the action-state value function):

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

- Optimal control:

$$\pi^* = \arg\max Q^\pi(s_0, a_0).$$

Optimal Control

- Bellman operator T^π:

$$T^\pi Q(s,a) = \mathbb{E}_{s',a'}[r(s,a,s') + \gamma Q(s',a')].$$

- Value iteration algorithm:

$$(T^\pi)^n Q \to Q^*.$$

Exact $T^\pi Q$ is hard to obtain with function approximation.

- Mean Squared Projected Bellman Error (MSPBE):

$$J(\theta) := \frac{1}{2}\|\Pi T^{\pi_\theta} Q_\theta - Q_\theta\|^2_{\mu_{s,a}},$$

Motivation:

- A single sample $x_t = (s_t, a_t, r_t, s_{t+1})$.
- Greedy-GQ:

$$\begin{cases} \theta_{t+1} = \theta_t - \eta_\theta\big(- \delta_{t+1}(\theta_t)\phi_t + \gamma(\omega_t^\top \phi_t)\widehat{\phi}_{t+1}(\theta_t)\big), \\ \omega_{t+1} = \omega_t - \eta_\omega\big(\phi_t^\top \omega_t - \delta_{t+1}(\theta_t)\big)\phi_t, \\ \pi_{\theta_{t+1}} = \mathcal{P}(\phi^\top \theta_{t+1}). \end{cases}$$

Can we develop the variance reduction for Greedy-GQ?

- Challenges:
  - Two-time scale.
  - Non-convex.

VR-Greedy-GQ Algorithm

- Gradient given a single sample:

$$G_{x_t}(\theta,\omega) := -\delta_{t+1}(\theta)\phi_t + \gamma(\omega^\top \phi_t)\widehat{\phi}_{t+1}(\theta),$$
$$H_{x_t}(\theta,\omega) := \big(\phi_t^\top \omega - \delta_{t+1}(\theta)\big)\phi_t.$$

- Gradient over a batch:

$$\widetilde{G}^{(m)} = \frac{1}{M}\sum_{k=(m-1)M}^{mM-1} G_{x_k}(\widetilde{\theta}^{(m)},\widetilde{\omega}^{(m)}), \quad \widetilde{H}^{(m)} = \frac{1}{M}\sum_{k=(m-1)M}^{mM-1} H_{x_k}(\widetilde{\theta}^{(m)},\widetilde{\omega}^{(m)}).$$

- VR-Greedy-GQ:

$$\theta_{t+1}^{(m)} = \Pi_R\Big[\theta_t^{(m)} - \eta_\theta\big(G_t^{(m)}(\theta_t^{(m)},\omega_t^{(m)}) - G_t^{(m)}(\widetilde{\theta}^{(m)},\widetilde{\omega}^{(m)}) + \widetilde{G}^{(m)}\big)\Big]$$
$$\omega_{t+1}^{(m)} = \Pi_R\Big[\omega_t^{(m)} - \eta_\omega\big(H_t^{(m)}(\theta_t^{(m)},\omega_t^{(m)}) - H_t^{(m)}(\tilde{\theta}^{(m)},\tilde{\omega}^{(m)}) + \widetilde{H}^{(m)}\big)\Big]$$
$$\text{Policy improvement}: \pi_{\theta_{t+1}^{(m)}} \leftarrow \mathcal{P}(\phi^\top \theta_{t+1}^{(m)}).$$

VR-Greedy-GQ Algorithm: Contributions

- Finite-time convergence analysis:

  Two-time scale + Markovian sample + Non-convex objectives.
- Improved sample complexity from $\mathcal{O}(\epsilon^{-3})$ to $\mathcal{O}(\epsilon^{-2})$.
- Novel two-time scale variance reduction structure.

Assumptions

- Feature boundedness
  The feature vectors are uniformly bounded, i.e., $\|\phi_{s,a}\| \leq 1$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$.
- Policy smoothness
  The mapping $\theta \mapsto \pi_\theta$ is k_1-Lipschitz and k_2-smooth.
- Problem solvability
  The matrix $C := \mathbb{E}[\phi_{s,a}\phi_{s,a}^\top]$ is non-singular.
- Geometric uniform ergodicity
  There exists $\Lambda > 0$ and $\rho \in (0, 1)$ such that

$$\sup_{s \in \mathcal{S}} d_{\mathrm{TV}}\big(\mathbb{P}(s_t | s_0 = s), \mu\big) \leq \Lambda \rho^t,$$

  for any $t > 0$, where d_{TV} is the total-variation distance.

Convergence Analysis

Theorem

$$\mathbb{E}\|\nabla J(\theta_\xi^{(\zeta)})\|^2 \leq \mathcal{O}\Big(\frac{1}{\eta_\theta TM} + \frac{1}{T}\big(\eta_\omega + \frac{\eta_\theta^2}{\eta_\omega^2}\big) + \big(\eta_\omega + \frac{\eta_\theta^2}{\eta_\omega^2}\big)^2 + \frac{1}{M}\Big),$$

Set $\eta_\theta = \mathcal{O}(\frac{1}{M})$, $\eta_\omega = \mathcal{O}(\eta_\theta^{2/3})$, and set $T, M = \mathcal{O}(\epsilon^{-1})$.

Sample complexity for achieving $\mathbb{E}\|\nabla J(\theta_\xi^{(\zeta)})\|^2 \leq \epsilon$ is $TM = \mathcal{O}(\epsilon^{-2})$.

Experiments

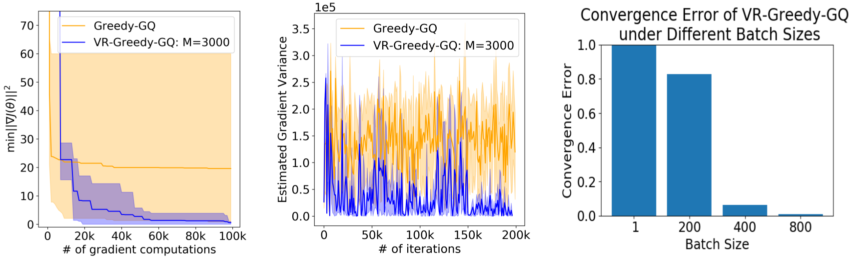


Figure 5: Comparison of Greedy-GQ and VR-Greedy-GQ in solving the Frozen Lake problem.

Future Directions

Future Directions

- Robust Policy Gradient algorithm (New, under review of JMLR).
  - Maximize the worst-case expected reward:

$$\max_{\pi} \min_{u} V_u^{\pi}(s) := E\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t) \mid s_0 = s, P_u, \pi\Big].$$

  - The agent is robust against the environment change (e.g. the transition kernel P_u)
  - We propose Robust Conservative Policy Iteration:
    - Iteration complexity $\mathcal{O}(\frac{1}{1-\gamma}\frac{1}{\epsilon^2})$,
    - Sample complexity: $\mathcal{O}(\epsilon^{-5})$.

Future Directions

- Zeroth-Order Optimization (New, under review of TMLR).
  - Minimize the hybrid loss with external parameters:

$$\min_{\theta, M_{\text{coarse}}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\Big(\text{NN}_\theta(M_{\text{fine}}, O^i_{\text{coarse}}), O^i_{\text{fine}}\Big),$$

  - M_{coarse} is the non-auto-differentiable external parameter.
  - We estimate a part of gradient:

$$\frac{\partial \mathcal{L}}{\partial M_{\text{coarse}}} = \frac{\partial \mathcal{L}}{\partial O_{\text{coarse}}} \cdot \underbrace{\frac{\partial O_{\text{coarse}}}{\partial M_{\text{coarse}}}}_{\text{Grad. Estimation.}} .$$

Thank You!

Publication List

- **Shaocong Ma**, James Diffenderfer, Bhavya Kailkhura, and Yi Zhou. ”End-to-End Mesh Optimization of a Hybrid Deep Learning Black-Box PDE Solver.” Advances in Neural Information Processing Systems (NeurIPS), Machine Learning and the Physical Sciences Workshop. 2023.
- **Shaocong Ma**, Ziyi Chen, Shaofeng Zou, and Yi Zhou. ”Decentralized Robust V-learning for Solving Markov Games with Model Uncertainty.” Journal of Machine Learning Research (JMLR). 2023.
- Ziyi Chen, **Shaocong Ma**, and Yi Zhou. ”Finding correlated equilibrium of constrained Markov game: A primal-dual approach.” Advances in Neural Information Processing Systems (NeurIPS). 2022.
- **Shaocong Ma**, Ziyi Chen, Yi Zhou, Kaiyi Ji, and Yingbin Liang. ”Data sampling affects the complexity of online sgd over dependent data.” In Uncertainty in Artificial Intelligence (UAI). 2022.

Publication List

- Ziyi Chen, **Shaocong Ma**, and Yi Zhou. ”Accelerated proximal alternating gradient-descent-ascent for nonconvex minimax machine learning.” 2022 IEEE International Symposium on Information Theory (ISIT). IEEE, 2022.
- Ziyi Chen, **Shaocong Ma**, and Yi Zhou. ”Sample efficient stochastic policy extragradient algorithm for zero-sum markov game.” The International Conference on Learning Representations (ICLR). 2021.
- **Shaocong Ma**, Ziyi Chen, Yi Zhou, Shaofeng Zou. ”Greedy-GQ with variance reduction: Finite-time analysis and improved complexity.” The International Conference on Learning Representations (ICLR). 2021.
- **Shaocong Ma**, Yi Zhou, and Shaofeng Zou. ”Variance-reduced off-policy TDC learning: Non-asymptotic convergence analysis.” Advances in Neural Information Processing Systems (NeurIPS). 2020.
- **Shaocong Ma**, and Yi Zhou. ”Understanding the impact of model incoherence on convergence of incremental sgd with random reshuffle.” International Conference on Machine Learning (ICML). 2020.