An Accelerated Algorithm for Stochastic Bilevel Optimization under Unbounded Smoothness

Xiaochuan Gong Jie Hao Mingrui Liu

Department of Computer Science, George Mason University

GEORGE MASON UNIVERSITY

NEURAL INFORMATION PROCESSING SYSTEMS

Overview

The bilevel optimization (BO) problem is formulated as:

$$\min_{x\in\mathbb{R}^{d_x}} \Phi(x) = f(x, y^*(x)) \qquad \text{(UL)}$$
$$\text{s.t.,}\quad y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_y}} g(x,y) \qquad \text{(LL)}$$

Goal: Design an accelerated bilevel optimization algorithm with:

- Upper-level (UL) function: nonconvex with unbounded smoothness
- Lower-level (LL) function: strongly convex

Contributions:

- We propose AccBO, which achieves an improved $\widetilde{O}(\epsilon^{-3})$ complexity to find ϵ-stationary points under the unbounded smoothness setting.
- Our proof relies on a novel lemma analyzing the dynamics of SNAG under distribution drift with high probability for the lower-level variable.
- Experiments on deep AUC maximization and data hyper-cleaning validate the effectiveness of our proposed algorithm.

Motivation and Problem Setting

- It is empirically observed in [1] that the smoothness constant scales linearly with the gradient norm of RNN (the upper-level function f) for both level variables.

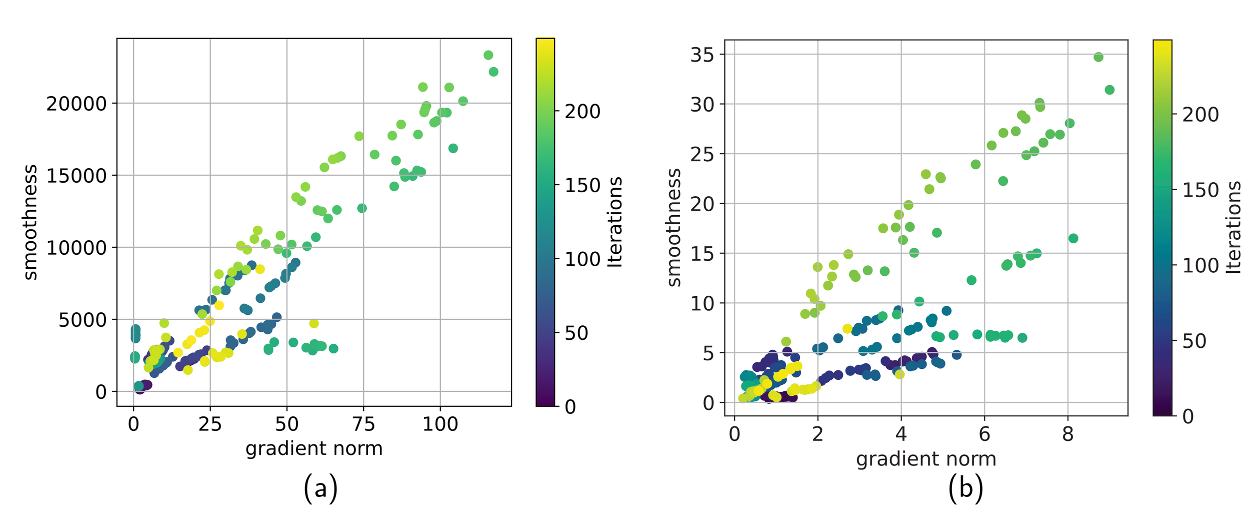

Figure 1:(a) UL variable: Local gradient Lipschitz constant vs. its gradient norm of an RNN. (b) LL variable: Local gradient Lipschitz constant vs. its gradient norm of an RNN.

- $(L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1})$-smoothness: Let $z=(x,y)$ and $z'=(x',y')$, there exists $L_{x,0}, L_{x,1}, L_{y,0}, L_{y,1} > 0$ such that if $\|z-z'\| \le 1/\sqrt{L_{x,1}^2 + L_{y,1}^2}$, then

$$\|\nabla_x f(z) - \nabla_x f(z')\| \le (L_{x,0} + L_{x,1}\|\nabla_x f(z)\|)\|z-z'\|,$$
$$\|\nabla_y f(z) - \nabla_y f(z')\| \le (L_{y,0} + L_{y,1}\|\nabla_y f(z)\|)\|z-z'\|.$$

Main Challenges and Solutions

- $\nabla\Phi(x) = \nabla_x f(x, y^*(x)) - \nabla_{xy}^2 g(x, y^*(x))[\nabla_{yy}^2 g(x, y^*(x))]^{-1}\nabla_y f(x, y^*(x))$. We use Neumann series approach [2] to estimate the hypergradient.
- Recent work [1, 3] only achieve $\widetilde{O}(\epsilon^{-4})$ complexity under the same setting. $\Longrightarrow$: Update the UL variable by normalized SGD with recursive momentum and the LL variable by SNAG with averaging.
- Potential function argument with an expectation-based analysis for L-smooth objectives [4, 5] cannot be applied due to randomness dependency issue. $\Longrightarrow$: Introduce novel techniques for analyzing the dynamics of SNAG under distribution drift with high probability for the LL variable.

AccBO Algorithm

Algorithm 1 Stochastic Nesterov Accelerated Gradient Method (SNAG)

1: **Input:** $x, \tilde{y}_{-1}, \tilde{y}_0, \tilde{\alpha}, T_0$ # $\mathtt{SNAG}(x, \tilde{y}_0, \tilde{\alpha}, T_0)$
2: **for** $t = 0, 1, \ldots, T_0 - 1$ **do**
3: Sample $\tilde{\pi}_t$ from distribution $\mathcal{D}_g$
4: $\tilde{z}_t = \tilde{y}_t + \gamma(\tilde{y}_t - \tilde{y}_{t-1})$
5: $\tilde{y}_{t+1} = \tilde{z}_t - \tilde{\alpha}\nabla_y G(x, \tilde{z}_t; \tilde{\pi}_t)$
6: **end for**

Algorithm 2 Accelerated Bilevel Optimization algorithm (AccBO)

1: **Input:** $\alpha^{\text{init}}, \alpha, \alpha', \beta, \gamma, \eta, \tau, I, S, T_0, T$, set $x_0, y_0^{\text{init}} = 0$
2: $y_0 = \mathtt{SNAG}(x_0, y_0^{\text{init}}, \alpha^{\text{init}}, T_0)$, and set $y_{-1} = \hat{y}_0 = y_0$ # Warm-start
3: **for** $t = 0, 1, \ldots, T-1$ **do**
4: Sample $\mathsf{q}(Q) \sim \text{Uniform}\{0, \ldots, Q-1\}$ and $\{\zeta_{t,s}^{(0)}, \ldots, \zeta_{t,s}^{(\mathsf{q}(Q))}\}_{s=1}^S \sim \mathcal{D}_g$
5: Sample $\{\xi_{t,s}\}_{s=1}^S \sim \mathcal{D}_f$, denote $\bar{\xi}_t \cup_{s=1}^S \{\mathsf{q}(Q), \xi_{t,s}, \zeta_{t,s}^{(0)}, \ldots, \zeta_{t,s}^{(\mathsf{q}(Q))}\}$
6: # Lower-Level: Stochastic Nesterov Accelerated Gradient Descent with Averaging
7: # Option I: from Line $8 \sim 9$ (for one-dimensional quadratic lower-level function)
8: $z_t = y_t + \gamma(y_t - y_{t-1})$
9: $y_{t+1} = z_t - \alpha\nabla_y G(x_t, z_t; \pi_t)$, where $\pi_t \sim \mathcal{D}_g$
10: # Option II: from Line $11 \sim 20$ (for general strongly convex lower-level function)
11: **if** $t > 0$ and t is a multiple of I **then**
12: Set $y_t^0 = y_t^{-1} = y_t$
13: **for** $j = 0, 1, \ldots, N-1$ **do**
14: $z_t^j = y_t^j + \gamma(y_t^j - y_t^{j-1})$
15: $y_t^{j+1} = z_t^j - \alpha\nabla_y G(x_t, z_t^j; \pi_t^j)$, where $\pi_t^j \sim \mathcal{D}_g$
16: **end for**
17: $y_{t+1} = y_t^{N+1}$
18: **else**
19: $y_{t+1} = y_t$
20: **end if**
21: $\hat{y}_{t+1} = (1-\tau)\hat{y}_t + \tau y_{t+1}$ # Averaging
22: # Upper-Level: Normalized Stochastic Gradient Descent with Recursive Momentum
23: $m_t = \beta m_{t-1} + (1-\beta)\bar{\nabla} f(x_t, \hat{y}_t; \bar{\xi}_t) + \beta(\bar{\nabla} f(x_t, \hat{y}_t; \bar{\xi}_t) - \bar{\nabla} f(x_{t-1}, \hat{y}_{t-1}; \bar{\xi}_t))$ if $t \ge 1$ else $m_0 = \bar{\nabla} f(x_0, \hat{y}_0; \bar{\xi}_0)$
24: $x_{t+1} = x_t - \eta\frac{m_t}{\|m_t\|}$
25: **end for**

Main Results

Under suitable choice of parameters, for small $\epsilon > 0$ and any given $\delta \in (0,1)$, both Option I and Option II guarantee with probability at least $1-\delta$ (over the randomness for updating $\{y_t\}$) that $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\| \le O(\epsilon)$, where the expectation is taken over all randomness except for that in updating $\{y_t\}$. The oracle complexity is $\widetilde{O}(\epsilon^{-3})$.

- Our complexity result is optimal in terms of ϵ up to logarithmic factors.

Key Lemma: SNAG under Distribution Drift

- (With drift) Let $\phi_t(y) = g(x_t, y) = \frac{\mu}{2}\|y - y^*(x_t)\|^2$ and $y \in \mathbb{R}$, then (V_t is the potential function)

$$V_t \le \left(1 - \frac{\sqrt{\mu\alpha}}{4}\right)^t V_0 + \left(2\alpha\sigma_{g,1}^2 + \frac{80\eta^2 l_{g,1}^2}{\mu^2\alpha}\right)\ln\frac{eT}{\delta}.$$

- (Without drift) Let $\phi_t(y) = g(x_t, y)$ be any strongly convex function in y and $y \in \mathbb{R}^d$, then

$$V_t \le \left(1 - \frac{\sqrt{\mu\alpha}}{4}\right)^t V_0 + 2\alpha\sigma_{g,1}^2 \ln\frac{eT}{\delta}.$$

Experiments

Deep AUC Maximization:

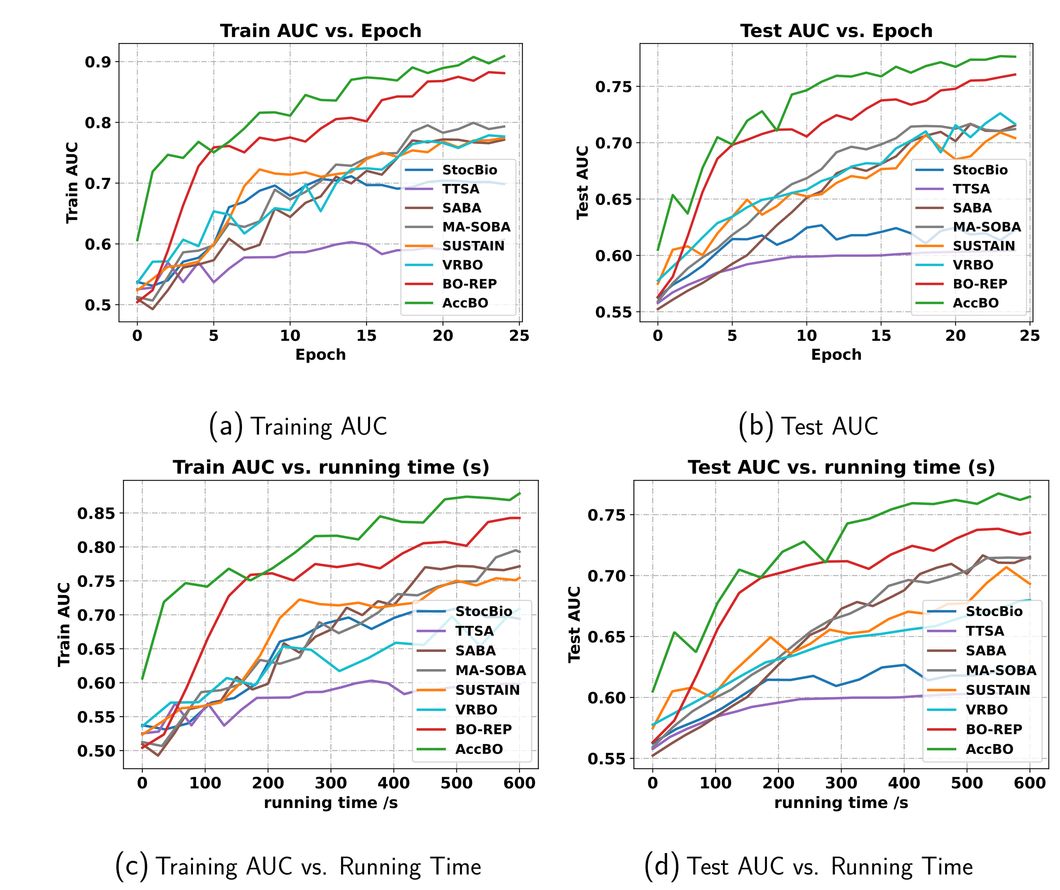

Figure 2:Results of bilevel optimization on deep AUC maximization. Figures (a), (b) are the results over epochs, and (c), (d) are the results over running time.

Data Hyper-Cleaning:

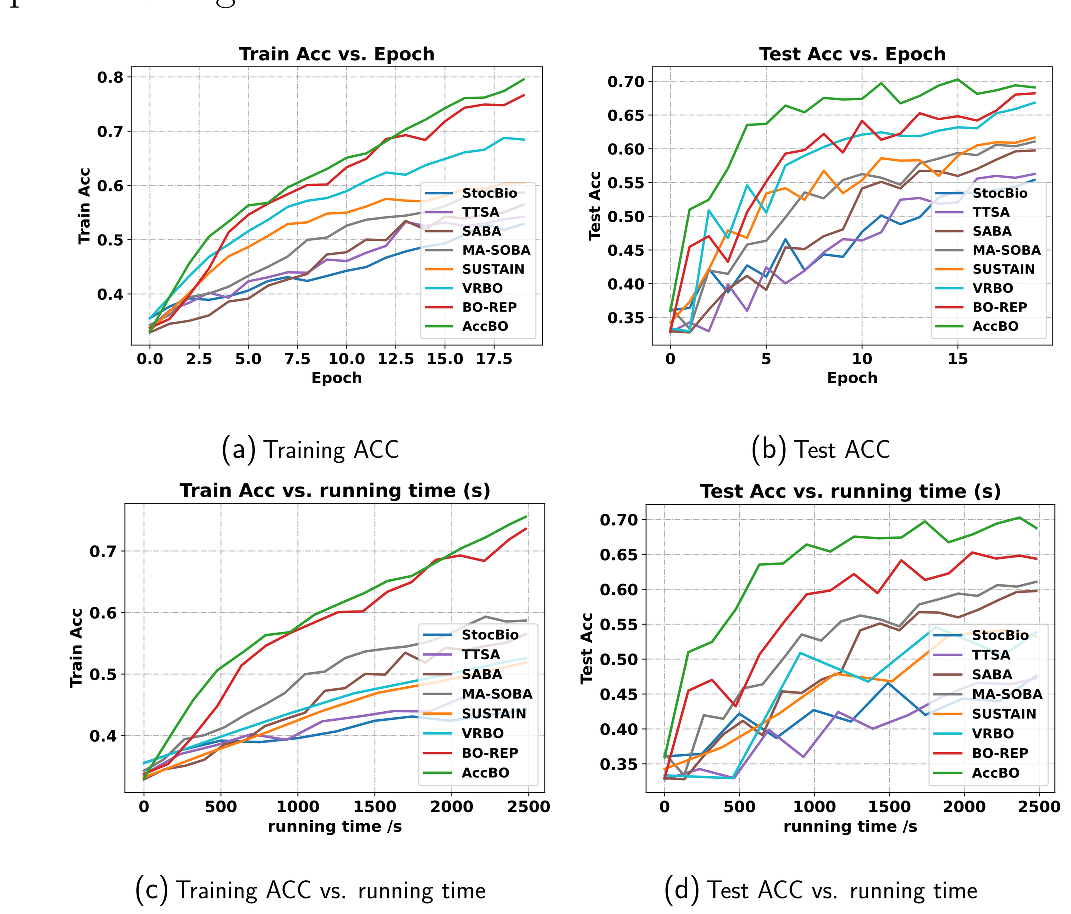

Figure 3:Results of bilevel optimization on data hyper-cleaning with $p = 0.1$. Figures (a), (b) are the results over epochs, and (c), (d) are the results over running time.

References

[1] Jie Hao, Xiaochuan Gong, and Mingrui Liu. Bilevel optimization under unbounded smoothness: A new algorithm and convergence analysis. In *The Twelfth International Conference on Learning Representations*, 2024.

[2] Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

[3] Xiaochuan Gong, Jie Hao, and Mingrui Liu. A nearly optimal single loop algorithm for stochastic bilevel optimization under unbounded smoothness. In *Forty-first International Conference on Machine Learning*, 2024.

[4] Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. *Advances in neural information processing systems*, 34:30271–30283, 2021.

[5] Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. *Advances in Neural Information Processing Systems*, 34:13670–13682, 2021.