Double Momentum Method for Lower-Level Constrained Bilevel Optimization

Wanli Shi [1,2] Yi Chang [1] Bin Gu [1,2,†]

[1]School of Artificial Intelligence, Jilin University, China [2]Mohamed bin Zayed University of Artificial Intelligence

Summary of Our Work

- Existing hypergradient-based methods for the lower-level constrained bilevel optimization problem are based on restrictive assumptions, i.e., optimality conditions satisfy the differentiability and invertibility conditions and lack a solid analysis of the convergence rate.
- In our paper, leveraging the theory of nonsmooth implicit function theorems, we propose a new method to calculate the hypergradient of LCBO without using restrictive assumptions. We also propose a new method to approximate the hypergradient based on randomized smoothing and the Neumann series.
- Using our hypergradient approximation, we propose a *single-loop single-timescale* algorithm for the lower-level constrained BO problems. We prove our methods can return a (δ, ϵ)-stationary point with $\tilde{\mathcal{O}}(d_2^2\epsilon^{-4})$ iterations.

Table 1. Several representative hypergradient approximation methods for the lower-level constrained BO problem. (The last column shows iteration numbers to find a stationary point. The gray color is used to highlight the main limitations of the listed algorithms)

Method	$F(x)$	Loop	Timescale	LL. Constraint	Restrictive Conditions	Iterations
AiPOD	Smooth	Double	×	Affine sets	Not need	$\tilde{\mathcal{O}}(\epsilon^{-2})$
IG-AL	Nonsmooth	Double	×	Half space	Not need	×
RMD-PCD	Nonsmooth	Double	×	Norm set	$y^*(x)$ is differentiable	×
JaxOpt	Nonsmooth	Double	×	Convex set	$y^*(x)$ is differentiable	×
DMLCBO (Ours)	Nonsmooth	Single	Single	Convex set	Not need	$\tilde{\mathcal{O}}(d_2^2\epsilon^{-4})$

Hypergrdient of Lower-level Constrained Bilevel Optimization Problem

In this paper, we consider the following problem-setting

$$\min_{x\in\mathbb{R}^{d_1}} F(x) = f(x, y^*(x)) \quad s.t.\quad y^*(x) = \arg\min_{y\in\mathcal{Y}\subseteq\mathbb{R}^{d_2}} g(x,y). \tag{1}$$

where $\mathcal{Y}$ is a convex subset of $\mathbb{R}^{d_2}$, g is strongly convex.

- Under strongly convex assumptions, we have the optimal solution to the lower-level problem is Lipschitz continuous with constant L_g/μ_g.
- Using the definition of generalized gradient, generalized Jacobian [1], and the Lipschitz continuousness of $y*(x)$, we have

$$\partial F(x) = \nabla_x f(x, y^*(x)) + (\partial y^*(x))^\top \nabla_y f(x, y^*(x)) \tag{2}$$

where $\partial F(x)$ is the generalized gradient and $\partial y^*(x)$ is generalized Jacobian.

- Using the Jacobian Chain Rule [1] on the optimal condition of the lower-level problem,

$$y^*(x) = \mathcal{P}_\mathcal{Y}(y^*(x) - \eta\nabla_y g(x, y^*(x))), \tag{3}$$

where $\eta > 0$ and $\mathcal{P}_\mathcal{Y}(\cdot)$ is the projection operator, we can obtain the following hypergradient

$$\partial F(x) = \{h | h = \nabla_x f(x, y^*(x)) - \eta\nabla^2_{xy} g(x, y^*(x)) H^\top \cdot \left[I_{d_2} - (I_{d_2} - \eta\nabla^2_{yy} g(x, y^*(x))) \cdot H^\top\right]^{-1} \cdot \nabla_y f(x, y^*(x)), H \in \partial\mathcal{P}_\mathcal{Y}(z^*)\}. \tag{4}$$

Stationary Point

- Our next step is to design an algorithm to find the point x satisfying the condition

$$\min\{\|h\| : h \in \partial F(x)\} \leq \epsilon. \tag{5}$$

However, finding an ϵ stationary point in nonsmooth nonconvex optimization can not be achieved by any finite-time algorithm given a fixed tolerance $\epsilon \in [0, 1)$.

- Define the δ-approximation generalized Jacobian: $\partial_\delta\mathcal{P}_\mathcal{Y}(z) := co\left(\bigcup_{z'\in\mathbb{B}_\delta(z)} \partial\mathcal{P}_\mathcal{Y}(z')\right)$. We can obtain the following approximation,

$$\bar{\partial}_\delta F(x) = \{h | h = \nabla_x f(x, y^*(x)) - \eta\nabla^2_{xy} g(x, y^*(x)) H^\top \cdot \left[I_{d_2} - (I_{d_2} - \eta\nabla^2_{yy} g(x, y^*(x))) \cdot H^\top\right]^{-1} \cdot \nabla_y f(x, y^*(x)), H \in \partial_\delta\mathcal{P}_\mathcal{Y}(z^*)\} \tag{6}$$

- Equipping with this approximation, one could find a point that is close to an ϵ-stationary point, i.e., (δ, ϵ)-stationary point:

$$\min\left\{\|h\| : h \in \bar{\partial}_\delta F(x)\right\} \leq \epsilon \tag{7}$$

If we can find a point x' at most distance δ away from x such that x' is ϵ-stationary, then we know x is (δ, ϵ)-stationary. However, the contrary is not true.

Randomized Smooth

- Given a non-expansive projection operator $\mathcal{P}_\mathcal{Y}(z)$ and uniform distribution $\mathbb{P}$ on a unit ball in ℓ_2-norm, we define the smoothing function as $\mathcal{P}_{\mathcal{Y}\delta}(z) = \mathbb{E}_{u\sim\mathbb{P}}[\mathcal{P}_\mathcal{Y}(z + \delta u)]$.
- Using this randomized smoothing function to replace the approximation generalized Jacobian in Eqn (6), we can approximate the hypergradient as follows,

$$\nabla F_\delta(x) = \nabla_x f(x, y^*(x)) - \eta\nabla^2_{xy} g(x, y^*(x))\nabla\mathcal{P}_{\mathcal{Y}\delta}(z^*)^\top \cdot \left[I_{d_2} - (I_{d_2} - \eta\nabla^2_{yy} g(x, y^*(x)))\nabla\mathcal{P}_{\mathcal{Y}\delta}(z^*)^\top\right]^{-1} \nabla_y f(x, y^*(x)).$$

- Under Assumptions on f and g, we have $\nabla F_\delta(x)$ is Lipschitz continuous w.r.t x.
- We have $\nabla\mathcal{P}_{\mathcal{Y}\delta}(z) \in \partial_\delta\mathcal{P}_\mathcal{Y}(z)$ for any $z \in \mathbb{R}^{d_2}$. Once we find a point satisfying the condition $\|\nabla F_\delta(x)\| \leq \epsilon$, then it is a (δ, ϵ)-stationary point.

Approximation of Hypergradient

- Since obtaining the optimal solution $y^*(x)$ is usually time-consuming, one proper method is to replace $y^*(x)$ with y.
- We can use the following unbiased estimator of the gradient $\nabla\mathcal{P}_{\mathcal{Y}\delta}(z)$ as a replacement,

$$\tilde{H}(z; u) = \sum_{i=1}^{d_2} \frac{1}{2\delta}\left(\mathcal{P}_\mathcal{Y}(z + \delta u_i) - \mathcal{P}_\mathcal{Y}(z - \delta u_i)\right) u_i^\top \tag{8}$$

- We can use the Neumann series to approximate the matrix inverse and obtain the following hypergradient approximation

$$\bar{\nabla} f_\delta(x, y; \bar{\xi}) = \nabla_x f(x, y) - \eta Q\nabla^2_{xy} g(x, y)\tilde{H}(z; u^0)^\top \prod_{i=1}^{c(Q)} \left((I_{d_2} - \eta\nabla^2_{yy} g(x, y))\tilde{H}(z; u^i)^\top\right) \nabla_y f(x, y) \tag{9}$$

where $\bar{\xi} := \{u^0, \cdots, u^{c(Q)}\}$, and $c(Q) \sim \mathcal{U}\{0, \cdots, Q - 1\}$.

Double-Momentum Method for Lower-level Constrained Bilevel Optimization

- Lower-level variable update rules:

$$\hat{y}_{k+1} = \mathcal{P}_\mathcal{Y}\left(y_k - \frac{\tau}{\mathcal{P}_{[1/c_u, 1/c_l]}(\sqrt{m_{1,k}} + G_0)} v_k\right),$$
$$y_{k+1} = (1 - \eta_k) y_k + \eta_k \hat{y}_{k+1},$$
$$v_{k+1} = (1 - \beta) v_k + \beta\nabla_y g(x_k, y_k),$$

where $\eta_k > 0$, $\tau > 0$ and $v_1 = \nabla_y g(x_1, y_1)$.

- Upper-level variable update rules:

$$x_{k+1} = x_k - \frac{\eta_k\gamma}{\mathcal{P}_{[1/c_u, 1/c_l]}(\sqrt{m_{2,k}} + G_0)} w_k,$$
$$w_{k+1} = (1 - \alpha) w_k + \alpha\bar{\nabla} f_\delta(x_k, y_k; \bar{\xi}_k),$$

where $w_1 = \bar{\nabla} f_\delta(x_1, y_1; \bar{\xi}_1)$.

Algorithm

Algorithm 1 DMLCBO

Input: Initialize $x_1 \in \mathcal{X}$, $y_1 \in \mathcal{Y}$, $v_1 = \nabla_y g(x_1, y_1)$, $w_1 = \bar{\nabla} f_\delta(x_1, y_1; \bar{\xi}_1)$, $\eta_k, \tau, \gamma, \beta, \alpha, Q$ and η.

for $k = 1, \cdots, K$ **do**
- Update $x_{k+1} = x_k - \frac{\eta_k\gamma}{\mathcal{P}_{[1/c_u, 1/c_l]}(\sqrt{m_{2,k}} + G_0)} w_k$.
- Update $y_{k+1} = (1 - \eta_k) y_k + \eta_k\mathcal{P}_\mathcal{Y}(y_k - \frac{\tau}{\mathcal{P}_{[1/c_u, 1/c_l]}(\sqrt{m_{1,k}} + G_0)} v_k)$
- Calculate the hyper-gradient $\bar{\nabla} f_\delta(x_{k+1}, y_{k+1}; \bar{\xi}_{k+1})$ according to Eqn. (9).
- Update $w_{k+1} = (1 - \alpha) w_k + \alpha\bar{\nabla} f_\delta(x_{k+1}, y_{k+1}; \bar{\xi}_{k+1})$.
- Update $v_{k+1} = (1 - \beta) v_k + \beta\nabla_y g(x_{k+1}, y_{k+1})$.

end for

Output: x_r where $r \in \{1, \cdots, K\}$ is uniformly sampled.

Theorem

Under Assumptions, with $\frac{1}{\mu_g}(1 - \frac{1}{4(2\pi)^{1/4}\sqrt{d_2}L_p}) \leq \eta < \frac{1}{\mu_g}$, $Q = \frac{1}{\mu_g\eta}\ln\frac{C_{gxy}C_{fy}K}{\mu_g}$, $0 \leq a \leq 2$, $\alpha = c_1\eta_k$, $\beta = c_2\eta_k$, $L_0 = \max(L_1(\frac{d_2}{\delta}), L_2(\frac{d_2}{\delta})) > 1$, $\Phi_1 = \mathbb{E}[F_\delta(x_1) + \frac{10L_0^a c_l}{\tau\mu_g c_u}\|y_1 - y^*(x_1)\|^2 + c_l(\|w_1 - \bar{\nabla} f_\delta(x_1, y_1) - R_1\|^2 + \|\nabla_y g(x_1, y_1) - v_1\|^2)]$, *and* $\eta_k = \frac{t}{(m+k)^{1/2}}$, $t > 0$, *we have*

$$\min\{\|h\| : h \in \bar{\partial}_\delta F(x_r)\} \leq \frac{4m^{1/4}\sqrt{G}}{\sqrt{Kt}} + \frac{4\sqrt{G}}{(Kt)^{1/4}}.$$

where $G = \frac{\Phi_1 - \Phi^*}{\gamma c_l} + \frac{17t}{4K^2}(m + K)^{1/2} + \frac{4}{3tK^2}(m + K)^{3/2} + (m\sigma_f(d_2))t^2\ln(m + K)$; *the range of* c_1,c_2 γ, τ *and* m *are given in the appendix.*

References

[1] Frank H Clarke.
Optimization and nonsmooth analysis.
SIAM, 1990.

Correspondence to: jsgubin@gmail.com.