AMSI VACATION RESEARCH SCHOLARSHIPS 2022–23

SET YOUR SIGHTS ON RESEARCH THIS SUMMER

Design and Analysis of Bilevel Optimisation Algorithms

Tran Khanh Hung Giang

Supervised by Dr. Nam Ho-Nguyen

The University of Sydney

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Abstract

In this paper, we study a class of simple bilevel optimisation problem whose goal is to minimise a function over the set of minimisers of another convex function over a compact and convex feasible region. Although some methods have been developed to tackle this problem, they rely on projections onto the feasible region which, for some applications is computationally expensive or intractable. In addition, adapting work of Braun et al, we provide an analysis using a relaxed linear optimisation oracle, which can ease the computational burden further.

1 Introduction

1.1 Problem Description

Simple bilevel optimisation aims to minimise a function subject to a solution set of minimizing problem. Precisely, the set up of the problem is as follows:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f_{\text{upper}}(x) \\ \text{s.t.} \quad & x \in \arg\min_{z\in S} f_{\text{lower}}(z), \end{aligned} \tag{1}$$

where S is a compact and convex set and $f_{\text{lower}}, f_{\text{upper}} : \mathbb{R}^n \mapsto \mathbb{R}$ are continuously differentiable functions on S. In this paper, the f_{lower} is assumed to be convex but strong convexity is not a must so that the set of minimisers of lower level optimisation problem may be different from a singleton.

Practically, simple bilevel optimisation plays a critical role in various applications such as hyper-parameter optimization, meta-learning, deep reinforcement learning as mentioned by Liu et al. [12]. Furthermore, as the big data is widely recognised and utilised, spending high computational resources for training statistical models is inevitable. Therefore, a cost-effective algorithm for this class of optimisation problem should be developed.

1.2 Related Literature

Recently, Jiang et al. [9] developed an algorithm based on the Frank-Wolf algorithm or conditional gradient (CG) method [4]. The key idea of Jiang et al is to approximate the solution set of the lower level problem by adopting a cutting plan approach. However, their convergence analysis failed to establish a result that can guarantee either the sequence or any subsequence of optimality gaps would converge to zero for both f_{lower} and f_{upper}. In this case, it is possible that the sequence generated by the algorithm may converge to point which does not belong to the exact solution of the simple bilevel optimisation problem.

Furthermore, even when the convergence issue is ignored, the most expensive step of the well-known conditional gradient method [4] as well as CG-BiO method [9] is the linear optimisation problem which may be expensive to compute if the feasible region is complex. To further relieve the computational burden, Braun et al. [3] recommended a method called lazifying conditional gradient for single-level convex optimisation problems.

1.3 Contribution and Outline

As indicated above, we establish the convergence rate for both upper-level and lower-level problems and decrease the computational expense from linear oracle. To be more specific, three primary goals of this paper are:

1. Firstly, we propose the adaptive conditional gradient-based bilevel optimisation (ACG-BiO) method based on the study of Jiang et al. [9]. to establish the convergence rate of $O(\frac{1}{K})$, where K is number of iterations, for both upper-level and lower-level problem under the assumptions of convexity and Lipschitz differentiability for both upper-level and lower-level objective functions. Additionally, under Holderian error bound Assumption 2, a more reliable convergence result can be achieved for the upper level.

2. Secondly, we introduce a relaxed version of ACG-BiO method, which is the relaxed adaptive conditional gradient-based bilevel optimisation (RACG-BiO) method, by integrating the weak separation oracle (LPsep) devised by Braun et al. [3] to replace the linear optimisation step. The key point of this oracle is that data from past iterations can be reused without calling the linear optimisation. Furthermore, we also prove that under the same assumptions as above, the same convergence rate for upper-level and lower-level problems can be established.

3. Thirdly, we propose a modification for ACG-BiO method, which is the unbounded adaptive conditional gradient-based bilevel optimisation (UACG-BiO) method, to relax the assumption of boundedness of the feasible region S. Moreover, we prove that such adjustment can bring about a convergence rate of $O(1/K^{1-p})$ for any $p \in (0, 1)$.

To convey those ideas, we will firstly go through some relevant concepts and properties in Section 2. In the same section, some discussion about the foundation of this study, the conditional gradient (CG) method [4] for solving single-level optimisation problem as well as the short-coming of the conditional gradient-based bilevel optimization (CG-BiO) method [9] and the cost-effectiveness of the weak separation oracle [3] will be discussed together with a motivating example of simple bilevel optimisation from regression. In Section 3, the ACG-BiO method and its convergence analysis will be presented. Afterwards, the relaxed version of ACG-BiO method and relevant convergence results will be shown in Section 4. Subsequently, the more generalized version of ACG-BiO to allow unboundedness of feasible region is analysed in Section 5. Finally, the efficiency of the proposed algorithm will be compared with CG-BiO method [9] via a numerical experiment in Section 6.

2 Preliminaries

2.1 Assumptions and Definitions

Definition 1. A set $S \subseteq \mathbb{R}^n$ is called convex if for every $a, b \in S$, we also have $ta + (1-t)b \in S$, for every $0 \leq t \leq 1$.

Definition 2. Given that S is a convex subset of $\mathbb{R}^n$, function $f : S \to \mathbb{R}$ is convex if and only if

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y), \forall x, y \in S, \forall \lambda \in [0,1]$$

Proposition 2.1 (Wright and Recht [16]). *Given that S is a convex subset of $\mathbb{R}^n$ and $f : S \to \mathbb{R}$ is a differentiable function, then f is convex if and only if*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \forall x, y \in S$$

Definition 3. Let $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^n$ and $\|\cdot\|_*$ be its dual norm. A function is Lipschitz over some set S if and only if

$$\|f(x) - f(y)\|_* \leq L\|x - y\|, \forall x, y \in S.$$

Proposition 2.2 (Wright and Recht [16]). *If a function $f : S \to \mathbb{R}$ has Lipschitz gradient over S, then*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2, \forall x, y \in S$$

The assumptions illustrated below will be used for convergence analysis of ACG-BiO method and the relaxed version of ACG-BiO method in sections (3), (4).

Assumption 1. Let $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^n$ and $\|\cdot\|_*$ be its dual norm. We assume

1. $S \subset \mathbb{R}^n$ is convex and compact with diameter D, i.e, $\|x - y\| \leq D, \forall x, y \in S$.
2. f_{lower} is convex, continuously differentiable on S, and its gradient is Lipschitz with constant L_{lower}.
3. f_{upper} is convex, continuously differentiable on S, and its gradient is Lipschitz with constant L_{upper}.

Definition 4.

1. $\chi_1^* := \arg\min_{x \in S} f_{\text{lower}}(x)$
2. $\chi_2^* := \arg\min\{f_{\text{upper}}(x) | x \in \chi_1^*\}$
3. $f_{\text{lower}}^* := \min_{x \in S} f_{\text{lower}}(x)$
4. $f_{\text{upper}}^* := \min\{f_{\text{upper}}(x) | x \in \chi_1^*\}$

Lemma 2.3. *Under Assumption 1, χ_1^* is non-empty, convex and compact. Therefore, χ_2^* is also non-empty, convex and compact.*

Proof. Under Assumption 1, since S is compact and f_{lower} is continuous over S, f_{lower} should have a minimum. Therefore, χ_1^* is nonempty. On the other hand, since χ_1^* is a subset of S and $\chi_1^* = f_{\text{lower}}^{-1}(f_{\text{lower}}^*)$, which is the pre-image of a closed set, it is bounded and relatively closed in S and in fact, it is compact since S is also closed. Furthermore, for any $v_1, v_2 \in \chi_1^*$, $f_{\text{lower}}^* \leq f_{\text{lower}}(\lambda v_1 + (1-\lambda)v_2) \leq \lambda f_{\text{lower}}(v_1) + (1-\lambda) f_{\text{lower}}(v_2) = f_{\text{lower}}^*, \forall \lambda \in [0,1]$. Therefore, χ_1^* is convex and compact. Following similar reasoning, χ_2^* is also non-empty, convex and compact. □

Although Assumption 1 is sufficient for us to establish some convergence guarantees for both lower and upper objective functions, it will be shown later that $f_{\text{upper}}(x_k) - f^*_{\text{upper}}$, where x_k is an output generated by ACG-BiO method, may be negative, which means we may have superoptimal solution. Under that case, we need some other assumptions to come up with a stronger convergence result for upper level objective function and Holderian error bound turns out to be the solution.

Assumption 2. The function f_{lower} satisfies Holderian error bound for some $\alpha > 0$ and $r \geq 1$, i.e,

$$\frac{\alpha}{r}\left(\inf_{x' \in \chi_1^*} \|x' - x\|\right)^r \leq f_{\text{lower}}(x) - f^*_{\text{lower}}.$$

In fact, it is known that this condition holds generally when the function f_{lower} is analytic and the feasible region S is bounded [13] or when f_{lower} is a piecewise convex polynomial and S is a polyhedron [11].

Theorem 2.4. *Let $\{a_n\}_n$ and $\{b_n\}_n$ be two sequences of real numbers. Assume that $\{b_n\}_n$ is a strictly monotone and divergent sequence (i.e. strictly increasing and approaching ∞ , or strictly decreasing and approaching $-\infty$) and the following limit exists:*

$$\lim_{n\to\infty} \frac{a_{n+1} - a_n}{b_{n+1} - b_n} = l$$

then the limit:

$$\lim_{n\to\infty} \frac{a_n}{b_n} = l.$$

2.2 Conditional Gradient Method

The conditional gradient (CG) method [4] can be used to solve the following problem:

$$\min_{x \in S} f(x)$$

under the following assumptions:

- S is convex, compact with diameter D.
- f is convex, continously differentiable over S.
- ∇f is Lipschitz with constant L.

Different from any projection-based method such as projected gradient descent or its accelerated version fast iterative shrinkage-thresholding algorithm [2], CG method [4] does not require the access to the projection oracle onto S. Instead, it assumes that we can access to any linear minimisation oracle over S, which is generally less expensive than projection. In fact, if S is a polyhedra then such oracle is reduced to a linear programming problem. Turning to the method itself, the specific steps are shown in Algorithm 1.

In terms of convergence guarantee, it can be showed that CG method [4] can obtain a convergence rate of $O(1/K)$. Such claim is discussed in Theorem 2.5.

Algorithm 1: [Frank and Wolfe [4]] Conditional gradient method for single-level optimisation problem (CG-BiO) .

Data: stepsizes $\{\alpha_k\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in S$ **for** $k = 0, 1, \ldots, K$ **do**

  Compute $s_k \leftarrow \arg\min_{s \in S} \langle \nabla f(x_k), s \rangle$

  Compute $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$

Theorem 2.5 (Frank and Wolfe [4]). *Under stepsizes $\alpha_k = \frac{2}{k+2}, \forall k \in \mathbb{N}$, let $\{x_k\}_{k=0,1,\ldots,K}$ be the sequence generated by Algorithm 1, we have that*

$$0 \leq f(x_K) - f^* \leq \frac{2LD^2}{K+2},$$

where f^* is the minimum of f over S.

2.3 Conditional Gradient-based Bilevel Optimization Method

Algorithm 2: [Jiang et al. [9]] Conditional gradient-based bilevel optimization (CG-BiO) .

Data: stepsizes $\{\alpha_k\}_k$, target accuaracies $\epsilon_1, \epsilon_2 > 0$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in S$ such that $0 \leq f_{\text{lower}}(x_0) - f^*_{\text{lower}} \leq \frac{\epsilon_1}{2}$;

for $k = 0, 1, \ldots, K$ **do**

  Compute $s_k \leftarrow \arg\min_{s \in \chi_k} \langle \nabla f_{\text{upper}}(x_k), s \rangle$ where

  $\chi_k := \{s \in S \mid \langle \nabla f_{\text{lower}}(x_k), s - x_k \rangle \leq f_{\text{lower}}(x_0) - f_{\text{lower}}(x_k)\}$;

if $\langle \nabla f_{\text{upper}}(x_k), x_k - s_k \rangle < \epsilon_2$ *and* $\langle \nabla f_{\text{lower}}(x_k), x_k - s_k \rangle < \frac{\epsilon_1}{2}$ **then**

  return x_k *and* ***STOP***;

else

  Compute $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$;

Before examining the convergence results of Algorithm 2, it is not obvious that the sequence of regions $\{\chi_k\}_k$ is non-empty at each iteration. Therefore, the validity of such sub linear minimisation problem can be ensured by Theorem 2.6 proven in Jiang et al. [9].

Lemma 2.6 (Jiang et al. [9]). *For any $k \in \mathbb{N}$, we have $\chi_1^* \subseteq \chi_k$.*

Along with Algorithm 2, Jiang et al. [9] came up with the following convergence guarantees under Assumption 1 for the sequence generated by the algorithm.

Theorem 2.7 (Jiang et al. [9]). *Suppose that Assumption 1 holds, under stepsizes* $\alpha_k = \frac{2}{k+2}, \forall k \in \mathbb{N}$, *let* $\{x_k\}_{k=1,\dots K}$, *which is the sequence generated by Algorithm 2, we have that*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \frac{2L_{\text{upper}}D^2}{K+2}$$
$$f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq \frac{2L_{\text{lower}}D^2}{K+2} + \frac{1}{2}\epsilon_1$$

From Theorem 2.7, when K approaches infinity, the optimality gap of f_{lower} does not neccessarily converge to 0 and therefore, it is possible that the sequence does not contain any accumulation point that is exact solution of the simplie bilevel optimisation problem (1). One cause for this undesirable property is the approximation χ_k for χ^*_1 proposed by Jiang et al. [9] involves $f_{\text{lower}}(x_0)$. This allows the positive gap ϵ_1 to appear in the optimality gap of f_{lower} in the convergence result.

2.4 Weak Separation Oracle

Algorithm 3: [Braun et al. [3]] Weak separation oracle - $\text{LPsep}_S(c, x, \Phi, K)$.

Data: linear objective $c \in \mathbb{R}^n$, point $x \in S$, accuracy $K \geq 1$, objective value $\Phi > 0$

Result: Either vertex $y \in S$ with $\langle c, x - y\rangle > \frac{\Phi}{K}$, or **false** : $\langle c, x - z\rangle \leq \Phi$ for all $z \in S$

```
if there exists y ∈ S cached with ⟨c, x − y⟩ > Φ/K then
    return y;
else
    y ← arg min{⟨c, z⟩ | z ∈ S} (add to cache);
    if ⟨c, x − y⟩ > Φ/K then
        return y;
    else
        return false
```

As claimed by Braun et al. [3], Algorithm 3 is much weaker than the approximate minimisation mentioned in the study done by Jaggi [8]. Precisely, the primary idea of this relaxation is the allowance of storing previous solutions and reusing them rather than calling for linear minimisation problem every iterations. In the worst case, i.e when none of linear gap values evaluated at current iteration's data and the previous solutions does not have sufficient improvement as defined by Φ, Algorithm 3, only has to call the linear minimisation in the same manner as the standard conditional gradient method [4] does.

2.5 Motivating Example - Over-parameterized Regression

An example involved solving simple bi-level optimisation problem is over-parameterized regression. Rather than unconstrainedly minimising the training loss function $L_{\text{train}}(\beta)$, which depends on the training data set D_{train}, this type of regression restricts the coefficient parameter β over some conditions, which are represented

by some set S. In this paper, we adopt an example where S is $\{\beta \in \mathbb{R}^d | \|\beta\|_1 \leq \lambda\}$ for some $\lambda > 0$. In general, if the covariate matrix X_{train} fails to have its columns linearly independent, then we expect there are multiple solutions for the over-parameterized problem. Despite having the same training loss, those solutions do potentially bring different outcomes on the loss $L_{\text{valid}}(\beta)$ of validation data set. Hence, it is natural for one to consider minimising another objective function such as the loss over validation data set D_{valid}. As it is the case, we have the following bilevel optimisation problem.

$$\begin{aligned} \min_{\beta} \quad & L_{\text{valid}}(\beta) \\ \text{s.t.} \quad & \beta \in \arg\min_{\xi \in S} L_{\text{train}}(\xi). \end{aligned} \tag{2}$$

When the loss function is chosen to be convex and L-smooth, both the upper-level and lower-level objectives are smooth and convex. Such situation is considered in a subproblem in hyperparameter selection problems proposed by Gao et al. [5]

3 Adaptive Conditional Gradient-based Bilevel Optimisation Method

3.1 Proposed Method

Despite such short coming in convergence guarantees as discussed in Section 2.3, the idea of approximating implicit feasible region by the cutting plane still plays critical role in our proposed algorithm. Indeed, rather than using $f_{\text{lower}}(x_0)$ in the right hand side of the equation of the plane, which is fixed, we need something more dynamic and adaptive to induce the optimality gap of f_{lower} converge to 0 as K goes to infinity. Hence, such goal can be achieve by introduing a sequence $\{\beta_k\}_k$ satisfying some properties, which will be discussed in Theorem 3.7.

Algorithm 4: Adaptive conditional gradient-based bilevel optimisation method - ACG-BiO.

Data: stepsizes $\{\alpha_k\}_k$, supports $\{\beta_k\}_k \in \mathbb{R}^n$, residuals $\{\gamma_k\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in S$;

for $k = 0, 1, \ldots, K$ **do**

  Compute s_k such that

$$\langle \nabla f_{\text{upper}}(x_k), s_k \rangle \leq \min_{s \in \chi_{1,k}} \langle \nabla f_{\text{upper}}(x_k), s \rangle + \gamma_k$$

  where $\chi_{1,k} := \{s \in S \mid \langle \nabla f_{\text{lower}}(x_k), s - x_k \rangle \leq \beta_k - f_{\text{lower}}(x_k)\}$;

  Compute $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$;

Note that in the linear oracle step of Algorithm 4, only one option should be done throughout the process and the second option can be considered as an approximation of the first option.

Before dicussing the convergence analysis of Algorithm 4, we will guarantee the appropriateness of the linear minimisation step by proving that the sequence $\{\chi_{1,k}\}_k$ is non-empty and in fact, convex and compact under some restriction on supports.

Lemma 3.1. *If the supports $\{\beta_k\}_k$ satisfy*

$$0 \leq \beta_k - f^*_{\text{lower}}, \forall k \in \mathbb{N},$$

we have $\chi_1^* \subseteq \chi_{1,k}$, for all $k \in \mathbb{N}$ in Algorithm 4. Consequently, $\chi_{1,k}$ is nonempty, compact, and convex for all $k \in \mathbb{N}$.

Proof. We have that for any $x_1^* \in \chi_1^*$, we have that $\langle \nabla f_{\text{lower}}(x_k), x_1^* - x_k \rangle \leq f_{\text{lower}}(x_1^*) - f_{\text{lower}}(x_k) \leq \beta_k - f_{\text{lower}}(x_k)$. Therefore, $x_1^* \in \chi_{1,k}$, which implies $\chi_1^* \subseteq \chi_{1,k}$. Eventually, the compactness and convexity of the sequence can be justified by seeing that the closed half-planes are closed, convex and S is compact and convex as well.

□

3.2 Convergence Analysis

Lemma 3.2. *Suppose that Assumption (1) holds, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0, \forall k \in \mathbb{N}$, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by the algorithm (4) we have that*

$$0 \leq f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{LD^2 i}{i+1} \right).$$

Proof. We have that $f_{\text{lower}}(x_{K+1}) \leq f_{\text{lower}}(x_K) + \langle \nabla f_{\text{lower}}(x_K), x_{K+1} - x_K \rangle + \frac{L_{\text{lower}}}{2} ||x_{K+1} - x_K||^2$ and we also obtain that $\langle \nabla f_{\text{lower}}(x_K), x_{K+1} - x_K \rangle = \alpha_K \langle \nabla f_{\text{lower}}(x_K), s_K - x_K \rangle \leq \alpha_K (\beta_K - f_{\text{lower}}(x_K))$. Hence, we have

$$\begin{aligned}
f_{\text{lower}}(x_{K+1}) &\leq f_{\text{lower}}(x_K) + \alpha_K \langle \nabla f_{\text{lower}}(x_K), s_K - x_K \rangle + \frac{L_{\text{lower}}}{2} \alpha_K^2 \|s_K - x_K\|^2 \\
\Rightarrow f_{\text{lower}}(x_{K+1}) &\leq f_{\text{lower}}(x_K) + \alpha_K (\beta_K - f_{\text{lower}}(x_K)) + \frac{L_{\text{lower}} D^2}{2} \alpha_K^2 \\
\Rightarrow f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}} &\leq (1 - \alpha_K)(f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + \alpha_K (\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2}{2} \alpha_K^2 \\
\Rightarrow f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}} &\leq \frac{K}{K+2} (f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + \frac{2}{K+2} (\beta_K - f^*_{\text{lower}}) + \frac{2 L_{\text{lower}} D^2}{(K+2)^2} \\
\Rightarrow (K+2)(K+1)[f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}}] &\leq (K+1)K (f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + 2(K+1) \left(\beta_K - f^*_{\text{lower}} + \frac{L_{\text{lower}} D^2}{K+2} \right) \\
\Rightarrow (K+1)K[f_{\text{lower}}(x_K) - f^*_{\text{lower}}] &\leq 2 \sum_{i=1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2 i}{i+1} \right) \\
\iff f_{\text{lower}}(x_K) - f^*_{\text{lower}} &\leq \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2 i}{i+1} \right).
\end{aligned}$$

□

Corollary 3.3. *Suppose that Assumption (1) holds, under stepsizes* $\alpha_k = \frac{2}{k+2}$, *supports* $\{\beta_k\}_k$ *such that* $\beta_k \geq f^*_{\text{lower}}$, *and residuals* $\{\gamma_k\}$ *such that* $\gamma_k \geq 0, \forall k \in \mathbb{N}$, *let* $\{x_k\}_{k=1,\dots K}$ *be the sequence generated by the algorithm (4), if* $\beta_k \to f^*_{\text{lower}}$ *then* $f_{\text{lower}}(x_K) \to f^*_{\text{lower}}$.

Proof. By applying Theorem 2.4 for the right hand side of the last inequality, we have that if $\beta_k \to f^*_{\text{lower}}$ then

$$\begin{aligned}
&\lim_{K\to\infty} \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2 i}{i+1} \right) \\
&= \lim_{K\to\infty} \frac{2}{(K+2)(K+1) - (K+1)K} \left[(K+1)(\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2 (K+1)}{K+2} \right] \\
&= \lim_{K\to\infty} \left[(\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}} D^2}{K+2} \right] = 0
\end{aligned}$$

By squeeze theorem, we have that

$$\lim_{K\to\infty} (f_{\text{lower}}(x_K) - f^*_{\text{lower}}) = 0.$$

□

Proposition 3.4. *Suppose that Assumption (1) holds, under stepsizes* $\alpha_k = \frac{2}{k+2}$, *supports* $\{\beta_k\}_k$ *such that* $\beta_k \geq f^*_{\text{lower}}$, *and residuals* $\{\gamma_k\}$ *such that* $\gamma_k \geq 0, \forall k \in \mathbb{N}$, *let* $\{x_k\}_{k=1,\dots K}$ *be the sequence generated by the algorithm (4) we have that*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i\gamma_{i-1} + \frac{L_{\text{upper}} D^2 i}{i+1} \right).$$

Proof. We have that

$$\begin{aligned}
\langle \nabla f_{\text{upper}}(x_k), x_{k+1} - x_k \rangle =& \alpha_k \langle \nabla f_{\text{upper}}(x_k), s_k - x_k \rangle \\
\leq& \alpha_k \left(\langle \nabla f_{\text{upper}}(x_k), x_2^* - x_k \rangle + \gamma_k \right) \\
\leq& \alpha_k (f^*_{\text{upper}} - f_{\text{upper}}(x_k) + \gamma_k),
\end{aligned}$$

for any $x_2^* \in \chi_2^*$. In addition, we have that

$$\begin{aligned}
f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \langle \nabla f_{\text{upper}}(x_K), x_{K+1} - x_K \rangle + \frac{L_{\text{upper}}}{2} ||x_{K+1} - x_K||^2 \\
\Rightarrow f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \alpha_K (f^*_{\text{upper}} - f_{\text{upper}}(x_K) + \gamma_K) + \frac{L_{\text{upper}} D^2}{2} \alpha_K^2 \\
\Rightarrow f_{\text{upper}}(x_{K+1}) - f^*_{\text{upper}} &\leq \frac{K}{K+2} (f_{\text{upper}}(x_K) - f^*_{\text{upper}}) + \frac{2}{K+2} \gamma_K + \frac{2 L_{\text{upper}} D^2}{(K+2)^2}
\end{aligned}$$

Similar to the proof of proposition (3.2), we have that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i\gamma_{i-1} + \frac{L_{\text{upper}} D^2 i}{i+1} \right).$$

□

Since $|\|\nabla f_{\text{upper}}(x)\|_* - \|\nabla f_{\text{upper}}(y)\|_*| \leq \|\nabla f(x) - \nabla f(y)\|_* \leq L_{\text{upper}} \|x - y\|$, the function $\|\nabla f_{\text{upper}}(x)\|_*$ is continuous over S and since S is compact, there exists a maximum of this function over S.

Proposition 3.5. *Suppose that Assumption (1) holds, under stepsizes $\alpha_k \in (0,1)$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0, \forall k \in \mathbb{N}$, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by the algorithm (4), we have that if Assumption (2) also holds, then*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -G\left(\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right)^{\frac{1}{r}}, \forall K \in \mathbb{N},$$

where $G := \max_{x \in S} \|\nabla f_{\text{upper}}(x)\|_*$.

Proof. Under assumption (2), we have that

$$\frac{\alpha}{r}\|x_K - v_K\|^r \leq f_{\text{lower}}(x_K) - f^*_{\text{lower}} \iff \|x_K - v_K\| \leq \left(\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right)^{\frac{1}{r}}$$

where $v_k \in \arg\min_{x' \in \chi_1^*} \|x' - x_K\|$. Thus, we have that

$$\begin{aligned} f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq f_{\text{upper}}(x_K) - f_{\text{upper}}(v_K) &\geq \langle \nabla f_{\text{upper}}(v_K), x_K - v_K\rangle \\ &\geq -\|f_{\text{upper}}(v_K)\|_*\|x_K - v_K\| \\ &\geq -G\left(\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right)^{\frac{1}{r}}. \end{aligned}$$

□

Corollary 3.6. *Suppose that Assumption (1) and Assumption (2) hold, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, $\beta_k \to f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0, \gamma_k \to 0, \forall k \in \mathbb{N}$, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by the algorithm (4), we have that*

$$f_{\text{upper}}(x_K) \to f^*_{\text{upper}}.$$

Theorem 3.7. *Suppose that Assumption (1) holds, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports β_k such that $0 \leq \beta_k - f^*_{\text{lower}} \leq O\left(\frac{1}{k}\right), 0 \leq \gamma_k \leq O\left(\frac{1}{K}\right), \forall k \in \mathbb{N}$, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by the algorithm (4) we have that*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq O\left(\frac{1}{K}\right),$$
$$f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq O\left(\frac{1}{K}\right),$$

and thus, $\{x_k\}_k$ admits a subsequence that converges to a point in χ_2^*. In fact, any accumulation point of $\{x_k\}_k$ is a solution of problem (1). In addition, under assumption (2), we have

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -O\left[\left(\frac{1}{K}\right)^{\frac{1}{r}}\right], \forall K \in \mathbb{N},$$

and thus, $f_{\text{lower}}(x_K) - f^*_{\text{lower}}, f_{\text{upper}}(x_K) - f^*_{\text{upper}} \to 0$ as $K \to \infty$.

Remark 3.1. In theorem (3.7), the sequence $\{\beta_k\}_k$ satisfying the stated conditions can be generated by running the conditional gradient method [4] for lower-level problem and obtain the sequence $\{y_k\}_k$ such that

$$0 \leq f_{\text{lower}}(y_k) - f^*_{\text{lower}} \leq \frac{2L_{\text{lower}}D^2}{t_k + 2} \leq O\left(\frac{1}{k}\right),$$

where t_k is the smallest integer such that the right most inequality is true. Then, let $\beta_k := f_{\text{lower}}(y_k), \forall k \in \mathbb{N}$. Similarly, the sequence $\{\gamma_k\}_k$ satisfying properties in Theorem 3.7 can be generated in the same way.

4 Application of LPsep to ACG-BiO Method

4.1 Proposed Method

Turning to the relaxed version of the adaptive conditional gradient-based bilevel optimisation method (5), the RACG-BiO method, we adopt the same recursive rule for $\{\Phi_k\}_k$ as Braun et al. [3], which is described in Algorithm 5. Noticably, there is no circumstances in which x_k is assigned to x_{k+1} as in the lazy conditional gradient algorithm due to the minor change discussed in the above paragraph. Indeed, such inconvenience also comes from the fact although we can make the reasoning $f_{\text{upper}}(x_{K+1}) - f^*_{\text{upper}} = f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \langle f_{\text{upper}}(x_K), x_K - x^* \rangle \leq \Phi_K$ as Braun et al. [3] do in the negative call case as we can prove $x^* \in \chi_{1,k}$ in the next theorem, we cannot make the same estimate for f_{lower}. As it is the case, the update rule $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$ is a must for both positive and negative call to establish that both $f_{\text{lower}}(x_k) - f^*_{\text{lower}}$ and $f_{\text{upper}}(x_k) - f^*_{\text{upper}}$ must not strictly greater than Φ_{k-1} for all positive integer k as in Theorem 4.2.

Algorithm 5: Relaxed adaptive conditional gradient-based bilevel optimisation method.

Data: stepsizes $\{\alpha_k\}_k$, supports $\{\beta_k\}_k$, initial optimality gap $\Phi_0 > 0$, target accuaracies $\epsilon_1, \epsilon_2 > 0$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in S$;

for $k = 0, 1, \ldots, K$ **do**

  Compute $s_k \leftarrow \text{LPsep}_{\chi_{1,k}}(\nabla f_{\text{upper}}(x_k), x_k, \Phi_k, 1)$, where

  $\chi_{1,k} := \{s \in S \mid \langle \nabla f_{\text{lower}}(x_k), s - x_k \rangle \leq \beta_k - f_{\text{lower}}(x_k)\}$;

  Compute $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$;

  Compute $\Phi_{k+1} \leftarrow \frac{\Phi_k + \frac{LD^2}{2}\alpha_{k+1}^2}{1+\alpha_{k+1}}$, where $L := \max\{L_{\text{lower}}, L_{\text{upper}}\}$;

Before jumping to the convergence analysis of Algorithm 5, Theorem 4.1 illustrates some properties of sequence $\{\Phi_k\}_k$ defined in Algorithm 5 as these properties will play central roles in proving important results summarised in Theorem 4.2.

Proposition 4.1. *When $\alpha_k = \frac{2}{k+2}, \forall k \in \mathbb{N}$, the sequence $\{\Phi_k\}_k$ defined in the algorithm (5) satisfies the following inequalities*

$$\frac{2LD^2k}{(k+3)(k+4)} \leq \Phi_k \leq \frac{3\max\{\Phi_0, LD^2\}}{k+3}, \forall k \in \mathbb{N}^+$$

Proof. By substituting $\alpha_k = \frac{2}{k+2}$ to the recursive formula of $\{\Phi_k\}_k$, we have that

$$\begin{aligned}
&\Phi_k = \frac{k+2}{k+4}\left[\Phi_{k-1} + \frac{2LD^2}{(k+2)^2}\right] \\
\iff &(k+4)(k+3)\Phi_k = (k+3)(k+2)\Phi_{k-1} + \frac{2LD^2(k+3)}{k+2} \\
\iff &(k+4)(k+3)\Phi_k = 12\Phi_0 + 2LD^2\sum_{i=1}^{k}\frac{i+3}{i+2}.
\end{aligned}$$

Considering the left hand side inequality, we have that

$$\begin{aligned}(k+4)(k+3)\Phi_k &\geq 2LD^2\sum_{i=1}^{k}1\\ \iff (k+4)(k+3)\Phi_k &\geq 2LD^2k\\ \iff \Phi_k &\geq \frac{2LD^2k}{(k+4)(k+3)}.\end{aligned}$$

Turning to the right inequality, we have that

$$\begin{aligned}(k+4)(k+3)\Phi_k &\leq 12\max\{\Phi_0, LD^2\} + 2\max\{\Phi_0, LD^2\}\sum_{i=1}^{k}\frac{3}{2}\\ \iff (k+4)(k+3)\Phi_k &\leq 3\max\{\Phi_0, LD^2\}(k+4)\\ \Rightarrow \Phi_k &\leq \frac{3\max\{\Phi_0, LD^2\}}{k+3}\end{aligned}$$

□

4.2 Convergence Analysis

Now, we have enough tools to establish some critical results which are summarized in the theorem (4.2). Noticably, despite the relaxation of the linear oracle, we successfully maintain the suboptimal convergence rate of $O(\frac{1}{K})$ for both objective functions.

Theorem 4.2. *Suppose that*

$$\Phi_i \geq \max\{\max\{f_{\text{upper}}(x)|x\in S\} - \min\{f_{\text{upper}}(x)|x\in S\}, \max\{f_{\text{lower}}(x)|x\in S\} - \min\{f_{\text{lower}}(x)|x\in S\},$$

$\forall 0 \leq i \leq 4$, *and Assumption 1 hold, under stepsizes* $\alpha_k = \frac{2}{k+2}$, *supports* β_k *such that* $0 \leq \beta_k - f^*_{\text{lower}} \leq O\left(\frac{1}{k}\right), \forall k \in \mathbb{N}$, *let* $\{x_k\}_{k=1,\dots K}$ *be the sequence generated by the Algorithm 5 we have that*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq O\left(\frac{1}{K}\right),$$

$$f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq O\left(\frac{1}{K}\right),$$

and thus, $\{x_k\}_k$ *admits a subsequence that converges to a point in* χ^*_2. *In fact, any accumulation point of* $\{x_k\}_k$ *is a solution of problem (1). In addition, under Assumption 2, we have*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -O\left[\left(\frac{1}{K}\right)^{\frac{1}{r}}\right], \forall K \in \mathbb{N},$$

and thus, $f_{\text{lower}}(x_K) - f^*_{\text{lower}}, f_{\text{upper}}(x_K) - f^*_{\text{upper}} \to 0$ *as* $K \to \infty$.

Proof. We only prove that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq O\left(\frac{1}{K}\right),$$

since the remaining claim can be proved by using similar arguments in Theorem 3.7. To do so, we use induction to prove that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \Phi_{K-1}, \forall K \in \mathbb{N}^+$$

From the assumption with respect to $\Phi_0, \Phi_1, \Phi_2, \Phi_3$, we have that the both inequalities are true for base cases $k = 1, 2, 3, 4$ and now we assume that such inequality is true up to $k = K > 4$. Firstly, we consider the case of positive call.

$$\begin{aligned}
f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \langle \nabla f_{\text{upper}}(x_K), x_{K+1} - x_K \rangle + \frac{L}{2}||x_{K+1} - x_K||^2 \\
\Rightarrow f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \alpha_K \langle \nabla f_{\text{upper}}(x_K), s_K - x_K \rangle + \alpha_K^2 \frac{L}{2}\|s_K - x_K\|^2 \\
\Rightarrow f_{\text{upper}}(x_{K+1}) - f^*_{\text{upper}} &\leq f_{\text{upper}}(x_K) - f^*_{\text{upper}} - \alpha_k \Phi_K + \frac{LD^2}{2}\alpha_K^2 \leq \Phi_{K-1} - \alpha_k \Phi_K + \frac{LD^2}{2}\alpha_K^2 = \Phi_K
\end{aligned}$$

Under the negative call, we have $\langle \nabla f_{\text{upper}}(x_K), s_K \rangle \leq \langle \nabla f_{\text{upper}}(x_K), x^* \rangle$ since $x^* \in \chi_1^* \subseteq \chi_{1,k}$. Hence, we have that

$$\begin{aligned}
f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \alpha_K \langle \nabla f_{\text{upper}}(x_K), x^* - x_K \rangle + \frac{LD^2}{2}\alpha_K^2 \\
f_{\text{upper}}(x_{K+1}) &\leq f_{\text{upper}}(x_K) + \alpha_K (f^*_{\text{upper}} - f_{\text{upper}}(x_K)) + \frac{LD^2}{2}\alpha_K^2 \\
f_{\text{upper}}(x_{K+1}) - f^*_{\text{upper}} &\leq \frac{K}{K+2}(f_{\text{upper}}(x_K) - f^*_{\text{upper}}) + \frac{2LD^2}{(K+2)^2} \leq \frac{K+2}{K+4}\Phi_{K-1} + \frac{2LD^2}{(K+2)^2} \leq \Phi_K
\end{aligned}$$

Where the last inequality comes from

$$\begin{aligned}
&\frac{K}{K+2}\Phi_{K-1} + \frac{2LD^2}{(K+2)^2} \leq \Phi_K \iff \frac{K}{K+2}\Phi_{K-1} + \frac{2LD^2}{(K+2)^2} \leq \frac{K+2}{K+4}\left[\Phi_{K-1} + \frac{2LD^2}{(K+2)^2}\right] \\
&\iff \frac{4LD^2}{(K+2)^2(K+4)} \leq \frac{4}{(K+2)(K+4)}\Phi_{K-1} \iff \Phi_{K-1} \geq \frac{LD^2}{K+2}
\end{aligned}$$

and the most right inequality can be sufficently justified by observing the following inequality, where the left term is the lower bound of Φ_{K-1}, which we obtain in the Theorem 4.1.

$$\frac{2LD^2(K-1)}{(K+3)(K+2)} \geq \frac{LD^2}{K+2} \iff \frac{(K-5)}{(K+2)(K+3)} \geq 0, \forall K \geq 5$$

Besides, from Theorem 4.1, we have that $\Phi_K \leq O\left(\frac{1}{K}\right)$. Therefore, the claim is true. □

Remark 4.1. The assumption $\Phi_0, \Phi_1, \Phi_2, \Phi_3 \geq \max\{f_{\text{upper}}(x)|x \in S\} - \min\{f_{\text{upper}}(x)|x \in S\}$, $\max\{f_{\text{lower}}(x)|x \in S\} - \min\{f_{\text{lower}}(x)|x \in S\}$ can be satisfied by appropriately setting Φ_0. Specifically, we should note the following estimate for any $x \in S$

$$\begin{aligned}
f_{\text{lower}}(x) &\geq f_{\text{lower}}(x_0) + \langle \nabla f_{\text{lower}}(x_0), x - x_0 \rangle \\
&\geq f_{\text{lower}}(x_0) - \|\nabla f_{\text{lower}}(x_0)\|_* \|x - x_0\| \\
&\geq f_{\text{lower}}(x_0) - \|\nabla f_{\text{lower}}(x_0)\|_* D := m_1.
\end{aligned}$$

Following the same reasoning, we also obtain

$$f_{\text{upper}}(x) \geq f_{\text{upper}}(x_0) - \|\nabla f_{\text{upper}}(x_0)\|_* D := m_2, \forall x \in S$$

In addition, we have that for all $x \in S$

$$\begin{aligned} f_{\text{lower}}(x) &\leq f_{\text{lower}}(x_0) + \langle \nabla f_{\text{lower}}(x_0), x - x_0 \rangle + \frac{L_{\text{lower}}}{2}||x - x_0||^2 \\ &\leq f_{\text{lower}}(x_0) + \|\nabla f_{\text{lower}}(x_0)\|_* \|x - x_0\| + \frac{L_{\text{lower}}}{2}||x - x_0||^2 \\ &\leq f_{\text{lower}}(x_0) + \|\nabla f_{\text{lower}}(x_0)\|_* D + \frac{L_{\text{lower}}}{2} D^2 := M_1 \end{aligned}$$

Similarly, we have that $f_{\text{upper}}(x) \leq f_{\text{upper}}(x_0) + \|\nabla f_{\text{upper}}(x_0)\|_* D + \frac{L_{\text{upper}}}{2} D^2 := M_2, \forall x \in S$. In the proof of Theorem 4.1, we have the following estimate

$$\Phi_k \geq \frac{12\Phi_0}{(k+4)(k+3)}, \forall k \in \mathbb{N}$$

By setting Φ_0 such that

$$\Phi_0 \geq \frac{(3+4)(3+3)}{12} \max\{M_2 - m_2, M_1 - m_1\},$$

we have the following consequences by noting that $M_1 \geq \max\{f_{\text{lower}}(z) | z \in S\}, M_2 \geq \max\{f_{\text{upper}}(z) | z \in S\}$ and $m_1 \leq \min\{f_{\text{lower}}(z) | z \in S\}, m_2 \leq \min\{f_{\text{upper}}(z) | z \in S\}$

$$\Phi_3, \Phi_2, \Phi_1, \Phi_0 \geq \max\{M_2 - m_2, M_1 - m_1\}$$

$$\Rightarrow \Phi_2, \Phi_1, \Phi_0 \geq \max\left\{\max\{f_{\text{upper}}(x) | x \in S\} - \min\{f_{\text{upper}}(x) | x \in S\}, \max\{f_{\text{lower}}(x) | x \in S\} - \min\{f_{\text{lower}}(x) | x \in S\}\right\}$$

5 ACG-BiO Method under The Relaxation of The Boundedness of Feasible Region

5.1 Proposed Method

In this section, we consider the feasible region S that is not necessarily bounded, closed and convex and we assume in this section that the bilevel problem still has solutions over the feasible region S, which is possibly unbounded.

Lemma 5.1. *Let $\{\Delta_k\}_k$ be a sequence of compact sets in $\mathbb{R}^n$ with diameters $\{\sigma_k\}_k$ such that $\Delta_k \subset \Delta_{k+1}, \forall k \in \mathbb{N}$ such that given any point a in $\mathbb{R}^n$, there exist some integer k_a that Δ_{k_a} contains a, then there exists $k_0 \in \mathbb{N}$ such that $x^* \in \chi_{1,k}$ and $f^*_{1,k} = f^*_{\text{lower}}, \forall k \geq k_0$, where $\chi_{1,k}$ is defined as in Algorithm 6, and $f^*_{1,k} := \min\{f_{\text{lower}}(x) | x \in S \cap \Delta_k\}$. Consequently, $\chi_{1,k}$ is nonempty, closed, and convex for all $k \in \mathbb{N}, k \geq k_0$.*

Proof. Under construction of $\{\Delta_k\}_k$, there exists $k_0 \in \mathbb{N}$ such that $\sigma_{k_0} \geq \sqrt{n}\|x_0 - x^*\| > \sigma_{k_0 - 1}$. In fact, it is sufficient to set $k_0 := \left\lceil \sigma^{-1}\left(\sqrt{n}\|x_0 - x^*\|\right)\right\rceil$. Therefore, $x^* \in \Delta(x_0, \sigma_{k_0}/\sqrt{n})$ and as a result, for every $k \geq k_0$, $x^* \in S \cap \Delta\left(x_0, \sigma_k/\sqrt{n}\right)$. Furthermore, since $\langle f_{\text{lower}}(x_k), x^* - x_k \rangle \leq f_{\text{lower}}(x^*) - f_{\text{lower}}(x_k) \leq \beta_k - f_{\text{lower}}(x_k), \forall k \in \mathbb{N}$, we have that $x^* \in \chi_{1,k}, \forall k \geq k_0$. As a consequence, $\min\{f_{\text{lower}}(x) | x \in S \cap \Delta(x_0, \sigma_k/\sqrt{n})\} = \min\{f_{\text{lower}}(x) | x \in S\}$ for every $k \geq k_0$, which implies that $f^*_{1,k} = f^*_{\text{lower}}, \forall k \geq k_0$. □

Algorithm 6: Unbounded adaptive conditional gradient-based bilevel optimisation method.

Data: stepsizes $\{\alpha_k\}_k$, supports $\{\beta_k\}_k$, residuals $\{\gamma_k\}_k$, regions $\{\Delta_k\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in S$;

for $k = 0, 1, \ldots, K$ **do**

  if $\chi_{1,k} := \{s \in S \cap \Delta(x_0, \sigma_k/\sqrt{n}) : \langle \nabla f_{\text{lower}}(x_k), s - x_k \rangle \leq \beta_k - f_{\text{lower}}(x_k)\} = \emptyset$ **then**

    Compute $s_k \leftarrow x_k$

  else

    Compute s_k such that

$$\langle \nabla f_{\text{upper}}(x_k), s_k \rangle \leq \min_{s \in \chi_{1,k}} \langle \nabla f_{\text{upper}}(x_k), s \rangle + \gamma_k$$

  Compute $x_{k+1} \leftarrow x_k + \alpha_k(s_k - x_k)$;

5.2 Convergence Analysis

Proposition 5.2. *Suppose that Assumption 1 holds, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0$, regions $\{\Delta_k\}_k$ satisfy conditions in Theorem 5.1, let $\{x_k\}_{k=1,\ldots K}$ be the sequence generated by Algorithm 6 we have that*

$$0 \leq f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq \frac{(k_0+1)k_0[f_{\text{lower}}(x_{k_0}) - f^*_{\text{lower}}]}{(K+1)K} + \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} \sigma^2_{i-1} i}{i+1} \right).$$

Proof. We have that $f_{\text{lower}}(x_{K+1}) \leq f_{\text{lower}}(x_K) + \langle \nabla f_{\text{lower}}(x_K), x_{K+1} - x_K \rangle + \frac{L_{\text{lower}}}{2} ||x_{K+1} - x_K||^2$ and we also obtain that $\langle \nabla f_{\text{lower}}(x_K), x_{K+1} - x_K \rangle = \alpha_K \langle \nabla f_{\text{lower}}(x_K), s_K - x_K \rangle \leq \alpha_K(\beta_K - f_{\text{lower}}(x_K))$. Hence, we have

$$\begin{aligned}
f_{\text{lower}}(x_{K+1}) &\leq f_{\text{lower}}(x_K) + \alpha_K \langle \nabla f_{\text{lower}}(x_K), s_K - x_K \rangle + \frac{L_{\text{lower}}}{2} \alpha^2_K \|s_K - x_K\|^2 \\
\Rightarrow f_{\text{lower}}(x_{K+1}) &\leq f_{\text{lower}}(x_K) + \alpha_K(\beta_K - f_{\text{lower}}(x_K)) + \frac{L_{\text{lower}} \sigma^2_K}{2} \alpha^2_K \\
\Rightarrow f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}} &\leq (1 - \alpha_K)(f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + \alpha_K(\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}} \sigma^2_K}{2} \alpha^2_K \\
\Rightarrow f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}} &\leq \frac{K}{K+2}(f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + \frac{2}{K+2}(\beta_K - f^*_{\text{lower}}) + \frac{2L_{\text{lower}} \sigma^2_K}{(K+2)^2} \\
\Rightarrow (K+2)(K+1)[f_{\text{lower}}(x_{K+1}) - f^*_{\text{lower}}] &\leq (K+1)K(f_{\text{lower}}(x_K) - f^*_{\text{lower}}) + 2(K+1) \left(\beta_K - f^*_{\text{lower}} + \frac{L_{\text{lower}} \sigma^2_K}{K+2} \right) \\
\Rightarrow (K+1)K[f_{\text{lower}}(x_K) - f^*_{\text{lower}}] &\leq (k_0+1)k_0[f_{\text{lower}}(x_{k_0}) - f^*_{\text{lower}}] \\
&\quad + 2 \sum_{i=k_0+1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} \sigma^2_{i-1} i}{i+1} \right) \\
\iff f_{\text{lower}}(x_K) - f^*_{\text{lower}} &\leq \frac{(k_0+1)k_0[f_{\text{lower}}(x_{k_0}) - f^*_{\text{lower}}]}{(K+1)K} \\
&\quad + \frac{2}{(K+1)K} \sum_{i=k_0+1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}} \sigma^2_{i-1} i}{i+1} \right).
\end{aligned}$$

□

Corollary 5.3. *Suppose that Assumption 1 holds, under stepsizes* $\alpha_k = \frac{2}{k+2}$, *supports* $\{\beta_k\}_k$ *such that* $\beta_k \geq f^*_{\text{lower}}$, *and residuals* $\{\gamma_k\}$ *such that* $\gamma_k \geq 0$, *regions* $\{\Delta_k\}_k$ *satisfy conditions in Theorem 5.1, let* $\{x_k\}_{k=1,\dots K}$ *be the sequence generated by Algorithm 6, if* $\beta_k \to f^*_{\text{lower}}$ *and* $\sigma_k^2/k \to 0$ *then* $f_{\text{lower}}(x_K) \to f^*_{\text{lower}}$.

Proof. By applying Theorem 2.4, we have that if $\beta_k \to f^*_{\text{lower}}$ and $\sigma_k^2/k \to 0$ then

$$\begin{aligned}
&\lim_{K\to\infty} \frac{2}{(K+1)K} \sum_{i=k_0+1}^{K} \left(i(\beta_{i-1} - f^*_{\text{lower}}) + \frac{L_{\text{lower}}\sigma_{i-1}^2 i}{i+1} \right) \\
&= \lim_{K\to\infty} \frac{2}{(K+2)(K+1) - (K+1)K} \left[(K+1)(\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}}\sigma_K^2 (K+1)}{K+2} \right] \\
&= \lim_{K\to\infty} \left[(\beta_K - f^*_{\text{lower}}) + \frac{L_{\text{lower}}\sigma_K^2}{K+2} \right] = 0
\end{aligned}$$

By squeeze theorem, we have that

$$\lim_{K\to\infty} (f_{\text{lower}}(x_K) - f^*_{\text{lower}}) = 0.$$

□

Proposition 5.4. *Suppose that Assumption (1) holds, under stepsizes* $\alpha_k = \frac{2}{k+2}$, *supports* $\{\beta_k\}_k$ *such that* $\beta_k \geq f^*_{\text{lower}}$, *and residuals* $\{\gamma_k\}$ *such that* $\gamma_k \geq 0$, *regions* $\{\Delta_k\}_k$ *satisfy conditions in Theorem 5.1, let* $\{x_k\}_{k=1,\dots K}$ *be the sequence generated by the algorithm (6) we have that*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \frac{(k_0+1)k_0[f_{\text{upper}}(x_{k_0}) - f^*_{\text{upper}}]}{(K+1)K} + \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i\gamma_{i-1} + \frac{L_{\text{upper}}\sigma_{i-1}^2 i}{i+1} \right).$$

Proof. We have that

$$\begin{aligned}
\langle \nabla f_{\text{upper}}(x_k), x_{k+1} - x_k \rangle =& \alpha_k \langle \nabla f_{\text{upper}}(x_k), s_k - x_k \rangle \\
\leq& \alpha_k \left(\langle \nabla f_{\text{upper}}(x_k), x_2^* - x_k \rangle + \gamma_k \right) \\
\leq& \alpha_k (f^*_{\text{upper}} - f_{\text{upper}}(x_k) + \gamma_k),
\end{aligned}$$

for any $x_2^* \in \chi_2^*$. In addition, we have that

$$\begin{aligned}
& f_{\text{upper}}(x_{K+1}) \leq f_{\text{upper}}(x_K) + \langle \nabla f_{\text{upper}}(x_K), x_{K+1} - x_K \rangle + \frac{L_{\text{upper}}}{2} ||x_{K+1} - x_K||^2 \\
\Rightarrow\ & f_{\text{upper}}(x_{K+1}) \leq f_{\text{upper}}(x_K) + \alpha_K (f^*_{\text{upper}} - f_{\text{upper}}(x_K) + \gamma_K) + \frac{L_{\text{upper}}\sigma_K^2}{2} \alpha_K^2 \\
\Rightarrow\ & f_{\text{upper}}(x_{K+1}) - f^*_{\text{upper}} \leq \frac{K}{K+2} (f_{\text{upper}}(x_K) - f^*_{\text{upper}}) + \frac{2}{K+2}\gamma_K + \frac{2L_{\text{upper}}\sigma_K^2}{(K+2)^2}
\end{aligned}$$

Similar to the proof of Theorem 3.2, we have that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq \frac{(k_0+1)k_0[f_{\text{upper}}(x_{k_0}) - f^*_{\text{upper}}]}{(K+1)K} + \frac{2}{(K+1)K} \sum_{i=1}^{K} \left(i\gamma_{i-1} + \frac{L_{\text{upper}}\sigma_{i-1}^2 i}{i+1} \right).$$

□

Proposition 5.5. *Suppose that Assumption 1 holds, under stepsizes $\alpha_k \in (0,1)$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0$, regions $\{\Delta_k\}_k$ satisfy conditions in Theorem 5.1, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by Algorithm 6, we have that if Assumption 2 also holds, then $\forall K \geq k_0, K \in \mathbb{N}$*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -\left(\|\nabla f_{\text{upper}}(x_0)\|_* + 2L_{\text{upper}}\sigma_K\right)\left[\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right]^{\frac{1}{r}} - L_{\text{upper}}\left[\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right]^{\frac{2}{r}}.$$

Proof. Under assumption (2), we have that

$$\frac{\alpha}{r}\|x_K - v_K\|^r \leq f_{\text{lower}}(x_K) - f^*_{\text{lower}} \iff \|x_K - v_K\| \leq \left[\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right]^{\frac{1}{r}}$$

where $v_k \in \arg\min_{x' \in \chi_1^*} \|x' - x_K\|$. Thus, we have that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq f_{\text{upper}}(x_K) - f_{\text{upper}}(v_K) \geq \langle \nabla f_{\text{upper}}(v_K), x_K - v_K\rangle$$

$$\langle \nabla f_{\text{upper}}(v_K), x_K - v_K\rangle \geq -\|\nabla f_{\text{upper}}(v_K)\|_*\|x_K - v_K\| \geq -(\|\nabla f_{\text{upper}}(x_K)\|_* + L_{\text{upper}}\|x_k - v_K\|)\|x_K - v_K\|$$

$$\Rightarrow \langle \nabla f_{\text{upper}}(v_K), x_K - v_K\rangle \geq -(\|\nabla f_{\text{upper}}(x_0)\|_* + L_{\text{upper}}\|x_k - x_0\| + L_{\text{upper}}\|x_k - v_K\|)\|x_K - v_K\|$$

$$\Rightarrow \langle \nabla f_{\text{upper}}(v_K), x_K - v_K\rangle \geq -(\|\nabla f_{\text{upper}}(x_0)\|_* + 2L_{\text{upper}}\sigma_K + L_{\text{upper}}\|x_k - v_K\|)\|x_K - v_K\|$$

Therefore, we have that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -\left(\|\nabla f_{\text{upper}}(x_0)\|_* + 2L_{\text{upper}}\sigma_K\right)\left[\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right]^{\frac{1}{r}} - L_{\text{upper}}\left[\frac{r}{\alpha}(f_{\text{lower}}(x_K) - f^*_{\text{lower}})\right]^{\frac{2}{r}}.$$

□

Corollary 5.6. *Suppose that Assumption 1 and Assumption 2 hold, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports $\{\beta_k\}_k$ such that $\beta_k \geq f^*_{\text{lower}}$, $\beta_k \to f^*_{\text{lower}}$, and residuals $\{\gamma_k\}$ such that $\gamma_k \geq 0$,$\gamma_k \to 0$, regions $\{\Delta_k\}_k$ satisfy conditions in Theorem 5.1 and diameters $\{\sigma_k\}_k$ such that $\sigma_k^{2+r}/k \to 0$, $\forall k \in \mathbb{N}$, let $\{x_k\}_{k=1,\dots K}$ be the sequence generated by Algorithm 6, we have that*

$$f_{\text{upper}}(x_K) \to f^*_{\text{upper}}.$$

Theorem 5.7. *Suppose that Assumption 1 holds, under stepsizes $\alpha_k = \frac{2}{k+2}$, supports β_k such that $0 \leq \beta_k - \min\{f_{\text{lower}}(x) | x \in S \cap \Delta(x_0, \sigma_k/\sqrt{n})\} \leq O(\sigma_k^2/k)$, if $\chi_k \neq \emptyset$, residuals $\{\gamma_k\}_k$ such that $\gamma_k \leq O\left(\frac{1}{k}\right)$, and regions $\{\Delta_k\}_k$ satisfy conditions in Theorem 5.1. For $\{x_k\}_{k=1,\dots K}$ be the sequence generated by Algorithm 6, then $\forall K \geq k_0$*

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \leq O\left(\frac{\sigma_K^2}{K}\right),$$

$$f_{\text{lower}}(x_K) - f^*_{\text{lower}} \leq O\left(\frac{\sigma_K^2}{K}\right).$$

In addition, under Assumption 2, we have that

$$f_{\text{upper}}(x_K) - f^*_{\text{upper}} \geq -O\left[\left(\frac{\sigma_K^{2+r}}{K}\right)^{\frac{1}{r}}\right], \forall K \geq k_0, K \in \mathbb{N},$$

Furthermore, if $\sigma_k^{2+r}/k \to 0$, *then* $f_{\text{lower}}(x_K) - f^*_{\text{lower}}, f_{\text{upper}}(x_K) - f^*_{\text{upper}} \to 0$ *as* $K \to \infty$.

Remark 5.1. To generate supports $\{\beta_k\}_k$ satisfy conditions in theorem (5.7), given an integer k, we run the CG method [4] on f_{lower} over $\chi_{1,k}$ for t_k iterations and obtain $y_k \in S$ such that

$$f_{\text{lower}}(y_k) - \min\{f_{\text{lower}}(x) | x \in S \cap \Delta(x_0, \sigma_k/\sqrt{n})\} \leq \frac{2L_{\text{lower}}(2\sigma_k)^2}{t_k + 2} \leq O\left(\frac{\sigma_k^2}{k}\right).$$

Then, we set $\beta_k := f_{\text{lower}}(y_k)$.

6 Computational Experiment

6.1 Experiment Description and Data Preprocessing

In this section, we aim to solve a sparse linear regression problem on the Wikipedia Math Essential dataset mentioned in the study conducted by Rozemberczki et al. [14]. Specifically, the data set consists of 731 attributes including numerical variables only. In such case, we select the attribute '0', which is the daily number of visits on page 'Mathematics' as the response variable, which resulted in data matrix $D \in \mathbb{R}^{n\times d}$ in which $n = 1068$ observations and $d = 730$ characteristics and an outcome vector $y \in \mathbb{R}^n$. In details, 60 % of the dataset is randomly assigned to training dataset $(X_{\text{train}}, y_{\text{train}})$, another 20 % of the dataset is chosen to be validation dataset $(X_{\text{valid}}, y_{\text{valid}})$ and the test dataset $(X_{\text{test}}, y_{\text{test}})$is made up of the final 20 % of the data set. In addition, we also performed a division of 10^4 for every elements of D and y to avoid numerical instability.

As squared loss function is adopted, the lower level objective function is the training error $f_{\text{lower}}(\beta) = \frac{1}{2}\|X_{\text{train}}\beta - y_{\text{train}}\|_2^2$, the upper-level objective function is the validation error $f_{\text{upper}}(\beta) = \frac{1}{2}\|X_{\text{valid}}\beta - y_{\text{valid}}\|_2^2$ and the feasible region in this case is $S = \{\beta \in \mathbb{R}^d | \|\beta\|_1 \leq \lambda\}$ for $\lambda = 1$ to induce sparsity in β. The performance of our training and validation procedure will be evaluated based on the test error $\frac{1}{2}\|X_{\text{test}}\beta - y_{\text{test}}\|_2^2$. These statistics of ACG-BiO will be compared to those of CG-BiO [9], the minimal norm gradient (MNG) method devised by Beck and Sabach [1], the bilevel gradient sequential averaging (BiG-SAM) method proposed by Sabach and Shtern [15], and averaging iteratively regularized gradient (a-IRG) method by Kaushik and Yousefian [10] to assess the efficiency.

6.2 Mathematical Formulation

With the above description, we have the following simple bilevel optimisation set up:

$$\begin{aligned} \min_{\beta\in\mathbb{R}^d} \quad & \frac{1}{2}\|X_{\text{valid}}\beta - y_{\text{valid}}\|_2^2 \\ \text{s.t.} \quad & \beta \in \arg\min_{\xi\in\mathbb{R}^d, \|\xi\|_1\leq\lambda} \frac{1}{2}\|X_{\text{train}}\xi - y_{\text{train}}\|_2^2, \end{aligned} \tag{3}$$

Under this formulation, we have that $S = S_\lambda := \{\beta \in \mathbb{R}^d \mid \|\beta\|_1 \leq \lambda\}$ is convex and compact with a diameter of $D = \lambda\sqrt{2}$. Furthermore, we also have

$$\nabla f_{\text{lower}}(\beta) = (X_{\text{train}})^T(X_{\text{train}}\beta - y_{\text{train}}) \Rightarrow D^2_\beta f_{\text{lower}}(\beta) = (X_{\text{train}})^T X_{\text{train}} \succeq 0, \forall \beta \in S$$

Hence, f_{lower} is a convex function over S. Similarly, we can see that f_{upper} is also a convex function over S. In addition, we also have

$$\|\nabla f_{\text{lower}}(\beta_1) - \nabla f_{\text{lower}}(\beta_2)\|_2 = \|(X_{\text{train}})^T X_{\text{train}}(\beta_1 - \beta_2)\|_2 \leq \lambda_{max}\left((X_{\text{train}})^T X_{\text{train}}\right)\|(\beta_1 - \beta_2)\|_2$$

Thus, ∇f_{lower} and ∇f_{upper} are Lipschit-continuous with $L_{\text{lower}} = \lambda_{max}\left((X_{\text{train}})^T X_{\text{train}}\right)$ and $L_{\text{upper}} = \lambda_{max}\left((X_{\text{valid}})^T X_{\text{valid}}\right)$ respectively. Therefore, both ACG-BiO and CG-BiO can be used to solve problem (3).

6.3 Implementation Details

6.3.1 Solving for optimal values.

To obtain the values of f^*_{lower} and f^*_{upper}, we solve the following two quadratic programming problems using CVX [6] [7]:

$$\begin{aligned} \min_{\beta \in \mathbb{R}^d} \quad & \frac{1}{2}\|X_{\text{train}}\beta - y_{\text{train}}\|_2^2 \\ \text{s.t.} \quad & \|\beta\|_1 \leq \lambda, \end{aligned} \tag{4}$$

and

$$\begin{aligned} \min_{\beta \in \mathbb{R}^d} \quad & \frac{1}{2}\|X_{\text{valid}}\beta - y_{\text{valid}}\|_2^2 \\ \text{s.t.} \quad & \|\beta\|_1 \leq \lambda, \\ & \frac{1}{2}\|X_{\text{train}}\beta - y_{\text{train}}\|_2^2 \leq f^*_{\text{lower}}. \end{aligned} \tag{5}$$

For all methods, we run them for no more than $K = 5 \times 10^4$ iterations. Before, entering the primary method, we would like to discuss a accelerated proximal gradient method known as the fast iterative shrinkage-thresholding algorithm (FISTA) by Beck and Teboulle [2], which has convergence rate of $O\left(1/K^2\right)$ as it will be used in implementing ACG-BiO method 4 and a projection oracle invented for this problem, which can save computational cost as it does not require solving any quadratic programming problem.

6.3.2 Fast Iterative Shrinkage-Thresholding Algorithm

To begin with, FISTA [2] can tackle to following problem

$$\min_{x \in \mathbb{R}^n} \quad f(x) := g(x) + h(x),$$

where g is a convex, differentiable function and h is a convex function. In addition, the proximal function with respect to a function h and a step parameter t can be defined as follows:

$$\text{prox}_{h,t} = \arg\min_{z \in \mathbb{R}^n}\left(\frac{1}{2t}\|x - z\|^2 + h(z)\right).$$

Algorithm 7: [Beck and Teboulle [2]] Fast iterative shrinkage-thresholding algorithm (FISTA).

Data: stepsizes $\{t_k\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 = x_{-1} \in \mathbb{R}^n$;

for $k = 1, \ldots, K$ **do**

  Compute $v \leftarrow x_{k-1} + \frac{k-2}{k-1}(x_{k-1} - x_{k-2})$;

  Compute $x_k = \text{prox}_{h,t_k}(v - t_k \nabla g(v))$;

Along the Algorithm 7, Beck and Teboulle [2] also came up with the following result for the convergence guarantee for the method.

Theorem 6.1 (Beck and Teboulle [2])**.** *Assuming that g is convex, $L-$smooth with $L > 0$, has an effective domain of $\mathbb{R}^n$ while h is convex, and the proximal is easy to evaluate. Under the constant stepsizes $t_k = t \leq 1/L, \forall k \in \mathbb{N}$, the sequence $\{x_k\}_k$ generated by Algorithm 7 satisfies the following estimate:*

$$f(x_k) - f^* \leq \frac{2\|x^* - x_0\|^2}{t(k+1)^2}.$$

In this example, we observe that $f := f_{\text{lower}}$ with $L := L_{\text{lower}} > 0$, and $h := I_S$, which is the indicator of the convex feasible region S and therefore, is also convex. In this case, the proximal is reduced to be the projection onto the set S, which will be discussed in the next section.

6.3.3 Projection Oracle

We devote this section to devise an inexpensive algorithm to solve exactly the following problem:

$$\begin{aligned} \min_{x \in \mathbb{R}^d} \quad & \|x - v\|_2^2 \\ \text{s.t.} \quad & \|x\|_1 \leq \lambda, \end{aligned} \tag{6}$$

where v is a given vector in $\mathbb{R}^d$, and λ is a given non-negative number. To begin with, in case $\|v\|_1 \leq \lambda$, the solution will be v itself. Therefore, we consider $\|v\|_1 > \lambda$. Under such case, we have the following minimisation problem:

$$\begin{aligned} \min_{y \in \mathbb{R}^d_+} \quad & \|y - v_{\text{abs}}\|_2^2 \\ \text{s.t.} \quad & \langle \mathbf{1}_d, y \rangle \leq \lambda, \end{aligned} \tag{7}$$

where v_{abs} is the vector such that $[v_{\text{abs}}]_i = |v_i|, \forall i = 1, \ldots, d$.

Lemma 6.2. *Let x^+ be the solution of the problem (6) and y^+ be the solution of problem (7) then we have:*

$$[x^+]_i = sign(v_i)\,[y^+]_i, \forall i = 1, \ldots, d$$

Algorithm 8: Projection oracle on set S-$\text{Proj}_{S_\lambda}(v)$.

Data: vector $v \in \mathbb{R}^d$

Result: projection of v onto S

Set $g := -1_d$;
Compute $l \leftarrow \dim(v)$;
if $\|v\|_1 \leq \lambda$ **then**
    return v;
else
    Compute $v_{\text{abs}} \leftarrow (|v_1|, \ldots, |v_d|)^T$;
    while $l > 0$ **do**
        Compute $v_{\text{abs}} \leftarrow \text{Proj}_{P_l}(v_{\text{abs}})$;
        Compute vector r composed by the indices of negative elements in g;
        if *there exists any non-positive element in* v_{abs} **then**
            for $i = 1, \ldots, dim(v_{\text{abs}})$ **do**
                if $[v_{\text{abs}}]_i \leq 0$ **then**
                    $g_{r_i} = 0$;
            end
            Compute v_{abs} such that it contains only positive elements of the current v_{abs};
            Compute $l \leftarrow \dim(v_{\text{abs}})$;
        else
            for $i = 1, \ldots, dim(v_{\text{abs}})$ **do**
                $g_{r_i} = [v_{\text{abs}}]_i$;
            end
            Compute $\text{Proj}_S(v) \leftarrow (g_1 \text{sign}(v_1), \ldots, g_d \text{sign}(v_d))$;
            Set $l := 0$;
            return $\text{Proj}_S(v)$;
        end
    end
end

Proof. Let $z \in \mathbb{R}^d_+$ such that $z_i = \text{sign}\,(v_i)\,[y^+]_i, \forall i = 1, \ldots, d$ then $\|z\|_1 = \|x^+\|_1 \leq \lambda$. We have the following estimate

$$\|z - v\|_2^2 \geq \|x^+ - v\|_2^2$$

On the other hand, we have that

$$\|x^+ - v\|_2^2 = \|x^+\|^2 - 2\langle x^+, v\rangle + \|v\|_2^2 \geq \|x^+\|^2 - 2\|x^+\|_2\|v\|_2 + \|v\|_2^2 = |\|z\|_2 - \|v\|_2|^2,$$

and since $z_i v_i = (\text{sign}\,(v_i)\, v_i)[y^+]_i \geq 0, \forall i = 1, \ldots, d$, we have that

$$\langle z, v\rangle = \|z\|_2\|v\|_2 \Rightarrow \|z - v\|_2 = |\|z\|_2 - \|v\|_2|$$

Hence, $\|z - v\|_2^2 = \|x^+ - v\|_2^2 \Rightarrow z = x^+$. □

Lemma 6.3. $\langle \mathbf{1}_d, y^+\rangle = \lambda$

Proof. Assume $\langle \mathbf{1}_d, y^+\rangle < \lambda$ and consider $g(t) := \langle \mathbf{1}_d, v_{\text{abs}} + t(y^+ - v_{\text{abs}})\rangle$, we have that g is continuous over $[0, 1]$ and $g(0)g(1) < 0$ then there exists $t_0 \in (0, 1)$ such that $g(t_0) = 0$. In that case, $v_{\text{abs}} + t_0(y^+ - v_{\text{abs}})$ belongs to the feasible region of problem (7). Nevertheless, we have that

$$\|v_{\text{abs}} + t_0(y^+ - v_{\text{abs}}) - v_{\text{abs}}\|_2^2 = (t_0)^2\|y^+ - v_{\text{abs}}\|_2^2 < \|y^+ - v_{\text{abs}}\|_2^2,$$

which contradicts the definition of y^+. □

Let $P_l := \{s \in \mathbb{R}^l | \langle \mathbf{1}_l, s\rangle = \lambda\}, l \in \mathbb{N}^*$ and $u := \text{Proj}_{P_d}(v_{\text{abs}})$, then we have the following observation.

Lemma 6.4. *For all $q \in \mathbb{R}^l, l \in \mathbb{N}^*$, we have that*

$$Proj_{P_l}(q) = \frac{\lambda - \langle \mathbf{1}_l, q\rangle}{l}\mathbf{1}_d - q.$$

Proof. For all $p \in P_l$ we have

$$\|p - q\|_2^2 = \sum_{i=1}^{l}(p_i - q_i)^2 \geq \frac{1}{l}\left(\sum_{i=1}^{d}(q_i - p_i)\right)^2 = \frac{1}{l}(\langle \mathbf{1}_l, q\rangle - \lambda)^2$$

The equalities happens when

$$q_1 - p_1 = \ldots = q_l - p_l, \langle \mathbf{1}_l, p\rangle = \lambda \iff p_i = \frac{\lambda - \langle \mathbf{1}_l, q\rangle}{l} - q_i, \forall i = 1, \ldots, l$$

□

Using the above lemma, we have that

$$\langle u - v_{\text{abs}}, p\rangle = 0, \forall p \in P_d \Rightarrow \|p - v_{\text{abs}}\|_2^2 = \|p - u\|_2^2 + \|u - v_{\text{abs}}\|_2^2.$$

Combining this fact with lemma 6.3, we have that problem (7) is equivalent to

$$\begin{aligned} \min_{y \in \mathbb{R}^d_+} \quad & \|y-u\|_2^2 \\ \text{s.t.} \quad & \langle \mathbf{1}_d, y \rangle = \lambda, \end{aligned} \tag{8}$$

It should be noted that any element in the projection on P_l should have at least one positive component. Without loss of generality, we assume that u has $d^{'} \leq d$ positive components and such components are $\{u_1, \ldots, u_{d'}\}$. In that case, we have:

$$\left(\sum_{i=1}^{d^{'}} u_i \right) \geq \left(\sum_{i=1}^{d} u_i \right) = \lambda$$

If $d^{'} = d$ then $y^+ = u$. Otherwise, let w be a $d^{'}$ vector such that its component is the positive components of u.

Lemma 6.5. *Let y^{++} be the solution of the following problem:*

$$\begin{aligned} \min_{y \in \mathbb{R}^{d^{'}}_+} \quad & \|y-w\|_2^2 \\ s.t. \quad & \langle \boldsymbol{1}_{d'}, y \rangle \leq \lambda, \end{aligned} \tag{9}$$

then we have

$$y^+ = \begin{bmatrix} y^{++} \\ \boldsymbol{0}_{(d-d')} \end{bmatrix}$$

Proof.

$$\|y^+ - u\|_2^2 = \sum_{i=1}^{d^{'}} \left((y^+)_i - u_i \right)^2 + \sum_{i=d'+1}^{d} \left((y^+)_i - u_i \right)^2 \geq \|y^{++} - w\|_2^2 + \sum_{i=d'+1}^{d} (-u_i)^2 = \left\| \begin{bmatrix} y^{++} \\ \mathbf{0}_{(d-d')} \end{bmatrix} - u \right\|_2^2$$

Hence, we have that

$$y^+ = \begin{bmatrix} y^{++} \\ \mathbf{0}_{(d-d')} \end{bmatrix}$$

□

Thus, by noting that $\|w\|_1 \geq \lambda$ we reduce a d dimensional problem to a $d^{'} < d$ dimensional problem with similar nature. Now, we are ready to introduce the projection oracle in algorithm (8).

Proposition 6.6. *Algorithm (8) always ends either step 4 in at most d iterations of **while** loop and returns $Proj_{S_\lambda}(v)$ for all $v \in \mathbb{R}^d$.*

Proof. If $\|v\|_1 \leq \lambda$, then v is also $\text{Proj}_{S_\lambda}(v)$ and algorithm (8) ends in step 4.

If $\|v\|_1 > \lambda$, then we will prove that **else** will happen in a finite amount of time. Assume that we are in step 9 after an amount of time and we are facing situation where we have some non-positive elements in current vector g. In that case, the update in step 13 is result of lemma (6.5) when we consider the minimisation with smaller dimension and fill in the zero value for components of $\text{Proj}_{S_\lambda}(v)$ corresponding to non-positive components of u . Since there are some non-positive elements in g, the new l in step 16 is strictly smaller than l before the update. As we note that the projection onto P_l has at least one positive component for any $l > 0$,

and l is always integer and decreases throughout time, it has to be that l will never be 0 in step 16 and that the algorithm (8) will enter step 17 in finite amount of time.

Note that as soon as we enter step 17, the components in u will either be 0 or -1 and the number of -1 components is exactly l. Since there are only positive element in v_{abs} and $v_{\text{abs}} \in P_l$. Therefore, by filling the -1 components of g by components of v_{abs} and applying lemma (6.2) as well as lemma (6.5), the returned vector is exactly the projection of v onto S_λ. □

6.3.4 Implementation of ACG-BiO Method

Instead of using Frank Wolfe method for generating the sequence of support $\{\beta_k\}_{k=0,1,\dots,K} \subset \mathbb{R}$ of ACG-BiO (4) mentioned in remark (3.1), we run FISTA [2] with step size $t_k = 1/L_{\text{lower}}$ over $3K$ iterations. In addition, to save the storing space, rather than storing the whole sequence, we only extract and utilise β_k over K iterations since the this element satisfies all the conditions that $\{\beta_k\}_{k=0,1,\dots,K-1}$ satisfy. Turning to the initial β_0, we set $\beta_0 := \beta_k$. In constrast to the implementation of CG-BiO, we still stick with the stepsizes $\alpha_k = 2/(k+2)$ as suggested in theorem (3.7). In addition, in each iteration, we need to solve the following sub-problem:

$$\begin{aligned} \min_{s\in\mathbb{R}^n} \quad & \langle \nabla f_{\text{upper}}(\beta_k), s\rangle \\ \text{s.t.} \quad & \|s\|_1 \le \lambda, \\ & \langle \nabla f_{\text{lower}}(\beta_k), s-\beta_k\rangle \le \beta_k - f_{\text{lower}}(\beta_k). \end{aligned} \tag{10}$$

As suggested by Jiang et al. [9], the above problem can be reformulated as a mere linear programming problem by introducing the following variable transformation $s := s_+ - s_-$, where $s_+, s_- \in \mathbb{R}^n_+$. In that case, the problem is turned into:

$$\begin{aligned} \min_{s\in\mathbb{R}^n_+} \quad & \langle \nabla f_{\text{upper}}(\beta_k), s_+ - s_-\rangle \\ \text{s.t.} \quad & \langle \mathbf{1}_n, s_+\rangle + \langle \mathbf{1}_n, s_-\rangle \le \lambda, \\ & \langle \nabla f_{\text{lower}}(\beta_k), s_+ - s_- - \beta_k\rangle \le \beta_k - f_{\text{lower}}(\beta_k). \end{aligned} \tag{11}$$

6.3.5 Implementation of CG-BiO, MNG, BiG-SAM, and a-IRG method.

For these methods, we follow the same setup as recommended by Jiang et al. [9]. Below, we show the details of the following methods: MNG, BiG-SAM, and a-IRG in algorithms (9), (10), and (11).

Algorithm 9: [Beck and Sabach [1]] Minimal norm gradient (MNG) method.

Data: hyperparameter $M \geq L_{\text{lower}}$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in \mathbb{R}^n$;

for $k = 0, \ldots, K-1$ **do**

Compute $x_{k+1} \leftarrow \arg\min_{x \in Q_k \cap W_k} f_{\text{upper}}(x)$, where

$$Q_k := \{z \in \mathbb{R}^n | \langle G_M(x_k), x_k - z \rangle \geq \frac{3}{4M} \|G_M(x_k)\|^2\},$$

$$W_k := \{z \in \mathbb{R}^n | \langle f_{\text{upper}}(x_k), z - x_k \rangle \geq 0\},$$

$$G_M(x) := M\left(x - \text{Proj}_S\left(x - \frac{1}{M}\nabla f_{\text{lower}}(x)\right)\right).$$

Algorithm 10: [Sabach and Shtern [15]] Bilevel gradient sequential averaging (BiG-SAM) method.

Data: $\eta_1 \leq 2/L_{\text{upper}}, \eta_2 \leq 1/L_{\text{lower}}, \gamma > 0, \{\alpha_k\}_k = \{\min\{\gamma/k, 1\}\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in \mathbb{R}^n$ **for** $k = 0, \ldots, K-1$ **do**

Compute $y_{k+1} \leftarrow \text{Proj}_S\left(x_k - \eta_1 \nabla f_{\text{lower}}(x_k)\right)$ Compute $z_{k+1} \leftarrow x_k - \eta_2 \nabla f_{\text{upper}}(x_k)$;

Compute $x_{k+1} \leftarrow \alpha_{k+1} z_{k+1} + (1 - \alpha_{k+1}) y_{k+1}$

Algorithm 11: [Kaushik and Yousefian [10]] Averaging iteratively regularized gradient (a-IRG) method.

Data: stepsizes $\{\gamma_k\}_k$, regularization parameters $\{\eta_k\}_k$

Result: sequence $\{x_k\}_k$

Initialize $x_0 \in \mathbb{R}^n$;

for $k = 0, \ldots, K-1$ **do**

Compute $x_{k+1} \leftarrow \text{Proj}_S\left(x_k - \gamma_k\left(\nabla f_{\text{lower}}(x_k) + \eta_k \nabla f_{\text{upper}}(x_k)\right)\right)$;

6.4 Performance Comparison

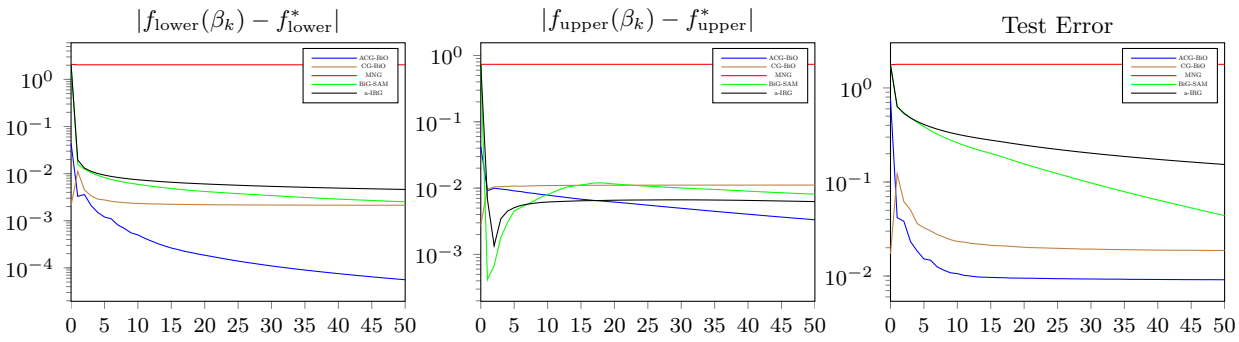

Figure a Figure b Figure c

From figure a and figure b, it can be seen that ACG-BiO converges faster than other methods despite some overshoot at iteration 5000th. As mentioned in section (2), CG-BiO method does show some problems in terms of convergence as after running 50,000 iterations, the optimality gap of f_{lower} of CG-BiO method seems to stable at around 0.001 rather than keeping decreasing as ACG-BiO method behaves. Turning to figure c, we observe that ACG-BiO method achieves the smallest test error as compared to the others. Moreover, it should be noted that the convergence of both f_{lower} and f_{upper} as $K \to \infty$ can be foreseen by the third claim in theorem (3.7). Specifically, the fact that our f_{lower} in this example satisfies assumption (2) relies on the following result , which is originated from theorem 3.1 proven by Li [11].

Proposition 6.7. *Let $g : \mathbb{R}^n \to \mathbb{R}$ be a convex polynomial function with degree d and P be a polyhedron in $\mathbb{R}^n$. Let $f := g + I_P$, then there exists $\tau > 0$ such that*

$$\operatorname{dist}\left(x, f^{-1}\left((-\infty, 0]\right)\right) \leq \tau\left([f(x)]_+ + [f(x)]_+^{[(d-1)^n+1]^{-1}}\right),$$

where $[a] = \max\{a, 0\}$.

Lemma 6.8. *Function f_{lower} over feasible region S_λ defined in problem (3) satisfies assumption (2).*

Proof. By noting S_λ defined in problem (3) is a polytope, considering function

$$\begin{aligned} g : \mathbb{R}^d &\to \mathbb{R} \\ x &\to \frac{1}{\max_{z \in S} f_{\text{lower}}(z)} \left(f_{\text{lower}}(x) - f^*_{\text{lower}}\right), \end{aligned}$$

and applying the proposition (6.7), we have

$$\operatorname{dist}(x, \chi_1^*) \leq \tau\left(g(x) + (g(x))^{\frac{1}{2}}\right), \forall x \in S.$$

In addition, over $S_\lambda, 0 \leq g(x) \leq 1$, and thus, $g(x) \leq (g(x))^{\frac{1}{2}}, \forall x \in S$. Hence, we have

$$\operatorname{dist}(x, \chi_1^*) \leq 2\tau(g(x))^{\frac{1}{2}}, \forall x \in S$$

$$\iff \frac{(\max_{z \in S} f_{\text{lower}}(z))^{\frac{1}{2}}}{2\tau} \operatorname{dist}(x, \chi_1^*) \leq \left(f_{\text{lower}}(x) - f^*_{\text{lower}}\right)^{\frac{1}{2}}, \forall x \in S$$

Method	Time (seconds)
ACG-BiO	2365.379
CG-BiO	2718.7925
MNG	2766.2252
BiG-SAM	30.7648
a-IRG	28.0632

Table 1: Time elapsed on running 50,000 iterations of ACG-BiO, CG-BiO, MNG, BiG-SAM, a-IRG.

$$\iff \frac{\max_{z\in S} f_{\text{lower}}(z)/\left(2\tau^2\right)}{2}\left(\text{dist}(x,\chi_1^*)\right)^2 \leq f_{\text{lower}}(x) - f_{\text{lower}}^*, \forall x \in S.$$

□

It should be noted that while ACG-BiO only takes us around 2365.379 seconds to complete 50,000 iterations, CG-BiO method and MNG method spend up to 2718.7925 seconds and 2766.2252 seconds doing the same thing as shown in table (1). As expected from the short cut of projection step in BiG-SAM method and a-IRG method, these two methods only take 30.7648 seconds and 28.0632 methods to run over 50,000 iterations. Nevertheless, three methods MNG, BiG-SAM and a-IRG show poor convergence results.

7 Conclusion

In this paper, we proposed the ACG-BiO method and its variants which relax either the minimisation oracle or the assumption of bounded feasible region to solve simple bilevel optimization problem with convex and L-smooth objective funtions over convex and compact feasible region. Specifically, we proved ACG-BiO method and its variant with the application of LPsep Braun et al. [3] can achieve the convergence rate of $O(\frac{1}{K})$ over K iterations for both objective functions and its variant with possible unbounded feasible region converges with a rate of $O\left(\frac{1}{K^p}\right)$ for any $p \in (0,1)$. The numerical results also showed the superior performance of our method compared to existing algorithms.

References

[1] A. Beck and S. Sabach. A first order method for finding minimal norm-like solutions of convex optimization problems. *Mathematical Programming*, 147(1-2):25–46, 2014. doi: 10.1007/s10107-013-0708-2.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. doi: 10.1137/080716542.

[3] G. Braun, S. Pokutta, and D. Zink. Lazifying conditional gradient algorithms. *Journal of Machine Learning Research*, 20(71):1–42, 2019.

[4] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3 (1-2):95–110, 1956.

[5] L. L. Gao, J. Ye, H. Yin, S. Zeng, and J. Zhang. Value function based difference-of-convex algorithm for bilevel hyperparameter selection problems. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 7164–7182. PMLR, 17–23 Jul 2022.

[6] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

[7] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, Mar. 2014.

[8] M. Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 427–435, 2013.

[9] R. Jiang, N. Abolfazli, A. Mokhtari, and E. Y. Hamedani. A conditional gradient-based method for simple bilevel optimization with convex lower-level problem, 2022.

[10] H. D. Kaushik and F. Yousefian. A method with convergence rates for optimization problems with variational inequality constraints. *SIAM Journal on Optimization*, 31(3):2171–2198, 2021. doi: 10.1137/20M1357378.

[11] G. Li. Global error bounds for piecewise convex polynomials. *Mathematical Programming*, pages 37–64, 2013. doi: 10.1007/s10107-011-0481-z.

[12] R. Liu, J. Gao, J. Zhang, D. Meng, and Z. Lin. Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond, 2021.

[13] Z.-Q. Luo and J.-S. Pang. Error bounds for analytic systems and their applications. *Mathematical Programming*, 67(1):1–28, 1994. doi: 10.1007/BF01582210.

[14] B. Rozemberczki, P. Scherer, Y. He, G. Panagopoulos, A. Riedel, M. Astefanoaei, O. Kiss, F. Beres, G. López, N. Collignon, and R. Sarkar. Pytorch geometric temporal: Spatiotemporal signal processing with neural machine learning models. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, CIKM '21, page 4564–4573. Association for Computing Machinery, 2021. ISBN 9781450384469. doi: 10.1145/3459637.3482014.

[15] S. Sabach and S. Shtern. A first order method for solving convex bilevel optimization problems. *SIAM Journal on Optimization*, 27(2):640–660, 2017. doi: 10.1137/16M105592X.

[16] S. J. Wright and B. Recht. *Foundations of Smooth Optimization*, page 15–25. Cambridge University Press, 2022. doi: 10.1017/9781009004282.003.