THE UNIVERSITY OF SYDNEY

THE UNIVERSITY OF SYDNEY BUSINESS SCHOOL

DISCIPLINE OF BUSINESS ANALYTICS

HONOURS THESIS

Projection-free methods for solving smooth convex bilevel optimisation problems

---

Author:

Khanh-Hung Giang-Tran

Supervisor:

Nam Ho-Nguyen

October 24, 2023

Statement of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at the University of Sydney or at any other educational institution, except where due acknowledgement is made in the thesis.

Any contribution made to the research by others, with whom I have worked at the University of Sydney or elsewhere, is explicitly acknowledged in this thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the projects design and conception or in style, presentation and linguistic expression is acknowledged.

Khanh-Hung Giang-Tran

October 2023

Acknowledgements

First, I would like to thank my supervisor and mentor, Dr. Nam Ho-Nguyen. Nam has helped me in formulating and refining the problem, as well as reviewing my work. Your professionalism and diligence will always be something I will look back upon. Without your guidance and support, this project would not have been possible.

I would also like to acknowledge the members of the Business School and Discipline of Business Analytics. In particular, I would like to thank Associated Professor Jie Yin and Dr. Nam Ho-Nguyen for coordinating the Business Analytics Honours program and the Business Honours Director, Dr Reuben Segara. Additionally, I would like to thank my lecturers Professor Artem Prokhorov, Associated Professor Dmytro Matsypura, and Associated Professor Jie Yin. The content presented in your lectures was challenging; however, they were enjoyable and rewarding, allowing me to develop new skills which I have no doubt will help me in my future career.

Also, I would like to thank my fellow students and friends in the Business Analytics honours program: Mike, Tuantuo. Your advice, support and presence made this experience one to remember.

Lastly, I would like to thank my family and friends, especially my parents, who have always supported me in my endeavours.

Abstract

Optimisation is a critical analytical technique used for quantitative decision-making in real-world problems. In practice, many situations call for decision-making in a hierarchical setting, called the *bilevel optimisation problem*. In this type of problem, we are provided with an *inner-level objective* and a *base domain*. Interestingly, the inner-level objective may possess multiple optimal solutions over the base domain. Hence, to select one of these solutions, one may consider a secondary objective, referred to as the *outer-level objective*, and minimise this objective over the optimal set of the inner-level objective. Typically, it is impractical to assume the optimal set of the inner-level objective over the base domain admits a simple characterisation or is explicitly given. Hence, solving bilevel problems requires optimisation techniques designed to account for the generally unknown feasible set. First-order methods have gained prominence due to their ability to efficiently solve high-dimensional optimisation problems. These techniques, for example, projected gradient descent, typically rely on a projection oracle to handle constraints. However, certain problems exhibit structure, which makes linear optimisation oracles much more efficient to implement, thus giving rise to conditional gradient methods. While various projection-based methods have been devised to solve the bilevel optimisation problems, currently, there is little work on projection-free methods for bilevel optimisation. Thus, this thesis examines various first-order projection-free schemes for solving bilevel problems which employ linear optimisation oracles.

Projection-free algorithms typically require a bounded domain, which restricts their application in practice. Using a truncation technique, we first provide an extension of the conditional gradient method to unbounded domains for single-level optimisation problems. Using this as a subroutine, we then suggest three first-order projection-free approaches designed for bilevel problems with smooth convex inner- and outer-level objectives and a closed convex base domain. Previously, to the best of our knowledge, projection-free algorithms for bilevel problems require linear minimisation oracles over complicated sets. In contrast, our approaches only require a linear optimisation oracle over

the base domain, or an appropriately truncated version of it.

We provide convergence guarantees for each of our methods, highlighting the trade-off between inner- and outer-level convergence rates, as well as the effect of truncation on unbounded domains. We demonstrate these performances through three numerical experiments in portfolio optimisation, low-rank matrix completion, and linear inverse problems.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem description

In this thesis, we are interested in solving the following convex bilevel optimisation problem:

$$\begin{aligned} &\min_{x \in X} \quad f(x) \\ &\text{s.t.} \quad x \in X_{\text{opt}}, \quad \text{where} \quad X_{\text{opt}} := \arg\min_{z \in X} g(z). \end{aligned} \tag{1.1}$$

We refer to g as the *inner-level objective function* and X as the *base domain*, and when g and X are convex, X_{opt} is also convex. We also assume f, the *outer-level objective function*, is convex, which makes (1.1) a convex problem. We define the inner- and outer-level optimal values as follows:

$$g_{\text{opt}} := \min_{x \in X} g(x), \quad f_{\text{opt}} := \min_{x \in X_{\text{opt}}} f(x). \tag{1.2}$$

In addition, we also assume throughout this thesis that problem (1.1) is solvable. We refer to any point $x \in X$ as an *inner-level feasible* point and any point $x^* \in X_{\text{opt}}$ as an *inner-level optimal* or *outer-level feasible* point.

Problem (1.1) is trivial if X_{opt} is a singleton, i.e., the inner-level objective function has a unique

solution over X. However, multiple optimal solutions can arise in many practical applications, and the bilevel problem (1.1) may be used to select a solution satisfying auxiliary desirable properties. Below, we discuss three applications of convex bilevel optimisation problems.

Example 1.1. Consider a least squares linear regression problem where the data are feature-output pairs (a_i, b_i) for $i \in [n]$. We stack the feature vectors into a matrix $A \in \mathsf{R}^{n\times d}$ and outputs into a vector $b \in \mathsf{R}^n$. In this case, $g(x) := \frac{1}{2}\|Ax-b\|_2^2$ measures the sum of squared errors between outputs b_i and estimates $a_i^\top x$, where x is the coefficient vector. When g is not strongly convex, which can happen when the number of features d is larger than the number of data points n, multiple optimal solutions may exist. Hence, a second objective f should be used to select one such solution. One application of this can be seen in the so-called *minimal norm problem*, which has a closed form solution of $x_{\text{opt}} = A^\dagger b$, where $A^\dagger$ is the pseudo-inverse of A, when the chosen outer-level objective is $f(x) := \frac{1}{2}\|x\|_2^2$. ■

Example 1.2. Consider the Markowitz portfolio optimisation problem [28], in which we are provided with n assets numbered $1, \ldots, n$, and an n-vector μ, whose ith entry is the expected return of the ith asset, and a positive semi-definite matrix Σ, whose (i, j) entry is the covariance of returns of ith and jth assets. The goal of this problem is to minimise the portfolio variance subject to the condition that the expected return should be at least a minimum threshold $r_0 > 0$, i.e., the problem is as follows:

$$\begin{aligned} \min_x \quad & \frac{1}{2}x^T\Sigma x \\ \text{s.t} \quad & \mu^T x \geq r_0, \quad \mathbf{1}^T x = 1, \quad x \geq 0, \end{aligned}$$

where variable x_i represents the allocation of wealth to the asset i. Multiple optimal solutions may exist when the covariance matrix is not full rank. In such case, Beck and Sabach [4, Section 5.1] considered the outer-level objective $f(x) := \frac{1}{2}\|x - a\|_2^2$, where $a = \mathbf{1}/n$, which is chosen to obtain a diverse portfolio. We will revisit this example in Chapter 7. ■

Example 1.3. The low-rank matrix completion problem seeks to find a low-rank $n \times p$ matrix X to approximate a subset of observed entries $M_{i,j}$, for $(i, j) \in \Omega \subset [n] \times [p]$. That is, the objective is to

minimise

$$g(X) := \frac{1}{2} \sum_{(i,j)\in\Omega} (X_{i,j} - M_{i,j})^2,$$

over $X \in \mathsf{R}^{n\times p}$ such that $\text{rank}(X) \leq \delta$ for some small $\delta > 0$. Due to the discrete nature of the rank function, Fazel [15] replaced the rank function with its convex envelope, which is the nuclear norm $\|\cdot\|_*$, defined as the sum of singular values. Thus, the low-rank matrix completion problem is as follows:

$$\begin{aligned} \min_{X\in\mathsf{R}^{n\times p}} \quad & g(X) := \frac{1}{2} \sum_{(i,j)\in\Omega} (X_{i,j} - M_{i,j})^2 \\ \text{s.t} \quad & \|X\|_* \leq \delta. \end{aligned}$$

The objective g is not strictly convex; therefore, it is possible to have multiple minimisers, and a second objective f should be used to select one such solution. For example, in the movie score prediction problem, where $M_{i,j}$ is the score of the jth movie from the ith customer, one can consider f to be the sum of the variances of scores within each movie. We will revisit this example in Chapter 7. ■

1.2 Main challenges

Solving the bilevel problem (1.1) is more complicated than a standard single-level problem. There are two primary challenges in solving (1.1). First, we do not have an explicit representation of the optimal set X_{opt} in general, which makes it intractable to apply the usual operations such as projection or linear optimisation on X_{opt}, thus preventing the use of projected gradient descent or conditional gradient methods. Thus, we alternatively consider the *value function formulation* of problem (1.1):

$$\begin{aligned} \min_{x\in X} \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq g_{\text{opt}}. \end{aligned} \tag{1.3}$$

However, the second challenge arises because, by the definition of g_{opt}, there exists no $x \in X$ such that $g(x) < g_{\text{opt}}$. Hence, problem (1.3) does not satisfy the Slater constraint qualification, which means that the Lagrangian dual of (1.3) may not be solvable. One may attempt to enforce Slater's condition by adding a small $\epsilon_g > 0$ to the right-hand side of the constraint, i.e., $g(x) \leq g_{\text{opt}} + \epsilon_g$, but this approach does not solve the actual problem (1.3), and may introduce numerical instability [24, Appendix D].

1.3 Literature review

Several schemes have been devised to tackle problem (1.1). These methods can be grouped into three categories: regularisation, sublevel set, and sequential averaging.

1.3.1 Regularisation approach

This approach combines inner- and outer-level objectives via Tikhonov regularisation, i.e., we optimise $\sigma f(x) + g(x)$, where $\sigma > 0$ is the so-called regularisation parameter. Under some mild conditions, Friedlander and Tseng [19] showed that for a sufficiently small $\sigma > 0$, the optimal set of the regularised problem $\arg\min_{x \in X}\{\sigma f(x) + g(x)\}$ is the same as that of (1.1). Friedlander and Tseng [19] showed that existence of such $\sigma > 0$ is equivalent to the Lagrangian dual of (1.3) being solvable. However, the value σ is a priori unknown. As an alternative, if we consider a positive sequence $\{\sigma_t\}_{t \geq 0}$ converging to 0 and define

$$s_t \in \arg\min_{x \in X}\{\sigma_t f(x) + g(x)\},$$

for each $t \geq 0$, then it is known that any accumulation point of $\{s_t\}_{t \geq 0}$ is a solution of (1.1). That said, finding s_t for each t is expensive.

A more efficient strategy is to employ only cheap first-order updates each time we update t. To the best of our knowledge, this idea dates back to Cabot [6], who proposed a proximal point-type algorithm to update the solutions in case $X := \mathsf{R}^n$. Cabot [6] showed that the iterates converge

asymptotically to the optimal solution set. Dutta and Pandit [14] extended the work to a general closed convex X. However, no convergence rates were provided by Cabot [6], Dutta and Pandit [14].

Since proximal point updates also involve solving expensive optimisation problems each iteration, Solodov [34] proposed the iterative regularised projected gradient (`IR-PG`) method, where only a projected gradient step is taken each iteration t; we describe this in detail in Section 5.1. Solodov [34] guaranteed asymptotic convergence when f and g are smooth, under the appropriate selection of the regularisation parameters $\{\sigma_t\}_{t\geq 0}$, but again no rates were provided. When f and g are possibly nonsmooth, Helou and Simões [22] proposed a variation of the ϵ-subgradient method with asymptotic convergence under Lipschitz continuity of f and g.

By choosing the relevant parameters appropriately, Amini and Yousefian [1] provided an inner-level objective convergence rate of $O\left(1/T^{1/2-b}\right)$ for any fixed $b \in (0, 1/2)$ when f and g are nonsmooth, but X is compact and f is strongly convex. Kaushik and Yousefian [25] provided an analysis that removed the strongly convex assumption on f and admits convergence rates for both inner- and outer-level objectives of $O\left(1/T^{b}\right)$ and $O\left(1/T^{1/2-b}\right)$ respectively, for any $b \in (0, 1/2)$. (They also replace the inner-level objective with a variational inequality.) Malitsky [27] studied a version of Tseng's accelerated gradient method [35] with a convergence rate of $o(1/T)$ for the inner-level objective.

Recently, Shen et al. [33] proposed two primal-dual-type algorithms in which σ_t is adaptively adjusted based on how close $g(x_t)$ is to g_{opt}. The first algorithm works with only convexity and Lipschitz continuity assumptions and converges with the rate of $O\left(1/T^{1/3}\right)$ for both inner- and outer-level objectives. The second algorithm, which utilises more structural information on the objective functions, including additional smoothness and strong convexity assumptions, converges with the rate of $O\left(1/T^{1/2}\right)$ for both inner- and outer-level objectives. Nevertheless, the algorithms of Shen et al. [33] demand a tolerance parameter for the inner-level objective to be set in advance; thus, do not enjoy asymptotic convergence guarantees.

1.3.2 Sublevel set approach

Another strategy is to replace the optimal sublevel set $X_{\text{opt}} = \{x \in X \mid g(x) \leq g_{\text{opt}}\}$ in (1.1) with an approximation. For instance, the minimal norm gradient (MNG) method [4] constructs an outer approximation of X_{opt} with two half-spaces and minimises f, which is assumed to be strongly convex and smooth, over this approximation. The MNG method converges with the rate of $O\left(1/T^{1/2}\right)$ for the inner-level objective when g is smooth convex, but no rates are provided for the outer-level objective.

Jiang et al. [24] introduced the conditional gradient-based bilevel optimisation (`CG-BiO`) method which approximates X_{opt} by replacing $g(x)$ with a linear approximation. Jiang et al. [24] provided the rate of $O(1/T)$ for both inner- and outer-level objectives when f, g are smooth and X is compact. Cao et al. [7] extended this to the stochastic setting, when $f(x) := \mathsf{E}_\theta[\tilde{f}(x,\theta)]$ and $g(x) := \mathsf{E}_\xi[\tilde{g}(x,\xi)]$ where θ, ξ are independent random variables. When θ and ξ have finite support, Cao et al. [7] showed a convergence rate of $O(\log(T)/T)$ for both objectives. Otherwise, when θ and ξ satisfy a sub-Gaussian property, the rate is $O\left(1/T^{1/2}\right)$. Both Jiang et al. [24] and Cao et al. [7] required the predetermination of a tolerance parameter $\epsilon_g > 0$, and their convergence analysis only ensures convergence to an $\frac{\epsilon_g}{2}$-suboptimal solution for the inner-level objective.

Instead of approximating X_{opt}, Doron and Shtern [12] provided an alternative formulation of (1.1) which relies on sublevel sets of the *outer-level* objective f. Based on this, they developed a method called the iterative approximation and level-set expansion (`ITALEX`) method [12], which at each iteration t performs either two proximal gradient or two generalised conditional gradient operations where one is over X, and the other is over a sublevel set $\{x \mid f(x) \leq \alpha_t\}$ of f. The sublevel set α_t of f is then updated. Doron and Shtern [12] showed the convergence rates of $O(1/T)$ and $O\left(1/T^{1/2}\right)$ for the inner- and outer-level objectives, respectively, when g is a composite function, i.e., a sum of smooth convex and nonsmooth convex functions, and f satisfies an error bound condition.

1.3.3 Sequential averaging approach.

In this approach, the update rule for x_{t+1} is a weighted average of two mappings computed from x_t. For instance, the bilevel gradient sequential averaging method (BiG-SAM) [31] takes a convex combination between a proximal gradient step with respect to the inner-level objective and a gradient step with respect to the outer function from the current iterate. Sabach and Shtern [31] showed asymptotic convergence for f without a rate, and convergence at rate $O(1/T)$ for g, when f is smooth and strongly convex, and g is a composite function. Shehu et al. [32] presented the inertial bilevel gradient sequential averaging method (iBiG-SAM), a variation of the BiG-SAM method with an inertial extrapolation step. Although Shehu et al. [32] showed the asymptotic convergence of the iBiG-SAM method without rates for both inner- and outer-level objectives under the same assumptions of the BiG-SAM method, several numerical examples conducted in the study indicated that the iBiG-SAM method outperformed the BiG-SAM method in those experiments.

Merchav and Sabach [29] proposed the bi-sub-gradient (`Bi-SG`) method for the case f is nonsmooth and g is a composite function. At each iteration, a proximal gradient step for g is calculated, followed by a subgradient descent step for f. Merchav and Sabach [29] showed the convergence rates of $O(1/T^{\alpha})$ and $O(1/T^{1-\alpha})$ for the inner- and outer-level objectives, respectively, with $\alpha \in (1/2, 1)$ when f satisfies a quasi-Lipschitz property, and the rate of $O\left(e^{-c(\beta/4)T^{1-\alpha}}\right)$ for the inner-level objective, where $c, \beta > 0$ are some relevant parameters, when f is a composite function with strongly convex smooth part, and the nonsmooth part of g is Lipschitz continuous.

1.4 Contributions

It is known that for certain convex base domains X, linear oracles can be implemented more efficiently than projection operations. For example, in a low-rank matrix completion problem, when X is a nuclear norm ball of the set of $n \times p$ matrices, projection onto X requires a complete singular value decomposition of a matrix and projection of the vector of singular values onto the probability simplex, i.e., $\{x \in \mathsf{R}^{\min\{n,p\}} \mid x \geq 0, \mathbf{1}^{\top}x = 1\}$ [3, Section 7.3.2], whereas the linear oracle only requires computing the maximum singular value, and the corresponding left and right

singular vectors [23, Section 4.2].

In light of this, in this thesis, we present three iterative methods for solving (1.1) that require only linear optimisation oracles over the base domain X at each iteration, or appropriate truncations of it when X is unbounded, under the assumptions that X is closed and f, g are smooth convex functions. Our methods are derived by studying the equivalent formulation (1.3). Our contributions are summarised as follows:

- In Chapter 3, due to the relaxation of the boundedness of domain X, which is typically assumed by projection-free schemes, we present the unbounded conditional gradient (`UCG`) method, which is an extension of the `CG` method [17] for unbounded domains. This serves as a fundamental tool that we will use in our algorithms for solving problem (1.1). By proposing a truncation scheme, we show asymptotic convergence to the optimal value of the objective function with the rate of $O\left(d_T^2/T\right)$ for any non-decreasing, divergent sequence $\{d_t\}_{t\geq 0}$ such that $d_t = o\left(t^{1/2}\right)$, where $\{d_t\}_{t\geq 0}$ are parameters which control the degree of truncation at each iteration. In case X is bounded, $\{d_t\}_{t\geq 0}$ can be taken to be a constant sequence.

- In Chapter 4, we propose the sublevel linearising conditional gradient (`SL-CG`) method, which is an improvement upon the `CG-BiO` method [24]. By adopting a sequence of non-increasing approximations $\{g_t\}_{t\geq 0}$ converging to g_{opt}, we show asymptotic convergence for the algorithm in Section 4.2. Under the assumption that the approximation sequence converges at the rate of $O\left(d_T^2/T\right)$, we establish the convergence rate of $O\left(d_T^2/T\right)$ for both inner- and outer-level objectives for any non-decreasing, divergent sequence $\{d_t\}_{t\geq 0}$ such that $d_t = o\left(t^{1/2}\right)$ when X is unbounded. When X is bounded, the required rate for the approximations and the convergence rates for both inner- and outer-level objectives are $O\left(1/T\right)$.

- In Chapter 5, we propose the iteratively regularised conditional gradient (`IR-CG`) method, which uses the regularisation approach to solve (1.1). Unlike other previous regularisation approaches, we utilise a novel averaging scheme that arises due to the analysis of the conditional gradient updates. We provide conditions on the relevant parameters which ensure asymptotic convergence, as well as convergence rates of $O(1/T^p)$ and $O\left(d_T^2/T^{1-p}\right)$ for the

inner- and outer-level objectives, respectively, for any $p \in (0,1)$, and for any non-decreasing, divergent sequence $\{d_t\}_{t\geq 0}$ such that $d_t = o\left(t^{(1-p)/2}\right)$.

- In Chapter 6, we propose the primal-dual conditional gradient (`PD-CG`) method. This method is an extension of a conditional gradient-type algorithm for solving single-level convex optimisation problems with functional constraints proposed by Lan et al. [26] to solve the bilevel problem (1.1). As we will discuss in Section 2.4, under some mild conditions, strong Lagrangian duality for (1.3) is guaranteed to hold, yet the dual problem is not guaranteed to be solvable. We provide a unified analysis that yields convergence guarantees when the Lagrangian dual is not solvable and also yields *improved* guarantees when the dual is solvable. Our algorithm does not need knowledge of the optimal dual solution to be implemented. Without an optimal dual solution, we prove the convergence rates of $O\left(1/T^{(1-p)/2}\right)$ and $O\left(\max\left\{1/T^{1-p}, d_T^4/T^p\right\}\right)$ for inner- and outer-level objectives, respectively, for any $p \in (0,1)$, and any non-decreasing, divergent sequence $\{d_t\}_{t\geq 0}$ such that $d_t = o\left(t^{p/4}\right)$. When the dual problem is solvable, the rate for the inner-level objective improves to $O\left(\max\left\{1/T^{1-p}, d_T^2/T^{1/2}\right\}\right)$.

- In Chapter 7, we investigate the numerical performance of our new algorithms via three numerical experiments based on Examples 1.1 to 1.3, where we compare the performance of the proposed methods to that of some existing methods including `IR-PG` [34], `Bi-SG` [29], `ITALEX` [12] with projection-free customisation, `CG-BiO` [24].

We note that the `CG-BiO` method of Jiang et al. [24] as well as the projection-free customisation of the `ITALEX` method of Doron and Shtern [12] also utilises linear optimisation oracles to solve (1.1) and only work under the boundedness of the base domain X. In case X is bounded, at each iteration, two of our methods, which are the `IR-CG` and `PD-CG` methods, only require linear oracles over the base domain X. In contrast, `CG-BiO` and our variant `SL-CG` require at each iteration a linear oracle over $X \cap H_t$ where H_t is some half-space, which can be significantly more complicated than a linear oracle over X. (We provide examples for this claim in Chapter 7) `ITALEX` requires a linear oracle over a sublevel set $\{x : f(x) \leq \alpha_t\}$ in addition to one over X. While some functions f admit simple linear oracles over their sublevel sets, another assumption required for convergence

of `ITALEX` is that the sublevel sets of f are bounded, whereas our algorithms apply to any smooth convex f. We provide an example of f with unbounded sublevel sets in Chapter 7.

Chapter 2

Preliminaries

In this chapter, we first list the notations utilised for this thesis in Section 2.1. In Section 2.2, we describe one set of assumptions involving smoothness and convexity and their implications. In Section 2.3, we describe how *super-optimal* solutions of (1.3) may often be encountered in bilevel optimisation. We also provide results on the convergence of iterates under additional assumptions, and bounds on the degree of super-optimality in these situations. In Section 2.4, we discuss a constraint qualification that ensures the solvability of the dual problem. This is used in the convergence analysis of one of our methods discussed in Chapter 6. In Section 2.5, we review the conditional gradient (CG) method [17], which provides a crucial foundation for our contributions.

2.1 Mathematical notation

In this thesis, we use R to denote the real numbers, R^n to denote the set of all real vectors with n components, R^n_+ to denote the set of all real vectors with n non-negative components, R^n_{++} to denote the set of all real vectors with n positive components, $\mathsf{R}^{n\times m}$ to denote the set of all real matrices with n rows and m columns.

The set $\{1, 2, \ldots, m\}$ for a positive integer m is denoted as $[m]$, and for a real number r, we denote $[r]_+ := \max\{r, 0\}$. For a matrix A, we denote the largest singular value or the spectral

norm as $\sigma_{\max}(A)$. Given two vectors a, b in R^n, we define the dot product between them as $a^\top b = \sum_{i\in[n]} a_i b_i$. We let $\mathbf{1}$ be the vector with entries equal to one, whose dimension depends on the context. Similarly, 0 will be used flexibly as either a number or a vector or a matrix, which depends on the context. We will use e_i to denote the vector in R^n whose entries are zero except for the ith entry whose value is 1.

Furthermore, we use the dot product as the inner product in R^n, and given an arbitrary norm $\|\cdot\|$ on R^n, the dual norm $\|\cdot\|_*$ is defined as follows:

$$\|x\|_* := \sup_{\|y\|\le 1} x^\top y, \quad \forall x \in \mathsf{R}^n.$$

Given a closed convex set $C \subseteq \mathsf{R}^n$ and a norm $\|\cdot\|$, the projection of x onto C is denoted by $\mathrm{Proj}_C(x) = \arg\min_{y\in C} \|y - x\|$, and the distance between x and C is denoted as $\mathrm{Dist}(x, C) = \min_{y\in C} \|y - x\|$. Given a bounded set C, we can define its diameter with respect to a norm $\|\cdot\|$ as $D_C := \sup\{\|x - y\| \mid x, y \in C\}$. Given a norm $\|\cdot\|$ on R^n, $a \in \mathsf{R}^n$, and $r > 0$, we use $\mathbf{B}_{\|\cdot\|}(a, r)$ to denote the set $\{x \in \mathsf{R}^n \mid \|x - a\| \le a\}$. Given a set C, we denote its convex hull, which is the set containing all convex combinations of points in C, as $\mathrm{Conv}\, C$.

Given two sequences $\{x_t, y_t\}_{t\ge 0}$ in R, we write $x_t \le O(y_t)$ if there exists $M > 0$ and a non-negative integer t_M such that for $t \ge t_M$, $x_t \le M y_t$. Additionally, we write $x_t = o(y_t)$ if for any $\epsilon > 0$, there exists a non-negative integer t_ϵ such that for any $t \ge t_\epsilon$, we have $|x_t| \le \epsilon y_t$.

2.2 Assumptions on smoothness, convexity, and implications

Before discussing Assumption 1 on problem (1.1), we introduce the definition of the smoothness of a function as stated below.

Definition 2.1. Suppose C is a subset of R^n, and $\|\cdot\|$ is a norm on R^n. Then a function h is called L_h-*smooth* on C for some $L_h > 0$ if it is continuously differentiable on an open neighbourhood of C and for any $x, y \in C$, we have $\|\nabla h(x) - \nabla h(y)\|_* \le L_h \|x - y\|$.

Assumption 1. Let $\|\cdot\|$ be an arbitrary norm on R^n. We consider the following conditions on f, g and X:

(a) $X \subseteq \mathsf{R}^n$ is a convex and closed set.

(b) f is convex, L_f-smooth on X for some $L_f > 0$.

(c) g is convex, L_g-smooth on X for some $L_g > 0$.

Remark 2.2.1. From now on, unless stated otherwise, any computation or definition involving the knowledge of norm (e.g., the distance from a set, the magnitude of the vector, the diameter of a set) will utilise norm $\|\cdot\|$ given in Assumption 1. ■

Before discussing frequently used results following from Assumption 1, we state without proof the consequence of smoothness, which is crucial to obtain important implications of Assumption 1 discussed later in this section.

Lemma 2.1 ([38, Lemma 2.2])**.** *Let h be an L_h-smooth function on X. Then we have*

$$|h(y) - h(x) - \nabla h(x)^\top (y - x)| \leq \frac{L_h}{2}\|y - x\|^2, \quad \forall x, y \in X$$

Conditional gradient-type algorithms typically have an update step of the form

$$x_{t+1} = x_t + \alpha_t(v_t - x_t), \quad \alpha_t \in [0, 1], \quad v_t \in X, \tag{2.1}$$

where α_t and v_t are carefully chosen to ensure convergence [23, Algorithms 1,2,3,4]. The following lemma provides an inequality for this update, which is standard in smooth convex optimisation and will be utilised extensively in our analysis.

Lemma 2.2. *Suppose h is an L_h-smooth function on X. Then we have*

$$h(y) - h(z) \leq (1 - \alpha)(h(x) - h(z)) + \alpha\left(\nabla h(x)^T (v - x) + h(x) - h(z)\right) + \frac{L_h\|v - x\|^2}{2}\alpha^2,$$

for any $x, z, v \in X, \alpha \in [0, 1], y = x + \alpha(v - x)$.

Proof. By the convexity of X, we have $y = (1-\alpha)x + \alpha v \in X$. By Lemma 2.1, we have that

$$h(y) \le h(x) + \nabla h(x)^\top (y - x) + \frac{L_h}{2}\|y - x\|^2$$

$$\iff h(y) \le h(x) + \alpha \nabla h(x)^\top (v - x) + \frac{L_h \alpha^2}{2}\|v - x\|^2.$$

By subtracting both sides by $h(z)$, we have that

$$\begin{aligned} & h(y) - h(z) \le h(x) - h(z) + \alpha \nabla h(x)^\top (v - x) + \frac{L_h \alpha^2}{2}\|v - x\|^2 \\ \iff & h(y) - h(z) \le (1-\alpha)(h(x) - h(z)) + \alpha \left(\nabla h(x)^T (v - x) + h(x) - h(z)\right) + \frac{L_h \|v - x\|^2}{2}\alpha^2. \end{aligned}$$

□

Suppose the sequence $\{x_t\}_{t\ge 0}$ generated by (2.1) given a $x_0 \in X$. For any $z \in X$ and $t \ge 0$, if we denote

$$\begin{aligned} \Delta_t &:= h(x_t) - h(z), \quad \Delta_{t+1} := h(x_{t+1}) - h(z), \\ \eta_t &:= \alpha_t \left(\nabla h(x_t)^T (v_t - x_t) + h(x_t) - h(z)\right) + \frac{L_h D^2 \alpha_t^2}{2}, \end{aligned} \tag{2.2}$$

then Lemma 2.2 shows that we have the following recursion between the function values of consecutive iterates:

$$\Delta_{t+1} \le (1 - \alpha_t)\Delta_t + \eta_t, \tag{2.3}$$

for any $t \ge 0$. This recursion is the key to analyse the convergence of a conditional gradient-type algorithm with Δ_t, η_t depending on the algorithm rather than being restricted to the one given in (2.2). Due to our scheme to deal with unbounded base domains, (2.3) may not hold for any $t \ge 0$ in our analysis but only for $t \ge t_0$ for some sufficiently large $t_0 \ge 0$. Therefore, in the next lemma, we provide an upper bound on Δ_t only based on $\{\alpha_t\}_{t\ge 0}$ and Δ_{t_0}.

Lemma 2.3. *Let $\{\Delta_t, \eta_t, \alpha_t\}_{t\ge 0}$ be sequences such that (2.3) holds for $t \ge t_0$, and $\alpha_0 \in [0, 1]$,*

$\alpha_t \in [0,1)$ *for* $t \geq 1$. *Then we have that for* $t > t_0$,

$$\Delta_t \leq H_0 a_{t-1} + a_{t-1} \sum_{i \in [t]} \frac{\eta_{i-1}}{a_{i-1}}, \tag{2.4}$$

where

$$a_0 := 1, \quad a_t := \prod_{i \in [t]} (1 - \alpha_i), \quad \forall t \geq 1,$$

$$H_0 := \begin{cases} (1 - \alpha_{t_0}) \frac{\Delta_{t_0}}{a_{t_0}} - \sum_{i \in [t_0]} \frac{\eta_{i-1}}{a_{i-1}}, & t_0 \geq 1 \\ (1 - \alpha_0) \frac{\Delta_0}{a_0} \quad , & t_0 = 0. \end{cases}$$

Proof. Dividing both sides of (2.3) by a_t and noting that $a_t = (1 - \alpha_t) a_{t-1}$ for any $t \geq 1$, we obtain

$$\frac{\Delta_{t+1}}{a_t} \leq \frac{\Delta_t}{a_{t-1}} + \frac{\eta_t}{a_t}, \quad \forall t \geq t_0.$$

Hence, given $t \geq t_0 + 2$, we have that

$$\frac{\Delta_t}{a_{t-1}} - \frac{\Delta_{t_0+1}}{a_{t_0}} = \sum_{i=t_0+1}^{t-1} \left(\frac{\Delta_{i+1}}{a_i} - \frac{\Delta_i}{a_{i-1}} \right) \leq \sum_{i=t_0+1}^{t-1} \frac{\eta_i}{a_i}. \tag{2.5}$$

Using the fact that $\Delta_{t_0+1} \leq (1 - \alpha_{t_0}) \Delta_{t_0} + \eta_{t_0}$, we multiply both sides of (2.5) by a_{t-1} to obtain

$$\Delta_t \leq (1 - \alpha_{t_0}) a_{t-1} \frac{\Delta_{t_0}}{a_{t_0}} + a_{t-1} \frac{\eta_{t_0}}{a_{t_0}} + a_{t-1} \sum_{i=t_0+1}^{t-1} \frac{\eta_i}{a_i} = (1 - \alpha_{t_0}) a_{t-1} \frac{\Delta_{t_0}}{a_{t_0}} + a_{t-1} \sum_{i=t_0}^{t-1} \frac{\eta_i}{a_i}. \tag{2.6}$$

We note that (2.6) is true for any $t \geq t_0 + 1$. If $t_0 = 0$ then

$$\Delta_t \leq (1 - \alpha_0) a_{t-1} \frac{\Delta_0}{a_0} + a_{t-1} \sum_{i=0}^{t-1} \frac{\eta_i}{a_i} = (1 - \alpha_0) a_{t-1} \frac{\Delta_0}{a_0} + a_{t-1} \sum_{i \in [t]} \frac{\eta_{i-1}}{a_{i-1}}.$$

If $t_0 \geq 1$ then

$$\Delta_t \leq (1 - \alpha_{t_0}) a_{t-1} \frac{\Delta_{t_0}}{a_{t_0}} + a_{t-1} \sum_{i=t_0}^{t-1} \frac{\eta_i}{a_i}$$

$$= (1-\alpha_{t_0})a_{t-1}\frac{\Delta_{t_0}}{a_{t_0}} + a_{t-1}\left(\sum_{i=0}^{t-1}\frac{\eta_i}{a_i} - \sum_{i=0}^{t_0-1}\frac{\eta_i}{a_i}\right)$$
$$= \left((1-\alpha_{t_0})\frac{\Delta_{t_0}}{a_{t_0}} - \sum_{i\in[t_0]}\frac{\eta_{i-1}}{a_{i-1}}\right)a_{t-1} + a_{t-1}\sum_{i\in[t]}\frac{\eta_{i-1}}{a_{i-1}}.$$

Hence, (2.4) is true for any $t > t_0$. □

In fact, we follow the well-studied stepsizes $\alpha_t = \frac{2}{t+2}$ for each $t \geq 0$ in the analysis of all proposed methods. Hence, the following corollary gives a more compact expression of the right-hand side of (2.4).

Corollary 2.4. *If $\alpha_t = \frac{2}{t+2}$ for each $t \geq 0$ then we have*

$$a_t = \frac{2}{(t+1)(t+2)}, \quad \forall t \geq 0. \tag{2.7}$$

Thus, inequality (2.4) becomes

$$\Delta_t \leq \frac{2H_0}{(t+1)t} + \frac{1}{(t+1)t}\sum_{i\in[t]}(i+1)i\eta_{i-1}, \quad \forall t \geq t_0+1. \tag{2.8}$$

Proof. First, we will prove (2.7) by induction. When $t = 0$, the claim is true. We assume that the claim is true up to $t \geq 0$, then we have that

$$a_{t+1} = (1-\alpha_{t+1})a_t = \left(1-\frac{2}{t+3}\right)\frac{2}{(t+1)(t+2)} = \frac{t+1}{t+3}\frac{2}{(t+1)(t+2)} = \frac{2}{(t+2)(t+3)}.$$

Inequality (2.8) follows as a consequence of (2.4) and (2.7). □

2.3 Super-optimality, assumptions on the coerciveness and error bound

The proposed methods in Chapters 4 to 6 will construct candidate solutions $\{z_t\}_{t\geq 0} \subset X$ from convex combinations of $\{x_t\}_{t\geq 0}$ (where the x_t-iterates are constructed using (2.1)) and our conver-

gence analysis will show that $\limsup_{t\to\infty} f(z_t) \le f_{\text{opt}}$, $\limsup_{t\to\infty} g(z_t) \le g_{\text{opt}}$. Since $g(z_t) \ge g_{\text{opt}}$ by definition, we have $\lim_{t\to\infty} g(z_t) = g_{\text{opt}}$. In general, it is possible to have $f(z_t) < f_{\text{opt}}$ when $g(z_t) > g_{\text{opt}}$, and in this case we say that z_t is a *super-optimal* solution for problem (1.1). Therefore, it is not clear a priori that we will have $f(z_t) \to f_{\text{opt}}$. The next lemma shows that we can ensure this under mild assumptions.

Lemma 2.5. *Suppose $\{z_t\}_{t\ge 0}$ is a bounded sequence in X such that*

$$\limsup_{t\to\infty} f(z_t) \le f_{\text{opt}}, \quad \lim_{t\to\infty} g(z_t) = g_{\text{opt}}. \tag{2.9}$$

If X is closed and f, g are continuous on X, then any accumulation point of $\{z_t\}_{t\ge 0}$ is a solution of problem (1.1), and that

$$\lim_{t\to\infty} f(z_t) = f_{\text{opt}}, \quad \lim_{t\to\infty} g(z_t) = g_{\text{opt}}. \tag{2.10}$$

Proof. According to the Bolzano-Weierstrass theorem, $\{z_t\}_{t\ge 0}$ must have at least an accumulation point and since X is closed, all accumulation points must be in X. Given $z^* \in X$ is an accumulation point of $\{z_t\}_{t\ge 0}$, there exists a subsequence $\{z_{t_k}\}_{k\ge 0}$ such that $z_{t_k} \to z^*$. By the continuity of g, we have

$$g_{\text{opt}} = \lim_{k\to\infty} g(z_{t_k}) = g\left(\lim_{k\to\infty} z_{t_k}\right) = g(z^*).$$

Thus, $z^* \in X_{\text{opt}}$. By the continuity of f and the definition of f_{opt}, we also have

$$f_{\text{opt}} \le f(z^*) = \lim_{k\to\infty} f(z_{t_k}) \le \limsup_{t\to\infty} f(z_t) \le f_{\text{opt}}.$$

Hence, $f(z^*) = f_{\text{opt}}$. Therefore, any accumulation point of $\{z_t\}_{t\ge 0}$ is a solution of problem (1.1). Since $\{z_t\}_{t\ge 0}$ is a bounded sequence in X, which is closed, the closure of it is a compact subset of X. Thus, by the continuity of f on X, $\{f(z_t)\}_{t\ge 0}$ is a bounded sequence, which means that $\liminf_{t\to\infty} f(z_t), \limsup_{t\to\infty} f(z_t)$ are both finite. Let $\{f(z_{t_i})\}_{i\ge 0}$ be a subsequence of $\{f(z_t)\}_{t\ge 0}$ that converges to $\liminf_{t\to\infty} f(z_t)$. Assume that $\{z_{t_i}\}_{i\ge 0}$ is convergent; otherwise, we extract a convergent subsequence, which must exist due to the boundedness of $\{z_{t_i}\}_{i\ge 0}$. Since the accumulation

point of $\{z_{t_i}\}_{i\geq 0}$ is a solution of problem (1.1), we have

$$\liminf_{t\to\infty} f(z_t) = \lim_{i\to\infty} f(z_{t_i}) = f\left(\lim_{i\to\infty} z_{t_i}\right) = f_{\text{opt}}.$$

Hence, $\lim_{t\to\infty} f(z_t) = f_{\text{opt}}$. □

In Lemma 2.5, the assumption on the boundedness of $\{z_t\}_{t\geq 0}$ is critical for asymptotic convergence, but may not hold in general. However, Assumption 2 stated below can ensure that $\{z_t\}_{t\geq 0}$ is bounded, as shown in Lemma 2.6.

Assumption 2. The pointwise maximum function $\ell(x) := \max\{g(x), f(x)\}$ is coercive on X, i.e., $\ell(z_t) \to \infty$ for any sequence $\{z_t\}_{t\geq 0}$ in X such that $\|z_t\| \to \infty$.

Lemma 2.6. *Let $\{z_t\}_{t\geq 0}$ be a sequence in X satisfying (2.9). If Assumption 2 holds, then $\{z_t\}_{t\geq 0}$ is bounded.*

Proof. If (2.9) holds then we have $\lim_{t\to\infty} g(z_t)$, $\limsup_{t\to\infty} f(z_t) < \max\{f_{\text{opt}}+1, g_{\text{opt}}+1\}$. From the definition of limit and superior limit, there exists a positive integer t_1 such that for $t > t_1$, $g(z_t), f(z_t) < \max\{f_{\text{opt}}+1, g_{\text{opt}}+1\}$ and hence, $\ell(z_t) < \max\{f_{\text{opt}}+1, g_{\text{opt}}+1\}$. Thus, we have

$$\limsup_{t\to\infty} \ell(z_t) \leq \max\{f_{\text{opt}}+1, g_{\text{opt}}+1\},$$

which implies $\{z_t\}_{t\geq 0}$ cannot be unbounded if Assumption 2 holds. □

For the methods proposed in this study, we do not rely on Assumption 2 to establish (2.9). Nevertheless, to ensure (2.10), Assumption 2 is sufficient and *almost necessary*, in the sense that we can construct a bilevel problem satisfying Assumption 1 but not Assumption 2 and a sequence satisfying (2.9) but not (2.10). Such construction is detailed in Example 2.1.

Example 2.1. We consider an example where

$$\begin{aligned} f(x) &:= x_2^2 - 2x_2, \\ g(x) &:= \frac{x_2^2}{x_1}, \\ X &:= \{x \in \mathsf{R}^2 \mid x_1 \geq 1, x_2^2 \leq x_1\}. \end{aligned} \tag{2.11}$$

First, we verify the problem (2.11) meets Assumption 1 with Euclidean norm $\|\cdot\|_2$. The base domain X is a closed convex set and hence, satisfies Assumption 1(a). Since $\nabla f(x) = (0, 2x_2 - 2)$, we have that

$$\|\nabla f(x) - \nabla f(y)\|_2 = 2|x_2 - y_2| \leq 2\|x - y\|_2,$$

for any $x, y \in X$. Combining this result with the fact that f is convex quadratic, the outer-level objective satisfies Assumption 1(b) with $L_f = 2$. We will prove that g satisfies Assumption 1(c). Since g is a quadratic over linear function, g is indeed a convex function on X. Following Wright and Recht [38, Lemma 2.3], we verify the smoothness of g by showing that $\nabla^2 g(x)$ has the spectral norm, i.e., the largest eigenvalue, bounded above by 4 over X, i.e., $L_g = 4$. We observe that

$$\nabla g(x) = \begin{bmatrix} -x_2^2/x_1^2 \\ 2x_2/x_1 \end{bmatrix} \implies \nabla^2 g(x) = \begin{bmatrix} 2x_2^2/x_1^3 & -2x_2/x_1^2 \\ -2x_2/x_1^2 & 2/x_1 \end{bmatrix} = \frac{2}{x_1} \begin{bmatrix} x_2/x_1 \\ -1 \end{bmatrix} \begin{bmatrix} x_2/x_1 & -1 \end{bmatrix}.$$

Before moving on, we need to prove the fact that $\sigma_{\max}(aa^\top) = a^\top a = \|a\|_2^2$ for any $a \in \mathsf{R}^2$. If $a = 0$, then the claim trivially holds. Otherwise, we notice that a is an eigenvector of $aa^\top$ with the associated eigenvalue of $\|a\|_2^2$. In addition, any non-zero vector that is orthogonal to a is also an eigenvector of $aa^\top$ with the associated eigenvalue of 0. Therefore, $aa^\top$ only has two distinct eigenvalues, which are $\|a\|_2^2$ and 0. Hence, the claim is true.

Using this fact, we have that

$$\sigma_{\max}(\nabla^2 g(x)) = \frac{2}{x_1}\left(\frac{x_2^2}{x_1^2} + 1\right),$$

and since $x_1 \geq 1$ and $x_2^2 \leq x_1$, we obtain

$$\sigma_{\max}(\nabla^2 g(x)) \leq \frac{2}{x_1}\left(\frac{1}{x_1}+1\right) \leq 4.$$

We can see that $X_{\text{opt}} = \{(s,0) \mid s \geq 1\}$, $f_{\text{opt}} = 0$, and $g_{\text{opt}} = 0$. Given positive integer t, we define $z_t := (t,1)$, which is inner-level feasible and satisfies

$$\lim_{t\to\infty} g(z_t) = \lim_{t\to\infty}\left(\frac{1}{t}\right) = 0 = g_{\text{opt}}, \quad \lim_{t\to\infty} f(z_t) = -1 < 0 = f_{\text{opt}}.$$

Therefore, $\{z_t\}_{t\geq 1}$ satisfies (2.9) but not (2.10). By noticing sequence $\{z_t\}_{t\geq 1}$ is unbounded, we have a counterexample to claim that f and g do not meet Assumption 2. ■

Given that it is possible that $f(z_t) < f_{\text{opt}}$, we may wish to lower bound $f(z_t) - f_{\text{opt}}$ for our algorithms. In fact, a generic bound exists if g satisfies the following.

Condition 2.1 (Hölderian error bound)**.** For some $\tau > 0$ and $r \geq 1$, g satisfies

$$\tau \operatorname{Dist}^r(x, X_{\text{opt}}) \leq g(x) - g_{\text{opt}}.$$

Condition 2.1 has been studied extensively in optimisation literature. When $r = 1$, we say that g possesses *weak sharp minima*, and this holds for all solvable linear programs as well as some classes of quadratic programs [5, Section 3.1–3.2]. The case $r = 2$ is known as the *quadratic growth condition* [13]. The following lemma shows that under Condition 2.1, lower bounds on $f(z_t) - f_{\text{opt}}$ can automatically be obtained from existing upper bounds on $g(z_t) - g_{\text{opt}}$.

Lemma 2.7. *Suppose that Assumption 1(a), Assumption 1(b) and Condition 2.1 hold. Then for any $z, x \in X$, we have*

$$f(z) - f_{\text{opt}} \geq -\frac{\|\nabla f(x)\|_* + L_f\|z-x\|}{\tau^{1/r}}(g(z) - g_{\text{opt}})^{1/r} - \frac{L_f}{\tau^{2/r}}(g(z) - g_{\text{opt}})^{2/r}.$$

Proof. Under Condition 2.1, we have that

$$\tau\|z-w\|^r \le g(z)-g_{\rm opt} \iff \|z-w\| \le \frac{1}{\tau^{1/r}}\left(g(z)-g_{\rm opt}\right)^{1/r},$$

where $w \in \arg\min_{u\in X_{\rm opt}} \|z-u\|$. Since $f(w) \ge f_{\rm opt}$, we have that

$$f(z)-f_{\rm opt} \ge f(z)-f(w) \ge \nabla f(w)^\top (z-w) \ge -\|\nabla f(w)\|_*\|z-w\|.$$

By the smoothness of f and the triangle inequality, we have

$$\begin{aligned}\|\nabla f(w)\|_* &= \|\nabla f(w)-\nabla f(x)+\nabla f(x)\|_* \\ &\le \|\nabla f(x)\|_* + \|\nabla f(w)-\nabla f(x)\|_* \\ &\le \|\nabla f(x)\|_* + L_f\|w-x\| \\ &\le \|\nabla f(x)\|_* + L_f\|x-z\| + L_f\|w-z\|.\end{aligned}$$

Therefore, we have

$$\begin{aligned}f(z)-f_{\rm opt} &\ge -\left(\|\nabla f(x)\|_* + L_f\|x-z\|\right)\|w-z\| - L_f\|w-z\|^2 \\ &\ge -\frac{\|\nabla f(x)\|_* + L_f\|x-z\|}{\tau^{1/r}}(g(z)-g_{\rm opt})^{1/r} - \frac{L_f}{\tau^{2/r}}(g(z)-g_{\rm opt})^{2/r}.\end{aligned}$$

□

It is important to note that none of our algorithms rely on Condition 2.1 to achieve (2.9) and (2.10).

2.4 Strong duality and the solvability of the dual problem

Interestingly, although Slater's condition never holds for problem (1.3), by adopting Assumption 3 stated below, we can still establish strong duality between problem (1.3) and the Lagrangian dual

defined as follows:

$$\sup_{\lambda \geq 0} \left\{ \inf_{x \in X} L(x, \lambda) \right\}, \text{ where } L(x, \lambda) := f(x) + \lambda(g(x) - g_{\text{opt}}), \tag{2.12}$$

as shown in Lemma 2.8.

Assumption 3. f is bounded below over X, i.e., $\underline{f} := \inf_{x \in X} f(x) > -\infty$.

Lemma 2.8. *If Assumptions 1–3 hold, then we have*

$$\sup_{\lambda \geq 0} \left\{ \min_{x \in X} L(x, \lambda) \right\} = \min_{x \in X} \left\{ \sup_{\lambda \geq 0} L(x, \lambda) \right\}. \tag{2.13}$$

Proof. First, we prove that given $\lambda > 0$, $\min_{x \in X} L(x, \lambda)$ is solvable. From Assumption 3, we have that

$$\begin{aligned} L(x, \lambda) &= f(x) - \underline{f} + \lambda(g(x) - g_{\text{opt}}) + \underline{f} \\ &\geq \min\{1, \lambda\} \left(f(x) - \underline{f} + g(x) - g_{\text{opt}} \right) + \underline{f} \\ &\geq \min\{1, \lambda\} \max\{g(x) - g_{\text{opt}}, f(x) - \underline{f}\} + \underline{f}. \end{aligned}$$

By defining $\xi := \max\{g_{\text{opt}}, \underline{f}\}$ and recalling that $\ell(x) := \max\{f(x), g(x)\}$, we have

$$\begin{aligned} L(x, \lambda) &\geq \min\{1, \lambda\} \max\{g(x) - \xi, f(x) - \xi\} + \underline{f} \\ &= \min\{1, \lambda\}(\ell(x) - \xi) + \underline{f}. \end{aligned}$$

Therefore, given $\lambda > 0$, $L(x, \lambda)$ is a coercive, continuous function with respect to $x \in X$. By Dhara and Dutta [10, Theorem 1.14], it has an attainable minimum over X.

Suppose $\{\lambda_t\}_{t \geq 0}$ is a positive, increasing, divergent sequence and $\{x_t\}_{t \geq 0}$ is a sequence defined as follows:

$$x_t \in \arg\min_{x \in X} L(x, \lambda_t), \quad \forall t \geq 0.$$

For any $t \geq 0$, we have that

$$L(x_t, \lambda_t) = \min_{x \in X} L(x, \lambda_t) \leq \sup_{\lambda \geq 0} \left\{ \min_{x \in X} L(x, \lambda) \right\}.$$

For any $x \in X, \lambda \geq 0$, we have that $\min_{x \in X} L(x, \lambda) \leq L(x, \lambda)$, which implies that for any $x \in X$,

$$\sup_{\lambda \geq 0} \left\{ \min_{x \in X} L(x, \lambda) \right\} \leq \sup_{\lambda \geq 0} L(x, \lambda).$$

Thus, we have that

$$\sup_{\lambda \geq 0} \left\{ \min_{x \in X} L(x, \lambda) \right\} \leq \inf_{x \in X} \left\{ \sup_{\lambda \geq 0} L(x, \lambda) \right\}.$$

Given $x \in X$, we have

$$\sup_{\lambda \geq 0} L(x, \lambda) = \begin{cases} f(x), & x \in X_{\text{opt}} \\ \infty, & x \in X \setminus X_{\text{opt}}, \end{cases}$$

and therefore,

$$\min_{x \in X} \left\{ \sup_{\lambda \geq 0} L(x, \lambda) \right\} = \min_{x \in X_{\text{opt}}} f(x) = f_{\text{opt}}. \tag{2.14}$$

Hence, we have that

$$f(x_t) + \lambda_t (g(x_t) - g_{\text{opt}}) \leq f_{\text{opt}}. \tag{2.15}$$

Using $f(x_t) \geq \underline{f}$ and dividing both sides of (2.15) by λ_t, we have

$$\frac{\underline{f}}{\lambda_t} + g(x_t) - g_{\text{opt}} \leq \frac{f_{\text{opt}}}{\lambda_t}.$$

By taking $t \to \infty$ and noting that $\lambda_t \to \infty$, we have that $g(x_t) \to g_{\text{opt}}$. Since $\lambda_t > 0$ and $g(x_t) \geq g_{\text{opt}}$, we have $f(x_t) \leq f_{\text{opt}}$, which implies $\limsup_{t \to \infty} f(x_t) \leq f_{\text{opt}}$. By Lemmas 2.5 to 2.6, we have that $\{x_t\}_{t \geq 0}$ must be bounded, $\lim_{t \to \infty} g(x_t) = g_{\text{opt}}$, and $\lim_{t \to \infty} f(x_t) = f_{\text{opt}}$.

Given $\lambda < \lambda'$, since $g(x) \geq g_{\text{opt}}$ for any $x \in X$, we have that $L(x, \lambda) \leq L(x, \lambda')$, which implies

$\min_{x\in X} L(x,\lambda)$ is a non-decreasing function with respect to λ. Hence, we have

$$\sup_{\lambda\geq 0}\left\{\min_{x\in X} L(x,\lambda)\right\} = \lim_{\lambda\to\infty}\left(\min_{x\in X} L(x,\lambda)\right) = \lim_{t\to\infty}\left(f(x_t)+\lambda_t(g(x_t)-g_{\text{opt}})\right) \geq \lim_{t\to\infty} f(x_t) = f_{\text{opt}}, \tag{2.16}$$

where in the rightmost inequality, we use the fact that $g(x_t)\geq g_{\text{opt}}, \lambda_t > 0$ for $t\geq 0$. From (2.15), we obtain

$$\sup_{\lambda\geq 0}\left\{\min_{x\in X} L(x,\lambda)\right\} = \lim_{\lambda\to\infty}\left(\min_{x\in X} L(x,\lambda)\right) = \lim_{t\to\infty}\left(f(x_t)+\lambda_t(g(x_t)-g_{\text{opt}})\right) \leq f_{\text{opt}}. \tag{2.17}$$

Using (2.14), (2.16) and (2.17), we obtain (2.13). □

Although Lemma 2.8 claims strong duality for problem (1.3) under some mild conditions, there is no guarantee that the dual problem (2.12) is *solvable*. (We will point out in Example 2.3 in which Assumptions 1–3 hold, but the dual problem is not solvable later in this section.) On the other hand, if there exists an optimal dual λ_{opt} as in Assumption 4 stated below, then convergence rates may be improved for some algorithms, e.g., the `PD-CG` algorithm from Chapter 6. We now present a more general constraint qualification, which guarantees Assumption 4 stated below.

Assumption 4. There exists $\lambda_{\text{opt}}\geq 0$ such that

$$\min_{x\in X}\{f(x)+\lambda_{\text{opt}}(g(x)-g_{\text{opt}})\} = f_{\text{opt}}.$$

First, we review the concept of set-valued mapping and the definition of *calmness*. Given two positive integers n_1, n_2, a set-valued mapping $\Gamma : \mathsf{R}^{n_1} \quad \mathsf{R}^{n_2}$ maps each point $u\in\mathsf{R}^{n_1}$ to a subset $\Gamma(u)$ of R^{n_2}. Under this definition, the graph of Γ is defined to be

$$\operatorname{gph}\Gamma := \{(u,v)\in\mathsf{R}^{n_1}\times\mathsf{R}^{n_2} \mid v\in\Gamma(u)\}.$$

Let $\|\cdot\|_{(1)}, \|\cdot\|_{(2)}$ be two norms on $\mathsf{R}^{n_1}, \mathsf{R}^{n_2}$ respectively. Following Penot [30, Definition 3.1], Γ is called *calm* at $(u,v)\in\operatorname{gph}\Gamma$ if there exists two open neighbourhoods $\mathcal{U},\mathcal{V}$ of u, v, respectively and a

constant $c > 0$ such that for any $u' \in \mathcal{U}$, every $v' \in \mathcal{V} \cap \Gamma(u')$ satisfies $\text{Dist}\,(v', \Gamma(u)) \leq c\|u - u'\|_{(1)}$, where Dist is computed via norm $\|\cdot\|_{(2)}$ for this definition.

Example 2.2. One can consider $\Lambda : \mathsf{R} \quad \mathsf{R}^n$ be defined as follows:

$$\Lambda(y) := \{x \in X \mid g(x) - g_{\text{opt}} \leq y\}, \quad \forall y \in \mathsf{R}$$

Suppose x^* is a minimiser of g over X. We observe that $(0, x^*) \in \text{gph}\,\Lambda$, $\Lambda(0) = X_{\text{opt}}$ and $\Lambda(y) = \emptyset$ for $y < 0$. Hence, Λ is calm at $(0, x^*)$ if and only if there exists $\epsilon, \delta > 0$, and $c > 0$ such that for any $y \in [0, \epsilon)$, every $x \in X \cap \mathsf{B}_{\|\cdot\|}(x^*, \delta)$ such that $g(x) - g_{\text{opt}} \leq y$ satisfies

$$\text{Dist}(x, X_{\text{opt}}) \leq cy. \tag{2.18}$$

■

Now, we are ready to state an optimality condition for problem (1.3), which was developed by Franke et al. [18, Theorem 5.6].

Lemma 2.9 ([18, Theorem 5.6])**.** *Let $\Lambda : \mathsf{R} \quad \mathsf{R}^n$ and $x^* \in X_{\text{opt}}$ be de ned as in Example 2.2. If Assumption 1(a) holds, f, g are convex, continuously di erentiable on an open neighborhood of X, and Λ is calm at $(0, x^*)$, then x^* is an optimal solution solution of problem (1.3) if and only if there exists $\lambda^* \geq 0$ such that*

$$\left(\nabla f(x^*) + \lambda^* \nabla g(x^*)\right)^\top (x - x^*) \geq 0, \quad \forall x \in X. \tag{2.19}$$

Remark 2.4.1. From Lemma 2.9, if (2.19) holds then $x^* \in X_{\text{opt}}$ must minimise $f(x) + \lambda^*(g(x) - g_{\text{opt}})$. Therefore, x^* must be a solution of (1.1), which implies Assumption 4 holds with $\lambda_{\text{opt}} = \lambda^*$. ■

In fact, the condition of Λ being calm at $(0, x^*)$ holds under Condition 2.1 with $r = 1$.

Lemma 2.10. *If Condition 2.1 holds with $r = 1$, then for any $x^* \in X_{\text{opt}}$, the set-valued mapping Λ as de ned in Example 2.2 is calm at $(0, x^*)$.*

Proof. Condition 2.1 holds with $r = 1$ means that for any $x \in X$ and $y \geq g(x) - g_{\text{opt}}$, we have

$$\text{Dist}(x, X_{\text{opt}}) \leq \frac{1}{\tau}(g(x) - g_{\text{opt}}) \leq \frac{1}{\tau}y.$$

Thus, (2.18) holds with $c := \frac{1}{\tau}$ and any $\epsilon, \delta > 0$. □

In addition to ensuring the solvability of the dual problem, we would like to point out that Condition 2.1 with $r = 1$ is nearly a necessary condition, in the sense that even Assumptions 1–3 and Condition 2.1 hold with $r > 1$ cannot guarantee the solvability of the Lagrangian dual. We justify this claim via the following example.

Example 2.3. Given $s > 1$, we consider an example in which:

$$\begin{aligned} f(x) &:= x_2 - x_1, \\ g(x) &:= x_2, \\ X &:= \{x \in \mathsf{R}^2 \mid 0 \leq x_1^s \leq x_2 \leq M_s\}, \end{aligned} \tag{2.20}$$

where $M_s > 0$ is a sufficiently large number we will choose later. We will prove that there is no optimal dual variable for problem (2.20) despite strong duality.

First, we observe that Assumption 1 holds for problem (2.20) with Euclidean norm. Since the base domain is compact, Assumptions 2–3 hold. We will prove that Condition 2.1 holds with $r = s$. For problem (2.20), the only optimal solution for g over X is $(0, 0)$, i.e., $X_{\text{opt}} = \{(0, 0)\}$. We assume x is an inner-level feasible point with $x_2 > 0$. We observe that

$$\frac{g(x) - g_{\text{opt}}}{\text{Dist}^s(x, X_{\text{opt}})} = \frac{x_2}{(x_1^2 + x_2^2)^{s/2}} = \frac{1}{\left(\frac{x_1^2 + x_2^2}{x_2^{2/s}}\right)^{s/2}} = \frac{1}{\left(\left(\frac{x_1^s}{x_2}\right)^{2/s} + x_2^{2-2/s}\right)^{s/2}}.$$

Since $M_s \geq x_2 \geq x_1^s, x_2 > 0$ and $s > 1$, we have that

$$0 < \left(\frac{x_1^s}{x_2}\right)^{2/s} + x_2^{2-2/s} \leq 1 + M_s^{2-2/s},$$

and we also define

$$\tau_s := \frac{1}{\left(1 + M_s^{2-2/s}\right)^{s/2}} > 0.$$

Therefore, for any inner-level feasible x such that $x_2 > 0$, we have

$$g(x) - g_{\text{opt}} \geq \tau_s \, \text{Dist}^s \left(x, X_{\text{opt}}\right). \tag{2.21}$$

In addition, we need to prove that (2.21) also holds any inner-level feasible x with x_2. However, this is trivial since the only inner-level feasible point with x_2 is $(0, 0)$, which obviously satisfies (2.21). Thus, (2.21) holds for any $x \in X$. Now, we prove that Condition 2.1 does not hold with $r = 1$. We assume for contradiction that there is $\tau > 0$ such that

$$g(x) - g_{\text{opt}} \geq \tau \, \text{Dist}\left(x, X_{\text{opt}}\right), \quad \forall x \in X.$$

Given $t \geq 1$, we define $z_t := \left(\frac{M_s}{t}, \frac{M_s^s}{t^s}\right)$, which is inner-level feasible. We observe that

$$\tau \leq \frac{g(z_t) - g_{\text{opt}}}{\text{Dist}\left(z_t, X_{\text{opt}}\right)} = \frac{(z_t)_2}{\left((z_t)_1^2 + (z_t)_2^2\right)^{1/2}} = \frac{1}{\left(\left(M_s t^{-1} M_s^{-s} t^s\right)^2 + 1\right)^{1/2}} = \frac{1}{\left((M_s^{1-s} t^{s-1})^2 + 1\right)^{1/2}}.$$

By noting $t^{s-1} \to \infty$ as we take t to ∞, we obtain $\tau \leq 0$, which is a contradiction. We consider the Lagrangian of problem (2.20), which is as follows:

$$L(x, \lambda) = x_2 - x_1 + \lambda x_2 = (1 + \lambda) x_2 - x_1.$$

Given $\lambda \geq 0$, for any inner-level feasible x, we have that $L(x, \lambda) \geq (1+\lambda) x_1^s - x_1$, and $(1+\lambda) x_1^s - x_1$ can be uniquely minimised over $x_1 \geq 0$ at

$$x_1^*(\lambda) := \left(\frac{1}{s(1+\lambda)}\right)^{1/(s-1)} > 0.$$

To ensure $x_1^*(\lambda)$ also minimises $(1+\lambda)x_1^s - x_1$ over $\left[0, M_s^{1/s}\right]$, we set

$$M_s := \left(\frac{1}{s(1+\lambda)}\right)^{s/(s-1)}.$$

Thus, for any $\lambda \geq 0$, $L(x,\lambda)$ is minimised uniquely at

$$x^*(\lambda) := \left(\left(\frac{1}{s(1+\lambda)}\right)^{1/(s-1)}, \left(\frac{1}{s(1+\lambda)}\right)^{s/(s-1)}\right),$$

and

$$L(x^*(\lambda),\lambda) = (1+\lambda)\left(\frac{1}{s(1+\lambda)}\right)^{s/(s-1)} - \left(\frac{1}{s(1+\lambda)}\right)^{1/(s-1)} = \left(\frac{1}{s}-1\right)\left(\frac{1}{s(1+\lambda)}\right)^{1/(s-1)} < 0.$$

Hence, given $\lambda \geq 0$, the minimiser of $L(x,\lambda)$ will never be $(0,0)$, which is the only solution of (2.20) and the optimal dual value, which is 0, is not attainable. ■

2.5 Conditional gradient method

As we seek projection-free schemes to solve problem (1.1), the CG method [17] should be an appropriate foundation for our methods. Instead of computing the projection onto the base domain, the CG method can solve the following single-level smooth convex optimisation problem (for convenience, we abuse the notations of the inner-level objective and base domain of problem (1.1)):

$$\min_{x\in X} g(x), \tag{2.22}$$

To avoid computing projection onto X, the CG method considers a linear Taylor series approximation of the objective function around the current iterate x_t and solves the following subproblem:

$$v_t \in \arg\min_{v\in X}\left\{g(x_t) + \nabla g(x_t)^\top (v - x_t)\right\} = \arg\min_{v\in X} \nabla g(x_t)^\top v.$$

Algorithm 1: [Frank and Wolfe [17]] Conditional gradient method (CG).

Data: Stepsizes $\{\alpha_t\}_{t\geq 0} \subseteq [0,1]$, number of iterations T.
Result: sequence $\{x_t\}_{t\in[T]}$.
Initialise $x_0 \in X$;
for $t = 0, 1, \ldots, T-1$ **do**
  Compute

$$v_t \in \arg\min_{v\in X} \nabla g(x_t)^\top v$$
$$x_{t+1} := x_t + \alpha_t(v_t - x_t).$$

Afterwards, the update step is as given in (2.1): $x_{t+1} := x_t + \alpha_t(v_t - x_t)$, for some carefully chosen $\alpha_t \in [0,1]$. The method is given in Algorithm 1.

Since x_0 is feasible then due to the convexity of X and the definition of $\{v_t\}_{t\geq 0}$, $\{x_t\}_{t\geq 0}$ is a sequence in X. Since the objective of the subproblem is a linear function, one may notice that this method is advantageous when X is a polyhedron in which the exact solution can be efficiently and exactly computed. Nevertheless, the subproblem may be unsolvable if X is unbounded. Hence, the CG method has traditionally been used with bounded domains. Nevertheless, this thesis also deals with unbounded domains, and therefore, we present an extension of the CG method to unbounded domains in Chapter 3.

By [23, Theorem 1], the well-studied stepsizes $\alpha_t = 2/(t+2)$ for $t \geq 0$ are recommended and the CG method converges at the rate of $O(1/T)$.

Lemma 2.11 ([23, Theorem 1])**.** *Suppose $\{x_t\}_{t\in[T]}$ is the sequence which is generated by Algorithm 1 with that stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1(a), Assumption 1(c) hold and X is bounded with diameter D, then*

$$g(x_T) - g_{\text{opt}} \leq \frac{2L_g D^2}{T+2}.$$

In practice, one may want to directly determine an ϵ_g-suboptimal solution without manually computing sufficient iterations to ensure such tolerance via Lemma 2.11. Thus, a more convenient stopping criterion should be utilised. Fortunately, we can rely on a quantity called duality surrogate

gap of g, which is defined by Jaggi [23, Equation 2] as follows:

$$S(x) := \max_{v \in X} \nabla g(x)^\top (x - v), \quad x \in X. \tag{2.23}$$

Under the convexity of g, given $x^* \in X_{\text{opt}}$ we have

$$S(x) \geq g(x)^\top (x - x^*) \geq g(x) - g(x^*) = g(x) - g_{\text{opt}}.$$

Hence, if $S(x) \leq \epsilon_g$, then x must be an ϵ_g-suboptimal solution. Furthermore, we have the following result.

Lemma 2.12 ([23, Theorem 2]). *Suppose $\{x_t\}_{t \in [T]}$ is the sequence which is generated by Algorithm 1 with that stepsizes $\alpha_t = 2/(t+2)$ for $t \geq 0$. If Assumption 1(a), Assumption 1(c) hold and X is bounded with diameter D, then there exists $k_T \in [T]$ such that*

$$S(x_{k_T}) \leq \frac{2\beta L_g D^2}{T+2},$$

where $\beta = \frac{27}{8} = 3.375$.

Given $\epsilon_g > 0$, Lemma 2.12 suggests that for a sufficiently large number of iterations T, we have $S(x_T) = g(x_T)^\top (x_T - v_T) < \epsilon_g$.

Chapter 3

Extension of the conditional gradient method to unbounded domains

In this chapter, we extend the CG method to solve problem (2.22)

$$\min_{x \in X} g(x),$$

without the boundedness of X required in Lemma 2.11 and Lemma 2.12. In Section 3.1, we motivate our approach and describe the algorithm. In Section 3.2, we discuss the convergence of the method. In Section 3.3, we describe its implications in solving problem (1.1).

3.1 Approach and method description

From Algorithm 1 and Lemma 2.11, the boundedness of X is critical as it allows the linear subproblem to be solvable and makes the upper bound on the optimality gap finite. Thus, one natural

move when we face the unboundedness is to "temporarily" consider a bounded truncation of X at each iteration so that at least the linear minimisation oracle is solvable and gradually increase the scope over iterates so that the algorithm will not miss any point in X. If we have a point $x_0 \in X$ then at the iteration t, the strategy naively goes as follows:

- First, we consider a ball $\mathbf{B}_{\|\cdot\|}(x_0, d_t/2)$, where $d_t > 0$.
- Second, we compute

$$v_t \in \underset{v \in X \cap \mathbf{B}_{\|\cdot\|}(x_0, d_t/2)}{\arg\min} \nabla g(x_t)^\top v,$$

  and update the next iterate as $x_{t+1} = x_t + \alpha(v_t - x_t)$, where $\alpha_t \in [0, 1]$ is the stepsize.
- Third, we choose the diameter $d_{t+1} > d_t$ for the next iteration.

At finitely many first iterations, the balls may not be sufficiently large to intersect X_{opt} but after sufficiently large iterations, we can have $\text{Dist}(x_0, X_{\text{opt}}) \leq d_t/2$. In fact, it is not a must to be a sequence of norm balls for the idea to work. We only need a sequence of sets $\{B_t\}_{t\geq 0}$, whose elements are referred to as the *coverings*, satisfies the following assumption.

Assumption 5. The sequence of coverings $\{B_t\}_{t\geq 0}$ consists of compact convex sets with diameters $\{d_t\}_{t\geq 0}$ and satisfies

$$B_t \subseteq B_{t+1}, \quad B_t \cap X \neq \emptyset, \quad \forall t \geq 0, \quad X \subseteq \left(\bigcup_{t=0}^{\infty} B_t \right).$$

The reason for such generalisation is that we would like to design $\{B_t\}_{t\geq 0}$ so that given X, the linear minimisation oracle over $X \cap B_t$ can be efficiently computed. If X is bounded, then Assumption 5 holds trivially by setting $B_t = X$ for all t. When X is unbounded, Assumption 5 implies that $\{d_t\}_{t\geq 0}$ is a non-decreasing and divergent sequence. To ensure asymptotic convergence of the method, as shown in the next section, we may have to control the growth rate of the diameters $\{d_t\}_{t\geq 0}$. To show that both tasks can be done, we provide an example of how to design the coverings given a specific base domain X and a schedule of the diameters.

Algorithm 2: Unbounded conditional gradient method (UCG).

Data: Parameters $\{\alpha_t\}_{t\geq 0} \subseteq [0,1]$, $\{B_t\}_{t\geq 0}$ satisfying Assumption 5, number of iterations T.
Result: sequence $\{x_t\}_{t\in[T]}$.
Initialise $x_0 \in X$;
for $t = 0, 1, \ldots, T-1$ **do**
  Compute

$$v_t \in \operatorname*{arg\,min}_{v\in X\cap B_t} \nabla g(x_t)^\top v$$
$$x_{t+1} := x_t + \alpha_t(v_t - x_t).$$

Example 3.1. We consider $X := \mathsf{R}^n_+$ and the desired sequence of diameters with respect to Euclidean norm $\{d_t\}_{t\geq 0}$ that is non-decreasing and diverges to ∞, then we can construct the coverings as follows:

$$B_t := \left\{ x \in \mathsf{R}^n \;\middle|\; 0 \leq x \leq \frac{d_t}{\sqrt{n}}\mathbf{1} \right\}, \quad \forall t \geq 0,$$

whose diameters are indeed $\{d_t\}_{t\geq 0}$. Then it is true that $B_t \subseteq B_{t+1}$ for $t \geq 0$. Given $t \geq 0$, we have that $B_t \cap X = B_t \neq \emptyset$. Due to the divergence of the diameters, given $x \in \mathsf{R}_+$, there exists a sufficiently large t such that $d_t \geq \sqrt{n}\max_{i\in[n]} x_i$, which implies $x \in \bigcup_{t=0}^\infty B_t$, and hence, $X \subseteq \bigcup_{t=0}^\infty B_t$. We show that a solution of the linear minimisation over $X \cap B_t$ can be efficiently computed for any $t \geq 0$. Given $c \in \mathsf{R}^n$, we have that a minimiser x^* of $c^\top x$ over $X \cap B_t$ is as follows:

$$x_i^* = \begin{cases} d_t/\sqrt{n} & c_i < 0 \\ 0 & c_i \geq 0, \end{cases} \quad \forall i \in [n].$$

■

Now, we formally describe our extension of CG, which is the *unbounded conditional gradient* (UCG) method, in Algorithm 2.

3.2 Convergence analysis

The following lemma will show how imposing Assumption 5 on the coverings leads to convergence of Algorithm 2.

Theorem 3.1. *Suppose $\{x_t\}_{t\in[T]}$ is the sequence which is generated by Algorithm 2. If Assumption 1(a), Assumption 1(c) and Assumption 5 hold, then*

$$g(x_T) - g_{\text{opt}} \leq O\left(\frac{d_T^2}{T}\right). \tag{3.1}$$

Proof. From the smoothness of g and the fact that $\|v_t - x_t\| \leq d_t$ for $t \geq 0$, we have that

$$g(x_{t+1}) - g(x) \leq \left(1 - \frac{2}{t+2}\right)(g(x_t) - g(x)) + \frac{2}{t+2}\left(\nabla g(x_t)^\top (v_t - x_t) + g(x_t) - g(x)\right) + \frac{2L_g d_t^2}{(t+2)^2}, \tag{3.2}$$

for any $x \in X$ according to Lemma 2.2. From Assumption 5, there exists $t^* \geq 1$ such that $X_{\text{opt}} \cap B_t \neq \emptyset$ for $t \geq t^*$. Therefore, for $t \geq t^*$, using the convexity of g and definition of v_t, we have that for some $x^* \in X_{\text{opt}} \cap B_t$,

$$\nabla g(x_t)^\top (v_t - x_t) \leq \nabla g(x_t)^\top (x^* - x_t) \leq g(x^*) - g(x_t).$$

Therefore, we substitute $x := x^*$ into (3.2) and using the above result to obtain that for any $t \geq t^*$

$$g(x_{t+1}) - g_{\text{opt}} \leq \left(1 - \frac{2}{t+2}\right)(g(x_t) - g_{\text{opt}}) + \frac{2L_g d_t^2}{(t+2)^2}.$$

Using Corollary 2.4, if $T > t^*$, then there exists a constant H^* such that

$$g(x_T) - g_{\text{opt}} \leq \frac{2H^*}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}(t+1)t\frac{2L_g d_{t-1}^2}{(t+1)^2} = \frac{2H^*}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}\frac{2L_g d_{t-1}^2 t}{t+1}.$$

Since $d_{t-1} \leq d_T$ and $\frac{t}{t+1} < 1$ for $t \in [T]$, we have that

$$g(x_T) - g_{\text{opt}} \leq \frac{2H^*}{(T+1)T} + \frac{2L_g d_T^2}{T+1}.$$

Since $d_T \geq d_0 > 0$, we have that $\lim_{T\to\infty}(T+1)d_T^2 = \infty$, and

$$\lim_{T\to\infty} \frac{T}{d_T^2}\left(\frac{2H^*}{(T+1)T} + \frac{2L_g d_T^2}{T+1}\right) = \lim_{T\to\infty}\left(\frac{2H^*}{(T+1)d_T^2} + \frac{2L_g T}{T+1}\right) = 2L_g,$$

Hence, there exists a sufficiently large M_g such that if $T > t^*$ then

$$g(x_T) - g_{\text{opt}} \leq \frac{2H^*}{(T+1)T} + \frac{2L_g d_T^2}{T+1} \leq \frac{M_g d_T^2}{T}.$$

□

By Theorem 3.1, we can guarantee asymptotic convergence of Algorithm 2, i.e. $\lim_{T\to\infty} g(x_T) = g_{\text{opt}}$, if $d_t = o\left(t^{1/2}\right)$.

3.3 Application to solving convex bilevel problems

In the `SL-CG` and `PD-CG` methods that we develop in Chapter 4 and Chapter 6, respectively, we will require a sequence $\{g_t\}_{t\geq 0}$ which approximates g_{opt}, in the sense that it satisfies the following assumption.

Assumption 6. Sequence $\{g_t\}_{t\geq 0}$ is non-increasing, converges to g_{opt} and there exists $Q > 0$ such that

$$0 \leq g_t - g_{\text{opt}} \leq \frac{Q d_t^2}{t+1}, \quad \forall t \geq 0.$$

We will use the `UCG` method to generate such a sequence in the following way. Given x_0 arbitrarily chosen in X, let $\{x_t\}_{t\geq 0}$ be the sequence generated by applying the `UCG` method to the inner-level objective under Assumption 1(a), Assumption 1(c), Assumption 5 and $d_t = o\left(t^{1/2}\right)$. By

Theorem 3.1, there exists a positive number Q_1 and a positive integer t_1 such that

$$g(x_t) - g_{\text{opt}} \leq \frac{Q_1 d_t^2}{t}, \quad \forall t \geq t_1.$$

We observe that

$$\frac{Q_1 d_t^2}{t} \leq \frac{2Q_1 d_t^2}{t+1}, \quad \forall t \geq t_1 \geq 1.$$

By defining

$$Q := \max\left\{2Q_1, \frac{1 \times (g(x_0) - g_{\text{opt}})}{d_0^2}, \ldots, \frac{t_1(g(x_{t_1-1}) - g_{\text{opt}})}{d_{t_1-1}^2}\right\} > 0,$$

we have that

$$g(x_t) - g_{\text{opt}} \leq \frac{Q d_t^2}{t+1}, \quad \forall t \geq 0.$$

Therefore, if we define $g_t := \min_{0 \leq i \leq t} g(x_i)$ for each $t \geq 0$, then $g_t \to g_{\text{opt}}$ as $t \to \infty$, and for $t \geq 0$, $g_t \geq g_{t+1}$ as well as

$$g_t - g_{\text{opt}} \leq g(x_t) - g_{\text{opt}} \leq \frac{Q d_t^2}{t+1}.$$

Furthermore, the idea of using the sequence of coverings $\{B_t\}_{t \geq 0}$ satisfying Assumption 5 can be applied in the proposed methods in Chapters 4 to 6 to account for the unbounded domain.

Chapter 4

Sublevel linearising conditional gradient method

In this chapter, we improve the `CG-BiO` method [24] that employs *outer approximations* $\{X_t\}_{t\geq 0}$ such that $X \supseteq X_t \supseteq X_{\text{opt}}$ for each $t \geq 0$. In Section 4.1, we describe the motivation for the approximation strategy and provide the algorithm. In Section 4.2, we establish the convergence rates for the method.

4.1 Approach and method description

If we viewed problem (1.3) as merely a single-level optimisation problem and had access to X_{opt}, which is possibly unbounded, then the `UCG` method applied to problem (1.1) would compute

$$v_t \in \operatorname*{arg\,min}_{v \in X_{\text{opt}} \cap B_t} \nabla f(x_t)^\top v,$$

then $x_{t+1} := x_t + \alpha_t(v_t - x_t)$. However, since in general we do not have an explicit description of X_{opt}, at each iteration t we will replace X_{opt} with a tractable outer approximation X_t such that $X_{\text{opt}} \subseteq X_t \subseteq X$, thus v_t is computed as $v_t := \arg\min_{v \in X_t \cap B_t} \nabla f(x_t)^\top v$ if the intersection $X_t \cap B_t$

Algorithm 3: Sublevel linearising conditional gradient method (SL-CG).

Data: Parameters $\{\alpha_t\}_{t\geq 0} \subseteq [0,1]$, $\{B_t\}_{t\geq 0}$ satisfying Assumption 5, $\{X_t\}_{t\geq 0}$ such that $X \supseteq X_t \supseteq X_{\text{opt}}$ for each $t \geq 0$, number of iterations T.
Result: Sequence $\{x_t\}_{t\in[T]}$.
Initialise $x_0 \in X \cap B_0$;
for $t = 0, 1, \ldots, T-1$ **do**
  if $X_t \cap B_t = \emptyset$ **then**
    Compute $v_t := x_t$
  else
    Compute $v_t \in \arg\min_{v\in X_t\cap B_t} \nabla f(x_t)^\top v$
  Compute $x_{t+1} \leftarrow x_t + \alpha_t(v_t - x_t)$.

is non-empty. Otherwise, we set $v_t := x_t$. A precise description is provided in Algorithm 3.

Remark 4.1.1. By Assumption 5, given a solution x_{opt} of problem (1.1), there exists a sufficiently large t_0 such that $x_{\text{opt}} \in B_t$ for $t \geq t_0$. We also have that $X_{\text{opt}} \subseteq X_t$ from the assumption. Hence, for $t \geq t_0$, we ensure $X_t \cap B_t \neq \emptyset$. ■

4.2 Convergence analysis

We devote this section to provide the formal description of X_t and establish the convergence rates for Algorithm 3.

First, we will motivate how certain choices of X_t can ensure convergence, as well as the differences between our X_t and that of Jiang et al. [24]. Suppose we are at iteration t of Algorithm 3. Given $t \geq 0$, from Lemma 2.2 and the fact that $\|v_t - x_t\| \leq d_t$ as $v_t, x_t \in B_t$, we have

$$f(x_{t+1}) - f(x) \leq (1-\alpha_t)(f(x_t) - f(x)) + \alpha_t(\nabla f(x_t)^\top (v_t - x_t) + f(x_t) - f(x)) + \alpha_t^2 \frac{L_f d_t^2}{2}, \quad (4.1)$$

for any $x \in X$. Since $X_{\text{opt}} \subseteq X_t$ by assumption, whenever $t \geq t_0$, we obtain

$$\nabla f(x_t)^\top (v_t - x_t) + f(x_t) - f(x_{\text{opt}}) \leq \nabla f(x_t)^\top (x_{\text{opt}} - x_t) + f(x_t) - f(x_{\text{opt}}) \leq 0, \quad (4.2)$$

according to the definition of v_t, the convexity of f, and Assumption 5. At this point, utilising

Corollary 2.4 enables us to prove the following bound on the outer-level objective.

Lemma 4.1. *Suppose $\{x_t\}_{t\in[T]}$ is the sequence generated by Algorithm 3 with stepsizes $\alpha_t = \frac{2}{t+2}$ for each $t \geq 0$. If Assumption 1 and Assumption 5 hold, then*

$$f(x_T) - f_{\text{opt}} \leq O\left(\frac{d_T^2}{T}\right).$$

Proof. From (4.1) and (4.2), we have

$$f(x_{t+1}) - f_{\text{opt}} \leq \left(1 - \frac{2}{t+2}\right)(f(x_t) - f_{\text{opt}}) + \frac{2L_f d_t^2}{(t+2)^2}, \quad \forall t \geq t_0.$$

By Corollary 2.4, if $T > t_0$ then there exists a constant H_0 such that

$$f(x_T) - f_{\text{opt}} \leq \frac{2H_0}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}(t+1)t\frac{2L_f d_{t-1}^2}{(t+1)^2} = \frac{2H_0}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}\frac{2L_f d_{t-1}^2 t}{t+1}.$$

Since $\frac{t}{t+1} < 1$ and $d_T \geq d_{t-1}$ for $t \in [T]$, we have

$$f(x_T) - f_{\text{opt}} \leq \frac{2H_0}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]} 2L_f d_T^2 = \frac{2H_0}{(T+1)T} + \frac{2L_f d_T^2}{T+1},$$

Since $d_T \geq d_0 > 0$, we have that $\lim_{T\to\infty}(T+1)d_T^2 = \infty$ and hence,

$$\lim_{T\to\infty}\frac{T}{d_T^2}\left(\frac{2H_0}{(T+1)T} + \frac{2L_f d_T^2}{T+1}\right) = \lim_{T\to\infty}\left(\frac{2H_0}{(T+1)d_T^2} + \frac{2L_f T}{T+1}\right) = 2L_f < \infty.$$

Thus, there exists a sufficiently large M_f such that if $T > t_0$ then

$$f(x_T) - f_{\text{opt}} \leq \frac{2H_0}{(T+1)T} + \frac{2L_f d_T^2}{T+1} \leq \frac{M_f d_T^2}{T}.$$

□

To claim asymptotic convergence for Algorithm 3, Lemma 2.5 suggests we need $\limsup_{T\to\infty} f(x_T) \leq f_{\text{opt}}$. To guarantee this, we impose Condition 4.1 stated below on the growth of $\{d_t\}_{t\geq 0}$, thus

Lemma 4.1 ensures that it holds.

Condition 4.1. Sequence $\{B_t\}_{t\geq 0}$ satisfies $d_t = o(t^{1/2})$.

For the inner-level objective, we obtain through Assumption 1(a), Assumption 1(c), and Lemma 2.2 the following inequality:

$$g(x_{t+1})-g_{\text{opt}} \leq (1-\alpha_t)(g(x_t)-g_{\text{opt}})+\alpha_t(\nabla g(x_t)^\top(v_t-x_t)+g(x_t)-g_{\text{opt}})+\alpha_t^2\frac{L_g d_t^2}{2}, \quad \forall t \geq 0. \tag{4.3}$$

We define

$$\tilde{g}_t := g(x_t) + \nabla g(x_t)^\top(v_t - x_t), \quad \forall t \geq 0. \tag{4.4}$$

The following lemma shows how these terms bound the inner-level objective.

Lemma 4.2. *Suppose $\{x_t\}_{t\in[T]}$ is the sequence generated by Algorithm 3 with stepsizes $\alpha_t = \frac{2}{t+2}$ for each $t \geq 0$. If Assumption 1 and Assumption 5 hold, then*

$$g(x_T) - g_{\text{opt}} \leq \frac{2}{(T+1)T}\sum_{t\in[T]} t(\tilde{g}_{t-1} - g_{\text{opt}}) + \frac{2L_g d_T^2}{T+1}. \tag{4.5}$$

Proof. From (4.3) and (4.4), we have that

$$g(x_{t+1}) - g_{\text{opt}} \leq \left(1 - \frac{2}{t+2}\right)(g(x_t) - g_{\text{opt}}) + \left[\frac{2}{t+2}(\tilde{g}_t - g_{\text{opt}}) + \frac{2L_g d_t^2}{(t+2)^2}\right], \quad \forall t \geq 0. \tag{4.6}$$

By applying Corollary 2.4 for the recursion (4.6), there exists a constant H_0' such that

$$\begin{aligned} g(x_T) - g_{\text{opt}} &\leq \frac{2H_0'}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}(t+1)t\left(\frac{2}{t+1}(\tilde{g}_{t-1} - g_{\text{opt}}) + \frac{2L_g d_{t-1}^2}{(t+1)^2}\right) \\ &= \frac{2H_0'}{(T+1)T} + \frac{2}{(T+1)T}\sum_{t\in[T]}\left(t(\tilde{g}_{t-1} - g_{\text{opt}}) + L_g d_{t-1}^2\frac{t}{t+1}\right). \end{aligned}$$

By using $\frac{t}{t+1} < 1$ and $d_{t-1} \leq d_T$ for $t \in [T]$, we have

$$g(x_T) - g_{\text{opt}} \leq \frac{2H_0'}{(T+1)T} + \frac{2}{(T+1)T}\sum_{t\in[T]} t(\tilde{g}_{t-1} - g_{\text{opt}}) + \frac{2L_g d_T^2}{T+1}.$$

Since $\alpha_0 = \frac{2}{0+2} = 1$ then from Lemma 2.3, we can choose $H'_0 = 0$. □

To ensure $g(x_t) \to g_{opt}$ as $t \to \infty$, it is sufficient to require the superior limit of the right-hand side of (4.5) is 0. Since Condition 4.1 already implies that $\frac{2L_g d_T^2}{T+1} \to 0$ as $T \to \infty$, we require

$$\limsup_{T\to\infty} \frac{2}{(T+1)T} \sum_{t\in[T]} t(\tilde{g}_{t-1} - g_{\text{opt}}) \leq 0.$$

This can be ensured if

$$\limsup_{T\to\infty} \tilde{g}_T \leq g_{\text{opt}}. \tag{4.7}$$

To justify this claim, we first introduce to following calculus result.

Lemma 4.3. *Let $\{u_t\}_{t\geq 1}$ and $\{v_t\}_{t\geq 1}$ be two sequences of real numbers. If $\{v_t\}_{t\geq 1}$ is strictly increasing and divergent, then*

$$\limsup_{t\to\infty} \frac{u_t}{v_t} \leq \limsup_{t\to\infty} \frac{u_{t+1} - u_t}{v_{t+1} - v_t}. \tag{4.8}$$

Proof. If

$$\limsup_{t\to\infty} \frac{u_{t+1} - u_t}{v_{t+1} - v_t} = \infty,$$

then (4.8) is true. Otherwise, given $c \in \mathsf{R}$ such that

$$\limsup_{t\to\infty} \frac{u_{t+1} - u_t}{v_{t+1} - v_t} < c, \tag{4.9}$$

there exists a positive integer t_c such that for any $t \geq t_c$

$$\frac{u_{t+1} - u_t}{v_{t+1} - v_t} < c \iff u_{t+1} - u_t < c\,(v_{t+1} - v_t)\,,$$

where the equivalence is true as $v_{t+1} > v_t$ for $t \geq 1$. Thus, for any $t > t_c$

$$u_t - u_{t_c} = \sum_{i=t_c}^{t-1} (u_{i+1} - u_i) < c \sum_{i=t_c}^{t-1} (v_{i+1} - v_i) = c(v_t - v_{t_c}).$$

Using $v_t \to \infty$, we that for sufficiently large t, $v_t > 0$. Therefore, we obtain

$$\limsup_{t\to\infty} \frac{u_t}{v_t} \leq \limsup_{t\to\infty} \frac{u_{t_c} + c(v_t - v_{t_c})}{v_t} = c.$$

Since c is chosen arbitrarily as long as c satisfies (4.9), (4.8) must be true. □

If (4.7) is true, then Lemma 4.3 implies that

$$\begin{aligned} \limsup_{T\to\infty} \frac{2}{(T+1)T} \sum_{t\in[T]} t(\tilde{g}_{t-1} - g_{\text{opt}}) &\leq \limsup_{T\to\infty} \frac{2(T+1)(\tilde{g}_T - g_{\text{opt}})}{(T+2)(T+1) - (T+1)T} \\ &= \limsup_{T\to\infty} (\tilde{g}_T - g_{\text{opt}}) \\ &\leq 0. \end{aligned} \tag{4.10}$$

So far, we have shown that the conditions on the outer approximations X_t, which ensure convergence of the inner- and outer-level objectives, are that $X_{\text{opt}} \subseteq X_t \subseteq X$ and (4.7). We recall that $X_{\text{opt}} = \{x \in X \mid g(x) \leq g_{\text{opt}}\}$, and the first-order Taylor expansion of g around x_t, which is also a lower bound of g by the convexity, is

$$g(x) \geq g(x_t) + \nabla g(x_t)^\top (x - x_t), \quad \forall x \in X.$$

If we choose g_t to be a computable upper bound of g_{opt} then any $x \in X$ satisfies the constraint $g(x) \leq g_{\text{opt}}$ then g_t must also meet the following inequality:

$$g(x_t) + \nabla g(x_t)^\top (x - x_t) \leq g_t.$$

Therefore, if we define X_t given x_t as follows:

$$X_t := \{x \in X \mid \nabla g(x_t) + g(x_t)^T (x - x_t) \leq g_t\}, \tag{4.11}$$

then it is true that $X_{\text{opt}} \subseteq X_t$ according to the above discussion. For each $t \geq t_0$, since $v_t \in X_t$, we

observe that

$$\tilde{g}_t = g(x_t) + \nabla g(x_t)^\top (v_t - x_t) \leq g_t, \quad \forall t \geq t_0.$$

Hence, to ensure (4.7) holds, we adopt Assumption 6 for sequence $\{g_t\}_{t\geq 0}$.

Remark 4.2.1. The `CG-BiO` method [24, Algorithm 1] solves problem (1.1) in which X is bounded with diameter D and chooses a constant schedule for $\{g_t\}_{t\geq 0}$, i.e., $g_t = g_0$ for each $t \geq 0$ where $g_0 - g_{\text{opt}} \leq \frac{\epsilon_g}{2}$ and $\epsilon_g > 0$ is a tolerance parameter. This choice does not satisfy Assumption 6 if $g_0 > g_{\text{opt}}$. Hence, there is no guarantee of asymptotic convergence of their method. Due to the boundedness of X assumed by Jiang et al. [24], t_0 defined in Remark 4.1.1 is 0 for their method. Then Lemma 4.2 implies the bound guaranteed is

$$g(x_T) - g_{\text{opt}} \leq \frac{2L_g D^2}{T+1} + \frac{2}{(T+1)T} \sum_{t\in[T]} t\frac{\epsilon_g}{2} = \frac{2L_g D^2}{T+1} + \frac{\epsilon_g}{2},$$

where in the last equality, we use the fact that $\sum_{t\in[T]} t = \frac{T(T+1)}{2}$. Thus, asymptotic convergence for `CG-BiO` cannot be claimed. ■

Finally, the following theorem establishes convergence rates for Algorithm 3.

Theorem 4.4. *Suppose $\{x_t\}_{t\in[T]}$ is the sequence generated by Algorithm 3 with stepsizes $\alpha_t = \frac{2}{t+2}$ for each $t \geq 0$ and outer approximations $\{X_t\}_{t\geq 0}$ as defined in (4.11). If Assumption 1, Assumptions 5–6 and Condition 4.1 hold, then*

$$f(x_T) - f_{\text{opt}} \leq O\left(\frac{d_T^2}{T}\right), \quad g(x_T) - g_{\text{opt}} \leq O\left(\frac{d_T^2}{T}\right).$$

Proof. From Assumption 6, we have that

$$g_t - g_{\text{opt}} \leq Q\frac{d_t^2}{t+1}, \quad \forall t \geq 0.$$

By Assumption 5, if $T > t_0$, then

$$\begin{aligned}
\frac{2}{(T+1)T}\sum_{t\in[T]} t(\tilde{g}_{t-1} - g_{\text{opt}}) &\leq \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + \sum_{t=t_0}^{T-1}(t+1)(g_t - g_{\text{opt}})\right) \\
&\leq \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q\sum_{t=t_0}^{T-1}(t+1)\frac{d_t^2}{(t+1)}\right) \\
&\leq \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q\sum_{t=t_0}^{T-1}(t+1)\frac{d_T^2}{(t+1)}\right) \\
&= \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q(T-t_0)d_T^2\right)
\end{aligned}$$

Hence, by Lemma 4.2, we have that

$$g(x_T) - g_{\text{opt}} \leq \frac{2L_g d_T^2}{T+1} + \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q(T-t_0)d_T^2\right).$$

Since $d_T \geq d_0 > 0$, we have $\lim_{T\to\infty}(T+1)d_T^2 = \infty$ and hence,

$$\begin{aligned}
&\lim_{T\to\infty}\frac{T}{d_T^2}\left(\frac{2L_g d_T^2}{T+1} + \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q(T-t_0)d_T^2\right)\right) \\
&= \lim_{T\to\infty}\left(\frac{2L_g T}{T+1} + \frac{2}{(T+1)d_T^2}\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + \frac{2Q(T-t_0)}{T+1}\right) \\
&= 2L_g + 2Q.
\end{aligned}$$

Thus, there exists a sufficiently large $M_g > 0$ such that if $T > t_0$, then

$$g(x_T) - g_{\text{opt}} \leq \frac{2L_g d_T^2}{T+1} + \frac{2}{(T+1)T}\left(\sum_{t=0}^{t_0-1}(t+1)(\tilde{g}_t - g_{\text{opt}}) + Q(T-t_0)d_T^2\right) \leq \frac{M_g d_T^2}{T}.$$

Using this result and Lemma 4.1, we finish the proof. □

Chapter 5

Iteratively regularised conditional gradient method

In this chapter, we describe a regularisation scheme to solve problem (1.3). In Section 6.1, we discuss the motivation for the approach and provide the algorithm. In Section 5.2, we derive the convergence rates for the method.

5.1 Approach and method description

Since problem (1.3) is a convex optimisation problem with a functional constraint, we consider the Lagrangian

$$L(x, \lambda) := f(x) + \lambda(g(x) - g_{\text{opt}}), \quad x \in X, \lambda \geq 0,$$

and the Lagrangian dual problem (2.12).

Recall that in the proof of Lemma 2.8 in Section 2.4, we showed that if $x_t \in \arg\min_{x \in X} L(x, \lambda_t)$ then $\lim_{t\to\infty} g(x_t) = g_{\text{opt}}$ and $\lim_{t\to\infty} f(x_t) = f_{\text{opt}}$, and also consequently that strong duality holds. Notice that, while $L(x, \lambda)$ contains g_{opt}, which generally is unknown a priori, obtaining x_t does not require knowledge of g_{opt} at all. Since getting x_t by optimising $L(x, \lambda_t)$ may be expensive, Solodov

Algorithm 4: Iteratively regularised conditional gradient method (`IR-CG`).

Data: Parameters $\{\alpha_t\}_{t\geq 0} \subseteq [0,1]$, $\{\sigma_t\}_{t\geq 0} \subseteq \mathsf{R}_{++}$, $\{B_t\}_{t\geq 0}$ satisfying Assumption 5, number of iterations T.

Result: Sequence $\{z_t\}_{t\in[T]}$.

Initialise $x_0 \in X \cap B_0$;

for $t = 0, 1, \ldots, T-1$ **do**

Compute

$$v_t \in \operatorname*{arg\,min}_{v\in X\cap B_t} (\sigma_t \nabla f(x_t) + \nabla g(x_t))^\top v$$

$$x_{t+1} := x_t + \alpha_t(v_t - x_t)$$

$$S_{t+1} := (t+2)(t+1)\sigma_{t+1} + \sum_{i\in[t+1]} (i+1)i(\sigma_{i-1} - \sigma_i) \tag{5.1}$$

$$z_{t+1} := \frac{(t+2)(t+1)\sigma_{t+1}x_{t+1} + \sum_{i\in[t+1]}(i+1)i(\sigma_{i-1}-\sigma_i)x_i}{S_{t+1}}. \tag{5.2}$$

[34] proposed simply performing a single projected gradient step via Euclidean norm to obtain x_{t+1}, i.e.,

$$x_{t+1} = \operatorname{Proj}_X\left(x_t - \frac{\alpha_t}{\lambda_t}\nabla_x L(x_t, \lambda_t)\right) = \operatorname{Proj}_X\left(x_t - \alpha_t\left(\frac{1}{\lambda_t}\nabla f(x_t) + \nabla g(x_t)\right)\right).$$

Solodov [34, Theorem 3.2] shows that if $\lambda_t \to \infty$ sufficiently slowly (in the sense that $\sum_{t=0}^{\infty} \frac{1}{\lambda_t} = \infty$) then the sequence x_t converges to the optimal solution set of (1.3).

Inspired by the results of Solodov [34], we propose what we call the *iteratively regularised conditional gradient* (`IR-CG`) method, outlined in Algorithm 4 below, which essentially replaces the projection step with an unbounded conditional gradient-type step. To simplify the analysis, it is convenient to define $\sigma_t := \frac{1}{\lambda_t}$ and

$$\Phi_t(x) := \frac{1}{\lambda_t} L(x, \lambda_t) + g_{\text{opt}} = \sigma_t f(x) + g(x).$$

Based on the discussion above, we impose the following on the sequence $\{\sigma_t\}_{t\geq 0}$.

Condition 5.1. The sequence $\{\sigma_t\}_{t\geq 0}$ is strictly decreasing, positive, and converges to 0.

Remark 5.1.1. From the definition of $\{S_t\}_{t\geq 1}$ in (5.1), for any $t \geq 1$, we have that

$$\begin{aligned} S_{t+1} - (t+2)(t+1)\sigma_{t+1} &= S_t - (t+1)t\sigma_t + (t+2)(t+1)(\sigma_t - \sigma_{t+1}) \\ \iff S_{t+1} &= S_t + 2(t+1)\sigma_t. \end{aligned}$$

When $t = 1$, we have $S_1 = 2\sigma_1 + 2(\sigma_0 - \sigma_1) = 2\sigma_0$. In fact, by substituting $t = 0$ to the recursion above, the computed S_1 is $S_0 + 2\sigma_0$, which implies we can define $S_0 := 0$. Similarly, from (5.2), for any $t \geq 1$, we have that

$$\begin{aligned} S_{t+1}z_{t+1} - (t+2)(t+1)\sigma_{t+1}x_{t+1} &= S_t z_t - (t+1)t\sigma_t x_t + (t+2)(t+1)(\sigma_t - \sigma_{t+1})x_{t+1} \\ \iff S_{t+1}z_{t+1} &= S_t z_t - (t+1)t\sigma_t x_t + (t+2)(t+1)\sigma_t x_{t+1} \\ \iff z_{t+1} &= \frac{S_t z_t - (t+1)t\sigma_t x_t + (t+2)(t+1)\sigma_t x_{t+1}}{S_{t+1}}. \end{aligned}$$

We have from the definition of z_1 in (5.2) that

$$z_1 = \frac{1}{S_1}(2\sigma_1 x_1 + 2(\sigma_0 - \sigma_1)x_1) = \frac{2\sigma_0 x_1}{2\sigma_0} = x_1.$$

When we substitute $t = 0$ to the recursion for $\{z_t\}_{t\geq 1}$ above, the computed z_1 agrees with the value computed from (5.2) for any value of z_0. Thus, we can define $z_0 := 0$ without loss of generality. Therefore, (5.1) and (5.2) can efficiently be computed using recursive formulae as follows: for any $t \geq 0$,

$$\begin{aligned} S_0 &= 0, & S_{t+1} &= S_t + 2(t+1)\sigma_t, \\ z_0 &= 0, & z_{t+1} &= \frac{S_t z_t - (t+1)t\sigma_t x_t + (t+2)(t+1)\sigma_t x_{t+1}}{S_{t+1}}. \end{aligned}$$

■

A critical difference from a typical conditional gradient-type algorithm in the analysis is that, instead of proving convergence for the sequence $\{x_t\}_{t\geq 0}$, we will show convergence for the sequence

$\{z_t\}_{t\geq 1}$. The weights of the convex combination arise naturally through applying Lemmas 2.2 to 2.3 and Corollary 2.4 to the conditional gradient step in Algorithm 4. By recalling from Remark 4.1.1 that $t_0 \geq 0$ is an integer such that there exists an optimal solution x_{opt} of problem (1.1) such that $x_{\text{opt}} \in B_t$ for each $t \geq t_0$, we state this formally in the following lemma.

Lemma 5.1. *Suppose $\{x_t\}_{0\leq t\leq T}$ are iterates generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1, Assumption 5, Condition 5.1 hold and $T > t_0$, then there exists a constant H_0 such that*

$$(T+1)T\left(g(x_T) - g_{\text{opt}}\right) + (T+1)T\sigma_T(f(x_T) - f_{\text{opt}}) + \sum_{t\in[T]}(t+1)t(\sigma_{t-1} - \sigma_t)(f(x_t) - f_{\text{opt}})$$
$$\leq 2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2.$$

Proof. Using the convexity of Φ_t and the definition of v_t, we have that

$$\nabla\Phi_t(x_t)^\top(v_t - x_t) \leq \nabla\Phi_t(x_t)^\top(x_{\text{opt}} - x_t) \leq \Phi_t(x_{\text{opt}}) - \Phi_t(x_t), \quad \forall t \geq t_0.$$

By Lemma 2.2 and the above result, we have that

$$\Phi_t(x_{t+1}) - \Phi_t(x_{\text{opt}}) \leq \left(1 - \frac{2}{t+2}\right)(\Phi_t(x_t) - \Phi_t(x_{\text{opt}})) + \frac{2(L_f\sigma_t + L_g)d_t^2}{(t+2)^2}, \quad \forall t \geq t_0,$$

which implies

$$\begin{aligned}\Phi_{t+1}(x_{t+1}) - \Phi_{t+1}(x_{\text{opt}}) \leq \left(1 - \frac{2}{t+2}\right)(\Phi_t(x_t) - \Phi_t(x_{\text{opt}})) + \frac{2(L_f\sigma_t + L_g)d_t^2}{(t+2)^2} \\ - (\sigma_t - \sigma_{t+1})(f(x_{t+1}) - f_{\text{opt}}),\end{aligned} \tag{5.3}$$

since

$$\begin{aligned}\Phi_t(x_{t+1}) - \Phi_t(x_{\text{opt}}) &= \sigma_{t+1}(f(x_{t+1}) - f_{\text{opt}}) + g(x_{t+1}) - g_{\text{opt}} + (\sigma_t - \sigma_{t+1})(f(x_{t+1}) - f_{\text{opt}}) \\ &= \Phi_{t+1}(x_{t+1}) - \Phi_{t+1}(x_{\text{opt}}) + (\sigma_t - \sigma_{t+1})(f(x_{t+1}) - f_{\text{opt}}).\end{aligned}$$

Applying Corollary 2.4 to (5.3) and using the fact that $\frac{t}{t+1} < 1$, there exists H_0 such that

$$\begin{aligned}&\Phi_T(x_T) - \Phi_T(x_{\text{opt}}) \\ &\leq \frac{2H_0}{(T+1)T} + \frac{1}{(T+1)T}\sum_{t\in[T]}\left(2(\sigma_{t-1}L_f + L_g)d_{t-1}^2 - (t+1)t(\sigma_{t-1}-\sigma_t)(f(x_t) - f_{\text{opt}})\right).\end{aligned} \tag{5.4}$$

Given $t \in [T]$, we have that $\sigma_{t-1} \leq \sigma_0$, $d_{t-1} \leq d_T$ by Condition 5.1 and Assumption 5. Thus, we obtain the inequality claimed in this lemma. □

From the convexity of f, we have that

$$f(z_T) - f_{\text{opt}} \leq \frac{2(L_f\sigma_0 + L_g)Td_T^2 + 2H_0}{S_T}. \tag{5.5}$$

In the next section, we will show that $g(z_T) \to g_{\text{opt}}$ and the right-hand side term of (5.5) converges to 0 as $T \to \infty$ under appropriate choice of $\{\sigma_t, B_t\}_{t\geq 0}$, which are then sufficient to apply Lemma 2.5 to guarantee asymptotic convergence of $\{z_t\}_{t\geq 1}$.

5.2 Convergence analysis

To establish convergence for Algorithm 4, we need to impose further conditions on the regularisation parameters and the coverings $\{\sigma_t, B_t\}_{t\geq 0}$, which are stated below.

Condition 5.2. Sequence $\{(t+1)\sigma_t\}_{t\geq 0}$ is strictly increasing and divergent.

Condition 5.3. There exists $L \in \mathsf{R}$ such that $L := \lim_{t\to\infty} t\left(\frac{\sigma_t}{\sigma_{t+1}} - 1\right)$.

Condition 5.4. Sequences $\{B_t, \sigma_t\}_{t\geq 0}$ satisfy $\frac{d_t^2}{\sigma_t} = o(t)$.

To prove some results in this section, which involve running sum, we utilise Lemma 5.2 stated below.

Lemma 5.2 ([8, 2.7.1 Theorem, 2.7.2 Theorem])**.** *Let $\{u_t\}_{t\geq 1}$ and $\{v_t\}_{t\geq 1}$ be two sequences of real*

numbers. If $\{v_t\}_{t\geq 1}$ *is strictly increasing and divergent, and*

$$\lim_{t\to\infty} \frac{u_{t+1}-u_t}{v_{t+1}-v_t} = l,$$

for $l \in \mathsf{R}\cup\{\pm\infty\}$*, then*

$$\lim_{t\to\infty} \frac{u_t}{v_t} = l.$$

Below, we provide some consequences of Conditions 5.1–5.3, which will be used extensively in this section to analyse Algorithm 4.

Lemma 5.3. *If Conditions 5.1{5.3 hold, then we have* $0 \leq L \leq 1$*,*

$$\lim_{T\to\infty} \frac{\sigma_T}{\sigma_{T+1}} = 1, \quad \text{and} \quad \lim_{t\to\infty} \frac{(t+2)(\sigma_t-\sigma_{t+1})}{(t+2)\sigma_{t+1}-t\sigma_t} = \frac{L}{2-L} \geq 0.$$

Proof. From Condition 5.1, we have that $L \geq 0$. If $L > 1$, then for sufficiently large t, we have

$$t\left(\frac{\sigma_t}{\sigma_{t+1}}-1\right) > 1 \iff \frac{\sigma_t}{\sigma_{t+1}} > \frac{t+1}{t} \iff (t+1)\sigma_{t+1} < t\sigma_t,$$

which contradicts to Condition 5.2. For the second claim, we have that

$$\lim_{T\to\infty}\left(\frac{\sigma_T}{\sigma_{T+1}}-1\right) = \lim_{T\to\infty}\frac{1}{T}T\left(\frac{\sigma_T}{\sigma_{T+1}}-1\right) = 0.$$

Turning to the third claim, by Condition 5.3, we have that

$$\lim_{t\to\infty}\frac{(t+2)(\sigma_t-\sigma_{t+1})}{(t+2)\sigma_{t+1}-t\sigma_t} = \lim_{t\to\infty}\frac{t+2}{t}\frac{t\left(\frac{\sigma_t}{\sigma_{t+1}}-1\right)}{2-t\left(\frac{\sigma_t}{\sigma_{t+1}}-1\right)} = \frac{L}{2-L} \geq 0.$$

□

Lemma 5.4. *If Conditions 5.1{5.3 hold, then we have*

$$\lim_{T\to\infty}\frac{1}{(T+1)T\sigma_T^2}\sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t)\sigma_t = \frac{L}{2(1-L)} \geq 0.$$

When $L = 1$, the right-hand side is ∞.

Proof. From Conditions 5.1–5.2, $\{t\sigma_t\}_{t\geq 0}$ is increasing since $t\sigma_t = (t+1)\sigma_t - \sigma_t$ for each $t \geq 0$. Accordingly, $\{(t+1)t\sigma_t^2\}_{t\geq 0}$ is increasing and diverges to ∞. Using Lemmas 5.2 to 5.3, we have that

$$\begin{aligned}
&\lim_{T\to\infty} \frac{1}{(T+1)T\sigma_T^2} \sum_{t\in[T]} (t+1)t(\sigma_{t-1} - \sigma_t)\sigma_t \\
&= \lim_{T\to\infty} \frac{(T+2)(T+1)(\sigma_T - \sigma_{T+1})\sigma_{T+1}}{(T+2)(T+1)\sigma_{T+1}^2 - (T+1)T\sigma_T^2} \\
&= \lim_{T\to\infty} \frac{(T+2)(\sigma_T - \sigma_{T+1})\sigma_{T+1}}{(T+2)\sigma_{T+1}^2 - T\sigma_T^2} \\
&= \lim_{T\to\infty} \frac{(T+2)(\sigma_T - \sigma_{T+1})\sigma_{T+1}}{(T+2)\sigma_{T+1}^2 + (T+2)\sigma_T\sigma_{T+1} - T\sigma_T\sigma_{T+1} - T\sigma_T^2 - 2\sigma_T\sigma_{T+1}} \\
&= \lim_{T\to\infty} \frac{(T+2)(\sigma_T - \sigma_{T+1})\sigma_{T+1}}{((T+2)\sigma_{T+1} - T\sigma_T)(\sigma_T + \sigma_{T+1}) - 2\sigma_T\sigma_{T+1}} \\
&= \lim_{T\to\infty} \frac{\left(\frac{T+2}{T}\right) T\left(\frac{\sigma_T}{\sigma_{T+1}} - 1\right)}{\left(2 - T\left(\frac{\sigma_T}{\sigma_{T+1}} - 1\right)\right)\left(\frac{\sigma_T}{\sigma_{T+1}} + 1\right) - 2\frac{\sigma_T}{\sigma_{T+1}}} = \frac{L}{(2-L)2-2} = \frac{L}{2(1-L)}.
\end{aligned}$$

□

Lemma 5.5 provides an $o(1)$ upper bound on $g(x_T) - g_{\text{opt}}$, which is then used in Lemma 5.6 to bound $g(z_T) - g_{\text{opt}}$.

Lemma 5.5. *Suppose $\{x_t\}_{0\leq t\leq T}$ are iterates generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1, Assumption 3, Assumption 5, and Conditions 5.1{5.4 hold, then*

$$g(x_T) - g_{\text{opt}} \leq C\sigma_T,$$

for some constant $C > 0$.

Proof. Based on Assumption 3, we define $F := f_{\text{opt}} - \underline{f}$, which is non-negative and finite. By Condition 5.1, and inequality (5.4), we have if $T > t_0$, then

$$\frac{\sigma_T(f(x_T) - f_{\text{opt}}) + g(x_T) - g_{\text{opt}}}{\sigma_T}$$

$$\leq \frac{1}{(T+1)T\sigma_T}\left(2H_0 + 2(\sigma_0 L_f + L_g)Td_T^2 + F\sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t)\right).$$

We will prove that the right-hand side term has a finite limit as $T \to \infty$ under Condition 5.4. By Conditions 5.2–5.3, and Lemmas 5.2 to 5.3, we observe that

$$\begin{aligned}\lim_{T\to\infty}\frac{1}{(T+1)T\sigma_T}\sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t) &= \lim_{T\to\infty}\frac{(T+2)(T+1)(\sigma_T-\sigma_{T+1})}{(T+2)(T+1)\sigma_{T+1}-(T+1)T\sigma_T}\\ &= \lim_{T\to\infty}\frac{(T+2)(\sigma_T-\sigma_{T+1})}{(T+2)\sigma_{T+1}-T\sigma_T}\\ &= \frac{L}{2-L},\end{aligned}$$

Using Condition 5.2 and Condition 5.4, we have

$$\frac{2H_0 + 2(\sigma_0 L_f + L_g)Td_T^2}{(T+1)T\sigma_T} = \frac{2H_0}{(T+1)T\sigma_T} + \frac{2(\sigma_0 L_f + L_g)d_T^2}{(T+1)\sigma_T} \to 0,$$

as $T \to \infty$. Hence, we have $\limsup_{T\to\infty}\frac{\Delta_T}{\sigma_T} < \infty$, which implies there exists a sufficiently large $U > 0$ such that $\Delta_T \leq U\sigma_T$ for $T \geq 1$. Since

$$g(x_T) - g_{\text{opt}} - F\sigma_T \leq \sigma_T(f(x_T) - f_{\text{opt}}) + g(x_T) - g_{\text{opt}} \leq U\sigma_T, \quad \forall T \geq 1,$$

we have $g(x_T) - g_{\text{opt}} \leq (F+U)\sigma_T$ for $T \geq 1$. Therefore, we can set $C := F + U$. □

Lemma 5.6. *Suppose $\{z_t\}_{t\in[T]}$ is the sequence generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1, Assumption 3, Assumption 5 and Conditions 5.1{5.4 hold , then*

$$\begin{aligned}f(z_T) - f_{\text{opt}} &\leq O\left(\frac{d_T^2}{(T+1)\sigma_T}\right),\\ g(z_T) - g_{\text{opt}} &\leq \frac{C}{(T+1)T\sigma_T}\left((T+1)T\sigma_T^2 + \sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t)\sigma_t\right),\end{aligned}$$

where C is the constant de ned in Lemma 5.5.

Proof. From (5.5), if $T > t_0$ then we have

$$f(z_T) - f_{\text{opt}} \leq \frac{2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2}{S_T} \leq \frac{2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2}{(T+1)T\sigma_T}.$$

Since $d_T \geq d_0 > 0$, we have $\lim_{T\to\infty} Td_T^2 = \infty$ and hence,

$$\lim_{T\to\infty} \frac{(T+1)\sigma_T}{d_T^2} \frac{2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2}{(T+1)T\sigma_T} = \lim_{T\to\infty} \frac{2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2}{Td_T^2} = 2(L_f\sigma_0 + L_g).$$

Therefore, there exists a sufficiently large $M_f > 0$ such that if $T > t_0$ then

$$f(z_T) - f_{\text{opt}} \leq \frac{2H_0 + 2(L_f\sigma_0 + L_g)Td_T^2}{(T+1)T\sigma_T} \leq M_f \frac{d_T^2}{(T+1)\sigma_T}.$$

Using the convexity of g and the upper bound on $g(x_T) - g_{\text{opt}}$ from Lemma 5.5, we have that

$$\begin{aligned} g(z_T) - g_{\text{opt}} &\leq \frac{1}{S_T}\left((T+1)T\sigma_T\left(g(x_T) - g_{\text{opt}}\right) + \sum_{t\in[T]}(t+1)t(\sigma_{t-1} - \sigma_t)\left(g(x_t) - g_{\text{opt}}\right)\right) \\ &\leq \frac{C}{S_T}\left((T+1)T\sigma_T^2 + \sum_{t\in[T]}(t+1)t(\sigma_{t-1} - \sigma_t)\sigma_t\right), \end{aligned}$$

which together with $S_T \geq (T+1)T\sigma_T$, implies

$$g(z_T) - g_{\text{opt}} \leq C\left(\sigma_T + \frac{1}{(T+1)T\sigma_T}\sum_{t\in[T]}(t+1)t(\sigma_{t-1} - \sigma_t)\sigma_t\right).$$

□

From Lemma 5.6, we can claim $g(z_T) \to g_{\text{opt}}$ as stated in the following lemma. Thus, if Assumption 2 also holds, then Lemma 2.5 implies asymptotic convergence of Algorithm 4.

Lemma 5.7. *Suppose $\{z_t\}_{t\in[T]}$ is the sequence generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1, Assumption 3, Assumption 5 and Conditions 5.1{5.4 hold , then we have $\lim_{T\to\infty} g(z_T) = g_{\text{opt}}$.*

Proof. Using Lemmas 5.2 to 5.3, we have that

$$\begin{aligned}\lim_{T\to\infty}\frac{1}{(T+1)T\sigma_T}\sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t)\sigma_t &= \lim_{T\to\infty}\frac{(T+2)(T+1)(\sigma_T-\sigma_{T+1})\sigma_{T+1}}{(T+2)(T+1)\sigma_{T+1}-(T+1)T\sigma_T}\\ &= \lim_{T\to\infty}\frac{(T+2)(\sigma_T-\sigma_{T+1})\sigma_{T+1}}{(T+2)\sigma_{T+1}-T\sigma_{T+1}}\\ &= \left(\frac{L}{2-L}\right)\times 0\\ &= 0.\end{aligned}$$

Therefore, by Condition 5.1 and Lemma 5.6, we obtain $\lim_{T\to\infty} g(z_T) = g_{\text{opt}}$. □

The following lemma simplifies the upper bound of $g(z_T) - g_{\text{opt}}$.

Lemma 5.8. *Suppose $\{z_t\}_{t\in[T]}$ is the sequence generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$. If Assumption 1, Assumption 3, Assumption 5, Conditions 5.1{5.4 hold, and L de ned in Condition 5.3 is strictly less than 1, then we have*

$$g(z_T) - g_{\text{opt}} \leq O(\sigma_T).$$

Proof. If $L < 1$, then from Lemma 5.4, we have that

$$\frac{1}{(T+1)T\sigma_T}\sum_{t\in[T]}(t+1)t(\sigma_{t-1}-\sigma_t)\sigma_t \leq V\sigma_T, \quad \forall T \geq 1$$

for a sufficiently large $V > 0$. Using this result as well as Lemma 5.6, we have that

$$g(z_T) - g_{\text{opt}} \leq C(1+V)\sigma_T,$$

for any $T \geq 1$, which implies $g(z_T) - g_{\text{opt}} \leq O(\sigma_T)$. □

Lemma 5.9 provides a parameter choice for $\{\sigma_t\}_{t\geq 0}$ which satisfies all required assumptions. Theorem 5.10 then establishes convergence rates of $O(1/T^p)$ and $O\left(d_T^2/T^{1-p}\right)$ for inner- and outer-level objectives, respectively with p in $(0, 1)$ and $d_t = o\left(t^{(1-p)/2}\right)$.

Lemma 5.9. *Given $p \in (0,1)$, regularisation parameters $\sigma_t = (t+1)^{-p}$ for each $t \geq 0$ satisfy Conditions 5.1{5.3, and $L = p$.*

Proof. It is true that $\{\sigma_t\}_{t\geq 0}$ is strictly decreasing and converges to zero. Given $p \in (0,1)$, $(t+1)\sigma_t = (t+1)^{1-p}$ increases as t increases and diverges to ∞ as $t \to \infty$. To validate Condition 5.3, we have that

$$L = \lim_{t\to\infty} t\left(\left(1+\frac{1}{t}\right)^p - 1\right) = \lim_{\Delta x\to 0} \frac{(1+\Delta x)^p - 1}{\Delta x} = p \in (0,1).$$

The above calculations require some justifications. Regarding the second equality, we employ the definition of the derivative of the function x^p at $x = 1$. □

Theorem 5.10. *Suppose $\{z_t\}_{t\in[T]}$ is the sequence which is generated by Algorithm 4 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$, regularisation parameters $\{\sigma_t\}_{t\geq 0}$ as given in Lemma 5.9. If Assumption 1, Assumption 3, Assumption 5, and Condition 5.4 hold, then*

$$f(z_T) - f_{\text{opt}} \leq O\left(\frac{d_T^2}{T^{1-p}}\right), \quad g(z_T) - g_{\text{opt}} \leq O\left(\frac{1}{T^p}\right).$$

Proof. The proof directly follows from Lemmas 5.6 to 5.9. □

Chapter 6

Primal-dual conditional gradient method

In this chapter, we develop a primal-dual method for solving (1.3) by again examining the Lagrangian

$$L(x, \lambda) := f(x) + \lambda(g(x) - g_{\text{opt}}), \quad x \in X, \lambda \geq 0.$$

In Section 6.1, we motivate the approach and provide the algorithm. In Section 6.2, we analyse the *approximate* Lagrangian functions, which will be the foundation to establish convergence rates in Section 6.3.

6.1 Approach and method description

Lan et al. [26] proposed a conditional gradient-type algorithm for single-level convex optimisation with functional constraints and provided convergence guarantees through bounds on the Lagrangian $L(x, \lambda)$. Our method (described fully in Algorithm 5) adapts the method of Lan et al. [26, Algorithm 2] to the bilevel setting. The crucial difference between single-level problems and our setting is that the constraint bound g_{opt} in (1.3) is not known in advance. Similar to the `SL-CG` method from

Chapter 4, we overcome this by employing an approximating sequence $\{g_t\}_{t\geq 1}$ such that $g_t \to g_{\text{opt}}$. Consequently, we will consider approximate Lagrangian functions

$$L_t(x,\lambda) := f(x) + \lambda(g(x) - g_t), \quad \forall t \geq 0, \quad x \in X, \lambda \geq 0, \tag{6.1}$$

and derive guarantees through bounds on $L_t(x,\lambda)$. Our first contribution in this chapter is to show that the analysis of Lan et al. [26] may be extended to account for the errors $g_t - g_{\text{opt}}$, and importantly, that these errors *do not accumulate* as the algorithm progresses. Another important difference between the single-level and bilevel settings is the presence (or absence) of Slater's condition: Lan et al. [26] provided guarantees under Assumption 4, i.e., that there exists a dual variable $\lambda_{\text{opt}} \geq 0$ for which minimising $f(x) + \lambda_{\text{opt}} g(x)$ will give a solution to problem (1.3). Typically, Slater constraint qualification is used to guarantee the existence of such a λ_{opt}; however, this is never satisfied in the bilevel setting. That said, strong duality with a solvable dual problem may still hold for bilevel optimisation under other (weaker) qualification conditions such as the one discussed in Section 2.4. Therefore, our second contribution in this section is to show that our method converges both *with* and *without* Assumption 4. Naturally, the convergence rate improves when λ_{opt} exists.

We now describe the *primal-dual conditional gradient* (`PD-CG`) method and highlight our adaptations to the bilevel setting. Similar to the `IR-CG` method, our primal updates will be of the form $x_{t+1} = x_t + \alpha_t(v_t - x_t)$, where $\alpha_t \in [0,1]$ and

$$v_t = \underset{v \in X \cap B_t}{\arg\min} \left(\nabla f(x_t) + u_t \nabla g(x_t)\right)^\top v,$$

for some chosen dual multiplier u_t. In typical primal-dual algorithms, we update the dual multiplier based on $\nabla_\lambda L(x_t, u_t) = g(x_t) - g_{\text{opt}}$, usually through a gradient ascent step. Since we approximate g_{opt} with g_t, we may consider instead $\nabla_\lambda L_t(x_t, u_t) = g(x_t) - g_t$. However, Lan et al. [26] proposed a different approach, modifying this in three ways:

- First, we consider $\nabla_\lambda L_t(v_t, u_t) = g(v_t) - g_t$ instead of $\nabla_\lambda L_t(x_t, u_t)$.

- Second, we consider the linear approximation of $g(v_t) - g_t$, taken at the point x_t, namely we replace $g(v_t)$ with $g(x_t) + \nabla g(x_t)^\top (v_t - x_t)$. To this end, we define

$$l_t(x, y) := g(x) + \nabla g(x)^\top (y - x) - g_t, \tag{6.2}$$

  and the quantity of interest is $l_t(x_t, v_t)$.

- Third, instead of just utilising the previous step $l_t(x_t, v_t)$, we consider an extrapolated term consisting of *two previous steps*:

$$q_t := (1 + \beta_t) l_{t-1}(x_{t-1}, v_{t-1}) - \beta_t l_{t-2}(x_{t-2}, v_{t-2}), \tag{6.3}$$

  where $\beta_t \geq 0$ are parameters to be tuned.

With these defined, the dual variable u_t is chosen by performing an ascent step, starting from a convex combination of the previous dual variable u_{t-1} and some fixed $u_{-1} \geq 0$, in the direction of q_t. More precisely:

$$u_t := \max \left\{ 0, \left(\frac{\tau_t}{\tau_t + \gamma_t} u_{t-1} + \left(1 - \frac{\tau_t}{\tau_t + \gamma_t} \right) u_{-1} \right) + \frac{1}{\tau_t + \gamma_t} q_t \right\}, \tag{6.4}$$

where $\gamma_t, \tau_t \geq 0$ are parameters to be tuned. We note that (6.4) is equivalent to

$$u_t := \arg\min_{u \geq 0} \left\{ -q_t u + \frac{\tau_t}{2} (u - u_{t-1})^2 + \frac{\gamma_t}{2} (u - u_{-1})^2 \right\}.$$

Finally, instead of analysing $\{u_t\}_{t \geq 0}$, we will construct an alternative dual sequence to analyse: choose $\lambda_0 \geq 0$ and for each $t \geq 0$ set

$$\lambda_{t+1} = \lambda_t + \alpha_t (u_t - \lambda_t).$$

The full description is provided in Algorithm 5.

Algorithm 5: Primal-dual conditional gradient method (PD-CG).

Data: Parameters $\{\alpha_t\}_{t\geq 0} \subseteq [0,1]$, $\{\beta_t\}_{t\geq 0} \subseteq \mathsf{R}_+$, $\{\gamma_t\}_{t\geq 0} \subseteq \mathsf{R}_{++}$, $\{\tau_t\}_{t\geq 0} \subseteq \mathsf{R}_{++}$, $\{g_t\}_{t\geq 0}$ satisfying Assumption 6, $\{B_t\}_{t\geq 0}$ satisfying Assumption 5, number of iterations T.

Result: Sequence $\{x_t\}_{t\in[T]}$.

Initialise $x_0 = x_{-1} = x_{-2} = v_{-1} = v_{-2} \in X \cap B_0, \lambda_0 \geq 0, u_{-1} \geq 0, g_{-2} = g_{-1} = g_0$.;

for $t = 0, 1, \ldots, T-1$ **do**

Compute

$$
\begin{aligned}
q_t &:= (1+\beta_t) l_{t-1}(x_{t-1}, v_{t-1}) - \beta_t l_{t-2}(x_{t-2}, v_{t-2}) \\
u_t &:= \operatorname*{arg\,min}_{u\geq 0} \left\{ -q_t u + \frac{\tau_t}{2}(u - u_{t-1})^2 + \frac{\gamma_t}{2}(u - u_{-1})^2 \right\} \\
v_t &\in \operatorname*{arg\,min}_{v\in X\cap B_t} (\nabla f(x_t) + u_t \nabla g(x_t))^\top v \\
x_{t+1} &:= x_t + \alpha_t (v_t - x_t) \\
\lambda_{t+1} &:= \lambda_t + \alpha_t (u_t - \lambda_t).
\end{aligned}
$$

By recalling from Remark 4.1.1 that $t_0 \geq 0$ is an integer such that there exists an optimal solution x_{opt} of problem (1.1) such that $x_{\text{opt}} \in B_t$ for each $t \geq t_0$, the key idea of our unified analysis, detailed in Sections 6.2 to 6.3, is to bound the approximate duality gap as follows: for any $\lambda \geq 0$,

$$
L_T(x_T, \lambda) - L_T(x_{\text{opt}}, \lambda_T) \leq A_T \lambda^2 + B_T \lambda + C_T, \tag{6.5}
$$

for some $A_T > 0, B_T, C_T$. Given such a bound, we can derive bounds on the inner- and outer-level optimality gaps, with and without the existence of λ_{opt}.

Lemma 6.1. *If Assumption 3, Assumption 6 and (6.5) hold, then we have*

$$
\begin{aligned}
&f(x_T) - f_{\text{opt}} \leq C_T, \\
&g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{A_T \left(C_T + f_{\text{opt}} - \underline{f}\right)}.
\end{aligned} \tag{6.6}
$$

If Assumption 4, Assumption 6 and (6.5) hold, then we have

$$
\begin{aligned}
&-\lambda_{\text{opt}} \left(g(x_T) - g_{\text{opt}}\right) \leq f(x_T) - f_{\text{opt}} \leq C_T, \\
&g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{2A_T C_T + 2\lambda_{\text{opt}} A_T \left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}} A_T\right)^2}.
\end{aligned} \tag{6.7}
$$

Proof. For any $\lambda \geq 0$, we have

$$\begin{aligned} L_T(x_T, \lambda) - L_T(x_{\text{opt}}, \lambda_T) &= f(x_T) - f_{\text{opt}} + \lambda(g(x_T) - g_T) - \lambda_T(g_{\text{opt}} - g_T) \\ &\geq f(x_T) - f_{\text{opt}} + \lambda(g(x_T) - g_T), \end{aligned}$$

where the last inequality is true since $\lambda_T \geq 0, g_T \geq g_{\text{opt}}$. Therefore, we have that

$$A_T\lambda^2 - ((g(x_T) - g_T) - B_T)\,\lambda + (C_T - (f(x_T) - f_{\text{opt}})) \geq 0, \quad \forall \lambda \geq 0.$$

Since $A_T > 0$, we minimise the left-hand side with respect to $\lambda \geq 0$ at

$$\lambda = \frac{[g(x_T) - g_T - B_T]_+}{2A_T},$$

and obtain that

$$-\frac{[(g(x_T) - g_T) - B_T]_+^2}{4A_T} + (C_T - (f(x_T) - f_{\text{opt}})) \geq 0.$$

By multiplying both sides by $4A_T$, we have

$$[(g(x_T) - g_T) - B_T]_+^2 \leq 4A_T\,(C_T - (f(x_T) - f_{\text{opt}}))\,,$$

which implies $C_T - (f(x_T) - f_{\text{opt}}) \geq 0$. This establishes the upper bound on $f(x_T) - f_{\text{opt}}$ in both cases, noting that neither Assumption 4 nor Assumption 3 were used. Under Assumption 3, we have that

$$[(g(x_T) - g_T) - B_T]_+^2 \leq 4A_T\left(C_T + f_{\text{opt}} - \underline{f}\right),$$

which implies that

$$g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{A_T\left(C_T + f_{\text{opt}} - \underline{f}\right)}.$$

Now, suppose only Assumption 4 holds. Then we have $f(x_T) - f_{\text{opt}} \geq -\lambda_{\text{opt}}\,(g(x_T) - g_{\text{opt}})$. Hence,

we have that

$$[(g(x_T) - g_T) - B_T]_+^2 \leq 4A_T \left(C_T + \lambda_{\text{opt}} \left(g(x_T) - g_{\text{opt}}\right)\right).$$

By using $ab \leq (a^2 + b^2)/2$, we obtain

$$\begin{aligned}
&4\lambda_{\text{opt}} A_T \left(g(x_T) - g_{\text{opt}}\right) \\
&= 4\lambda_{\text{opt}} A_T \left(g(x_T) - g_T - B_T + g_T - g_{\text{opt}} + B_T\right) \\
&\leq \left(4\lambda_{\text{opt}} A_T\right) \left([g(x_T) - g_T - B_T]_+\right) + 4\lambda_{\text{opt}} A_T \left(g_T - g_{\text{opt}} + B_T\right) \\
&\leq \frac{1}{2}[g(x_T) - g_T - B_T]_+^2 + 8\left(\lambda_{\text{opt}} A_T\right)^2 + 4\lambda_{\text{opt}} A_T \left(g_T - g_{\text{opt}} + B_T\right).
\end{aligned}$$

Therefore, we have

$$\frac{1}{2}[(g(x_T) - g_T) - B_T]_+^2 \leq 4A_T C_T + 4\lambda_{\text{opt}} A_T \left(g_T - g_{\text{opt}} + B_T\right) + 8\left(\lambda_{\text{opt}} A_T\right)^2,$$

which implies

$$g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{2A_T C_T + 2\lambda_{\text{opt}} A_T \left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}} A_T\right)^2}.$$

□

In Section 6.2, we will show that (6.5) can be obtained under Assumption 6 and some conditions imposed on the parameters $\{\alpha_t, \beta_t, \gamma_t, \tau_t\}_{t\geq 0}$. We then provide specific choices of the parameters which ensure A_T, B_T, C_T are all $o(1)$, thus guaranteeing convergence for inner- and outer-level objectives.

6.2 Duality gap analysis

For the convenience of analysing Algorithm 5, we define the following functions:

$$h_t(x) := g(x) - g_t, \quad \forall t \geq -2,$$

$$l_f(x, y) := f(x) + \nabla f(x)^\top (y - x),$$

First, we derive the following bounds, which are utilised extensively in this section.

Lemma 6.2. *If Assumption 1 and Assumptions 5{6 hold, then there exists a constant $M > 0$ such that the iterates generated by Algorithm 5 satisfy*

$$|l_{t-1}(x_{t-1}, v_{t-1}) - l_{t-2}(x_{t-2}, v_{t-2})| \leq M d_t^2, \quad \forall t \geq 0.$$

Proof. Given any $t \geq 0$, using triangle inequality, we have that

$$\begin{aligned}
&|l_{t-1}(x_{t-1}, v_{t-1}) - l_{t-2}(x_{t-2}, v_{t-2})| \\
&= |g(x_{t-1}) - g(x_{t-2}) + \nabla g(x_{t-1})^\top (v_{t-1} - x_{t-1}) - \nabla g(x_{t-2})^\top (v_{t-2} - x_{t-2}) + g_{t-2} - g_{t-1}| \\
&\leq |g(x_{t-1}) - g(x_{t-2})| + |\nabla g(x_{t-1})^\top (v_{t-1} - x_{t-1})| + |\nabla g(x_{t-2})^\top (v_{t-2} - x_{t-2})| + |g_{t-2} - g_{t-1}|.
\end{aligned} \tag{6.8}$$

Using mean value theorem, there exists c_t lies in segment formed by x_{t-2}, x_{t-1} satisfying

$$g(x_{t-1}) - g(x_{t-2}) = \nabla g(c_t)^\top (x_{t-1} - x_{t-2}).$$

Using this result, Cauchy-Schwartz inequality and the fact that $c_t, x_{t-1}, x_{t-2}, v_{t-1}, v_{t-2} \in B_t$, (6.8) becomes

$$\begin{aligned}
&|l_{t-1}(x_{t-1}, v_{t-1}) - l_{t-2}(x_{t-2}, v_{t-2})| \\
&\leq |\nabla g(c_t)^\top (x_{t-1} - x_{t-2})| + |\nabla g(x_{t-1})^\top (v_{t-1} - x_{t-1})| + |\nabla g(x_{t-2})^\top (v_{t-2} - x_{t-2})| + |g_{t-2} - g_{t-1}| \\
&\leq \|\nabla g(c_t)\|_* \|x_t - x_{t-1}\| + \|\nabla g(x_{t-1})\|_* \|v_{t-1} - x_{t-1}\| + \|\nabla g(x_{t-2})\|_* \|v_{t-2} - x_{t-2}\| + |g_{t-2} - g_{t-1}| \\
&\leq (\|\nabla g(c_t)\|_* + \|\nabla g(x_{t-1})\|_* + \|\nabla g(x_{t-2})\|_*) \, d_t + |g_{t-2} - g_{t-1}| \\
&\leq 3 \left(\max_{x \in X \cap B_t} \|\nabla g(x)\|_* \right) d_t + |g_{t-2} - g_{t-1}|.
\end{aligned}$$

Given $x \in X \cap B_t$, using the smoothness of g and the triangle inequality, we have that

$$\begin{aligned}\|\nabla g(x)\|_* &= \|\nabla g(x_0) + (\nabla g(x) - \nabla g(x_0))\|_* \\ &\leq \|\nabla g(x_0)\|_* + \|\nabla g(x) - \nabla g(x_0)\|_* \\ &\leq \|\nabla g(x_0)\|_* + L_g\|x - x_0\| \\ &\leq \|\nabla g(x_0)\|_* + L_g d_t.\end{aligned}$$

Since $\{g_t\}_{t\geq 0}$ is non-increasing, $g_t \geq g_{\text{opt}}$ for each $t \geq 0$, and $g_{-1} = g_{-2} = g_0$, we have that

$$|g_{t-2} - g_{t-1}| = g_{t-2} - g_{t-1} \leq g_0 - g_{\text{opt}}, \quad \forall t \geq 0.$$

Therefore, we have that

$$|l_t(x_t, v_t) - l_{t-1}(x_{t-1}, v_{t-1})| \leq 3\left(\|\nabla g(x_0)\|_* + L_g d_t\right) d_t + g_0 - g_{\text{opt}}. \tag{6.9}$$

If sequence $\{d_t\}_{t\geq 0}$ is unbounded, we have

$$\limsup_{t\to\infty} \frac{|l_t(x_t, v_t) - l_{t-1}(x_{t-1}, v_{t-1})|}{d_t^2} \leq \limsup_{t\to\infty} \frac{3\left(\|\nabla g(x_0)\|_* + L_g d_t\right) d_t + g_0 - g_{\text{opt}}}{d_t^2} = 3L_g < \infty.$$

If $\{d_t\}_{t\geq 0}$ is bounded, then since the sequence is always non-decreasing, there exists $d := \lim_{t\to\infty} d_t$, which is positive and finite. This implies

$$\begin{aligned}\limsup_{t\to\infty} \frac{|l_t(x_t, v_t) - l_{t-1}(x_{t-1}, v_{t-1})|}{d_t^2} &\leq \limsup_{t\to\infty} \frac{3\left(\|\nabla g(x_0)\|_* + L_g d_t\right) d_t + g_0 - g_{\text{opt}}}{d_t^2} \\ &= \frac{3\left(\|\nabla g(x_0)\|_* + L_g d\right) d + g_0 - g_{\text{opt}}}{d^2} < \infty.\end{aligned}$$

In both cases, there exists a positive constant $M > 0$ as claimed. □

We state the following lemma, which directly results from the definition of u_t from Algorithm 5. From that, Lemma 6.4 establishes a critical recursive rule for the primal-dual gap terms given as

$L_t(x_t, \lambda) - L_t(x, \lambda_t)$ for $t \geq 0$.

Lemma 6.3. *Given $\lambda \geq 0$, the sequences $\{u_t, q_t\}_{0 \leq t < T}$ generated by Algorithm 5, we have*

$$-q_t(u_t - \lambda) + \frac{\tau_t}{2}(u_t - u_{t-1})^2 + \frac{\gamma_t}{2}(u_t - u_{-1})^2 \leq \frac{\tau_t}{2}(\lambda - u_{t-1})^2 - \frac{\tau_t + \gamma_t}{2}(\lambda - u_t)^2 + \frac{\gamma_t}{2}(\lambda - u_{-1})^2. \quad (6.10)$$

Proof. From the optimality condition for u_t, we have that

$$\begin{aligned} &(\tau_t(u_t - u_{t-1}) + \gamma_t(u_t - u_{-1}) - q_t)(\lambda - u_t) \geq 0 \\ \iff &(\tau_t(u_t - u_{t-1}) + \gamma_t(u_t - u_{-1}))(\lambda - u_t) \geq -q_t(u_t - \lambda), \end{aligned} \quad (6.11)$$

for any $\lambda \geq 0$. We also obtain that

$$\begin{aligned} &\frac{\tau_t}{2}(\lambda - u_{t-1})^2 - \frac{\tau_t}{2}(u_t - u_{t-1})^2 - \frac{\tau_t}{2}(\lambda - u_t)^2 \\ &= \frac{\tau_t}{2}(2\lambda - u_{t-1} - u_t)(u_t - u_{t-1}) - \frac{\tau_t}{2}(u_t - u_{t-1})^2 \\ &= \frac{\tau_t}{2}(2\lambda - u_{t-1} - u_t - u_t + u_{t-1})(u_t - u_{t-1}) \\ &= \tau_t(\lambda - u_t)(u_t - u_{t-1}), \end{aligned} \quad (6.12)$$

and

$$\begin{aligned} &\frac{\gamma_t}{2}(\lambda - u_{-1})^2 - \frac{\gamma_t}{2}(\lambda - u_t)^2 - \frac{\gamma_t}{2}(u_t - u_{-1})^2 \\ &= \frac{\gamma_t}{2}(2\lambda - u_{-1} - u_t)(u_t - u_{-1}) - \frac{\gamma_t}{2}(u_t - u_{-1})^2 \\ &= \frac{\gamma_t}{2}(2\lambda - u_{-1} - u_t - u_t + u_{-1})(u_t - u_{-1}) \\ &= \gamma_t(\lambda - u_t)(u_t - u_{-1}) \end{aligned} \quad (6.13)$$

By summing (6.11), (6.12) and (6.13), we have (6.10). □

Lemma 6.4. *If Assumption 1 and Assumptions 5{6 hold, then the iterates generated by Algorithm 5 satisfy*

$$\begin{aligned} L_{t+1}(x_{t+1}, \lambda) - L_{t+1}(x, \lambda_{t+1}) \leq (1 - \alpha_t)(L_t(x_t, \lambda) - L_t(x, \lambda_t)) \\ + \delta_{1,t}(\lambda) + \delta_{2,t}(\lambda) + \delta_{3,t}(\lambda) + \lambda(g_t - g_{t+1}), \quad \forall t \geq 0. \end{aligned} \quad (6.14)$$

for any $x \in X \cap B_t, \lambda \geq 0$, *where*

$$\delta_{1,t}(\lambda) := \alpha_t(\lambda - u_t)(l_t(x_t, v_t) - l_{t-1}(x_{t-1}, v_{t-1}) - \alpha_t\beta_t(\lambda - u_{t-1})(l_{t-1}(x_{t-1}, v_{t-1}) - l_{t-2}(x_{t-2}, v_{t-2})),$$
$$\delta_{2,t}(\lambda) := \frac{\alpha_t\tau_t}{2}(\lambda - u_{t-1})^2 - \frac{\alpha_t(\tau_t + \gamma_t)}{2}(\lambda - u_t)^2,$$
$$\delta_{3,t}(\lambda) := M^2\frac{\alpha_t\beta_t^2 d_t^4}{2\tau_t} + \frac{(L_f + \lambda L_g)\alpha_t^2 d_t^2}{2} + \frac{\alpha_t\gamma_t}{2}(\lambda - u_{-1})^2.$$

Proof. By the smoothness of g and f, we have that

$$\begin{aligned}
f(x_{t+1}) &\leq f(x_t) + \alpha_t\nabla f(x_t)^\top(v_t - x_t) + \frac{L_f\alpha_t^2}{2}\|v_t - x_t\|^2 \\
&= (1 - \alpha_t)f(x_t) + \alpha_t l_f(x_t, v_t) + \frac{L_f\alpha_t^2}{2}\|v_t - x_t\|^2, \\
h_t(x_{t+1}) &\leq h_t(x_t) + \alpha_t\nabla h_t(x_t)^\top(v_t - x_t) + \frac{L_g\alpha_t^2}{2}\|v_t - x_t\|^2 \\
\iff h_{t+1}(x_{t+1}) &\leq (1 - \alpha_t)h_t(x_t) + \alpha_t l_t(x_t, v_t) + \frac{L_g\alpha_t^2}{2}\|v_t - x_t\|^2 + g_t - g_{t+1}.
\end{aligned}$$

By the above results, We obtain the bound on the approximate duality gap as follows:

$$\begin{aligned}
&L_{t+1}(x_{t+1}, \lambda) - L_{t+1}(x, \lambda_{t+1}) \\
&= f(x_{t+1}) - f(x) + \lambda h_{t+1}(x_{t+1}) - \lambda_{t+1}h_{t+1}(x) \\
&\leq (1 - \alpha_t)f(x_t) + \alpha_t l_f(x_t, v_t) + \frac{L_f\alpha_t^2}{2}\|v_t - x_t\|^2 - f(x) \\
&\quad + \lambda\left((1 - \alpha_t)h_t(x_t) + \alpha_t l_t(x_t, v_t) + \frac{L_g\alpha_t^2}{2}\|v_t - x_t\|^2 + g_t - g_{t+1}\right) - \lambda_{t+1}h_{t+1}(x) \\
&= (1 - \alpha_t)\left(f(x_t) - f(x) + \lambda h_t(x_t)\right) + \alpha_t\left[l_f(x_t, v_t) - f(x) + \lambda l_t(x_t, v_t)\right] \\
&\quad - \lambda_{t+1}h_{t+1}(x) + \frac{(L_f + \lambda L_g)\alpha_t^2}{2}\|v_t - x_t\|^2 + \lambda(g_t - g_{t+1}),
\end{aligned} \tag{6.15}$$

for any $x \in X, \lambda \geq 0$. Using the monotonicity of $\{g_t\}_{t\geq 0}$ from Assumption 6, we observe

$$\begin{aligned}
-\lambda_{t+1}h_{t+1}(x) &= -\lambda_{t+1}(g(x) - g_{t+1}) \\
&= \lambda_{t+1}(g_{t+1} - g(x))
\end{aligned}$$

$$\begin{aligned}
&\leq \lambda_{t+1}(g_t - g(x)) \\
&= -(1-\alpha_t)\lambda_t h_t(x) - \alpha_t u_t h_t(x),
\end{aligned}$$

which together with (6.15), implies that

$$\begin{aligned}
L_{t+1}(x_{t+1}, \lambda) - L_{t+1}(x, \lambda_{t+1}) \leq (1-\alpha_t)\left(L_t(x_t, \lambda) - L_t(x, \lambda_t)\right) \\
+ \alpha_t\left[l_f(x_t, v_t) - f(x) + \lambda l_t(x_t, v_t) - u_t h_t(x)\right] \\
+ \frac{(L_f + \lambda L_g)\alpha_t^2}{2}\|v_t - x_t\|^2 + \lambda(g_t - g_{t+1}).
\end{aligned}$$

By the convexity of h_t, f and the definition of v_t, if $x \in B_t \cap X$, we have that

$$\begin{aligned}
l_f(x_t, v_t) + u_t l_t(x_t, v_t) &= f(x_t) + u_t h_t(x_t) + (\nabla f(x_t) + u_t \nabla g(x_t))^\top (v_t - x_t) \\
&\leq f(x_t) + u_t h_t(x_t) + (\nabla f(x_t) + u_t \nabla g(x_t))^\top (x - x_t) \\
&= f(x_t) + \nabla f(x_t)^\top (x - x_t) + u_t\left(h_t(x_t) + \nabla h_t(x_t)^\top (x - x_t)\right) \\
&\leq f(x) + u_t h_t(x),
\end{aligned}$$

which implies that for any $x \in B_t \cap X$,

$$\begin{aligned}
L_{t+1}(x_{t+1}, \lambda) - L_{t+1}(x, \lambda_{t+1}) \leq (1-\alpha_t)\left(L_t(x_t, \lambda) - L_t(x, \lambda_t)\right) \\
+ \alpha_t(\lambda - u_t) l_t(x_t, v_t) + \frac{(L_f + \lambda L_g)\alpha_t^2}{2}\|v_t - x_t\|^2 + \lambda(g_t - g_{t+1}).
\end{aligned} \tag{6.16}$$

By multiplying both sides of (6.10) by α_t and summing them up with (6.16), we obtain

$$\begin{aligned}
&L_{t+1}(x_{t+1},\lambda) - L_{t+1}(x,\lambda_{t+1}) \\
&\leq (1-\alpha_t)\left(L_t(x_t,\lambda) - L_t(x,\lambda_t)\right) + \alpha_t(\lambda-u_t)(l_t(x_t,v_t)-q_t) \\
&\quad + \frac{\alpha_t\tau_t}{2}\left[(\lambda-u_{t-1})^2 - (u_t-u_{t-1})^2\right] - \frac{\alpha_t(\tau_t+\gamma_t)}{2}(\lambda-u_t)^2 \\
&\quad + \frac{\alpha_t\gamma_t}{2}\left[(\lambda-u_{-1})^2 - (u_t-u_{-1})^2\right] + \frac{(L_f+\lambda L_g)\alpha_t^2}{2}\|v_t-x_t\|^2 + \lambda(g_t-g_{t+1}) \\
&\leq (1-\alpha_t)\left(L_t(x_t,\lambda) - L_t(x,\lambda_t)\right) + \alpha_t(\lambda-u_t)(l_t(x_t,v_t)-q_t) - \frac{\alpha_t\tau_t}{2}(u_t-u_{t-1})^2 \\
&\quad \delta_{2,t}(\lambda) + \frac{\alpha_t\gamma_t}{2}(\lambda-u_{-1})^2 + \frac{(L_f+\lambda L_g)\alpha_t^2 d_t^2}{2} + \lambda(g_t-g_{t+1}),
\end{aligned} \tag{6.17}$$

where in the second inequality, we use the fact that $\frac{\alpha_t\gamma_t}{2}(u_t-u_{-1})^2 \geq 0$ and $\|x_t-v_t\| \leq d_t$. By using the definition of q_t, we have

$$\begin{aligned}
&(\lambda-u_t)(l_t(x_t,v_t)-q_t) - \frac{\tau_t}{2}(u_t-u_{t-1})^2 \\
&= (\lambda-u_t)(l_t(x_t,v_t) - l_{t-1}(x_{t-1},v_{t-1}) - \beta_t(l_{t-1}(x_{t-1},v_{t-1}) - l_{t-2}(x_{t-2},v_{t-2}))) - \frac{\tau_t}{2}(u_t-u_{t-1})^2 \\
&= (\lambda-u_t)(l_t(x_t,v_t) - l_{t-1}(x_{t-1},v_{t-1})) \\
&\quad - \beta_t(\lambda-u_{t-1}+u_{t-1}-u_t)(l_{t-1}(x_{t-1},v_{t-1}) - l_{t-2}(x_{t-2},v_{t-2})) - \frac{\tau_t}{2}(u_t-u_{t-1})^2 \\
&= \frac{\delta_{1,t}(\lambda)}{\alpha_t} + \beta_t(u_t-u_{t-1})(l_{t-1}(x_{t-1},v_{t-1}) - l_{t-2}(x_{t-2},v_{t-2})) - \frac{\tau_t}{2}(u_t-u_{t-1})^2.
\end{aligned} \tag{6.18}$$

Using $ab - \frac{a^2c}{2} \leq \frac{b^2}{2c}$ with $c>0$ and Lemma 6.2, we have that

$$\begin{aligned}
&\beta_t(u_t-u_{t-1})(l_{t-1}(x_{t-1},v_{t-1}) - l_{t-2}(x_{t-2},v_{t-2})) - \frac{\tau_t}{2}(u_t-u_{t-1})^2 \\
&\leq \frac{\beta_t^2}{2\tau_t}(l_{t-1}(x_{t-1},v_{t-1}) - l_{t-2}(x_{t-2},v_{t-2}))^2 \\
&\leq \frac{M^2\beta_t^2 d_t^4}{2\tau_t}.
\end{aligned} \tag{6.19}$$

By adding (6.19) to (6.18), we obtain

$$(\lambda-u_t)(l_t(x_t,v_t)-q_t) - \frac{\tau_t}{2}(u_t-u_{t-1})^2 \leq \frac{\delta_{1,t}(\lambda)}{\alpha_t} + M^2\frac{\beta_t^2 d_t^4}{2\tau_t}. \tag{6.20}$$

Multiplying both sides of (6.20) by α_t and adding to (6.17), we observe

$$\begin{aligned}
&L_{t+1}(x_{t+1},\lambda) - L_{t+1}(x,\lambda_{t+1}) \\
&\leq (1-\alpha_t)\left(L_t(x_t,\lambda) - L_t(x,\lambda_t)\right) + \delta_{1,t}(\lambda) + \delta_{2,t}(\lambda) \\
&\quad + M^2\frac{\alpha_t\beta_t^2 d_t^4}{2\tau_t} + \frac{(L_f+\lambda L_g)\alpha_t^2 d_t^2}{2} + \frac{\alpha_t\gamma_t}{2}(\lambda-u_{-1})^2 + \lambda(g_t - g_{t+1}) \\
&= (1-\alpha_t)\left(L_t(x_t,\lambda) - L_t(x,\lambda_t)\right) + \delta_{1,t}(\lambda) + \delta_{2,t}(\lambda) + \delta_{3,t}(\lambda) + \lambda(g_t - g_{t+1}).
\end{aligned}$$

□

If $T > t_0$ and $\alpha_t < 1$ for $t \geq 1$, using Corollary 2.4 for (6.14) and substituting $x := x_{\text{opt}}$, we obtain the following bound on the duality gap:

$$\begin{aligned}
&L_T(x_T,\lambda) - L_T(x_{\text{opt}},\lambda_T) \\
&\leq H_0 a_{T-1} + a_{T-1}\sum_{t\in[T]}\left(\frac{\delta_{1,t-1}(\lambda) + \delta_{2,t-1}(\lambda) + \delta_{3,t-1}(\lambda) + \lambda(g_{t-1} - g_t)}{a_{t-1}}\right),
\end{aligned} \tag{6.21}$$

for any $\lambda \geq 0$ and constant H_0 defined as in Lemma 2.3. The following lemma simplifies the first two δ-terms in (6.21).

Lemma 6.5. *If $T > 1$ and $\alpha_t < 1$ for $t \geq 1$, then the iterates generated by Algorithm 5 satisfy*

$$\sum_{t\in[T]}\frac{\delta_{1,t-1}(\lambda)}{a_{t-1}} = \frac{\alpha_{T-1}\rho_{T-1}(\lambda)}{a_{T-1}} + \sum_{t\in[T-1]}\left(\alpha_{t-1} - \frac{\alpha_t\beta_t}{1-\alpha_t}\right)\frac{\rho_{t-1}(\lambda)}{a_{t-1}}, \tag{6.22}$$

and

$$\begin{aligned}
\sum_{t\in[T]}\frac{\delta_{2,t-1}(\lambda)}{a_{t-1}} &= \frac{\alpha_0\tau_0}{2}(\lambda-u_{-1})^2 - \frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2a_{T-1}}(\lambda-u_{T-1})^2 \\
&\quad + \sum_{t\in[T-1]}\left(\frac{\alpha_t\tau_t}{1-\alpha_t} - \alpha_{t-1}(\tau_{t-1}+\gamma_{t-1})\right)\frac{(\lambda-u_{t-1})^2}{2a_{t-1}},
\end{aligned} \tag{6.23}$$

where $\delta_{1,t}(\lambda), \delta_{2,t}(\lambda)$ are defined as in Lemma 6.4 and $\rho_t(\lambda) := (\lambda - u_t)(l_t(x_t,v_t) - l_{t-1}(x_{t-1},v_{t-1}))$ for $t \geq 0$.

Proof. Using definition of $\delta_{1,t}(\lambda)$, we obtain

$$\begin{aligned}\sum_{t\in[T]}\frac{\delta_{1,t-1}(\lambda)}{a_{t-1}} &= \sum_{t\in[T]}\frac{\alpha_{t-1}\rho_{t-1}(\lambda)-\alpha_{t-1}\beta_{t-1}\rho_{t-2}(\lambda)}{a_{t-1}}\\ &= \frac{\alpha_{T-1}\rho_{T-1}(\lambda)}{a_{T-1}}+\sum_{t\in[T-1]}\left(\frac{\alpha_{t-1}}{a_{t-1}}-\frac{\alpha_t\beta_t}{a_t}\right)\rho_{t-1}(\lambda)-\frac{\alpha_0\beta_0\rho_{-1}(\lambda)}{a_0}\\ &= \frac{\alpha_{T-1}\rho_{T-1}(\lambda)}{a_{T-1}}+\sum_{t\in[T-1]}\left(\alpha_{t-1}-\frac{\alpha_t\beta_t}{1-\alpha_t}\right)\frac{\rho_{t-1}(\lambda)}{a_{t-1}},\end{aligned}$$

where in the last equality, we use $a_t=(1-\alpha_t)a_{t-1}$ for each $t\geq 1$ and $\rho_{-1}(\lambda)=0$ by $v_{-1}=v_{-2}$, $x_{-2}=x_{-1}$, $g_{-2}=g_{-1}$. To prove (6.23), we observe

$$\begin{aligned}\sum_{t\in[T]}\frac{\delta_{2,t-1}(\lambda)}{a_{t-1}} &= \sum_{t\in[T]}\left(\frac{\alpha_{t-1}\tau_{t-1}}{2a_{t-1}}(\lambda-u_{t-2})^2-\frac{\alpha_{t-1}(\tau_{t-1}+\gamma_{t-1})}{2a_{t-1}}(\lambda-u_{t-1})^2\right)\\ &= \frac{\alpha_0\tau_0}{2}(\lambda-u_{-1})^2-\frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2a_{T-1}}(\lambda-u_{T-1})^2\\ &\quad+\sum_{t\in[T-1]}\left(\frac{\alpha_t\tau_t}{2a_t}-\frac{\alpha_{t-1}(\tau_{t-1}+\gamma_{t-1})}{2a_{t-1}}\right)(\lambda-u_{t-1})^2\\ &= \frac{\alpha_0\tau_0}{2}(\lambda-u_{-1})^2-\frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2a_{T-1}}(\lambda-u_{T-1})^2\\ &\quad+\sum_{t\in[T-1]}\left(\frac{\alpha_t\tau_t}{1-\alpha_t}-\alpha_{t-1}(\tau_{t-1}+\gamma_{t-1})\right)\frac{(\lambda-u_{t-1})^2}{2a_{t-1}}.\end{aligned}$$

□

To simplify (6.22)–(6.23), we will choose the parameters $\{\alpha_t,\beta_t,\gamma_t,\tau_t\}_{t\geq 0}$ to make the sums on the right-hand sides equal to 0. Therefore, we impose the following condition on the parameters.

Condition 6.1. Sequences $\{\alpha_t,\beta_t,\gamma_t,\tau_t\}_{t\geq 0}$ satisfy that for any $t\geq 1$, we have

$$\alpha_0=1,\quad \alpha_t<1,\quad \frac{\beta_t\alpha_t}{1-\alpha_t}=\alpha_{t-1},\quad \frac{\alpha_t\tau_t}{1-\alpha_t}=\alpha_{t-1}(\tau_{t-1}+\gamma_{t-1}).$$

From (6.22), we can see that if $\beta_t=0$ (i.e., there is no extrapolation term in defining q_t in Algorithm 5) then we would not be able to remove the sum term in (6.22). Under Condition 6.1,

(6.21) can be simplified as follows.

Lemma 6.6. *Suppose $\{x_t, \lambda_t\}_{t\in[T]}$ are the sequences generated by Algorithm 5. If Assumption 1, Assumptions 5{6, Condition 6.1 hold, and $T > \max\{t_0, 1\}$, then for any $\lambda \geq 0$, we have*

$$L_T(x_T, \lambda) - L_T(x_{\text{opt}}, \lambda_T) \leq A_T\lambda^2 + B'_T\lambda + C'_T, \tag{6.24}$$

where

$$\begin{aligned}
A_T &:= a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}} + \frac{\alpha_0\tau_0 a_{T-1}}{2},\\
B'_T &:= -2u_{-1}\left(a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}} + \frac{\alpha_0\tau_0 a_{T-1}}{2}\right) + a_{T-1}\sum_{t\in[T]}\frac{L_g d_{t-1}^2\alpha_{t-1}^2}{2a_{t-1}}\\
&\quad + a_{T-1}\left(g_0 - g_{\text{opt}} + \sum_{t\in[T-1]}\left(\frac{1}{a_t} - \frac{1}{a_{t-1}}\right)(g_t - g_{\text{opt}})\right),\\
C'_T &:= a_{T-1}\sum_{t\in[T]}\left(\frac{M^2\alpha_{t-1}\beta_{t-1}^2 d_{t-1}^4}{2\tau_{t-1}a_{t-1}} + \frac{L_f d_{t-1}^2\alpha_{t-1}^2}{2a_{t-1}}\right) + (u_{-1})^2\left(a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}} + \frac{\alpha_0\tau_0 a_{T-1}}{2}\right)\\
&\quad + \frac{M^2\alpha_{T-1}d_T^4}{2(\tau_{T-1}+\gamma_{T-1})} + H_0 a_{T-1}.
\end{aligned} \tag{6.25}$$

Proof. Under Condition 6.1, (6.22) and (6.23) become

$$\begin{aligned}
\sum_{t\in[T]}\left(\frac{\delta_{1,t-1}(\lambda)}{a_{t-1}}\right) &= \frac{\alpha_{T-1}\rho_{T-1}(\lambda)}{a_{T-1}},\\
\sum_{t\in[T]}\left(\frac{\delta_{2,t-1}(\lambda)}{a_{t-1}}\right) &= \frac{\alpha_0\tau_0}{2}(\lambda - u_{-1})^2 - \frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})(\lambda - u_{T-1})^2}{2a_{T-1}}.
\end{aligned}$$

Thus, (6.21) becomes

$$\begin{aligned}
&L_T(x_T,\lambda) - L_T(x_{\text{opt}},\lambda_T) \\
&\leq a_{T-1}\sum_{t\in[T]}\left(\frac{\delta_{3,t-1}(\lambda)+\lambda(g_{t-1}-g_t)}{a_{t-1}}\right) \\
&\quad + \alpha_{T-1}\rho_{T-1}(\lambda) - \frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2}(\lambda-u_{T-1})^2 + \frac{\alpha_0\tau_0 a_{T-1}}{2}(\lambda-u_{-1})^2 + H_0 a_{T-1}.
\end{aligned} \tag{6.26}$$

Using $ab - \frac{a^2c}{2} \leq \frac{b^2}{2c}$ with $c>0$, we have

$$\begin{aligned}
&\alpha_{T-1}\rho_{T-1}(\lambda) - \frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2}(\lambda-u_{T-1})^2 \\
&= \alpha_{T-1}(\lambda-u_{T-1})(l_{T-1}(x_{T-1},v_{T-1}) - l_{T-2}(x_{T-2},v_{T-2})) - \frac{\alpha_{T-1}(\tau_{T-1}+\gamma_{T-1})}{2}(\lambda-u_{T-1})^2 \\
&\leq \frac{\alpha_{T-1}(l_{T-1}(x_{T-1},v_{T-1}) - l_{T-2}(x_{T-2},v_{T-2}))^2}{2\,(\tau_{T-1}+\gamma_{T-1})} \\
&\leq \frac{M^2\alpha_{T-1}d_T^4}{2(\tau_{T-1}+\gamma_{T-1})}.
\end{aligned} \tag{6.27}$$

In addition, we have that

$$\begin{aligned}
\sum_{t\in[T]}\frac{g_{t-1}-g_t}{a_{t-1}} &= \sum_{t=0}^{T-1}\frac{g_t - g_{\text{opt}} - (g_{t+1}-g_{\text{opt}})}{a_t} \\
&= \sum_{t=0}^{T-1}\frac{g_t-g_{\text{opt}}}{a_t} - \sum_{t=1}^{T}\frac{g_t-g_{\text{opt}}}{a_{t-1}} \\
&= g_0 - g_{\text{opt}} + \sum_{t\in[T-1]}\left(\frac{1}{a_t}-\frac{1}{a_{t-1}}\right)(g_t-g_{\text{opt}}) - \frac{g_T-g_{\text{opt}}}{a_T} \\
&\leq g_0 - g_{\text{opt}} + \sum_{t\in[T-1]}\left(\frac{1}{a_t}-\frac{1}{a_{t-1}}\right)(g_t-g_{\text{opt}}).
\end{aligned}$$

Hence, we have that

$$\begin{aligned}
&\sum_{t\in[T]} \frac{\delta_{3,t-1}(\lambda) + \lambda(g_{t-1} - g_t)}{a_{t-1}} \\
&\leq \sum_{t\in[T]} \left(\frac{M^2\alpha_{t-1}\beta_{t-1}^2 d_{t-1}^4}{2\tau_{t-1}a_{t-1}} + \frac{(L_f + \lambda L_g) d_{t-1}^2 \alpha_{t-1}^2}{2a_{t-1}} + \frac{\alpha_{t-1}\gamma_{t-1}(\lambda - u_{-1})^2}{2a_{t-1}} \right) \\
&\quad + \lambda \left(g_0 - g_{\text{opt}} + \sum_{t\in[T-1]} \left(\frac{1}{a_t} - \frac{1}{a_{t-1}} \right) (g_t - g_{\text{opt}}) \right).
\end{aligned} \tag{6.28}$$

Summing (6.27), (6.28) to (6.26), we obtain

$$\begin{aligned}
&L_T(x_T, \lambda) - L_T(x_{\text{opt}}, \lambda_T) \\
&\leq a_{T-1} \sum_{t\in[T]} \left(\frac{M^2\alpha_{t-1}\beta_{t-1}^2 d_{t-1}^4}{2\tau_{t-1}a_{t-1}} + \frac{(L_f + \lambda L_g) d_{t-1}^2 \alpha_{t-1}^2}{2a_{t-1}} + \frac{\alpha_{t-1}\gamma_{t-1}(\lambda - u_{-1})^2}{2a_{t-1}} \right) \\
&\quad + a_{T-1} \left(g_0 - g_{\text{opt}} + \sum_{t\in[T-1]} \left(\frac{1}{a_t} - \frac{1}{a_{t-1}} \right) (g_t - g_{\text{opt}}) \right) \lambda \\
&\quad + \frac{M^2\alpha_{T-1} d_T^4}{2(\tau_{T-1} + \gamma_{T-1})} + \frac{\alpha_0\tau_0 a_{T-1}}{2}(\lambda - u_{-1})^2 + H_0 a_{T-1}.
\end{aligned} \tag{6.29}$$

□

6.3 Convergence analysis

Using the analysis on the duality gap conducted in Section 6.2, we now provide specific convergence rates for the PD-CG method. First of all, Lemma 6.7 provides a choice of parameters for $\{\alpha_t, \beta_t, \gamma_t, \tau_t\}_{t\geq 0}$ satisfying Condition 6.1. Based on that choice, Corollary 6.8 establishes a more specific upper bound on the duality gap.

Lemma 6.7. *The following choice of sequences $\{\alpha_t, \beta_t, \gamma_t, \tau_t\}_{t\geq 0}$ satisfy Condition 6.1.*

$$\alpha_t = \frac{2}{t+2}, \quad \beta_t = \frac{t}{t+1}, \quad \tau_t = R(t+1)^p, \quad \gamma_t = \frac{R(t+2)^{1+p}}{t+1} - \tau_t, \quad \forall t \geq 0,$$

given $p \in (0,1)$ *and* $R > 0$.

Proof. Given $t \geq 1$, we have $\alpha_t = \frac{2}{t+2} < \frac{2}{0+2} = 1$ and

$$\begin{aligned}
\frac{\beta_t \alpha_t}{1-\alpha_t} &= \frac{\left(\frac{t}{t+1}\right)\left(\frac{2}{t+2}\right)}{\frac{t}{t+2}} = \frac{2}{t+1} = \alpha_{t-1}, \\
\frac{\alpha_t \tau_t}{1-\alpha_t} &= \frac{\left(\frac{2}{t+2}\right) R(t+1)^p}{\frac{t}{t+2}} = 2R\frac{(t+1)^p}{t}, \\
\alpha_{t-1}(\tau_{t-1}+\gamma_{t-1}) &= \frac{2}{t+1}\frac{R(t+1)^{1+p}}{t} = 2R\frac{(t+1)^p}{t}.
\end{aligned}$$

□

Corollary 6.8. *Suppose* $\{x_t, \lambda_t\}_{t\in[T]}$ *are the sequences generated by Algorithm 5 with parameters* $\{\alpha_t, \beta_t, \gamma_t, \tau_t\}_{t\geq 0}$ *as given in Lemma 6.7 and* A_T, B_T, C_T *are de ned as in Lemma 6.1. If Assumption 1, Assumptions 5{6 hold and* $T > \max\{t_0, 1\}$*, then*

$$\begin{aligned}
A_T &= \frac{R(T+1)^p}{T}, \\
B'_T \leq B_T &:= -\frac{2u_{-1}R(T+1)^p}{T} + \frac{2(L_g+Q)d_T^2}{T+1}, \\
C'_T \leq C_T &:= \frac{M^2}{R(2-p)}\frac{T^{1-p}d_T^4}{T+1} + \frac{2L_f d_T^2}{T+1} + \frac{(u_{-1})^2 R(T+1)^p}{T} + \frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}} + \frac{2H_0}{(T+1)T},
\end{aligned} \tag{6.30}$$

and hence, for any $\lambda \geq 0$*, we have*

$$L_T(x_T, \lambda) - L_T(x_{\text{opt}}, \lambda_T) \leq A_T\lambda^2 + B_T\lambda + C_T. \tag{6.31}$$

Proof. From Corollary 2.4, we have that

$$a_t = \prod_{i\in[t]}(1-\alpha_i) = \frac{2}{(t+1)(t+2)}, \quad \forall t \geq 0.$$

Now, we simplify the running sums from (6.25) as follows:

$$\begin{aligned}
\sum_{t\in[T]} \frac{\alpha_{t-1}^2 d_{t-1}^2}{a_{t-1}} &= \sum_{t\in[T]} \frac{\left(\frac{2}{t+1}\right)^2 d_{t-1}^2}{\frac{2}{t(t+1)}} \leq 2d_T^2 \sum_{t\in[T]} \frac{t}{t+1} \leq 2Td_T^2 \\
\sum_{t\in[T]} \frac{\alpha_{t-1}\gamma_{t-1}}{a_{t-1}} &= R\sum_{t\in[T]} \left((t+1)^{1+p} - t^{1+p}\right) = R\left((T+1)^{1+p} - 1\right) \\
\sum_{t\in[T]} \frac{\alpha_{t-1}\beta_{t-1}^2 d_{t-1}^4}{\tau_{t-1}a_{t-1}} &\leq d_T^4 \sum_{t=0}^{T-1} \frac{\left(\frac{2}{t+1}\right)\left(\frac{t-1}{t}\right)^2}{Rt^p \frac{2}{t(t+1)}} \\
&= \frac{d_T^4}{R} \sum_{t\in[T]} \frac{(t-1)^2}{t^{1+p}} \\
&\leq \frac{d_T^4}{R} \sum_{t\in[T]} (t-1)^{1-p} \\
&\leq \frac{d_T^4}{R} \int_0^T s^{1-p} ds \\
&= \frac{T^{2-p} d_T^4}{R(2-p)}.
\end{aligned}$$

We also have that

$$\begin{aligned}
\frac{M^2\alpha_{T-1}d_T^4}{2(\tau_{T-1}+\gamma_{T-1})} &= \frac{M^2\left(\frac{2}{T+1}\right)d_T^4}{2R\frac{(T+1)^{1+p}}{T}} = \frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}}, \\
\frac{\alpha_0\tau_0 a_{T-1}}{2} &= \frac{R}{2}\frac{2}{T(T+1)} = \frac{R}{(T+1)T}.
\end{aligned}$$

From Assumption 6, we have

$$g_t - g_{\text{opt}} \leq \frac{Qd_t^2}{t+1}, \quad \forall t \leq T,$$

and hence,

$$g_0 - g_{\text{opt}} + \sum_{t\in[T-1]} \left(\frac{1}{a_t} - \frac{1}{a_{t-1}}\right)(g_t - g_{\text{opt}}) = \sum_{t=0}^{T-1}(t+1)(g_t - g_{\text{opt}}) \leq \sum_{t=0}^{T-1}(t+1)\frac{Qd_t^2}{t+1} \leq QTd_T^2.$$

Therefore, from (6.25), we observe that

$$
\begin{aligned}
A_T &= a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}}+\frac{\alpha_0\tau_0 a_{T-1}}{2}=\frac{2}{(T+1)T}\frac{R\left((T+1)^{1+p}-1\right)}{2}+\frac{R}{(T+1)T}=\frac{R(T+1)^p}{T},\\
B'_T &= -2u_{-1}\left(a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}}+\frac{\alpha_0\tau_0 a_{T-1}}{2}\right)+a_{T-1}\sum_{t\in[T]}\frac{L_g d_{t-1}^2\alpha_{t-1}^2}{2a_{t-1}}\\
&\quad + a_{T-1}\left(g_0-g_{\text{opt}}+\sum_{t\in[T-1]}\left(\frac{1}{a_t}-\frac{1}{a_{t-1}}\right)(g_t-g_{\text{opt}})\right)\\
&\le -2u_{-1}\frac{R(T+1)^p}{T}+\frac{2}{(T+1)T}\frac{L_g 2Td_T^2}{2}+\frac{2QTd_T^2}{(T+1)T}\\
&= -\frac{2u_{-1}R(T+1)^p}{T}+\frac{2(L_g+Q)d_T^2}{T+1},\\
C'_T &= a_{T-1}\sum_{t\in[T]}\left(\frac{M^2\alpha_{t-1}\beta_{t-1}^2 d_{t-1}^4}{2\tau_{t-1}a_{t-1}}+\frac{L_f d_{t-1}^2\alpha_{t-1}^2}{2a_{t-1}}\right)+(u_{-1})^2\left(a_{T-1}\sum_{t\in[T]}\frac{\alpha_{t-1}\gamma_{t-1}}{2a_{t-1}}+\frac{\alpha_0\tau_0 a_{T-1}}{2}\right)\\
&\quad + \frac{M^2\alpha_{T-1}d_T^4}{2(\tau_{T-1}+\gamma_{T-1})}+H_0 a_{T-1}\\
&\le \frac{2}{(T+1)T}\left(\frac{M^2}{2}\frac{T^{2-p}d_T^4}{R(2-p)}+\frac{L_f 2Td_T^2}{2}\right)+(u_{-1})^2\frac{R(T+1)^p}{T}+\frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}}+\frac{2H_0}{(T+1)T}\\
&= \frac{M^2}{R(2-p)}\frac{T^{1-p}d_T^4}{T+1}+\frac{2L_f d_T^2}{T+1}+\frac{(u_{-1})^2R(T+1)^p}{T}+\frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}}+\frac{2H_0}{(T+1)T}.
\end{aligned}
$$

□

Before establishing convergence rates of Algorithm 5, we impose Condition 6.2 on the growth of $\{d_t\}_{t\ge 0}$. From there, we establish convergence rates and asymptotic convergence guaranteed by Algorithm 5, shown in Theorem 6.9. Moreover, we also prove that if Assumption 4 holds, we have improved convergence rates and super-optimality bounds for Algorithm 5 in Theorem 6.10.

Condition 6.2. Sequence $\{B_t\}_{t\ge 0}$ satisfies $d_t = o\left(t^{p/4}\right)$, where p is defined as in Lemma 6.7.

Theorem 6.9. *Suppose $\{x_t\}_{t\in[T]}$ is the sequence generated by Algorithm 5 with parameters $\{\alpha_t,\beta_t,\gamma_t,\tau_t\}_{t\ge 0}$ as given in Lemma 6.7. If Assumption 1, Assumption 3, Assumptions 5{6 and Condition 6.2 hold,*

then

$$\begin{aligned} g(x_T) - g_{\text{opt}} &\leq O\left(\frac{1}{T^{(1-p)/2}}\right), \\ f(x_T) - f_{\text{opt}} &\leq O\left(\max\left\{\frac{1}{T^{1-p}}, \frac{d_T^4}{T^p}\right\}\right). \end{aligned} \tag{6.32}$$

Proof. By Lemma 6.1 and Corollary 6.8, for $T > \max\{t_0, 1\}$, we have

$$f(x_T) - f_{\text{opt}} \leq C_T, \quad g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{A_T\left(C_T + f_{\text{opt}} - \underline{f}\right)}.$$

By Assumption 6 and (6.30), we have

$$g_T - g_{\text{opt}} + B_T \leq \frac{Qd_T^2}{T+1} - \frac{2u_{-1}R(T+1)^p}{T} + \frac{2(L_g+Q)d_T^2}{T+1} \leq \frac{(2L_g+3Q)d_T^2}{T+1}. \tag{6.33}$$

Since $d_T \geq d_0 > 0$, we have that $\lim_{T\to\infty} T^{1-p}d_T^2 = \infty$. Thus, we have that

$$\begin{aligned} &\lim_{T\to\infty} \frac{T^p}{d_T^4}\left(\frac{M^2}{R(2-p)}\frac{T^{1-p}d_T^4}{T+1} + \frac{2L_f d_T^2}{T+1} + \frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}}\right) \\ &= \lim_{T\to\infty}\left(\frac{M^2}{R(2-p)}\frac{T}{T+1} + 2L_f\frac{T}{T+1}\frac{1}{T^{1-p}d_T^2} + \frac{M^2}{R}\frac{T^{1+p}}{(T+1)^{2+p}}\right) \\ &= \frac{M^2}{R(2-p)}, \\ &\lim_{T\to\infty} T^{1-p}\left(\frac{(u_{-1})^2R(T+1)^p}{T} + \frac{2H_0}{(T+1)T}\right) \\ &= \lim_{T\to\infty}\left(\frac{(u_{-1})^2R(T+1)^p}{T^p} + \frac{2H_0}{(T+1)T^p}\right) \\ &= (u_{-1})^2R. \end{aligned}$$

Hence, there exists $M_1, M_2 > 0$ such that if $T > \max\{t_0, 1\}$, then

$$\begin{aligned} \frac{M^2}{R(2-p)}\frac{T^{1-p}d_T^4}{T+1} + \frac{2L_f d_T^2}{T+1} + \frac{M^2}{R}\frac{Td_T^4}{(T+1)^{2+p}} &\leq M_1\frac{d_T^4}{T^p}, \\ \frac{(u_{-1})^2R(T+1)^p}{T} + \frac{2H_0}{(T+1)T} &\leq M_2\frac{1}{T^{1-p}}, \end{aligned}$$

which implies

$$f(x_T) - f_{\text{opt}} \leq C_T \leq M_1 \frac{d_T^4}{T^p} + M_2 \frac{1}{T^{1-p}} \leq (M_1 + M_2) \max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\}. \tag{6.34}$$

We observe that

$$A_T \left(C_T + f_{\text{opt}} - \underline{f}\right) \leq \frac{R(T+1)^p}{T} \left((M_1 + M_2) \max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + f_{\text{opt}} - \underline{f}\right).$$

By Condition 6.2, we have

$$\begin{aligned}
&\lim_{T\to\infty} T^{1-p} \frac{R(T+1)^p}{T} \left((M_1 + M_2) \max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + f_{\text{opt}} - \underline{f}\right) \\
&= \lim_{T\to\infty} \frac{R(T+1)^p}{T^p} \left((M_1 + M_2) \max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + f_{\text{opt}} - \underline{f}\right) \\
&= R\left(f_{\text{opt}} - \underline{f}\right),
\end{aligned}$$

which implies there exists $M_3 > 0$ such that if $T > \max\{t_0, 1\}$, then

$$A_T \left(C_T + f_{\text{opt}} - \underline{f}\right) \leq M_3^2 \frac{1}{T^{1-p}}.$$

Thus, we obtain

$$g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{A_T \left(C_T + f_{\text{opt}} - \underline{f}\right)} \leq \frac{(2L_g + 3Q)d_T^2}{T+1} + \frac{2M_3}{T^{(1-p)/2}}.$$

By Condition 6.2, we observe that

$$\lim_{T\to\infty} T^{(1-p)/2} \left(\frac{(2L_g + 3Q)d_T^2}{T+1} + \frac{2M_3}{T^{(1-p)/2}}\right) = \lim_{T\to\infty} \left((2L_g + 3Q)\left(\frac{d_T}{T^{p/4}}\right)^2 \frac{T^{1/2}}{T+1} + 2M_3\right) = 2M_3,$$

which implies there exists $M_4 > 0$ such that if $T > \max\{t_0, 1\}$, then

$$g(x_T) - g_{\text{opt}} \leq \frac{(2L_g + 3Q)d_T^2}{T+1} + \frac{2M_3}{T^{(1-p)/2}} \leq \frac{M_4}{T^{(1-p)/2}}.$$

Thus, we finish the proof. □

Theorem 6.10. *Suppose* $\{x_t\}_{t\in[T]}$ *is the sequence generated by Algorithm 5 with parameters* $\{\alpha_t, \beta_t, \gamma_t, \tau_t\}_{t\geq 0}$ *as given in Lemma 6.7. If Assumption 1, Assumptions 4{6 and Condition 6.2 hold, then*

$$\begin{aligned} g(x_T) - g_{\text{opt}} &\leq O\left(\max\left\{\frac{1}{T^{1-p}}, \frac{d_T^2}{T^{1/2}}\right\}\right), \\ -O\left(\max\left\{\frac{1}{T^{1-p}}, \frac{d_T^2}{T^{1/2}}\right\}\right) \leq f(x_T) - f_{\text{opt}} &\leq O\left(\max\left\{\frac{1}{T^{1-p}}, \frac{d_T^4}{T^p}\right\}\right). \end{aligned} \tag{6.35}$$

Proof. By Lemma 6.1 and Corollary 6.8, if $T > \max\{t_0, 1\}$, we have

$$\begin{aligned} &-\lambda_{\text{opt}}\left(g(x_T) - g_{\text{opt}}\right) \leq f(x_T) - f_{\text{opt}} \leq C_T, \\ &g(x_T) - g_{\text{opt}} \leq g_T - g_{\text{opt}} + B_T + 2\sqrt{2A_TC_T + 2\lambda_{\text{opt}}A_T\left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}}A_T\right)^2}. \end{aligned}$$

By (6.33), (6.34), (we note that Assumption 3 was not used to prove these two results) the fact that $d_T^4 \geq d_T^2 d_0^2$ and $T+1 > T \geq T^p$, we observe that

$$\begin{aligned} &2A_TC_T + 2\lambda_{\text{opt}}A_T\left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}}A_T\right)^2 \\ &\leq 2\frac{R(T+1)^p}{T}\left((M_1+M_2)\max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + \lambda_{\text{opt}}\frac{(2L_g+3Q)d_T^2}{T+1} + 2\lambda_{\text{opt}}^2\frac{R(T+1)^p}{T}\right) \\ &\leq 2\frac{R(2T)^p}{T}\left((M_1+M_2)\max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + \frac{\lambda_{\text{opt}}(2L_g+3Q)}{d_0^2}\frac{d_T^4}{T+1} + 2\lambda_{\text{opt}}^2\frac{R(2T)^p}{T}\right) \\ &\leq \frac{R2^{p+1}}{T^{1-p}}\left((M_1+M_2)\max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\} + \frac{\lambda_{\text{opt}}(2L_g+3Q)}{d_0^2}\frac{d_T^4}{T^p} + \frac{\lambda_{\text{opt}}^2R2^{p+1}}{T^{1-p}}\right) \\ &\leq \frac{R2^{p+1}}{T^{1-p}}\left(M_1+M_2+\frac{\lambda_{\text{opt}}(2L_g+3Q)}{d_0^2} + \lambda_{\text{opt}}^2R2^{p+1}\right)\max\left\{\frac{d_T^4}{T^p}, \frac{1}{T^{1-p}}\right\}. \end{aligned}$$

By defining $M_5 > 0$ such that $M_5^2 = R2^{p+1}\left(M_1+M_2+\frac{\lambda_{\text{opt}}(2L_g+3Q)}{d_0^2} + \lambda_{\text{opt}}^2R2^{p+1}\right)$, we have that if $T > \max\{t_0, 1\}$, then

$$2A_TC_T + 2\lambda_{\text{opt}}A_T\left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}}A_T\right)^2 \leq M_5^2\max\left\{\frac{d_T^4}{T}, \frac{1}{T^{2-2p}}\right\}.$$

Hence, by (6.33) and the fact that $T+1 > T \geq T^{1/2}$, we have that if $T > \max\{t_0, 1\}$, then

$$\begin{aligned} g(x_T) - g_{\text{opt}} &\leq g_T - g_{\text{opt}} + B_T + 2\sqrt{2A_TC_T + 2\lambda_{\text{opt}}A_T\left(g_T - g_{\text{opt}} + B_T\right) + 4\left(\lambda_{\text{opt}}A_T\right)^2} \\ &\leq \frac{(2L_g + 3Q)d_T^2}{T+1} + 2M_5 \max\left\{\frac{d_T^2}{T^{1/2}}, \frac{1}{T^{1-p}}\right\} \\ &\leq \frac{(2L_g + 3Q)d_T^2}{T^{1/2}} + 2M_5 \max\left\{\frac{d_T^2}{T^{1/2}}, \frac{1}{T^{1-p}}\right\} \\ &\leq (2L_g + 3Q + 2M_5) \max\left\{\frac{d_T^2}{T^{1/2}}, \frac{1}{T^{1-p}}\right\}. \end{aligned}$$

Therefore, (6.35) is true. □

Chapter 7

Numerical experiments

7.1 Markowitz portfolio optimisation

We revisit the bilevel variant of the Markowitz portfolio optimisation problem described in Example 1.2, originally considered by Beck and Sabach [4, Section 5.1], which is formulated as follows:

$$\begin{aligned} \min_{x} \quad & \frac{1}{2}\|x-a\|_2^2 \\ \text{s.t} \quad & x \in \arg\min_{z} \left\{ \frac{1}{2} z^T \Sigma z \;\middle|\; \mu^T z \geq r_0, \mathbf{1}^T z = 1, z \geq 0 \right\}, \end{aligned} \tag{7.1}$$

where a, Σ, μ, r_0 are defined as in Example 1.2. It is easy to verify that (7.1) satisfies Assumption 1 with $L_g = \sigma_{\max}(\Sigma)$ and $L_f = 1$ with respect to the Euclidean norm. Since the base domain of this instance is bounded, Assumptions 2–3 are true.

7.1.1 Data description

Following [4, Section 5.1], we utilised a real data set from Vanderbei [36], which stores the yearly returns from 1973 to 1994 of $n = 8$ types of assets, including US 3-month treasury bills, US government long bonds, SP 500, Wilshire 500, NASDAQ composite, corporate bond index, EAFE

and Gold. The data between the years 1974 and 1977 were utilised to estimate μ and Σ as follows:

$$\mu = \frac{1}{\mathcal{T}} R\mathbf{1}, \quad \Sigma = \frac{1}{\mathcal{T}-1} R\left(\mathbf{I}_{\mathcal{T}} - \frac{1}{\mathcal{T}}\mathbf{1}\mathbf{1}^T\right) R^T,$$

where $\mathcal{T} = 4$ and R is an 8×4 matrix containing the assets' returns for each of the 4 years. Since the rank of R^T is at most 4, the rank of Σ cannot be greater than 4, which implies Σ is singular and hence not positive definite. To induce multiple solutions for the inner-level objective over the base domain, we set $r_0 = 1.05$ as recommended in [4, Section 5.1].

7.1.2 Algorithms

Using frameworks in Chapters 4 to 6, we implemented Algorithms 3 to 5 to solve problem (7.2). As the base domain is bounded, we set $B_t := X$ for each $t \geq 0$.

For `SL-CG`, we adopted the stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 4.4 and generated $\{g_t\}_{t\geq 0}$ as outlined in Section 3.3.

For `IR-CG`, we adopted the recommended schedule for stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 5.10 but for regularisation parameters $\sigma_t = 0.1(t+1)^{1/2}$ for each $t \geq 0$, which ensures inner- and outer-level objectives converge at rate $O(1/T^{1/2})$. We note that this parameter choice for regularisation parameters does not violate Conditions 5.1–5.3 since they are invariant under positive scaling.

For `PD-CG`, we chose the parameters $\{\alpha_t, \beta_t, \gamma_t, \tau_t, g_t\}_{t\geq 0}$ as provided in Lemma 6.7 and strategy outlined in Section 3.3, where $u_{-1} = 300$, $R = 300$, and p was chosen to be $\frac{1}{3}$ to ensure rates for both inner- and outer-level objectives are $O(1/T^{1/3})$.

For performance comparison, we also implemented `IR-PG` [34], `Bi-SG` [29], `CG-BiO` [24], and `ITALEX` (projection-free version) [12] . We chose the parameters for the implementation of these algorithms based on the criteria described in the corresponding papers. Specifically, for `CG-BiO`, we set the stepsizes to be $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$ and $\epsilon_g = 10^{-3}$. Following the notation in the original papers: for `IR-PG`, we set $\theta = \tilde{\alpha} = \eta = \frac{1}{3}$ and regularisation parameters $\sigma_t = 0.1(t+1)^{1/2}$ for each $t \geq 0$; for `Bi-SG`, we set $c = \min\left\{\frac{1}{L_f}, 1\right\} = 1$ and $\alpha = \frac{1}{2-0.01}$ to ensure the convergence rates of both inner- and outer-level objectives close to $O\left(1/T^{1/2}\right)$, for `ITALEX`, we set $\epsilon_1 = 10^{-2}$, $\alpha_0 = 0$,

$z_0 = 0 \in \mathsf{R}^n$, and u_0 to be the same as the starting point x_0, which was generated as discussed below.

The starting points for all algorithms except the `CG-BiO` method were set the be the point x_0 constructed by the following procedure. First, we determined the set $\mathcal{K} := \{i \in [n] \mid \mu_i \geq r_0\}$, which must be non-empty since otherwise, for any inner feasible x, $\mu^\top x \leq (\max_{i\in[n]} \mu_i)\mathbf{1}^\top x < r_0$. Second, for any $i \in [n]$, we set $(x_0)_i = \frac{1}{|\mathcal{K}|}$, where $|\mathcal{K}|$ is the number of elements in $\mathcal{K}$, if $i \in \mathcal{K}$ and $(x_0)_i = 0$ otherwise.

Initialising `CG-BiO` required an inner feasible x_0' that satisfies $g(x_0') - g_{\text{opt}} \leq \frac{\epsilon_g}{2}$. To generate this point, we ran the `CG` method [17] for the inner-level objective with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$ which terminated when the surrogate gap $S(x_t)$ as defined in (2.23) was not greater than $\frac{\epsilon_g}{2}$. We initialised this step with the starting point x_0 of other algorithms.

For the `IR-CG` and `PD-CG` methods, we solved the linear minimisation subproblem over the base domain of (7.1), which is discussed in Section 7.4.1. For the `ITALEX` with projection-free customisation method, we solved two linear minimisation subproblems: one over the base domain, which is similar to that of `IR-CG` and `PD-CG`, and one over a sublevel set of the outer-level objective. Specifically, we computed a minimiser x^* of a linear function $c^\top x$ over a sublevel set of outer-level objective, i.e., $\left\{x \in \mathsf{R}^n \mid \frac{1}{2}\|x - a\|_2^2 \leq \alpha\right\}$ for some $\alpha \geq 0$ as follows:

$$x^* = \begin{cases} 0 & c = 0 \\ -\frac{\sqrt{2\alpha}}{\|c\|_2} c + a & c \neq 0. \end{cases}$$

For the `SL-CG` and `CG-BiO` methods, we solved the linear minimisation oracle over the base domain intersecting with a half-space by the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11]. To compute the projection under the Euclidean norm onto the base domain, which was required for the `IR-PG` and `Bi-SG` methods, we used the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11].

To approximate the value of g_{opt}, we numerically optimised the inner-level objective over the base domain by the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11].

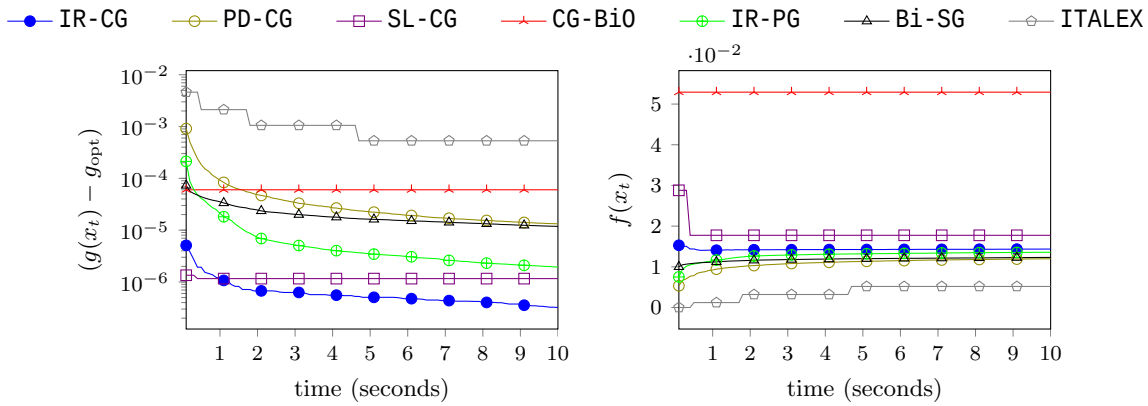


Figure 7.1: Plot of the best inner-level objective value found by each algorithm (left) and the corresponding outer-level objective value (right) on the Markowitz portfolio instance, at each point in time. Note that y-axis is in logarithmic scale on the left figure.

Method	Number of iterations executed
SL-CG	496
IR-CG	4453
PD-CG	4225
IR-PG	1026
Bi-SG	1051
CG-BiO	1014
ITALEX	5

Table 7.1: Comparison of the number of iterations by the algorithms, on the Markowitz portfolio instance, executed within 10 seconds.

We set a time limit of 10 seconds for all algorithms. All experiments were run on a server with a 2.4GHz processor and 32 GB memory, using Python 3.10.9. For certain subroutines, we also used the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11] and `scipy.spatial.ConvexHull` package.

7.1.3 Results comparison

Fig. 7.1 illustrates the values of inner optimality gap (on the left) and outer-level objective (on the right) generated by `SL-CG`, `IR-CG`, `PD-CG`, `IR-PG`, `Bi-SG`, `CG-BiO`, `ITALEX` within 10 seconds.

Table 7.1 shows the number of iterations executed by `SL-CG`, `IR-CG`, `PD-CG`, `IR-PG`, `Bi-SG`, `CG-BiO`, `ITALEX` within 10 seconds. In terms of the inner optimality gap, we observe that `IR-CG` performs the best and is followed by `PD-CG`. We highlight that the relatively poor performance of `ITALEX` for this instance is indeed anticipated due to the fact that the method needs an inner loop, whose iteration also contains another loop, at each iteration. Hence, it may be time-consuming to complete an iteration, which explains only 5 iterations performed by `ITALEX`. Although the theoretical convergence rates for `SL-CG` and `CG-BiO` for this instance are both $O(1/T)$, their complicated linear minimisation oracle as compared to that of `IR-CG` and `PD-CG` lead to their relatively poorer performance via fewer executed iterations. We would like to point out that the inner optimality gaps generated `CG-BiO` seem to not go below the level of 5×10^{-4}, which matches our discussion about the lack of asymptotic convergence of this method in Remark 4.2.1. Fig. 7.1 also highlights that the outer-level objective values of these algorithms are directly correlated to the inner optimality gaps. We note that some methods converge from below in terms of the outer-level objective due to the super-optimality phenomenon as discussed in Section 2.3.

7.2 Low-rank matrix completion

We perform numerical experiments on the bilevel variant of the low-rank matrix completion problem described in Example 1.3, which is formulated as follows:

$$\begin{aligned} \min_{X \in \mathbf{R}^{n \times p}} \quad & \frac{1}{2} \sum_{j \in [p]} \sum_{i \in [n]} \left(X_{i,j} - \overline{X}_j \right)^2 \\ \text{s.t} \quad & X \in \arg\min_{\|Z\|_* \leq \delta} \left\{ \frac{1}{2} \sum_{(i,j) \in \Omega} (Z_{i,j} - M_{i,j})^2 \right\}, \end{aligned} \tag{7.2}$$

where

$$\overline{X}_j := \frac{1}{n} \sum_{i \in [n]} X_{i,j}, \quad \forall j \in [p].$$

First, it is easy to see that the base domain and g satisfy Assumption 1(a) and Assumption 1(c) with $L_g = 1$ in the Frobenius norm $\|\cdot\|_F$, defined as the square root of the sum of squared entries.

Since the base domain is also bounded and f is continuous, Assumption 2 and Assumption 3 hold. To verify Assumption 1(b), we rewrite f as

$$f(X) = \frac{1}{2}\|UX\|_F^2, \tag{7.3}$$

where

$$U := \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top.$$

As f is the sum of the square of linear functions of entries of X, it is a convex quadratic function. We can see that $U^\top U = U$, which implies U is positive semi-definite and has the largest eigenvalue of 1. In addition, the gradient of f is

$$\nabla f(X) = U^\top UX = UX.$$

Given any $X, Y \in \mathsf{R}^{n\times p}$, we let $z_1, \ldots, z_p$ be the columns of $(X - Y)$ and observe that

$$\|\nabla f(X) - \nabla f(Y)\|_F^2 = \|U(X-Y)\|_F^2 = \sum_{j\in[p]} \|Uz_j\|_2^2 \le \sum_{j\in[p]} \|z_j\|_2^2 = \|X-Y\|_F^2.$$

By using the fact that the Frobenius norm is self-dual, f satisfies the smoothness assumption with $L_f = 1$.

7.2.1 Data description

We used the MovieLens 1M data set [21]. This data set contains ratings of 3952 movies from 6040 users, made on a 5-star scale. Therefore $n = 6040$, $p = 3952$, and each $M_{i,j} \in [5]$ for $(i,j) \in \Omega$. In the dataset, there are $|\Omega| = 1,000,209$ observed entries, which is $\approx 4.19\%$ of total possible entries. In our experiments, we set the nuclear norm radius to be $\delta = 5$.

7.2.2 Algorithms

Using frameworks in Chapters 4 to 6, we implemented Algorithms 3 to 5 to solve problem (7.2). We set B_t to be the base domain of problem (7.2) for $t \geq 0$.

For `SL-CG`, we adopted the stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 4.4 and generate $\{g_t\}_{t\geq 0}$ as outlined in Section 3.3.

For `IR-CG`, we adopted the recommended schedule for stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 5.10 but for regularisation parameters $\sigma_t = 0.05(t+1)^{1/2}$ for each $t \geq 0$, which ensures inner- and outer-level objectives converge at rate $O(1/T^{1/2})$. We note that this parameter choice for regularisation parameters does not violate Conditions 5.1–5.3 since they are invariant under positive scaling.

For `PD-CG`, we chose the parameters $\{\alpha_t, \beta_t, \gamma_t, \tau_t, g_t\}_{t\geq 0}$ as provided in Lemma 6.7 and strategy outlined in Section 3.3, where $u_{-1} = 50$, $R = 10$, and p was chosen to be $\frac{1}{3}$ to ensure rates for both inner- and outer-level objectives are both $O(1/T^{1/3})$.

Since $f(X)$ attains it minimum over $\mathsf{R}^{n\times p}$ at any matrix of the form $\alpha\mathbf{1}\mathbf{1}^\top$ for $\alpha \in \mathsf{R}$, sublevel sets of f are not compact. Hence, a critical assumption of the `ITALEX` method with projection-free customisation [12] is violated. As a result, for performance comparison, we only implemented `CG-BiO` [24], `IR-PG` [34], and `Bi-SG` [29]. We chose the parameters for the implementation of these algorithms based on the criteria described in the corresponding papers. Specifically, for `CG-BiO`, we set the stepsizes to be $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$ and $\epsilon_g = 10^{-4}$. Following the notation in the original papers: for `IR-PG`, we set $\theta = \tilde{\alpha} = \eta = \frac{1}{3}$ and regularisation parameters $\sigma_t = 0.05(t+1)^{1/2}$ for each $t \geq 0$; or `Bi-SG`, we set $c = \min\left\{\frac{1}{L_f}, 1\right\} = 1$ and $\alpha = \frac{1}{2-0.01}$ to ensure the convergence rates of both inner- and outer-level objectives close to $O\left(1/T^{1/2}\right)$.

The starting points for all algorithms except `CG-BiO` were set to be the following matrix:

$$X_0 := 0.01 \times \delta \begin{bmatrix} \mathbf{I}_p/p & \mathbf{0}_{p\times(n-p)} \end{bmatrix}^\top.$$

For `CG-BiO`, we required an inner feasible point X_0' that satisfies $g(X_0') - g_{\text{opt}} \leq \epsilon_g/2$. To generate such point, we ran `CG` [17] for the inner-level objective with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$ until the surrogate gap $S(x_t)$ as defined in (2.23) was not greater than $\frac{\epsilon_g}{2}$. We initialised this phase with X_0

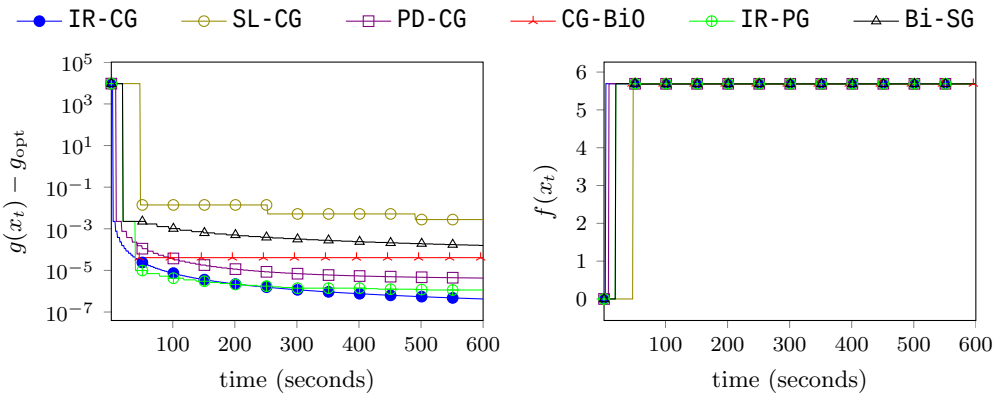


Figure 7.2: Plot of the best inner-level objective value found by each algorithm (left) and the corresponding outer-level objective value (right) on the low-rank matrix completion instance, at each point in time. Note that y-axis is in logarithmic scale on the left figure.

given above.

For the `IR-CG` and `PD-CG` methods, we solved the linear minimisation subproblem over the nuclear norm ball, whose solution is discussed in Section 7.4.2. For the `SL-CG` and `CG-BiO` methods, we solved the linear minimisation subproblem over the nuclear norm ball intersecting with a half-space, which is discussed in Section 7.4.3. To compute the projection onto the nuclear norm ball required for the `IR-PG` and `Bi-SG` methods, we followed the steps given in Section 7.4.4.

To approximate the inner optimal value g_{opt}, we implemented `CG` [17] starting from X_0 with stepsizes $\alpha_t = \frac{2}{t+2}$ for $t \geq 0$ to retrieve a 10^{-5}-sub-optimal solution using duality surrogate gap as stopping criterion. Then we used this suboptimal solution as a starting point for the implementation of another `CG` [17] to obtain a 10^{-12}-suboptimal solution and g_{opt} was approximated by the corresponding inner-level objective value. We found that this warm-up approximation scheme saved time significantly compared to running `CG` [17] only once.

We set a time limit of 10 minutes (600 seconds) for all algorithms. All experiments were run on a server with a 2.4GHz processor and 32 GB memory, using Python 3.10.9. For certain subroutines, we also used the bounded Brent method [16] via package `scipy.optimize.minimize-scalar` (version 1.11.3) [37].

Method	Number of iterations executed
SL-CG	4
IR-CG	164
PD-CG	71
CG-BiO	3
IR-PG	20
Bi-SG	33

Table 7.2: Comparison of the number of iterations executed by the algorithms on the low-rank matrix completion instance, within 10 minutes.

7.2.3 Results comparison

Fig. 7.2 illustrates the values of the inner optimality gap (on the left) and outer-level objective (on the right) generated by SL-CG, IR-CG, PD-CG, CG-BiO, IR-PG, Bi-SG within 10 minutes. Table 7.2 shows the number of iterations executed by SL-CG, IR-CG, PD-CG, CG-BiO, IR-PG, Bi-SG within 10 minutes. Regarding the inner optimality gap, we observe that IR-CG and IR-PG perform the best and are followed by PD-CG. According to Table 7.2, despite only performing 20 iterations as compared to 164 iterations, the last iteration of IR-PG is comparable to that of IR-CG, which can be anticipated because the former method exploits somewhat adaptive stepsizes selection scheme via an inner loop which guarantees a sufficient improvement in the current regularised objective while IR-CG adopts a schedule of stepsizes that only depends on t. CG-BiO makes only a modest improvement compared to the initialised point X_0' since it can only perform 3 iterations. This can be explained by the complicated structure of the linear minimisation subproblem as compared to that of IR-CG and PD-CG. This is also the justification for the poor performance of SL-CG, which can only perform a total of 4 iterations. The quality of the sequence $\{g_t\}_{t\in[T]}$ generated for SL-CG in parallel with the iterates within this time limit is inferior to $g(X_0')$ generated by the initialisation phase of CG-BiO, which demands 10 iterations. Although Bi-SG is known to have a theoretical convergence rate of $O(1/T^{1/(2-0.01)})$ for the inner-level objective in this particular problem class, the fact that the total number of iterations executed is 33 leads to its inferior performance as compared to IR-CG and PD-CG, which run 164 and 71 iterations, respectively. Fig. 7.2 also highlights that the outer-level objective values of these algorithms are directly correlated to the inner optimality gaps. The reason

we observe the outer-level objective values increase over time for some methods can be explained by the super-optimality of the iterates as discussed in Section 2.3.

7.3 Linear inverse problem

Given data $A \in \mathsf{R}^{m\times n}, b \in \mathsf{R}^n$, the goal of the linear inverse problem is to obtain a solution $x \in \mathsf{R}^n$ to the linear system of equations $Ax = b$. If A is rank-deficient, there can be either multiple solutions or no exact solution. To deal with the potential nonexistence of a solution, the inner-level objective is selected to minimise the squared residuals:

$$\min_{x\in X} \quad \frac{1}{2}\|Ax - b\|_2^2. \tag{7.4}$$

In case A is rank-deficient, (7.4) may not have a unique solution. Thus, the second criterion is used to select a single solution out of the optimal set:

$$\min_{x\in X_{\text{opt}}} \quad \frac{1}{2}x^\top Qx, \tag{7.5}$$

where $Q \in \mathsf{R}^{n\times n}$ is a positive definite matrix. To test the performance of the `SL-CG`, `IR-CG`, and `PD-CG` methods on an unbounded domain, we let $X? := \mathsf{R}^n_+$. We note that (7.5) satisfies Assumption 1 with $L_g = \sigma^2_{\max}(A)$ and $L_f = \sigma_{\max}(Q)$ in the Euclidean norm. Since for any $x \in \mathsf{R}^n$, $f(x) \geq \frac{1}{2}\sigma_{\min}(Q)\|x\|_2^2$, where $\sigma_{\min}(Q)$ is the smallest eigenvalue of Q, Assumptions 2–3 hold.

7.3.1 Data description

For this problem class, there are three standard test problems `foxgood`, `baart` and `phillips` that originate from the MATLAB package "regularisation tools". Based on parameters such as the problem dimension, this MATLAB package follows a deterministic scheme to generate the problem instances with the specific problem characteristics. More specifically, for each of the test problems, there is a corresponding function that generates a tuple of data $A_+ \in \mathsf{R}^{n\times n}, b_+ \in \mathsf{R}^n, x_+ \in \mathsf{R}^n$ that satisfy $A_+x_+ = b_+$. We constructed our instances using these functions by letting $m = n =$

$1000, A := A_+, b := b_+ + \rho\epsilon$, where $\epsilon \in \mathsf{R}^n$ is a random vector sampled from the standard normal distribution to model additive noise, and $\rho = 10^{-2}$ determines the magnitude of the noise. We set $Q := L_+L_+^\top + \mathbf{I}_n$, where $L_+ \in \mathsf{R}^{(n+1)\times n}$ is generated by the function `get-l` from the same package.

7.3.2 Algorithms

Using frameworks in Chapters 4 to 6, we implemented Algorithms 3 to 5 to solve problem (7.2). As suggested by Example 3.1, Condition 4.1, Condition 5.4, Lemma 5.9 and Condition 6.2, we set the coverings as follows:

$$B_t := \{x \in \mathsf{R}^n \mid 0 \leq x \leq (\log(t+2))\mathbf{1}\}, \quad \forall t \geq 0.$$

For `SL-CG`, we adopted the stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 4.4 and generate $\{g_t\}_{t\geq 0}$ as outlined in Section 3.3.

For `IR-CG`, we adopted the recommended schedule for stepsizes $\{\alpha_t\}_{t\geq 0}$ as in Theorem 5.10 but for regularisation parameters $\sigma_t = 0.01(t+1)^{1/2}$ for each $t \geq 0$, which ensures inner- and outer-level objectives converge at rates $O(1/T^{1/2})$ and $O(\log^2(T)/T^{1/2})$ respectively. We note that this parameter choice for regularisation parameters does not violate Conditions 5.1–5.3 since they are invariant under positive scaling.

For `PD-CG`, we chose the parameters $\{\alpha_t, \beta_t, \gamma_t, \tau_t, g_t\}_{t\geq 0}$ as provided in Lemma 6.7 and strategy outlined in Section 3.3, where $u_{-1} = 200$, $R = 100$, and p was chosen to be $\frac{1}{3}$ to ensure rates for inner- and outer-level objectives are $O(1/T^{1/3})$ and $O(\log^4(T)/T^{1/3})$ respectively.

Since X is unbounded, and to the best of our knowledge, there is no first-order projection-free method that is provided with a feasible starting point and a tolerance $\epsilon > 0$ and returns a ϵ-sub-optimal solution of minimising g over X. Hence, a critical assumption of the `ITALEX` method with projection-free customisation [12] is violated. Moreover, as the base domain is unbounded, `CG-BiO` [24] is not applicable. As a result, for performance comparison, we only implemented `IR-PG` [34], and `Bi-SG` [29]. We chose the parameters for the implementation of these algorithms based on the criteria described in the corresponding papers. Following the notation in the original papers: for

IR-PG, we set $\theta = \tilde{\alpha} = \eta = \frac{1}{3}$ and regularisation parameters $\sigma_t = 0.01(t+1)^{1/2}$ for each $t \geq 0$; or Bi-SG, we set $c = \min\left\{\frac{1}{L_f}, 1\right\}$ and $\alpha = 1/(2-0.01)$ to ensure the convergence rates of both inner- and outer-level objectives close to $O\left(1/T^{1/2}\right)$.

The starting points for all algorithms were set to be $x_0 = \mathbf{1}$. For the IR-CG and PD-CG methods, we solved the linear minimisation subproblem over a high-dimensional box. That is, we have to solve a linear problem of the form

$$\begin{aligned} \min_{x \in \mathsf{R}^n} \quad & c^\top x \\ \text{s.t.} \quad & 0 \leq x \leq r\mathbf{1}, \end{aligned} \tag{7.6}$$

where $c \in \mathsf{R}^n, d \in \mathsf{R}, r > 0$ are given. A solution x^* of (7.6) is as follows:

$$(x^*)_i = \begin{cases} 0 & c_i \geq 0 \\ r & c_i < 0, \end{cases} \qquad \forall i \in [n].$$

For the SL-CG method, we solved the linear minimisation subproblem over a high-dimensional box intersecting with a half-space, which is discussed in Section 7.4.5. To compute the projection of a point x onto the base domain required for the IR-PG and Bi-SG method, we used the well-known result

$$\mathrm{Proj}_{\mathsf{R}^n_+}(x) = ([x_1]_+, \ldots, [x_n]_+).$$

To approximate the inner optimal value g_{opt}, we numerically optimised the inner-level objective over the base domain by the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11].

We set a time limit of 10 seconds for all algorithms. All experiments were run on a server with a 2.4GHz processor and 32 GB memory, using Python 3.10.9. For certain subroutines, we also used the MOSEK solver (version 10.0.40) [2] via CVXPY package (version 1.3.1) [11].

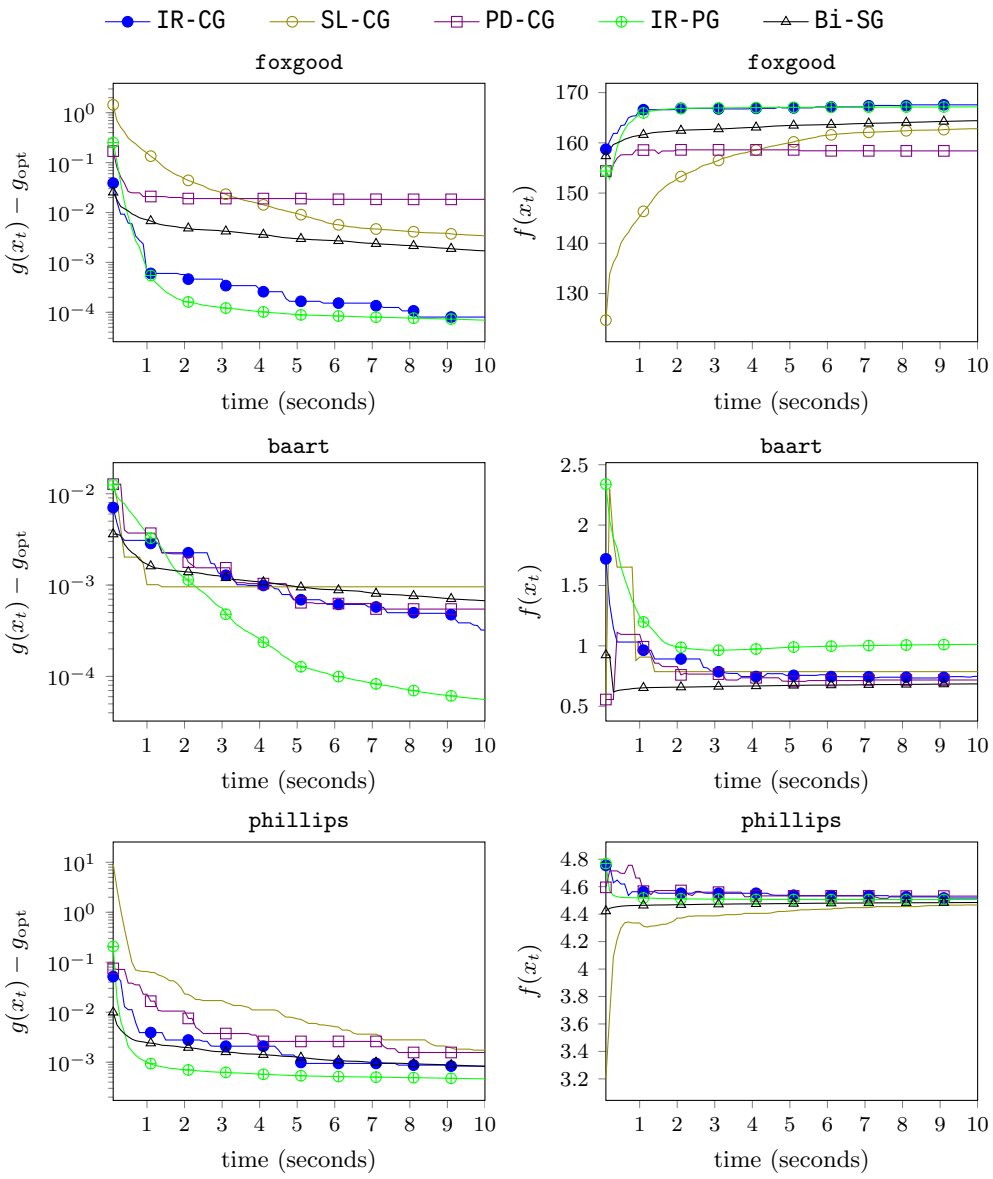


Figure 7.3: Plot of the best inner-level objective value found by each algorithm (left) and the corresponding outer-level objective value (right) on linear inverse problem instances `foxgood`, `baart`, and `phillips`, at each point in time. Note that y-axis is in logarithmic scale on the left figures.

Method	Number of iterations executed		
	foxgood	baart	phillips
SL-CG	1509	1720	2790
IR-CG	7927	7362	8170
PD-CG	3545	4331	3867
IR-PG	5802	6086	6405
Bi-SG	12634	11869	11849

Table 7.3: Comparison of the number of iterations executed by the algorithms on linear inverse problem instances foxgood, baart, and phillips, within 10 seconds.

7.3.3 Results comparison

Fig. 7.3 illustrates the values of the inner optimality gap (on the left) and outer-level objective (on the right) generated by SL-CG, IR-CG, PD-CG,IR-PG, Bi-SG within 10 seconds. Table 7.3 shows the number of iterations executed by SL-CG, IR-CG, PD-CG, IR-PG, Bi-SG within 10 seconds. Regarding the inner optimality gap, we observe that IR-PG performs the best and is followed by IR-CG and Bi-SG. We highlight that the better performance of projection-based methods for these instances is indeed anticipated since the projection onto R^n_+ is simpler to compute as compared to the linear minimisation over the truncations. This observation can be confirmed by examining Table 7.2, which shows that the number of iterations executed by IR-CG is nearly a quarter of that of IR-PG over three instances. The complicated structure of the linear minimisation subproblem over a high-dimensional box intersecting with a half-space justifies the relatively poor performance of SL-CG on foxgood and phillips, where the total of iterations are 1509 and 2790. Despite having a simple linear subproblem over a high-dimension box, PD-CG has to call two oracles at each iteration: one for generating g_t and one for computing the primal variable x_t, which explains for lower number of iterations as compared to most methods. Although Bi-SG is known to have a theoretical convergence rate of $O(1/T^{1/(2-0.01)})$ for the inner-level objective in this particular problem class, the fact that the total number of iterations executed is twice the other methods on three instance leads to its superior performance as compared to SL-CG, IR-CG, PD-CG. Fig. 7.3 also highlights that the outer-level objective values of these algorithms are directly correlated to the inner optimality gaps. The reason we observe the outer-level objective values of some methods increase over time is the

super-optimality of the iterates as discussed in Section 2.3.

7.4 Subroutines implementation

7.4.1 Linear minimisation over the sliced probability simplex

In this subsection, we provide a scheme to tackle the subproblem of the `IR-CG` and `PD-CG` methods and a part of the subproblem of the `ITALEX` method with projection-free customisation in solving problem (7.1). That is, we are solving

$$\begin{aligned} \min_{x} \quad & c^\top x \\ \text{s.t.} \quad & \mu^\top x \geq r_0, \quad \mathbf{1}^\top x = 1, \quad x \geq 0, \end{aligned} \tag{7.7}$$

where $c \in \mathsf{R}^n$ is given. We note that we only consider the case problem (7.7) is feasible, which happens if and only if there exists $i \in [n]$ such that $\mu_i \geq r_0$. The dual problem is

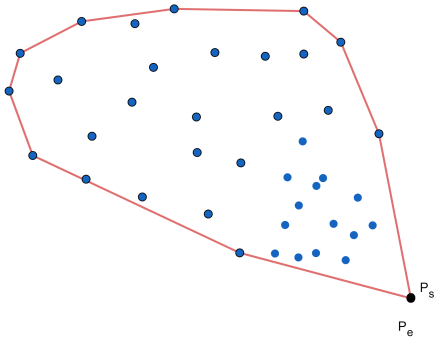
$$\begin{aligned} \max_{\lambda,\eta} \quad & -\lambda + r_0\eta \\ \text{s.t.} \quad & c_i \geq -\lambda + \mu_i\eta, \quad \forall i \in [n], \\ & \eta \geq 0. \end{aligned} \tag{7.8}$$

Geometrically, problem (7.8) can be interpreted as follows:

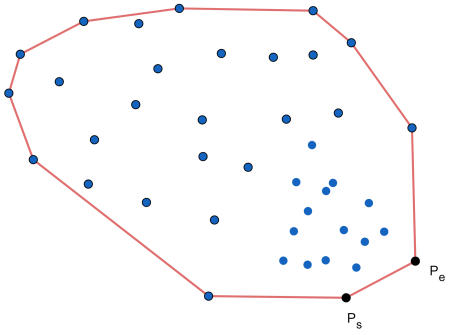
- We are given points $\{P_i\}_{i\in[n]} \subset \mathsf{R}^2$ where $P_i := (\mu_i, c_i)$ for each $i \in [n]$.
- Let $\lambda \in \mathsf{R}, \eta \geq 0$ be two numbers such that the half-space $\{(\mu, c) \in \mathsf{R}^2 \mid c \geq -\lambda + \eta\mu\}$ contains $\{P_i\}_{i\in[n]}$.
- Find λ, η satisfying the above conditions such that $-\lambda + \eta r_0$ is maximised.

We denote $\mathbf{P} := \text{Conv}\left(\{P_i\}_{i\in[n]}\right)$, and two points

$$P_s := (\mu_s, c_s), \quad \text{where} \quad c_s = \min_{k\in[n]} c_k, \quad \mu_s = \max_{j\in[n]} \{\mu_j \mid c_j = c_s\}, \tag{7.9}$$



(a) $P_s = P_e$



(b) $P_s \neq P_e$

Figure 7.4: Two examples of the points $\{P_i\}_{t\in[n]}$, the vertices of $\mathbf{P}$, and P_s, P_e.

and

$$P_e := (\mu_e, c_e) \quad \text{where} \quad \mu_e = \max_{k\in[n]} \mu_k, \quad c_e = \min_{j\in[n]} \{c_j \mid \mu_j = \mu_e\}. \tag{7.10}$$

We note that P_s and P_e can be the same point; see Fig. 7.4a. When $P_e = P_s$, the solution of problem (7.7) is given in the below lemma.

Lemma 7.1. *Suppose $P_s = P_e$ and $t_1 \in [n]$ is an integer such that $P_{t_1} = P_s = P_e$. Then $x^* = e_{t_1}$ is an optimal solution to problem (7.7).*

Proof. Since $P_s = P_e$, we have $r_0 \leq \mu_s, \mu_e$. We also have that for each $i \in [n]$, $c_i \leq c_s = c_e$ and

$\mu_i \geq \mu_s = \mu_e$. Given a feasible λ, μ, we have

$$-\lambda + \eta r_0 \leq -\lambda + \eta \mu_{t_1} \leq c_{t_1}.$$

Thus, the optimal dual value is $c_{t_1} = c_s$, and one optimal dual solution is $(\lambda^*, \eta^*) = (-c_s, 0)$ (the feasibility of this solution can be easily checked by the definition of c_s).

Let x^* be an optimal solution of (7.7). Since x^* must minimise the Lagrangian $L(x, \lambda^*, \eta^*) = (c + \lambda^* \mathbf{1} - \eta^* \mu)^\top x - \lambda^* + r_0 \eta^*$ over $x \geq 0$, we have that if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k = c_k - c_s = 0$ then $(x^*)_k \geq 0$, and if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k = c_k - c_s > 0$ then $(x^*)_k = 0$ for each $k \in [n]$. Therefore, x^* must satisfy

$$\sum_{k \in [n]: c_k = c_s} (x^*)_k = 1, \qquad \sum_{k \in [n]: c_k = c_s} \mu_k (x^*)_k \geq r_0.$$

To ensure the feasibility of x^* for problem (7.7), we set $x^* = e_{t_1}$ since P_{t_1} is the point with largest x coordinate among points with y coordinate of $c_{t_1} = c_s$. □

Now, we consider when $P_s \neq P_e$, i.e., $c_s < c_e, \mu_s < \mu_e$. We let $\mathcal{I} \subseteq [n]$ be a set such that $\mathrm{Conv}\left(\{P_i\}_{i \in \mathcal{I}}\right) = \mathbf{P}$ and that there exists no $j \in \mathcal{I}$ such that P_j can be written as a convex combination of $\{P_i\}_{i \in \mathcal{I} \setminus \{j\}}$. That is, $\mathcal{I}$ is the index set of the vertices of $\mathbf{P}$.

By definition, P_s, P_e are vertices of $\mathbf{P}$, which implies that $P_s, P_e \in \{P_i\}_{i \in \mathcal{I}}$. Indeed, we assume that P_s is a convex combination of some points in $\{P_i\}_{i \in \mathcal{I}}$. By Caratheodory's theorem, we can find three such points $P_{i_1}, P_{i_2}, P_{i_3}$ with $i_1, i_2, i_3 \in \mathcal{I}$ (since the dimension is 2). Then there exists $\lambda_1, \lambda_2, \lambda_3 \geq 0$ such that

$$\lambda_1 c_{i_1} + \lambda_2 c_{i_2} + \lambda_3 c_{i_1} = c_s, \quad \lambda_1 \mu_{i_1} + \lambda_2 \mu_{i_2} + \lambda_3 \mu_{i_1} = \mu_s, \quad \lambda_1 + \lambda_2 + \lambda_3 = 1.$$

However c_s is the minimum of the c_i's, so we must have $c_{i_1} = c_{i_2} = c_{i_3} = c_s$. In addition, μ_s is the maximum of the c_i's such that $c_i = c_s$, so we must have $\mu_{i_1} = \mu_{i_2} = \mu_{i_3} = \mu_s$. Hence, P_s is a vertice. Following the similar reasoning, P_e is also a vertice.

We denote a set $\mathcal{J} \subseteq \mathcal{I}$ such that

$$\forall i \in \mathcal{J}, \quad P_i = P_s \quad \text{or} \quad P_i = P_e \quad \text{or} \quad c_i < \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_i - \mu_s) + c_S. \tag{7.11}$$

That is, sequence $\{P_i\}_{i\in\mathcal{J}}$ includes vertices that are either P_s, P_e or strictly below the line defined by P_s, P_e.

Remark 7.4.1. To compute $\mathcal{I}$, we can compute the index of vertices of $\mathbf{P}$ via package `scipy.spatial.ConvexHull` (version 1.11.3) [37]. From the definition given in (7.11), we can compute $\mathcal{J}$ by iterating over $\mathcal{I}$.

In case $P_s = P_e$, we can define $\mathcal{J}$ is the index set of $P_s = P_e$. ■

Before continuing, we present the following result, which is a foundation for establishing an ordering in $\{P_i\}_{i\in\mathcal{J}}$.

Lemma 7.2. *For any $i, j \in \mathcal{J}$, $\mu_i > \mu_j$ if and only if $c_i > c_j$.*

Proof. For any $i \in \mathcal{J}$, since $c_i \geq c_s, \mu_i \leq \mu_e$, we must have

$$\begin{aligned} c_i \leq \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_i - \mu_s) + c_s \leq \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_e - \mu_s) + c_s \implies c_i \leq c_e, \\ c_s \leq c_i \leq \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_i - \mu_s) + c_s \implies \mu_i \geq \mu_s. \end{aligned}$$

Now, we prove that for any $i \in \mathcal{J}$, if $c_i = c_s$ or $\mu_i = \mu_s$ then $P_i = P_s$. If $c_i = c_s$ and $\mu_i \neq \mu_s$, then from (7.11), we have

$$c_i < \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_i - \mu_s) + c_s \implies \mu_i > \mu_s,$$

but this is contrary to the definition of μ_s. If $\mu_i = \mu_s$ and $c_i \neq c_s$, then from (7.11), we have

$$c_i < \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_i - \mu_s) + c_s \implies c_i < c_s,$$

but this is contrary to the definition of c_s. Follow similar reasoning, we can prove that for any $i \in \mathcal{J}$, if $c_i = c_e$ or $\mu_i = \mu_e$ then $P_i = P_e$. According to the above observations, the claim is true if $c_i = c_s$ or $\mu_i = \mu_s$ or $c_i = c_e$ or $\mu_i = \mu_e$. Hence, from now on, we only need to prove the claim for

the case $c_s < c_i, c_j < c_e$ and $\mu_s < \mu_i, \mu_j < \mu_e$, if any.

We assume for contradiction that there exists $i, j \in \mathcal{J}$ such that $c_i > c_j$, $\mu_j \geq \mu_i$. We note that the triangle defined by P_s, P_e, P_j can be described as the intersection of three closed half-spaces as follows: $[P_s, P_j] \cap [P_e, P_j] \cap [P_s, P_e]$ where

$$[P_s, P_j] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \geq \frac{c_j - c_s}{\mu_j - \mu_s}(x - \mu_s) + c_s \right\},$$
$$[P_e, P_j] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \geq \frac{c_e - c_j}{\mu_e - \mu_j}(x - \mu_e) + c_e \right\},$$
$$[P_s, P_e] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \leq \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_e) + c_e \right\}.$$

Now, we will prove that P_i is in the interior of $[P_s, P_j]$, $[P_e, P_j]$, $[P_s, P_e]$. Considering $[P_s, P_j]$, we have that

$$\frac{c_j - c_s}{\mu_j - \mu_s}(\mu_i - \mu_s) + c_s \leq \frac{c_j - c_s}{\mu_j - \mu_s}(\mu_j - \mu_s) + c_s = c_j - c_s + c_s < c_i,$$

which implies P_i is in the interior of $[P_s, P_j]$. Considering $[P_e, P_j]$, we observe that

$$\frac{c_e - c_j}{\mu_e - \mu_j}(\mu_i - \mu_e) + c_e \leq \frac{c_e - c_j}{\mu_e - \mu_j}(\mu_j - \mu_e) + c_e = -c_e + c_j + c_e < c_i,$$

which implies P_i is in the interior of $[P_e, P_j]$. Since we only consider $c_s < c_i < c_e$, then by (7.11), P_i is in the interior of $[P_s, P_e]$. Thus, P_i is a convex combination of P_s, P_j, P_e, and this is contrary to the definition of set $\mathcal{I}$. By interchanging the roles of the x and y axes and following similar reasoning, we can prove that there does not exist any $i, j \in \mathcal{J}$ such that $c_i \geq c_j$ and $\mu_i < \mu_j$. □

From Lemma 7.2, we can define an ordering $(t_1, \ldots, t_{|\mathcal{J}|})$ of $\mathcal{J}$ such that

$$\mu_s = \mu_{t_1} < \ldots, \mu_{t_{|\mathcal{J}|}} = \mu_e \quad \text{and} \quad c_s = c_{t_1} < \ldots, c_{t_{|\mathcal{J}|}} = c_e \tag{7.12}$$

Lemma 7.3. *Suppose $|\mathcal{J}| \geq 3$. Then we have that*

$$\frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} < \frac{c_{t_{i+2}} - c_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_{i+1}}},$$

for any $i \in [|\mathcal{J}| - 2]$.

Proof. We assume for contradiction that there exists $i \in [|\mathcal{J}| - 2]$ such that

$$\frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} \geq \frac{c_{t_{i+2}} - c_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_{i+1}}}. \tag{7.13}$$

If the equality holds for (7.13), then

$$c_{t_{i+1}} = \frac{\mu_{t_{i+2}} - \mu_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_i}} c_{t_i} + \frac{\mu_{t_{i+1}} - \mu_{t_i}}{\mu_{t_{i+2}} - \mu_{t_i}} c_{t_{i+2}}.$$

We also have that

$$\mu_{t_{i+1}} = \frac{\mu_{t_{i+2}} - \mu_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_i}} \mu_{t_i} + \frac{\mu_{t_{i+1}} - \mu_{t_i}}{\mu_{t_{i+2}} - \mu_{t_i}} \mu_{t_{i+2}}.$$

Hence, $P_{t_{i+1}}$ is a convex combination of $P_{t_i}, P_{t_{i+2}}$, which is contrary to the definition of $\mathcal{I}$. Thus, (7.13) holds with strict inequality.

If (7.13) holds with strict inequality, then we show that $P_{t_{i+1}}$ is in the interior of the triangle $\mathcal{P}$ defined by P_{t_i}, $P_{t_{i+2}}$, $(\mu_{t_i}, c_{t_{i+2}})$. We note that $\mathcal{P}$ can be described as the intersection of three closed half-spaces as follows:

$$[P_{t_i}, P_{t_{i+2}}] \cap [P_{t_{i+2}}, (\mu_{t_i}, c_{t_{i+2}})] \cap [(\mu_{t_i}, c_{t_{i+2}}), P_{t_i}],$$

where

$$\begin{aligned}
[P_{t_i}, P_{t_{i+2}}] &:= \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \geq \frac{c_{t_{i+2}} - c_{t_i}}{\mu_{t_{i+2}} - \mu_{t_i}} (x - \mu_{t_i}) + c_{t_i} \right\}, \\
[P_{t_{i+2}}, (\mu_{t_i}, c_{t_{i+2}})] &:= \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \leq c_{t_{i+2}} \right\}, \\
[(\mu_{t_i}, c_{t_{i+2}}), P_{t_i}] &:= \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; x \geq \mu_{t_i} \right\}.
\end{aligned}$$

Since $c_{t_{i+1}} < c_{t_{i+2}}$ and $\mu_{t_{i+1}} > \mu_{t_i}$, $P_{t_{i+1}}$ is in the interior of $[P_{t_{i+2}}, (\mu_{t_i}, c_{t_{i+2}})] \cap [(\mu_{t_i}, c_{t_{i+2}}), P_{t_i}]$. Since (7.13) strictly holds, we observe that

$$\begin{aligned}
&\frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} > \frac{c_{t_{i+2}} - c_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_{i+1}}} \\
\iff &\frac{\mu_{t_{i+2}} - \mu_{t_{i+1}}}{\mu_{t_{i+1}} - \mu_{t_i}} > \frac{c_{t_{i+2}} - c_{t_{i+1}}}{c_{t_{i+1}} - c_{t_i}} \\
\iff &\frac{\mu_{t_{i+2}} - \mu_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} > \frac{c_{t_{i+2}} - c_{t_i}}{c_{t_{i+1}} - c_{t_i}} \\
\iff &c_{t_{i+1}} > \frac{c_{t_{i+2}} - c_{t_i}}{\mu_{t_{i+2}} - \mu_{t_i}} (\mu_{t_{i+1}} - \mu_{t_i}) + c_{t_i},
\end{aligned}$$

which implies $P_{t_{i+1}}$ is in the interior of $[P_{t_i}, P_{t_{i+2}}]$. Hence, $P_{t_{i+1}}$ is in the interior of the triangle $\mathcal{P}$. In addition, $P_{t_{i+1}}$ is also in the interior of the half-space

$$[P_s, P_e] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \le \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s \right\}.$$

Hence, $P_{t_{i+1}}$ is in the interior of $[P_e, P_s] \cap \mathcal{P}$.

If $(\mu_{t_i}, c_{t_{i+2}}) \in [P_s, P_e]$, then $P_{t_{i+2}} \neq P_e$ since otherwise we have $c_{t_{i+2}} = c_e$, and

$$c_{t_{i+2}} \le \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_{t_i} - \mu_s) + c_s \implies c_e \le \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_{t_i} - \mu_s) + c_s \implies 1 \le \frac{\mu_{t_i} - \mu_s}{\mu_e - \mu_s} \implies \mu_{t_i} \ge \mu_e,$$

which is contrary to the fact that $\mu_e \ge \mu_{t_{i+1}} > \mu_{t_i}$. Similarly, we observe that $P_{t_i} \neq P_s$ since otherwise, we have $\mu_{t_i} = \mu_s$ and

$$c_{t_{i+2}} \le \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_{t_i} - \mu_s) + c_s = \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_s - \mu_s) + c_s = c_s,$$

which is contrary to the fact that $c_s \le c_{t_{i+1}} < c_{t_{i+2}}$.

Since $P_{t_i}, P_{t_{i+2}} \in [P_s, P_e]$ by (7.11), we have $[P_s, P_e] \cap \mathcal{P} = \mathcal{P}$ and hence, $P_{t_{i+1}}$ is a convex combination of $P_{t_i}, (\mu_{t_i}, c_{t_{i+2}}), P_{t_{i+2}}$. Since $\mu_{t_i} \in [\mu_s, \mu_e]$, $(\mu_{t_i}, c_{t_{i+2}})$ is either in the interior of $[P_s, P_e] \cap [\mu_s, \mu_e] \cap [c_s, c_e]$ or on the segment defined by P_s, P_e. If $(\mu_{t_i}, c_{t_{i+2}})$ is in the interior of $[P_s, P_e]$, then we will prove that $(\mu_{t_i}, c_{t_{i+2}})$ is in the interior of the triangle defined by $P_s, P_{t_{i+2}}, P_e$.

We note that this triangle is the intersection of three closed half-spaces as follows: $[P_s, P_e] \cap [P_s, P_{t_{i+2}}] \cap [P_e, P_{t_{i+2}}]$, where

$$[P_s, P_{t_{i+2}}] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \geq \frac{c_{t_{i+2}} - c_s}{\mu_{t_{i+2}} - \mu_s}(x - \mu_s) + c_s \right\},$$
$$[P_e, P_{t_{i+2}}] := \left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y \geq \frac{c_e - c_{t_{i+2}}}{\mu_e - \mu_{t_{i+2}}}(x - \mu_e) + c_e \right\}.$$

Considering $[P_s, P_{t_{i+2}}]$, we have that

$$\frac{c_{t_{i+2}} - c_s}{\mu_{t_{i+2}} - \mu_s}(\mu_{t_i} - \mu_s) + c_s < \frac{c_{t_{i+2}} - c_s}{\mu_{t_{i+2}} - \mu_s}(\mu_{t_{i+2}} - \mu_s) + c_s = c_{t_{i+2}},$$

which implies $(\mu_{t_i}, c_{t_{i+2}})$ is in the interior of $[P_s, P_{t_{i+2}}]$. Considering $[P_e, P_{t_{i+2}}]$, we have that

$$\frac{c_e - c_{t_{i+2}}}{\mu_e - \mu_{t_{i+2}}}(\mu_{t_i} - \mu_e) + c_e < \frac{c_e - c_{t_{i+2}}}{\mu_e - \mu_{t_{i+2}}}(\mu_{t_{i+1}} - \mu_e) + c_e = c_{t_{i+2}},$$

which implies $(\mu_{t_i}, c_{t_{i+2}})$ is in the interior of $[P_e, P_{t_{i+2}}]$. Thus, $(\mu_{t_i}, c_{t_{i+2}})$ is in the interior of the triangle defined by $P_s, P_{t_{i+2}}, P_e$. If $(\mu_{t_i}, c_{t_{i+2}})$ is on the segment defined by P_s, P_e, then $(\mu_{t_i}, c_{t_{i+2}})$ is a convex combination of P_s and P_e. Therefore, we have $P_{t_{i+1}}$ is a convex combination of P_s,$P_{t_i}, P_{t_{i+2}}, P_e$, which is contrary to the definition of $\mathcal{I}$.

If $(\mu_{t_i}, c_{t_{i+2}}) \notin [P_e, P_s]$, i.e., $c_{t_{i+2}} > \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_{t_i} - \mu_s) + c_s$, then the segment defined by P_{t_i}, $(\mu_{t_i}, c_{t_{i+2}})$, and the segment defined by $P_{t_{i+2}}, (\mu_{t_i}, c_{t_{i+2}})$ intersect the boundary of $[P_s, P_e]$, i.e., the line

$$\left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; y = \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s \right\},$$

at two points

$$\tilde{P}_{t_i} := \left(\mu_{t_i}, \frac{c_e - c_s}{\mu_e - \mu_s}(\mu_{t_i} - \mu_s) + c_s \right),$$
$$\tilde{P}_{t_{i+2}} := \left(\frac{\mu_e - \mu_s}{c_e - c_s}(c_{t_{i+2}} - c_s) + \mu_s, c_{t_{i+2}} \right),$$

respectively, which are in the segment defined by P_s, P_e. Since $P_{t_i}, P_{t_{i+2}} \in [P_s, P_e]$ by (7.11),

$[P_e, P_s] \cap \mathcal{P}$ is the convex hull of $\tilde{P}_{t_i}, \tilde{P}_{t_{i+2}}, P_{t_i}, P_{t_{i+2}}$. Thus, $P_{t_{i+1}}$ is in the interior point of the convex hull of $\tilde{P}_{t_i}, \tilde{P}_{t_{i+2}}, P_{t_i}, P_{t_{i+2}}$. Since $\tilde{P}_{t_i}, \tilde{P}_{t_{i+2}}$ are two convex combinations of P_s and P_e, $P_{t_{i+1}}$ is a convex combination of $P_s, P_{t_i}, P_{t_{i+2}}, P_e$, which is in contrary to the definition of $\mathcal{I}$. □

For each $1 \leq i < |\mathcal{J}|$, we define

$$\ell_i(x) := \frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}}(x - \mu_{t_i}) + c_{t_i}, \quad \forall x \in \mathsf{R}.$$

Then, we observe the following result.

Lemma 7.4. *Suppose function $\mathcal{S}$ is de ned as follows*

$$\mathcal{S}(x) := \max_{1 \leq i < |\mathcal{J}|} \ell_i(x), \quad \forall x \in \mathsf{R}.$$

Then we have that

$$\mathcal{S}(x) = \begin{cases} \ell_1(x) & x \leq \mu_{t_1}, \\ \ell_i(x) & x \in [\mu_{t_i}, \mu_{t_{i+1}}], \quad \forall 1 \leq i < |\mathcal{J}|, \\ \ell_{|\mathcal{J}|-1}(x) & x \geq \mu_{t_{|\mathcal{J}|}}. \end{cases}$$

Proof. If $|\mathcal{J}| = 2$, the claim is true. If $|\mathcal{J}| \geq 3$, for any $i \in [|\mathcal{J}| - 2]$, by Lemma 7.3 we have

$$\ell_i(x) = \frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}}(x - \mu_{t_{i+1}}) + c_{t_{i+1}} \leq \frac{c_{t_{i+2}} - c_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_{i+1}}}(x - \mu_{t_{i+1}}) + c_{t_{i+1}} = \ell_{i+1}(x), \quad \forall x \geq \mu_{t_{i+1}},$$
$$\ell_i(x) = \frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}}(x - \mu_{t_{i+1}}) + c_{t_{i+1}} \geq \frac{c_{t_{i+2}} - c_{t_{i+1}}}{\mu_{t_{i+2}} - \mu_{t_{i+1}}}(x - \mu_{t_{i+1}}) + c_{t_{i+1}} = \ell_{i+1}(x), \quad \forall x \leq \mu_{t_{i+1}}.$$

Thus, the claim is true. □

From Lemma 7.4, for any $i \in [|\mathcal{J}| - 1]$, and $j \in \mathcal{J}$, we have $\ell_i(\mu_j) \leq \mathcal{S}(\mu_j) = c_j$. Hence, the closed half-spaces above the lines defined by ℓ_i for each $i \in [|\mathcal{J}| - 1]$ contain $\{P_i\}_{i \in \mathcal{J}}$. Additionally,

we would like to note that for any $i \in \mathcal{I} \setminus \mathcal{J}$, if any, P_i belongs to the set

$$\left\{ (x, y) \in \mathsf{R}^2 \;\middle|\; x \leq \mu_e, y \geq c_s, y \geq \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s \right\}.$$

Hence, to show that each of those half-spaces contains $\mathbf{P}$, we devote the next lemma to show a result implying that those half-spaces also contain $\{P_i\}_{i \in \mathcal{I} \setminus \mathcal{J}}$, if any.

Lemma 7.5. *Suppose $\mathcal{S}$ is de ned as in Lemma 7.4. Then we have $\mathcal{S}(x) \leq y$, for any x, y such that $x \leq \mu_e, y \geq c_s$ and*

$$y \geq \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s.$$

Proof. For $\mu_s \leq x \leq \mu_e$ and

$$y \geq \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s,$$

using Lemma 7.4 and convexity of $\mathcal{S}$, we have that

$$\begin{aligned} \mathcal{S}(x) = \mathcal{S}\left(\frac{\mu_e - x}{\mu_e - \mu_s}\mu_s + \frac{x - \mu_s}{\mu_e - \mu_s}\mu_e \right) &\leq \frac{\mu_e - x}{\mu_e - \mu_s}\mathcal{S}(\mu_s) + \frac{x - \mu_s}{\mu_e - \mu_s}\mathcal{S}(\mu_e) \\ &= c_s \frac{\mu_e - x}{\mu_e - \mu_s} + c_e \frac{x - \mu_s}{\mu_e - \mu_s} \\ &= \frac{c_e - c_s}{\mu_e - \mu_s}(x - \mu_s) + c_s \\ &\leq y. \end{aligned}$$

Since for each $1 \leq i < |\mathcal{J}|$ we have $\frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} > 0$, $\mathcal{S}$ is an increasing function. Thus, for $x \leq \mu_s$ and $y \geq c_s$, we have that

$$\mathcal{S}(x) \leq \mathcal{S}(\mu_s) = c_s \leq y.$$

Hence, we finish the proof. □

Now, we present a critical result, which gives an exact solution to problem (7.7).

Lemma 7.6. *When $r_0 \leq \mu_{t_1}$, e_{t_1} is a solution of (7.7). When there exists $1 \leq i < |\mathcal{J}|$ such that*

$\mu_{t_i} \le r_0 \le \mu_{t_{i+1}}$, $x_{t_i} e_{t_i} + x_{t_{i+1}} e_{t_{i+1}}$ *is a solution of (7.7), where*

$$x_{t_i} = \frac{\mu_{t_{i+1}} - r_0}{\mu_{t_{i+1}} - \mu_{t_i}}, \quad x_{t_{i+1}} = \frac{r_0 - \mu_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}}. \tag{7.14}$$

Proof. When $r_0 \le \mu_{t_1} = \mu_s$, given a feasible λ, μ, we have

$$-\lambda + \eta r_0 \le -\lambda + \eta \mu_{t_1} \le c_{t_1}.$$

Thus, one dual optimal solution is $\lambda^* = -c_{t_1}$ and $\eta^* = 0$ (the feasibility of this solution can be easily checked by the definition of c_s). Since x^* must minimise the Lagrangian $L(x, \lambda^*, \eta^*) = (c + \lambda^* \mathbf{1} - \eta^* \mu)^\top x - \lambda^* + r_0 \eta^*$ over $x \ge 0$, if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k = c_k - c_s = 0$ then $(x^*)_k \ge 0$ and if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k = c_k - c_s > 0$ then $(x^*)_k = 0$ for each $k \in [n]$. Therefore, x^* must satisfy

$$\sum_{k \in [n]: c_k = c_s} (x^*)_k = 1, \qquad \sum_{k \in [n]: c_k = c_s} \mu_k (x^*)_k \ge r_0.$$

To ensure the feasibility of x^*, we set $x^* = e_{t_1}$ since P_{t_1} is the point with largest x coordinate among points with y coordinate of $c_{t_1} = c_s$.

When there is $1 \le i < |\mathcal{J}|$ such that $\mu_{t_i} \le r_0 \le \mu_{t_{i+1}}$, we have that for any feasible λ, η

$$\begin{aligned}
-\lambda + \eta r_0 &= \frac{\mu_{t_{i+1}} - r_0}{\mu_{t_{i+1}} - \mu_{t_i}} (-\lambda + \eta \mu_{t_i}) + \frac{r_0 - \mu_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} (-\lambda + \eta \mu_{t_{i+1}}) \\
&\le \frac{\mu_{t_{i+1}} - r_0}{\mu_{t_{i+1}} - \mu_{t_i}} c_{t_i} + \frac{r_0 - \mu_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} c_{t_{i+1}} \\
&= \frac{c_{t_{i+1}} - c_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} (r_0 - \mu_{t_i}) + c_{t_i} \\
&= \mathcal{S}(r_0).
\end{aligned}$$

In fact, this upper bound is attainable with an optimal dual variable (λ^*, η^*) satisfies $-\lambda^* + \eta^* x = \ell_i(x)$ for any $x \in \mathsf{R}$. We will prove that (λ^*, η^*) is feasible for problem (7.8). For any $j \in \mathcal{I} \setminus \mathcal{J}$, we have $-\lambda^* + \eta^* \mu_j = \ell_i(x_j) \le \mathcal{S}(\mu_j) \le c_j$ by Lemma 7.5 and for any $j \in \mathcal{J}$, we have $-\lambda^* + \eta^* \mu_j \le \mathcal{S}(\mu_i) = c_i$. This implies λ^*, η^* are feasible. By definition and Lemma 7.4, $\mathcal{S}(x) = -\lambda^* + \eta^* x$ for

Algorithm 6: Linear oracle over a sliced probability simplex

Data: $c \in \mathsf{R}^n$.
Result: Solution of problem (7.7).
Compute P_s and P_e as defined in (7.9) and (7.10);
Compute set $\mathcal{J}$ as defined in (7.11);
Compute integers $t_1, \ldots, t_{|\mathcal{J}|}$ satisfying (7.12);
if $r_0 \leq \mu_{t_1}$ **then**
  Return e_{t_1}.
else
  Compute $i, x_{t_i}, x_{t_{i+1}}$ satisfying (7.14);
  Return $x_{t_i} e_{t_i} + x_{t_{i+1}} e_{t_{i+1}}$

any $x \in [\mu_{t_i}, \mu_{t_{i+1}}]$, which implies the upper bound is attainable.

Since x^* must minimise the Lagrangian $L(x, \lambda^*, \eta^*) = (c + \lambda^* \mathbf{1} - \eta^* \mu)^\top x - \lambda^* + r_0 \eta^*$ over $x \geq 0$, if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k = 0$ then $(x^*)_k \geq 0$, and if $(c + \mathbf{1}\lambda^* - \mu\eta^*)_k > 0$ then $(x^*)_k = 0$ for each $k \in [n]$. Therefore, x^* must satisfy

$$\sum_{k \in [n]: \ell_i(\mu_k) = c_k} (x^*)_k = 1, \qquad \sum_{k \in [n]: \ell_i(\mu_k) = c_k} \mu_k (x^*)_k \geq r_0.$$

To ensure the feasibility of x^*, we set $x^* := x_{t_i} e_{t_i} + x_{t_{i+1}} e_{t_{i+1}}$ where

$$\begin{cases} \mu_{t_i} x_{t_i} + \mu_{t_{i+1}} x_{t_{i+1}} = r_0 \\ x_{t_i} + x_{t_{i+1}} = 1 \end{cases} \iff \begin{cases} x_{t_i} = \frac{\mu_{t_{i+1}} - r_0}{\mu_{t_{i+1}} - \mu_{t_i}} \geq 0 \\ x_{t_{i+1}} = \frac{r_0 - \mu_{t_i}}{\mu_{t_{i+1}} - \mu_{t_i}} \geq 0. \end{cases} \tag{7.15}$$

□

Now, we have enough tools to construct a finite algorithm to find a solution for problem (7.7) as provided in Algorithm 6.

7.4.2 Linear minimisation over a nuclear norm ball

In this subsection, we provide a scheme to tackle the subproblem of the `IR-CG` and `PD-CG` methods in solving problem (7.2). The corresponding linear subproblem is as follows:

$$\begin{aligned} \min_{V} \quad & \text{Trace}(C^\top V) \\ \text{s.t.} \quad & \|V\|_* \leq \delta \end{aligned} \tag{7.16}$$

Let $u^{(1)}, v^{(1)}$ are left and right leading singular vectors of C. Jaggi [23, Section 4.2] provided an optimal solution to (7.16) which is $V^* := -\delta u^{(1)} \left(v^{(1)}\right)^\top$. To compute this solution, we compute a leading eigenvalue $v^{(1)}$ with length 1 and largest eigenvalue $\sigma^2_{\max}(C)$ of $C^\top C$ with the Lanczos process [20, Section 10.1] via package `scipy.linalg.eigh` (version 1.11.3) [37]. Vector $u^{(1)}$ is computed as $-\frac{1}{\sigma_{\max}(C)} C v^{(1)}$.

7.4.3 Linear minimisation over a sliced nuclear norm ball

In this subsection, we provide a scheme to tackle the subproblem of the `SL-CG` and `CG-BiO` methods in solving problem (7.2). The problem is as follows:

$$\begin{aligned} \min_{V} \quad & \text{Trace}(C^\top V) \\ \text{s.t.} \quad & \|V\|_* \leq \delta \\ & \text{Trace}(A^\top V) \leq b. \end{aligned} \tag{7.17}$$

Since the size of V is large, it is impractical to use off-the-shelf conic optimisation solvers for problem (7.17). Therefore we provide an efficient custom algorithm. We note that we will consider the case when (7.17) is feasible, since they are generated from outer approximations of X_{opt}, which we assume to be non-empty. This implies that

$$b \geq \min_{\|V\|_* \leq \delta} \text{Trace}(A^\top V) = -\delta \sigma_{\max}(A).$$

Before continuing, we have the following observation.

Lemma 7.7. *If $b > -\delta\sigma_{\max}(A)$, then Slater's condition holds for problem (7.17). If $A \neq 0$, then the reverse is true.*

Proof. If $b > -\delta\sigma_{\max}(A) = \min_{\|V\|_* \leq \delta} \mathrm{Trace}(A^\top V)$, we let V^* be a minimiser of $\mathrm{Trace}(A^\top V)$ over $\|V\|_* \leq \delta$ such that $\|V^*\|_* = \delta$ (this minimiser always exists according to Section 7.4.2). Since $\lim_{U \to V^*} \mathrm{Trace}(A^\top U) = \mathrm{Trace}(A^\top V^*) < b$, there exists a sufficiently small $\epsilon > 0$ such that for any U such that $\|U - V^*\|_* < \epsilon$, we have $\mathrm{Trace}(A^\top U) < b$. Since V^* is on the boundary of the nuclear norm ball, there must exist an U satisfies $\|U - V^*\|_* < \epsilon$ and is in the interior of the nuclear norm ball, i.e., $\|U\|_* < \delta$. Thus, Slater's condition holds for problem (7.17).

Now, we assume $A \neq 0$. If Slater's condition holds for problem (7.17), there exists U such that

$$\|U\|_* < \delta, \quad \mathrm{Trace}(A^\top U) \leq b.$$

Since $A \neq 0$ and $\|U\|_* < \delta$, we have that

$$b \geq \mathrm{Trace}(A^\top U) \geq -\sigma_{\max}(A)\|U\|_* > -\delta\sigma_{\max}(A),$$

where in the second inequality, we use the fact that the spectral norm is the dual norm of the nuclear norm. □

First, we consider the case in which Slater's condition does not hold. In this case, by Lemma 7.7, we have

$$\min_{\|V\|_* \leq \delta} \mathrm{Trace}(A^\top V) = b,$$

and

$$\{V \in \mathbf{R}^{n \times p} \mid \mathrm{Trace}(A^T V) \geq b, \|V\|_* \leq \delta\} = \arg\min_{\|U\|_* \leq \delta} \mathrm{Trace}(A^\top U).$$

Therefore, a solution to (7.17) in this case is

$$V^* \in \arg\min_V \left\{ \mathrm{Trace}(C^\top V) \;\middle|\; V \in \arg\min_{\|U\|_* \leq \delta} \mathrm{Trace}(A^\top U) \right\}.$$

When Slater's condition holds for problem (7.17), we consider the Lagrangian

$$L(V,\lambda,\mu) = \text{Trace}((C+\lambda A)^\top V) - \lambda b + \mu\left(\|V\|_* - \delta\right), \quad V \in \mathsf{R}^{n\times p}, \lambda, \mu \geq 0,$$

then the dual function is

$$\mathcal{D}(\lambda,\mu) = \inf_{V\in\mathsf{R}^{n\times p}} L(V,\lambda,\mu) = \begin{cases} -b\lambda - \delta\mu & \sigma_{\max}(C+\lambda A) \leq \mu \\ -\infty & \text{otherwise.} \end{cases}$$

Thus, the dual problem is

$$\begin{aligned} \max_{\lambda,\mu\geq 0} \quad & -b\lambda - \delta\mu \\ \text{s.t.} \quad & \sigma_{\max}(C+\lambda A) \leq \mu, \end{aligned} \tag{7.18}$$

which is equivalent to

$$\min_{\lambda\geq 0} \quad \delta\sigma_{\max}(C+\lambda A) + b\lambda. \tag{7.19}$$

We also note that in case Slater's condition holds, we have $b > -\delta\sigma_{\max}(A)$ if $A \neq 0$ by Lemma 7.7. At any optimal solution λ^* of problem (7.19), the objective must not be greater than that at $\lambda = 0$. Therefore, we can obtain an upper bound for λ^* as follows:

$$\begin{aligned} \delta\sigma_{\max}(C) &\geq \delta\sigma_{\max}(C+\lambda^* A) + b\lambda^* \\ &\geq \delta\left(\sigma_{\max}(\lambda^* A) - \sigma_{\max}(-C)\right) + b\lambda^*, \\ \implies 2\delta\sigma_{\max}(C) &\geq \left(b + \delta\sigma_{\max}(A)\right)\lambda^*, \\ \implies \frac{2\delta\sigma_{\max}(C)}{b+\delta\sigma_{\max}(A)} &\geq \lambda^*, \end{aligned}$$

where we use the triangle inequality for the spectral norm in the second inequality. Given an optimal dual variable λ^* by solving (7.19), the optimal solution of problem (7.17) is also the solution of minimising $\text{Trace}((C+\lambda^* A)^\top V)$ over the nuclear ball. If $A = 0$, we observe that $\lambda^* = 0$ minimises problem (7.19) since $b \geq -\delta\sigma_{\max}(A) = 0$.

Remark 7.4.2. To solve problem (7.19), if $A \neq 0$, we compute λ^* by conducting line search of function $\delta\sigma_{\max}(C + \lambda A) + b\lambda$ over the interval

$$\left[0, \frac{2\delta\sigma_{\max}(C)}{b + \delta\sigma_{\max}(A)}\right],$$

with the bounded Brent method [16] via package `scipy.optimize.minimize-scalar` (version 1.11.3) [37]. If $A = 0$, we set $\lambda^* := 0$. ■

Given λ^*, we need to ensure the solution V^* we get from minimising $\mathrm{Trace}((C + \lambda^* A)^\top V)$ over the nuclear ball satisfies the linear inequality constraint $\mathrm{Trace}(A^\top V^*) \leq b$. Hence, we can compute such a solution as follows:

$$V^* \in \arg\min_V \left\{ \mathrm{Trace}(A^\top V) \;\middle|\; V \in \arg\min_{\|U\|_* \leq \delta} \mathrm{Trace}((C + \lambda^* A)^\top U) \right\}.$$

Therefore, both cases require us to solve bilevel linear problems over the nuclear norm ball. To do this, we need the following results.

Lemma 7.8. *Given a matrix $P \in \mathsf{R}^{n\times p}$, let E_1 be the eigenspace associated with the leading eigenvalue of matrix $P^\top P$ and*

$$\mathcal{E}_1 := \left\{ \frac{-\delta}{\sigma_{\max}(P)} P v v^\top \;\middle|\; v \in E_1, \|v\|_2 = 1 \right\}.$$

Then

$$\arg\min_{\|Z\|_* \leq \delta} \mathrm{Trace}(P^\top Z) = \mathrm{Conv}(\mathcal{E}_1).$$

Proof. Note that the optimal value is $-\delta\sigma_{\max}(P)$. First, we prove that $\arg\min_{\|Z\|_* \leq \delta} \mathrm{Trace}(P^\top Z) \supseteq \mathrm{Conv}(\mathcal{E}_1)$ holds. Given $X \in \mathcal{E}_1$, there exists $v \in E_1, \|v\|_2 = 1$, such that

$$X = \frac{-\delta}{\sigma_{\max}(P)} P v v^\top.$$

Observe that $X^\top X = \frac{\delta^2}{\sigma^2_{\max}(P)} v v^\top P^\top P v v^\top = \delta^2 v v^\top$. Note that there is only one non-zero eigen-

value δ^2 of $X^\top X$ with eigenvector v. Any vector that is orthogonal to v has eigenvalue 0. Therefore $(X^\top X)^{1/2} = \delta v v^\top$, and $\|X\|_* = \mathrm{Tr}((X^\top X)^{1/2}) = \delta$, thus X is feasible. This implies that any point in $\mathrm{Conv}(\mathcal{E}_1)$ is also feasible since the nuclear norm ball is a convex set. We have that

$$\mathrm{Trace}(P^\top X) = \frac{-\delta}{\sigma_{\max}(P)} \mathrm{Trace}(v^\top P^\top P v) = -\delta \sigma_{\max}(P),$$

thus X is optimal, which implies $\mathcal{E}_1 \subseteq \arg\min_{\|Z\|_* \leq \delta} \mathrm{Trace}(P^\top Z)$. Taking convex hulls of both sets, we obtain the required result.

Now, we prove $\arg\min_{\|Z\|_* \leq \delta} \mathrm{Trace}(P^\top Z) \subseteq \mathrm{Conv}(\mathcal{E}_1)$ holds. Given a feasible X, let a singular value decomposition of X be

$$X = \sum_{i \in [\min\{n,p\}]} \sigma_i u_i v_i^\top,$$

where $\sigma_1 \geq \cdots \geq \sigma_{\min\{n,p\}} \geq 0$. Since $\|X\|_* \leq \delta$, we have $\sum_{i \in [\min\{n,p\}]} \sigma_i \leq \delta$.

By using the Cauchy-Schwarz inequality and the fact that $\{u_i\}_{i \in [\min\{n,p\}]}$ and $\{v_i\}_{i \in [\min\{n,p\}]}$ are two sets orthonormal vectors, we have that

$$\begin{aligned}
\mathrm{Trace}(P^\top X) &= \sum_{i \in [\min\{n,p\}]} \sigma_i v_i^\top P^\top u_i \\
&\geq \sum_{i \in [\min\{n,p\}]} \sigma_i \left(-\|u_i\|_2 \|P v_i\|_2\right) \\
&= -\sum_{i \in [\min\{n,p\}]} \sigma_i \|P v_i\|_2 \\
&\geq -\sigma_{\max}(P) \sum_{i \in [\min\{n,p\}]} \sigma_i \\
&\geq -\delta \sigma_{\max}(P)
\end{aligned}$$

The first inequality becomes equality if and only if given $i \in [\min\{n,p\}]$, we have that $\sigma_i = 0$ or $P v_i = k_i u_i$ for some $k_i \leq 0$. The second inequality becomes equality if and only if given

$i \in [\min\{n,p\}]$, $\sigma_i = 0$ or v_i is a leading eigenvector of $P^\top P$. Hence, in order to have

$$X \in \arg\min_{\|Z\|_* \leq \delta} \mathrm{Trace}(P^\top Z),$$

given $i \in [\min\{n,p\}]$, we have that $\sigma_i = 0$ or v_i is a leading eigenvector of $P^\top P$. In summary, in order for X to be optimal (i.e., all inequalities above hold with equality), we need v_i to be a leading eigenvector of $P^\top P$, $Pv_i = -\sigma_{\max}(P)u_i$ whenever $\sigma_i \neq 0$, and $\sum_{i\in[\min\{n,p\}]} \sigma_i = \delta$. Furthermore, note that if we define $u_i = -\frac{1}{\sigma_{\max}(P)} Pv_i$ whenever $v_i \in E_1$, then $u_i^\top u_j = \frac{1}{\sigma_{\max}(P)^2} v_i^\top P^\top P v_j = v_i^\top v_j = 0$ for any other $j \in [\min\{n,p\}]$. On the other hand, $u_i^\top u_i = v_i^\top v_i = 1$. Therefore $\{u_i\}_{i\in[\min\{n,p\}]}$ defined in this way is also an orthonormal set of vectors. Since $\{v_i\}_{i\in[\min\{n,p\}]}$ is a set of orthonormal vectors in E_1, we must have $\sigma_i = 0, \forall i > \dim(E_1)$. Thus, if X is a minimiser of $\mathrm{Trace}(P^\top Z)$ over $\|Z\|_* \leq \delta$ then

$$X = \sum_{i\in[\dim(E_1)]} \sigma_i u_i v_i^\top = -\frac{1}{\sigma_{\max}(P)} \sum_{i\in[\dim(E_1)]} \sigma_i P v_i v_i^\top,$$

where $\{v_1, \ldots, v_{\dim(E_1)}\}$ is an orthonormal basis of E_1 and $\sigma_i \geq 0, \sum_{i\in[\dim(E_1)]} \sigma_i = \delta$. Therefore, $X \in \mathrm{Conv}(\mathcal{E}_1)$ as required. □

Lemma 7.9. *Given $Q \in \mathsf{R}^{n\times p}$, let P, E_1 be de ned as in Lemma 7.8, $R \in \mathsf{R}^{p\times\dim(E_1)}$ be a matrix whose columns form an orthonormal basis of E_1, $S \in \mathsf{R}^{\dim(E_1)\times\dim(E_1)}$ be a symmetric matrix de ned as follows:*

$$S := R^\top \left(\frac{Q^\top P + P^\top Q}{2}\right) R,$$

and $s_1 \in \mathsf{R}^{d_1}$ be a leading eigenvector of S with length 1. Then we have that

$$-\frac{\delta}{\sigma_{\max}(P)} P(Rs_1)(Rs_1)^\top \in \arg\min_X \left\{ \mathrm{Trace}(Q^\top X) \;\middle|\; X \in \arg\min_{\|Z\|_*\leq\delta} \mathrm{Trace}(P^\top Z) \right\}.$$

Proof. By Lemma 7.8, we have that

$$\arg\min_X \left\{ \mathrm{Trace}(Q^\top X) \;\middle|\; X \in \arg\min_{\|Z\|_*\leq\delta} \mathrm{Trace}(P^\top Z) \right\} \supseteq \arg\min_{X\in\mathcal{E}_1} \mathrm{Trace}(Q^\top X).$$

Let $X \in \mathcal{E}_1$, there exists $v \in E_1, \|v\|_2 = 1$ such that

$$X = \frac{-\delta}{\sigma_{\max}(P)} P v v^\top.$$

Then we have

$$\begin{aligned} \text{Trace}(Q^\top X) &= -\frac{-\delta}{\sigma_{\max}(P)} v^\top (Q^\top P) v \\ &\geq -\frac{\delta}{\sigma_{\max}(P)} \max_{\|u\|_2=1, u\in E_1} u^\top (Q^\top P) u. \end{aligned}$$

Such lower bound can be obtained when we set

$$v \in \arg\max_{\|u\|_2=1, u\in E_1} u^\top (Q^\top P) u.$$

Given $\|u\|_2 = 1, u \in E_1$, we have $u = Rs$, where $s \in \mathsf{R}^{\dim(E_1)}$ is a vector with length 1. Hence, we have

$$\arg\max_{\|u\|_2=1, u\in E_1} u^\top (Q^\top P) u = \arg\max_{\|s\|_2=1} s^\top (R^\top Q^\top P R) s = \arg\max_{\|s\|_2=1} s^\top S s,$$

where in the last equality, we use the fact that $s^\top (R^\top Q^\top P R) s = s^\top (R^\top P^\top Q R) s$. Thus, we can choose $v = R s_1$. □

Remark 7.4.3. To compute R as defined in Lemma 7.9, we used package `scipy.linalg.eigh` (version 1.11.3) [37] to compute leading eigenvalue of matrix $P^\top P$ and the associated eigenvectors whose lengths are 1. ■

Now, we have enough tools to address problem (7.17), which are shown in Algorithms 7 to 8.

7.4.4 Projection onto a nuclear norm ball

In this subsection, we provide a scheme to compute the projection onto the base domain required by the `IR-PG` and `Bi-SG` methods in solving (7.1). Given a matrix X, let its singular value decom-

Algorithm 7: Bilevel linear oracle over a nuclear norm ball - NB-BLO

Data: $P, Q \in \mathsf{R}^{n\times p}, \delta > 0$.

Result: $V^* \in \arg\min_V \left\{ \mathrm{Trace}(Q^\top V) \;\middle|\; V \in \arg\min_{\|U\|_* \le \delta} \mathrm{Trace}(P^\top U) \right\}$.

Compute

$$R \quad \text{as defined in Lemma 7.9}$$
$$S := \frac{1}{2} R^\top (Q^\top P + P^\top Q) R$$
$$s_1 \in \arg\max_{\|s\|_2 = 1} s^\top S s$$
$$V^* := \frac{-\delta}{\sigma_{\max}(P)} P(Rs_1)(Rs_1)^\top.$$

Algorithm 8: Linear oracle over a sliced nuclear norm ball

Data: $C \in \mathsf{R}^{n\times p}, A \in \mathsf{R}^{n\times p}, b \in \mathsf{R}, \delta > 0$.

Result: V^*- a solution of (7.17).

if $b = -\delta\sigma_{\max}(A)$ **then**

Compute

$$V^* := \text{NB-BLO}(A, C, \delta).$$

else

Compute

$$\lambda^* \in \arg\min_{\lambda \ge 0} \{\delta\sigma_{\max}(C + \lambda A) + b\lambda\}$$
$$V^* := \text{NB-BLO}(C + \lambda^* A, A, \delta).$$

position be as follows:

$$X = \sum_{i\in[k]} \sigma_i u_i v_i^\top,$$

in which $k = \min\{n, p\}$ and $\sigma_1 \ge \cdots \ge \sigma_k \ge 0$. Let $s \in \mathsf{R}^k$ be the Euclidean projection of $(\sigma_1, \ldots, \sigma_k)$ onto the simplex $S_\delta := \{x \in \mathsf{R}^k \mid \mathbf{1}^\top x \le \delta, x \ge 0\}$. Beck [3, Section 7.3.2] provides the following expression for the Frobenius norm projection of X onto the nuclear ball $\{V \in \mathsf{R}^{n\times p} \mid \|V\|_* \le \delta\}$ is

$$\sum_{i\in[k]} s_i u_i v_i^\top.$$

Now we discuss how the projection onto S_δ can be computed. Given $x \in \mathsf{R}^k$, we have that the projection onto S_δ can be computed as follows:

$$\mathrm{Proj}_{S_\delta}(x) = \delta \,\mathrm{Proj}_{S_1}\left(\frac{x}{\delta}\right),$$

where the projection onto the probability simplex S_1 can be efficiently computed via [9, Algorithm 1].

7.4.5 Linear minimisation over a sliced box

In this subsection, we provide a solution to the linear minimisation subproblem required for SL-CG in solving (7.5), which is formulated as follows:

$$\begin{aligned} \min_{x \in \mathsf{R}^n} \quad & c^\top x \\ \text{s.t.} \quad & a^\top x \le d, \quad 0 \le x \le r\mathbf{1}, \end{aligned} \tag{7.20}$$

where $a, c \in \mathsf{R}^n, d \in \mathsf{R}, r > 0$ are given. Before solving problem (7.20), we need to determine when this problem is infeasible. This issue can be addressed by the following result.

Lemma 7.10. *Given $p \in \mathsf{R}^n$ and $r > 0$, let $\Gamma_1 : \mathsf{R} \quad \mathsf{R}$ be a set-valued mapping such that*

$$\Gamma_1(x) := \begin{cases} \{0\} & x > 0 \\ \{r\} & x < 0 \\ [0, r] & x = 0. \end{cases}$$

Then we have that

$$X^{(p)}_{\mathrm{opt}} := \operatorname*{arg\,min}_{0 \le x \le r\mathbf{1}} p^\top x = \Gamma_1(p_1) \times \cdots \times \Gamma_1(p_n), \tag{7.21}$$

and the optimal value is $\sum_{i \in [n]} r \min\{0, p_i\}$.

Proof. Given x such that $0 \leq x \leq r\mathbf{1}$, we have that

$$p^\top x = \sum_{i\in[n]} p_i x_i.$$

If $p_i > 0$ then $p_i x_i \geq 0$ and if $p_i < 0$ then $p_i x_i \geq p_i r$. Hence, we have $p_i x_i \geq r \min\{0, p_i\}$ with the equality happens if and only if $p_i = 0$ or $x_i = 0, p_i > 0$ or $x_i = r, p_i < 0$. Therefore, we obtain

$$p^\top x \geq \sum_{i\in[n]} r \min\{0, p_i\},$$

and the equality holds if and only if for any $i \in [n]$, $p_i = 0, 0 \leq x_i \leq r$ or $x_i = 0, p_i > 0$ or $x_i = r, p_i < 0$. Thus, (7.21) is true. □

By Lemma 7.10, problem (7.20) is infeasible if and only if

$$d < \min_{0\leq x\leq r\mathbf{1}} a^\top x = r \sum_{i\in[n]} \min\{a_i, 0\}. \tag{7.22}$$

When (7.20) is feasible, Slater's condition holds. Thus, we consider the Lagrangian

$$L(x, \lambda) := c^\top x + \lambda(a^\top x - d) = (c + \lambda a)^\top x - d\lambda, \quad 0 \leq x \leq r\mathbf{1}, \lambda \geq 0.$$

Therefore, the dual function can be written as

$$\mathcal{D}(\lambda) = \min_{0\leq x\leq r\mathbf{1}} \{(c + \lambda a)^\top x - d\lambda\} = r \sum_{i\in[n]} \min\{c_i + \lambda a_i, 0\} - d\lambda, \quad \forall \lambda \geq 0.$$

Since $\mathcal{D}(\lambda)$ is a piecewise linear function with respect to $\lambda \geq 0$, its attainable maximum can be obtained at either $\lambda = 0$ or some positive knot(s) of $\mathcal{D}(\lambda)$. Thus, a maximiser of $\mathcal{D}(\lambda)$ over $\lambda \geq 0$ should be in the set

$$\left\{ -\frac{c_i}{a_i} \;\middle|\; i \in [n], a_i \neq 0, -\frac{c_i}{a_i} > 0 \right\} \cup \{0\}.$$

Given a optimal dual variable $\lambda^* \geq 0$, any solution of (7.20) is also a solution of minimising

$(c + \lambda^* a)^\top x$ over $0 \leq x \leq r\mathbf{1}$. Since the latter problem might have multiple solutions, some of which may be infeasible for (7.20), it is sufficient to compute a solution of (7.20) by solving

$$\begin{aligned} \min_{x \in \mathsf{R}^n} \quad & a^\top x \\ \text{s.t.} \quad & x \in \arg\min_{0 \leq z \leq r\mathbf{1}} \left\{ (c + \lambda^* a)^\top z \right\}. \end{aligned} \tag{7.23}$$

To solve (7.23), we need the following lemma.

Lemma 7.11. *Given $q \in \mathsf{R}^n$, let $p \in \mathsf{R}^n, r > 0, X^{(p)}_{\text{opt}}, \Gamma_1 : \mathsf{R} \quad \mathsf{R}$ be de ned as in Lemma 7.10, and $\Gamma_2 : \mathsf{R}^2 \quad \mathsf{R}$ be a set-valued mapping such that*

$$\Gamma_2(x, y) := \begin{cases} \{0\} & x > 0 \text{ or } x = 0, y > 0 \\ \{r\} & x < 0 \text{ or } x = 0, y < 0 \\ [0, r] & x = 0, y = 0. \end{cases}$$

Then we have that

$$\arg\min_{x \in X^{(p)}_{\text{opt}}} q^\top x = \Gamma_2(p_1, q_1) \times \cdots \times \Gamma_2(p_n, q_n). \tag{7.24}$$

Proof. Given $x \in X^{(p)}_{\text{opt}}$, by Lemma 7.10, we have that $x_i = 0$ if $p_i > 0$, $x_i = r$ if $p_i < 0$, and $x_i \in [0, r]$ if $p_i = 0$ for each $i \in [n]$. We define $\mathcal{I} = \{i \in [n] \mid p_i = 0\}$ then minimising $q^\top x$ over $x \in X^{(p)}_{\text{opt}}$ is equivalent to solving

$$\begin{aligned} \min_{\{x_i\}_{i \in \mathcal{I}}} \quad & \sum_{i \in \mathcal{I}} q_i x_i \\ \text{s.t.} \quad & x_i \in [0, r], \quad \forall i \in \mathcal{I}. \end{aligned} \tag{7.25}$$

We apply Lemma 7.10 for problem (7.25) to have that x^* is a minimiser of $q^\top x$ over $x \in X^{(p)}_{\text{opt}}$ if and only if $(x^*)_i \in \Gamma_1(p_i)$ for each $i \notin \mathcal{I}$, and $(x^*)_i \in \Gamma_1(q_i)$ for each $i \in \mathcal{I}$. We notice that this is equivalent to that $(x^*)_i \in \Gamma_1(p_i, q_i)$, for each $i \in [n]$. □

From Lemma 7.11, a solution of (7.20), if the problem is feasible, can be computed via Algo-

Algorithm 9: Linear oracle over a sliced box

Data: $a, c \in \mathsf{R}^n$, $d \in \mathsf{R}$, $r > 0$ unsatisfying (7.22).
Result: x^* - a solution of (7.20)
Compute

$$\mathcal{C} := \left\{ -\frac{c_i}{a_i} \;\middle|\; i \in [n], a_i \neq 0, -\frac{c_i}{a_i} > 0 \right\} \cup \{0\}$$

$$\lambda^* \in \arg\max_{\lambda \in \mathcal{C}} \left\{ r \sum_{i \in [n]} \min\{c_i + \lambda a_i, 0\} - d\lambda \right\}$$

$$(x^*)_i := \begin{cases} 0 & , c_i + \lambda^* a_i > 0 \text{ or } c_i + \lambda^* a_i = 0, a_i \geq 0 \\ r & , c_i + \lambda^* a_i < 0 \text{ or } c_i + \lambda^* a_i = 0, a_i < 0 \end{cases}, \quad \forall i \in [n].$$

rithm 9, as shown below.

Chapter 8

Concluding remarks

Technological innovation and development have led to efficiency gains in all aspects of decision-making science; however, many practical problems are still tasked with determining solutions to bilevel (more generally, hierarchical) optimisation problems. Along with the rise of the interdependent economy, where more capital resources are being shared amongst multiple agents with competing interests, businesses in these markets must model multiple objectives with clear priorities to ensure efficiency gains over time to maintain a competitive advantage in their market. Although bilevel optimisation problems have been independently studied in the existing literature, a simple first-order projection-free scheme to solve such problems has not been considered.

This thesis addresses the smooth convex bilevel optimisation problems numerically via three projection-free methods. We assume that the base domain is convex and closed, both objective functions are smooth convex on an open subset of the base domain, and the pointwise maximum of the inner- and outer- objectives is coercive over the base domain. We solve the bilevel problem via three approaches: inner-level optimal set approximation, regularisation and primal-dual-type update. Other than the assumptions mentioned above, we also require the outer objective to be bounded from below over the base domain to ensure the convergence of the regularisation- and primal-dual-based methods. A critical difference in assumptions between our methods and other projection-free algorithms is the boundedness of the base domain. To account for such relaxation,

Method	SL-CG	IR-CG	PD-CG	
			without Assumption 4	with Assumption 4
Inner-level rate	$O\left(d_T^2/T\right)$	$O\left(1/T^p\right)$	$O\left(1/T^{(1-p)/2}\right)$	$O\left(\max\left\{1/T^{1-p}, d_T^2/T^{1/2}\right\}\right)$
Outer-level rate	$O\left(d_T^2/T\right)$	$O\left(d_T^2/T^{1-p}\right)$	$O\left(\max\left\{1/T^{1-p}, d_T^4/T^p\right\}\right)$	
Diameters' growth	$o\left(t^{1/2}\right)$	$o\left(t^{(1-p)/2}\right)$	$o\left(t^{p/4}\right)$	
Reference	Theorem 4.4	Theorem 5.10	Theorem 6.9	Theorem 6.10

Table 8.1: Comparison of convergence rates of the SL-CG, IR-CG and PD-CG methods.

we introduce a sequence of coverings with the intuition of enforcing the boundedness for each iteration but still maintaining the unboundedness in the long term. Furthermore, to measure of the super-optimality of the methods, we adopt a global error condition on the inner-level objective over the base domain.

A summary of the proposed algorithms' convergence rates is shown in Table 8.1. While the SL-CG method has the fastest theoretical convergence rates, the corresponding linear minimisation oracle tends to be more complicated due to the linear inequality constraint from approximating the g_{opt}-sublevel set and, hence, may not perform better than IR-CG and PD-CG given a time limit as shown in the experiments discussed in Chapter 7. The IR-CG method is the projection-free method that can balance the simplicity in the implementation when only minimising linear objective over the base domain is required and the quality of convergence rates.

A natural extension of the methods studied in this thesis is to explore the stochastic version of SL-CG, IR-CG, and PD-CG, which may be helpful in large-scale data fitting problems in which exact gradient computation should be expensive. Another possible research direction is considering a non-smooth version of the proposed methods. Additionally, for unbounded base domains, it would be interesting to provide a theory on designing the coverings $\{B_t\}_{t\geq 0}$ given the base domain X.

Bibliography

[1] M. Amini and F. Yousefian. An iterative regularized incremental projected subgradient method for a class of bilevel optimization problems. In *2019 American Control Conference (ACC)*, pages 4069–4074, 2019.

[2] M. ApS. *MOSEK Optimizer API for Python 10.1.10*, 2019. URL https://docs.mosek.com/latest/pythonapi/index.html.

[3] A. Beck. *First-order methods in optimization*. MOS-SIAM series on optimization ; 25. Society for Industrial and Applied Mathematics, 2017. ISBN 9781611974997.

[4] 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.

[5] J. V. Burke and M. C. Ferris. Weak sharp minima in mathematical programming. *SIAM Journal on Control and Optimization*, 31(5):1340–1359, 1993.

[6] A. Cabot. Proximal point algorithm controlled by a slowly vanishing term: Applications to hierarchical minimization. *SIAM Journal on Optimization*, 15:555–572, 2005.

[7] J. Cao, R. Jiang, N. Abolfazli, E. Y. Hamedani, and A. Mokhtari. Projection-free methods for stochastic simple bilevel optimization with convex lower-level problem. Technical report, Department of Electrical and Computer Engineering, The University of Texas at Austin, 2023. URL https://arxiv.org/pdf/2308.07536.pdf.

[8] A. D. R. Choudary. *Real Analysis on Intervals*. Springer India, New Delhi, 1st ed. 2014. edition, 2014. ISBN 81-322-2148-6.

[9] L. Condat. Fast projection onto the simplex and the l1 ball. *Mathematical Programming*, 158 (1-2):575–585, 2016.

[10] A. Dhara and J. Dutta. *Optimality Conditions in Convex Optimization: A Finite-Dimensional View*. CRC Press, 1 edition, 2011. ISBN 1439868239.

[11] S. Diamond and S. Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[12] L. Doron and S. Shtern. Methodology and first-order algorithms for solving nonsmooth and non-strongly convex bilevel optimization problems. *Mathematical Programming*, 2022.

[13] D. Drusvyatskiy and A. S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *Mathematics of operations research*, 43(3):919–948, 2018.

[14] J. Dutta and T. Pandit. Algorithms for simple bilevel programming. In *Bilevel Optimization*, Springer Optimization and Its Applications, pages 253–291. Springer International Publishing, 2020. ISBN 9783030521189.

[15] M. Fazel. Matrix rank minimization with applications. Technical report, The Department of Electrical Engineering, Standford University, 2002. URL https://faculty.washington.edu/mfazel/thesis-final.pdf. PhD thesis.

[16] G. Forsythe, M. Malcolm, and C. Moler. Computer methods for mathematical computations. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 59(2):141–142, 1979.

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

[18] S. Franke, P. Mehlitz, and M. Pilecka. Optimality conditions for the simple convex bilevel programming problem in banach spaces. *Optimization*, 67(2):237–268, 2018.

[19] M. P. Friedlander and P. Tseng. Exact regularization of convex programs. *SIAM Journal on Optimization*, 18(4):1326–1350, 2008.

[20] G. H. G. H. Golub and C. F. Van Loan. *Matrix computations*. The Johns Hopkins University Press, fourth edition. edition, 2013. ISBN 9781421408590.

[21] Grouplens. Movielens 1m dataset, 2003. Data retrieved from Grouplens, https://grouplens.org/datasets/movielens/1m/.

[22] E. S. Helou and L. E. A. Simões. ϵ-subgradient algorithms for bilevel convex optimization. *Inverse problems*, 33(5):55020–, 2017.

[23] M. Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 427–435. PMLR, 17–19 Jun 2013. URL https://proceedings.mlr.press/v28/jaggi13.html.

[24] R. Jiang, N. Abolfazli, A. Mokhtari, and E. Yazdandoost Hamedani. A conditional gradient-based method for simple bilevel optimization with convex lower-level problem. In *Proceedings of The 26th International Conference on Arti cial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 10305–10323. PMLR, 25–27 Apr 2023.

[25] 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.

[26] G. Lan, E. Romeijn, and Z. Zhou. Conditional gradient methods for convex optimization with general affine and nonlinear constraints. *SIAM Journal on Optimization*, 31(3):2307–2339, 2021.

[27] Y. Malitsky. The primal-dual hybrid gradient method reduces to a primal method for linearly constrained optimization problems. Technical report, Institute for Numerical and Applied Mathematics, University of Göttingen, 2017. URL https://arxiv.org/pdf/1706.02602.pdf.

[28] H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.

[29] R. Merchav and S. Sabach. Convex bi-level optimization problems with non-smooth outer objective function. Technical report, Faculty of Data and Decision Sciences, Technion–Israel Institute of Technology, 2023. URL https://arxiv.org/pdf/2307.08245.pdf.

[30] J.-P. Penot. Error bounds, calmness and their applications in nonsmooth analysis. In *Contemporary mathematics - American Mathematical Society*, volume 514, pages 225–247. American Mathematical Society, 2010.

[31] S. Sabach and S. Shtern. A first order method for solving convex bilevel optimization problems. *SIAM journal on Optimization*, 27(2):640–660, 2017.

[32] Y. Shehu, P. T. Vuong, and A. Zemkoho. An inertial extrapolation method for convex simple bilevel optimization. *Optimization Methods and Software*, 36(1):1–19, 2021.

[33] L. Shen, N. Ho-Nguyen, and F. Kılınç-Karzan. An online convex optimization-based framework for convex bilevel optimization. *Mathematical Programming*, 198(2):1519–1582, 2023.

[34] M. V. Solodov. An explicit descent method for bilevel convex optimization. *Journal of Convex Analysis*, 14:227–237, 2006.

[35] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Technical report, Department of Mathematics, University of Washington, 2008. URL https://www.mit.edu/~dimitrib/PTseng/papers/apgm.pdf.

[36] R. J. Vanderbei. Markowitz models for portfolio optimization, n.d. Data retrieved from, https://vanderbei.princeton.edu/ampl/nlmodels/markowitz/.

[37] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt,

and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[38] S. J. Wright and B. Recht. *Optimization for data analysis*. Cambridge University Press, 2022. ISBN 9781009004282.