THEORY AND APPLICATIONS OF FIRST-ORDER METHODS FOR CONVEX OPTIMIZATION WITH FUNCTION CONSTRAINTS

A Dissertation
Presented to
The Academic Faculty

By

Zhiqiang Zhou

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology

August 2020

THEORY AND APPLICATIONS OF FIRST-ORDER METHODS FOR CONVEX OPTIMIZATION WITH FUNCTION CONSTRAINTS

Approved by:

Dr. Edwin Romeijn, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Guanghui Lan
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Arkadi Nemirovski
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Yao Xie
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Le Song
School of Computational Science and Engineering
Georgia Institute of Technology

Date Approved: July 16, 2020

To my dear parents.

ACKNOWLEDGEMENTS

The past five years have been the most enjoyable journey in my life. I am deeply indebted to a number of remarkable people; without their support, the completion of my dissertation would not be possible.

I have been extremely fortunate to collaborate with my advisor, Professor Edwin Romeijn, who has inspired me with his expertise and vision in the health care area; his professional and enthusiastic attitude have great impact on my research career.

Besides my advisor, I would like to express sincerest attitude to Dr. Guanghui Lan, who led me into the PhD study and trained me with solid foundation in operations research. It was him who equipped me with the rigorous mathematics tools in development of optimization algorithms. Moreover, I would also like to express my deepest appreciation to the rest of my committee members, Dr. Arkadi Nemirovski, Dr. Yao Xie and Dr. Le Song, for taking their precious time attending my defense and their generous and valuable comments to the thesis. I am also very thankful to all the faculty members in ISyE since I have been benefited so much from the courses I took among these years, especially Natashia Boland, David Goldberg, Tuo Zhao, Alexander Shapiro, Santanu Dey, etc. I would also like to thank all the staff of ISyE, including but not limited to Amanda Ford, Laurie Haigh, Peggy Hand, Jonathan Etress, for their assistance.

I am indebted to my fellow students and friends accompanying me in these five years. Thank my officemates Yu Yang and Tianyi Liu for discussion on varies topics during my PhD study. Many thanks go to Yi Zhou, Digvijay Boob, Georgios Kotsalis, Yi Chen for being an excellent research group to work together. I am also fortunate to have friends Zhaowei She, Junqi Hu, Bocheng Wu, Kaixin Yang, Rui Zhang, Liexiao Ding and many others, for playing together and helping me through all difficult time during my PhD study.

Last but definitely not the least, I would like to express the deepest gratitude to my family for their unconditional help and understanding through my entire life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This dissertation focuses on the development of efficient first-order methods for function constrained convex optimization and their applications in a few different areas, including healthcare, finance and machine learning. The thesis consists of three major studies.

The first part of the thesis considers the problem of minimizing an expectation function over a closed convex set, coupled with a functional or expectation constraint on either decision variables or problem parameters. We first present a new stochastic approximation (SA) type algorithm, namely the cooperative SA (CSA), to handle problems with the constraint on devision variables. We show that this algorithm exhibits the optimal $\mathcal{O}(1/\epsilon^2)$ rate of convergence, in terms of both optimality gap and constraint violation, when the objective and constraint functions are generally convex, where ϵ denotes the optimality gap and infeasibility. Moreover, we show that this rate of convergence can be improved to $\mathcal{O}(1/\epsilon)$ if the objective and constraint functions are strongly convex. We then present a variant of CSA, namely the cooperative stochastic parameter approximation (CSPA) algorithm, to deal with the situation when the constraint is defined over problem parameters and show that it exhibits similar optimal rate of convergence to CSA. It is worth noting that CSA and CSPA are primal methods which do not require the iterations on the dual space and/or the estimation on the size of the dual variables. To the best of our knowledge, this is the first time that such optimal SA methods for solving functional or expectation constrained stochastic optimization are presented in the literature. In addition, we apply the CSA and CSPA methods to an asset allocation problem, and a combined classification and metric learning problem, respectively.

The second part of the thesis is devoted to conditional gradient methods which have attracted much attention in both machine learning and optimization communities recently. These simple methods can guarantee the generation of sparse solutions. In addition, without the computation of full gradients, they can handle huge-scale problems sometimes even

with an exponentially increasing number of decision variables. This study aims to significantly expand the application areas of these methods by presenting new conditional gradient methods for solving convex optimization problems with general affine and nonlinear constraints. More specifically, we first present a new constraint extrapolated condition gradient (CoexCG) method that can achieve an $\mathcal{O}(1/\epsilon^2)$ iteration complexity for both smooth and structured nonsmooth function constrained convex optimization. We further develop novel variants of CoexCG, namely constraint extrapolated and dual regularized conditional gradient (CoexDurCG) methods, that can achieve similar iteration complexity to CoexCG but allow adaptive selection for algorithmic parameters. We illustrate the effectiveness of these methods for solving an important class of radiation therapy treatment planning problems arising from healthcare industry.

In the third part of the thesis, we extend the convex function constrained optimization to the multi-stage setting, i.e., multi-stage stochastic optimization problems with convex objectives and conic constraints at each stage. We present a new stochastic first-order method, namely the dynamic stochastic approximation (DSA) algorithm, for solving these types of stochastic optimization problems. We show that DSA can achieve an optimal $\mathcal{O}(1/\epsilon^4)$ rate of convergence in terms of the total number of required scenarios when applied to a three-stage stochastic optimization problem. We further show that this rate of convergence can be improved to $\mathcal{O}(1/\epsilon^2)$ when the objective function is strongly convex. We also discuss variants of DSA for solving more general multi-stage stochastic optimization problems with the number of stages $T > 3$. The developed DSA algorithms only need to go through the scenario tree once in order to compute an ϵ-solution of the multi-stage stochastic optimization problem. As a result, the memory required by DSA only grows linearly with respect to the number of stages. To the best of our knowledge, this is the first time that stochastic approximation type methods are generalized for multi-stage stochastic optimization with $T \geq 3$. We apply the DSA method for solving a class of multi-stage asset allocation problem and demonstrate its potential advantages over existing methods, especially when the

planning horizon T is relatively short but the number of assets is large.

CHAPTER 1
INTRODUCTION

Much recent research effort have been devoted to the applications of the first-order methods to problems in many areas such as operations research, finance, data analysis and machine learning, etc. This dissertation aims to develop efficient first-order methods for function constrained convex optimization and their applications in a few different areas.

The dissertation is driven by and concentrated on the following three different types of problems.

1.1 Stochastic optimization with function or expectation constraints

For this type of problem, we study two related classes of stochastic programming with function or expectation constraints.The first one is a classical SP problem with the function constraint over the decision variables, formally defined as

$$\begin{aligned} \min\ & f(x) := \mathbb{E}[F(x,\xi)] \\ \text{s.t. } & g(x) \leq 0, \\ & x \in X, \end{aligned} \tag{1.1}$$

where $X \subseteq \mathbb{R}^n$ is a convex compact set, ξ are random vectors supported on $\mathcal{P} \subseteq \mathbb{R}^p$, $F(x,\xi) : X \times \mathcal{P} \mapsto \mathbb{R}$ and $g(x) : X \mapsto \mathbb{R}$ are closed convex functions w.r.t. x for a.e. $\xi \in \mathcal{P}$. Moreover, we assume that ξ are independent of x. Under these assumptions, (1.1) is a convex optimization problem.

In particular, the constraint function $g(x)$ in problem (1.1) can be given in the form of expectation as

$$g(x) := \mathbb{E}_\xi[G(x,\xi)], \tag{1.2}$$

where $G(x,\xi) : X \times \mathcal{P} \mapsto \mathbb{R}$ are closed convex functions w.r.t. x for a.e. $\xi \in \mathcal{P}$. Such problems have many applications in operations research, finance and data analysis. One motivating example is SP with the conditional value at risk (CVaR) constraint. In an important work [1], Rockafellar and Uryasev shows that a class of asset allocation problem can be modeled as

$$\begin{array}{ll} \min_{x,\tau} & -\mu^T x \\ \text{s.t.} & \tau + \frac{1}{\beta}\mathbb{E}\{[-\xi^T x - \tau]_+\} \leq 0, \\ & \sum_{i=1}^n x_i = 1, x \geq 0, \end{array} \tag{1.3}$$

where ξ denotes the random return with mean $\mu = \mathbb{E}[\xi]$. Expectation constraints also play an important role in providing tight convex approximation to chance constrained problems (e.g., Nemirovksi and Shapiro [2]). Some other important applications of (1.1) can be found in semi-supervised learning (see, e.g., [3]). For example, one can use the objective function to define the fidelity of the model for the labelled data, while using the constraint to enforce some other properties of the model for the unlabelled data (e.g., proximity for data with similar features).

While problem (1.1) covers a wide class of problems with constraints over the decision variables, in practice we often encounter the situation where the constraint is defined over the problem parameters. Under these circumstances our goal is to find a pair of parameters x^* and decision variables $y^*(x^*)$ such that

$$y^*(x^*) \in \text{Argmin}_{y \in Y} \left\{\phi(x^*, y) := \mathbb{E}[\Phi(x^*, y, \zeta)]\right\}, \tag{1.4}$$

$$x^* \in \left\{x \in X | g(x) := \mathbb{E}[G(x,\xi)] \leq 0\right\}. \tag{1.5}$$

Here $\Phi(x,y,\zeta)$ is convex w.r.t. y for a.e. $\zeta \in \mathcal{Q} \subseteq \mathbb{R}^q$ but possibly nonconvex w.r.t. (x,y) jointly, and $g(\cdot)$ is convex w.r.t. x. Moreover, we assume that ζ is independent of x and y, while ζ is not necessarily independent of x^*. Note that (1.4)-(1.5) defines a pair of optimization and feasibility problems coupled through the following ways: a) the

solution to (1.5) defines an admissible parameter of (1.4); b) ξ can be a random variable with probability distribution parameterized by x^*.

Problem (1.4)-(1.5) also has many applications, especially in data analysis. One such example is to learn a classifier w with a certain metric $\bar{A}$ using the support vector machine model:

$$\min_{w} \mathbb{E}[l(w; (\bar{A}^{\frac{1}{2}}u, v))] + \tfrac{\lambda}{2}\|w\|^2, \tag{1.6}$$

$$\bar{A} \in \left\{A \succeq 0 | \mathbb{E}[|\mathrm{Tr}(A(u_i - v_j)(u_i - v_j)^T) - b_{ij}|] \leq 0, \mathrm{Tr}(A) \leq C\right\}, \tag{1.7}$$

where $l(w; (\theta, y)) = \max\{0, 1 - y\langle w, \theta\rangle\}$ denotes the hinge loss function, $u, u_i, u_j \in \mathbb{R}^n$, $v, v_i, v_j \in \{+1, -1\}$, and $b_{ij} \in \mathbb{R}$ are the random variables satisfying certain probability distributions, and $\lambda, C > 0$ are certain given parameters. In this problem, (1.6) is used to learn the classifer w by using the metric $\bar{A}$ satisfying certain requirements in (1.7), including the low rank (or nuclear norm) assumption. Problem (1.4)-(1.5) can also be used in some data-driven applications, where one can use (1.5) to specify the parameters for the probabilistic models associated with the random variable ξ, as well as some other applications for multi-objective stochastic optimization.

In spite of its wide applicability, the study on efficient solution methods for expectation constrained optimization is still limited. For the sake of simplicity, suppose for now that ξ is given as a deterministic vector and hence that the objective functions f and ϕ in (1.1) and (1.4) are easily computable. One popular method to solve stochastic optimization problems is called the sample average approximation (SAA) approach ([4, 5, 6]). To apply SAA for (1.1) and (1.5), we first generate a random sample $\xi_i, i = 1, \ldots, N$, for some $N \geq 1$ and then approximate g by $\tilde{g}(x) = \frac{1}{N}\sum_{i=1}^{N} G(x, \xi_i)$. The main issues associated with the SAA for solving (1.1) include: i) the deterministic SAA problem might not be feasible; ii) the resulting deterministic SAA problem is often difficult to solve especially when N is large, requiring going through the whole sample $\{\xi_1, \ldots, \xi_N\}$ at each iteration; and ii) it is not

applicable to the on-line setting where one needs to update the decision variable whenever a new piece of sample ξ_i, $i = 1, \ldots N$, is collected.

A different approach to solve stochastic optimization problems is called stochastic approximation (SA), which was initially proposed in a seminal paper by Robbins and Monro[7] in 1951 for solving strongly convex SP problems. This algorithm mimics the gradient descent method by using the stochastic gradient $F'(x, \xi_i)$ rather than the original gradient $f'(x)$ for minimizing $f(x)$ in (1.1) over a simple convex set X (see also [8, 9, 10, 11, 12, 13]). An important improvement of this algorithm was developed by Polyak and Juditsky([14],[15]) through using longer steps and then averaging the obtained iterates. Their method was shown to be more robust with respect to the choice of stepsize than classic SA method for solving strongly convex SP problems. More recently, Nemirovski et al. [16] presented a modified SA method, namely, the mirror descent SA method, and demonstrated its superior numerical performance for solving a general class of nonsmooth convex SP problems. The SA algorithms have been intensively studied over the past few years (see, e.g., [17, 18, 19, 20, 21, 22, 23, 24]). It should be noted, however, that none of these SA algorithms are applicable to expectation constrained problems, since each iteration of these algorithms requires the projection over the feasible set $\{x \in X | g(x) \leq 0\}$, which is computationally prohibitive as g is given in the form of expectation.

1.2 Huge-scale convex optimization with function constraints

For this type of problem, we focus on the development of conditional gradient type methods for solving the convex optimization problem in following form:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = 0, \\ & h_i(x) \leq 0, \;\; i = 1, \ldots, d, \\ & x \in X. \end{aligned} \tag{1.8}$$

Here $X \subseteq \mathbb{R}^n$ is a compact convex set, $f : X \to \mathbb{R}$ and $h_i : X \to \mathbb{R}$, $i = 1, \ldots, d$, are proper lower semicontinuous convex functions, $A : \mathbb{R}^n \to \mathbb{R}^m$ denotes a linear mapping, and b is a given vector in $\mathbb{R}^m$. We assume that X is relatively simple in the sense that one can minimize a linear function over X easily. Throughout this chapter we assume that an optimal solution x^* of problem (1.8) exists. For notational convenience, we often denote $h(x) \equiv (h_1(x); \ldots, h_d(x))$.

The conditional gradient method, initially developed by Frank and Wolfe in 1956 [25], is one of the earliest first-order methods for convex optimization. It has been widely used for solving problems with relatively simple convex sets, i.e., when the constraints $g(x) = 0$ and $h_i(x) \leq 0$ do not appear in problem (1.8). Each iteration of this method computes the gradient of f at the current search point x_k, and then solves the subproblem $\min_{x \in X} \langle \nabla f(x_k), x \rangle$ to update the solution. In comparison with most other first-order methods, it does not require the projection over X, which in many cases could be computationally more expensive than to minimize a linear function over X (e.g.. when X is a spectrahedron given by $X := \{X \succeq 0 : \mathrm{Tr}(X) = 1\}$). These simple methods can also guarantee the generation of sparse solutions, e.g., when X is a simplex or spectrahedron. In addition, without the computation of full gradients, they can handle huge-scale problems sometimes even with an exponentially increasing number of decision variables.

Much recent research effort has been devoted to the complexity analysis of conditional gradient methods over simple convex set X. It is well-known that if f is a smooth convex function, then this algorithm can find an ϵ-solution (i.e., a point $\bar{x} \in X$ s.t. $f(\bar{x}) - f^* \leq \epsilon$) in at most $\mathcal{O}(1/\epsilon)$ iterations (see [26, 27, 28, 29, 30]). In fact, such a complexity result has been established for the conditional gradient method under a stronger termination criterion called Wolfe Gap, based on the first-order optimality condition [26, 27, 28, 29, 30]. As shown in [26, 28, 31], this $\mathcal{O}(1/\epsilon)$ iteration complexity bound is tight for smooth convex optimization. In addition, if f is a nonsmooth function with a saddle point structure, one can not achieve an iteration complexity better than $\mathcal{O}(1/\epsilon^2)$ [28], in terms of the number

of times to solve the linear optimization subproblem. One possible way to improve the complexity bounds is to use the conditional gradient sliding methods developed in [32] to reduce the number of gradient evaluations. Many other variants of conditional gradient methods have also been proposed in the literature (see, e.g.,[33, 34, 35, 36, 29, 37, 30, 26, 27, 38, 39, 40, 41, 42, 43]) and Chapter 7 of [44] for an overview of these methods).

It should be noted, however, that none of existing conditional gradient methods can be used to efficiently solve the more general function constrained convex optimization problem in (1.8). With these function constraints ($g(x) = 0$ and $h_i(x) \leq 0$), linear optimization over the feasible region of problem (1.8) could become much more difficult. As an example, if X is the aforementioned spectrahedron and h does not exist, the linear optimization problem over the feasible region $\{x \succeq 0 : g(x) = 0, \mathrm{Tr}(X) = 1\}$ becomes a general semidefinite programming problem. Adding nonlinear function constraints $h_i(x) \leq 0$ usually makes the subproblem even harder. In fact, our study has been directly motivated by a convex optimization problem with nonlinear function constraints arising from radiation therapy treatment planing (see [45, 46, 47, 48, 49, 50, 51] and Section 4.5 for more details). The objective function of this problem, representing the quality of the treatment plan, is smooth and convex. Besides a simplex constraint, it consists of two types of nonlinear function constraints, namely the group sparsity constraint to reduce radiation exposure for the patients, and the risk averse constraints to avoid overdose (resp., underdose) to healthy (resp., tumor) structures. This problem is highly challenging because the dimension of the decision variables can increase exponentially with respect to the size of data, which prevents the computation of full gradients as required by most existing optimization methods dealing with function constraints.

1.3 Multi-stage stochastic optimization problem

Multi-stage stochastic optimization aims at optimal decision-making over multiple periods of time, where the decision in the current period has to take into account what will happen

in the future. This type of decision-making is very important to a few applications areas, including finance, logistics, robotics and clinic trials etc. In this chapter, we are interested in solving a class of multi-stage stochastic optimization problems given by

$$\begin{array}{lll} \min h^1(x^1, c^1) + & \mathbb{E}_{|\xi^1}\left[\min h^2(x^2, c^2) \right. & + \mathbb{E}_{|\xi^{[2]}}\left[\ldots + \right. \quad \mathbb{E}_{|\xi^{[T-1]}}\left[\min\ h^T(x^T, c^T)\right]\right]\right] \\ \text{s.t. } A^1x^1 - b^1 \in K^1 & \text{s.t. } A^2x^2 - b^2 - B^2x^1 \in K^2, & \text{s.t. } A^Tx^T - b^T - B^Tx^{T-1} \in K^T, \\ x^1 \in X^1, & x^2 \in X^2, & x^T \in X^T. \end{array} \tag{1.9}$$

Here T denotes the number of stages, $h^t(\cdot, c^t)$ are relatively simple convex functions, K^t are closed convex cones, $X^t \subseteq \mathbb{R}^{n_t}$ are compact convex sets for some $n_t > 0$, $h^t : X^t \to \mathbb{R}$ are relatively simple convex functions, and A^t denote the linear mappings from $\mathbb{R}^{n_t}$ to $\mathbb{R}^{m_t}$ for some $m_t > 0$. Moreover, $\xi^1 := (A^1, b^1, c^1)$ is a given deterministic vector, $\xi^t := (A^t, b^t, B^t, c^t)$, $t = 2, \ldots, T$, are the random vectors supported on Ξ^t at stage t. Throughout this chapter, we use $\xi^{[t]} := (\xi^1, \ldots, \xi^t)$ to denote the stochastic process up to time period t, and $\mathbb{E}_{|\xi^{[t]}}(\cdot) \equiv \mathbb{E}[\cdot|\xi^{[t]}]$ denote the expectation conditional on $\xi^{[t]}$. It is worth noting that $\xi^{[1]} = \xi^1$ and that $\mathbb{E}_{|\xi^1}[\cdot] \equiv \mathbb{E}_{|\xi^{[1]}}[\cdot] = \mathbb{E}[\cdot]$ since ξ^1 is deterministic. By defining value functions, we can write problem (1.9) equivalently as

$$\begin{array}{l} \min\ h^1(x^1, c^1) + v^2(x^1, \xi^{[1]}) \\ \text{s.t. } A^1x^1 - b^1 \in K^1, \\ x^1 \in X^1, \end{array} \tag{1.10}$$

where the value factions v^t are recursively defined by

$$\begin{array}{rcl} v^t(x^{t-1}, \xi^{[t-1]}) & := & \mathbb{E}[V^t(x^{t-1}, \xi^{[t]})|\xi^{[t-1]}], \ \ t = 2, \ldots, T-1, \\ V^t(x^{t-1}, \xi^{[t]}) & := & \min\ h^t(x^t, c^t) + v^{t+1}(x^t, \xi^{[t]}) \\ & & \text{s.t. } A^tx^t - b^t - B^tx^{t-1} \in K^t, \\ & & x^t \in X^t, \end{array} \tag{1.11}$$

and

$$\begin{aligned} v^T(x^{T-1}, \xi^{[T-1]}) &:= \mathbb{E}[V^T(x^{T-1}, \xi^{[T]})|\xi^{[T-1]}], \\ V^T(x^{T-1}, \xi^{[T]}) &:= \min\ h^T(x^T, c^T) \\ &\qquad \text{s.t. } A^T x^T - b^T - B^T x^{T-1} \in K^T, \\ &\qquad\qquad x^T \in X^T. \end{aligned} \tag{1.12}$$

In particular, if h^t are affine, $K^t = \{0\}$ and X^t are polyhedral, then problem (1.9) reduces to the well-known multi-stage stochastic linear programming problem (see, e.g., [52, 53]). The incorporation of the nonlinear (but convex) objective function $h^t(x^t, c^t)$ and conic constraints $A^t x^t - b^t - B^t x^{t-1} \in K^t$ allows us to model a much wider class of problems. Moreover, if $T = 2$, then problem (1.9) is often referred to as a two-stage (or static) stochastic programming problem.

In spite of its wide applicability, multi-stage stochastic optimization remains highly challenging to solve. Many existing methods for multi-stage stochastic optimization are based on sample average approximation (see Nemirovski and Shapiro [54] and Shapiro [55]). In this approach, one first generates a deterministic counterpart of (1.9) by replacing the expectations with (conditional) sample averages. In particular, if the number of stages $T = 3$, the total number of samples (a.k.a. scenarios) cannot be smaller than $\mathcal{O}(1/\epsilon^4)$ in general. Once after a deterministic approximation of (1.9) is generated, one can then develop decomposition methods to solve it to certain accuracy. The most popular decomposition methods consist of stage-based and scenario-based decomposition method. One widely-used stage-based method is the stochastic dual dynamic programming (SDDP) algorithm, which is essentially an approximate cutting plane method, first presented by Pereira and Pinto [56] and later studied by Shapiro [57], Philpott et. al. [58], Donohue and Birge [59], Hindsberger [60], and Kozmík and Morton [61] etc. This method has been shown to be effective for solving multi-stage stochastic optimization problems with a large number of stages, but a small number of decision variables. The progressive hedging algorithm by Rockafellar and Wets [62] is a well-known scenario-based decomposition method, which

basically applies an augmented Lagrangian method to penalize the violation of the non-anticipativity constraints. Other interesting bundle type decomposition methods have also been developed (see, e.g., [63]). These methods assume that the scenario tree has been generated and will go through the scenario tree many times. Usually there are no performance guarantees provided regarding their rate of convergence, i.e., the number of times one needs to go through the scenario tree. In SDDP, one also needs to assume that random vectors are stage-wise independent.

Recently, a different approach called stochastic approximation (SA) has attracted much attention for solving static stochastic optimization problems given in the form of

$$\min_{x \in X} \{f(x) := \mathbb{E}_\xi[F(x, \xi)]\}, \tag{1.13}$$

where X is a closed convex set, ξ denotes the random vecctor and $F(\cdot, \xi)$ is a closed convex function. Observe that when $T = 2$, problem (1.9) can be cast in the form of (1.13) and hence one can apply the aforementioned SA methods to solve these two-stage stochastic optimization problems (see [64, 65]). The basic SA algorithm, initially proposed by Robbins and Monro [7], mimics the simple projected gradient descent method by replacing exact gradient with its unbiased estimator. Important improvements for the SA methods have been made by Nemirovski and Yudin [66] and later by Polayk and Juditsky [14, 15]. During the past few years, significant progress has been made in SA methods (e.g., [64, 17, 18, 19, 67, 68, 69, 70, 71, 72]). In particular, Nemirovski et. al. [64] presented a properly modified SA approach, namely, mirror descent SA for solving general nonsmooth convex SP problems. Lan [17] introduced an accelerated SA method, based on Nesterov's accelerated gradient method [73], for solving smooth, nonsmooth and stochastic optimization in a uniform manner. Novel nonconvex SA methods and their accelerated versions have been studied in [67, 69, 74, 71]. Some interesting progresses have also been made in [70, 72] for solving more complicated compositional stochastic optimization problems.

All these SA algorithms only need to access one single ξ_k at each iteration, and hence do not require much memory. It has been shown in [64, 65] that SA methods can significantly outperform the SAA approach for solving static (or two-stage) stochastic programming problems. However, it remains unclear whether these SA methods can be generalized for multi-stage stochastic optimization problems with $T \geq 3$.

CHAPTER 2
ALGORITHMS FOR STOCHASTIC OPTIMIZATION WITH FUNCTION OR EXPECTATION CONSTRAINTS

2.1 Introduction

In this chapter, we intend to develop efficient solution methods for solving expectation constrained problems by properly addressing the aforementioned issues associated with existing SA methods. Our contribution mainly exists in the following several aspects. Firstly, inspired by Polayk's subgradient method for constrained optimization [75] and Nesterov's note [76], we present a new SA algorithm, namely the cooperative SA (CSA) method for solving the SP problem with expectation constraint in (1.1) with constraint (1.2). At the k-th iteration, CSA performs a projected subgradient step along either $F'(x_k, \xi_k)$ or $G'(x_k, \xi_k)$ over the set X, depending on whether an unbiased estimator $\hat{G}_k$ of $g(x_k)$ satisfies $\hat{G}_k \leq \eta_k$ or not. Observe that the aforementioned estimator $\hat{G}_k$ can be easily computed in many cases by using the structure of the problem, e.g., the linear dependence $\xi^T x$ in (1.3) (see Section 4.1 in [65] and Section 2.1 for more details). We introduce an index set $\mathcal{B} := \{1 \leq k \leq N : \hat{G}_k \leq \eta_k\}$ in order to compute the output solution as a weighted average of the iterates in $\mathcal{B}$. By carefully bounding $|\mathcal{B}|$, we show that the number of iterations performed by the CSA algorithm to find an ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathbb{E}[f(\bar{x}) - f^*] \leq \epsilon$ and $\mathbb{E}[g(\bar{x})] \leq \epsilon$, can be bounded by $\mathcal{O}(1/\epsilon^2)$. Moreover, when both f and g are strongly convex, by using a different set of algorithmic parameters we show that the complexity of the CSA method can be significantly improved to $\mathcal{O}(1/\epsilon)$. It it is worth mentioning that this result is new even for solving deterministic strongly convex problems with function constraints. We also established the large-deviation properties for the CSA method under certain light-tail assumptions.

Secondly, we develop a variant of CSA, namely the cooperative stochastic parameter approximation (CSPA) method for solving the SP problem with expectation constraints on problem parameters in (1.4)-(1.5). In CSPA, we update parameter x by running the mirror descend SA iterates whenever a certain easily verifiable condition is violated. Otherwise, we update the decision variable y while keeping x intact. We show that the number of iterations performed by the CSPA algorithm to find an ϵ-solution of (1.4)-(1.5), i.e., a pair of solution $(\bar{x}, \bar{y})$ s.t. $\mathbb{E}[g(\bar{x})] \leq \epsilon$ and $\mathbb{E}[\phi(\bar{x}, \bar{y}) - \phi(\bar{x}, y^*(\bar{x})] \leq \epsilon$, can be bounded by $\mathcal{O}(1/\epsilon^2)$. Moreover, this bound can be significantly improved to $\mathcal{O}(1/\epsilon)$ if G and Φ are strongly convex w.r.t. x and y, respectively.

To the best of our knowledge, all the aforementioned algorithmic developments are new in the stochastic optimization literature. It is also worth mentioning a few alternative or related methods to solve (1.1) and (1.4)-(1.5). First, without efficient methods to directly solve (1.1), current practice resorts to reformulate it as $\min_{x \in X} \lambda f(x) + (1 - \lambda) g(x)$ for some $\lambda \in (0, 1)$. However, one then has to face the difficulty of properly specifying λ, since an optimal selection would depend on the unknown dual multiplier. As a consequence, we cannot assess the quality of the solutions obtained by solving this reformulated problem. Second, one alternative approach to solve (1.1) is the penalty-based or primal-dual approach. However these methods would require either the estimation of the optimal dual variables or iterations performed on the dual space (see [77], [16] and [78]). Moreover, the rate of convergence of these methods for function constrained problems has not been well-understood other than conic constraints even for the deterministic setting. Third, in [79] (and see references therein), Jiang and Shanbhag developed a coupled SA method to solve a stochastic optimization problem with parameters given by another optimization problem, and hence is not applicable to problem (1.4)-(1.5). Moreover, each iteration of their method requires two stochastic subgradient projection steps and hence is more expensive than that of CSPA.

The remaining part of this paper is organized as follows. In Section 2, we present the

CSA algorithm and establish its convergence properties under general convexity and strong convexity assumptions. Then in Section 3, we develop a variant of the CSA algorithm, namely the CSPA for solving SP problems with the expectation constraint over problem parameters and discuss its convergence properties. We then present some numerical results for these new SA methods in section 4. Finally some concluding remarks are added in Section 5.

2.2 function or expectation constraints over decision variables

In this section we present the cooperative SA (CSA) algorithm for solving convex stochastic optimization problems with the constraint over decision variables. More specifically, we first briefly review the distance generating function and prox-mapping in Subsection 2.2.1. We then describe the CSA algorithm in Subsection 2.2.2 and discuss its convergence properties in terms of expectation and large deviation for solving general convex problems in Subsection 2.2.3. Then we show how to apply the CSA algorithm to problem (1.1) with expectation constraint and discuss its large deviation properties in Subsection 2.2.4. Finally, we show how to improve the convergence of this algorithm by imposing strong convexity assumptions to problem (1.1) in Subsection 2.2.5.

2.2.1 Preliminary: prox-mapping

Recall that a function $\omega_X : X \mapsto R$ is a distance generating function with parameter α, if ω_X is continuously differentiable and strongly convex with parameter α with respect to $\|\cdot\|$. Without loss of generality, we assume throughout this paper that $\alpha = 1$, because we can always rescale $\omega_X(x)$ to $\bar{\omega}_X(x) = \omega_X(x)/\alpha$. Therefore, we have

$$\langle x - z, \nabla\omega_X(x) - \nabla\omega_X(z)\rangle \geq \|x - z\|^2, \forall x, z \in X.$$

The prox-function associated with ω is given by

$$V_X(z,x) = \omega_X(x) - \omega_X(z) - \langle \nabla\omega_X(z), x - z\rangle.$$

$V_X(\cdot,\cdot)$ is also called the Bregman's distance, which was initially studied by Bregman [80] and later by many others (see [81],[82] and [83]). In this paper we assume the prox-function $V_X(x,z)$ is chosen such that, for a given $x \in X$, the prox-mapping $P_{x,X} : \mathbb{R}^n \mapsto \mathbb{R}^n$ defined as

$$P_{x,X}(\cdot) := \text{argmin}_{z\in X}\{\langle \cdot, z\rangle + V_X(x,z)\} \tag{2.1}$$

is easily computed.

It can be seen from the strong convexity of $\omega(\cdot,\cdot)$ that

$$V_X(x,z) \geq \tfrac{1}{2}\|x - z\|^2, \forall x, z \in X. \tag{2.2}$$

Whenever the set X is bounded, the distance generating function ω_X also gives rise to the diameter of X that will be used frequently in our convergence analysis:

$$D_X \equiv D_{X,\omega_X} := \sqrt{\max_{x,z\in X} V_X(x,z)}. \tag{2.3}$$

The following lemma follows from the optimality condition of (2.1) and the definition of the prox-function (see the proof in [16]).

Lemma 1 *For every $u, x \in X$, and $y \in \mathbb{R}^n$, we have*

$$V_X(P_{x,X}(y), u) \leq V_X(x,u) + y^T(u - x) + \tfrac{1}{2}\|y\|_*^2,$$

where the $\|\cdot\|_*$ denotes the conjugate of $\|\cdot\|$, i.e., $\|y\|_* = \max\{\langle x, y\rangle | \|x\| \leq 1\}$.

2.2.2 The CSA method

In this section, we present a generic algorithmic framework for solving the constrained optimization problem in (1.1). We assume the expectation function $f(x)$ and constraint $g(x)$, in addition to being well-defined and finite-valued for every $x \in X$, are continuous and convex on X.

The CSA method can be viewed as a stochastic counterpart of Polayk's subgradient method, which was originally designed for solving deterministic nonsmooth convex optimization problems (see [75] and a more recent generalization in [84]). At each iterate x_k, $k \geq 0$, depending on whether $g(x_k) \leq \eta_k$ for some tolerance $\eta_k > 0$, it moves either along the subgradient direction $f'(x_k)$ or $g'(x_k)$, with an appropriately chosen stepsize γ_k which also depends on $\|f'(x_k)\|_*$ and $\|g'(x_k)\|_*$. However, Polayk's subgradient method cannot be applied to solve (1.1) because we do not have access to exact information about f', g' and g. The CSA method differs from Polyak's subgradient method in the following three aspects. Firstly, the search direction h_k is defined in a stochastic manner: we first check if the solution x_k we computed at iteration k violates the condition $\hat{G}_k \leq \eta_k$ for some $\eta_k \geq 0$. If so, we set the $h_k = G'(x_k, \xi_k)$ for a random realization ξ_k of ξ (Note that for deterministic constraint in (1.1), $h_k = g'(x_k)$) in order to control the violation of expectation constraint. Otherwise, we set $h_k = F'(x_k, \xi_k)$. Secondly, for some $1 \leq s \leq N$, we partition the indices $I = \{s, ..., N\}$ into two subsets: $\mathcal{B} = \{s \leq k \leq N | \hat{G}_k \leq \eta_k\}$ and $\mathcal{N} = I \setminus \mathcal{B}$, and define the output $\bar{x}_{N,s}$ as an ergodic mean of x_k over $\mathcal{B}$. This differs from the Polyak's subgradient method that defines the output solution as the best $x_k, k \in \mathcal{B}$, with the smallest objective value. Thirdly, while the original Polayk's subgradient method were developed only for general nonsmooth problems, we show that the CSA method also exhibits an optimal rate of convergence for solving strongly convex problems by properly choosing $\{\gamma_k\}$ and $\{\eta_k\}$.

Notice that every iteration of CSA requires an unbiased estimator of $g(x_k)$. Suppose there is no uncertainty associated with the constraint in (1.1), we can evaluate $g(x_k)$ exactly.

Algorithm 1 The cooperative SA algorithm

Input: initial point $x_1 \in X$, stepsizes $\{\gamma_k\}$ and tolerances $\{\eta_k\}$.

for $k = 1, 2, \ldots, N$

Let $\hat{G}_k$ be an unbiased estimator of $g(x_k)$. Set

$$h_k = \begin{cases} F'(x_k, \xi_k), & \text{if } \hat{G}_k \leq \eta_k; \\ G'(x_k, \xi_k), & \text{otherwise.} \end{cases} \tag{2.4}$$

$$x_{k+1} = P_{x_k, X}(\gamma_k h_k). \tag{2.5}$$

end for

Output: Set $\mathcal{B} = \{s \leq k \leq N | \hat{G}_k \leq \eta_k\}$ for some $1 \leq s \leq N$, and define the output

$$\bar{x}_{N,s} = (\textstyle\sum_{k \in \mathcal{B}} \gamma_k)^{-1} (\sum_{k \in \mathcal{B}} \gamma_k x_k), \tag{2.6}$$

If g is given in the form of expectation, one natural way is to generate a J-sized i.i.d. random sample of ξ and then evaluate the constraint function value by $\hat{G}_k = \frac{1}{J} \sum_{j=1}^{J} G(x_k, \xi_j)$. However, this basic scheme can be much improved by using some structural information for constraint evaluation. For instance, one ubiquitous structure existing in machine learning and portfolio optimization applications is the linear combination of $\xi^T x$. For a given $x \in X$, we can define a new random variable $\bar{\xi} = \xi^T x$ and generate samples of $\bar{\xi}$ instead of ξ. $\bar{\xi}$ is only of dimension one and it is computationally much cheaper to simulate. Given the distribution of ξ, below we provide a few examples where the distribution of $\bar{\xi}$ can be explicitly computed or approximated. For instance, if $x \in \mathbb{R}^d$, ξ_i are independent normal $N(\mu_i, \sigma_i)$, then $\bar{\xi}$ follows $N(\sum_{i=1}^{d} \mu_i, [\sum_{i=1}^{d} x_i^2 \sigma_i^2]^{1/2})$. If ξ_i follows independent $\exp(\lambda_i)$, then the probability density function of $\bar{\xi}$ is

$$f_{\bar{\xi}}(y) = (\prod_{i=1}^{d} \hat{\lambda}_i) \sum_{j=1}^{d} \frac{e^{-\hat{\lambda}_j y}}{\prod_{k \neq j, k=1}^{d} (\hat{\lambda}_k \hat{\lambda}_j)},$$

where $\hat{\lambda}_i = \lambda_i / x_i$. If ξ_i follows independent Uniform(a, b), then the cumulative distribution

function of $\bar{\xi}$ is

$$F_{\bar{\xi}}(y) = \frac{1}{d!\prod_{i=1}^{d} x_i}\{(\frac{y-a\sum_{i=1}^{d} x_i}{b-a}^{+})^d + \sum_{v=1}^{d}(-1)^v \sum_{j_1=1}^{d}\sum_{j_2=j_1+1}^{d}\cdots \sum_{j_v=j_{v-1}+1}^{d}\{[\frac{y-a\sum_{i=1}^{d} x_i}{b-a} - (x_{j_1}+x_{j_2}+\ldots+x_{j_v})]^+\}\}.$$

If the ξ_i are dependent normal random variables with mean μ and covariance C (by Cholesky decomposition, $C = LL'$), we can estimate $\sum_{i=1}\xi_i x_i$ by $\sum_{i=1}^{d}\mu_i x_i + \bar{r}[\sum_{i=1}^{d}(L^T x)_i^2]^{1/2}$, where $\bar{r}$ follows $N(0,1)$. In fact, when the dimension d is large enough, by central limit theorem, we can use a normal distribution to approximate the new random variable $\bar{\xi}$. These are a few examples showing that to simulate $\bar{\xi}$ can be much faster than to simulate the original random variables for constraint evaluation.

2.2.3 Convergence of CSA for SP with function constraints

In this subsection, we consider the case when the constraint function g is deterministic (i.e., $\hat{G}_k = g'(x_k)$). Our goal is to establish the rate of convergence associated with CSA, in terms of both the distance to the optimal value and the violation of constraints. It should also be noted that Algorithm 1 is conceptional only as we have not specified a few algorithmic parameters (e.g. $\{\gamma_k\}$ and $\{\eta_k\}$). We will come back to this issue after establishing some general properties about this method. Throughout this subsection, we make the following assumptions.

Assumption 1 *For any $x \in X$, a.e. $\xi \in \mathcal{P}$,*

$$\mathbb{E}[\|F'(x,\xi)\|_*^2] \le M_F^2 \ \ \textit{and} \ \ \|g'(x)\|_*^2 \le M_G^2,$$

where $F'(x,\xi) \in \partial_x F(x,\xi)$ and $g'(x) \in \partial_x g(x)$.

The following result establishes a simple but important recursion about the CSA method

for problem (1.1).

Proposition 2 *For any $1 \leq s \leq N$, we have*

$$\begin{aligned} &\textstyle\sum_{k\in\mathcal{N}} \gamma_k(\eta_k - g(x)) + \sum_{k\in\mathcal{B}} \gamma_k \langle F'(x_k,\xi_k), x_k - x\rangle \\ &\leq V(x_s,x) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 \|g'(x_k)\|_*^2, \end{aligned} \tag{2.7}$$

for all $x \in X$.

Proof. For any $s \leq k \leq N$, using Lemma 1, we have

$$V(x_{k+1},x) \leq V(x_k,x) + \gamma_k\langle h_k, x - x_k\rangle + \tfrac{1}{2}\gamma_k^2\|h_k\|_*^2. \tag{2.8}$$

Observe that if $k \in \mathcal{B}$, we have $h_k = F'(x_k,\xi_k)$, and

$$\langle h_k, x_k - x\rangle = \langle F'(x_k,\xi_k), x_k - x\rangle.$$

Moreover, if $k \in \mathcal{N}$, we have $h_k = g'(x_k)$ and

$$\langle h_k, x_k - x\rangle = \langle g'(x_k), x_k - x\rangle \geq g(x_k) - g(x) \geq \eta_k - g(x).$$

Summing up the inequalities in (2.8) from $k = s$ to N and using the previous two observations, we obtain

$$\begin{aligned} V(x_{k+1},x) &\leq V(x_s,x) - \textstyle\sum_{k=s}^N \gamma_k\langle h_k, x_k - x\rangle + \tfrac{1}{2}\sum_{k=s}^N \gamma_k^2\|h_k\|_*^2 \\ &\leq V(x_s,x) - \big[\textstyle\sum_{k\in\mathcal{N}} \gamma_k\langle g'(x_k), x_k - x\rangle + \sum_{k\in\mathcal{B}} \gamma_k\langle F'(x_k,\xi_k), x_k - x\rangle\big] \\ &\quad + \tfrac{1}{2}\textstyle\sum_{k=s}^N \gamma_k^2\|h_k\|_*^2 \\ &\leq V(x_s,x) - \big[\textstyle\sum_{k\in\mathcal{N}} \gamma_k(\eta_k - g(x)) + \sum_{k\in\mathcal{B}} \gamma_k\langle F'(x_k,\xi_k), x_k - x\rangle\big] \\ &\quad + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2\|F'(x_k,\xi_k)\|_*^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2\|g'(x_k)\|_*^2. \end{aligned} \tag{2.9}$$

Rearranging the terms in above inequality, we obtain (2.7) ∎

Using Proposition 2, we present below a sufficient condition under which the output solution $\bar{x}_{N,s}$ is well-defined.

Lemma 3 *Let x^* be an optimal solution of* (1.1)*. If*

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k\eta_k > D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2, \tag{2.10}$$

then $\mathcal{B} \neq \emptyset$, i.e., $\bar{x}_{N,s}$ is well-defined. Moreover, we have one of the following two statements holds,

a) $|\mathcal{B}| \geq (N-s+1)/2$,

b) $\sum_{k\in\mathcal{B}} \gamma_k\langle f'(x_k), x_k - x^*\rangle \leq 0$.

Proof. Taking expectation w.r.t. ξ_k on both sides of (2.7) and fixing $x = x^*$, we have

$$\begin{aligned} &\textstyle\sum_{k\in\mathcal{N}} \gamma_k[\eta_k - g(x^*)] + \sum_{k\in\mathcal{B}} \gamma_k\langle f'(x_k), x_k - x^*\rangle \\ &\leq V(x_s, x^*) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2 \\ &\leq D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2. \end{aligned} \tag{2.11}$$

Suppose for contradiction that $\mathcal{B} = \emptyset$. We then conclude from the above relation and the fact $g(x^*) \leq 0$ that

$$N \min_{k\in\mathcal{N}} \gamma_k\eta_k \leq \textstyle\sum_{k\in\mathcal{N}} \gamma_k[\eta_k - g(x^*)] \leq D_X^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2,$$

which contradicts with (2.10). Hence, we must have $\mathcal{B} \neq \emptyset$.

Now if $\sum_{k\in\mathcal{B}} \gamma_k\langle f'(x_k), x_k - x^*\rangle \leq 0$, part b) holds. Otherwise, if $\sum_{k\in\mathcal{B}} \gamma_k\langle f'(x_k), x_k - x^*\rangle \geq 0$, we have

$$\textstyle\sum_{k\in\mathcal{N}} \gamma_k[\eta_k - g(x^*)] \leq V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2,$$

which, in view of $g(x^*) \leq 0$, implies that

$$\sum_{k\in\mathcal{N}} \gamma_k\eta_k \leq V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2. \tag{2.12}$$

Suppose that $|\mathcal{B}| < (N-s+1)/2$, i.e., $|\mathcal{N}| \geq (N-s+1)/2$. Then,

$$\sum_{k\in\mathcal{N}} \gamma_k\eta_k \geq \tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k\eta_k > V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2,$$

which contradicts with (2.12). Hence, part a) holds. ∎

Now we are ready to establish the main convergence properties of the CSA method.

Theorem 4 *Suppose that* $\{\gamma_k\}$ *and* $\{\eta_k\}$ *in the CSA algorithm are chosen such that* (2.10) *holds. Then for any* $1 \leq s \leq N$*, we have*

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \leq \frac{2D_X^2 + M^2 \sum_{s\leq k\leq N} \gamma_k^2}{(N-s+1)\min_{s\leq k\leq N} \gamma_k}, \tag{2.13}$$

$$g(\bar{x}_{N,s}) \leq (\textstyle\sum_{k\in\mathcal{B}} \gamma_k)^{-1}(\sum_{k\in\mathcal{B}} \gamma_k\eta_k), \tag{2.14}$$

where $M := \max\{M_F, M_G\}$.

Proof. We first show (2.13). By Lemma 2, if Lemma 2 part (b) holds, dividing both sides by $\sum_{k\in\mathcal{B}} \gamma_k$ and taking expectation, we have

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \leq 0. \tag{2.15}$$

If $|\mathcal{B}| \geq (N-s+1)/2$, we have $\sum_{k\in\mathcal{B}} \gamma_k \geq |\mathcal{B}| \min_{k\in\mathcal{B}} \gamma_k \geq \frac{N-s+1}{2} \min_{k\in\mathcal{B}} \gamma_k$. It follows from the definition of $\bar{x}_{N,s}$ in (2.6), the convexity of $f(\cdot)$ and (2.11) that

$$\begin{aligned} \sum_{k\in\mathcal{N}} \gamma_k\eta_k + \sum_{k\in\mathcal{B}} \gamma_k E[f(\bar{x}_{N,s}) - f(x^*)] &\leq \sum_{k\in\mathcal{N}} \gamma_k\eta_k + \sum_{k\in\mathcal{B}} E[\gamma_k(f(x_k) - f(x^*))] \\ &\leq D_X^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2, \end{aligned}$$

which implies that

$$|\mathcal{N}| \min_{k\in\mathcal{N}} \gamma_k\eta_k + \left(\sum_{k\in\mathcal{B}} \gamma_k\right) E[f(\bar{x}_{N,s}) - f(x^*)] \leq D_X^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2. \tag{2.16}$$

Using this bound and the fact $\gamma_k\eta_k \geq 0$ in (2.16), we have

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \leq \frac{2D_X^2 + \sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2}{(N-s+1)\min_{k\in I}\gamma_k} \leq \frac{2D_X^2 + M^2\sum_{s\leq k\leq N}\gamma_k^2}{(N-s+1)\min_{k\in\mathcal{B}}\gamma_k}. \tag{2.17}$$

Combining these two inequalities (2.15) and (2.17), we have (2.13). Now we show that (2.14) holds. For any $k \in \mathcal{B}$, we have $g(x_k) \leq \eta_k$. Then, in view of the definition of $\bar{x}_{N,s}$ in (2.6) and the convexity of $g(\cdot)$, then implies that

$$g(\bar{x}_{N,s}) \leq \frac{\sum_{k\in\mathcal{B}} \gamma_k g(x_k)}{\sum_{k\in\mathcal{B}} \gamma_k} \leq \frac{\sum_{k\in\mathcal{B}} \gamma_k\eta_k}{\sum_{k\in\mathcal{B}} \gamma_k}. \tag{2.18}$$

■

Below we provide a few specific selections of $\{\gamma_k\}$, $\{\eta_k\}$ and s that lead to the optimal rate of convergence for the CSA method. In particular, we will present a constant and variable stepsize policy, respectively, in Corollaries 5 and 6.

Corollary 5 *If s=1,* $\gamma_k = \frac{D_X}{\sqrt{N}(M_F+M_G)}$ *and* $\eta_k = \frac{4(M_F+M_G)D_X}{\sqrt{N}}$, $k = 1, ...N$, *then*

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \leq \tfrac{4D_X(M_F+M_G)}{\sqrt{N}},$$
$$g(\bar{x}_{N,s}) \leq \tfrac{4D_X(M_F+M_G)}{\sqrt{N}}.$$

Proof. First, observe that condition (2.10) holds by using the facts that

$$\begin{aligned}
&\tfrac{N-s+1}{2}\min_{k\in\mathcal{N}}\gamma_k\eta_k = \tfrac{N}{2}\tfrac{4D_X^2}{N} = 2D_X^2,\\
&D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}}\gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2 M_G^2\\
&\le D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}}\frac{D_X^2 M_F^2}{N(M_F+M_G)^2} + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\frac{D_X^2 M_G^2}{N(M_F+M_G)^2}\\
&\le D_X^2 + \tfrac{1}{2}\textstyle\sum_{k=1}^{N}\frac{D_X^2}{N} \le 2D_X^2.
\end{aligned}$$

It then follows from Lemma 2 and Theorem 4 that

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \le \frac{2D_X(M_F+M_G) + \sum_{k\in\mathcal{B}}\frac{D_X M_F^2}{N(M_F+M_G)} + \sum_{k\in\mathcal{N}}\frac{D_X M_G^2}{N(M_F+M_G)}}{\sqrt{N}} \le \frac{4D_X(M_F+M_G)}{\sqrt{N}},$$
$$g(\bar{x}_{N,s}) \le \max_{s\le k\le N}\eta_k = \frac{4D_X(M_F+M_G)}{\sqrt{N}}.$$

■

Corollary 6 *If* $s = \frac{N}{2}$, $\gamma_k = \frac{D_X}{\sqrt{k}(M_F+M_G)}$ *and* $\eta_k = \frac{4D_X(M_F+M_G)}{\sqrt{k}}$, $k = 1, 2, ..., N$, *then*

$$\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \le \frac{4D_X(1+\frac{1}{2}\log 2)(M_F+M_G)}{\sqrt{N}},$$
$$g(\bar{x}_{N,s}) \le \frac{4\sqrt{2}D_X(M_F+M_G)}{\sqrt{N}}.$$

Proof. The proof is similar to that of corollary 4 and hence the details are skipped. ■

In view of Corollaries 5 and 6, the CSA algorithm achieves an $\mathcal{O}(1/\sqrt{N})$ rate of convergence for solving problem (1.1). This convergence rate seems to be unimprovable as it matches the optimal rate of convergence for deterministic convex optimization problems with function constraints [76]. However, to the best of our knowledge, no such complexity bounds have been obtained before for solving stochastic optimization problems with function constraints.

In the Corollary 5 and 6, we established the expected convergence properties over many runs of the CSA algorithm. In the remaining part of this subsection, we are interested in

the large deviation properties for a single run of this method.

First note that by Corollary 6 and the Markov's inequality, we have

$$\text{Prob}\left(f(\bar{x}_{N,s}) - f(x^*) > \lambda_1 \frac{4D_X(1+\frac{1}{2}\log 2)(M_F+M_G)}{\sqrt{N}}\right) < \frac{1}{\lambda_1}, \forall \lambda_1 \geq 0.$$

It then follows that in order to find a solution $\bar{x}_{N,s} \in X$ such that

$$\text{Prob}\left(f(\bar{x}_{N,s}) - f(x^*) \leq \epsilon\right) > 1 - \Lambda,$$

the number of iteration performed by the CSA method can be bounded by

$$\mathcal{O}\left\{\frac{1}{\epsilon^2\Lambda^2}\right\}. \tag{2.19}$$

We will show that this result can be significantly improved if Assumption A1 is augmented by the following "light-tail" assumption, which is satisfied by a wide class of distributions (e.g., Gaussian and t-distribution).

Assumption 2 *For and $x \in X$,*

$$\mathbb{E}[\exp\{\|F'(x,\xi)\|_*^2/M_F^2\}] \leq \exp\{1\}.$$

We first present the following Bernstein inequality that will be used to establish the large-deviation properties of the CSA method (e.g. see [16]). Note that in the sequel, we denote $\xi_{[k]} := \{\xi_1, \ldots, \xi_k\}$.

Lemma 7 *Let $\xi_1, \xi_2, ...$ be a sequence of i.i.d. random variables, and $\xi_t = \xi(\xi_{[t]})$ be deterministic Borel functions of $\xi_{[t]}$ such that $\mathbb{E}[\xi_t] = 0$ a.s. and $\mathbb{E}[\exp\{\xi_t^2/\sigma_t^2\}] \leq \exp\{1\}$ a.s., where $\sigma_t > 0$ are deterministic. Then*

$$\forall \lambda \geq 0 : \text{Prob}\left\{\sum_{t=1}^N \xi_t > \lambda\sqrt{\sum_{t=1}^N \sigma_t^2}\right\} \leq \exp\{-\lambda^2/3\}.$$

Now we are ready to establish the large deviation properties of the CSA algorithm.

Theorem 8 *Under Assumption 2,* $\forall \lambda \geq 0$*,*

$$\text{Prob}\{f(\bar{x}_{N,s}) - f(x^*) \geq K_0 + \lambda K_1\} \leq \exp\{-\lambda\} + \exp\{-\tfrac{\lambda^2}{3}\}, \tag{2.20}$$

where $K_0 = \dfrac{\frac{1}{2}D_X^2 + M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2 + M_G^2 \sum_{k\in\mathcal{N}} \gamma_k^2}{\sum_{k\in\mathcal{B}} \gamma_k}$ *and*

$$K_1 = \frac{M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2 + M_G^2 \sum_{k\in\mathcal{N}} \gamma_k^2 + \sigma\sqrt{\sum_{k\in\mathcal{N}} \gamma_k^2} + M_F D_X \sqrt{\sum_{k\in\mathcal{B}} \gamma_k^2}}{\sum_{k\in\mathcal{B}} \gamma_k}.$$

Proof. Let $F'(x_k, \xi_k) = f'(x_k) + \Delta_k$. It follows from the inequality (2.7) (with $x = x^*$) and the fact $g(x^*) \leq 0$ that

$$\begin{aligned}\textstyle\sum_{k\in\mathcal{N}} \gamma_k \eta_k + (\sum_{k\in\mathcal{B}} \gamma_k)(f(\bar{x}_{N,s}) - f(x^*)) \leq D_X^2 + \textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k, \xi_k)\|_*^2 \\ + \textstyle\sum_{k\in\mathcal{N}} \gamma_k^2 \|g'(x_k)\|_*^2 - \sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^* \rangle.\end{aligned} \tag{2.21}$$

Now we provide probabilistic bounds for $\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k, \xi_k)\|_*^2$ and $\sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^* \rangle$. First, setting $\theta_k = \gamma_k^2 / \sum_{k\in\mathcal{B}} \gamma_k^2$, using the fact that $\mathbb{E}[\exp\{\|F'(x_k, \xi_k)\|_*^2 / M_F^2\}] \leq exp\{1\}$ and Jensens inequality, we have

$$\exp\{\textstyle\sum_{k\in\mathcal{B}} \theta_k (\|F'(x_k, \xi_k)\|_*^2 / M_F^2)\} \leq \sum_{k\in\mathcal{B}} \theta_k \exp\{\|F'(x_k, \xi_k)\|_*^2 / M_F^2\},$$

and hence that

$$\mathbb{E}[\exp\{\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k, \xi_k)\|_*^2 / M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2\}] \leq \exp\{1\}.$$

It then follows from Markov's inequality that $\forall \lambda \geq 0$,

$$\begin{aligned} &\text{Prob}(\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2 > (1+\lambda) M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2) \\ &= \text{Prob}\left(\exp\left\{\frac{\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2}{M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2}\right\} > \exp(1+\lambda)\right) \\ &\leq \tfrac{\exp\{1\}}{\exp\{1+\lambda\}} \leq \exp\{-\lambda\}. \end{aligned} \tag{2.22}$$

Then, let us consider $\sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^* \rangle$. Setting $\beta_k = \gamma_k \langle \Delta_k, x_k - x^* \rangle$ and noting that $\mathbb{E}[\|\Delta_k\|_*^2] \leq (2M_F)^2$, we have

$$\mathbb{E}[\exp\{\beta_k^2/(2M_F\gamma_k D_X)^2\}] \leq \exp\{1\},$$

which, in view of Lemma 7, implies that

$$\text{Prob}\left\{\textstyle\sum_{k\in\mathcal{B}} \beta_k > 2\lambda M_F D_X \sqrt{\sum_{k\in\mathcal{B}} \gamma_k^2}\right\} \leq \exp\{-\lambda^2/3\}. \tag{2.23}$$

Combining (2.22) and (2.23), and rearranging the terms we get (2.20). ∎

Applying the stepsize strategy in Corollary 5 to Theorem 8, then it follows that the number of iterations performed by the CSA method can be bounded by

$$\mathcal{O}\left\{\tfrac{1}{\epsilon^2}(\log\tfrac{1}{\Lambda})^2\right\}.$$

We can see that the above result significantly improves the one in (2.19).

2.2.4 Convergence of CSA for SP with expectation constraints

In this subsection, we focus on the SP problem (1.1)-(1.2) with the expectation constraint. We assume the expectation functions $f(x)$ and $g(x)$, in addition to being well-defined and finite-valued for every $x \in X$, are continuous and convex on X. Throughout this section, we assume the Assumption 2 holds. Moreover, with a little abuse of notation, we make the

following assumption.

Assumption 3 *for and* $x \in X$*,*

$$\mathbb{E}[\exp\{\|G'(x,\xi)\|_*^2/M_G^2\}] \leq \exp\{1\}, \tag{2.24}$$

$$\mathbb{E}[\exp\{(G(x,\xi)-g(x))^2/\sigma^2\}] \leq \exp\{1\}. \tag{2.25}$$

We will use (2.24) and (2.25) to bound the error associated with stochastic subgradient and function value for the constraint g, respectively. As discussed in subsection 2.2, there may exist different ways to simulate the random variable ξ for constraint evaluation, e.g., by generating a J-sized i.i.d. random sample of ξ or its linear transformation $\bar{\xi} = \xi^T x$. However, regardless of the way to simulate the random variable ξ, the light-tail assumption (2.25) holds for the constraint value $G(x,\xi)$. Our goal in this subsection is to show how the sample size (or iteration count) N to compute stochastic subgradients, as well as the sample size J to evaluate the constraint value, will affect the quality of the solutions generated by CSA.

The following result establishes a simple but important recursion about the CSA method for stochastic optimization with expectation constraints.

Proposition 9 *For any* $1 \leq s \leq N$*, we have*

$$\begin{aligned}&\textstyle\sum_{k\in\mathcal{N}} \gamma_k(G(x_k,\xi_k)-G(x,\xi_k)) + \sum_{k\in\mathcal{B}} \gamma_k\langle F'(x_k,\xi_k), x_k - x\rangle \\ &\quad \leq V(x_s,x) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2\|F'(x_k,\xi_k)\|_*^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2\|G'(x_k,\xi_k)\|_*^2,\ \forall x\in X.\end{aligned} \tag{2.26}$$

Proof. For any $s \leq k \leq N$, using Lemma 1, we have

$$V(x_{k+1},x) \leq V(x_k,x) + \gamma_k\langle h_k, x - x_k\rangle + \tfrac{1}{2}\gamma_k^2\|h_k\|_*^2. \tag{2.27}$$

Observe that if $k \in \mathcal{B}$, we have $h_k = F'(x_k, \xi_k)$, and

$$\langle h_k, x_k - x\rangle = \langle F'(x_k, \xi_k), x_k - x\rangle.$$

Moreover, if $k \in \mathcal{N}$, we have $h_k = G'(x_k, \xi_k)$ and

$$\langle h_k, x_k - x\rangle = \langle G'(x_k, \xi_k), x_k - x\rangle \geq G(x_k, \xi_k) - G(x, \xi_k).$$

Summing up the inequalities in (2.27) from $k = s$ to N and using the previous two observations, we obtain

$$\begin{aligned} V(x_{k+1}, x) &\leq V(x_s, x) - \textstyle\sum_{k=s}^{N} \gamma_k \langle h_k, x_k - x\rangle + \frac{1}{2}\sum_{k=s}^{N} \gamma_k^2 \|h_k\|_*^2 \\ &\leq V(x_s, x) - \left[\textstyle\sum_{k\in\mathcal{N}} \gamma_k \langle G'(x_k, \xi_k), x_k - x\rangle + \sum_{k\in\mathcal{B}} \gamma_k \langle F'(x_k, \xi_k), x_k - x\rangle\right] + \frac{1}{2}\sum_{k=s}^{N} \gamma_k^2 \|h \\ &= V(x_s, x) - \left[\textstyle\sum_{k\in\mathcal{N}} \gamma_k (G(x_k, \xi_k) - G(x, \xi_k)) + \sum_{k\in\mathcal{B}} \gamma_k \langle F'(x_k, \xi_k), x_k - x\rangle\right] \\ &\quad + \frac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k, \xi_k)\|_*^2 + \frac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 \|G'(x_k, \xi_k)\|_*^2. \end{aligned} \tag{2.28}$$

Rearranging the terms in above inequality, we obtain (2.26). ∎

Using Proposition 9, we present below a sufficient condition under which the output solution $\bar{x}_{N,s}$ is well-defined.

Lemma 10 *Let x^* be an optimal solution of* (1.1)-(1.2)*. Under Assumption 3, for any given $\lambda > 0$, if*

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k \eta_k > V(x_s, x^*) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 M_F^2 + \frac{1}{2}\sum_{k\in\mathcal{N}} \gamma_k^2 M_G^2 + \frac{\lambda\sigma}{\sqrt{J}} \sum_{k\in\mathcal{N}} \gamma_k, \tag{2.29}$$

where J is the number of random samples to estimate $g(x_k)$ in each iteration, then $\mathcal{B} \neq \emptyset$, i.e., $\bar{x}_{N,s}$ is well-defined. Moreover, we have one of the following two statements holds,

a) Prob$\{|\mathcal{B}| \geq (N - s + 1)/2\} \geq 1 - |\mathcal{N}| \exp\{-\frac{\lambda^2}{3}\}$,

b) $\sum_{k\in\mathcal{B}}\gamma_k\langle f'(x_k), x_k - x^*\rangle \le 0$.

Proof. Taking expectation w.r.t. ξ_k on both sides of (2.26), fixing $x = x^*$ and noting that Assumption 3 implies that $\mathbb{E}[\|G'(x,\xi)\|_*^2] \le M_G^2$, we have

$$\begin{aligned}\sum_{k\in\mathcal{N}}\gamma_k[g(x_k) - g(x^*)] + \sum_{k\in\mathcal{B}}\gamma_k\langle f'(x_k), x_k - x^*\rangle \\ \le V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}}\gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2 M_G^2.\end{aligned} \tag{2.30}$$

If $\sum_{k\in\mathcal{B}}\gamma_k\langle f'(x_k), x_k - x^*\rangle \le 0$, part b) holds. If $\sum_{k\in\mathcal{B}}\gamma_k\langle f'(x_k), x_k - x^*\rangle \ge 0$, we have

$$\sum_{k\in\mathcal{N}}\gamma_k[g(x_k) - g(x^*)] \le V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}}\gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2 M_G^2,$$

which, in view of $g(x^*) \le 0$, implies that

$$\sum_{k\in\mathcal{N}}\gamma_k g(x_k) \le V(x_s, x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}}\gamma_k^2 M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2 M_G^2. \tag{2.31}$$

It follows from (2.4), Assumption 3 and Lemma 7 that, for $k \in \mathcal{N}$, we have $\hat{G}_k > \eta_k$ and $\text{Prob}\{\hat{G}_k \ge g(x_k) + \lambda\sigma/\sqrt{J}\} \le \exp\{-\lambda^2/3\}$, which implies, $\text{Prob}\{g(x_k) \le \eta_k - \lambda\sigma/\sqrt{J}\} \le \exp\{-\lambda^2/3\}$. Therefore,

$$\begin{aligned}&\text{Prob}\{\textstyle\sum_{k\in\mathcal{N}}\gamma_k g(x_k) \le \sum_{k\in\mathcal{N}}\gamma_k\eta_k - \frac{\lambda\sigma}{\sqrt{J}}\sum_{k\in\mathcal{N}}\gamma_k\} \\ &\le \text{Prob}\{\exists k \in \mathcal{N}, \gamma_k g(x_k) \le \eta_k - \tfrac{\lambda\sigma}{\sqrt{J}}\} \le 1 - (1 - \exp\{-\tfrac{\lambda^2}{3}\})^{|\mathcal{N}|} \le |\mathcal{N}|\exp\{-\tfrac{\lambda^2}{3}\}.\end{aligned} \tag{2.32}$$

Combining (2.31) and (2.32), we have

$$\begin{aligned}\text{Prob}\{\textstyle\sum_{k\in\mathcal{N}}\gamma_k\eta_k < V(x_s, x^*) + \frac{1}{2}\sum_{k\in\mathcal{B}}\gamma_k^2 M_F^2 + \frac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2 M_G^2 + \frac{\lambda\sigma}{\sqrt{J}}\sum_{k\in\mathcal{N}}\gamma_k\} \\ \ge 1 - |\mathcal{N}|\exp\{-\tfrac{\lambda^2}{3}\}.\end{aligned}$$

* Suppose that $|\mathcal{B}| < (N - s + 1)/2$, i.e., $|\mathcal{N}| \ge (N - s + 1)/2$. Then, the condition in

(2.29) implies that

$$\sum_{k\in\mathcal{N}}\gamma_k\eta_k \geq \frac{N-s+1}{2}\min_{k\in\mathcal{N}}\gamma_k\eta_k > V(x_s,x^*)+\frac{1}{2}\sum_{k\in\mathcal{B}}\gamma_k^2M_F^2+\frac{1}{2}\sum_{k\in\mathcal{N}}\gamma_k^2M_G^2+\frac{\lambda\sigma}{\sqrt{J}}\sum_{k\in\mathcal{N}}\gamma_k.$$

It then follows from the previous two observations that $\text{Prob}\{|\mathcal{B}| \geq (N-s+1)/2\} \geq 1-|\mathcal{N}|\exp\{-\frac{\lambda^2}{3}\}$. ∎

Now we are ready to establish the large deviation properties of the CSA algorithm.

Theorem 11 *Suppose that Assumptions 2 and 3 hold.*

a) *For any given partition $\mathcal{B}$ and $\mathcal{N}$ of $I=\{s,\ldots,N\}$, we have, $\forall\lambda\geq 0$,*

$$\text{Prob}\{f(\bar{x}_{N,s})-f(x^*)\geq K_0+\lambda K_1\}\leq 2\exp\{-\lambda\}+(|\mathcal{N}|+2)\exp\{-\tfrac{\lambda^2}{3}\}, \tag{2.33}$$

$$\text{Prob}\left\{g(\bar{x}_{N,s})\geq\left(\sum_{k\in\mathcal{B}}\gamma_k\right)^{-1}\left(\sum_{k\in\mathcal{B}}\gamma_k\eta_k\right)+\frac{\lambda\sigma}{\sqrt{J}}\right\}\leq|\mathcal{B}|\exp\{-\lambda^2/3\}, \tag{2.34}$$

where $K_0=\left(\sum_{k\in\mathcal{B}}\gamma_k\right)^{-1}\left(D_X^2+\frac{M_F^2}{2}\sum_{k\in\mathcal{B}}\gamma_k^2+\frac{M_G^2}{2}\sum_{k\in\mathcal{N}}\gamma_k^2\right)$ *and*
$K_1=\left(\sum_{k\in\mathcal{B}}\gamma_k\right)^{-1}(\frac{M_F^2}{2}\sum_{k\in\mathcal{B}}\gamma_k^2+\frac{M_G^2}{2}\sum_{k\in\mathcal{N}}\gamma_k^2+2\sigma\sqrt{\sum_{k\in\mathcal{N}}\gamma_k^2}$
$+2M_FD_X\sqrt{\sum_{k\in\mathcal{B}}\gamma_k^2}+\frac{\sigma}{\sqrt{J}}\sum_{k\in\mathcal{N}}\gamma_k)$.

b) *For any $\Lambda\in(0,1)$, if we choose λ such that $N\exp\{-\lambda^2/3\}\leq\Lambda$ and set*

$$\begin{aligned}
&s=1,\ \gamma_k=\tfrac{D_X}{\sqrt{N}M},\ \eta_k=\tfrac{4MD_X}{\sqrt{N}}+\tfrac{2\lambda\sigma}{\sqrt{J}},\\
&N=\max\{\tfrac{2C}{\epsilon^2}(\log\tfrac{4}{\Lambda})^2,\tfrac{6C}{\epsilon^2}\log\tfrac{18D_X^2M^2}{\epsilon^2\Lambda},\tfrac{64M^2D_X^2}{\vartheta^2}\},\\
&J=\max\{\tfrac{8\sigma^2}{\epsilon^2}(\log\tfrac{4}{\Lambda})^2,\tfrac{24\sigma^2}{\epsilon^2}\log\tfrac{18D_X^2M^2}{\epsilon^2\Lambda},\tfrac{36\sigma^2}{\vartheta^2}\log\tfrac{1}{\Lambda^3},\tfrac{36\sigma^2}{\vartheta^2}\log\tfrac{18D_X^2M^2}{\epsilon^2\Lambda}\},
\end{aligned} \tag{2.35}$$

where $M=\max\{M_F,M_G\}$ and $C=\max\{9D_X^2M^2,\ 4\sigma^2\}$, then we have

$$\text{Prob}\{g(\bar{x}_{N,s})\leq\vartheta\}\geq 1-\Lambda \text{ and } \text{Prob}\{f(\bar{x}_{N,s})-f(x^*)\leq\epsilon\}\geq(1-\Lambda)^2. \tag{2.36}$$

Proof. Let us first show part a) holds. Observe that the constraint evaluation and hence

the partition of $\mathcal{B}$ and $\mathcal{N}$ is independent of the trajectory. Let $G(x,\xi_k) = g(x) + \delta_k$ and $F'(x_k,\xi_k) = f'(x_k) + \Delta_k$. It follows from the inequality (2.26) (with $x = x^*$) and the fact $g(x^*) \leq 0$ that

$$\begin{aligned}\textstyle\sum_{k\in\mathcal{N}} \gamma_k g(x_k) + (\sum_{k\in\mathcal{B}} \gamma_k)(f(\bar{x}_{N,s}) - f(x^*)) \leq V(x_s,x^*) + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2 \\ + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{N}} \gamma_k^2 \|G'(x_k,\xi_k)\|_*^2 + 2\sum_{k\in\mathcal{N}} \gamma_k\delta_k - \sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^*\rangle. \quad (2.37)\end{aligned}$$

Now we provide probabilistic bounds for $\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2$, $\sum_{k\in\mathcal{N}} \gamma_k^2 \|G'(x_k,\xi_k)\|_*^2$, $\sum_{k\in\mathcal{N}} \gamma_k\delta_k$ and $\sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^*\rangle$. First, setting $\theta_k = \gamma_k^2 / \sum_{k\in\mathcal{B}} \gamma_k^2$, using the fact that $\mathbb{E}[\exp\{\|F'(x_k,\xi_k)\|_*^2 / M_F^2\}] \leq exp\{1\}$ and Jensens inequality, we have $\exp\{\sum_{k\in\mathcal{B}} \theta_k (\|F'(x_k,\xi_k)\|_*^2 / M_F^2)\} \leq \sum_{k\in\mathcal{B}} \theta_k \exp\{\|F'(x_k,\xi_k)\|_*^2 / M_F^2\}$, and hence that $\mathbb{E}[\exp\{\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2 / M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2\}] \leq \exp\{1\}$. It then follows from Markov's inequality that $\forall \lambda \geq 0$,

$$\begin{aligned}&\text{Prob}(\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2 > (1+\lambda) M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2) \\ &= \text{Prob}\left(\exp\left\{\frac{\sum_{k\in\mathcal{B}} \gamma_k^2 \|F'(x_k,\xi_k)\|_*^2}{M_F^2 \sum_{k\in\mathcal{B}} \gamma_k^2}\right\} > \exp(1+\lambda)\right) \leq \tfrac{\exp\{1\}}{\exp\{1+\lambda\}} \leq \exp\{-\lambda\}.\end{aligned} \quad (2.38)$$

Similarly, we have

$$\text{Prob}\left(\textstyle\sum_{k\in\mathcal{N}} \gamma_k^2 \|G'(x_k,\xi_k)\|_*^2 > (1+\lambda) M_G^2 \sum_{k\in\mathcal{N}} \gamma_k^2\right) \leq \exp\{-\lambda\}. \quad (2.39)$$

Second, for $\sum_{k\in\mathcal{N}} \gamma_k\delta_k$, setting $\iota_k = \gamma_k / \sum_{k\in\mathcal{B}} \gamma_k$, and noting that $\mathbb{E}[\delta_k] = 0$ and $\mathbb{E}[\exp\{\delta_k^2/\sigma^2\}] \leq \exp\{1\}$, we obtain $\mathbb{E}[\iota_k\delta_k] = 0$, $\mathbb{E}[\exp\{\iota_k^2\delta_k^2/\xi_k^2\sigma^2\}] \leq \exp\{1\}$. By lemma 7, we have

$$\text{Prob}\left\{\textstyle\sum_{k\in\mathcal{N}} \gamma_k\delta_k > \lambda\sigma\sqrt{\sum_{k\in\mathcal{N}} \gamma_k^2}\right\} \leq \exp\{-\lambda^2/3\}. \quad (2.40)$$

Lastly, let us consider $\sum_{k\in\mathcal{B}} \gamma_k \langle \Delta_k, x_k - x^*\rangle$. Setting $\beta_k = \gamma_k \langle \Delta_k, x_k - x^*\rangle$ and noting

that $\mathbb{E}[\|\Delta_k\|_*^2] \leq (2M_F)^2$, we have $\mathbb{E}[\exp\{\beta_k^2/(2M_F\gamma_k D_X)^2\}] \leq \exp\{1\}$, which, in view of Lemma 7, implies that

$$\text{Prob}\left\{\sum_{k\in\mathcal{B}}\beta_k > 2\lambda M_F D_X\sqrt{\sum_{k\in\mathcal{B}}\gamma_k^2}\right\} \leq \exp\{-\lambda^2/3\}. \tag{2.41}$$

Combining (2.38),(2.39), (2.40), (2.41) and (2.32), and rearranging the terms we get (2.33). Let us show that (2.34) holds. Clearly, by the convexity of $g(\cdot)$ and definition of $\bar{x}_{N,s}$, we have

$$g(\bar{x}_{N,s}) = g(\textstyle\sum_{k\in\mathcal{B}}\iota_k x_k) \leq \left(\sum_{k\in\mathcal{B}}\gamma_k\right)^{-1}\sum_{k\in\mathcal{B}}\gamma_k g(x_k).$$

Using this observation and an argument similar to the proof of (2.32), we obtain (2.34).

Then, let us show part b) holds. First, easily observe that condition (2.29) holds by using the selection of s, $\{\gamma_k\}$ and $\{\eta_k\}$. From Lemma 10, we have either one of the following two statements holds,

a) $\text{Prob}\{|\mathcal{B}| \geq (N-s+1)/2\} \geq 1 - |\mathcal{N}|\exp\{-\frac{\lambda^2}{3}\} \geq 1-\Lambda$,

b) $\sum_{k\in\mathcal{B}}\gamma_k\langle f'(x_k), x_k - x^*\rangle \leq 0$, which, in view of the convexity of f, implies, $f(\bar{x}_{N,s}) - f(x^*) \leq 0$.

Also, from (2.34) and (2.35), we have

$$\text{Prob}\left\{g(\bar{x}_{N,s}) \geq \tfrac{4MD_X}{\sqrt{N}} + \tfrac{3\lambda\sigma}{\sqrt{J}}\right\} \leq |\mathcal{B}|\exp\{-\lambda^2/3\}$$
$$\text{Prob}\left\{g(\bar{x}_{N,s}) \geq \vartheta\right\} \leq \Lambda.$$

Moreover, conditional on that $|\mathcal{B}| \geq N/2$, it then follows Theorem 11 and (2.35) that

$$\text{Prob}\left\{f(\bar{x}_{N,s}) - f(x^*) \geq \tfrac{3D_X M}{\sqrt{N}} + \lambda(\tfrac{3\sqrt{2}MD_X}{\sqrt{N}} + \tfrac{2\sqrt{2}\sigma}{\sqrt{N}} + \tfrac{\sqrt{2}\sigma}{\sqrt{J}})\right\}$$
$$\leq 2\exp\{-\lambda\} + (|\mathcal{N}|+2)\exp\{-\tfrac{\lambda^2}{3}\},$$

By implementing the selection of N and J, we have (2.36). ■

In view of Theorem 11, the complexity in terms of the number of iterations N of the CSA algorithm can be bounded by $\mathcal{O}(\max\{\frac{1}{\epsilon^2}(\log\frac{1}{\Lambda})^2, \frac{1}{\vartheta^2}\})$, and the sample size J for estimating constraint in every iteration of the CSA algorithm can be bounded by $\mathcal{O}(\max\{\frac{1}{\epsilon^2}(\log\frac{1}{\Lambda})^2, \frac{1}{\vartheta^2}\log\frac{1}{\Lambda^3}\})$ for solving problem (1.1)-(1.2).

2.2.5 Strongly convex objective and strongly convex constraints

In this subsection, we are interested in establishing the convergence of the CSA algorithm applied to strongly convex problems. More specifically, we assume that the objective function F and constraint function g in problem (1.1), where g is given in the form of function constraint, are both strongly convex w.r.t. x, i.e., $\exists \mu_F > 0$ and $\mu_G > 0$ s.t.

$$F(x_1,\xi) \geq F(x_2,\xi) + \langle F'(x_2,\xi), x_1 - x_2\rangle + \tfrac{\mu_F}{2}\|x_1 - x_2\|^2, \forall x_1, x_2 \in X,$$
$$g(x_1) \geq g(x_2) + \langle g'(x_2), x_1 - x_2\rangle + \tfrac{\mu_G}{2}\|x_1 - x_2\|^2, \forall x_1, x_2 \in X.$$

For the sake of simplicity, we focus on the case when the constraint function g can be evaluated exactly (i.e., $\hat{G}_k = g'(x_k)$). However, expectation constraints can be dealt with using similar techniques discussed in Section 2.2.4.

In order to estimate the convergent rate of the CSA algorithm for solving strongly convex problems, we need to assume that the prox-function $V_X(\cdot,\cdot)$ and $V_Y(\cdot,\cdot)$ satisfies a quadratic growth condition

$$V_X(z,x) \leq \tfrac{Q}{2}\|z - x\|^2, \forall z, x \in X \text{ and } V_Y(z,y) \leq \tfrac{Q}{2}\|z - y\|^2, \forall z, y \in Y. \tag{2.42}$$

Moreover, letting γ_k be the stepsizes used in the CSA method, and denoting

$$a_k = \begin{cases} \frac{\mu_F\gamma_k}{Q}, & k \in \mathcal{B}, \\ \frac{\mu_G\gamma_k}{Q}, & k \in \mathcal{N}, \end{cases} \quad A_k = \begin{cases} 1, & k = 1, \\ (1-a_k)A_{k-1}, & k \geq 2, \end{cases} \text{ and } \rho_k = \tfrac{\gamma_k}{A_k},$$

we define

$$\bar{x}_{N,s} = \frac{\sum_{k\in\mathcal{B}} \rho_k x_k}{\sum_{k\in\mathcal{B}} \rho_k} \tag{2.43}$$

as the output of Algorithm 1.

The following simple result will be used in the convergence analysis of the CSA method.

Lemma 12 *If* $a_k \in (0,1]$, *k = 0,1,2,...,* $A_k > 0, \forall k \geq 1$, *and* $\{\Delta_k\}$ *satisfies*

$$\Delta_{k+1} \leq (1-a_k)\Delta_k + B_k, \forall k \geq 1,$$

then we have

$$\frac{\Delta_{k+1}}{A_k} \leq (1-a_1)\Delta_1 + \sum_{i=1}^{k} \frac{B_i}{A_i}.$$

Below we provide an important recursion about CSA applied to strongly convex problems. This result differs from Proposition 2 for the general convex case in that we use different weight ρ_k rather than γ_k.

Proposition 13 *For any* $1 \leq s \leq N$, *we have*

$$\begin{aligned}\sum_{k\in\mathcal{N}} \rho_k(\eta_k - G(x,\xi_k)) + \sum_{k\in\mathcal{B}} \rho_k[F(x_k,\xi_k) - F(x,\xi_k)] &\leq (1-a_s)D_X^2 \\ &+ \tfrac{1}{2}\sum_{k\in\mathcal{B}} \rho_k\gamma_k\|F'(x_k,\xi_k)\|_*^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \rho_k\gamma_k\|g'(x_k)\|_*^2. \end{aligned} \tag{2.44}$$

Proof. Consider the iteration k, $\forall s \leq k \leq N$. If $k \in \mathcal{B}$, by Lemma 1 and the strong convexity of $F(x,\xi)$, we have

$$\begin{aligned} V(x_{k+1},x) &\leq V(x_k,x) - \gamma_k\langle h_k, x_k - x\rangle + \tfrac{1}{2}\gamma_k^2\|F'(x_k,\xi_k)\|_*^2 \\ &= V(x_k,x) - \gamma_k\langle F'(x_k,\xi_k), x_k - x\rangle + \tfrac{1}{2}\gamma_k^2\|F'(x_k,\xi_k)\|_*^2 \\ &\leq V(x_k,x) - \gamma_k\left[F(x_k,\xi_k) - F(x,\xi_k) + \tfrac{\mu_F}{2}\|x_k - x\|^2\right] + \tfrac{1}{2}\gamma_k^2\|F'(x_k,\xi_k)\|_*^2 \\ &\leq \left(1 - \tfrac{\mu_F\gamma_k}{Q}\right)V(x_k,x) - \gamma_k[F(x_k,\xi_k) - F(x,\xi_k)] + \tfrac{1}{2}\gamma_k^2\|F'(x_k,\xi_k)\|_*^2. \end{aligned}$$

Similarly for $k \in \mathcal{N}$, using Lemma 1 and the strong convexity of $g(x)$, we have

$$\begin{aligned} V(x_{k+1}, x) &\leq V(x_k, x) - \gamma_k \langle h_k, x_k - x \rangle + \tfrac{1}{2}\gamma_k^2 \|g'(x_k)\|_*^2 \\ &= V(x_k, x) - \gamma_k \langle g'(x_k), x_k - x \rangle + \tfrac{1}{2}\gamma_k^2 \|g'(x_k)\|_*^2 \\ &\leq V(x_k, x) - \gamma_k \left[(g(x_k) - g(x)) + \tfrac{\mu_G}{2}\|x_k - x\|^2 \right] + \tfrac{1}{2}\gamma_k^2 \|g'(x_k)\|_*^2 \\ &\leq \left(1 - \tfrac{\mu_G \gamma_k}{Q}\right) V(x_k, x) - \gamma_k (\eta_k - g(x)) + \tfrac{1}{2}\gamma_k^2 \|g'(x_k)\|_*^2. \end{aligned}$$

Summing up these inequalities for $s \leq k \leq N$ and using Lemma 12, we have

$$\begin{aligned} \tfrac{V(x_{N+1},x)}{A_N} \leq (1 - a_s)\, V(x_s, x) - \left[\textstyle\sum_{k\in\mathcal{N}} \tfrac{\gamma_k}{A_k}(\eta_k - g(x)) + \sum_{k\in\mathcal{B}} \tfrac{\gamma_k}{A_k}[F(x_k, \xi_k) - F(x, \xi_k)] \right] \\ + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{N}} \tfrac{\gamma_k^2}{A_k}\|g'(x_k)\|_*^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \tfrac{\gamma_k^2}{A_k}\|F'(x_k, \xi_k)\|_*^2, \end{aligned}$$

Using the fact $V(x_{N+1}, x)/A_N \geq 0$ and the definition of ρ_k, and rearranging the terms in the above inequality, we obtain (2.44). ∎

Lemma 14 below provides a sufficient condition which guarantees $\bar{x}_{N,s}$ to be well-defined.

Lemma 14 *Let x^* be the optimal solution of* (1.1)*. If*

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \rho_k \eta_k > (1 - a_s) D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{N}} \rho_k \gamma_k M_G^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \rho_k \gamma_k M_F^2, \tag{2.45}$$

then $\mathcal{B} \neq \emptyset$ and hence $\bar{x}_{N,s}$ is well-defined. Moreover, we have one of the following two statements holds,

a) $|\mathcal{B}| \geq (N - s + 1)/2$,

b) $\sum_{k\in\mathcal{B}} \rho_k [f(x_k) - f(x^*)] \leq 0$.

Proof. The proof of this result is similar to that of Lemma 2 and hence the details are skipped. ∎

With the help of Proposition 13, we are ready to establish the main convergence prop-

erties of the CSA method for solving strongly convex problems.

Theorem 15 *Suppose that* $\{\gamma_k\}$ *and* $\{\eta_k\}$ *in the CSA algorithm are chosen such that* (2.45) *holds. Then for any* $1 \leq s \leq N$*, we have*

$$\begin{aligned}\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \leq & ((N-s+1)\min_{s\leq k\leq N}\rho_k)^{-1} \\ & \left(2(1-a_s)D_X^2 + \textstyle\sum_{k\in\mathcal{B}}\rho_k\gamma_k M_F^2 + \sum_{k\in\mathcal{N}}\rho_k\gamma_k M_G^2\right), \qquad (2.46)\\ g(\bar{x}_{N,s}) \leq & (\textstyle\sum_{k\in\mathcal{B}}\rho_k)^{-1}(\sum_{k\in\mathcal{B}}\rho_k\eta_k). \qquad (2.47)\end{aligned}$$

Proof. Taking expectation w.r.t. $\xi_i, 1 \leq i \leq k$, on both sides of (2.44) (with $x = x^*$) and using Assumption 1, we have

$$\begin{aligned}&\textstyle\sum_{k\in\mathcal{N}}\rho_k(\eta_k - g(x^*)) + \sum_{k\in\mathcal{B}}\rho_k\mathbb{E}[f(x_k) - f(x^*)] \\ &\leq (1-a_s)D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}}\rho_k\gamma_k M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}}\rho_k\gamma_k M_G^2.\end{aligned}$$

(2.46) then immediately follows from the above inequality, (2.43), the convexity of f and the fact that $g(x^*) \leq 0$. Moreover, (2.47) follows similarly to (2.18). ∎

Below we provide a stepsize policy of s, γ_k and η_k in order to achieve the optimal rate of convergence for solving strongly convex problems.

Corollary 16 *Let* $s = \frac{N}{2}$, $\gamma_k = \begin{cases} \frac{2Q}{\mu_F(k+1)}, & \text{if } k \in \mathcal{B}; \\ \frac{2Q}{\mu_G(k+1)}, & \text{if } k \in \mathcal{N}, \end{cases}$, $\eta_k = \frac{2\mu_G Q}{k}\left(\frac{2D_X^2}{k} + \max\left\{\frac{M_F^2}{\mu_F^2}, \frac{M_G^2}{\mu_G^2}\right\}\right)$, *then we have*

$$\begin{aligned}\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] &\leq \tfrac{8\mu_F D_X^2}{N^2 Q} + \tfrac{4\mu_F Q}{N}\max\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\}, \\ g(\bar{x}_{N,s}) &\leq \tfrac{16\mu_G Q D_X^2}{N^2} + \tfrac{4\mu_G Q}{N}\max\left\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\right\}.\end{aligned}$$

Proof. Based on our selection of s, γ_k, η_k and the definition of a_k, A_k and ρ_k, we have

$$a_k = \tfrac{2}{k+1},\ A_k = \prod_{i=2}^{k}(1-a_i) = \tfrac{2}{k(k+1)},\ \rho_k = \begin{cases} \frac{kQ}{\mu_F}, & \text{if } k \in \mathcal{B}; \\ \frac{kQ}{\mu_G}, & \text{if } k \in \mathcal{N}, \end{cases}$$

For $\forall s \le k \le N$, by the definition of s, γ_k and η_k, we have

$$\begin{aligned}(1-a_s)V(x_s,x) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{N}} \rho_k\gamma_k M_G^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \rho_k\gamma_k M_F^2 \\ \le D_X^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \frac{\gamma_k^2}{A_k} M_F^2 + \tfrac{1}{2}\sum_{k\in\mathcal{N}} \frac{\gamma_k^2}{A_k} M_G^2 \\ \le D_X^2 + Q^2(|\mathcal{B}|\tfrac{M_F^2}{\mu_F^2} + |\mathcal{N}|\tfrac{M_G^2}{\mu_G^2}) \le D_X^2 + \tfrac{Q^2N}{2}\max\left\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\right\},\end{aligned}$$

$$\tfrac{N-s+1}{2}\min_{k\in\mathcal{N}} \rho_k\eta_k = \tfrac{N}{4}\min_{k\in\mathcal{N}} \tfrac{kQ}{\mu_G}\tfrac{2\mu_G Q}{k}\left(\tfrac{2D_X^2}{k} + \max\left\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\right\}\right) \ge D_X^2 + \tfrac{Q^2N}{2}\max\left\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\right\}.$$

Combining the above two inequalities, we can easily see that condition (2.45) holds. It then follows from Theorem 15 that

$$\begin{aligned}&\mathbb{E}[f(\bar{x}_{N,s}) - f(x^*)] \\ &\le ((N-s+1)\min_{s\le k\le N}\rho_k)^{-1}\left(2(1-a_s)D_X^2 + \textstyle\sum_{k\in\mathcal{B}}\rho_k\gamma_k M_F^2 + \sum_{k\in\mathcal{N}}\rho_k\gamma_k M_G^2\right) \\ &\le \tfrac{8\mu_F D_X^2}{N^2Q} + \tfrac{4\mu_F Q}{N}\max\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\}, \\ &g(\bar{x}_{N,s}) \le (\textstyle\sum_{k\in\mathcal{B}}\rho_k)^{-1}(\sum_{k\in\mathcal{B}}\rho_k\eta_k) \le \tfrac{16\mu_G Q D_X^2}{N^2} + \tfrac{4\mu_G Q}{N}\max\left\{\tfrac{M_F^2}{\mu_F^2}, \tfrac{M_G^2}{\mu_G^2}\right\}.\end{aligned}$$

■

In view of Corollary 16, the CSA algorithm can achieve the optimal rate of convergence for strongly convex optimization with strongly convex constraints. To the best of our knowledge, this is the first time such a complexity result is obtained in the literature and this result is new also for the deterministic setting.

2.3 Expectation constraints over problem parameters

In this section, we are interested in solving a class of parameterized stochastic optimization problems whose parameters are defined by expectation constraints as described in (1.4)-(1.5), under the assumption that such a pair of solutions satisfying (1.4)-(1.5) exists.

Our goal in this section is to present a variant of the CSA algorithm to approximately

solve problem (1.4)-(1.5) and establish its convergence properties. More specifically, we discuss this variant of the CSA algorithm when applied to the parameterized stochastic optimization problem in (1.4)-(1.5) and then consider a modified problem by imposing certain strong convexity assumptions to the function $\Phi(x, y, \zeta)$ w.r.t. y and $G(x, \xi)$ w.r.t. x in Subsections 4.1 and 4.2, respectively. In Subsection 4.3, we discuss some large deviation properties for the variant of the CSA method for the problem defined by (1.4)-(1.5).

2.3.1 Stochastic optimization with parameter feasibility constraints

Given tolerance $\eta > 0$ and target accuracy $\epsilon > 0$, we will present a variant of the CSA algorithm, namely cooperative stochastic parameter approximation (CSPA), to find a pair of approximate solutions $(\bar{x}, \bar{y}) \in X \times Y$ s.t. $\mathbb{E}[g(\bar{x})] \leq \eta$ and $\mathbb{E}[\phi(\bar{x}, \bar{y}) - \phi(\bar{x}, y)] \leq \epsilon, \ \forall y \in Y$, in this subsection. Before we describe the CSPA method, we need slightly modify Assumption 1.

Assumption 4 *For any $x \in X$ and $y \in Y$,*

$$\mathbb{E}[\|\Phi'(x, y, \zeta)\|_*^2] \leq M_\Phi^2 \quad \textit{and} \quad \mathbb{E}[\|G'(x, \xi)\|_*^2] \leq M_G^2,$$

where $\Phi'(x, y, \zeta) \in \partial_y \Phi(x, y, \zeta)$ and $G'(x, \xi) \in \partial_x G(x, \xi)$.

We will also discuss the convergent properties under the light-tail assumptions as follows.

Assumption 5

$$\begin{aligned}
\mathbb{E}[\exp\{\|\Phi'(x, y, \zeta)\|_*^2 / M_\Phi^2\}] &\leq \exp\{1\}, \\
\mathbb{E}[\exp\{(\Phi(x, y, \zeta) - \phi(x, y))^2 / \sigma^2\}] &\leq \exp\{1\}, \\
\mathbb{E}[\exp\{(G(x, \xi) - g(x))^2 / \sigma^2\}] &\leq \exp\{1\}.
\end{aligned}$$

We assume that the distance generating functions $\omega_X : X \mapsto \mathbb{R}$ and $\omega_Y : Y \mapsto \mathbb{R}$ are strongly convex with modulus 1 w.r.t. given norms in $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively, and that

their associated prox-mappings $P_{x,X}$ and $P_{y,Y}$ (see (2.1)) are easily computable.

We make the following modifications to the CSA method in Section 2.1 in order to apply it to solve problem (1.4)-(1.5). Firstly, we still check the solution (x_k, y_k) to see whether x_k violates the condition $\sum_{i=1}^k \gamma_i G(x_i, \xi_i)/\sum_{i=1}^k \gamma_i \leq \eta_k$. If so, we set the search direction as $G'(x_k, \xi_k)$ to update x_k, while keeping y_k intact. Otherwise, we only update y_k along the direction $\Phi'(\bar{x}_k, y_k, \zeta_k)$. Secondly, we define the output as a randomly selected $(\bar{x}_k, y_k)$ according to a certain probability distribution instead of the ergodic mean of $\{(\bar{x}_k, y_k)\}$, where $\bar{x}_k$ denotes the average of $\{x_k\}$ (see (2.1)). Since we are solving a coupled optimization and feasibility problem, each iteration of our algorithm only updates either y_k or x_k and requires the computation of either Φ' or G' depending on whether $\sum_{i=1}^k \gamma_i G(x_i, \xi_i)/\sum_{i=1}^k \gamma_i \leq \eta_k$. This differs from the SA method used in Jiang and Shanbhag [79] that requires two projection steps and the computation of two subgradients at each iteration to solve a different parameterized stochastic optimization problem.

Algorithm 2 The cooperative stochastic parameter approximation method

Input: initial point (x_1, y_1), stepsize $\{\gamma_k\}$, tolerance $\{\eta_k\}$, number of iterations N, $\tau(1) = 1$.

for k=1,2,...,N

if $\sum_{i=1}^{\tau(k)} \gamma_i G(x_i, \xi_i)/\sum_{i=1}^{\tau(k)} \gamma_i \leq \eta_k$

$$y_{k+1} = P_{y_k,Y}(\gamma_k \Phi'(\bar{x}_k, y_k, \zeta_k)),\ \tau(k+1) = \tau(k),\ \text{where } \bar{x}_k = \textstyle\sum_{i=1}^{\tau(k)} \gamma_i x_i/\sum_{i=1}^{\tau(k)} \gamma_i; \tag{2.1}$$

else

$$l = \tau(k), x_{l+1} = P_{x_l,X}(\gamma_l G'(x_l, \xi_l)), y_{k+1} = y_k,\ \tau(k+1) = \tau(k) + 1. \tag{2.2}$$

end if

end for

Output: Set $\mathcal{B} := \{s \leq k \leq N | \sum_{i=1}^{\tau(k)} \gamma_i G(x_i, \xi_i)/\sum_{i=1}^{\tau(k)} \gamma_i \leq \eta_k\}$ for some $1 \leq s \leq N$, and define the output $(\bar{x}_R, y_R)$, where R is randomly chosen according to

$$\text{Prob}\{R = k\} = \frac{\gamma_k}{\sum_{k\in\mathcal{B}} \gamma_k}, k \in \mathcal{B}. \tag{2.3}$$

With a little abuse of notation, we still use $\mathcal{B}$ to represent the set $\{s \leq k \leq N | \sum_{i=1}^{\tau(k)} \gamma_i G(x_i, \xi_i)/\sum_{i=1}^{\tau(k)} \gamma_i \leq \eta_k\}$, $I = \{s, \ldots, N\}$, and $\mathcal{N} = I \setminus \mathcal{B}$. The

following result mimics Proposition 2.

Proposition 17 *For any $1 \le s \le N$, we have*

$$\sum_{k\in\mathcal{B}} \gamma_k \langle \Phi'(\bar{x}_k, y_k, \zeta_k), y_k - y\rangle \le D_Y^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2, \ \forall y \in Y, \quad (2.4)$$

$$\sum_{i=\tau(s)}^{\tau(N)} \gamma_i [G(x_i, \xi_i) - G(x, \xi_i)] \le D_X^2 + \tfrac{1}{2}\sum_{i=\tau(s)}^{\tau(N)} \gamma_i^2 \|G'(x_i, \xi_i)\|_*^2, \ \forall x \in X, \quad (2.5)$$

where $D_X \equiv D_{X,w_x}$ and $D_Y \equiv D_{Y,w_y}$ are defined as in (2.3)*.*

Proof. By Lemma 1, if $k \in \mathcal{B}$,

$$V(y_{k+1}, y) \le V(y_k, y) + \gamma_k \langle \Phi'(\bar{x}_k, y_k, \zeta_k), y - y_k\rangle + \tfrac{1}{2}\gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2.$$

Also note that $V(y_{k+1}, y) = V(y_k, y)$ for $k \in \mathcal{N}$. Summing up these relations for $k \in \mathcal{B}\cup\mathcal{N}$ and using the fact that $V(y_s, y) \le D_Y^2$, we have

$$\begin{aligned} V(y_{N+1}, y) &\le V(y_s, y) + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2 - \sum_{k\in\mathcal{B}} \gamma_k \langle \Phi'(\bar{x}_k, y_k, \zeta_k), y_k - y\rangle \\ &\le D_Y^2 + \tfrac{1}{2}\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2 - \sum_{k\in\mathcal{B}} \gamma_k \langle \Phi'(\bar{x}_k, y_k, \zeta_k), y_k - y\rangle. \end{aligned} \quad (2.6)$$

Similarly for $\tau(s) \le i \le \tau(N)$, we have

$$V(x_{i+1}, x) \le V(x_i, x) + \gamma_i \langle G'(x_i, \xi_i), x - x_i\rangle + \tfrac{1}{2}\gamma_i^2 \|G'(x_i, \xi_i)\|_*^2.$$

Summing up these relations for $\tau(s) \le i \le \tau(N)$ and using the fact that $V(x_{\tau(s)}, x) \le D_X^2$, we obtain

$$V(x_{\tau(N)+1}, x) \le D_X^2 + \textstyle\sum_{i=\tau(s)}^{\tau(N)} \gamma_i^2 \|G'(x_i, \xi_i)\|_*^2 - \sum_{i=\tau(s)}^{\tau(N)} (G(x_i, \xi_i) - G(x, \xi_i)). \quad (2.7)$$

Using the facts $V(y_{N+1}, y) \ge 0$ and $V(x_{\tau(N)+1}, x) \ge 0$, and rearranging the terms in (2.6)

and (2.7), we then obtain (2.4) and (2.5), respectively. ■

The following result provides a sufficient condition under which $(\bar{x}_R, y_R)$ is well-defined.

Lemma 18 *The following statements holds.*

a) *Under Assumption 4, if for any $\lambda > 0$, we have*

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k\eta_k > D_X^2 + \lambda \tfrac{M_G^2}{2} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2, \tag{2.8}$$

then Prob$\{|\mathcal{B}| \geq \frac{N-s+1}{2}\} \geq 1 - 1/\lambda$.

b) *Under Assumption 5, if for any $\lambda > 0$, we have*

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k\eta_k > D_X^2 + (1+\lambda) \tfrac{M_G^2}{2} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2 + \lambda\sigma \sqrt{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2}, \tag{2.9}$$

then Prob$\{|\mathcal{B}| \geq \frac{N-s+1}{2}\} \geq 1 - 2\exp\{-\frac{\lambda^2}{3}\}$.

Proof. First let us show part a), set $\delta_k = G(x^*, \xi_k) - g(x^*)$, it follows from (2.5) and fixing $x = x^*$ that

$$\textstyle\sum_{i=\tau(s)}^{\tau(N)} \gamma_i G(x_i, \xi_i) - \sum_{i=\tau(s)}^{\tau(N)} \gamma_i g(x^*) \leq D_X^2 + \frac{1}{2} \sum_{i=\tau(s)}^{\tau(N)} \gamma_i^2 \|G'(x_i, \xi_i)\|_*^2 + \sum_{i=\tau(s)}^{\tau(N)} \gamma_i \delta_i.$$

For contradiction, suppose that $|\mathcal{B}| < \frac{N-s+1}{2}$, i.e., $\tau(N) - \tau(s) = |\mathcal{N}| \geq \frac{N-s+1}{2}$. The above relation, in view of $g(x^*) \leq 0$ and the fact $\sum_{i=\tau(s)}^{\tau(N)} \gamma_i G(x_i, \xi_i) \geq \eta_{\tau(N)} \sum_{i=\tau(s)}^{\tau(N)} \gamma_i$, implies that

$$\tfrac{N-s+1}{2} \min_{k\in\mathcal{N}} \gamma_k\eta_k \leq \eta_{\tau(N)} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \eta_k \leq D_X^2 + \frac{1}{2} \sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2 \|G'(x_k, \xi_k)\|_*^2 + \sum_{k=\tau(s)}^{\tau(N)} \gamma_k \delta_k.$$

Under Assumption 4, for any $\lambda > 0$, taking expectation on both sides and using Markov's

inequality, we have

$$\text{Prob}\{\tfrac{N-s+1}{2}\min_{k\in\mathcal{N}}\gamma_k\eta_k \le D_X^2+\lambda\tfrac{M_G^2}{2}\textstyle\sum_{k=\tau(s)}^{\tau(N)}\gamma_k^2\}\ge 1-1/\lambda.$$

Hence, part a) holds. Similarly we can show part b), and the details are skipped. ■

Theorem 19 summarizes the main convergence properties of Algorithm 2 applied to problem (1.4)-(1.5).

Theorem 19 *The following statements holds for the CSPA algorithm.*

a) *Under Assumption 4, we have,* $\forall\lambda>0$*,*

$$\mathbb{E}[\phi(\bar{x}_R,y_R)-\phi(\bar{x}_R,y^*(\bar{x}_R))]\le\tfrac{2D_Y^2+M_\Phi^2\sum_{k\in\mathcal{B}}\gamma_k^2}{2\sum_{k\in\mathcal{B}}\gamma_k}, \tag{2.10}$$

$$\text{Prob}\left\{\phi(\bar{x}_R,y_R)-\phi(\bar{x}_R,y^*(\bar{x}_R))\ge\lambda(\tfrac{2D_Y^2+M_\Phi^2\sum_{k\in\mathcal{B}}\gamma_k^2}{2\sum_{k\in\mathcal{B}}\gamma_k})\right\}\le\tfrac{1}{\lambda}, \tag{2.11}$$

$$\text{Prob}\left\{g(\bar{x}_R)\ge\eta_R+\lambda\sigma\frac{\sqrt{\sum_{k=\tau(s)}^{\tau(N)}\gamma_k^2}}{\sum_{k=\tau(s)}^{\tau(N)}\gamma_k}\right\}\le\tfrac{1}{\lambda^2}. \tag{2.12}$$

b) *Under Assumption 5, we have,* $\forall\lambda>0$*,*

$$\mathbb{E}[\phi(\bar{x}_R,y_R)-\phi(\bar{x}_R,y^*(\bar{x}_R))]\le\tfrac{2D_Y^2+M_\Phi^2\sum_{k\in\mathcal{B}}\gamma_k^2}{2\sum_{k\in\mathcal{B}}\gamma_k}, \tag{2.13}$$

$$\text{Prob}\,\{\phi(\bar{x}_R,y_R)-\phi(\bar{x}_R,y^*(\bar{x}_R))\ge K_0+\lambda K_1\}\le\exp\{-\lambda\}+\exp\{-\lambda^2/3\}, \tag{2.14}$$

$$\text{Prob}\left\{g(\bar{x}_R)\ge\eta_R+\lambda\sigma\frac{\sqrt{\sum_{k=\tau(s)}^{\tau(N)}\gamma_k^2}}{\sum_{k=\tau(s)}^{\tau(N)}\gamma_k}\right\}\le\exp\{-\lambda^2/3\}, \tag{2.15}$$

where $K_0=\frac{2D_Y^2+M_\Phi^2\sum_{k\in\mathcal{B}}\gamma_k^2}{2\sum_{k\in\mathcal{B}}\gamma_k}$ *and* $K_1=\frac{M_\Phi^2\sum_{k\in\mathcal{B}}\gamma_k^2+4M_\Phi D_Y\sqrt{\sum_{k\in\mathcal{B}}\gamma_k^2}}{2\sum_{k\in\mathcal{B}}\gamma_k}$.

where the expectation is taken w.r.t. R *and* $\zeta_1,\ldots,\zeta_N$.

Proof. Let us prove part a) first. Set $\Delta_k = \Phi(\bar{x}_k, y_k, \zeta_k) - \phi(\bar{x}_k, y_k)$, it follows from (2.4) (fix $y = y^*$) that

$$\sum_{k\in\mathcal{B}} \gamma_k \left[\phi(x_k, y_k) - \phi(x_k, y^*(x_k))\right] \leq D_Y^2 + \tfrac{1}{2}\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2 + \sum_{k\in\mathcal{B}} \gamma_k \Delta_k (y - y_k). \tag{2.16}$$

Since conditional on $\zeta_{[k-1]}$, the expectation of Δ_k equals to zero, then taking expectation on both sides of (2.16), and dividing both sides by $\sum_{k\in\mathcal{B}} \gamma_k$, we have (2.10). Hence, using the Markov inquality, we have (2.11). Denote $\delta_k = G(x_k, \xi_k) - g(x_k)$. It then follows from the convexity of $g(\cdot)$ and the definition of the set $\mathcal{B}$ that

$$g(\bar{x}_k) \leq \frac{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k g(x_k)}{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k} \leq \eta_k - \frac{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k \delta_k}{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k}. \tag{2.17}$$

Using the fact that $\mathbb{E}[\delta_k|\xi_{[k-1]}] = 0$ and $\mathbb{E}[|\delta_k|^2] \leq \sigma^2$, we have

$$\mathbb{E}\left[\left|\frac{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k \delta_k}{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k}\right|^2\right] \leq \frac{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2 \sigma^2}{(\sum_{k=\tau(s)}^{\tau(N)} \gamma_k)^2}.$$

From the Markov inequality, we have (2.12). Hence the part a) holds.

Under Assumption 5, (2.13) still holds. Using the fact that $\mathbb{E}[\exp\{\|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2/M_\Phi^2\}] \leq \exp\{1\}$ and Jensens inequality, we have $\mathbb{E}[\exp\{\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2/M_\Phi^2 \sum_{k\in\mathcal{B}} \gamma_k^2\}] \leq \exp\{1\}$. It then follows from Markov's inequality that $\forall \lambda \geq 0$,

$$\text{Prob}(\textstyle\sum_{k\in\mathcal{B}} \gamma_k^2 \|\Phi'(\bar{x}_k, y_k, \zeta_k)\|_*^2 > (1+\lambda) M_\Phi^2 \sum_{k\in\mathcal{B}} \gamma_k^2) \leq \frac{\exp\{1\}}{\exp\{1+\lambda\}} \leq \exp\{-\lambda\}. \tag{2.18}$$

Also,

$$\text{Prob}\{\textstyle\sum_{k\in\mathcal{B}} \gamma_k \Delta_k (y - y_k) > 2\lambda M_\Phi D_Y \sqrt{\sum_{k\in\mathcal{B}} \gamma_k^2}\} \leq \exp\{-\lambda^2/3\} \tag{2.19}$$

Combining (2.16), (2.18) and (2.19), we have (2.14). Similarly, we have

$$\text{Prob}\{\textstyle\sum_{k=\tau(s)}^{\tau(N)} \gamma_k \delta_k \geq \lambda\sigma\sqrt{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2}\} \leq \exp\{-\lambda^2/3\} \tag{2.20}$$

Combining (2.17) and (2.20), we have (2.15). ■

Below we provide a special selection of s, $\{\gamma_k\}$ and $\{\eta_k\}$.

Corollary 20 *Let* $s = \frac{N}{2} + 1$, $\gamma_k = \frac{D_X}{M_G\sqrt{k}}$ *and* $\eta_k = \frac{4M_G D_X}{\sqrt{k}}$ *for* $k = 1, \ldots, N$. *Then we have*

$$\mathbb{E}[\phi(x_R, y_R) - \phi(x_R, y^*(x_R))] \leq \tfrac{8M_\Phi D_Y}{\sqrt{N}} \max\{\nu, \tfrac{1}{\nu}\},$$

where $\nu := (M_G D_Y)/(M_\Phi D_X)$. *Moreover, the following statements hold.*

a) *Under Assumption 4,*

$$\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \leq \lambda\tfrac{8M_\Phi D_Y}{\sqrt{N}} \max\{\nu, \tfrac{1}{\nu}\}\right\} \geq (1 - \tfrac{1}{\lambda})^2, \tag{2.21}$$

$$\text{Prob}\left\{g(\bar{x}_R) \leq \lambda\tfrac{\sqrt{2}D_X}{M_G\sqrt{N}}\right\} \geq (1 - \tfrac{1}{\lambda})^2. \tag{2.22}$$

b) *Under Assumption 5,*

$$\begin{aligned}
&\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \leq K_0 + \lambda K_1\right\} \\
&\geq (1 - 2\exp\{-\lambda^2/3\})(1 - \exp\{-\lambda\} - \exp\{-\lambda^2/3\}), \\
&\text{Prob}\left\{g(\bar{x}_R) \leq \tfrac{\sqrt{2}D_X}{M_G\sqrt{N}} + \lambda\tfrac{5\sigma}{\sqrt{N}}\right\} \geq (1 - 2\exp\{-\lambda^2/3\})(1 - \exp\{-\lambda^2/3\}),
\end{aligned}$$

where $K_0 = \frac{8M_\Phi D_Y}{\sqrt{N}} \max\{\nu, \frac{1}{\nu}\}$ *and* $K_1 = \frac{1}{\sqrt{N}}\left(\frac{4M_\Phi^2 D_X}{M_G} + 10M_\Phi D_Y\right)$.

Proof. Similarly to Corollary 5, we can show that (2.8) holds. It then follows from

Lemma 18 and Theorem 19.a) that

$$\sum_{k\in\mathcal{B}}\gamma_k = \sum_{k\in\mathcal{B}}\frac{D_X}{M_G\sqrt{k}} \geq \frac{D_X}{M_G}\frac{N}{4}\frac{1}{\sqrt{N}} = \frac{D_X\sqrt{N}}{4M_G}.$$

$$\begin{aligned}\mathbb{E}[\phi(x_R,y_R)-\phi(x_R,y^*)] &\leq \frac{2M_G}{D_X\sqrt{N}}\left[2D_Y^2+\sum_{k\in\mathcal{B}}\frac{D_X^2M_\Phi^2}{M_G^2k}\right] \leq \frac{2M_G}{D_X\sqrt{N}}\left[2D_Y^2+\sum_{k=N/2}^{N}\frac{D_X^2M_\Phi^2}{M_G^2k}\right]\\ &\leq \frac{2M_G}{D_X\sqrt{N}}[2D_Y^2+\log 2D_X^2\frac{M_\Phi^2}{M_G^2}] \leq \frac{8M_\Phi D_Y}{\sqrt{N}}\max\{\nu,\tfrac{1}{\nu}\}.\end{aligned}$$

Similarly, part b) follows from Theorem 19.b).

■

By Corollary (20), the CSPA method applied to (1.4)-(1.5) can achieve an $\mathcal{O}(1/\sqrt{N})$ rate of convergence.

2.3.2 CSPA with strong convexity assumptions

In this subsection, we modify problem (1.4)-(1.5) by imposing certain strong convexity assumptions to Φ and G with respect to y and x, respectively, i.e., $\exists\mu_\Phi,\mu_G>0$, s.t.

$$\Phi(x,y_1,\zeta)\geq\Phi(x,y_2,\zeta)+\langle\Phi'(x,y_2,\zeta),y_1-y_2\rangle+\tfrac{\mu_\Phi}{2}\|y_1-y_2\|^2,\ \ \forall y_1,y_2\in Y. \quad (2.23)$$

$$G(x_1,\xi)\geq G(x_2,\xi)+\langle G'(x_2,\xi),x_1-x_2\rangle+\tfrac{\mu_G}{2}\|x_1-x_2\|^2,\ \ \forall x_1,x_2\in X. \quad (2.24)$$

We also assume that the pair of solutions (x^*,y^*) exists for problem (1.4)-(1.5). Our main goal in this subsection is to estimate the convergence properties of the CSPA algorithm under these new assumptions.

We need to modify the probability distribution (2.3) used in the CSPA algorithm as follows. Given the stepsize γ_k, modulus μ_G and μ_Φ, and growth parameter Q (see (2.42)), let us define

$$a_k := (\mu_\Phi\gamma_k)/Q \text{ and } A_k := \begin{cases}1, & k=1;\\ \prod_{i\leq k,\ i\in\mathcal{B}}(1-a_i), & k>1,\end{cases} \quad (2.25)$$

and denote

$$b_k := (\mu_G \gamma_k)/Q \text{ and } B_k := \begin{cases} 1, & k = 1; \\ \prod_{i=1}^{k}(1 - b_i), & k > 1. \end{cases} \tag{2.26}$$

Also the probability distribution of R is modified to

$$\text{Prob}\{R = k\} = \frac{\gamma_k / A_k}{\sum_{i \in \mathcal{B}} \gamma_i / A_i}, k \in \mathcal{B}. \tag{2.27}$$

The following result shows some simple but important properties for the modified CSPA method applied to problem (1.4)-(1.5).

Proposition 21 *For any $s \leq k \leq m$, we have*

$$\begin{aligned} \textstyle\sum_{k \in \mathcal{B}} \frac{\gamma_k}{A_k}[\Phi(x_k, y_k, \zeta_k) - \Phi(x_k, y, \zeta_k)] \leq & (1 - \tfrac{\mu_\Phi \gamma_s}{Q}) V_Y(y_s, y) \\ & + \tfrac{1}{2} \textstyle\sum_{k \in \mathcal{B}} \frac{\gamma_k^2}{A_k} \|\Phi'(x_k, y_k, \zeta_k)\|_*^2, \ \ \forall y \in Y \end{aligned} \tag{2.28}$$

$$\begin{aligned} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \frac{\gamma_k}{B_k} [\eta_k - G(x, \xi_k)] \leq & \left(1 - \tfrac{\mu_G \gamma_s}{Q}\right) V_X(x_s, x) \\ & + \tfrac{1}{2} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \frac{\gamma_k^2}{B_k} \|G'(x_k, \xi_k)\|_*^2, \ \ \forall x \in X. \end{aligned} \tag{2.29}$$

Proof. Using Lemma 1 and the strong convexity of Φ w.r.t. y, for $k \in B$, we have

$$\begin{aligned} V_Y(y_{k+1}, y) &\leq V_Y(y_k, y) - \gamma_k \langle \Phi'(x_k, y_k, \zeta_k), y_k - y \rangle + \tfrac{1}{2} \gamma_k^2 \|\Phi'(x_k, y_k, \zeta_k)\|_*^2 \\ &\leq V_Y(y_k, y) - \gamma_k \left[\Phi(x_k, y_k, \zeta_k) - \Phi(x_k, y, \zeta_k) + \tfrac{\mu_\Phi}{2} \|y_k - y\|^2\right] \\ &\quad + \tfrac{1}{2} \gamma_k^2 \|\Phi'(x_k, y_k, \zeta_k)\|_*^2 \\ &\leq \left(1 - \tfrac{\mu_\Phi \gamma_k}{Q}\right) V_Y(y_k, y) - \gamma_k [\Phi(x_k, y_k, \zeta_k) - \Phi(x_k, y, \zeta_k)] + \tfrac{1}{2} \gamma_k^2 \|\Phi'(x_k, y_k, \zeta_k)\|_*^2. \end{aligned}$$

Also note that $V_Y(y_{k+1}, y) = V_Y(y_k, y)$ for all $k \in \mathcal{N}$. Summing up these relations for all $k \in \mathcal{B} \cup \mathcal{N}$ and using Lemma 12, we obtain

$$\begin{aligned} \tfrac{V_Y(y_N, y)}{A_{N+1}} \leq & \left(1 - \tfrac{\mu_\Phi \gamma_s}{Q}\right) V_Y(y_s, y) - \textstyle\sum_{k \in \mathcal{B}} \frac{\gamma_k}{A_k} [\Phi(x_k, y_k, \zeta_k) - \Phi(x_k, y, \zeta_k)] \\ & + \tfrac{1}{2} \textstyle\sum_{k \in \mathcal{B}} \frac{\gamma_k^2}{A_k} \|\Phi'(x_k, y_k, \zeta_k)\|_*^2. \end{aligned} \tag{2.30}$$

Similarly for $\tau(s) \leq k \leq \tau(N)$, we have

$$\begin{aligned} V_X(x_{k+1}, x) &\leq V_X(x_k, x) - \gamma_k \langle G'(x_k, \xi_k), x_k - x \rangle + \tfrac{1}{2}\gamma_k^2 \|G'(x_k, \xi_k)\|_*^2 \\ &\leq V_X(x_k, x) - \gamma_k \left[G(x_k, \xi_k) - G(x, \xi_k) + \tfrac{\mu_G}{2}\|x_k - x\|^2 \right] + \tfrac{1}{2}\gamma_k^2 \|G'(x_k, \xi_k)\|_*^2 \\ &\leq \left(1 - \tfrac{\mu_G \gamma_k}{Q}\right) V_X(x_k, x) - \gamma_k [G(x_k, \xi_k) - G(x, \xi_k)] + \tfrac{1}{2}\gamma_k^2 \|G'(x_k, \xi_k)\|_*^2, \end{aligned}$$

Summing up these relations for $\tau(s) \leq k \leq \tau(N)$ and using Lemma 12, we have

$$\tfrac{V_X(x_{N+1}, x)}{A_N} \leq \left(1 - \tfrac{\mu_G \gamma_s}{Q}\right) V_X(x_s, x) - \textstyle\sum_{k=\tau(s)}^{\tau(N)} \tfrac{\gamma_k}{A_k} [\eta_k - G(x, \xi_k)] + \tfrac{1}{2} \sum_{k=\tau(s)}^{\tau(N)} \tfrac{\gamma_k^2}{A_k} \|G'(x_k, \xi_k)\|_*^2. \tag{2.31}$$

Using the facts that $V_Y(y_{N+1}, y)/A_N \geq 0$ and $V_X(x_{N+1}, x)/A_N \geq 0$, and rearranging the terms in (2.30) and (2.31), we obtain (2.28) and (2.29), respectively.

■

Lemma 22 below provides a sufficient condition which guarantees that the output solution $(\bar{x}_R, y_R)$ is well-defined.

Lemma 22 *The following statements hold.*

a) *Under Assumption 4, if for any $\lambda > 0$, we have*

$$\tfrac{N-s+1}{2} \min_{k \in \mathcal{N}} \tfrac{\gamma_k \eta_k}{B_k} > \left(1 - \tfrac{\mu_G \gamma_s}{Q}\right) D_X^2 + \lambda \tfrac{M_G^2}{2} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \tfrac{\gamma_k^2}{B_k}, \tag{2.32}$$

then Prob$\{|\mathcal{B}| \geq \tfrac{N-s+1}{2}\} \geq 1 - 1/\lambda$.

b) *Under Assumption 5, if for any $\lambda > 0$, we have*

$$\tfrac{N-s+1}{2} \min_{k \in \mathcal{N}} \tfrac{\gamma_k \eta_k}{B_k} > \left(1 - \tfrac{\mu_G \gamma_s}{Q}\right) D_X^2 + (1+\lambda) \tfrac{M_G^2}{2} \textstyle\sum_{k=\tau(s)}^{\tau(N)} \tfrac{\gamma_k^2}{B_k} + \lambda \sigma \sqrt{\textstyle\sum_{k=\tau(s)}^{\tau(N)} \tfrac{\gamma_k^2}{B_k^2}}, \tag{2.33}$$

then Prob$\{|\mathcal{B}| \geq \tfrac{N-s+1}{2}\} \geq 1 - 2\exp\{-\lambda^2/3\}$.

Proof. The proof is similar to the one of Lemma 18 and hence the details are skipped.

■

Now let us establish the rate of convergence of the modified CSPA method for problem (1.4)-(1.5).

Theorem 23 *Suppose that $\{\gamma_k\}$ and $\{\eta_k\}$ are chosen according to Lemma 22. Then*

$$\mathbb{E}[\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R))] \leq \left(\sum_{k\in B} \frac{\gamma_k}{A_k}\right)^{-1} \left((1 - \frac{\mu_\Phi \gamma_s}{Q})D_Y^2 + \frac{M_\Phi^2}{2}\sum_{k\in B} \frac{\gamma_k^2}{A_k}\right). \tag{2.34}$$

Moreover, under Assumption 4, we have for any $\lambda > 0$,

$$\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \geq \lambda \left(\sum_{k\in B} \frac{\gamma_k}{A_k}\right)^{-1} \left[(1 - \frac{\mu_\Phi \gamma_s}{Q})D_Y^2 + \frac{M_\Phi^2}{2}\sum_{k\in B} \frac{\gamma_k^2}{A_k}\right]\right\} \leq \frac{1}{\lambda}, \tag{2.35}$$

$$\text{Prob}\left\{g(\bar{x}_R) \geq \eta_R + \lambda\sigma \frac{\sqrt{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2/B_k^2}}{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k/B_k}\right\} \leq \frac{1}{\lambda^2}. \tag{2.36}$$

In addition, under Assumption 5, we have for any $\lambda > 0$,

$$\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \geq K_0 + \lambda K_1\right\} \leq \exp\{-\lambda\} + \exp\{-\lambda^2/3\}, \tag{2.37}$$

$$\text{Prob}\left\{g(\bar{x}_R) \geq \eta_R + \lambda\sigma \frac{\sqrt{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k^2/B_k^2}}{\sum_{k=\tau(s)}^{\tau(N)} \gamma_k/B_k}\right\} \leq \exp\{-\lambda^2/3\}, \tag{2.38}$$

where $K_0 = \left(\sum_{k\in\mathcal{B}} \frac{\gamma_k}{A_k}\right)^{-1} \left[(1 - \frac{\mu_\Phi \gamma_s}{Q})D_Y^2 + \frac{M_\Phi^2}{2}\sum_{k\in\mathcal{B}} \frac{\gamma_k^2}{A_k}\right]$
and $K_1 = \left(\sum_{k\in\mathcal{B}} \frac{\gamma_k}{A_k}\right)^{-1} \left[M_\Phi^2 \sum_{k\in\mathcal{B}} \frac{\gamma_k^2}{A_k} + 4M_\Phi D_Y \sqrt{\sum_{k\in\mathcal{B}} \frac{\gamma_k^2}{A_k^2}}\right]$.

Proof. The proof is similar to the proof of Theorem 19, and hence the details are skipped. ■

Now we provide a specific selection of $\{\gamma_k\}$ and $\{\eta_k\}$ that satisfies the condition of

Lemma 22. While the selection of η_k only depends on iteration index k, i.e.,

$$\eta_k = \tfrac{8QM_G^2}{k\mu_G}, \tag{2.39}$$

the selection of γ_k depends on the particular position of iteration index k in set $\mathcal{B}$ or $\mathcal{N}$. More specifically, let $\tau_{\mathcal{B}(k)}$ and $\tau(k)$ be the position of index k in set $\mathcal{B}$ and set $\mathcal{N}$, respectively (for example, $\mathcal{B} = \{1, 3, 5, 9, 10\}$ and $\mathcal{N} = \{2, 4, 6, 7, 8\}$. If $k = 9$, then $\tau_{B(k)} = 4$). We define γ_k as

$$\gamma_k = \begin{cases} \frac{2Q}{\mu_\Phi(\tau_{\mathcal{B}(k)}+1)}, & k \in \mathcal{B}; \\ \frac{2Q}{\mu_G(\tau(k)+1)}, & k \in \mathcal{N}. \end{cases} \tag{2.40}$$

Such a selection of γ_k can be conveniently implemented by using two separate counters in each iteration to represent $\tau_{\mathcal{B}(k)}$ and $\tau(k)$.

Corollary 24 *Let $s = 1$, η_k and γ_k be given in* (2.39) *and* (2.40)*, respectively. Then we have*

$$\mathbb{E}[\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R))] \leq \tfrac{8QM_\Phi^2}{(N+2)\mu_\Phi}.$$

Moreover, under Assumption 4, we have for any $\lambda > 0$,

$$\begin{aligned} &\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \leq \lambda \tfrac{8QM_\Phi^2}{(N+2)\mu_\Phi}\right\} \geq (1 - \tfrac{1}{\lambda})^2, \\ &\text{Prob}\left\{g(\bar{x}_R) \leq \lambda \tfrac{16QM_G^2}{N\mu_G}\right\} \geq (1 - \tfrac{1}{\lambda})^2. \end{aligned}$$

In addition, under Assumption 5, we have for any $\lambda > 0$,

$$\begin{aligned} &\text{Prob}\left\{\phi(\bar{x}_R, y_R) - \phi(\bar{x}_R, y^*(\bar{x}_R)) \leq K_0 + \lambda K_1\right\} \\ &\geq (1 - 2\exp\{-\lambda^2/3\})(1 - \exp\{-\lambda\} - \exp\{-\lambda^2/3\}), \\ &\text{Prob}\left\{g(\bar{x}_R) \leq \tfrac{16QM_G^2}{N\mu_G} + \lambda\tfrac{2\sigma}{\sqrt{N}}\right\} \geq (1 - 2\exp\{-\lambda^2/3\})(1 - \exp\{-\lambda^2/3\}), \end{aligned}$$

where $K_0 = 8QM_\Phi^2/[(N+2)\mu_\Phi]$ and $K_1 = 8QM_\Phi^2/[(N+2)\mu_\Phi] + 64M_\Phi D_Y/\sqrt{N}$.

Proof. The proof is similar to the proof of Corollary 20 and hence the details are skipped. ∎

Note that Corollary 24.a) implies an $\mathcal{O}(1/N)$ rate of convergence, while Corollary 24.b) show an $\mathcal{O}(1/\sqrt{N})$ rate of convergence with much improved dependence on λ. One possible approach to improve the result in part b) is to shrink the feasible set Y from time to time in order to obtain an $\mathcal{O}(1/N)$ rate of convergence (see [19]).

2.4 Numerical Experiment

In this section, we present some numerical results of our computational experiments for solving two problems: an asset allocation problem with conditional value at risk (CVaR) constraint and a parameterized classification problem. More specifically, we report the numerical results obtained from the CSA and CSPA method applied to these two problems in Subsection 4.1 and 4.2, respectively.

2.4.1 Asset allocation problem

Our goal of this subsection is to examine the performance of the CSA method applied to the CVaR constrained problem in (1.3).

Apparently, there is one problem associated with applying the CSA algorithm to this model – the feasible region X is unbounded. Lan, Nemirovski and Shapiro (see [65] Section 4.2) show that τ can be restricted to $\left[\underline{\mu} + \sqrt{\frac{\beta}{1-\beta}}\sigma, \bar{\mu} + \sqrt{\frac{1-\beta}{\beta}}\sigma\right]$, where $\underline{\mu} := \min_{y\in Y}\{-\bar{\xi}^T y\}$ and $\bar{\mu} := \max_{y\in Y}\{-\bar{\xi}^T y\}$.

In this experiment, we consider four instances. The first three instances are randomly generated according to the factor model in Goldfarb and Iyengar (see Section 7 of [85]) with different number of stocks ($d = 500$, 1000 and 2000), while the last instance consists of the 95 stocks from $S\&P100$ (excluding SBC, ATI, GS, LU and VIA-B) obtained from [6], the mean $\bar{\xi}$ and covariance Σ are estimated by the historical monthly data from 1996 to 2002. The reliability level $\beta = 0.05$, the number of samples to estimate $g(x)$ is $J = 100$

and the number of samples used to evaluate the solution is $n = 50,000$. It is worth noting that, by utilizing the linear structure of $\xi^T x$ (where $x \in \mathbb{R}^d$) in constraint function, in k-th iteration we generate J-sized i.i.d. samples of $\bar{\xi} := \xi^T x_k$ (with dimension 1) to estimate $\xi^T x$ in constraint function, instead of J-sized i.i.d. samples of ξ (with dimension d). For SAA algorithm, the deterministic SAA problem to (1.3) is defined by

$$\begin{array}{ll} \min_{x,\tau} & -\mu^T x \\ \text{s.t.} & \tau + \frac{1}{\beta N}\sum_{i=1}^{N}[-\xi_i^T x - \tau]_+ \leq 0, \\ & \sum_{i=1}^{n} x_i = 1, x \geq 0, \end{array} \tag{2.1}$$

We implemented the SAA approach by using Polyak's subgradient method for solving convex programming problems with function constraints (see [75]). The main reasons why we did not use the linear programming (LP) method to (2.1) include: 1) problem (2.1) might be infeasible for some instances; and 2) we tried the LP method with CVX toolbox for an instance with 500 stocks and the CPU time is thousands times larger than that of the CSA method. In our experiment, we adjust the stepsize strategy by multiplying γ_k and η_k with some scaling parameters c_g and c_e, respectively. These parameters are chosen as a result of pilot runs of our algorithm (see [65] for more details). We have found that the "best parameters" in Table 2.1 slightly outperforms other parameter settings we have considered.

Table 2.1: The stepsize factor

		best c_g	best c_e
Number	500	0.5	0.005
of stocks	1000	0.5	0.05
	2000	0.5	0.05

Notations in Tables 2.2-2.5.

N: the sample size(the number of steps in SA, and the size of the sample used to SAA approximation).

Obj.: the objective function value of our solution, i.e. the loss of the portfolio.

Cons.: the constraint function value of our solution.

CPU: the processing time in seconds for each method.

Table 2.2: Random Sample with 500 Assets

		N=500	N=1000	N=2000	N=5000
CSA	Obj.	-4.883	-4.870	-4.953	-4.984
	Cons.	5.330	4.096	5.167	2.859
	CPU	1.671e-01	3.383e-01	6.271e-01	1.470e+00
SAA	Obj.	-4.978	-4.981	-4.977	-4.977
	Cons.	4.372	3.071	2.330	2.249
	CPU	2.031e+00	9.926e+00	4.132e+01	2.591e+02

Table 2.3: Random Sample with 1000 Assets

		N=500	N=1000	N=2000	N=5000
CSA	Obj.	-4.532	-4.704	-4.838	-4.949
	Cons.	27.660	24.901	23.825	20.785
	CPU	4.193e-01	8.578e-01	1.659e+00	4.001e+00
SAA	Obj.	-4.965	-4.981	-4.981	-4.977
	Cons.	60.421	47.745	33.940	20.357
	CPU	1.513e+01	5.954e+01	2.774e+02	1.524e+03

Table 2.4: Random Sample with 2000 Assets

		N=500	N=1000	N=2000	N=5000
CSA	Obj.	-4.299	-4.077	-4.355	-4.859
	Cons.	144.92	112.54	89.74	82.65
	CPU	1.374e+00	2.810e+00	5.538e+00	2.716e+01
SAA	Obj.	-4.752	-4.699	-4.721	-4.727
	Cons.	279.43	218.96	147.93	94.46
	CPU	1.968e+01	6.571e+01	2.940e+02	3.697e+03

The following conclusions can be made from the numerical results. First, as far as the quality of solutions is concerned, the CSA method is at least as good as SAA method and it may outperform SAA for some instances especially as N increases. Second, the CSA method can significantly reduce the processing time than SAA method for all the instances.

Table 2.5: Comparing the CSA and SAA for the CVaR model

		N=500	N=1000	N=2000	N=5000	N=10000
CSA	Obj.	-3.531	-3.537	-3.542	-3.548	-3.560
	Cons.	3.382e+00	2.188e-01	1.106e-01	2.724e-01	-7.102e-01
	CPU	8.315e-02	1.422e-01	2.778e-01	7.251e-01	1.415e+00
SAA	Obj.	-3.530	-3.541	-3.541	-3.544	-3.559
	Cons.	3.385e+00	7.163e-01	6.989e-01	6.988e-01	7.061e-01
	CPU	3.155e+00	1.221e+01	4.834e+01	3.799e+02	1.462e+03

2.4.2 Classification and metric learning problem

In this subsection, our goal is to examine the efficiency of the CSPA algorithm applied to a classification problem with the metric as parameter. In this experiment, we use the expectation of hinge loss function, described in [86], as objective function, and formulate the constraint with the loss function of metric learning problem in [87], see formal definition in (1.6)-(1.7). For each i, j, we are given samples $u_i, u_j \in \mathbb{R}^d$ and a measure $b_{ij} \geq 0$ of the similarity between the samples u_i and u_j ($b_{ij} = 0$ means u_i and u_j are the same). The goal is to learn a metric A such that $\langle (u_i - u_j), A(u_i - u_j) \rangle \approx b_{ij}$, and to do classification among all the samples u projected by the learned metric A.

For solving this class of problems in machine learning, one widely accepted approach is to learn the metric in the first step and then solve the classification problem with the obtained optimal metric. However, this approach is not applicable to the online setting since once the dataset is updated with new samples, this approach has to go through all the samples to update A and ω. On the other hand, the CSPA algorithm optimizes the metric A and classifier ω simultaneously, and only needs to take one new sample in each iteration.

In this experiment, our goal is to test the solution quality of the CSPA algorithm with respect to the number of iterations. More specifically, we consider 2 instances of this problem with different dimension ($d = 100$ and 200, respectively). Since we are dealing with the online setting, our sample size for training A and ω is increasing with the number of iterations. The size for the sample used to estimate the parameters and the one used to evaluate the quality of solution (or testing sample) are set to 100 and $10,000$, respectively.

Within each trial, we test the objective and constraint value of the output solution over training sample and testing sample, respectively. Since R is randomly picked up from all the series $\{\bar{x}_k, y_k\}$, we generate 5 candidate R, instead of one, in order to increase the probability of getting a better solution. Intuitively, the latter solutions in the series should be better than the earlier ones, hence, we also put the last pair of the solution $(\bar{x}_N, y_N)$ into the candidate list. In each trial, we compare these 6 candidate solutions. First, we choose three pairs with smallest constraint function values, then, choose the one with the smallest objective function value from these three selected solutions as our output solution.

Table 2.6 and Table 2.7 shows the CSPA method decreases the objective value and constraint value as the sample size (number of iterations N) increases. These experiments demonstrate that we can improve both the metric and the classifier simultaneously by using the CSPA method as more and more data are collected.

Notations in Table 2.6 and 2.7.

Obj. Train: The objective function value using training sample at the output solution.

Cons. Train: The constraint function value using training sample at the output solution.

Obj. Test: The objective function value using testing sample at the output solution.

Cons. Test: The constraint function value using testing sample at the output solution.

Table 2.6: d = 100

N	Obj. Train	Cons. Train	Obj. Test	Cons. Test
100	3.175	3.056	1.042	3.068
200	2.737	3.058	0.811	3.006
600	0.654	3.077	0.157	3.104
800	0.529	3.087	0.126	3.102
1000	0.398	3.057	0.102	3.082

Table 2.7: d = 200

N	Obj. Train	Cons. Train	Obj. Test	Cons. Test
100	0.716	1.137	0.699	1.132
200	0.374	1.061	0.371	1.030
1000	0.360	1.020	0.364	1.031
2000	0.351	1.016	0.355	1.030
5000	0.291	0.951	0.135	0.989

2.5 Conclusions

In this chapter, we present a new stochastic approximation type method, the CSA method, for solving the stochastic convex optimization problems with function or expectation constraints. Moreover, we show that a variant of CSA method, the CSPA method, is applicable to a class of parameterized stochastic problem in (1.4)-(1.5). We show that these methods exhibit theoretically optimal rate of convergence for solving a few different classes of function or expectation constrained stochastic optimization problems and demonstrated their effectiveness through some preliminary numerical experiments.

CHAPTER 3
CONDITIONAL GRADIENT METHODS FOR CONVEX OPTIMIZATION WITH FUNCTION CONSTRAINTS

3.1 Introduction

This chapter aims to fill in the aforementioned gap in the literature by presenting a new class of conditional gradient methods for solving problem (1.8). Our main contributions are briefly summarized as follows. Firstly, inspired by the constraint-extrapolation (ConEx) method for function constrained convex optimization in [88], we develop a novel constraint-extrapolated conditional gradient (CoexCG) method for solving problem (1.8). While both methods are single-loop primal-dual type methods for solving convex optimization problems with function constraints, CoexCG only requires us to minimize a linear function, rather than to perform projection, over X. In the basic setting when both f and h_i are smooth convex functions with Lipschitz continuous gradients, we show that the total number of iterations performed by CoexCG before finding a ϵ-solution of problem (1.8), i.e., a point $\bar{x} \in X$ s.t. $f(\bar{x}) - f(x^*) \leq \epsilon$ and $\|g(\bar{x})\|_2 + \|[h(\bar{x})]_+\|_2 \leq \epsilon$, can be bounded by $\mathcal{O}(1/\epsilon^2)$. Here $[\cdot]_+ := \max\{\cdot, 0\}$.

Secondly, we consider more general function constrained optimization problems where either the objective function f or some constraint functions h_i are possibly nondifferentiable, but contains certain saddle point structure. We extend the CoexCG method for solving these problems in combination with the well-known Nesterov's smoothing scheme [89]. In general, even equipped with such smoothing technique, nonsmooth optimization is more difficult than smooth optimization, and its associated iteration complexity is worse than that for smooth ones by orders of magnitude. However, we show that a similar $\mathcal{O}(1/\epsilon^2)$ complexity bound can be achieved by CoexCG for solving these nonsmooth function con-

strained optimization problems. This seemly surprising result can be attributed to an inherent acceleration scheme in CoexCG that can reduce the impact of the Lipschitz constants induced by the smoothing scheme.

Thirdly, one possible shortcoming of CoexCG exists in that it requires the total number of iterations N fixed a priori before we run the algorithm in order to achieve the best rate of convergence. Therefore it is inconvenient to implement this algorithm when such an iteration limit is not available. In order to address this issue, we propose a constraint-extrapolated and dual-regularized conditional gradient (CoexDurCG) method by adding a diminishing regularization term for the dual updates. This modification allows us to design a novel adaptive stepsize policy which does not require N given in advance. Moreover, we show that the complexity of CoexDurCG is still in the same order of magnitude as CoexCG with a slightly larger constant factor. We also extend CoexDurCG for solving the aforementioned structured nonsmooth problems, and demonstrate that it is not necessary to explicitly define the smooth approximation problem. We note that this technique of adding a diminishing regularization term can be applied for solving problems with either unbounded primal feasible region (e.g., stochastic subgradient descent [64] and stochastic accelerated gradient descent [17]), or unbounded dual feasible region (e.g., ConEx [88]), for which one often requires the number of iterations fixed in advance.

Finally, we apply the developed algorithms for solving the radiation therapy treatment planning problem on both randomly generated instances and a real data set. We show that CoexDurCG performs comparably to CoexCG in terms of solution quality and computation time. We demonstrate that the incorporation of function constraints helps us not only to find feasible treatment plans satisfying clinical criteria, but also generate alternative treatment plans that can possibly reduce radiation exposure time for the patients.

To the best of our knowledge, all the algorithmic schemes as well as their complexity results are new in the area of projection-free methods for convex optimization.

This chapter is organized as follows. Section 3.2 is devoted to the CoexCG method.

We first present the CoexCG method for smooth function constrained convex optimization in Subsection 3.2.1 and extend it for solving structured nonsmooth function constrained convex optimization in Subsection 3.2.2. We then discuss the CoexDurCG method in Section 3.3, including its basic version for smooth function constrained convex optimization in Subsection 3.3.1 and its extended version for directly solving structured nonsmooth function constrained convex optimization problems in Subsection 3.3.2. We apply these methods for radiation therapy treatment planning in Section 4.5, and conclude the chapter with a brief summary in Section 3.5.

3.2 Constraint-extrapolated conditional gradient method

In this section, we present a basic version of the constraint-extrapolated conditional gradient method for solving convex optimization problem (1.8). Subsection 3.2.1 focuses on the case when f and h_i are smooth convex functions, while subsection 3.2.2 extends our discussion to the situation where f and h_i are not necessarily differentiable.

3.2.1 Smooth functions

Throughout this subsection, we assume that f and h_i are differential and their gradients are Lipschitz continuous s.t.

$$\|\nabla f(x_1) - \nabla f(x_2)\|_* \leq L_f \|x_1 - x_2\|, \ \forall x_1, x_2 \in X, \tag{3.1}$$

$$\|\nabla h_i(x_1) - \nabla h_i(x_2)\|_* \leq L_{h,i} \|x_1 - x_2\|, \forall x_1, x_2 \in X, i = 1, \ldots, d. \tag{3.2}$$

Here $\|\cdot\|$ denotes an arbitrary norm which is not necessarily associated with the inner product $\langle \cdot, \cdot \rangle$ ($\|\cdot\|_*$ is the conjugate norm of $\|\cdot\|$). For notational convenience, we denote

$$L_h = (L_{h,1}; \ldots; L_{h,d}) \ \text{ and } \ \bar{L}_h = \|L_h\|_2.$$

We need to use the Lipschitz continuity of the constraint function h_i when developing conditional gradient methods for function constrained problems. Clearly, under the boundedness assumption of X, the constraint functions h_i are Lipschitz continuous with constant $M_{h,i}$, i.e.,

$$\|\nabla h_i(x)\|_* \leq M_{h,i}, \ \forall x \in X. \tag{3.3}$$

In particular, letting x^* be an optimal solution of problem (1.8), we have $M_{h,i} \leq \nabla f(x^*) + L_{h,i} D_X$, where D_X denotes the diameter of X given by

$$D_X := \max_{x_1, x_2 \in X} \|x_1 - x_2\|. \tag{3.4}$$

Note that a different way to bound on $M_{h,i}$ will be discussed for certain structured nonsmooth problems in Subsection 3.2.2. For the sake of notational convenience, we also denote

$$\bar{M}_h = \sqrt{\textstyle\sum_{i=1}^d M_{h,i}^2}. \tag{3.5}$$

Since we can only perform linear optimization over the feasible region X, one natural way to solve problem (1.8) is to consider its saddle point reformulation

$$\min_{x \in X} \max_{y \in \mathbb{R}^m, z \in \mathbb{R}^d_+} f(x) + \langle g(x), y \rangle + \langle h(x), z \rangle. \tag{3.6}$$

Throughout the chapter, we assume that the standard Slater condition holds for problem (1.8) so that a pair of optimal dual solutions (y^*, z^*) of problem (3.6) exists.

In [28] (see also Chapter 7 of [44]), Lan presented a smoothing conditional gradient method for solving problems in the form of (3.6). This method applies the conditional gradient algorithm for a properly smoothed version of the objective function of (3.6). However, this scheme is not applicable for our setting due to the following reasons. Firstly, the smoothing conditional gradient method only solves bilinear saddle point problems with linear coupling terms given by $\langle g(x), y \rangle$ and cannot deal with the nonlinear coupling term

$\langle h(x), z\rangle$. Secondly, even for the bilinear saddle point problems, the smoothing conditional gradient method in [44, 28] requires the feasible set of y to be bounded, which does not hold for problem (3.6).

Our development has been inspired the constraint extrapolation (ConEx) method recently introduced by Boob, Deng and Lan [88] for solving problem (3.6). ConEx is an accelerated primal-dual type method which updates both the primal variable x and dual variables (y, z) in the each iteration. In comparison with some previously developed accelerated primal-dual methods for solving saddle point problems with nonlinear coupling terms [90, 91], one distinctive feature of ConEx is that it defines the acceleration (or momentum) step by extrapolating the linear approximation of the nonlinear function h. As a consequence, it can deal with unbounded feasible regions for the dual variable z (or y) and thus solve the function (or affine) constrained convex optimization problems. However, each iteration of the ConEx method requires the projection onto the feasible region X, and hence is not applicable to our problem setting.

In order to address the above issues for solving problem (1.8) (or (3.6)), we present a novel constraint-extrapolated conditional gradient (CoexCG) method, which incorporates some basic ideas of the ConEx method into the conditional gradient method. As shown in Algorithm 3, the CoexCG method first performs in (3.9) an extrapolation step for the affine constraint g. Then in (3.10) it performs an extrapolation step based on the linear approximation of the constraint function h given by

$$l_{h_i}(\bar{x}, x) := h_i(\bar{x}) + \langle \nabla h_i(\bar{x}), x - \bar{x}\rangle, \tag{3.7}$$

$$l_h(\bar{x}, x) := (l_{h_1}(\bar{x}, x); \ldots, l_{h_d}(\bar{x}, x)). \tag{3.8}$$

Utilizing the extrapolated constraint values $\tilde{g}_k$ and $\tilde{h}_k$, it then updates the dual variables q_k and r_k associated with the affine constraint $g(x) = 0$ and the nonlinear constraints $h(x) \leq 0$ in (3.11) and (3.12), respectively. With these updated dual variables and linear

approximation $l_f(x_{k-1}, x)$ and $l_h(x_{k-1}, x)$, it solves a linear optimization problem over X to update the primal variable $p_k \in X$ in (3.13). Finally, the output solution x_k is computed as a convex combination of x_{k-1} and p_k in (3.14).

Algorithm 3 Constraint-**ex**trapolated **C**onditional **G**radient (CoexCG)

Let the initial points $p_0 = p_{-1} \in X$, $x_0 = x_{-1} = x_{-2} \in X$, $q_0 \in \mathbb{R}^m$ and $r_0 \in \mathbb{R}^d_+$ be given. Also let the stepsize parameters $\lambda_k \geq 0$, $\tau_k \geq 0$ and $\alpha_k \in [0, 1]$ be given.

for $k = 1$ **to** N **do**

$$\tilde{g}_k = g(p_{k-1}) + \lambda_k[g(p_{k-1}) - g(p_{k-2})], \tag{3.9}$$

$$\tilde{h}_k = l_h(x_{k-2}, p_{k-1}) + \lambda_k[l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2})], \tag{3.10}$$

$$q_k = \text{argmin}_{y \in \mathbb{R}^m}\{\langle -\tilde{g}_k, y\rangle + \tfrac{\tau_k}{2}\|y - q_{k-1}\|_2^2\}, \tag{3.11}$$

$$r_k = \text{argmin}_{z \in \mathbb{R}^d_+}\{\langle -\tilde{h}_k, z\rangle + \tfrac{\tau_k}{2}\|z - r_{k-1}\|_2^2\}, \tag{3.12}$$

$$p_k = \text{argmin}_{x \in X}\{l_f(x_{k-1}, x) + \langle g(x), q_k\rangle + \langle l_h(x_{k-1}, x), r_k\rangle\}, \tag{3.13}$$

$$x_k = (1 - \alpha_k)x_{k-1} + \alpha_k p_k. \tag{3.14}$$

end for

It is interesting to build some connections between the CoexCG method and the ConEx method in [88]. In particular, by replacing the relations in (3.13) and (3.14) with

$$p_k = \text{argmin}_{x \in X}\{l_f(p_{k-1}, x) + \langle g(x), q_k\rangle + \langle l_h(p_{k-1}, x), r_k\rangle + \tfrac{\eta_k}{2}\|x - p_{k-1}\|_2^2\},$$

then we essentially obtain the ConEx method. Comparing these relations, we observe that the CoexCG method differs from the ConEx method in the following few aspects. Firstly, p_t in CoexCG is computed by solving a linear optimization problem, while the one in the ConEx method is computed by using a projection. The use of linear optimization enables the CoexCG method to generate sparse solutions in feasible sets X with a huge large number of extreme points (see Section 4.5). Secondly, the linear approximation models l_f and

l_h in the ConEx method is built on the search point p_{k-1}, while the one in the CoexCG method is built on x_{k-1}, or equivalently, the convex combination of all previous search points p_i, $i = 1, \ldots, k-1$.

We need to add a few more remarks about the CoexCG method. Firstly, by (3.11) and (3.12), we can define q_k and r_k equivalently as

$$q_k = q_{k-1} + \tfrac{1}{\tau_k}\tilde{g}_k,$$
$$r_k = \max\{r_{k-1} + \tfrac{1}{\tau_k}\tilde{h}_k, 0\}.$$

It is also worth noting that we can generalize the CoexCG method to deal with conic inequality constraint $h(x) \in \mathcal{K}$, by simply replacing the constraint $z \in \mathbb{R}^d_+$ in (3.12) with $z \in -\mathcal{K}^*$. Here $\mathcal{K} \subset \mathbb{R}^l$ is a given closed convex cone and $\mathcal{K}^*$ denotes its the dual cone.

Secondly, in addition to the primal output solution x_k in (3.14), we can also define the dual output solutions y_k and z_k as

$$y_k = (1 - \alpha_k)y_{k-1} + \alpha_k q_k, \tag{3.15}$$
$$z_k = (1 - \alpha_k)z_{k-1} + \alpha_k r_k. \tag{3.16}$$

Different from x_k, these dual variables y_k and z_k do not participate in the updating of any other search points. However, both of them will be used intensively in the convergence analysis of the CoexCG method.

Thirdly, even though we do not need to select the parameter η_k when defining p_k as in the ConEx method, we do need to specify the stepsize parameter τ_k to update the dual variables q_k and r_k. We also need to determine the parameters λ_k and α_k, respectively, to define the extrapolation steps and the output solution x_k. We will discuss the selection of these algorithmic parameters after establishing some general convergence properties of the

CoexCG method.

Our goal in the remaining part of this subsection is to establish the convergence of the CoexCG method. Let x_k, y_k, and z_k be defined in (3.14), (3.15), and (3.16). Throughout this section, we denote $w_k \equiv (x_k, y_k, z_k)$ and $w \equiv (x, y, z)$, and define the gap function $Q(w_k, w)$ as

$$Q(w_k, w) := f(x_k) - f(x) + \langle g(x_k), y\rangle - \langle g(x), y_k\rangle + \langle h(x_k), z\rangle - \langle h(x), z_k\rangle. \quad (3.17)$$

We start by stating some well-known technical results that have been used in the convergence analysis of many first-order methods. The first result, often referred to "three-point lemma" (see, e.g., Lemma 3.1 of [44]), characterizes the optimality conditions of (3.11) and (3.12).

Lemma 1 *Let q_k and r_k be defined in (3.11) and (3.12), respectively. Then,*

$$\langle -\tilde{g}_k, q_k - y\rangle + \tfrac{\tau_k}{2}\|q_k - q_{k-1}\|_2^2 \le \tfrac{\tau_k}{2}\|y - q_{k-1}\|_2^2 - \tfrac{\tau_k}{2}\|y - q_k\|_2^2, \forall y \in \mathbb{R}^m, \quad (3.18)$$

$$\langle -\tilde{h}_k, r_k - z\rangle + \tfrac{\tau_k}{2}\|r_k - r_{k-1}\|_2^2 \le \tfrac{\tau_k}{2}\|z - r_{k-1}\|_2^2 - \tfrac{\tau_k}{2}\|z - r_k\|_2^2, \forall z \in \mathbb{R}^d_+. \quad (3.19)$$

The following result helps us to take telescoping sums (see Lemma 3.17 of [44]).

Lemma 2 *Let $\alpha_k \in (0, 1], k = 0, 1, 2, \ldots$, be given and denote*

$$\Gamma_k = \begin{cases} 1, & \text{if } k = 1; \\ (1 - \alpha_k)\Gamma_{k-1}, & \text{if } k > 1. \end{cases} \quad (3.20)$$

If $\{\Delta_k\}$ satisfies

$$\Delta_{k+1} \le (1 - \alpha_k)\Delta_k + B_k, \forall k \ge 1,$$

then we have

$$\tfrac{\Delta_{k+1}}{\Gamma_k} \le (1 - \alpha_1)\Delta_1 + \textstyle\sum_{i=1}^k \tfrac{B_i}{\Gamma_i}.$$

We now establish an important recursion of the CoexCG method.

Proposition 3 *For any $k > 1$, we have*

$$\begin{aligned}
Q(w_k, w) \le (1-\alpha_k)Q(w_{k-1}, w) &+ \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} + \tfrac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k} \\
&+ \alpha_k[\langle A(p_k - p_{k-1}), y - q_k\rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1}\rangle] \\
&+ \alpha_k[\langle l_h(x_{k-1}, p_k) - l_h(x_{k-2}, p_{k-1}), z - r_k\rangle \\
&- \lambda_k \langle l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2}), z - r_{k-1}\rangle] \\
&+ \tfrac{\alpha_k \tau_k}{2}[\|y - q_{k-1}\|_2^2 - \|y - q_k\|_2^2 + \|z - r_{k-1}\|_2^2 - \|z - r_k\|_2^2],\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+,
\end{aligned}$$

where D_X is defined in (3.4).

Proof. It follows from the smoothness of f and h (e.g., Lemma 3.2 of [44]) and the definition of x_k in (3.14) that

$$\begin{aligned}
f(x_k) &\le l_f(x_{k-1}, x_k) + \tfrac{L_f}{2}\|x_k - x_{k-1}\|^2 \\
&= (1-\alpha_k) l_f(x_{k-1}, x_{k-1}) + \alpha_k l_f(x_{k-1}, p_k) + \tfrac{L_f \alpha_k^2}{2}\|p_k - x_{k-1}\|^2 \\
&= (1-\alpha_k) f(x_{k-1}) + \alpha_k l_f(x_{k-1}, p_k) + \tfrac{L_f \alpha_k^2}{2}\|p_k - x_{k-1}\|^2. \\
h_i(x_k) &\le (1-\alpha_k) h_i(x_{k-1}) + \alpha_k l_{h_i}(x_{k-1}, p_k) + \tfrac{L_{h,i} \alpha_k^2}{2}\|p_k - x_{k-1}\|^2.
\end{aligned}$$

Using the above two relations in the definition of $Q(w_k, w)$ in (3.17), we have for any

$w \equiv (x, y, z) \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$,

$$
\begin{aligned}
Q(w_k, w) &= f(x_k) - f(x) + \langle g(x_k), y\rangle - \langle g(x), y_k\rangle + \langle h(x_k), z\rangle - \langle h(x), z_k\rangle \\
&\le (1-\alpha_k) f(x_{k-1}) + \alpha_k l_f(x_{k-1}, p_k) - f(x) + \langle g(x_k), y\rangle - \langle g(x), y_k\rangle \\
&\quad + \langle (1-\alpha_k) h(x_{k-1}) + \alpha_k l_h(x_{k-1}, p_k)), z\rangle - \langle h(x), z_k\rangle \\
&\quad + \tfrac{(L_f + z^T L_h)\alpha_k^2}{2} \|p_k - x_{k-1}\|^2 \\
&= (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2}{2} \|p_k - x_{k-1}\|^2 \\
&\quad + \alpha_k [l_f(x_{k-1}, p_k) - f(x) + \langle g(p_k), y\rangle - \langle g(x), q_k\rangle + \langle l_h(x_{k-1}, p_k), z\rangle - \langle h(x), r_k\rangle].
\end{aligned}
$$

Moreover, by the definition of x_k in (3.14) and the convexity of f and h_i, we have

$$
\begin{aligned}
& l_f(x_{k-1}, p_k) + \langle g(p_k), q_k\rangle + \langle l_h(x_{k-1}, p_k), r_k\rangle \\
& \le l_f(x_{k-1}, x) + \langle g(x), q_k\rangle + \langle l_h(x_{k-1}, x), r_k\rangle \\
& \le f(x) + \langle g(x), q_k\rangle + \langle h(x), r_k\rangle, \ \forall x \in X.
\end{aligned}
$$

Combining the above two relations, we obtain

$$
\begin{aligned}
Q(w_k, w) &\le (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2}{2} \|p_k - x_{k-1}\|^2 \\
&\quad + \alpha_k [\langle g(p_k), y - q_k\rangle + \langle l_h(x_{k-1}, p_k), z - r_k\rangle] \\
&\le 1 - \alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} \\
&\quad + \alpha_k [\langle g(p_k), y - q_k\rangle + \langle l_h(x_{k-1}, p_k), z - r_k\rangle], \ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+.
\end{aligned} \tag{3.21}
$$

Multiplying both sides of (3.18) and (3.19) by α_k and summing them up with the above

inequality, we have

$$
\begin{aligned}
Q(w_k, w) \leq (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} \\
+ \alpha_k \langle g(p_k) - \tilde{g}_k), y - q_k \rangle + \alpha_k \langle l_h(x_{k-1}, p_k) - \tilde{h}_k, z - r_k \rangle \\
+ \tfrac{\alpha_k \tau_k}{2} [\|y - q_{k-1}\|_2^2 - \|y - q_k\|_2^2 - \|q_k - q_{k-1}\|_2^2] \\
+ \tfrac{\alpha_k \tau_k}{2} [\|z - r_{k-1}\|_2^2 - \|z - r_k\|_2^2 - \|r_k - r_{k-1}\|_2^2], \ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}_+^d.
\end{aligned} \tag{3.22}
$$

Now observe that by the definition of $\tilde{g}_k$ in (3.9) and the fact that $g(x) = Ax - b$, we have

$$
\begin{aligned}
&\langle g(p_k) - \tilde{g}_k), y - q_k \rangle - \tfrac{\tau_k}{2} \|q_k - q_{k-1}\|_2^2 \\
&= \langle A[(p_k - p_{k-1}) - \lambda_k (p_{k-1} - p_{k-2})], y - q_k \rangle - \tfrac{\tau_k}{2} \|q_k - q_{k-1}\|_2^2 \\
&= \langle A(p_k - p_{k-1}), y - q_k \rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1} \rangle \\
&\quad + \lambda_k \langle A(p_{k-1} - p_{k-2}), q_k - q_{k-1} \rangle - \tfrac{\tau_k}{2} \|q_k - q_{k-1}\|_2^2 \\
&\leq \langle A(p_k - p_{k-1}), y - q_k \rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1} \rangle \\
&\quad + \tfrac{\lambda_k^2}{2\tau_k} \|A\|^2 \|p_k - p_{k-1}\|_2^2 \\
&\leq \langle A(p_k - p_{k-1}), y - q_k \rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1} \rangle + \tfrac{\lambda_k^2}{2\tau_k} \|A\|^2 D_X^2,
\end{aligned} \tag{3.23}
$$

where the first inequality follows from Young's inequality and the last one follows from the definition of D_X in (3.4). In addition, by the definition of $\tilde{h}_k$ in (3.10), we have

$$
\begin{aligned}
&\langle l_h(x_{k-1}, p_k) - \tilde{h}_k, z - r_k \rangle - \tfrac{\tau_k}{2} \|r_k - r_{k-1}\|_2^2 \\
&\leq \langle l_h(x_{k-1}, p_k) - l_h(x_{k-2}, p_{k-1}), z - r_k \rangle - \lambda_k \langle l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2}), z - r_{k-1} \rangle \\
&\quad + \lambda_k \langle l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2}), r_k - r_{k-1} \rangle - \tfrac{\tau_k}{2} \|r_k - r_{k-1}\|_2^2 \\
&\leq \langle l_h(x_{k-1}, p_k) - l_h(x_{k-2}, p_{k-1}), z - r_k \rangle \\
&\quad - \lambda_k \langle l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2}), z - r_{k-1} \rangle + \tfrac{9\lambda_k^2 \bar{M}_h^2 D_X^2}{2\tau_k},
\end{aligned} \tag{3.24}
$$

where the last inequality follows from

$$\begin{aligned}
&\lambda_k\langle l_h(x_{k-2},p_{k-1})-l_h(x_{k-3},p_{k-2}),r_k-r_{k-1}\rangle-\tfrac{\tau_k}{2}\|r_k-r_{k-1}\|_2^2\\
&\le \tfrac{\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d[l_{h_i}(x_{k-2},p_{k-1})-l_{h_i}(x_{k-3},p_{k-2})]^2\\
&= \tfrac{\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d[h_i(x_{k-2})-h_i(x_{k-3})+\langle\nabla h_i(x_{k-2}),p_{k-1}-x_{k-2}\rangle+\langle\nabla h_i(x_{k-3}),p_{k-2}-x_{k-3}\rangle]^2\\
&\le \tfrac{9\lambda_k^2D_X^2}{2\tau_k}\textstyle\sum_{i=1}^d M_{h,i}^2=\tfrac{9\lambda_k^2\bar{M}_h^2D_X^2}{2\tau_k}. \qquad (3.25)
\end{aligned}$$

The result then follows by plugging relations (3.23) and (3.24) into (3.22). ∎

We are now ready to establish the main convergence properties for the CoexCG method.

Theorem 4 *Let Γ_k be defined in* (3.20) *and assume that the algorithmic parameters α_k,τ_k and λ_k in the CoexCG method satisfy*

$$\alpha_1=1,\ \tfrac{\lambda_k\alpha_k}{\Gamma_k}=\tfrac{\alpha_{k-1}}{\Gamma_{k-1}}\ \textit{and}\ \tfrac{\alpha_k\tau_k}{\Gamma_k}\le\tfrac{\alpha_{k-1}\tau_{k-1}}{\Gamma_{k-1}},\forall k\ge 2. \qquad (3.26)$$

Then we have

$$\begin{aligned}
Q(w_N,w)\le\Gamma_N\textstyle\sum_{k=1}^N\left[\tfrac{(L_f+z^TL_h)\alpha_k^2D_X^2}{2\Gamma_k}+\tfrac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}\right]\\
+\tfrac{\alpha_N(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_N}+\tfrac{\tau_1\Gamma_N}{2}\|y-q_0\|_2^2+\tfrac{\tau_1\Gamma_N}{2}\|z-r_0\|_2^2,\ \forall w\in X\times\mathbb{R}^m\times\mathbb{R}^d_+, \qquad (3.27)
\end{aligned}$$

where D_X is defined in (3.4). As a consequence, we have

$$\begin{aligned}
f(x_N)-f(x^*)\le\Gamma_N\textstyle\sum_{k=1}^N\left[\tfrac{L_f\alpha_k^2D_X^2}{2\Gamma_k}+\tfrac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}\right]\\
+\tfrac{\alpha_N(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_N}+\tfrac{\tau_1\Gamma_N}{2}(\|q_0\|_2^2+\|r_0\|_2^2) \qquad (3.28)
\end{aligned}$$

and

$$\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \leq \Gamma_N \sum_{k=1}^N \left[\frac{[L_f+(\|z^*\|_2+1)\bar{L}_h]\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}\right]$$
$$+ \frac{\alpha_N(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_N} + \tau_1\Gamma_N[(\|y^*\|_2+1)^2 + \|q_0\|_2^2 + (\|z^*\|_2+1)^2 + \|r_0\|_2^2], \quad (3.29)$$

where (x^*, y^*, z^*) *denotes a triple of optimal solutions for problem (3.6).*

Proof. It follows from Lemma 2 and Proposition 3 that

$$\begin{aligned}
\tfrac{Q(w_N,w)}{\Gamma_N} \leq & (1-\alpha_1)Q(w_0,w) + \textstyle\sum_{k=1}^N [\frac{(L_f+z^TL_h)\alpha_k^2D_X^2}{2\Gamma_k} + \frac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k}[\langle A(p_k-p_{k-1}), y-q_k\rangle - \lambda_k\langle A(p_{k-1}-p_{k-2}), y-q_{k-1}\rangle] \\
& + \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k}[\langle l_h(x_{k-1},p_k) - l_h(x_{k-2},p_{k-1}), z-r_k\rangle \\
& \quad - \lambda_k\langle l_h(x_{k-2},p_{k-1}) - l_h(x_{k-3},p_{k-2}), z-r_{k-1}\rangle] \\
& + \textstyle\sum_{k=1}^N \frac{\alpha_k\tau_k}{2\Gamma_k}[\|y-q_{k-1}\|_2^2 - \|y-q_k\|_2^2 + \|z-r_{k-1}\|_2^2 - \|z-r_k\|_2^2],
\end{aligned}$$

which, in view of (3.26), then implies that

$$\begin{aligned}
Q(w_N,w) \leq & \Gamma_N \textstyle\sum_{k=1}^N [\frac{(L_f+z^TL_h)\alpha_k^2D_X^2}{2\Gamma_k} + \frac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \alpha_N\langle A(p_N-p_{N-1}), y-q_N\rangle - \tfrac{\alpha_N\tau_N}{2}\|y-q_N\|_2^2 \\
& + \alpha_N\langle l_h(x_{N-1},p_N) - l_h(x_{N-2},p_{N-1}), z-r_N\rangle - \tfrac{\alpha_N\tau_N}{2}\|z-r_N\|_2^2 \\
& + \tfrac{\alpha_1\tau_1\Gamma_N}{2}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2] \\
\leq & \Gamma_N \textstyle\sum_{k=1}^N [\frac{(L_f+z^TL_h)\alpha_k^2D_X^2}{2\Gamma_k} + \frac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \tfrac{\alpha_N}{2\tau_N}\|A\|^2\|p_N-p_{N-1}\|_2^2 + \tfrac{9\bar{M}_h^2\alpha_N D_X^2}{2\tau_N} \\
& + \tfrac{\alpha_1\tau_1\Gamma_N}{2}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2],
\end{aligned}$$

where the last relation follows from Young's inequality and a result similar to (3.25). The result in (3.27) then immediately follows from the above inequality.

Note that by the definition of $Q(w_k, w)$ in (3.17), and the facts that $g(x^*) = 0$ and $h(x^*) \leq 0$, we have $f(x_N) - f(x^*) \leq Q(w_N, (x^*, 0, 0))$. Using this observation and fixing $x = x^*, y = 0, z = 0$ in (3.27), we obtain (3.28). Now let us denote

$$\hat{y}_N := (\|y^*\|_2 + 1)\tfrac{g(x_N)}{\|g(x_N)\|_2}, \tag{3.30}$$

$$\hat{z}_N := (\|z^*\|_2 + 1)\tfrac{[h(x_N)]_+}{\|[h(x_N)]_+\|_2}, \tag{3.31}$$

$$\hat{w}_N^* := (x^*, \hat{y}_N, \hat{z}_N). \tag{3.32}$$

Note that by the optimality condition of (3.6), we have

$$\begin{aligned} 0 \leq Q(w_N, w^*) &= f(x_N) - f(x^*) + \langle g(x_N), y^* \rangle + \langle h(x_N), z^* \rangle \\ &\leq f(x_N) - f(x^*) + \|g(x_N)\|_2 \cdot \|y^*\|_2 + \|[h(x_N)]_+\|_2 \cdot \|z^*\|_2. \end{aligned}$$

In addition, using the fact that $g(x^*) = 0$ and $\langle h(x^*), \hat{z}_N \rangle \leq 0$, we have

$$\begin{aligned} Q(w_N, \hat{w}_N^*) &\geq f(x_N) - f(x^*) + \langle g(x_N), \hat{y}_N \rangle + \langle h(x_N), \hat{z}_N \rangle \\ &= f(x_N) - f(x^*) + \|g(x_N)\|_2(\|y^*\|_2 + 1) + \|[h(x_N)]_+\|_2(\|z^*\|_2 + 1). \end{aligned}$$

Combining the previous two observations, we conclude that

$$\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \leq Q(w_N, \hat{w}_N^*). \tag{3.33}$$

The previous conclusion, together with (3.27) and the facts that

$$\|\hat{y}_N - q_0\|_2^2 \leq 2[\|\hat{y}_N\|_2^2 + \|q_0\|_2^2] = 2[(\|y^*\|_2 + 1)^2 + \|q_0\|_2^2], \tag{3.34}$$

$$\|\hat{z}_N - r_0\|_2^2 \leq 2[\hat{z}_N\|_2^2 + \|r_0\|_2^2] = 2[(\|z^*\|_2 + 1)^2 + \|r_0\|_2^2], \tag{3.35}$$

$$\hat{z}_N^T L_h \leq \|\hat{z}_N^T\|_2 \|L_h\|_2 = (\|z^*\|_2 + 1)\bar{L}_h, \tag{3.36}$$

then imply that

$$
\begin{aligned}
&\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \\
&\le \Gamma_N \textstyle\sum_{k=1}^N \left[\frac{(L_f + \hat{z}_N^T L_h)\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \right] \\
&\quad + \frac{\alpha_N (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_N} + \frac{\tau_1 \Gamma_N}{2} \|\hat{y}_N - q_0\|_2^2 + \frac{\tau_1 \Gamma_N}{2} \|\hat{z}_N - r_0\|_2^2 \\
&\le \Gamma_N \textstyle\sum_{k=1}^N \left[\frac{[L_f + (\|z^*\|_2 + 1)\bar{L}_h]\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \right] \\
&\quad + \frac{\alpha_N (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_N} + \tau_1 \Gamma_N [(\|y^*\|_2 + 1)^2 + \|q_0\|_2^2 + (\|z^*\|_2 + 1)^2 + \|r_0\|_2^2].
\end{aligned}
$$

■

Below we provide a specific selection of the algorithmic parameters α_k, λ_k and τ_k and establish the associated rate of convergence for the CoexCG method.

Corollary 4.1 *If the number of iterations N is fixed a priori, and*

$$
\alpha_k = \tfrac{2}{k+1}, \lambda_k = \tfrac{k-1}{k}, \ \tau_k = \tfrac{N^{3/2}}{k} D_X \sqrt{9\|M_h\|^2 + \|A\|^2}, k = 1, \ldots, N, \tag{3.37}
$$

then we have

$$
\begin{aligned}
&Q(w_N, w) \le \tfrac{2(L_f + z^T L_h) D_X^2}{N+1} + \tfrac{D_X \sqrt{9\bar{M}_h^2 + \|A\|^2}}{\sqrt{N}} \left(\|y - q_0\|_2^2 + \|z - r_0\|_2^2 + 1 \right), \\
&\qquad \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+, \qquad (3.38) \\
&f(x_N) - f(x^*) \le \tfrac{2L_f D_X^2}{N+1} + \tfrac{D_X \sqrt{9\bar{M}_h^2 + \|A\|^2}}{\sqrt{N}} (\|(q_0; r_0)\|_2^2 + 1), \qquad (3.39) \\
&\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \le \tfrac{2[L_f + (\|z^*\|_2 + 1)\bar{L}_h] D_X^2}{N+1} \\
&\quad + \tfrac{2D_X \sqrt{9\bar{M}_h^2 + \|A\|^2}}{\sqrt{N}} [2\|(y^*; z^*)\|_2^2 + \|(q_0; r_0)\|_2^2 + 5]. \qquad (3.40)
\end{aligned}
$$

Proof. By (3.20) and the definition of α_k in (3.37), we have $\Gamma_k = 2/[k(k+1)]$ and $\alpha_k / \Gamma_k = k$. We can easily see from these identities and (3.37) that the conditions in (3.26)

hold. It is also easy to verify that

$$\begin{aligned}\textstyle\sum_{k=1}^{N}\frac{\alpha_k^2}{\Gamma_k} &= 2\textstyle\sum_{k=1}^{N}\frac{k}{k+1} \le 2N,\\ \textstyle\sum_{k=1}^{N}\frac{\alpha_k\lambda_k^2}{\tau_k\Gamma_k} &= \frac{\sum_{k=1}^{N}(k-1)^2}{2N^{3/2}D_X\sqrt{9\|M_h\|^2+\|A\|^2}} \le \frac{N^{3/2}}{6D_X\sqrt{9\|M_h\|^2+\|A\|^2}}.\end{aligned}$$

Using these relations in (3.27), (3.28) and (3.29), we conclude that

$$\begin{aligned}Q(w_N,w) &\le \tfrac{2(L_f+z^TL_h)D_X^2}{N+1} + \tfrac{\sqrt{N}D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{6(N+1)}\\ &\quad + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{(N+1)\sqrt{N}} + \tfrac{\sqrt{N}D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{N+1}(\|y-q_0\|_2^2+\|z-r_0\|_2^2)\\ &= \tfrac{2(L_f+z^TL_h)D_X^2}{N+1} + [\tfrac{\sqrt{N}}{6(N+1)} + \tfrac{1}{(N+1)\sqrt{N}} + \tfrac{\sqrt{N}}{N+1}(\|y-q_0\|_2^2\\ &\quad + \|z-r_0\|_2^2)]D_X\sqrt{9\bar{M}_h^2+\|A\|^2}\\ &\le \tfrac{2(L_f+z^TL_h)D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}\left(\|y-q_0\|_2^2+\|z-r_0\|_2^2+1\right),\\ f(x_N)-f(x^*) &\le \tfrac{2L_fD_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}(\|q_0\|_2^2+\|r_0\|_2^2+1),\end{aligned}$$

and

$$\begin{aligned}&\|g(x_N)\|_2+\|[h(x_N)]_+\|_2\\ &\le \tfrac{2[L_f+(\|z^*\|_2+1)\bar{L}_h]D_X^2}{N+1} + \tfrac{\sqrt{N}D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{6(N+1)} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{(N+1)\sqrt{N}}\\ &\quad + \tfrac{2\sqrt{N}D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{N+1}[(\|y^*\|_2+1)^2+\|q_0\|_2^2+(\|z^*\|_2+1)^2+\|r_0\|_2^2]\\ &\le \tfrac{2[L_f+(\|z^*\|_2+1)\bar{L}_h]D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}[1+2(\|y^*\|_2+1)^2+2\|q_0\|_2^2\\ &\quad + 2(\|z^*\|_2+1)^2+2\|r_0\|_2^2]\\ &\le \tfrac{2[L_f+(\|z^*\|_2+1)\bar{L}_h]D_X^2}{N+1} + \tfrac{2D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}[2(\|y^*\|_2^2+\|z^*\|_2^2)+\|q_0\|_2^2+\|r_0\|_2^2+5].\end{aligned}$$

■

A few remarks about the results obtained in Theorem 4 and Corollary 4.1 are in place. Firstly, in view of (3.38), the gap function $Q(w_N, w)$ converges to 0 with the rate of convergence given by $\mathcal{O}(1/\sqrt{N})$. This bound has been shown to be not improvable in [28] (see

also Chapter 7 of [44]). Secondly, in view of (3.39) and (3.40), the number of iterations required by the CoexCG method to find a ϵ-solution of problem (1.8), i.e., a point $\bar{x} \in X$ s.t. $f(\bar{x}) - f(x^*) \leq \epsilon$ and $\|g(\bar{x})\|_2 + \|[h(\bar{x})]_+\|_2 \leq \epsilon$, is bounded by $\mathcal{O}(1/\epsilon^2)$. Thirdly, it is interesting to observe that in both (3.39) and (3.40), the Lipschitz constants L_f and $\bar{L}_h$ do not impact too much the rate of convergence of the CoexCG method, since both of them appear only in the non-dominant terms. We will explore further this property of the CoexCG method in order to solve problems with certain nonsmooth objective and constraint functions. Finally, it is worth noting that in the parameter setting (3.37), we need to fix the total number of iterations N in advance. This is not desirable for the implementation of the CoexCG method, especially for the situation when one has finished the scheduled N iterations, but then realizes that a more accurate solution is needed. In this case, one has to completely restart the CoexCG method with a different parameter setting that depends on the modified iteration limit. We will discuss how to address this issue in Section 3.3.

3.2.2 Structured nonsmooth functions

In this subsection, we still consider problem (1.8), but the objective function f and constraint functions h_i are not necessarily differentiable. More specifically, we assume that $f(\cdot)$ and $h_i(\cdot)$ are given in the following form:

$$\begin{aligned} f(x) &= \max_{q \in Q} \{\langle Bx, q\rangle - \hat{f}(q)\}, \\ h_i(x) &= \max_{s \in S_i} \{\langle C_i x, s\rangle - \hat{h}_i(s)\}, i = 1, \ldots, d, \end{aligned} \tag{3.41}$$

where $Q \subseteq \mathbb{R}^{m_0}$ and $S \subseteq \mathbb{R}^{m_i}$ are closed convex sets, and $\hat{f}$ and $\hat{h}_i$ are simple convex functions. Many nonsmooth functions can be represented in this form (see [89]). In this chapter, we assume that $\hat{f}$ and $\hat{h}_i$ are possibly strongly convex w.r.t. the given norms in the

respective spaces, i.e..

$$\hat{f}(q_1) - \hat{f}(q_2) - \langle \hat{f}'(q_2), q_1 - q_2 \rangle \geq \tfrac{\mu_0}{2}\|q_1 - q_2\|^2, \forall q_1, q_2 \in Q \quad (3.42)$$

$$\hat{h}_i(s_1) - \hat{h}_i(s_2) - \langle \hat{h}_i'(s_2), s_1 - s_2 \rangle \geq \tfrac{\mu_i}{2}\|s_1 - s_2\|^2, \forall s_1, s_2 \in S_i, i = 1, \ldots, d, \quad (3.43)$$

for some $\mu_i \geq 0$. If $\mu_0 > 0$ (resp., $\mu_i > 0$), then f (resp., h_i) must be differentiable with Lipschitz continuous gradients. Therefore, our nonsmooth formulation in (3.41) allows either the objective and/or some constraint functions to be smooth.

Our goal in this subsection is to generalize the CoexCG method to solve these structured nonsmooth convex optimization problems. In fact, we show that that the number of CoexCG iterations required to solve these problems is in the same order of magnitude as if f and h_i's are smooth convex functions.

Since f and h_i are possibly not differentiable, we cannot directly apply the CoexCG algorithm to solve problem (1.8). However, as pointed out by Nesterov [89], these nonsmooth functions can be closely approximated by smooth convex ones. Let us first consider the objective function f. Assume that $u : Q \to \mathbb{R}$ is a given strongly convex function with modulus 1 w.r.t. a given norm $\|\cdot\|$ in $\mathbb{R}^{m_0}$, i.e.,

$$u(q_1) \geq u(q_2) + \langle u'(q_2), q_1 - q_2 \rangle + \tfrac{1}{2}\|q_1 - q_2\|^2, \forall q_1, q_2 \in Q.$$

Let us denote $c_u := \mathrm{argmin}_{q \in Q} u(y)$, $U(q) := u(q) - u(c_u) - \langle \nabla u(c_u), q - c_u \rangle$ and

$$D_U := [\max_{q \in Q} U(y)]^{1/2}, \quad (3.44)$$

and define

$$f_{\eta_0}(x) := \max_{q \in Q}\{\langle Bx, q \rangle - \hat{f}(q) - \eta_0 U(q)]\} \quad (3.45)$$

for some $\eta_0 \geq 0$. Then, we can show that f_{η_0} is differentiable and its gradients satisfy (see

[89])

$$\|\nabla f_{\eta_0}(x_1) - \nabla f_{\eta_0}(x_2)\|_* \leq L_{f,\eta}\|x_1 - x_2\|,\ \forall x_1, x_2 \in X \ \text{ with } \ L_{f,\eta} := \tfrac{\|B\|^2}{\mu_0+\eta_0}. \quad (3.46)$$

In addition, we have

$$f_{\eta_0}(x) \leq f(x) \leq f_{\eta_0}(x) + \eta_0 D_U^2,\ \forall x \in X. \quad (3.47)$$

In our algorithmic scheme, we will set $\eta_0 = 0$ whenever $\hat{f}$ is strongly convex, i.e., $\mu_0 > 0$.

Similarly, let us assume that $v_i : S_i \to \mathbb{R}$ are strongly convex with modulus 1 w.r.t. a given norm $\|\cdot\|$ in $\mathbb{R}^{m_i}$, $i = 1, \ldots, d$. Also let us denote $c_{v_i} := \operatorname{argmin}_{s\in S_i} v_i(s)$, $V_i(s) := v_i(s) - v_i(c_{v_i}) - \langle \nabla v_i(c_{v_i}), s - c_{v_i}\rangle$ and

$$D_{V_i} := [\max_{s\in S_i} V_i(s)]^{1/2}, \quad (3.48)$$

and define

$$h_{i,\eta_i}(x) = \max_{s\in S_i}\{\langle C_i x, s\rangle - \hat{h}_i(s) - \eta_i V_i(s)\} \quad (3.49)$$

for some $\eta_i \geq 0$. We can show that for all $i = 1, \ldots, d$,

$$\|\nabla h_{i,\eta_i}(x_1) - \nabla h_{i,\eta_i}(x_2)\|_* \leq \tfrac{\|C_i\|^2}{\mu_i+\eta_i}\|x_1 - x_2\|,\ \forall x_1, x_2 \in X, \quad (3.50)$$

$$h_{i,\eta_i}(x) \leq h_i(x) \leq h_{i,\eta_i}(x) + \eta_i D_{V_i}^2,\ \forall x \in X. \quad (3.51)$$

In our algorithmic scheme, we will set $\eta_i = 0$ whenever $\hat{h}_i$ is strongly convex, i.e., $\mu_i > 0$. For notational convenience, we denote

$$h_\eta(x) := (h_{1,\eta_1}(x); \ldots; h_{d,\eta_d}(x)),\ L_{h,\eta} := (\tfrac{\|C_1\|^2}{\mu_{\hat{h}_1}+\eta_1}; \ldots; \tfrac{\|C_d\|^2}{\mu_{\hat{h}_d}+\eta_d}) \ \text{ and } \ \bar{L}_{h,\eta} := \|L_{h,\eta}\|_2. \quad (3.52)$$

Different from the objective function, we need to show that the gradient of the h_{i,η_i} is

bounded. Note that the boundedness of the gradients for smooth constraint functions (with $\mu_i > 0$ and hence $\eta_i = 0$) follows from the boundedness of X (see Section 3.2.1). For those nonsmooth constraint functions h_i (with $\mu_i = 0$), we need to assume that S_i's are compact. For a given $x \in X$, let $s^*(x)$ be the optimal solution of (3.49). Then

$$\begin{aligned}\|\nabla h_{i,\eta_i}(x)\|_* = \|C_i^T \cdot s^*(x)\|_* &\le \|C_i\|\|s^*(x)\| \\ &\le \|C_i\|(\|c_{v_i}\| + \|s^*(x) - c_{v_i}\|) \\ &\le \|C_i\|(\|c_{v_i}\| + \sqrt{2}D_{V_i}) =: M_{C_i,V_i}, i = 1, \ldots, d. \end{aligned} \tag{3.53}$$

For notational convenience, we also denote

$$\bar{M}_{C,V} := \sqrt{\textstyle\sum_{i=1}^d M_{C_i,V_i}^2}. \tag{3.54}$$

Observe that the Lipschitz constants M_{C_i,V_i} defined in (3.53) do not depend on the smoothing parameters η_i, $i = 1, \ldots, d$. This fact will be important for us to derive the complexity bound of the CoexCG method for solving convex optimization problems with nonsmooth function constraints.

Instead of solving the original problem (1.8), we suggest to apply the CoexCG method to the smooth approximation problem

$$\begin{aligned} \min \quad & f_{\eta_0}(x) \\ \text{s.t.} \quad & g(x) = 0, \\ & h_{i,\eta_i}(x) \le 0, \forall i = 1, \ldots, d, \\ & x \in X. \end{aligned} \tag{3.55}$$

More specifically, we replace the linear approximation functions l_h and l_f used in (3.10) and (3.13) by $l_{h_{i,\eta_i}}$ and $l_{f_{\eta_o}}$, respectively. However, we will establish the convergence of this method in terms of the solution of the original problem in (1.8) rather than the approx-

imation problem in (3.55). Our convergence analysis below exploits the smoothness of f_{η_0} (resp., h_{i,η_i}), the closeness between f and f_{η_0} (resp., h_i and h_{i,η_i}), and also importantly, the fact that $h_{i,\eta_i}(x)$ underestimates $h_i(x)$ for all $x \in X$.

Theorem 5 *Consider the CoexCG method applied to the smooth approximation problem (3.55). Assume that the number of iterations N is fixed a priori, and that the parameters $\{\alpha_k\}, \{\tau_k\}$ and $\{\lambda_k\}$ are set to* (3.37) *with $\bar{M}_h$ replaced by $\bar{M}_{C,V}$ in (3.54). Then we have*

$$f(x_N) - f(x^*) \le \tfrac{2L_{f,\eta}D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(\|q_0\|_2^2 + \|r_0\|_2^2 + 1\right) + \eta_0 D_U^2, \tag{3.56}$$

$$\begin{aligned}\|[h(x_N)]_+\| + \|Ax_N\| \le{}& \tfrac{2[L_{f,\eta}+(\|z^*\|_2+1)\bar{L}_{h,\eta}])D_X^2}{N+1} + \tfrac{2D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}(2\|(y^*;z^*)\|_2^2 \\ &+ \|(q_0;r_0)\|_2^2 + 5) + \eta_0 D_U^2 + (\|z^*\|_2+1)(\textstyle\sum_{i=1}^d(\eta_i D_{V_i}^2)^2)^{1/2},\end{aligned} \tag{3.57}$$

where (x^*, y^*, z^*) is a triple of optimal solutions for problem (3.6), $L_{f,\eta}$ and $\bar{L}_{h,\eta}$ are defined in (3.46) and (3.52), respectively, and D_X, D_U and D_{V_i} are defined in (3.4), (3.44) and (3.48), respectively.

Proof. Denote $Q_\eta(w_N, w) := f_{\eta_0}(x_N) - f_{\eta_0}(x) + \langle g(x_N), y\rangle - \langle g(x), y_N\rangle + \langle h_\eta(x_N), z\rangle - \langle h_\eta(x), z_N\rangle$. In view of Corollary 4.1, we have

$$Q_\eta(w_N, w) \le \tfrac{(L_{f,\eta}+z^T L_{h,\eta})D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(\|y-q_0\|_2^2 + \|z-r_0\|_2^2 + 1\right) \tag{3.58}$$

for any $w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$. Using the relations in (3.47) and (3.51), and the fact that $z, z_N \in \mathbb{R}^d_+$, we can see that

$$\begin{aligned}Q(w_N, w) &\le Q_\eta(w_N, w) + \eta_0 D_U^2 + \textstyle\sum_{i=1}^d(\eta_i z_i D_{V_i}^2) \\ &\le Q_\eta(w_N, w) + \eta_0 D_U^2 + \|z\|_2(\textstyle\sum_{i=1}^d(\eta_i D_{V_i}^2)^2)^{1/2},\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+.\end{aligned} \tag{3.59}$$

By letting $x = x^*$, $y = 0$ and $z = 0$, we have

$$f(x_N) - f(x^*) \le Q(w_N, z) \le Q_\eta(z_N, z) + \eta_0 D_U^2,$$

which, in view of (3.58), then implies (3.56). Now let $\hat{w}_N^*$ be defined in (3.32). By (3.33), (3.58) and (3.59), we have

$$\begin{aligned}
&\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \le Q(w_N, \hat{w}_N^*) \\
&\le Q_\eta(w_N, \hat{w}_N^*) + \eta_0 D_U^2 + \|\hat{z}_N\|_2 (\textstyle\sum_{i=1}^d (\eta_i D_{V_i}^2)^2)^{1/2} \\
&\le \tfrac{(L_{f,\eta} + \hat{z}_N^T L_{h,\eta}) D_X^2}{N+1} + \tfrac{D_X \sqrt{9\bar{M}_{C,V}^2 + \|A\|^2}}{\sqrt{N}} \left(\|\hat{y}_N - q_0\|_2^2 + \|\hat{z}_N - r_0\|_2^2 + 1 \right) \\
&\quad + \eta_0 D_U^2 + \|\hat{z}_N\|_2 (\textstyle\sum_{i=1}^d (\eta_i D_{V_i}^2)^2)^{1/2} \\
&\le \tfrac{[L_{f,\eta} + (\|z^*\|_2 + 1)\bar{L}_{h,\eta}]) D_X^2}{N+1} + \tfrac{2 D_X \sqrt{9\bar{M}_{C,V}^2 + \|A\|^2}}{\sqrt{N}} \left(2\|(y^*; z^*)\|_2^2 + \|(q_0; r_0)\|_2^2 + 5 \right) \\
&\quad + \eta_0 D_U^2 + (\|z^*\|_2 + 1)(\textstyle\sum_{i=1}^d (\eta_i D_{V_i}^2)^2)^{1/2},
\end{aligned}$$

where the last inequality follows from the bounds in (3.34) and (3.35), and the facts that $\|\hat{z}_N\|_2 \le \|z^*\|_2 + 1$ and $\hat{z}_N^T L_{h,\eta} \le \|\hat{z}_N^T\|_2 \|L_{h,\eta}\|_2 = (\|z^*\|_2 + 1)\bar{L}_{h,\eta}$. ∎

We now specify the selection of the smoothing parameters η_i, $i = 0, \ldots, d$. We consider only the most challenging case when the objective and all constraint functions are nonsmooth and establish the rate of convergence of the aforementioned CoexCG method for nonsmooth convex optimization.

Corollary 5.1 *Suppose that the smoothing parameters in problem (3.55) are set to*

$$\eta_0 = \tfrac{\|B\| D_X}{D_U \sqrt{N}} \ \ \textit{and} \ \ \eta_i = \tfrac{\|C_i\| D_X}{D_{V_i} \sqrt{N}}, i = 1, \ldots, d. \tag{3.60}$$

Then under the same premise of Theorem 5, we have

$$f(x_N) - f(x^*) \le \frac{3D_X D_U \|B\|}{\sqrt{N}} + \frac{D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(\|q_0\|_2^2 + \|r_0\|_2^2 + 1\right), \tag{3.61}$$

$$\begin{aligned}\|[h(x_N)]_+\| + \|Ax_N\| \le{}& \frac{3D_X D_U \|B\|}{\sqrt{N}} + \frac{2(\|z^*\|_2+1)D_X\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{\sqrt{N}} \\ &+ \frac{2D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(2\|(y^*;z^*)\|_2^2 + \|(q_0;r_0)\|_2^2 + 5\right).\end{aligned} \tag{3.62}$$

Proof. It follows from (3.46), (3.52) and (3.60) that

$$\begin{aligned}L_{f,\eta} &= \tfrac{\|B\|^2}{\eta_0} = \tfrac{D_U\|B\|\sqrt{N}}{D_X}, \\ \bar{L}_{h,\eta} &= \sqrt{\textstyle\sum_{i=1}^d \left(\tfrac{\|C_i\|^2}{\eta_i}\right)^2} = \sqrt{\textstyle\sum_{i=1}^d \left(\tfrac{D_{V_i}\|C_i\|\sqrt{N}}{D_X}\right)^2} = \tfrac{\sqrt{N}\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{D_X}.\end{aligned}$$

Also notice that

$$\begin{aligned}\eta_0 D_U^2 &= \tfrac{D_X D_U\|B\|}{\sqrt{N}} \\ (\textstyle\sum_{i=1}^d (\eta_i D_{V_i}^2)^2)^{1/2} &= \left(\textstyle\sum_{i=1}^d \tfrac{\|C_i\|^2 D_X^2 D_{V_i}^2}{N}\right)^{1/2} = \tfrac{D_X\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{\sqrt{N}}.\end{aligned}$$

Using these identities and the assumptions in (3.56) and (3.57), we have

$$\begin{aligned}f(x_N) - f(x^*) &\le \tfrac{2D_X D_U\|B\|}{\sqrt{N+1}} + \tfrac{D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(\|q_0\|_2^2 + \|r_0\|_2^2 + 1\right) + \tfrac{D_X D_U\|B\|}{\sqrt{N}} \\ &\le \tfrac{3D_X D_U\|B\|}{\sqrt{N}} + \tfrac{D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(\|q_0\|_2^2 + \|r_0\|_2^2 + 1\right), \\ \|[h(x_N)]_+\| + \|Ax_N\| &\le \tfrac{2D_X D_U\|B\|}{\sqrt{N+1}} + \tfrac{(\|z^*\|_2+1)D_X\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{\sqrt{N+1}} \\ &\quad + \tfrac{2D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(2\|(y^*;z^*)\|_2^2 + \|(q_0;r_0)\|_2^2 + 5\right) \\ &\quad + \tfrac{D_X D_U\|B\|}{\sqrt{N}} + \tfrac{(\|z^*\|_2+1)D_X\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{\sqrt{N}} \\ &\le \tfrac{3D_X D_U\|B\|}{\sqrt{N}} + \tfrac{2(\|z^*\|_2+1)D_X\sqrt{\sum_{i=1}^d (D_{V_i}\|C_i\|)^2}}{\sqrt{N}} \\ &\quad + \tfrac{2D_X\sqrt{9\bar{M}_{C,V}^2+\|A\|^2}}{\sqrt{N}}\left(2\|(y^*;z^*)\|_2^2 + \|(q_0;r_0)\|_2^2 + 5\right).\end{aligned}$$

■

We add a few remarks about the results obtained in Theorem 5 and Corollary 5.1. Firstly, in view of Corollary 5.1, even if f and h_i are nonsmooth functions, the number of CoexCG iterations required to find an ϵ-solution of problem (1.8) is still bounded by $\mathcal{O}(1/\epsilon^2)$. Therefore, by utilizing the structural information of f and h_i, the CoexCG can solve this type of nonsmooth problem efficiently as if they are smooth functions. Secondly, if either the objective function or some constraint functions are smooth, we can set the corresponding smoothing parameter to be zero and obtain slightly improved complexity bounds than those in Corollary 5.1. Thirdly, similar to the CoexCG method applied for solving problem (1.8) with smooth objective and constraint functions, we need to fix the number of iterations N in advance when specifying the algorithmic parameters and smoothing parameters. We will address this issue in next section.

3.3 Constraint-extrapolated and dual-regularized conditional gradient method

One critical shortcoming associated with the basic version of the CoexCG method is that we need to fix the number of iterations N a priori. Our goal in this section is to develop a variant of CoexCG which does not have this requirement. We consider the case when f and h_i are smooth and structured nonsmooth functions, respectively, in Subsections 3.3.1 and 3.3.2.

3.3.1 Smooth functions

In order to remove the assumption of fixing N a priori, we suggest to modify the dual projection steps (3.11) and (3.12) in the CoexCG method. More specifically, we add an additional regularization term with diminishing weights into these steps. This variant of CoexCG is formally described in Algorithm 4.

Algorithm 4 Constraint-**ex**trapolated and **Du**al-**r**egularized Conditional Gradient (CoexDurCG)

The algorithm is the same as CoexCG except that (3.11) and (3.12) are replaced by

$$q_k = \operatorname{argmin}_{y\in\mathbb{R}^m}\{\langle -\tilde{g}_k, y\rangle + \tfrac{\tau_k}{2}\|y - q_{k-1}\|_2^2 + \tfrac{\gamma_k}{2}\|y - q_0\|_2^2\}, \tag{3.63}$$

$$r_k = \operatorname{argmin}_{z\in\mathbb{R}^d_+}\{\langle -\tilde{h}_k, z\rangle + \tfrac{\tau_k}{2}\|z - r_{k-1}\|_2^2 + \tfrac{\gamma_k}{2}\|z - r_0\|_2^2\}, \tag{3.64}$$

for some $\gamma_k \geq 0$.

Clearly, we can write q_k and r_k in (3.63) and (3.64) equivalently as

$$\begin{aligned} q_k &= \tfrac{1}{\tau_k+\gamma_k}(\tau_k q_{k-1} + \gamma_k q_0 + \tilde{g}_k), \\ r_k &= \max\left\{\tfrac{1}{\tau_k+\gamma_k}(\tau_k r_{k-1} + \gamma_k r_0 + \tilde{h}_k), 0\right\} \end{aligned}$$

Similar to the CoexCG method, it is also possible to generalize CoexDurCG for solving problems with conic inequality constraints. The following result, whose proof can be found in Lemma 3.5 of [44], characterizes the optimality conditions for (3.63) and (3.64).

Lemma 6 *Let q_k and r_k be defined in (3.63) and (3.64), respectively. Then,*

$$\begin{aligned} &\langle -\tilde{g}_k, q_k - y\rangle + \tfrac{\tau_k}{2}\|q_k - q_{k-1}\|_2^2 + \tfrac{\gamma_k}{2}\|q_k - q_0\|_2^2 \\ &\quad \leq \tfrac{\tau_k}{2}\|y - q_{k-1}\|_2^2 - \tfrac{\tau_k+\gamma_k}{2}\|y - q_k\|_2^2 + \tfrac{\gamma_k}{2}\|y - q_0\|_2^2,\ \forall y \in \mathbb{R}^m, \end{aligned} \tag{3.65}$$

$$\begin{aligned} &\langle -\tilde{h}_k, r_k - z\rangle + \tfrac{\tau_k}{2}\|r_k - r_{k-1}\|_2^2 + \tfrac{\gamma_k}{2}\|r_k - r_0\|_2^2 \\ &\quad \leq \tfrac{\tau_k}{2}\|z - r_{k-1}\|_2^2 - \tfrac{\tau_k+\gamma_k}{2}\|z - r_k\|_2^2 + \tfrac{\gamma_k}{2}\|y - r_0\|_2^2,\ \forall z \in \mathbb{R}^d_+. \end{aligned} \tag{3.66}$$

We now establish an important recursion about the CoexDurCG method, which can be viewed as a counterpart of Proposition 3 for the CoexCG method.

Proposition 7 *For any $k > 1$, we have*

$$
\begin{aligned}
Q(w_k, w) \leq\ & (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} + \tfrac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k} \\
& + \alpha_k [\langle A(p_k - p_{k-1}), y - q_k \rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1} \rangle] \\
& + \alpha_k [\langle l_h(x_{k-1}, p_k) - l_h(x_{k-2}, p_{k-1}), z - r_k \rangle \\
& - \lambda_k \langle l_h(x_{k-2}, p_{k-1}) - l_h(x_{k-3}, p_{k-2}), z - r_{k-1} \rangle] \\
& + \tfrac{\alpha_k \tau_k}{2} (\|y - q_{k-1}\|_2^2 + \|z - r_{k-1}\|_2^2) - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2} (\|y - q_k\|_2^2 + \|z - r_k\|_2^2) \\
& + \tfrac{\alpha_k \gamma_k}{2} [\|y - q_0\|_2^2 + \|z - r_0\|_2^2],\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+,
\end{aligned}
$$

where D_X is defined in (3.4).

Proof. Multiplying both sides of (3.65) and (3.66) by α_k and summing them up with the inequality in (3.21), we have

$$
\begin{aligned}
Q(w_k, w) \leq\ & (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} \\
& + \alpha_k \langle g(p_k) - \tilde{g}_k), y - q_k \rangle + \alpha_k \langle l_h(x_{k-1}, p_k) - \tilde{h}_k, z - r_k \rangle \\
& + \tfrac{\alpha_k \tau_k}{2} [\|y - q_{k-1}\|_2^2 - \|q_k - q_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2} \|y - q_k\|_2^2 \\
& + \tfrac{\alpha_k \tau_k}{2} [\|z - r_{k-1}\|_2^2 - \|r_k - r_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2} \|z - r_k\|_2^2 \\
& + \tfrac{\alpha_k \gamma_k}{2} [\|y - q_0\|^2 - \|q_k - q_0\|^2] + \tfrac{\alpha_k \gamma_k}{2} [\|z - r_0\|^2 - \|z_k - r_0\|^2] \\
\leq\ & (1-\alpha_k) Q(w_{k-1}, w) + \tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2} \\
& + \alpha_k \langle g(p_k) - \tilde{g}_k), y - q_k \rangle + \alpha_k \langle l_h(x_{k-1}, p_k) - \tilde{h}_k, z - r_k \rangle \\
& + \tfrac{\alpha_k \tau_k}{2} [\|y - q_{k-1}\|_2^2 - \|q_k - q_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2} \|y - q_k\|_2^2 \\
& + \tfrac{\alpha_k \tau_k}{2} [\|z - r_{k-1}\|_2^2 - \|r_k - r_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2} \|z - r_k\|_2^2 \\
& + \tfrac{\alpha_k \gamma_k}{2} [\|y - q_0\|_2^2 + \|z - r_0\|_2^2],\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+. \qquad (3.67)
\end{aligned}
$$

The result then follows by plugging relations (3.23) and (3.24) into (3.67). ∎

We are now ready to establish the main convergence properties of the CoexDurCG

method.

Theorem 8 *Let Γ_k be defined in* (3.20) *and assume that the algorithmic parameters α_k, τ_k and λ_k in the CoexDurCG method satisfy*

$$\alpha_1 = 1,\ \tfrac{\lambda_k \alpha_k}{\Gamma_k} = \tfrac{\alpha_{k-1}}{\Gamma_{k-1}} \text{ and } \tfrac{\alpha_k \tau_k}{\Gamma_k} \le \tfrac{\alpha_{k-1}(\tau_{k-1}+\gamma_{k-1})}{\Gamma_{k-1}}\ \forall k \ge 2. \tag{3.68}$$

Then we have

$$\begin{aligned} Q(w_N, w) \le \Gamma_N \textstyle\sum_{k=1}^N \left[\tfrac{(L_f + z^T L_h)\alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \right] + \tfrac{\alpha_N (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} \\ + \Gamma_N \left(\tfrac{\tau_1}{2} + \textstyle\sum_{k=1}^N \tfrac{\alpha_k \gamma_k}{2\Gamma_k} \right) (\|y - q_0\|_2^2 + \|z - r_0\|_2^2),\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+, \end{aligned} \tag{3.69}$$

where D_X is defined in (3.4). As a consequence, we have

$$\begin{aligned} f(x_N) - f(x^*) \le \Gamma_N \textstyle\sum_{k=1}^N \left[\tfrac{L_f \alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \right] + \tfrac{\alpha_N (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} \\ + \Gamma_N \left(\tfrac{\tau_1}{2} + \textstyle\sum_{k=1}^N \tfrac{\alpha_k \gamma_k}{2\Gamma_k} \right) (\|q_0\|_2^2 + \|r_0\|_2^2), \end{aligned} \tag{3.70}$$

and

$$\begin{aligned} \|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \le \Gamma_N \textstyle\sum_{k=1}^N \left[\tfrac{[L_f + (\|z^*\|_2 + 1)\bar{L}_h]\alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k \lambda_k^2 (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \right] \\ + \tfrac{\alpha_N (9\bar{M}_h^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} + \Gamma_N \left(\tfrac{\tau_1}{2} + \textstyle\sum_{k=1}^N \tfrac{\alpha_k \gamma_k}{2\Gamma_k} \right) [(\|y^*\|_2 + 1)^2 + \|q_0\|_2^2 \\ + (\|z^*\|_2 + 1)^2 + \|r_0\|_2^2], \end{aligned} \tag{3.71}$$

where (x^*, y^*, z^*) denotes a triple of optimal solutions for problem (3.6).

Proof. It follows from Lemma 2 and Proposition 7 that

$$\begin{aligned}
\tfrac{Q(w_N,w)}{\Gamma_N} \leq & (1-\alpha_1)Q(w_0,w) + \textstyle\sum_{k=1}^N [\tfrac{(L_f+z^T L_h)\alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \textstyle\sum_{k=1}^N \tfrac{\alpha_k}{\Gamma_k}[\langle A(p_k-p_{k-1}), y-q_k\rangle - \lambda_k\langle A(p_{k-1}-p_{k-2}), y-q_{k-1}\rangle] \\
& + \textstyle\sum_{k=1}^N \tfrac{\alpha_k}{\Gamma_k}[\langle l_h(x_{k-1},p_k) - l_h(x_{k-2},p_{k-1}), z-r_k\rangle \\
& \quad - \lambda_k\langle l_h(x_{k-2},p_{k-1}) - l_h(x_{k-3},p_{k-2}), z-r_{k-1}\rangle] \\
& + \textstyle\sum_{k=1}^N \left[\tfrac{\alpha_k\tau_k}{2\Gamma_k}(\|y-q_{k-1}\|_2^2 + \|z-r_{k-1}\|_2^2) - \tfrac{\alpha_k(\tau_k+\gamma_k)}{2\Gamma_k}(\|y-q_k\|_2^2 + \|z-r_k\|_2^2)\right] \\
& + \textstyle\sum_{k=1}^N \tfrac{\alpha_k\gamma_k}{2\Gamma_k}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2],
\end{aligned}$$

which, in view of (3.68), then implies that

$$\begin{aligned}
Q(w_N,w) \leq & \Gamma_N \textstyle\sum_{k=1}^N [\tfrac{(L_f+z^T L_h)\alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \alpha_N\langle A(p_N-p_{N-1}), y-q_N\rangle - \tfrac{\alpha_N(\tau_N+\gamma_N)}{2}\|y-q_N\|_2^2 \\
& + \alpha_N\langle l_h(x_{N-1},p_N) - l_h(x_{N-2},p_{N-1}), z-r_N\rangle - \tfrac{\alpha_N(\tau_N+\gamma_N)}{2}\|z-r_N\|_2^2 \\
& + \tfrac{\alpha_1\tau_1\Gamma_N}{2}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2] \\
& + \Gamma_N \textstyle\sum_{k=1}^N \tfrac{\alpha_k\gamma_k}{2\Gamma_k}[\|y-q_0\|^2 + \|z-r_0\|^2] \\
\leq & \Gamma_N \textstyle\sum_{k=1}^N [\tfrac{(L_f+z^T L_h)\alpha_k^2 D_X^2}{2\Gamma_k} + \tfrac{\alpha_k\lambda_k^2(9\bar{M}_h^2+\|A\|^2)D_X^2}{2\tau_k\Gamma_k}] \\
& + \tfrac{\alpha_N}{2(\tau_N+\gamma_N)}\|A\|^2\|p_N-p_{N-1}\|_2^2 + \tfrac{9\bar{M}_h^2\alpha_N D_X^2}{2(\tau_N+\gamma_N)} \\
& + \tfrac{\alpha_1\tau_1\Gamma_N}{2}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2] \\
& + \Gamma_N \textstyle\sum_{k=1}^N \tfrac{\alpha_k\gamma_k}{2\Gamma_k}[\|y-q_0\|_2^2 + \|z-r_0\|_2^2],
\end{aligned}$$

where the last relation follows from Young's inequality and a result similar to (3.25). The result in (3.27) then immediately follows from the above inequality. We can show (3.70) and (3.71) similarly to (3.28) and (3.29), and hence the details are skipped. ∎

Corollary 8.1 below shows how to specify the algorithmic parameters, including the regularization weight γ_k, for the CoexDurCG method. In particular, the selection of τ_k was

inspired by the one used in (3.37), and γ_k was chosen so that the last relation in (3.68) is satisfied.

Corollary 8.1 *If the algorithmic parameters α_k, λ_k, τ_k and γ_k of the CoexDurCG method are set to*

$$\alpha_k = \tfrac{2}{k+1}, \lambda_k = \tfrac{k-1}{k},\ \tau_k = \beta\sqrt{k},\ \textit{and}\ \gamma_k = \tfrac{\beta}{k}[(k+1)\sqrt{k+1} - k\sqrt{k}], \tag{3.72}$$

with $\beta = D_X\sqrt{9\bar{M}_h^2 + \|A\|^2}$ for $k \geq 1$, then we have, $\forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$,

$$Q(z_k, z) \leq \tfrac{2(L_f + z^T L_h)D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}\left[3(\|y - q_0\|_2^2 + \|z - r_0\|_2^2) + 1\right]. \tag{3.73}$$

In addition, we have

$$f(x_N) - f(x^*) \leq \tfrac{2L_f D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}\left[3(\|q_0\|_2^2 + \|r_0\|_2^2) + 1\right] \tag{3.74}$$

and

$$\begin{aligned}\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \leq\ & \tfrac{2(L_f + (\|z^*\|_2+1)\bar{L}_h)D_X^2}{N+1} \\ & + \tfrac{D_X\sqrt{9\bar{M}_h^2+\|A\|^2}}{\sqrt{N}}\left[3[(\|y^*\|_2 + 1)^2 + (\|z^*\|_2 + 1)^2 + \|q_0\|_2^2 + \|r_0\|_2^2] + 1\right],\end{aligned} \tag{3.75}$$

where (x^*, y^*, z^*) denotes a triple of optimal solutions for problem (3.6).

Proof. From the definition of α_k in (3.72), we have $\Gamma_k = 2/[k(k+1)]$ and $\alpha_k/\Gamma_k = k$. Hence the first two conditions in (3.68) hold. In addition, it follows from these identities and (3.72) that $\frac{\alpha_k\tau_k}{\Gamma_k} = \beta k\sqrt{k}$ and

$$\tfrac{\alpha_{k-1}(\tau_{k-1}+\gamma_{k-1})}{\Gamma_{k-1}} = (k-1)\left[\beta\sqrt{k-1} + \tfrac{\beta}{k-1}[k\sqrt{k} - (k-1)\sqrt{k-1}]\right] = \beta k\sqrt{k},$$

and hence that the last relation in (3.68) also holds. Observe that by (3.72),

$$\sum_{k=1}^{N} \frac{\alpha_k^2}{\Gamma_k} = 2\sum_{k=1}^{N} \frac{k}{k+1} \le 2N, \tag{3.76}$$

$$\sum_{k=1}^{N} \frac{\alpha_k \gamma_k}{\Gamma_k} = \beta \sum_{k=1}^{N} [(k+1)\sqrt{k+1} - k\sqrt{k}] = \beta[(N+1)\sqrt{N+1} - 1], \tag{3.77}$$

$$\sum_{k=1}^{N} \frac{\alpha_k \lambda_k^2}{\tau_k \Gamma_k} = \sum_{k=1}^{N} \frac{(k-1)^2}{\beta k\sqrt{k}} \le \frac{1}{\beta} \sum_{k=1}^{N} \sqrt{k-1} \le \frac{1}{\beta} \int_0^N \sqrt{t} dt = \frac{2}{3\beta} N^{3/2}. \tag{3.78}$$

Using these relations in (3.69), we have

$$\begin{aligned} Q(w_N, w) &\le \tfrac{2(L_f + z^T L_h) D_X^2}{N+1} + \tfrac{2\sqrt{N}(9\bar{M}_h^2 + \|A\|^2) D_X^2}{3\beta(N+1)} + \tfrac{N(9\bar{M}_h^2 + \|A\|^2) D_X^2}{\beta(N+1)^2\sqrt{N+1}} \\ &\quad + \tfrac{2\beta\sqrt{N+1}}{N}(\|y - q_0\|_2^2 + \|z - r_0\|_2^2) \\ &= \tfrac{2(L_f + z^T L_h) D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2 + \|A\|^2}}{\sqrt{N}} \left[\tfrac{2}{3} + \tfrac{N}{(N+1)^2} + \tfrac{2\sqrt{N+1}}{\sqrt{N}}(\|y - q_0\|_2^2 + \|z - r_0\|_2^2)\right] \\ &\le \tfrac{2(L_f + z^T L_h) D_X^2}{N+1} + \tfrac{D_X\sqrt{9\bar{M}_h^2 + \|A\|^2}}{\sqrt{N}} \left[3(\|y - q_0\|_2^2 + \|z - r_0\|_2^2) + 1\right]. \end{aligned}$$

The bounds in (3.74) and (3.75) can be shown similarly and the details are skipped. ■

In view of the results obtained in Corollary 8.1, the rate of convergence of CoexDurCG matches that of CoexCG. Moreover, the cost of each iteration of the CoexDurCG is the same as that of CoexCG.

3.3.2 Structured Nonsmooth Functions

In this subsection, we consider problem (1.8) with structured nonsmooth functions f and h_i given in (3.41). One possible way to solve this nonsmooth problem is to apply the CoexDurCG method for the smooth approximation problem (3.55). However, this approach still requires us to fix the number of iterations N when choosing smoothing parameters η_i, $i = 0, \ldots, d$.

Our goal in this subsection is to generalize the CoexDurCG method to solve this structured nonsmooth problem directly. Rather than applying this algorithm to problem (3.55), we modify the smoothing parameters η_i, $i = 0, \ldots, d$, at each iteration. More specifically,

we assume that

$$\eta_i^1 \geq \eta_i^2 \geq \ldots \geq \eta_i^k, \ \forall i = 0, \ldots, d, \tag{3.79}$$

and define a sequence of smoothing functions $f_{\eta_0^k}(x)$ and $h_{i,\eta_i^k}(x)$, $i = 1, \ldots, d$, according to (3.45) and (3.49), respectively. For simplicity, we denote

$$f^k(x) \equiv f_{\eta_0^k}(x), \ \ h_i^k(x) \equiv h_{i,\eta_i^k}(x) \text{ and } h^k(x) \equiv (h_1^k(x); \ldots; h_d^k(x)).$$

Also let us define the Lipschitz constants

$$L_f^k \equiv \tfrac{\|B\|^2}{\mu_0+\eta_0^k}, \ L_h^k \equiv (\tfrac{\|C_1\|^2}{\mu_1+\eta_1^k}; \ldots; \tfrac{\|C_d\|^2}{\mu_d+\eta_d^k}), \text{ and } \bar{L}_h^k \equiv \|L_h^k\|_2.$$

It can be seen from (3.79) that

$$f^{k-1}(x) \leq f^k(x) \leq f^{k-1}(x) + (\eta_0^{k-1} - \eta_0^k) D_U^2, \ \forall x \in X. \tag{3.80}$$

Indeed, it suffices to show the second relation in (3.80). By definition, we have

$$\begin{aligned} f^k(x) &= \max_{q \in Q} \{\langle Bx, q\rangle - \hat{f}(q) - \eta_0^k U(q)\} \\ &= \max_{q \in Q} \{\langle Bx, q\rangle - \hat{f}(q) - \eta_0^{k-1} U(q) + (\eta_0^{k-1} - \eta_0^k) U(q)\} \\ &\leq \max_{q \in Q} \{\langle Bx, q\rangle - \hat{f}(q) - \eta_0^{k-1} U(q) + (\eta_0^{k-1} - \eta_0^k) D_U^2\} \\ &= f^{k-1}(x) + (\eta_0^{k-1} - \eta_0^k) D_U^2, \end{aligned}$$

where the inequality follows from the definition of D_U in (3.44) and the assumption $\eta_0^{k-1} \geq \eta_0^k$ in (3.79). Similarly, we have

$$h_i^{k-1}(x) \leq h_i^k(x) \leq h_i^{k-1}(x) + (\eta_i^{k-1} - \eta_i^k) D_{V_i}^2, \ \forall x X, \ i = 1, \ldots, d. \tag{3.81}$$

Note that in our algorithmic scheme, we can set $\eta_i^k = 0$, $i = 0, 1, \ldots, d$, if the correspond-

ing objective or constraint functions are smooth (i.e., $\mu_i = 0$).

We now describe the more general CoexDurCG method for solving structured nonsmooth problems.

Algorithm 5 CoexDurCG for Structured Nonsmooth Problems

The algorithm is the same as Algorithm 4 except that the extrapolation step (3.10) is replaced by

$$\tilde{h}_k = l_{h^{k-1}}(x_{k-2}, p_{k-1}) + \lambda_k[l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2})], \tag{3.82}$$

and the linear optimization step is replaced by

$$p_k = \operatorname{argmin}_{x \in X}\{l_{f^k}(x_{k-1}, x) + \langle g(x), q_k\rangle + \langle l_{h^k}(x_{k-1}, x), r_k\rangle\}. \tag{3.83}$$

In Algorithm 5 we do not explicitly use the smooth approximation problem (3.55). Instead, we incorporate in (3.82) and (3.83) the adaptive linear approximation functions l_{h^k} and l_{f^k} for the objective and constraints, respectively. The convergence analysis of this algorithm relies on the adaptive primal-dual gap function:

$$Q^k(\bar{w}, w) \equiv Q_{\eta^k}(\bar{w}, w) := f^k(\bar{x}) - f^k(x) + \langle g(\bar{x}), y\rangle - \langle g(x), \bar{y}\rangle + \langle h^k(\bar{x}), z\rangle - \langle h^k(x), \bar{z}\rangle, \tag{3.84}$$

as demonstrated in the following result.

Proposition 9 *For any $k > 1$, we have*

$$
\begin{aligned}
Q^k(w_k, w) \le\ & (1-\alpha_k) Q^{k-1}(w_{k-1}, w) + \tfrac{(L_f^k + z^T L_h^k)\alpha_k^2 D_X^2}{2} \\
& + (1-\alpha_k)[(\eta_0^{k-1} - \eta_0^k) D_U^2 + \textstyle\sum_{i=1}^d (\eta_i^{k-1} - \eta_i^k) z_i D_{V_i}^2] \\
& + \tfrac{\alpha_k \lambda_k^2 (12 \bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k} + \tfrac{3\lambda_k^2}{\tau_k} \textstyle\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 \\
& + \alpha_k[\langle A(p_k - p_{k-1}), y - q_k\rangle - \lambda_k \langle A(p_{k-1} - p_{k-2}), y - q_{k-1}\rangle] \\
& + \alpha_k[\langle l_{h^k}(x_{k-1}, p_k) - l_{h^{k-1}}(x_{k-2}, p_{k-1}), z - r_k\rangle \\
& \quad - \lambda_k \langle l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2}), z - r_{k-1}\rangle] \\
& + \tfrac{\alpha_k \tau_k}{2}(\|y - q_{k-1}\|_2^2 + \|z - r_{k-1}\|_2^2) - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2}(\|y - q_k\|_2^2 \\
& + \|z - r_k\|_2^2) + \tfrac{\alpha_k \gamma_k}{2}[\|y - q_0\|_2^2 + \|z - r_0\|_2^2],\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}_+^d,
\end{aligned}
$$

where D_X is defined in (3.4).

Proof. Similar to (3.67), we can show that

$$
\begin{aligned}
Q^k(w_k, w) \le\ & (1-\alpha_k) Q^k(w_{k-1}, w) + \tfrac{(L_f^k + z^T L_h^k)\alpha_k^2 D_X^2}{2} \\
& + \alpha_k \langle g(p_k) - \tilde{g}_k), y - q_k\rangle + \alpha_k \langle l_{h^k}(x_{k-1}, p_k) - \tilde{h}_k, z - r_k\rangle \\
& + \tfrac{\alpha_k \tau_k}{2}[\|y - q_{k-1}\|_2^2 - \|q_k - q_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2}\|y - q_k\|_2^2 \\
& + \tfrac{\alpha_k \tau_k}{2}[\|z - r_{k-1}\|_2^2 - \|r_k - r_{k-1}\|_2^2] - \tfrac{\alpha_k(\tau_k + \gamma_k)}{2}\|z - r_k\|_2^2 \\
& + \tfrac{\alpha_k \gamma_k}{2}[\|y - q_0\|_2^2 + \|z - r_0\|_2^2],\ \forall w \in X \times \mathbb{R}^m \times \mathbb{R}_+^d. \qquad (3.85)
\end{aligned}
$$

Moreover, by the definition of $\tilde{h}_k$ in (3.82), we have

$$
\begin{aligned}
&\langle l_{h^k}(x_{k-1}, p_k) - \tilde{h}_k, z - r_k\rangle - \tfrac{\tau_k}{2}\|r_k - r_{k-1}\|_2^2 \\
&= \langle l_{h^k}(x_{k-1}, p_k) - l_{h^{k-1}}(x_{k-2}, p_{k-1}), z - r_k\rangle \qquad (3.86)\\
&- \lambda_k\langle l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2}), z - r_{k-1}\rangle \\
&\quad + \lambda_k\langle l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2}), r_k - r_{k-1}\rangle - \tfrac{\tau_k}{2}\|r_k - r_{k-1}\|_2^2 \\
&\le \langle l_{h^k}(x_{k-1}, p_k) - l_{h^{k-1}}(x_{k-2}, p_{k-1}), z - r_k\rangle \\
&\quad - \lambda_k\langle l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2}), z - r_{k-1}\rangle \\
&\quad + \tfrac{6\lambda_k^2 D_X^2 \bar{M}_{C,V}^2}{\tau_k} + \tfrac{3\lambda_k^2}{\tau_k}\textstyle\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4, \qquad (3.87)
\end{aligned}
$$

where the last inequality follows from

$$
\begin{aligned}
&\lambda_k\langle l_{h^{k-1}}(x_{k-2}, p_{k-1}) - l_{h^{k-2}}(x_{k-3}, p_{k-2}), r_k - r_{k-1}\rangle - \tfrac{\tau_k}{2}\|r_k - r_{k-1}\|_2^2 \\
&\le \tfrac{\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d [l_{h_i^{k-1}}(x_{k-2}, p_{k-1}) - l_{h_i^{k-2}}(x_{k-3}, p_{k-2})]^2 \\
&= \tfrac{\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d [h_i^{k-1}(x_{k-2}) - h_i^{k-2}(x_{k-3}) \qquad (3.88)\\
&\quad + \langle \nabla h_i^{k-1}(x_{k-2}), p_{k-1} - x_{k-2}\rangle + \langle \nabla h_i^{k-2}(x_{k-3}), p_{k-2} - x_{k-3}\rangle]^2 \\
&\le \tfrac{3\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d \left[(h_i^{k-1}(x_{k-2}) - h_i^{k-2}(x_{k-3}))^2 + 2M_{C_i,V_i}^2 D_X^2\rangle\right] \\
&\le \tfrac{3\lambda_k^2}{2\tau_k}\textstyle\sum_{i=1}^d \left[2(h_i^{k-2}(x_{k-2}) - h_i^{k-2}(x_{k-3}))^2 + 2(\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 + 2M_{C_i,V_i}^2 D_X^2\rangle\right] \\
&\le \tfrac{6\lambda_k^2 D_X^2}{\tau_k}\textstyle\sum_{i=1}^d M_{C_i,V_i}^2 + \tfrac{3\lambda_k^2}{\tau_k}\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 \\
&= \tfrac{6\lambda_k^2 D_X^2 \bar{M}_{C,V}^2}{\tau_k} + \tfrac{3\lambda_k^2}{\tau_k}\textstyle\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4. \qquad (3.89)
\end{aligned}
$$

Here, the first inequality follows from Young's inequality, the second inequality follows from the cauchy-schwarz inequality, the definition of D_X in (3.4) and the bound of ∇h_i^k in (3.53), the third inequality follows by the relation between h_i^{k-1} and h_i^{k-2} in (3.81) and the simple fact that $(a+b)^2 \le 2a^2 + 2b^2$, and the last inequality follows from the Lipschitz continuity of h_i^{k-2} and the bound in (3.53). In addition, it follows from (3.80) and (3.81)

that for any $w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$,

$$Q^k(w_{k-1}, w) \leq Q^{k-1}(w_{k-1}, w) + (\eta_0^{k-1} - \eta_0^k) D_U^2 + \textstyle\sum_{i=1}^d (\eta_i^{k-1} - \eta_i^k) z_i D_{V_i}^2. \quad (3.90)$$

The result follows by combining (3.85), (3.87), (3.90) and the bound in (3.23). ∎

Theorem 10 *Let Γ_k be defined in* (3.20) *and assume that the algorithmic parameters α_k, τ_k and λ_k in the CoexDurCG method in Algorithm 5 satisfy (3.68). Then we have, $\forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$,*

$$\begin{aligned} Q(w_N, w) \leq{}& \Gamma_N \textstyle\sum_{k=1}^N \left[\frac{(L_f^k + z^T L_h^k)\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} + \frac{3\lambda_k^2}{\tau_k \Gamma_k} \sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 \right] \\ &+ \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k} (\eta_0^k D_U^2 + \sum_{i=1}^d \eta_i^k z_i D_{V_i}^2) + \frac{\alpha_N (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} + \frac{6\alpha_N \sum_{i=1}^d (\eta_i^{N-1} - \eta_i^N)^2 D_{V_i}^4}{2(\tau_N + \gamma_N)} \\ &+ \Gamma_N \left(\tfrac{\tau_1}{2} + \textstyle\sum_{k=1}^N \frac{\alpha_k \gamma_k}{2\Gamma_k} \right) (\|y - q_0\|_2^2 + \|z - r_0\|_2^2) + \eta_0^N D_U^2 + \|z\| (\textstyle\sum_{i=1}^d (\eta_i^N D_{V_i}^2)^2)^{1/2}, \end{aligned} \quad (3.91)$$

where D_X is defined in (3.4). As a consequence, we have

$$\begin{aligned} f(x_N) - f(x^*) \leq{}& \Gamma_N \textstyle\sum_{k=1}^N \left[\frac{L_f^k \alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} + \frac{3\lambda_k^2}{\tau_k \Gamma_k} \sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 \right] \\ &+ \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k} \eta_0^k D_U^2 + \frac{\alpha_N (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} + \frac{6\alpha_N \sum_{i=1}^d (\eta_i^{N-1} - \eta_i^N)^2 D_{V_i}^4}{2(\tau_N + \gamma_N)} \\ &+ \Gamma_N \left(\tfrac{\tau_1}{2} + \textstyle\sum_{k=1}^N \frac{\alpha_k \gamma_k}{2\Gamma_k} \right) (\|q_0\|_2^2 + \|r_0\|_2^2) + \eta_0^N D_U^2 \end{aligned} \quad (3.92)$$

and

$$\begin{aligned} &\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \\ &\leq \Gamma_N \textstyle\sum_{k=1}^N \left[\frac{[L_f^k + (\|z^*\|_2 + 1)\bar{L}_h^k]\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} + \frac{3\lambda_k^2}{\tau_k \Gamma_k} \sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4 \right] \\ &\quad + \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k} (\eta_0^k D_U^2 + \sum_{i=1}^d \eta_i^k \hat{z}_i D_{V_i}^2) + \frac{\alpha_N (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2(\tau_N + \gamma_N)} + \frac{6\alpha_N \sum_{i=1}^d (\eta_i^{N-1} - \eta_i^N)^2 D_{V_i}^4}{2(\tau_N + \gamma_N)} \\ &\quad + \Gamma_N \left(\tau_1 + \textstyle\sum_{k=1}^N \frac{\alpha_k \gamma_k}{2\Gamma_k} \right) [(\|y^*\|_2 + 1)^2 + \|q_0\|_2^2 + (\|z^*\|_2 + 1)^2 + \|r_0\|_2^2] \\ &\quad + \eta_0^N D_U^2 + (\|z^*\|_2 + 1)(\textstyle\sum_{i=1}^d (\eta_i^N D_{V_i}^2)^2)^{1/2}, \end{aligned} \quad (3.93)$$

where (x^*, y^*, z^*) *denotes a triple of optimal solutions for problem (3.6).*

Proof. It follows from Lemma 2, Proposition 9 and (3.68) that

$$\begin{aligned}
Q^N(w_N, w) \le\ & \Gamma_N \textstyle\sum_{k=1}^N [\frac{(L_f^k + z^T L_h^k)\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \\
& + \tfrac{3\lambda_k^2}{\tau_k \Gamma_k} \textstyle\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4] + \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k} (\eta_0^k D_U^2 + \sum_{i=1}^d \eta_i^k z_i D_{V_i}^2) \\
& + \alpha_N \langle A(p_N - p_{N-1}), y - q_N \rangle - \tfrac{\alpha_N(\tau_N + \gamma_N)}{2} \|y - q_N\|_2^2 \\
& + \alpha_N \langle l_{h^N}(x_{N-1}, p_N) - l_{h^{N-1}}(x_{N-2}, p_{N-1}), z - r_N \rangle - \tfrac{\alpha_N(\tau_N + \gamma_N)}{2} \|z - r_N\|_2^2 \\
& + \tfrac{\alpha_1 \tau_1 \Gamma_N}{2} [\|y - q_0\|_2^2 + \|z - r_0\|_2^2] + \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k \gamma_k}{2\Gamma_k} [\|y - q_0\|^2 + \|z - r_0\|^2] \\
\le\ & \Gamma_N \textstyle\sum_{k=1}^N [\frac{(L_f^k + z^T L_h^k)\alpha_k^2 D_X^2}{2\Gamma_k} + \frac{\alpha_k \lambda_k^2 (12\bar{M}_{C,V}^2 + \|A\|^2) D_X^2}{2\tau_k \Gamma_k} \\
& + \tfrac{3\lambda_k^2}{\tau_k \Gamma_k} \textstyle\sum_{i=1}^d (\eta_i^{k-2} - \eta_i^{k-1})^2 D_{V_i}^4] + \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k}{\Gamma_k} (\eta_0^k D_U^2 + \sum_{i=1}^d \eta_i^k z_i D_{V_i}^2) \\
& + \tfrac{\alpha_N}{2(\tau_N + \gamma_N)} \|A\|^2 \|p_N - p_{N-1}\|_2^2 + \tfrac{12\bar{M}_{C,V}^2 \alpha_N D_X^2}{2(\tau_N + \gamma_N)} + \tfrac{6\alpha_N \sum_{i=1}^d (\eta_i^{N-1} - \eta_i^N)^2 D_{V_i}^4}{2(\tau_N + \gamma_N)} \\
& + \tfrac{\alpha_1 \tau_1 \Gamma_N}{2} [\|y - q_0\|_2^2 + \|z - r_0\|_2^2] \\
& + \Gamma_N \textstyle\sum_{k=1}^N \frac{\alpha_k \gamma_k}{2\Gamma_k} [\|y - q_0\|_2^2 + \|z - r_0\|_2^2],
\end{aligned}$$

where the last relation follows from Young's inequality and a result similar to (3.89). The result in (3.91) then immediately follows from the above inequality and the observation that $Q(w^N, w) \le Q^N(w^N, w) + \eta_0^N D_U^2 + \|z\|(\sum_{i=1}^d (\eta_i^N D_{V_i}^2)^2)^{1/2}$ due to (3.59). We can show (3.92) and (3.93) similarly to (3.56) and (3.57), and hence the details are skipped. ∎

Corollary 10.1 below shows how to specify the smoothing parameter $\{\eta_i^k\}$ in (3.79) and other parameters for the CoexDurCG method in Algorithm 5. We focus on the most challenging case when the objective function f and all the constraint functions are nonsmooth (i.e., $\mu_i = 0$, $i = 1, \ldots, n$). Slightly improved rate of convergence can be obtained by setting $\eta_i^k = 0$ for those component functions with $\mu_i > 0$.

Corollary 10.1 *Suppose that the parameters* α_k, λ_k, τ_k *and* γ_k *in Algorithm 5 are set to (3.72) with* $\beta = D_X \sqrt{12\bar{M}_{C,V}^2 + \|A\|^2}$ *for* $k \ge 1$. *If the smoothing parameters* η_i^k *are set*

to

$$\eta_0^k = \frac{\|B\|D_X}{\sqrt{k}D_U},\ \eta_i^k = \frac{\|C_i\|D_X}{\sqrt{k}D_{V_i}},\ \forall i = 1, \ldots, d, \tag{3.94}$$

then we have, $\forall w \in X \times \mathbb{R}^m \times \mathbb{R}^d_+$,

$$Q(w_k, w) \leq \frac{8(\|B\|D_U + \sum_{i=1}^d z_i\|C_i\|D_{V_i})D_X}{3\sqrt{N}} + \frac{\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}D_X}{\sqrt{N}}[2(\|y - q_0\|^2 + \|z - r_0\|^2) + 2] + \frac{12\sum_{i=1}^d \|C_i\|^2 D_X D^2_{V_i}}{\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}(N+1)\sqrt{N}} + \frac{D_X}{\sqrt{N}}(\|B\|D_U + \|z\|\sqrt{\sum_{i=1}^d \|C_i\|^2 D^2_{V_i}}). \tag{3.95}$$

In addition, we have

$$f(x_N) - f(x^*) \leq \frac{11\|B\|D_U D_X}{3\sqrt{N}} + \frac{\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}D_X}{\sqrt{N}}[2(\|q_0\|^2 + \|r_0\|^2) + 2] + \frac{12\sum_{i=1}^d \|C_i\|^2 D_X D^2_{V_i}}{\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}(N+1)\sqrt{N}} \tag{3.96}$$

and

$$\|g(x_N)\|_2 + \|[h(x_N)]_+\|_2 \leq \frac{7(\|B\|D_U + (\|z^*\|+1)\sqrt{\sum_{i=1}^d \|C_i\|^2 D^2_{V_i}})D_X}{3\sqrt{N}} + \frac{12\sum_{i=1}^d \|C_i\|^2 D_X D^2_{V_i}}{\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}(N+1)\sqrt{N}} + \frac{2\sqrt{12\bar{M}^2_{C,V}+\|A\|^2}D_X}{\sqrt{N}}[4[(\|y^*\|_2 + 1)^2 + (\|z^*\|_2 + 1)^2 + \|q_0\|_2^2 + \|r_0\|_2^2] + 2], \tag{3.97}$$

where (x^*, y^*, z^*) *denotes a triple of optimal solutions for problem (3.6).*

Proof. From the definition of α_k in (3.72), we have $\Gamma_k = 2/[k(k+1)]$ and $\alpha_k/\Gamma_k = k$. Similarly to Corollary 8.1, we can check that condition (3.68), and the bounds in (3.76)-(3.78) hold. In addition, it follows from the definition of η_i^k in (3.94) that

$$(\eta_i^{k-2} - \eta_i^{k-1})^2 = \frac{\|C_i\|^2 D_X^2}{D^2_{V_i}}(\tfrac{1}{k-1} + \tfrac{1}{k-2} - \tfrac{2}{\sqrt{k-1}\sqrt{k-2}}) \leq \frac{\|C_i\|^2 D_X^2}{(k-1)(k-2)D^2_{V_i}},$$

$$L_f^k = \frac{\|B\|D_U\sqrt{k}}{D_X},\ \text{and}\ L_{h,i}^k = \frac{\|C_i\|D_{V_i}\sqrt{k}}{D_X},\ \forall i = 1, \ldots, d.$$

Using these relations in (3.91), we have

$$
\begin{aligned}
Q(w_N, w) \le & \frac{2(\|B\|D_U + \sum_{i=1}^d z_i \|C_i\| D_{V_i}) D_X}{N(N+1)} \sum_{k=1}^N \frac{k\sqrt{k}}{k+1} + \frac{2\sqrt{12\bar{M}_{C,V}^2 + \|A\|^2} D_X}{3\sqrt{N}} \\
& + \frac{3}{\beta N(N+1)} \sum_{k=1}^N (\sqrt{k}(k+1) \sum_{i=1}^d \frac{\|C_i\|^2 D_X^2 D_{V_i}^2}{(k-1)(k-2)}) \\
& + \frac{2(\|B\|D_U + \sum_{i=1}^d z_i \|C_i\| D_{V_i}) D_X}{N(N+1)} \sum_{k=1}^N \sqrt{k} + \frac{\sqrt{12\bar{M}_{C,V}^2 + \|A\|^2} D_X}{(N+1)\sqrt{N+1}} \\
& + \frac{3\sum_{i=1}^d \|C_i\|^2 D_X^2 D_{V_i}^2}{(N+1)\sqrt{N+1}(N-1)(N-2)} + \frac{\beta}{N(N+1)} [(N+1)\sqrt{N+1}][\|y - q_0\|^2 + \|z - r_0\|^2] \\
& + \eta_0^N D_U^2 + \|z\| (\sum_{i=1}^d (\eta_i^N D_{V_i}^2)^2)^{1/2},
\end{aligned}
$$

which implies (3.95) after simplification. (3.96) and (3.97) can be shown similarly and the details are skipped. ∎

Comparing the results in Corollary 10.1 with those in Corollary 5.1, we can see that the rate of convergence of CoexDurCG is about the same as that of CoexCG for nonsmooth optimization. However, it is more convenient to implement CoexDurCG since it does not require us to fix the number of iterations a priori.

3.4 Numerical Experiments

In this section, we apply the proposed algorithms to the intensity modulated radiation therapy (IMRT) problem briefly discussed in Section 1.

3.4.1 Problem Formulation

In IMRT, the patient will be irradiated by a linear accelerator (linac) from several angles and in each angle the device uses different apertures. In traditional IMRT, we select and fix 5-9 angles and then design and optimize the apertures and their corresponding intensity. Following [47], we would like to integrate the angle selection into direct aperture optimization in order to use a small number of angles and apertures in the final treatment plan.

To model the IMRT treatment planning, we discretize each structure s of the patient into small cubic volume elements called *voxels*, $\mathcal{V}$. There are a finite number of angles, denoted by A, around the patient. A beam in each angle, b_a, is decomposed into a rectangular grid of *beamlets*. A beamlet (i, j) is effective if it is not blocked by either the left, l_i, and right, r_i, leaves. An aperture is then defined as the collection of effective beamlets. The relative motion of the leaves controls the set of effective beamlets and thus the shape of the aperture. The estimated dose received by voxel v from beamlet (i, j) at unit intensity is denoted by $D_{(i,j)v}$ in Gy. The dose absorbed by a given voxel is the summation of the dose from each individual beamlet.

Let P_a be the set of allowed apertures determined by the position of the left and right leaves in beam angle a. Suppose that the rectangular grid in each angle has m rows and n columns, and the leaves move along each row independently. Then the number of possible apertures in each angle amounts to $(\frac{n(n-1)}{2})^m$. We use $\mathbf{x}^{a,t}$, comprised of binary decision variables $x^{a,t}_{(i,j)}$, to describe the shape of aperture $t \in P_a$. In particular, $x^{a,t}_{(i,j)} = 1$ if beamlet (i, j) is effective, i.e., falling within the left and right leaves of row i, otherwise $x^{a,t}_{(i,j)} = 0$. In addition to selecting angles and apertures, we also need to determine the influence rate $y^{a,t}$ for aperture $t \in P_a$, which will be used to determine the dose intensity and the amount of radiation time from aperture t. The dose absorbed by voxel v is computed by $z_v = \sum_{a\in\mathcal{A}} \sum_{t\in P_a} \sum_{i=1}^{m} \sum_{j=1}^{n} RD_{(i,j)v}\, x^{a,t}_{ij} y^{a,t}$, based on the dose-influence matrix D, the aperture shape $\mathbf{x}^k$, and the aperture influence rate y^k. We measure the treatment quality by $f(\mathbf{z}) := \sum_{v\in\mathcal{V}} \underline{w}_v \,[\underline{T}_v - z_v]^2_+ + \overline{w}_v \,[z_v - \overline{T}_v]^2_+$ via voxel-based quadratic penalty, where $[\cdot]_+$ denotes $\max\{0, \cdot\}$, and $\underline{T}_v$ and $\overline{T}_v$ are pre-specified lower and upper dose thresholds for voxel v.

We also need to consider a few important function constraints. Firstly, in order to obtain a sparse solution with a small number of angles, we add the following group sparsity constraint $\sum_{a\in\mathcal{A}} \max_{t\in P_a} y^{a,t} \le \Phi$ for some properly chosen $\Phi > 0$. Intuitively, this constraint will encourage the selection of apertures in those angles P_a that have already

contained some nonzero elements of $y^{a,t}$, $t \in P_a$. Secondly, we need to meet a few critical clinical criteria to avoid underdose (resp., overdose) for tumor (resp., healthy) structures. These criteria are usually specified as value at risk (VaR) constraints. For example, in the prostate benchmark dataset, the clinical criterion of "PTV56:V56$\geq 95\%$" means that the percentage of voxels in structure PTV56 that receive at least 56 Gy dose should be at least 95%. Similarly, the criterion of "PTV68: V74.8$\leq 10\%$" implies that the percentage of voxels in structure PTV68 that receive more than 74.8 Gy dose should be at most 10%. One possible way to satisfy these criterions is to tune the weights $((\underline{w}_v, \overline{w}_v))$ in $f(\mathbf{z})$. However, it would be time consuming to tune these weights to satisfy all the prescribed clinical criteria. Therefore, we suggest to incorporate a few critical criteria as problem constraints explicitly.

Instead of using VaR, we will use its convex approximation, commonly referred to as Conditional Value at Risk (CVaR) in the constraints [1]. Recall the following definitions of VaR and CVaR

$$\text{Upper tail: } \mathrm{VaR}_\alpha(X) = \inf_\tau\{\tau : P(X \leq \tau) \geq \alpha\}, \mathrm{CVaR}_\alpha(X) = \inf_\tau \tau + \tfrac{1}{1-\alpha}\mathbb{E}[X-\tau]_+.$$

$$\text{Lower tail: } \mathrm{VaR}_\alpha(X) = \sup_\tau\{\tau : P(X \geq \tau) \geq \alpha\}, \mathrm{CVaR}_\alpha(X) = \sup_\tau \tau - \tfrac{1}{1-\alpha}\mathbb{E}[\tau-X]_+.$$

The upper (resp., lower) tail CVaR will be used to enforce the underdose (resp., overdose) clinical criteria. For example, letting S_1 and S_2 denote structures PTV68 and PTV 56, and N_1 and N_2 be the number of voxels in these structures, we can approximately formulate the criterion of "PTV68: V74.8$\leq 10\%$" as $\inf_\tau \tau_1 + \frac{1}{(1-0.9)N_1}\sum_{v\in S_1}[z_v - \tau_1]_+ \leq b$ for some $b \geq 74.8$. Separately, the criterion of "PTV56:V56$\geq 95\%$" will be approximated by $\sup_\tau \tau - \frac{1}{(1-0.95)N_2}\sum_{v\in S_2}[\tau - z_v]_+ \geq b$, or equivalently $\inf_\tau -\tau + \frac{1}{(1-0.95)N_2}\sum_{v\in S_2}[\tau - z_v]_+ \leq -b$ for some $b \leq 56$. Putting the above discussions together and denoting $\hat{D}_v^{a,t} :=$

$\sum_{i=1}^{m}\sum_{j=1}^{n} D_{(i,j)v}\, x_{ij}^{a,t}$, we obtain the following problem formulation.

$$\begin{aligned}
\min \quad & f(\mathbf{z}) := \tfrac{1}{N_v}\textstyle\sum_{v\in\mathcal{V}} \underline{w}_v\,[\underline{T}_v - z_v]_+^2 + \overline{w}_v\,[z_v - \overline{T}_v]_+^2 && \text{(3.98a)}\\
\text{s.t.} \quad & z_v = \sum_{a\in\mathcal{A}}\sum_{t\in P_a} R\hat{D}_v^{a,t} y^{a,t}, && \text{(3.98b)}\\
& -\tau_i + \tfrac{1}{p_i N_i}\textstyle\sum_{v\in S_i}[\tau_i - z_v]_+ \le -b_i, \forall i \in \mathrm{UD}, && \text{(3.98c)}\\
& \tau_i + \tfrac{1}{p_i N_i}\textstyle\sum_{v\in S_i}[z_v - \tau_i]_+ \le b_i, \forall i \in \mathrm{OD}, && \text{(3.98d)}\\
& \textstyle\sum_{a\in\mathcal{A}} \max_{t\in P_a} y^{a,t} \le \Phi, && \text{(3.98e)}\\
& \textstyle\sum_{a\in\mathcal{A}}\sum_{t\in P_a} y^{a,t} \le 1, && \text{(3.98f)}\\
& y^{a,t} \ge 0, && \text{(3.98g)}\\
& \tau_i \in [\underline{\tau}_i, \bar{\tau}_i], \forall i \in \mathrm{UD}\ \&\ \mathrm{OD}, && \text{(3.98h)}
\end{aligned}$$

where OD and UD denote the set of overdose and underdose clinical criteria, respectively. Clearly, the objective function f is convex and smooth. Constraints in (3.98c), (3.98d) and (3.98e) are structured nonsmooth function constraints, while (3.98f)-(3.98g) define the simplex constraint. The bounds $\underline{\tau}$ and $\bar{\tau}$ in constraints (3.98h) can be obtained from the corresponding clinical criteria. For example, the criterion of "PTV68:V68$\ge 95\%$" implies that value at risk ≥ 68. By the definition of CVaR, the optimal τ equals to the value at risk, hence we set $\underline{\tau} = 68$. In a similar way, we set $\bar{\tau} = 74.8$ in view of the criterion of "PTV68: V74.8$\le 10\%$".

We can apply the CoexCG and CoexDurCG methods described in Subsections 3.2.2 and 3.3.2, respectively, to solve problem (3.98a)-(3.98h). Since the number of the potential apertures (i.e., the dimension of $y^{a,t}$) increases exponentially w.r.t. m, we cannot compute the full gradient of the objective and constraint functions w.r.t. $y^{a,t}$. Instead, we will perform gradient computation and linear optimization simultaneously. Let us focus on the CoexCG method for illustration. Denote the constraints (3.98c)-(3.98e) as h_i, $i \in OD \cup UD$, and let the corresponding smooth approximation h_{i,η_i} be defined by (3.49) (using entropy

distances for smoothing). For a given search point $x_{k-1} := (\{y_{k-1}^{a,t}\}, \{\tau_{i,k-1}\})$ and dual variable $\{r_{i,k-1}\}$, let us denote $\pi_{k-1}^{f} = \partial f(\mathbf{x}_{k-1})/\partial \mathbf{z}$ and $\pi_{k-1}^{h_i} = \partial h_{i,\eta_i}(\mathbf{x}_{k-1})/\partial \mathbf{z}$. Clearly, in view of (3.13), $y_{k-1}^{a,t}$ will be updated to a properly chosen extreme point of the simplex constraint in (3.98f)-(3.98g). In order to determine this extreme point, we need to find the aperture with the smallest coefficient in the linear objective of (3.13) given by:

$$\begin{aligned}\psi^{a,t} :=& \pi_{k-1}^{f} \tfrac{\partial z}{\partial y^{a,t}} + \textstyle\sum_i r_{i,k-1} \pi_{k-1}^{h_i} \tfrac{\partial z}{\partial y^{a,t}} \\ =& R \textstyle\sum_{i=1}^{m} \sum_{j=1}^{n} (\sum_v D_{(i,j)v} (\pi_{v,k-1}^{f} + \sum_i r_{i,k-1} \pi_{v,k-1}^{h_i})) x_{ij},\ x_{ij} \in \{0,1\}.\end{aligned}$$

This can be achieved by using the following constructive approach. For any row i of the rectangular grid in angle a, we find the column indices c_1 and c_2, respectively, for the left and right leaves, that give the most negative value of $\sum_{c_1<j<c_2} \sum_v D_{(i,j)v} (\pi_{v,k-1}^{f} + \sum_i r_{i,k} \pi_{v,k-1}^{h_i})$. Repeating this process row by row, we construct the aperture with the smallest value of $\psi^{a,t}$ in angle a. We construct one aperture similar to this for each angle, and then choose the one with the most negative value of $\psi^{a,t}$ among all the angles.

3.4.2 Comparison of CoexCG and CoexDurCG on randomly generated instances

Due to the privacy issue, publicly available IMRT datasets for real patients are very limited. To test the performance of our proposed algorithms we first randomly generate some problem instances as follows. Let $V = [-l, l]^3 \subseteq \mathbb{R}^3$ be a cube with length l. Viewing V as the human body, we then arbitrarily choose two (or more) cuboids as healthy organs, and randomly choose 2 cubes inside V as the target tumor tissues. For a given accuracy $\delta > 0$, we discretize all these structures into small cubes with length δ to define a voxel. Around the cube V, we generate a circle with radius $2l$ on the plane $\{x = 0\}$, and define every two degrees as one angle for radiation therapy. In each angle, we consider the aperture as a square in $[-l, l]^2$, and also discretize it with small squares with length δ, resulting in a grid with size $\frac{2l}{\delta} \times \frac{2l}{\delta}$. After that, we randomly generate N_a beamlets with coordinate

Table 3.1: Data Instances with $\Phi = 0.2$

Index	# of voxels	# of apertures	b_i & p_i
Ins. 1	4096	46080	[30,40,200] & [0.05,0.05,0.05]
Ins. 2	4096	46080	[40,50,100] & [0.01,0.01,0.05]
Ins. 3	4096	46080	[50,60,80] & [0.01,0.01,0.01]
Ins. 4	262144	737280	[40,50,100] & [0.01,0.01,0.05]
Ins. 5	262144	737280	[50,60,80] & [0.01,0.01,0.01]

$(x', y') \in [-l, l]^2$ for each angle a. As for the matrix D (recording the dose received by voxel v from each beamlet), we first check if the voxel is radiated by the beamlet since each beamlet is a line perpendicular to the aperture plane. If so, the dose received by the voxel from this beamlet will be set to $2/d$, where d is the distance between the voxel and the aperture plane; otherwise, the dose is 0. By choosing different accuracy δ, we can create instances with different sizes in terms of the number of voxels and potential apertures. Table 3.1 shows five different test instances generated with $l = 8$. We set $\delta = 1$ and 0.25 for the first three instances (Ins. 1, Ins. 2 and Ins. 3), and the last two instances (Ins. 4 and Ins. 5), respectively. Note that we consider 2 underdose and 1 overdose constraints and their corresponding r.h.s. b and p are shown in the last column of Table 3.1. We set the $\underline{T}_v = \bar{T}_v = 56$ for tumor tissue and $\underline{T}_v = \bar{T}_v = 0$ for healthy organ in (3.98a). In addition, we set $\Phi = 0.2$ for the group sparsity constraint in (3.98e).

We implement in Matlab the CoexCG and CoexDurCG algorithms for structured nonsmooth problems, and report the computational results in Table 3.2. Here we use $x_N := (y_N, \tau_N)$, $f(x_N)$ and $\|h(x_N)\|$, respectively, to denote the output solution, the objective value and constraint violations. The CPU times are in seconds on a Macbook Pro with 2.6 GHz 6-Core Intel Core i7 processor. As shown in Table 3.2, both CoexCG and CoexDurCG exhibit comparable performance in terms of objective value, constraint violation and CPU time for different iteration limit N. However, unlike the CoexDurCG algorithm, we need to rerun CoexCG for all the experiments whenever N changes.

Table 3.2: Results for different Instances

Index	N	CoexCG			CoexDurCG		
		$f(x_N)$	$\|h(x_N)\|$	CPU(s)	$f(x_N)$	$\|h(x_N)\|$	CPU(s)
Ins. 1	1	46.8723	1.7237e+03				
	100	0.0683	0.4234	34	0.0616	0.3705	33
	1000	0.0197	0.0319	323	0.0210	0.0219	327
Ins. 2	1	46.8723	1.7237e+03				
	100	0.0568	0.4424	33	0.0583	0.5002	34
	1000	0.0224	0.0426	327	0.0232	0.0334	339
Ins. 3	1	46.8723	1.7237e+03				
	100	0.0625	13.7567	33	0.0604	7.3929	33
	1000	0.0227	0.0514	332	0.0226	0.0193	332
Ins. 4	1	47.7099	8.7850e+03				
	100	0.4643	163.3043	1645	0.4643	163.3043	1645
	1000	0.0398	12.1765	17254	0.0398	12.1765	17356
Ins. 5	1	47.7099	8.7850e+03				
	100	0.4866	253.9389	1644	0.4581	206.9143	1637
	1000	0.0406	39.2051	17146	0.0417	38.6486	17607

3.4.3 Results for real dataset

In this subsection, we apply CoexDurCG to the real dataset for a patient with prostate cancer (`https://github.com/cerr/CERR/wiki`), and evaluate the generated solution from the clinical point of view. Dose volume histogram (DVH), a histogram relating radiation dose to tissue volume in radiation therapy planning, is commonly used as a plan evaluation tool to compare doses received by different structures under different plans [45, 46]. In this prostate dataset, there are totally 10 DVH criteria as follows, PTV56: V56$\geq$ 95%; PTV68: V68$\geq$ 95%, V74.8$\leq$ 10%; Rectum: V30$\leq$ 80%, V50$\leq$ 50%, V65$\leq$ 25%; Bladder: V40$\leq$ 70%, V65$\leq$ 30%; Left femoral head: V50$\leq$ 1%; Right femoral head: V50$\leq$ 1%. For this dataset, we have $3,047,040$ voxels, 180 angles and over 2×10^{30} potential apertures in each angle.

Since a smaller number of angles results in shorter treatment duration, we study the quality of the treatment plan generated when enforcing the group sparsity requirement with different Φ in (3.98e). In order to balance the scale of the constraint violation, we normalized all the constraints (3.98c)-(3.98e) by dividing both sides of the inequalities by the right hand side b_i or Φ. The total number of apertures in a typical treatment plan for this dataset

Table 3.3: Group Sparsity

Φ	# of apertures	# of angles	Obj. Val.	Con. Vio.
1	96	39	0.0902	0
0.1	96	39	0.0902	0
0.005	96	8	0.1027	0.098
0.0005	97	3	0.1357	0.0589

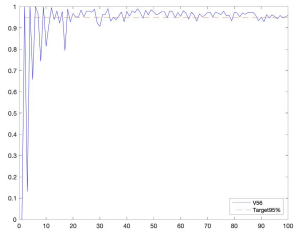
would not be greater than 100. Thus, we set the iteration limit to 100 since the CoexDurCG algorithm generates at most one new aperture in each iteration.

Table 3.3 shows the number of apertures/angles, objective value and constraints violation for different solutions given different values of Φ. Figure 3.1 plots the DVH performance of the generated treatment plans by presenting how the percentage of voxels in each organs changes over different iterations. If $\Phi = 1$, the constraint (3.98e) is redundant and we obtain a solution with the smallest function value and zero constraints violation, but with the largest number of angles as shown in Table 3.3. In addition, the plots in the first column (i.e., parts (a), (d), (g), (j) and (m)) of Figure 3.1 show that the generated plan satisfy all the DVH criteria. Comparing the first two rows in Table 3.3, we see that the solutions remain the same when $\Phi \geq 0.1$. By keeping decreasing Φ, we can obtain solutions with fewer angles. Plots in the second column of Figure 3.1 shows that most DVH criteria are still satisfied even if the number of angles in the solution reduces from 39 to 8. Moreover, the number of angles can be decreased to 3 if we are willing to sacrifice certain DVH criteria as we can see from the plots in the third column of Figure 3.1.

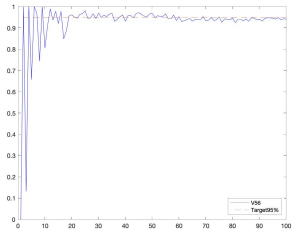
3.5 Concluding Remarks

In this chapter, we propose new constraint-extrapolated conditional gradient (CoexCG) methods for solving general convex optimization problems with function constraints. These methods are simple and requires only linear optimization rather than projection over the simple convex set X. We establish the $\mathcal{O}(1/\epsilon^2)$ iteration complexity of the CoexCG method and show that the same complexity bound still holds even if the objective or constraint func-

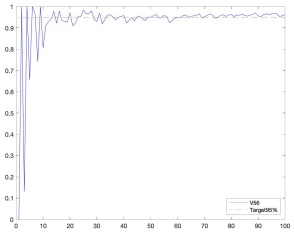
tions are nonsmooth with certain structures. We further present dual regularized algorithms that does not require us to fix the number of iterations a priori and show that they can attain similar complexity bounds to CoexCG. Effectiveness of these methods are demonstrated for solving a class of challenging function constrained convex optimization problems arising from IMRT treatment planning.



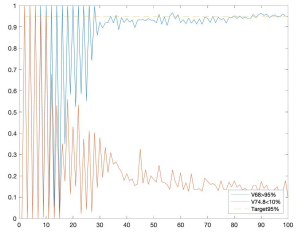
(a) PTV56 when $\Phi = 1$



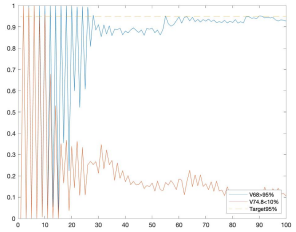
(b) PTV56 when $\Phi = 0.005$



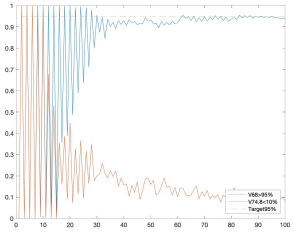
(c) PTV56 when $\Phi = 0.0005$



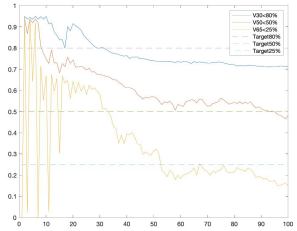
(d) PTV68 when $\Phi = 1$



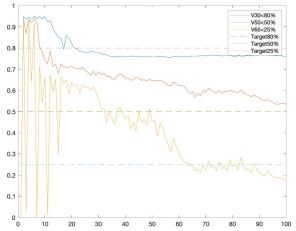
(e) PTV68 when $\Phi = 0.005$



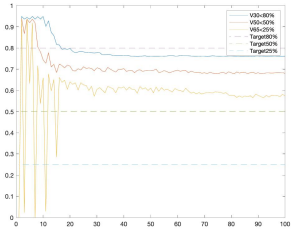
(f) PTV68 when $\Phi = 0.0005$



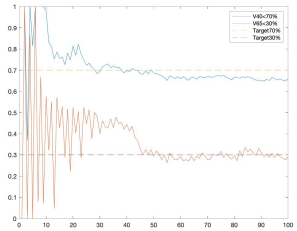
(g) Rectum when $\Phi = 1$



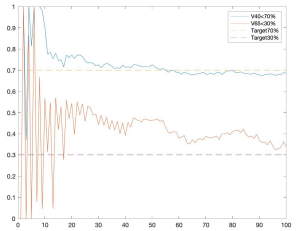
(h) Rectum when $\Phi = 0.005$



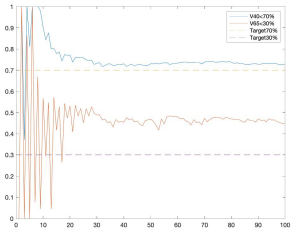
(i) Rectum when $\Phi = 0.0005$



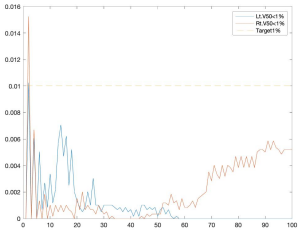
(j) Bladder when $\Phi = 1$



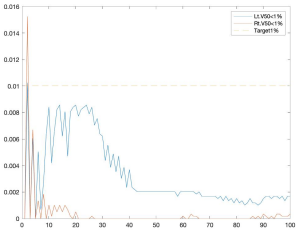
(k) Bladder when $\Phi = 0.005$



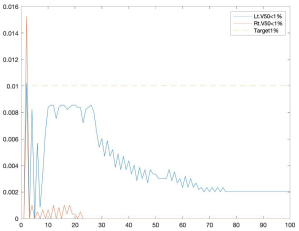
(l) Bladder when $\Phi = 0.0005$



(m) Lt. & Rt. when $\Phi = 1$



(n) Lt. & Rt. when $\Phi = 0.005$



(o) Lt. & Rt. when $\Phi = 0.0005$

Figure 3.1: Percentage of voxels in different organs

CHAPTER 4
DYNAMIC STOCHASTIC APPROXIMATION FOR MULTI-STAGE STOCHASTIC OPTIMIZATION

4.1 Introduction

In this chapter, we attempt to shed some light on this problem by presenting a dynamic stochastic approximation (DSA) method for multi-stage stochastic optimization. The basic idea of the DSA method is to apply an inexact primal-dual SA method for solving the t-th stage optimization problem to compute an approximate stochastic subgradient for its associated value functions v^t. In the pursuit of this idea, we manage to resolve the following difficulties. First, the first-order information for the value function v^{t+1} used to solve the t-stage subproblem is not only stochastic, but also biased. We need to control the bias associated with such first-order information. In addition, we need to develop a relationship between the primal-dual gap and the error associated with approximate stochastic subgradients. Second, in order to establish the convergence of stochastic optimization subroutines for solving the t-stage problem, we need to guarantee that the variance of approximate stochastic subgradients and hence the dual multipliers associated with the $(t+1)$-stage problem are bounded, while no such results exist in the current SA literature. Third, we need to make sure that the errors associated with approximate stochastic subgradients do not accumulate quickly as the number of stages T increases. By properly addressing these issues, we were able to show that the DSA method can achieve an optimal $\mathcal{O}(1/\epsilon^4)$ rate of convergence in terms of the number of random samples when applied to a three-stage stochastic optimization problem. We further show that this rate of convergence can be improved to $\mathcal{O}(1/\epsilon^2)$ when the objective function is strongly convex. To the best of our knowledge, this is the first time that this improved $\mathcal{O}(1/\epsilon^2)$ complexity has been obtained

for solving three-stage problems under the strong convexity setting. Even though the value functions for these problems are still convex (rather than strongly convex), by exploiting the structural information that the cost function h^t at each stage is strongly convex, our algorithm can compute the approximate stochastic subgradients more efficiently than the more general situation where the cost function h^t at each stage is convex. Moreover, we discuss variants of the DSA method which exhibit optimal rate of convergence for solving more general multi-stage stochastic optimization problems with $T > 3$. The developed DSA algorithms only need to go through the scenario tree once in order to compute an ϵ-solution of the multi-stage stochastic optimization problem. As a result, the required memory for DSA increases only linearly with respect to T. To the best of our knowledge, this is the first time that stochastic approximation type methods are generalized to and their complexities are established for multi-stage stochastic optimization. It should be also mentioned that although the main motivation and contribution of this chapter lie on the theoretical side of stochastic optimization, the developed DSA algorithm provides an effective approach for solving stochastic optimization problems with a large number of decision variables and a relatively smaller number of stages such as for those arising from hierarchical operations management and clinical trials.

This chapter is organized as follows. In Section 2, we introduce the basic scheme of the DSA algorithm and establish its main convergence properties for solving three-stage stochastic optimization problems. In Section 3, we show that the convergence rate of the DSA algorithm can be significantly improved under the strongly convex assumption on the objective function at each stage. and we then develop variants of the DSA method for solving more general form of (1.9) with $T > 3$ in Section 4. Finally, some concluding remarks are made in Section 4.6.

4.1.1 Notation and terminology

For a closed convex set X, a function $\omega_X : X \mapsto R$ is called a distance generating function with parameter α_X, if ω_X is continuously differentiable and strongly convex with parameter α_X with respect to $\|\cdot\|$. Therefore, we have

$$\langle y - x, \nabla\omega_X(y) - \nabla\omega_X(x)\rangle \geq \alpha_X\|y - x\|^2, \forall x, y \in X.$$

The prox-function associated with ω_X is given by

$$P_X(x, y) = \omega_X(y) - \omega_X(x) - \langle\nabla\omega_X(x), y - x\rangle, \forall x, y \in X.$$

It can be easily seen that

$$P_X(x, y) \geq \tfrac{\alpha_X}{2}\|y - x\|^2, \ \forall x, y \in X. \tag{4.1}$$

If X is bounded, we define the diameter of the set X as

$$\Omega_X^2 := \max_{x,y\in X} P_X(x, y). \tag{4.2}$$

For a given closed convex cone K_*, we choose the distance generating function $\omega_{K_*}(y) = \|y\|_2^2/2$. For simplicity, we often skip the subscript of $\|\cdot\|_2$ whenever we apply it to an unbounded set (such as a cone).

For a given closed convex set $X \subseteq \mathbb{R}^n$ and a closed convex function $V : X \to \mathbb{R}$, $g(x)$ is called an ϵ-subgradient of V at $x \in X$ if

$$V(y) \geq V(x) + \langle g(x), y - x\rangle - \epsilon \ \forall y \in X. \tag{4.3}$$

The collection of all such ϵ-subgradients of V at x is called the ϵ-subdeifferential of V at

x, denoted by $\partial_\epsilon V(x)$.

Assume that V is Lipschitz continuous in an ϵ-neighborhood of X, i.e.,

$$|V(y)-V(x)| \leq M_0\|y-x\|,\ \forall x,y \in X_\epsilon := \{p \in \mathbb{R}^n : p = r+x, x \in X, \|r\| \leq \epsilon\}. \quad (4.4)$$

We can show that

$$\|g(x)\|_* \leq M_0 + 1\ \ \forall x \in X. \quad (4.5)$$

Indeed, if $\|\cdot\| = \|\cdot\|_2$, the result follows immediately by setting $d = \epsilon g(x)/\|g(x)\|_2$ and $y = x + d$ in (4.3). Otherwise, we need to choose d properly s.t. $\|d\| = \epsilon$ and $\langle g(x), d\rangle = \epsilon\|g(x)\|_*$. It should be noted, however, that if V is Lipschitz continuous over X (rather than X_ϵ), then one cannot guarantee the boundedness of an ϵ-subgradient of V.

4.2 Three-stage problems with generally convex objectives

Our goal in this section is to introduce the basic scheme of the DSA algorithm and discuss its convergence properties. For the sake of simplicity, we will focus on three-stage stochastic optimization problems with simple convex objective functions in this section. Extensions to strongly convex cases and more general form of multi-stage stochastic optimization problems will be studied in later sections.

4.2.1 Value functions and stochastic ϵ-subgradients

Consider the following three-stage stochastic programming problem:

$$\begin{array}{lll} \min h^1(x^1,c^1)+ & \mathbb{E}_{|\xi^1}[\min\ h^2(x^2,c^2)+ & \mathbb{E}_{|\xi^{[2]}}[\min\ h^3(x^3,c^3)]] \\ \text{s.t. } A^1x^1 - b^1 \in K^1 & \text{s.t.} A^2x^2 - b^2 - B^2x^1 \in K^2 & \text{s.t. } A^3x^3 - b^3 - B^3x^2 \in K^3, \\ x^1 \in X^1, & x^2 \in X^2, & x^3 \in X^3. \end{array} \quad (4.6)$$

As a particular example, if $h^t(x^t, c^t) = \langle c^t, x^t \rangle$, $K^t = \{0\}$ and X^t are polyhedronal, then problem (4.6) reduces to a well-known three-stage stochastic linear programming problem.

We can write problem (4.6) in a more compact form by using value functions as discussed in Section 4.1. More specifically, let $V^3(x^2, \xi^3|\xi^2)$ be the stochastic value function at the third stage and $v^3(x^2)$ be the corresponding expected value function conditionally on $\xi^{[2]}$:

$$\begin{aligned} V^3(x^2, \xi^{[3]}) &:= \min\ h^3(x^3, c^3) \\ &\qquad \text{s.t. } A^3x^3 - b^3 - B^3x^2 \in K^3, \\ &\qquad\qquad x^3 \in X^3. \\ v^3(x^2, \xi^{[2]}) &:= \mathbb{E}[V^3(x^2, \xi^{[3]})|\xi^{[2]}]. \end{aligned} \tag{4.7}$$

We can then define the stochastic value function $V^2(x^1, \xi^2)$ and its corresponding (expected) value function as

$$\begin{aligned} V^2(x^1, \xi^{[2]}) &:= \min\ \left\{h^2(x^2, c^2) + v^3(x^2, \xi^{[2]})\right\} \\ &\qquad \text{s.t. } A^2x^2 - b^2 - B^2x^1 \in K^2, \\ &\qquad\qquad x^2 \in X^2. \\ v^2(x^1, \xi^1) &:= \mathbb{E}[V^2(x^1, \xi^{[2]})|\xi^1] = \mathbb{E}[V^2(x^1, \xi^2)]. \end{aligned} \tag{4.8}$$

Problem (4.6) can then be formulated equivalently as

$$\begin{aligned} &\min\ \{h^1(x^1, c^1) + v^2(x^1, \xi^1)\} \\ &\text{s.t. } A^1x^1 - b^1 \in K^1, \\ &\qquad x^1 \in X^1. \end{aligned} \tag{4.9}$$

Throughout this chapter, we assume that the expected value functions $v^2(x^1, \xi^1)$ and $v^3(x^2, \xi^{[2]})$, respectively, are well-defined and finite-valued for a given ξ^1 and any $x^1 \in X^1$, and any $x^2 \in X^2, \xi^2 \in \Xi^2$ almost surely. We observe that the assumption that the values functions are well-defined holds under various regularity conditions (see Section 3.2 of [92] for a more detailed discussion). It is also worth noting that in the above formulation, we

assume that the value functions v^t depend on the immediately preceding decisions x^{t-1}, rather than all earlier decisions $x^1, \ldots, x^{t-1}$ for the sake of convenience. In the latter case, one can reformulate the problems in the form of (4.8) by introducing the so-called model state variables (Section 3.1.2 of [92]).

In order to solve problem (4.9), we need to understand how to compute first-order information about the value functions v^2 and v^3. Since both v^2 and v^3 are given in the form of (conditional) expectation, their exact first-order information is hard to compute. We resort to the computation of a stochastic ϵ-subgradient of these value functions defined as follows.

Definition 11 *$G(u, \xi^{[t]})$ is called a stochastic ϵ-subgradient of the value function $v^t(u, \xi^{[t-1]}) = \mathbb{E}[V^t(u, \xi^{[t]})|\xi^{[t-1]}]$ if $G(u, \xi^{[t]})$ is an unbiased estimator of an ϵ-subgradient of $v^t(u, \xi^{[t-1]})$ with respect to u, i.e.,*

$$\mathbb{E}[G(u, \xi)|\xi^{[t-1]}] = g(u, \xi^{[t-1]}) \ \textit{and} \ g(u, \xi^{[t-1]}) \in \partial_\epsilon v^t(u, \xi^{[t-1]}). \tag{4.10}$$

To compute a stochastic ϵ-subgradient of v^2 (resp., v^3), we have to compute an approximate subgradient of the corresponding stochastic value function $V^2(x^1, \xi^{[2]})$ (resp., $V^3(x^2, \xi^{[3])})$. To this end, we further assume that strong Lagrange duality holds for the optimization problems defined in (4.8) (resp.,(4.7)) almost surely. In other words, these problems can be formulated as saddle point problems:

$$V^2(x^1, \xi^{[2]}) = \max_{y^2 \in K_*^2} \min_{x^2 \in X^2} \langle b^2 + B^2 x^1 - A^2 x^2, y^2 \rangle + h^2(x^2, c^2) + v^3(x^2, \xi^{[2]}), \tag{4.11}$$

$$V^3(x^2, \xi^{[3]}) = \max_{y^3 \in K_*^3} \min_{x^3 \in X^3} \langle b^3 + B^3 x^2 - A^3 x^3, y^3 \rangle + h^3(x^3, c^3), \tag{4.12}$$

where K_*^2 and K_*^3 are corresponding dual cones to K^2 and K^3, respectively. One set of sufficient conditions to guarantee the equivalence between (4.8) (resp.,(4.7)) and (4.11) (resp., (4.12)) is that (4.8) (resp.,(4.7)) is solvable and the slater condition holds [93].

Observe that in order to solve (4.11) and (4.12), we need to solve a more generic saddle point problem:

$$V(u,\xi) \equiv V(u,(A,b,B,C)) := \max_{y\in K_*}\min_{x\in X}\langle b+Bu-Ax,y\rangle + h(x,c)+\tilde{v}(x), \quad (4.13)$$

where $A:\mathbb{R}^n \to m$ and $B:\mathbb{R}^{n_0}\to m$ denote the linear mappings. For example, (4.12) is a special case of (4.13) with $u=x^2$, $y=y^3$, $K_*=K_*^3$, $b=b^3$, $B=B^3$, $A=A^3$, $h=h^3$ and $\tilde{v}=0$. It is worth noting that the first stage problem can also be viewed as a special case of (4.13), since (4.9) is equivalent to

$$\max_{y\in K_*^1}\min_{x^1\in X^1}\left\{\langle b^1-A^1x^1,y^1\rangle + h^1(x^1,c^1)+v^2(x^1,\xi^1)\right\}. \quad (4.14)$$

Let

$$(x_*,y_*)\in Z\equiv X\times K_*$$

be a pair of optimal solutions of the saddle point problem (4.11), i.e.,

$$V(u,\xi)=\langle y_*,b+Bu-Ax_*\rangle + h(x_*,c)+\tilde{v}(x_*)=h(x_*,c)+\tilde{v}(x_*), \quad (4.15)$$

where the second identity follows from the complementary slackness of Lagrange duality. Below we provide a different characterization of an ϵ-subgradient of V other than the one in (4.3).

Lemma 25 *Let $\bar{z}:=(\bar{x},\bar{y})\in Z$ and $u\in\mathbb{R}^{n_0}$ be given. If*

$$\begin{aligned} Q(\bar{z};x,y_*) &:= \langle y_*,b+Bu-A\bar{x}\rangle + h(\bar{x},c)+\tilde{v}(\bar{x}) \\ &\quad -\langle \bar{y},b+Bu-Ax\rangle - h(x,c)-\tilde{v}(x)\le\epsilon,\ \forall x\in X, \end{aligned} \quad (4.16)$$

then $B^T\bar{y}$ is an ϵ-subgradient of $V(u,\xi)$ at u.

Proof. For simplicity, let us denote $V(u)\equiv V(u,\xi)$. For any $u_1\in \mathrm{dom}V$, we denote

(x_1^*, y_1^*) as a pair of primal-dual solution of (4.13) (with $u = u_1$). Hence,

$$V(u_1) = \langle y_1^*, b + Bu_1 - Ax_1^* \rangle + h(x_1^*, c) + \tilde{v}(x_1^*). \tag{4.17}$$

It follows from the definition of V in (4.13) and (4.16) that

$$\begin{aligned} V(u) &= \langle y_*, b + Bu - Ax_* \rangle + h(x_*, c) + \tilde{v}(x_*) \\ &\leq \langle y_*, b + Bu - A\bar{x} \rangle + h(\bar{x}, c) + \tilde{v}(\bar{x}) \\ &\leq \langle \bar{y}, b + Bu - Ax_1^* \rangle + h(x_1^*, c) + \tilde{v}(x_1^*) + \epsilon. \end{aligned} \tag{4.18}$$

Observe that

$$\begin{aligned} \langle \bar{y}, b + Bu - Ax_1^* \rangle &= \langle \bar{y}, B(u - u_1) \rangle + \langle \bar{y}, b + Bu_1 - Ax_1^* \rangle \\ &\leq \langle \bar{y}, B(u - u_1) \rangle + \langle y_1^*, b + Bu_1 - Ax_1^* \rangle, \end{aligned}$$

where the last inequality follows from the assumption that (x_1^*, y_1^*) is a pair of optimal solution of (4.13) with $u = u_1$. Combining these two observations and using (4.17), we have

$$V(u) \leq \langle B^T \bar{y}, u - u_1 \rangle + V(u_1) + \epsilon,$$

which, in view of (4.3), implies that $B^T \bar{y}$ is an ϵ-subgradient of $V(u)$. ∎

In view of Lemma 25, in order to compute a stochastic subgradient of $v^t(u, \xi^{[t-1]}) = \mathbb{E}[V^t(u, \xi^{[t]})|\xi^{[t-1]}]$ at a given point u, we can first generate a random realization ξ^t conditionally on $\xi^{[t-1]}$ and then try to find a pair of solutions $(\bar{x}, \bar{y})$ satisfying

$$\begin{aligned} &\langle y_*^t, b^t + B^t u - A^t \bar{x} \rangle + h(\bar{x}, c^t) + v^{t+1}(\bar{x}, \xi^{[t]}) \\ &\qquad - \langle \bar{y}, b^t + B^t u - A^t x \rangle - h(x, c^t) - v^{t+1}(x, \xi^{[t]}) \leq \epsilon, \ \forall x \in X, \end{aligned}$$

where $y_*^t \equiv y_*^t(\xi^{[t]})$ denotes the optimal solution for the t-th stage problem associated

with the random realization $\xi^{[t]}$. We will then use $B^T\bar{y}$ as a stochastic ϵ-subgradient of $v^t(u, \xi^{[t-1]})$ at u. However, the difficulty associated with this approach exists in that the function $v^{t+1}(\bar{x}, \xi^{[t]})$ is also given in the form of expectation. We will explore this approach and discuss how to address these issues in more details in the next subsection.

4.2.2 The DSA algorithm

Our goal in this subsection is to present the basic scheme of our dynamic stochastic approximation algorithm applied to problem (4.9).

Our algorithm relies on the following three key primal-dual steps, referred to as stochastic primal-dual transformation (SPDT), applied to the generic saddle point problem in (4.13) at every stage.

$(p_+, d_+, \tilde{d}) = \mathrm{SPDT}(p, d, d_-, \tilde{v}', u, \xi, h, X, K_*, \theta, \tau, \eta)$:

$$\tilde{d} = \theta(d - d_-) + d. \tag{4.19}$$

$$p_+ = \mathrm{argmin}_{x\in X}\langle b + Bu - Ax, \tilde{d}\rangle + h(x, c) + \langle \tilde{v}', x\rangle + \tau P_X(p, x). \tag{4.20}$$

$$d_+ = \mathrm{argmin}_{y\in K_*}\langle -b - Bu + Ap_+, y\rangle + \tfrac{\eta}{2}\|y - d\|^2. \tag{4.21}$$

In the above primal-dual tranformation, the input (p, d, d_-) denotes the current primal solution, dual solution, and the previous dual solution, respectively. Moreover, the input $\tilde{v}'$ denotes a stochastic ϵ-subgradient for $\tilde{v}$ at the current search point p. The parameters (u, ξ, h, X, K_*) describes the problem in (4.13) and (θ, τ, η) are certain algorithmic parameters to be specified. Given these input parameters, the relation in (4.19) defines a dual extrapolation (or prediction) step to estimate the dual variable $\tilde{d}$ for the next iterate. Based on this estimate, (4.20) performs a primal prox-mapping to compute p_+, and then (4.21) updates in the dual space to compute d_+ by using the updated p_+. We assume that the above SPDT operator can be performed very fast or even has explicit expressions. The primal-dual transformation is closely related to the alternating direction method of multipliers and

was first formally presented by Chambolle and Pork in [94] for solving saddle point problems. Its inherent relationship with Nesterov's acceleration has also been recently studied by Lan and Zhou [95].

Observe that by the optimality conditions of (4.20) and (4.21) (see, e.g., Lemma 1 of [78]), the solution $(p_+, d_+, \tilde{d})$ obtained from SPDT satisfies

$$\langle -A(p_+ - x), \tilde{d}\rangle + h(p_+, c) - h(x, c) + \langle \tilde{v}', p_+ - x\rangle \leq \tau[P_X(p, x) - P_X(p_+, x) - P_X(p, p_+)], \forall x \in X, \quad (4.22)$$

$$\langle -b - Bu + Ap_+, d_+ - y\rangle \leq \tfrac{\eta}{2}[\|d - y\|^2 - \|d_+ - y\|^2 - \|d_+ - d\|^2], \forall y \in K_*. \quad (4.23)$$

In order to solve problem (4.9), we will combine the above primal-dual transformation applied to all the three stages, the scenario generation for the random variables ξ^2 and ξ^3 in the second and third stage, and certain averaging steps in both the primal and dual spaces. We are now ready to describe the basic scheme of the DSA algorithm.

Algorithm 6 The basic DSA algorithm for three-stage problems

Input: initial points (z_0^1, z_0^2, z_0^3).

$\xi^1 = (A^1, b^1, c^1)$.

for $i = 1, 2, \ldots, N_1$ **do**

Generate a random realization of $\xi_i^2 = (A_i^2, B_i^2, b_i^2, c_i^2)$.

for $j = 1, 2, \ldots, N_2$ **do**

Generate a random realization of $\xi_j^3 = (A_j^3, B_j^3, b_j^3, c_j^3)$ (conditional on ξ_i^2).

for $k = 1, 2, \ldots, N_3$ **do**

$(x_k^3, y_k^3, \tilde{y}_k^3) = \mathrm{SPDT}(x_{k-1}^3, y_{k-1}^3, y_{k-2}^3, 0, x_{j-1}^2, \xi_j^3, h^3, X^3, K_*^3, \theta_k^3, \tau_k^3, \eta_k^3)$.

end for

$(\bar{x}_j^3, \bar{y}_j^3) = \sum_{k=1}^{N_3} w_k^3 (x_k^3, y_k^3) / \sum_{k=1}^{N_3} w_k^3$.

$(x_j^2, y_j^2, \tilde{y}_j^2) = \mathrm{SPDT}(x_{j-1}^2, y_{j-1}^2, y_{j-2}^2, (B_j^3)^T \bar{y}_j^3, x_{i-1}^1, \xi_i^2, h^2, X^2, K_*^2, \theta_j^2, \tau_j^2, \eta_j^2)$.

end for

$(\bar{x}_i^2, \bar{y}_i^2) = \sum_{j=1}^{N_2} w_j^2 (x_j^2, y_j^2) / \sum_{j=1}^{N_2} w_j^2$.

$(x_i^1, y_i^1, \tilde{y}_i^1) = \mathrm{SPDT}(x_{i-1}^1, y_{i-1}^1, y_{i-2}^1, (B_i^2)^T \bar{y}_i^2, 0, \xi^1, h^1, X^1, K_*^1, \theta_i^1, \tau_i^1, \eta_i^1)$.

end for

Output: $(\bar{x}^1, \bar{y}^1) = \sum_{i=1}^{N_1} w_i^1 (x_i^1, y_i^1) / \sum_{i=1}^{N_1} w_i^1$.

This algorithm consists of three loops. The innermost (third) loop runs N_3 steps of SPDT in order to compute an approximate stochastic subgradient ($(B_j^3)^T \bar{y}_j^3$) of the value function v^3 of the third stage. The second loop consists of N_2 SPDTs applied to the saddle point formulation of the second-stage problem, which requires the output from the third loop. The outer loop applies N_1 SPDTs to the saddle point formulation of the first-stage optimization problem in (4.9), using the approximate stochastic subgradients ($(B_i^2)^T \bar{y}_i^2$) for v^2 computed by the second loop. In this algorithm, we need to generate N_1 and $N_1 \times N_2$ realizations for the random vectors ξ^2 and ξ^3, respectively. Observe that the DSA algorithm described above is conceptual only since we have not specified any algorithmic parameters yet. We will come back to this issue after establishing some general convergence properties

about this method in the next two subsections.

4.2.3 Basic tools: inexact primal-dual stochastic approximation

In this subsection, we provide some basic tools for the convergence analysis of the DSA method. In particular, we will develop an inexact primal-dual stochastic approximation (I-PDSA) method (see Algorithm 2), which consists of iterative applications of the SPDTs defined in (4.19), (4.20) and (4.21) to solve the generic stochastic saddle point problem in (4.13).

The I-PDSA method evolves from the primal-dual method in [94], an efficient and simple method for solving saddle point problems. While the primal-dual method in [94] can be viewed as a refined version of the primal-dual hybrid gradient method by Arrow et al. [96], its design and analysis is more closely related to a few recent important works which established the $\mathcal{O}(1/k)$ rate of convergence for solving bilinear saddle point problems (e.g., [89, 90, 97, 98]). In particular, it is equivalent to a linearized version of the alternative direction method of multipliers. The first stochastic version of the primal-dual method was studied by Chen, Lan and Ouyang [77] together with an acceleration scheme and an extension to non-Euclidean projection. Using a special non-Euclidean geometry, Lan and Zhou [95] further established an inherent relationship between the primal-dual method and Nesterov's accelerated gradient method. However, to the best of our knowledge, none of existing stochastic primal-dual methods can deal with biased stochastic subgradient information for the value function $\tilde{v}$. Moreover, in order to generate an approximate stochastic subgradient of $V(\cdot, \xi)$ with bounded variance, we will show how to guarantee the boundedness of output dual solution, while none of existing stochastic optimization methods, including stochastic primal-dual methods, can guarantee the boundedness of the generated solutions.

Algorithm 7 Inexact primal-dual stochastic approximation

$\xi = (A, B, b, c)$.
for $k = 1, 2, \cdots, N$ **do**
Let G_{k-1} be a stochastic, independent of x_{k-1}, $\bar{\epsilon}$-subgradient of $\tilde{v}$, i.e.,

$$g(x_{k-1}) \equiv \mathbb{E}[G_{k-1}] \in \partial_{\bar{\epsilon}} \tilde{v}(x_{k-1}). \tag{4.24}$$

$(x_k, y_k, \tilde{y}_k) = \mathrm{SPDT}(x_{k-1}, y_{k-1}, y_{k-2}, G_{k-1}, u, \xi, h, X, K_*, \theta_k, \tau_k, \eta_k)$.
end for
Output: $\bar{z}_N \equiv (\bar{x}_N, \bar{y}_N) = \sum_{k=1}^N w_k(x_k, y_k) / \sum_{k=1}^N w_k$.

Throughout this subsection, we assume that there exists $M > 0$ such that

$$\mathbb{E}[\|G_k\|_*^2] \leq M^2 \ \ \forall k \geq 1. \tag{4.25}$$

This assumption, in view of (4.24) and Jensen's inequality, then implies that $\|g(x_k)\|_* \leq M$. For notational convenience, we assume that the Lipschitz constant of the function $\tilde{v}$ is also bounded by M. Indeed, by definition, any exact subgradient can be viewed as an $\bar{\epsilon}$-subgradient. Hence, the size of subgradient (and the Lipschtiz constant of $\tilde{v}$) can also be bounded by M. Since the condition in (4.4) about the Lipschitz continuity of the value function $\tilde{v}$ over a neighborhood of X is hard to verify in practice, we will discuss different ways to ensure that the assumption in (4.25) holds later in this section (see Corollary 31).

Below we discuss some convergence properties for Algorithm 2. More specifically, we will first establish in Proposition 26 the relation between (x_{k-1}, y_{k-1}) and (x_k, y_k) after running one step of SPDT, and then discuss in Theorems 27 and 29 the convergence properties of Algorithm 2 applied to problem (4.13). A few consequences of these results will be discussed in Corollary 30 and Corollary 31. Moreover, we will establish some technical results regarding our termination criterion and the size of the dual multipliers in Lemma 32 and Lemma 33, respectively.

Proposition 26 *Let Q be defined in (4.16). For any $1 \leq k \leq N$ and $(x, y) \in X \times K_*$, we*

have

$$
\begin{aligned}
&Q(z_k, z) + \langle A(x_k - x), y_k - y_{k-1}\rangle - \theta_k\langle A(x_{k-1} - x), y_{k-1} - y_{k-2}\rangle \\
&\le \tau_k[P_X(x_{k-1}, x) - P_X(x_k, x)] + \tfrac{\eta_k}{2}(\|y - y_{k-1}\|^2 - \|y - y_k\|^2) - \tfrac{\alpha_X \tau_k}{2}\|x_k - x_{k-1}\|^2 \\
&\quad - \tfrac{\eta_k}{2}\|y_{k-1} - y_k\|^2 + \langle \Delta_{k-1}, x_{k-1} - x\rangle + (M + \|G_{k-1}\|_*)\|x_k - x_{k-1}\| + \bar{\epsilon} \\
&\quad + \theta_k\langle A(x_k - x_{k-1}), y_{k-1} - y_{k-2}\rangle,
\end{aligned} \tag{4.26}
$$

where

$$
\Delta_k := g(x_k) - G_k. \tag{4.27}
$$

Proof. Denote $\xi = (A, B, b, c)$. By the Lipschitz continuity of $\tilde{v}$ and the definition of an $\bar{\epsilon}$-subgradient, we have

$$
\begin{aligned}
\tilde{v}(x_k) &\le \tilde{v}(x_{k-1}) + M\|x_k - x_{k-1}\| \\
&\le \tilde{v}(x) + \langle g(x_{k-1}), x_{k-1} - x\rangle + M\|x_k - x_{k-1}\| + \bar{\epsilon}.
\end{aligned}
$$

Moreover, by (4.27), we have

$$
\begin{aligned}
\langle g(x_{k-1}), x_{k-1} - x\rangle &= \langle G_{k-1}, x_{k-1} - x\rangle + \langle \Delta_{k-1}, x_{k-1} - x\rangle \\
&= \langle G_{k-1}, x_k - x\rangle + \langle G_{k-1}, x_{k-1} - x_k\rangle + \langle \Delta_{k-1}, x_{k-1} - x\rangle \\
&\le \langle G_{k-1}, x_k - x\rangle + \|G_{k-1}\|_*\|x_k - x_{k-1}\| + \langle \Delta_{k-1}, x_{k-1} - x\rangle.
\end{aligned}
$$

Combining the above two inequalities, we obtain

$$
\tilde{v}(x_k) - \tilde{v}(x) \le \langle G_{k-1}, x_k - x\rangle + \langle \Delta_{k-1}, x_{k-1} - x\rangle + (M + \|G_{k-1}\|_*)\|x_k - x_{k-1}\| + \bar{\epsilon}. \tag{4.28}
$$

Moreover, by (4.22) and (4.23) (with input $p = x_{k-1}, d = y_{k-1}, d_- = y_{k-2}, \tilde{v}' = G_{k-1}, u = u, h = h, X = X, K_* = K_*, \theta = \theta_k, \tau = \tau_k, \eta = \eta_k$, output $(p_+, d_+, \tilde{d}) = (x_k, y_k, \tilde{y}_k)$, we

have

$$\langle -A(x_k - x), \tilde{y}_k \rangle + h(x_k, c) - h(x, c) + \langle G_{k-1}, x_k - x \rangle \\ \leq \tau_k [P_X(x_{k-1}, x) - P_X(x_k, x) - P_X(x_{k-1}, x_k)], \forall x \in X, \tag{4.29}$$

$$\langle -b - Bu + Ax_k, y_k - y \rangle \leq \tfrac{\eta_k}{2} [\|y_{k-1} - y\|^2 - \|y_k - y\|^2 - \|y_{k-1} - y_k\|^2], \forall y \in K_*. \tag{4.30}$$

Using the definition of Q in (4.16) and the relations (4.28), (4.29) and (4.30), we have

$$Q(z_k, z) + \langle A(x_k - x), y_k - \tilde{y}_k \rangle \leq \tau_k [P_X(x_{k-1}, x) - P_X(x_k, x)] + \tfrac{\eta_k}{2} [\|y_{k-1} - y\|^2 - \|y_k - y\|^2] \\ - \tau_k P_X(x_{k-1}, x_k) - \tfrac{\eta_k}{2} \|y_{k-1} - y_k\|^2 + \langle \Delta_{k-1}, x_{k-1} - x \rangle + (M + \|G_{k-1}\|_*) \|x_k - x_{k-1}\| + \bar{\epsilon}.$$

Also note that by the definition of $\tilde{y}_k$ (i.e., $\tilde{d}$ in (4.19)), we have $\tilde{y}_k = \theta_k(y_{k-1} - y_{k-2}) + y_{k-1}$ and hence

$$\begin{aligned} \langle A(x_k - x), y_k - \tilde{y}_k \rangle &= \langle A(x_k - x), y_k - y_{k-1} \rangle - \theta_k \langle A(x_k - x), y_{k-1} - y_{k-2} \rangle \\ &= \langle A(x_k - x), y_k - y_{k-1} \rangle - \theta_k \langle A(x_{k-1} - x), y_{k-1} - y_{k-2} \rangle \\ &\quad - \theta_k \langle A(x_k - x_{k-1}), y_{k-1} - y_{k-2} \rangle. \end{aligned}$$

Our result then immediately follows from the above two relations and the strong convexity of P_X (see (4.1)). ∎

We are now ready to establish some important convergence properties for the iterative applications of SPDTs stated in Algorithm 2.

Theorem 27 *If the parameters $\{\theta_k\}$, $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$ in Algorithm 2 satisfy*

$$\begin{aligned} w_k\theta_k &= w_{k-1}, 1 \le k \le N, && (a)\\ w_k\tau_k &\ge w_{k+1}\tau_{k+1}, 1 \le k \le N-1, && (b)\\ w_k\eta_k &\ge w_{k+1}\eta_{k+1}, 1 \le k \le N-1, && (c)\\ w_k\tau_k\eta_{k-1}\alpha_X &\ge 2w_{k-1}\|A\|^2, 1 \le k \le N-1, && (d)\\ \tau_N\eta_N\alpha_X &\ge 2\|A\|^2, && (e) \end{aligned} \tag{4.31}$$

then we have

$$Q(\bar{z}_N, z) \le \frac{1}{\sum_{k=1}^N w_k}\left(w_1\tau_1 P_X(x_0, x) + \frac{w_1\eta_1}{2}\|y_0 - y\|^2 - \frac{w_N\eta_N}{2}\|y_N - y\|^2 + \sum_{k=1}^N \Lambda_k\right) \tag{4.32}$$

for any $z \in Z$, where

$$\Lambda_k := w_k\left[(M + \|G_{k-1}\|_*)^2/(\alpha_X\tau_k) + \langle \Delta_k, x_{k-1} - x\rangle + \bar{\epsilon}\right]. \tag{4.33}$$

Proof. Multiplying both sides of (4.26) by w_k for each $k \ge 1$, summing them up over $1 \le k \le N$ and using the relations in (4.31).a), (4.31).b) and (4.31).c), we have

$$\begin{aligned} &\textstyle\sum_{k=1}^N w_k Q(z_k, z)\\ \le& w_1\tau_1 P_X(x_0, x) + \tfrac{w_1\eta_1}{2}\|y_0 - y\|^2 - \tfrac{w_N\eta_N}{2}\|y_N - y\|^2 + \textstyle\sum_{k=1}^N w_k\bar{\epsilon}\\ &- w_N\tau_N P_X(x_N, x) - w_N\langle A(x_N - x), y_N - y_{N-1}\rangle - \tfrac{w_N\eta_N}{2}\|y_N - y_{N-1}\|^2\\ &- \textstyle\sum_{k=1}^N [\tfrac{\alpha_X w_k\tau_k}{4}\|x_k - x_{k-1}\|^2 + \tfrac{w_{k-1}\eta_{k-1}}{2}\|y_{k-1} - y_{k-2}\|^2\\ &+ w_{k-1}\langle A(x_k - x_{k-1}), y_{k-1} - y_{k-2}\rangle] - \textstyle\sum_{k=1}^N \tfrac{\alpha_X w_k\tau_k}{4}\|x_k - x_{k-1}\|^2\\ &+ \textstyle\sum_{k=1}^N w_k(M + \|G_{k-1}\|_*)\|x_k - x_{k-1}\| + \sum_{k=1}^N w_k\langle \Delta_k, x_{k-1} - x\rangle. \end{aligned} \tag{4.34}$$

Now, by the Cauchy-Schwarz inequality and the strong convexity of P_X and (4.31).e),

$$\begin{aligned}&-\tau_N P_X(x_N,x)-\langle A(x_N-x),y_N-y_{N-1}\rangle-\tfrac{\eta_N}{2}\|y_N-y_{N-1}\|^2\\ \le&-\tfrac{\alpha_X\tau_N}{2}\|x-x_N\|^2+\|A\|\|x_N-x\|\|y_N-y_{N-1}\|-\tfrac{\eta_N}{2}\|y_N-y_{N-1}\|^2\le 0.\end{aligned}$$

Similarly, by the Cauchy-Schwarz inequality and (4.31).d), we have

$$\begin{aligned}&-\textstyle\sum_{k=1}^N[\tfrac{\alpha_X w_k\tau_k}{4}\|x_k-x_{k-1}\|^2+\tfrac{w_{k-1}\eta_{k-1}}{2}\|y_{k-1}-y_{k-2}\|^2\\ &+w_{k-1}\langle A(x_k-x_{k-1}),y_{k-1}-y_{k-2}\rangle]\le 0.\end{aligned}$$

Moreover, using the fact that $-at^2/2+b\le b^2/(2a)$, we can easily see that

$$-\textstyle\sum_{k=1}^N\left[\tfrac{\alpha_X\tau_k}{4}\|x_k-x_{k-1}\|^2+(M+\|G_{k-1}\|_*)\|x_k-x_{k-1}\|\right]\le\sum_{k=1}^N\tfrac{(M+\|G_{k-1}\|_*)^2}{\tau_k\alpha_X}.$$

Using the above three inequalities in (4.34), we have

$$\begin{aligned}\textstyle\sum_{k=1}^N w_kQ(z_k,z)\le&\ w_1\tau_1P_X(x_0,x)+\tfrac{w_1\eta_1}{2}\|y_0-y\|^2-\tfrac{w_N\eta_N}{2}\|y_N-y\|^2\\ &+\textstyle\sum_{k=1}^N w_k\left(\tfrac{(M+\|G_{k-1}\|_*)^2}{\alpha_X\tau_k}+\langle\Delta_k,x_{k-1}-x\rangle+\bar\epsilon\right).\end{aligned}$$

Dividing both sides of above inequality by $\sum_{k=1}^N w_k$, and using the convexity of Q and the definition of $\bar z_N$, we obtain (4.32). ∎

We also need the following technical result for the analysis of Algorithm 2.

Lemma 28 *Let $x_0^v\equiv x_0$ and*

$$x_k^v:=\operatorname{argmin}_{x\in X}\{\langle\Delta_{k-1},x\rangle+\tau_kP_X(x_{k-1}^v,x)\}\tag{4.35}$$

for any $k \geq 1$. Then for any $x \in X$,

$$\sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1}^v - x \rangle \leq \sum_{k=1}^{N} w_k \tau_k [P_X(x_{k-1}, x) - P_X(x_k, x)] + \sum_{k=1}^{N} \frac{w_k \|\Delta_{k-1}\|_*^2}{2\alpha_X \tau_k}. \tag{4.36}$$

Proof. It follows from the definition of x_k^v in (4.35) and Lemma 2.1 of [64] that

$$\tau_k P_X(x_k^v, x) \leq \tau_k P_X(x_{k-1}^v, x) - \langle \Delta_{k-1}, x_{k-1}^v - x \rangle + \frac{\|\Delta_{k-1}\|_*^2}{2\alpha_X \tau_k},$$

for all $k \geq 1$. Multiplying w_k on both sides of the above inequality and summing them up from $k = 1$ to N, we obtain (4.36). ∎

Theorem 29 below provides certain bounds for the following two gap functions:

$$\mathrm{gap}_*(\bar{z}) \equiv \mathrm{gap}_*(\bar{z}, X) := \max \{Q(\bar{z}; x, y_*) : x \in X\}, \tag{4.37}$$

$$\mathrm{gap}_\delta(\bar{z}) \equiv \mathrm{gap}_\delta(\bar{z}, X, K_*) := \max \{Q(\bar{z}, x, y) + \langle \delta, y \rangle : (x, y) \in X \times K_*\}. \tag{4.38}$$

The gap function in (4.37) will be used to measure the error associated with an approximate subgradient, while the perturbed gap function in (4.38) will be used to measure both functional optimality gap and infeasibility of the conic constraint. In particular, we will apply the first gap function to the second and third stage, and the latter one to the first stage when analyzing the DSA algorithm.

Theorem 29 *Suppose the parameters $\{\theta_k\}$, $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$ in Algorithm 2 satisfy* (4.31).

a) *For any $N \geq 1$, we have*

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \leq \left(\sum_{k=1}^{N} w_k\right)^{-1} \left[2w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_* - y_0\|^2 + \sum_{k=1}^{N} \frac{6w_k M^2}{\alpha_X \tau_k}\right] + \bar{\epsilon}. \tag{4.39}$$

b) *If, in addition,* $w_1\eta_1 = \ldots = w_N\eta_N$*, then*

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \le (\textstyle\sum_{k=1}^N w_k)^{-1}\left[2w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_0\|^2 + \sum_{k=1}^N \frac{6w_kM^2}{\alpha_X\tau_k}\right] + \bar{\epsilon}, \quad (4.40)$$

$$\mathbb{E}[\|\delta\|] \le \frac{w_1\eta_1}{\sum_{k=1}^N w_k}\left[2\|y_* - y_0\| + 2\sqrt{\tfrac{\tau_1}{\eta_1}}\Omega_X + \sqrt{\tfrac{2}{w_1\eta_1}\textstyle\sum_{k=1}^N w_k\left(\frac{6M^2}{\alpha_X\tau_k} + \bar{\epsilon}\right)}\right], \quad (4.41)$$

$$\mathbb{E}[\|y_* - \bar{y}_N\|^2] \le \|y_* - y_0\|^2 + (\sum_{k=1}^N w_k)^{-1}\sum_{k=1}^N \tfrac{2}{\eta_k}\left[2w_1\tau_1\Omega_X^2 + \sum_{i=1}^k w_i(\tfrac{6M^2}{\tau_i} + \bar{\epsilon})\right], \quad (4.42)$$

where $\delta := (\sum_{k=1}^N w_k)^{-1}[w_1\eta_1(y_0 - y_N)]$.

Proof. We first prove part (a). Letting $y = y_*$ in (4.32) and using the definition of Ω_X in (4.2), we have

$$Q(\bar{z}_N; x, y_*) \le (\textstyle\sum_{k=1}^N w_k)^{-1}\left[w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_* - y_0\|^2 - \frac{w_N\eta_N}{2}\|y_* - y_N\|^2 + \sum_{k=1}^N \Lambda_k\right]. \quad (4.43)$$

Maximizing w.r.t. $x \in X$ and then taking expectation on both sides of (4.44), we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \le (\textstyle\sum_{k=1}^N w_k)^{-1}\left[w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_* - y_0\|^2 + \mathbb{E}[\sum_{k=1}^N \Lambda_k]\right]. \quad (4.44)$$

Now it follows from (4.33) and (4.36) that

$$\begin{aligned}\textstyle\sum_{k=1}^N \Lambda_k &= \textstyle\sum_{k=1}^N w_k\left(\frac{(M+\|G_{k-1}\|_*)^2}{\tau_k\alpha_X} + \bar{\epsilon} + \langle\Delta_{k-1}, x_{k-1} - x_{k-1}^v\rangle + \langle\Delta_{k-1}, x_{k-1}^v - x\rangle\right)\\ &\le \textstyle\sum_{k=1}^N w_k\left(\frac{2M^2+2\|G_{k-1}\|_*^2}{\tau_k\alpha_X} + \bar{\epsilon} + \langle\Delta_{k-1}, x_{k-1} - x_{k-1}^v\rangle\right)\\ &\quad + w_1\tau_1\Omega_X^2 + \textstyle\sum_{k=1}^N \frac{w_k\|\Delta_{k-1}\|_*^2}{2\alpha_X\tau_k}.\end{aligned}$$

Note that the random noises Δ_k are independent of x_{k-1} and $\mathbb{E}[\Delta_k] = 0$, hence $\mathbb{E}[\langle\Delta_k, x_{k-1} - x_k^v\rangle] = 0$. Moreover, using the relations that $\mathbb{E}[\|G_{k-1}\|_*^2] \le M^2$, $\|g(x_{k-1})\| \le M$ and the

triangle inequality, we have

$$\mathbb{E}[\|\Delta_{k-1}\|_*^2] = \mathbb{E}[\|G_{k-1} - g(x_{k-1})\|_*^2] \leq \mathbb{E}[(\|G_{k-1}\|_* + \|g(x_{k-1})\|_*)^2] \leq 4M^2. \quad (4.45)$$

Therefore,

$$\mathbb{E}[\textstyle\sum_{k=1}^N \Lambda_k] \leq w_1\tau_1\Omega_X^2 + \sum_{k=1}^N w_k \left(\frac{6M^2}{\alpha_X\tau_k} + \bar{\epsilon}\right). \quad (4.46)$$

The result (4.39) then follows by using the above relation in (4.44).

We now show part (b) holds. Adding $\langle \delta, y\rangle$ to both sides of (4.32) and using the fact that $w_1\eta_1 = w_N\eta_N$, we have

$$\begin{aligned} Q(\bar{z}_N, z) + \langle \delta, y\rangle &\leq (\textstyle\sum_{k=1}^N w_k)^{-1}[w_1\tau_1 P_X(x_0, x) + w_1\eta_1(\frac{1}{2}\|y_0 - y\|^2 - \frac{1}{2}\|y_N - y\|^2 \\ &\quad + \langle y_0 - y_N, y\rangle) + \textstyle\sum_{k=1}^N \Lambda_k] \\ &\leq (\textstyle\sum_{k=1}^N w_k)^{-1}[w_1\tau_1 P_X(x_0, x) + \frac{w_1\eta_1}{2}\|y_0\|^2 + \textstyle\sum_{k=1}^N \Lambda_k]. \end{aligned}$$

Maximizing both sides of the above inequality w.r.t. $(x, y) \in X \times K_*$, taking expectation and using (4.38), we obtain

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \leq (\textstyle\sum_{k=1}^N w_k)^{-1}\left[w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_0\|^2 + \mathbb{E}[\textstyle\sum_{k=1}^N \Lambda_k]\right].$$

The result in (4.40) then follows from the above inequality and (4.46). Now fixing $x = x_*$ in (4.43) and using the fact $Q(\bar{z}_N; x_*, y_*) \geq 0$, we have

$$\tfrac{w_N\eta_N}{2}\|y_* - y_N\|^2 \leq w_1\tau_1\Omega_X^2 + \tfrac{w_1\eta_1}{2}\|y_* - y_0\|^2 + \textstyle\sum_{k=1}^N \Lambda_k.$$

Taking expectation on both sides of the above inequality and using (4.46), we conclude

$$\tfrac{w_N\eta_N}{2}\mathbb{E}[\|y_* - y_N\|^2] \leq 2w_1\tau_1\Omega_X^2 + \tfrac{w_1\eta_1}{2}\|y_* - y_0\|^2 + \textstyle\sum_{k=1}^N w_k \left(\frac{6M^2}{\alpha_X\tau_k} + \bar{\epsilon}\right), \quad (4.47)$$

which implies that

$$\mathbb{E}[\|y_* - y_N\|] \leq 2\sqrt{\tfrac{\tau_1}{\eta_1}}\Omega_X + \|y_* - y_0\| + \sqrt{\tfrac{2}{w_1\eta_1}\sum_{k=1}^N w_k\left(\tfrac{6M^2}{\alpha_X\tau_k} + \bar{\epsilon}\right)}.$$

Using the above inequality and the fact that $\|\delta\| \leq (\sum_{k=1}^N w_k)^{-1}[w_1\eta_1(\|y_0 - y_*\| + \|y_* - y_N\|)$, we obtain (4.41). Observe that (4.47) holds for any y_k, $k = 1, \ldots, N$, and hence that

$$\tfrac{w_k\eta_k}{2}\mathbb{E}[\|y_* - y_k\|^2] \leq 2w_1\tau_1\Omega_X^2 + \tfrac{w_1\eta_1}{2}\|y_* - y_0\|^2 + \textstyle\sum_{i=1}^k w_i\left(\tfrac{6M^2}{\alpha_X\tau_i} + \bar{\epsilon}\right).$$

Using the above inequality, the convexity of $\|\cdot\|^2$ and the fact that $\bar{y}_N = \sum_{k=1}^N (w_k y_k)/\sum_{k=1}^N w_k$, we conclude that

$$\begin{aligned}\mathbb{E}[\|y_* - \bar{y}_N\|^2] &\leq (\textstyle\sum_{k=1}^N w_k)^{-1}\sum_{k=1}^N\left[\tfrac{4w_1\tau_1\Omega_X^2}{\eta_k} + \tfrac{w_1\eta_1}{\eta_k}\|y_* - y_0\|^2 + \tfrac{2}{\eta_k}\textstyle\sum_{i=1}^k w_i(\tfrac{6M^2}{\tau_i} + \bar{\epsilon})\right] \\ &= \|y_* - y_0\|^2 + (\textstyle\sum_{k=1}^N w_k)^{-1}\sum_{k=1}^N\left[\tfrac{4w_1\tau_1\Omega_X^2}{\eta_k} + \tfrac{2}{\eta_k}\textstyle\sum_{i=1}^k w_i(\tfrac{6M^2}{\tau_i} + \bar{\epsilon})\right],\end{aligned}$$

where the second identity follows from the fact that $w_k\eta_k = w_1\eta_1$. ∎

Below we provide two different parameter settings for $\{w_k\}, \{\tau_k\}$ and $\{\eta_k\}$ satisfying (4.31). While the first one in Corollary 30 leads to slightly better rate of convergence, the second one in Corollary 31 can guarantee the boundedness of the dual solution in expectation. We will discuss how to use these results when analyzing the convergence of the DSA algorithm.

Corollary 30 *If*

$$w_k = w = 1, \tau_k = \tau = \max\{\tfrac{M\sqrt{3N}}{\Omega_X\sqrt{\alpha_X}}, \tfrac{\sqrt{2}\|A\|}{\sqrt{\alpha_X}}\} \text{ and } \eta_k = \eta = \tfrac{\sqrt{2}\|A\|}{\sqrt{\alpha_X}}, \forall 1 \leq k \leq N, \quad (4.48)$$

then

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \le \tfrac{\sqrt{2}\|A\|(2\Omega_X^2+\|y_*-y_0\|^2)}{\sqrt{\alpha_X}N} + \tfrac{4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X}N} + \bar{\epsilon}, \tag{4.49}$$

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \le \tfrac{\sqrt{2}\|A\|(2\Omega_X^2+\|y_0\|^2)}{\sqrt{\alpha_X}N} + \tfrac{4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X}N} + \bar{\epsilon}, \tag{4.50}$$

$$\mathbb{E}[\|\delta\|] \le \tfrac{2\sqrt{2\alpha_X}\|A\|\|y_*-y_0\|+4\Omega_X\|A\|}{\alpha_X N} + \tfrac{2M(\sqrt{6}\|A\|+\sqrt{3\alpha_X})}{\alpha_X\sqrt{N}} + \sqrt{\tfrac{3\|A\|\bar{\epsilon}}{N\sqrt{\alpha_X}}}, \tag{4.51}$$

$$\mathbb{E}[\|y_*-\bar{y}_N\|^2] \le \|y_*-y_0\|^2 + 4\Omega_X^2 + \tfrac{2\sqrt{6N}M\Omega_X}{\|A\|} + \tfrac{3\alpha_X(N+1)M^2}{\|A\|^2} + \tfrac{(N+1)\bar{\epsilon}}{2}. \tag{4.52}$$

Proof. We can easily check that the parameter setting in (4.48) satisfies (4.31). It follows from (4.39) and (4.48) that

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \le \tfrac{1}{N}\left[2\tau\Omega_X^2 + \tfrac{\eta}{2}\|y_*-y_0\|^2 + \tfrac{6NM^2}{\alpha_X\tau}\right] + \bar{\epsilon} \le \tfrac{\sqrt{2}\|A\|(2\Omega_X^2+\|y_*-y_0\|^2)}{\sqrt{\alpha_X}N} + \tfrac{4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X}N} + \bar{\epsilon}.$$

Moreover, we have $w_1\eta_1 = w_N\eta_N$. Hence, by (4.40) and (4.48),

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \le \tfrac{1}{N}\left[2\tau\Omega_X^2 + \tfrac{\eta}{2}\|y_0\|^2 + \tfrac{6NM^2}{\alpha_X\tau}\right] + \bar{\epsilon} \le \tfrac{\sqrt{2}\|A\|(2\Omega_X^2+\|y_0\|^2)}{\sqrt{\alpha_X}N} + \tfrac{4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X}N} + \bar{\epsilon}.$$

Also by (4.41) and (4.48),

$$\begin{aligned}\mathbb{E}[\|\delta\|] &\le \tfrac{\eta}{N}\left[2\|y_*-y_0\| + 2\sqrt{\tfrac{\tau}{\eta}}\Omega_X + \sqrt{\tfrac{2N}{\eta}\left(\tfrac{6M^2}{\alpha_X\tau}+\bar{\epsilon}\right)}\right] \\ &\le \tfrac{2\sqrt{2}\|A\|\|y_*-y_0\|}{N\sqrt{\alpha_X}} + \tfrac{2\Omega_X}{N}\left(\tfrac{2\|A\|}{\alpha_X} + \tfrac{\sqrt{6N}\|A\|M}{\Omega_X\alpha_X}\right) + \tfrac{2\sqrt{M}}{\sqrt{\alpha_X N}} + \tfrac{\sqrt{2\bar{\epsilon}}}{\sqrt{N}}\sqrt{\tfrac{\sqrt{2}\|A\|}{\sqrt{\alpha_X}}},\end{aligned}$$

which implies (4.51). Finally, by (4.41) and (4.48),

$$\begin{aligned}\mathbb{E}[\|y_*-\bar{y}_N\|^2] &\le \|y_*-y_0\|^2 + \tfrac{1}{N}\left[\textstyle\sum_{k=1}^N \tfrac{4\tau_k}{\eta_k}\Omega_X^2 + \sum_{k=1}^N \tfrac{2}{\eta_k}\sum_{i=1}^k\left(\tfrac{6M^2}{\tau_i}+\bar{\epsilon}\right)\right] \\ &\le \|y_*-y_0\|^2 + 4\Omega_X^2 + \tfrac{2\sqrt{6N}M\Omega_X}{\|A\|} + \tfrac{3\alpha_X(N+1)M^2}{\|A\|^2} + \tfrac{(N+1)\bar{\epsilon}}{2}.\end{aligned}$$

■

In view of (4.52), if $M > 0$ or N is not properly chosen, we cannot guarantee that $\mathbb{E}[\|y_* - \bar{y}_N\|^2]$ is bounded. In the following corollary, we will modify the selection of τ and η in (4.48) in order to guarantee the boundedness of $\mathbb{E}[\|y_* - \bar{y}_N\|^2]$ even when $M > 0$.

Corollary 31 *If*

$$w_k = w = 1, \tau_k = \tau = \max\{\tfrac{M\sqrt{3N}}{\Omega_X\sqrt{\alpha_X}}, \tfrac{\sqrt{2}\|A\|}{\sqrt{\alpha_X N}}\} \text{ and } \eta_k = \eta = \tfrac{\sqrt{2N}\|A\|}{\sqrt{\alpha_X}}, \forall 1 \leq k \leq N, \tag{4.53}$$

then

$$\mathbb{E}[\text{gap}_*(\bar{z}_N)] \leq \tfrac{2\sqrt{2}\|A\|\Omega_X^2}{N\sqrt{\alpha_X N}} + \tfrac{\|A\|\|y_* - y_0\|^2 + 4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X N}} + \bar{\epsilon}, \tag{4.54}$$

$$\mathbb{E}[\text{gap}_\delta(\bar{z}_N)] \leq \tfrac{2\sqrt{2}\|A\|\Omega_X^2}{N\sqrt{\alpha_X N}} + \tfrac{\|A\|\|y_0\|^2 + 4\sqrt{3}M\Omega_X}{\sqrt{\alpha_X N}} + \bar{\epsilon}, \tag{4.55}$$

$$\mathbb{E}[\|\delta\|] \leq \tfrac{2\sqrt{2}\|A\|\|y_* - y_0\| + 4\sqrt{M\|A\|\Omega_X}}{\sqrt{\alpha_X N}} + \tfrac{2\sqrt{6}\|A\|M}{\alpha_X} + \tfrac{4\Omega_X^2\|A\|^2}{N\alpha_X} + \sqrt{\tfrac{3\|A\|\bar{\epsilon}}{\sqrt{\alpha_X N}}}, \tag{4.56}$$

$$\mathbb{E}[\|y_* - \bar{y}_N\|^2] \leq \|y_* - y_0\|^2 + \tfrac{2\Omega_X^2}{N} + \tfrac{\sqrt{6}(1+\alpha_X)M\Omega_X}{\|A\|} + \tfrac{\sqrt{\alpha_X N}\bar{\epsilon}}{\sqrt{2}\|A\|}. \tag{4.57}$$

Proof. The proofs of (4.54)-(4.57) are similar to Corollary 30 and hence the details are skipped. ■

Note that by using the parameter setting (4.53), we still obtain the optimal rate of convergence in terms of the dependence on N, with a slightly worse dependence on $\|A\|$ and $\|y_*\|$ than the one obtained by using the parameter setting in (4.48). However, using the setting (4.53), we can bound $\mathbb{E}[\|\bar{y}_N - y_*\|^2]$ as long as $N = \mathcal{O}(1/\bar{\epsilon}^2)$, while this statement does not necessarily hold for the parameter setting in (4.48).

We now state one technical result regarding the functional optimality gap and primal infeasibility, which generalizes Proposition 2.1 of [99] to conic programming.

Lemma 32 *If there exist random vectors $\delta \in \mathbb{R}^m$ and $\bar{z} \equiv (\bar{x}, \bar{y}) \in Z$ such that*

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z})] \leq \epsilon_o, \tag{4.58}$$

then

$$\mathbb{E}[h(\bar{x}, c) + \tilde{v}(\bar{x}) - (h(x^*, c) + \tilde{v}(x^*))] \leq \epsilon_0,$$
$$A\bar{x} - Bu - b - \delta \in K \text{ a.s.},$$

where x^* is an optimal solution of problem (4.13).

Proof. Letting $x = x^*$ and $y = 0$ in the definition of (4.38), we can easily see that

$$h(\bar{x}, c) + \tilde{v}(\bar{x}) - (h(x^*, c) + \tilde{v}(x^*)) \leq \mathrm{gap}_\delta(\bar{z}).$$

Moreover, in view of (4.16) and (4.38), we must have $A\bar{x} - Bu - b - \delta \in K$ almost surely. Otherwise, $\mathbb{E}[\mathrm{gap}_\delta(\bar{z})]$ would be unbounded as y runs throughout K^* in the definition of $\mathrm{gap}_\delta(\bar{z})$. ■

In the next result, we will provide a bound on the optimal dual variable y_*. By doing so, we show that the complexity of Algorithm 2 only depends on the parameters for the primal problem along with the smallest nonzero eigenvalue of A and the initial point y_0, even though the algorithm is a primal-dual type method.

Lemma 33 *Let (x^*, y^*) be an optimal solution to problem* (4.13)*. If the subgradients of the objective function $v_h(x) := h(x, c) + \tilde{v}(\cdot)$ are bounded, i.e., $\|v_h'(x)\|_2 \leq M_h$ for any $x \in X$, then there exists y^* s.t.*

$$\|y^*\| \leq \tfrac{M_h}{\sigma_{min}(A)}, \tag{4.59}$$

where $\sigma_{min}(A)$ denotes the smallest nonzero singular value of A.

Proof. We consider two cases. Case 1: $A^T y^* = 0$, i.e., y_* belongs to the null space of A. Since for any $\lambda \geq 0$, λy^* is still an optimal dual solution to problem (4.13), we have

(4.59) holds.

Case 2: $A^T y^* \neq 0$. By the definition of the saddle point, we have

$$\langle b + Bu - Ax^*, y^* \rangle + h(x^*, c) + \tilde{v}(x^*) \leq \langle b + Bu - Ax, y^* \rangle + h(x, c) + \tilde{v}(x),\ \forall x \in X,$$

which implies

$$h(x^*, c) + \tilde{v}(x^*) + \langle A^T y^*, x - x^* \rangle \leq h(x, c) + \tilde{v}(x),\ \forall x \in X. \tag{4.60}$$

Hence $A^T y^*$ is a subgradient of v_h at the point x^*. Without loss of generality, we assume that y^* belongs to the column space of A^T (i.e., y^* is perpendicular to the eigenspace associated with eigenvalue 0). Otherwise we can show that the projection of y^* onto the column space of A^T will also satisfy (4.60). Using this observation, we have

$$\|A^T y^*\|_2^2 = (y^*)^T A A^T y^* = (y^*)^T U^T \Lambda U y^* \geq \sigma_{min}(AA^T)\|Uy^*\|^2 = \sigma_{min}^2(A)\|y^*\|^2,$$

where U is an orthonormal matrix whose rows consist of the eigenvectors of AA^T and Λ is the diagonal matrix whose elements are the corresponding eigenvalues. Our result then follows from the above inequality and the assumption that $\|A^T y^*\|_2 \leq M_h$. ■

4.2.4 Convergence analysis for DSA

Our goal in this subsection is to establish the complexity of the DSA algorithm for solving problem 4.9.

The basic idea is to apply the results we obtained in the previous section regarding the I-PDSA algorithm to the three loops stated in the DSA algorithm. More specifically, we will show how to generate stochastic ϵ-subgradients for the value functions v^2 and v^3 in the middle and innermost loops, respectively, and how to compute a nearly optimal solution for problem 4.9 in the outer loop of the DSA algorithm .

In order to apply these results to the saddle-point reformulation for the second and first stage problems (see (4.11) and (4.14)), we need to make sure that the condition in (4.25) holds for the value functions, v^3 and v^2 respectively, associated with the optimization problems in their subsequent stages. For this purpose, we assume that the less aggressive algorithmic parameter setting in (4.53) is applied to solve the second stage saddle point problems in (4.11), while a more aggressive parameter setting in (4.48) is used to solve the first stage and last stage saddle point problems in (4.14) and (4.12), respectively. Moreover, we need the boundedness of the operators B^2 and B^3:

$$\|B^2\| \leq \mathcal{B}_2 \text{ and } \|B^3\| \leq \mathcal{B}_3 \tag{4.61}$$

in order to guarantee that the generated stochastic subgradients for the value functions v^2 and v^3 have bounded variance.

For notational convenience, we use $\Omega_i \equiv \Omega_{X^i}$ and $\alpha_i \equiv \alpha_{X^i}$, $i = 1, 2, 3$, to denote the diameter and strongly convex modulus associated with the distance generating function for the feasible set X^i (see (4.2)). Lemma 34 shows some convergence properties for the innermost loop of the DSA algorithm.

Lemma 34 *If the parameters $\{w_k^3\}$, $\{\tau_k^3\}$ and $\{\eta_k^3\}$ are set to* (4.48) *(with $M = 0$ and $A = A_j^3$) and*

$$N_3 \equiv N_{3,j} := \frac{3\sqrt{2}\|A_j^3\|[2(\Omega_3)^2 + \|y_{*,j}^3 - y_0^3\|^2]}{\sqrt{\alpha_3}\epsilon}, \tag{4.62}$$

then $B_j^3 \bar{y}_j^3$ is a stochastic $(\epsilon/3)$-subgradient of the value function v^3 at x_{j-1}^2. Moreover, given random variable $\xi^{[2]}$, there exists a constant M_3 such that $\|v^3(x_1, \xi^{[2]}) - v^3(x_2, \xi^{[2]})\| \leq M_3\|x_1 - x_2\|, \forall x_1, x_2 \in X^2$ and

$$\mathbb{E}[\|B_j^3 \bar{y}_j^3\|_*^2 | \xi^{[2]}] \leq M_3^2. \tag{4.63}$$

In addition, there exists a vector $\delta \in \mathbb{R}^{m^3}$ s.t.

$$\begin{aligned} &\mathbb{E}[h^3(\bar{x}^3, c^3) - V^3(\bar{x}^2, \xi^{[3]})|\xi^{[2]}] \leq \epsilon/3, \\ &A^3\bar{x}^3 - B^3\bar{x}^2 - b^3 - \delta \in K^3 \text{ a.s.}, \\ &\mathbb{E}[\|\delta\||\xi^{[2]}] \leq \epsilon/3. \end{aligned} \tag{4.64}$$

Proof. The innermost loop of the DSA algorithm is equivalent to the application of Algorithm 2 to the last stage saddle point problem in (4.12). Note that for this problem, we do not have any subsequent stages and hence $\tilde{v} = 0$. In other words, the subgradients of $\tilde{v}$ are exact. In view of Corollary 30 (with $M = 0$ and $\bar{\epsilon} = 0$), the definition of N_3 in (4.62) and conditional on $\xi^{[2]}$, we have

$$\mathbb{E}[\text{gap}_*(\bar{z}_j^3)|\xi^{[2]}] \leq \frac{\sqrt{2}\|A_j^3\|[2(\Omega_3)^2 + \|y_*^3 - y_0^3\|^2]}{\sqrt{\alpha_3}N_3} \leq \frac{\epsilon}{3}.$$

This observation, in view of Lemma 25, then implies that $B_j^3\bar{y}_j^3$ is a stochastic $(\epsilon/3)$-subgradient of v^3 at x_{j-1}^2. By the Lipschitz continuity of v^3, the Lipschitz constant M_3 should satisfy

$$M_3 \geq \mathbb{E}[\|B_j^3 y_{*,j}^3\||\xi^{[2]}], \ \forall y_{*,j}^3 \in Y_*^3, \tag{4.65}$$

where Y_*^3 denotes the set of optimal dual solutions of problem (4.12). Moreover, it follows from (4.52) (with $M = 0$ and $\bar{\epsilon} = 0$) that

$$\begin{aligned} &\mathbb{E}[\|y_{*,j}^3 - \bar{y}_j^3\|^2|\xi^{[2]}] \leq \mathbb{E}[\|y_{*,j}^3 - y_0^3\|^2|\xi^{[2]}] + 4(\Omega_3)^2, \\ &\mathbb{E}[\|\bar{y}_j^3\|^2|\xi^{[2]}] \leq 2\mathbb{E}[\|y_{*,j}^3\| + \|y_{*,j}^3 - y_0^3\|^2|\xi^{[2]}] + 8\Omega_3^2. \end{aligned}$$

This inequality, in view of (4.61), implies that

$$\mathbb{E}[\|B_j^3\bar{y}_j^3\|_*^2|\xi^{[2]}] \leq \mathcal{B}_3^2\mathbb{E}[(2\|y_{*,j}^3\| + 2\|y_{*,j}^3 - y_0^3\|^2 + 8\Omega_3^2)|\xi^{[2]}]. \tag{4.66}$$

Hence, combining (4.63), (4.65) and (4.66), we can see that the latter part of our result

holds with

$$M_3 = \max\left\{\max_{y\in Y_*^3}\mathbb{E}[\|B_j^3 y\||\xi^{[2]}], \mathcal{B}_3\sqrt{\mathbb{E}[(2\|y_{*,j}^3\| + 2\|y_{*,j}^3 - y_0^3\|^2 + 8\Omega_3^2)|\xi^{[2]}]}\right\}.$$

The results in (4.64) directly follow from Lemma 32. In view of Corollary 30 (with $M = 0$ and $\bar{\epsilon} = 0$) and the definition of N_3 in (4.62), we conclude that there exist $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\mathbb{E}_{\xi^2}[\|\delta\|] \leq \frac{2\sqrt{2\alpha_3}\|A^3\|\|y_*^3 - y_0^3\| + 4\Omega_3\|A^3\|}{\alpha_3 N_3} \leq \epsilon/3,$$

which together with Lemma 32 then imply our result. ∎

Lemma 35 describes some convergence properties for the middle loop of the DSA algorithm.

Lemma 35 *Assume that the parameters for the innermost loop are set according to Lemma 34. If the parameters $\{w_j^2\}$, $\{\tau_j^2\}$ and $\{\eta_j^2\}$ for the middle loop are set to* (4.53) *(with $M = M_3$ and $A = A_i^2$) and*

$$N_2 \equiv N_{2,i} := \left(\frac{12\sqrt{2}\|A_i^2\|\Omega_2}{\sqrt{\alpha_2}\epsilon}\right)^{\frac{2}{3}} + \left[\frac{6\left(\|A_i^2\|\|y_{*,i}^2 - y_0^2\|^2 + 4\sqrt{3}M_3\Omega_2\right)}{\sqrt{\alpha_2}\epsilon}\right]^2, \tag{4.67}$$

then $B_i^2\bar{y}_i^2$ is a stochastic $(2\epsilon/3)$-subgradient of the value function v^2 at x_{i-1}^1. Moreover, there exists a constant M_2 such that $\|v^2(x_1) - v^2(x_2)\| \leq M_2\|x_1 - x_2\|, \forall x_1, x_2 \in X^2$ and

$$\mathbb{E}[\|B_i^2\bar{y}_i^2\|_*^2|\xi^{[1]}] \leq M_2^2, \tag{4.68}$$

In addition, there exists a vector $\delta \in \mathbb{R}^{m^2}$ s.t.

$$\begin{aligned}
&\mathbb{E}[h^2(\bar{x}^2, c^2) + v^3(\bar{x}^2|\xi^2) - V^2(\bar{x}^1, \xi^{[2]})] \leq 2\epsilon/3,\\
&A^2\bar{x}^2 - B^2\bar{x}^1 - b^2 - \delta \in K^2 \text{ a.s.},\\
&\mathbb{E}[\|\delta\||\xi^{[2]}] \leq 2\epsilon/3.
\end{aligned}$$

Proof. The middle loop of the DSA algorithm is equivalent to the application of Algorithm 2 to the second stage saddle point problem in (4.11). Note that for this problem, we have $\tilde{v} = v^3$. Moreover, by Lemma 34, the stochastic subgradients of v^3 are computed by the innermost loop with tolerance $\bar{\epsilon} = \epsilon/3$. In view of Corollary 31 (with $M = M_3$ and $\bar{\epsilon} = \epsilon/3$) and the definition of N_2 in (4.67), we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_i^2)|\xi^{[1]}] \leq \tfrac{2\sqrt{2}\|A_i^2\|\Omega_2}{N_2\sqrt{\alpha_2 N_2}} + \tfrac{\|A_i^2\|\|y_{*,i}^2 - y_0^2\|^2 + 4\sqrt{3}M_3\Omega_2}{\sqrt{\alpha_2 N_2}} + \bar{\epsilon} \leq \tfrac{2\epsilon}{3}.$$

This observation, in view of Lemma 25, then implies that $B_i^2\bar{y}_i^2$ is a stochastic $(2\epsilon/3)$-subgradient v^2 at x_{i-1}^1. By the Lipschitz continuity of v^2, the Lipschitz constant M_2 should satisfy

$$M_2 \geq \mathbb{E}[\|B_i^2 y_{*,i}^2\||\xi^{[1]}],\ \forall y_{*,i}^2 \in Y_*^2, \tag{4.69}$$

where Y_*^2 denotes the set of optimal dual solutions of problem (4.11). Moreover, it follows from (4.57) (with $M = M_3$ and $\bar{\epsilon} = \epsilon/3$) that

$$\begin{aligned}
&\mathbb{E}[\|y_{*,i}^2 - \bar{y}_i^2\|^2|\xi^{[1]}] \leq \mathbb{E}[\|y_{*,i}^2 - y_0^2\|^2 + \tfrac{2\Omega_2^2}{N_2} + \tfrac{\sqrt{6}(1+\alpha_2)M_3\Omega_2}{\|A_i^2\|} + \tfrac{\sqrt{\alpha_2 N_2}\epsilon}{3\sqrt{2}\|A_i^2\|}|\xi^{[1]}]], \\
&\mathbb{E}[\|\bar{y}_i^2\|^2|\xi^{[1]}] \leq \mathbb{E}[2\|y_{*,i}^2\|^2 + 2\|y_{*,i}^2 - y_0^2\|^2 + \tfrac{4\Omega_2^2}{N_2} + \tfrac{2\sqrt{6}(1+\alpha_2)M_3\Omega_2}{\|A_i^2\|} + \tfrac{\sqrt{2\alpha_2 N_2}\epsilon}{3\|A_i^2\|}|\xi^{[1]}].
\end{aligned}$$

This inequality, in view of (4.61), implies that

$$\mathbb{E}[\|B_i^2\bar{y}_i^2\|_*^2|\xi^{[1]}] \leq \mathcal{B}_2^2\mathbb{E}\left[2\|y_{*,i}^2\|^2 + 2\|y_{*,i}^2 - y_0^2\|^2 + \tfrac{4\Omega_2^2}{N_2} + \tfrac{2\sqrt{6}(1+\alpha_2)M_3\Omega_2}{\|A_i^2\|} + \tfrac{\sqrt{2\alpha_2 N_2}\epsilon}{3\|A_i^2\|}|\xi^{[1]}\right], \tag{4.70}$$

where N_2 is defined in (4.67). Hence, combining these observations, we can see that the latter part of our results holds with M_2 satisfying both (4.69) and

$$M_2 \geq \mathcal{B}_2\left\{\mathbb{E}\left[2\|y_{*,i}^2\|^2 + 2\|y_{*,i}^2 - y_0^2\|^2 + \tfrac{4\Omega_2^2}{N_2} + \tfrac{2\sqrt{6}(1+\alpha_2)M_3\Omega_2}{\|A_i^2\|} + \tfrac{\sqrt{2\alpha_2 N_2}\epsilon}{3\|A_i^2\|}|\xi^{[1]}\right]\right\}^{\frac{1}{2}}.$$

In view of Corollary 30 (with $M = M_3$ and $\bar{\epsilon} = \epsilon/3$) and the definition of N_2 in (4.67), we

conclude that there exist $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\mathbb{E}_{\xi^2}[\|\delta\|] \leq \frac{2\sqrt{2\alpha_2}\|A^2\|\|y_*^2 - y_0^2\| + 4\Omega_2\|A^2\|}{\alpha_2 N_2} + \frac{2M_2(\sqrt{6}\|A^2\| + \sqrt{3\alpha_2})}{\alpha_2\sqrt{N_2}} + \sqrt{\frac{2\|A^2\|\epsilon}{N_2\sqrt{\alpha_2}}} \leq 2\epsilon/3,$$

which together with Lemma 32 then imply our result. ∎

We are now ready to establish the main convergence properties of the DSA algorithm applied to a three-stage stochastic optimization problem.

Theorem 36 *Suppose that the parameters for the innermost and middle loop in the DSA algorithm are set according to Lemma 34 and Lemma 35, respectively. If the parameters* $\{w_i\}$, $\{\tau_i\}$ *and* $\{\eta_i\}$ *for the outer loop are set to (4.48) (with* $M = M_2$ *and* $A = A^1$*) and*

$$\begin{aligned} N_1 := \max\Bigg\{ & \frac{6\sqrt{2}\|A^1\|[2(\Omega_1)^2 + \|y_0^1\|^2]}{\sqrt{\alpha_1}\epsilon} + \left(\frac{24\sqrt{3}M_2\Omega_1}{\sqrt{\alpha_1}\epsilon}\right)^2, \\ & \frac{6\|A^1\|(\sqrt{2\alpha_1}\|y_*^1 - y_0^1\| + 2\Omega_1 + 3\sqrt{\alpha_1})}{\alpha_1\epsilon} + \left(\frac{6\sqrt{3}M_2(\sqrt{2}\|A^1\| + \sqrt{\alpha_1})}{\alpha_1\epsilon}\right)^2 \Bigg\}, \end{aligned} \tag{4.71}$$

then we will find a solution $\bar{x}^1 \in X^1$ *and a vector* $\delta \in \mathbb{R}^{m^1}$ *s.t.*

$$\begin{aligned} & \mathbb{E}[h(\bar{x}^1, c) + v^2(\bar{x}^1, \xi^1) - (h(x^*, c) + v^2(x^*, \xi^1))] \leq \epsilon, \\ & A\bar{x}^1 - b - \delta \in K^1, a.s., \\ & \mathbb{E}[\|\delta\|] \leq \epsilon, \end{aligned}$$

where x^* *denotes the optimal solution of problem 4.9.*

Proof. The outer loop of the DSA algorithm is equivalent to the application of Algorithm 2 to the first stage saddle point problem in (4.14). Note that for this problem, we have $\tilde{v} = v^2$. Moreover, by Lemma 35, the stochastic subgradients of v^2 are computed by the middle loop with tolerance $\bar{\epsilon} = 2\epsilon/3$. In view of Corollary 30 (with $M = M_2$ and

$\bar{\epsilon} = 2\epsilon/3$) and the definition of N_1 in (4.71), we conclude that there exist $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\begin{aligned}
\mathbb{E}_{\xi^2}[\mathrm{gap}_\delta(\bar{z}_N^1)] &\leq \tfrac{\sqrt{2}\|A^1\|(2\Omega_1^2+\|y_0^1\|^2)}{\sqrt{\alpha_1}N_1} + \tfrac{4\sqrt{3}M_2\Omega_1}{\sqrt{\alpha_1}N_1} + \tfrac{2\epsilon}{3} \leq \epsilon, \\
\mathbb{E}_{\xi^2}[\|\delta\|] &\leq \tfrac{2\sqrt{2\alpha_1}\|A^1\|\|y_*^1-y_0^1\|+4\Omega_1\|A^1\|}{\alpha_1 N_1} + \tfrac{2M_2(\sqrt{6}\|A^1\|+\sqrt{3\alpha_1})}{\alpha_1\sqrt{N_1}} + \sqrt{\tfrac{2\|A^1\|\epsilon}{N_1\sqrt{\alpha_1}}} \leq \epsilon,
\end{aligned}$$

which together with Lemma 32 then imply our result. ∎

We now add a few remarks about the convergence of the DSA algorithm. Firstly, in view of (4.67) and (4.62), N_2 and N_3 are random variables since they depend on the random variables $\xi^{[2]}$ and $\xi^{[3]}$, respectively. The selection of N_2 and N_3 allows us to remove the boundedness assumptions for a few random variables such as A_i^2 and A_j^3. Secondly, if the random variables appearing in the definition of N_2, i.e., A_i^2 and $y_{*,i}$, are bounded, we can see from Lemma 35 and Theorem 36 that the number of random samples ξ^2 and ξ^3 are given by

$$N_1 = \mathcal{O}(1/\epsilon^2) \;\text{ and }\; N_1 \times N_2 = \mathcal{O}(1/\epsilon^4), \tag{4.72}$$

respectively. It is also possible to obtain an upper bound for N_2 and $N_1 \times N_2$ in expectation with respect to ξ^2 without assuming the boundedness of A_i^2 and $y_{*,i}$. Thirdly, it appears that the convergence of the DSA algorithm relies on y_*^1, $y_{*,i}^2$, and $y_{*,j}^3$. However, the size of these dual variable can be estimated by using Lemma 33. and possibly some tools from random matrix theory [100] to estimate the smallest singular values in case these quantities are not easily computable.

It should be noted that our analysis of DSA focuses on the optimality of the first-stage decisions, and the decisions we generated for the later stages are mainly used for computing the approximate stochastic subgradients for the values functions at each stage. Except for the first stage decision $\bar{x}^1$, the performance guarantees (e.g., feasibility and optimality) that we can provide for later stages (see Lemma 34 and 35) are dependent on the sequences of random variables (or scenarios) we generated. We do not generate history-dependent policy or suggest a prefixed sequence of decisions for general multi-stage stochastic optimization

problems. However, in some cases such prefixed sequence can still be extracted from the output of the algorithm. In particular, if one can separate the state and control variables, then we can use the obtained solutions for the initial state variable and the ones for the control variables in later stages as a prefixed control policy (see Section 4.5 for an example in portfolio optimization). In general, one possible way to guarantee the feasibility and optimality of the decisions in the later stages would be to re-run the DSA algorithm in each stage. More specifically, at the beginning of each stage, we already know the realization of the random variable at this stage and the decisions from the previous stage, we can run the DSA algorithm now for a smaller multi-stage stochastic optimization problem, i.e., the number of stages will decrease by 1 every time we run the algorithm. One can see that the computational cost for these subsequent runs of the DSA algorithm will decrease exponentially with respect to the remaining number of stages. Therefore, the total amount of computational cost over all these subsequent runs will be in the same order of magnitude as that for the first run of the DSA method.

4.3 Three-stage problems with strongly convex objectives

In this section, we show that the complexity of the DSA algorithm can be significantly improved if the objective functions h^i, $i = 1, 2, 3$, are strongly convex. We will first refine the convergence properties of Algorithm 2 under the strong convexity assumption about $h(x, c)$ and then use these results to improve the complexity results of the DSA algorithm.

4.3.1 Basic tools: inexact primal-dual stochastic approximation under strong convexity

Our goal in this subsection is to study the convergence properties of Algorithm 2 applied to problem (4.13) under the assumption that $h(x, c)$ is strongly convex, i.e., $\exists \mu_h > 0$ s.t.

$$h(x_1, c) - h(x_2, c) - \langle h'(x_2, c), x_1 - x_2 \rangle \geq \mu_h P_X(x_2, x_1), \ \forall x_1, x_2 \in X. \tag{4.73}$$

Proposition 37 below shows the relation between (x_{k-1}, y_{k-1}) and (x_k, y_k) after running one step of SPDT when the assumption about h in (4.73) is satisfied.

Proposition 37 *Let Q and Δ_k be defined in (4.16) and (4.27), respectively. For any $1 \leq k \leq N$ and $(x, y) \in X \times K_*$, we have*

$$\begin{aligned} &Q(z_k, z) + \langle A(x_k - x), y_k - y_{k-1}\rangle - \theta_k \langle A(x_{k-1} - x), y_{k-1} - y_{k-2}\rangle \\ &\leq \tau_k P_X(x_{k-1}, x) - (\tau_k + \mu_h) P_X(x_k, x) + \tfrac{\eta_k}{2}[\|y_{k-1} - y\|^2 - \|y_k - y\|^2] \\ &\quad - \tfrac{\alpha_X \tau_k}{2}\|x_k - x_{k-1}\|^2 - \tfrac{\eta_k}{2}\|y_{k-1} - y_k\|^2 + \bar{\epsilon} + (M + \|G_{k-1}\|_*)\|x_k - x_{k-1}\| \\ &\quad + \theta_k \langle A(x_k - x_{k-1}), y_{k-1} - y_{k-2}\rangle + \langle \Delta_{k-1}, x_{k-1} - x\rangle, \end{aligned} \tag{4.74}$$

Proof. Since h is strongly convex, we can rewrite (4.29) as

$$\begin{aligned} \langle -A_k(x_k - x), \tilde{y}_k\rangle + h(x_k, c_k) - h(x, c_k) + \langle G(x_{k-1}, \xi_k), x_k - x\rangle \\ \leq \tau_k P_X(x_{k-1}, x) - (\tau_k + \mu_h) P_X(x_k, x) - \tau_k P_X(x_{k-1}, x_k). \end{aligned}$$

It then follows from (4.16), (4.28), (4.30) and the above inequality that

$$\begin{aligned} &Q(z_k, z) + \langle A(x_k - x), y_k - \tilde{y}_k\rangle \leq \tau_k P_X(x_{k-1}, x) - (\tau_k + \mu_h) P_X(x_k, x) \\ &- \tau_k P_X(x_{k-1}, x_k) + \tfrac{\eta_k}{2}[\|y_{k-1} - y\|^2 - \|y_k - y\|^2 - \|y_{k-1} - y_k\|^2] \\ &+ (M + \|G_{k-1}\|_*)\|x_k - x_{k-1}\| + \langle \Delta_{k-1}, x_{k-1} - x\rangle + \bar{\epsilon}. \end{aligned}$$

Similarly to the proof of (26), using the above relation, the definition of $\tilde{y}_k$ in (4.19) and the strong convexity of P in (4.1), we have (4.74). ∎

With the help of Proposition 37, we can provide bounds of two gap functions $\mathrm{gap}_*(\bar{z}_N)$ and $\mathrm{gap}_*\delta(\bar{z}_N)$ under the strong convexity assumption of h.

Theorem 38 *Suppose that the parameters $\{\theta_k\}$, $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$ satisfy (4.31) with*

(4.31).b) replaced by

$$w_k(\mu_h + \tau_k) \geq w_{k+1}\tau_{k+1},\ k = 1, \ldots, N-1. \tag{4.75}$$

a) *For $N \geq 1$, we have*

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \leq (\textstyle\sum_{k=1}^N w_k)^{-1}[2w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_0 - y_*\|^2 + \sum_{k=1}^N \frac{6M^2w_k}{\alpha_X\tau_k}] + \bar{\epsilon}. \tag{4.76}$$

b) *If, in addition, $w_1\eta_1 = \ldots = w_N\eta_N$, then*

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \leq (\textstyle\sum_{k=1}^N w_k)^{-1}[2w_1\tau_1\Omega_X^2 + \frac{w_1\eta_1}{2}\|y_0\|^2 + \sum_{k=1}^N \frac{6M^2w_k}{2\alpha_X\tau_k}] + \bar{\epsilon}, \tag{4.77}$$

$$\mathbb{E}[\|\delta\|] \leq \frac{w_1\eta_1}{\sum_{k=1}^N w_k}\left[2\|y_* - y_0\| + 2\sqrt{\tfrac{\tau_1}{\eta_1}}\Omega_X + \sqrt{\tfrac{2}{w_1\eta_1}\textstyle\sum_{k=1}^N w_k\left(\frac{6M^2}{\alpha_X\tau_k} + \bar{\epsilon}\right)}\right], \tag{4.78}$$

$$\mathbb{E}[\|y_* - \bar{y}_N\|^2] \leq \|y_* - y_0\|^2 + (\textstyle\sum_{k=1}^N w_k)^{-1}\sum_{k=1}^N \frac{2}{\eta_k}\left[2w_1\tau_1\Omega_X^2 + \sum_{i=1}^k w_i(\frac{6M^2}{\tau_i} + \bar{\epsilon})\right],$$

where $\delta = (\sum_{k=1}^N w_k)^{-1}[w_1\eta_1(y_0 - y_N)]$.

Proof. We first show part a) holds. Multiplying both sides of (4.74) by w_k for every $k \geq 1$, summing up the resulting inequalities over $1 \leq k \leq N$, and using the relations in

(4.31) and (4.75), we have

$$
\begin{aligned}
&\textstyle\sum_{k=1}^{N} w_k Q(z_k, z) \\
&\leq \textstyle\sum_{k=1}^{N} [w_k \tau_k P_X(x_{k-1}, x) - w_k(\tau_k + \mu_h) P_X(x_k, x)] - \sum_{k=1}^{N} \frac{\alpha_X w_k \tau_k}{2} \|x_k - x_{k-1}\|^2 \\
&\quad + \textstyle\sum_{k=1}^{N} [\frac{w_k \eta_k}{2} \|y_{k-1} - y\|^2 - \frac{w_k \eta_k}{2} \|y_k - y\|^2] - \sum_{k=1}^{N} \frac{w_k \eta_k}{2} \|y_{k-1} - y_k\|^2 \\
&\quad + \textstyle\sum_{k=1}^{N} w_{k-1} \langle A(x_{k-1} - x_k), y_{k-1} - y_{k-2} \rangle + \sum_{k=1}^{N} w_k \bar{\epsilon} + w_N \langle A(x - x_N), y_N - y_{N-1} \rangle \\
&\quad + \textstyle\sum_{k=1}^{N} w_k (M + \|G_{k-1}\|_*) \|x_k - x_{k-1}\| + \sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1} - x \rangle \\
&\leq w_1 \tau_1 P_X(x_0, x) + \tfrac{w_1 \eta_1}{2} \|y_0 - y\|^2 - \tfrac{w_N \eta_N}{2} \|y_N - y\|^2 \\
&\quad + \textstyle\sum_{k=1}^{N} w_k \bar{\epsilon} + \sum_{k=1}^{N} \frac{(M + \|G_{k-1}\|_*)^2 w_k}{\alpha_X \tau_k} + \sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1} - x \rangle \\
&\quad - w_N(\tau_N + \mu_h) P_X(x_N, x) + w_N \langle A(x - x_N), y_N - y_{N-1} \rangle - \tfrac{w_N \eta_N}{2} \|y_N - y_{N-1}\|^2 \\
&\leq w_1 \tau_1 P_X(x_0, x) + \tfrac{w_1 \eta_1}{2} \|y_0 - y\|^2 - \tfrac{w_N \eta_N}{2} \|y_N - y\|^2 \\
&\quad + \textstyle\sum_{k=1}^{N} w_k \bar{\epsilon} + \sum_{k=1}^{N} \frac{(M + \|G_{k-1}\|_*)^2 w_k}{\alpha_X \tau_k} + \sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1} - x \rangle,
\end{aligned}
$$

where the last two inequalities follows from similar techniques in the proof of Theorem 27. Dividing both sides of the above inequality, and using the convexity of Q and the definition of $\bar{z}_N$, we have

$$
\begin{aligned}
\max_{z \in X \times K_*} Q(\bar{z}_N, z) \leq (\textstyle\sum_{k=1}^{N} w_k)^{-1} [&w_1 \tau_1 \Omega_X^2 + \tfrac{w_1 \eta_1}{2} \|y_0 - y\|^2 - \tfrac{w_N \eta_N}{2} \|y_N - y\|^2 \\
&+ \textstyle\sum_{k=1}^{N} w_k \bar{\epsilon} + \sum_{k=1}^{N} \frac{(M + \|G_{k-1}\|_*)^2 w_k}{\alpha_X \tau_k} + \sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1} - x \rangle],
\end{aligned} \tag{4.79}
$$

which, in view of (4.36) and (4.37), then implies

$$
\begin{aligned}
\mathrm{gap}_*(\bar{z}_N) \leq (\textstyle\sum_{k=1}^{N} w_k)^{-1} [&2 w_1 \tau_1 \Omega_X^2 + \tfrac{w_1 \eta_1}{2} \|y_0 - y_*\|^2 - \tfrac{w_N \eta_N}{2} \|y_N - y_*\|^2 \\
&+ \textstyle\sum_{k=1}^{N} w_k \epsilon + \sum_{k=1}^{N} \frac{[\|\Delta_k\|_*^2 + 2(M + \|G_{k-1}\|_*)^2] w_k}{2 \alpha_X \tau_k} + \sum_{k=1}^{N} w_k \langle \Delta_{k-1}, x_{k-1} - x_{k-1}^v \rangle].
\end{aligned}
$$

Taking expectation w.r.t. ξ_k on both sides of above inequality, and using (4.45) and the fact

that $x_{k-1} - x_{k-1}^v$ is independent of Δ_{k-1}, we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \leq (\sum_{k=1}^N w_k)^{-1}[2w_1\tau_1\Omega_X^2 + \tfrac{w_1\eta_1}{2}\|y_0 - y_*\|^2 + \sum_{k=1}^N \tfrac{6M^2w_k}{\alpha_X\tau_k}] + \bar{\epsilon}.$$

The proof of part b) is similar to the one for Theorem 29.b) and hence the details are skipped. ■

In the following two corollaries, we provide two different parameter settings for the selection of $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$, both of which can guarantee the convergence of Algorithm 2 in terms of the gap functions $\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)]$ and $\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)]$. Moreover, the first one in Corollary 39 shows that if $M = 0$ and N is properly chosen, then one can ensure the boundedness of $\mathbb{E}[\|y_* - \bar{y}_N\|^2]$, while the other one in Corollary 40 can guarantee the boundedness of $\mathbb{E}[\|y_* - \bar{y}_N\|^2]$ by properly choosing N, even under the assumption that $M > 0$.

Corollary 39 *If*

$$w_k = k,\ \tau_k = \tfrac{k-1}{2}\mu_h \text{ and } \eta_k = \tfrac{4\|A\|^2}{k\alpha_X\mu_h}, \tag{4.80}$$

then for any $N \geq 1$, we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \leq \tfrac{8\|A\|^2\|y_0-y_*\|^2}{\alpha_X\mu_h(N+1)N} + \tfrac{24M^2}{\alpha_X\mu_h(N+1)} + \bar{\epsilon}, \tag{4.81}$$

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \leq \tfrac{8\|A\|^2\|y_0\|^2}{\alpha_X\mu_h(N+1)N} + \tfrac{24M^2}{\alpha_X\mu_h(N+1)} + \bar{\epsilon}, \tag{4.82}$$

$$\mathbb{E}[\|\delta\|] \leq \tfrac{16\|A\|^2\|y_*-y_0\|}{N(N+1)\alpha_X\mu_h} + \tfrac{8\sqrt{6}\|A\|M}{\alpha_X\mu_h N^{3/2}} + \tfrac{4\|A\|\sqrt{\bar{\epsilon}}}{(N+1)\sqrt{\alpha_X\mu_h}}, \tag{4.83}$$

$$\mathbb{E}[\|y_* - \bar{y}_N\|^2] \leq \|y_* - y_0\|^2 + \tfrac{12M^2\alpha_X N}{\|A\|^2} + \tfrac{N(N+1)\alpha_X\mu_h}{2\|A\|^2}\bar{\epsilon}. \tag{4.84}$$

Proof. Clearly, the parameters w_k, τ_k and η_k in (4.80) satisfy (4.31) with (4.31).b)

replaced by (4.75). It then follows from Theorem 38 and (4.80) that

$$\begin{aligned}
\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] &\leq \tfrac{2}{N(N+1)}\left[\tfrac{4\|A\|^2\|y_*-y_0\|^2}{\alpha_X\mu_h}+\tfrac{12M^2N}{\alpha_X\mu_h}\right]+\bar{\epsilon}\\
&\leq \tfrac{8\|A\|^2\|y_0-y_*\|^2}{\alpha_X\mu_h(N+1)N}+\tfrac{24M^2}{\alpha_X\mu_h(N+1)}+\bar{\epsilon},\\
\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] &\leq \tfrac{8\|A\|^2\|y_0\|^2}{\alpha_X\mu_h(N+1)N}+\tfrac{24M^2}{\alpha_X\mu_h(N+1)}+\bar{\epsilon},\\
\mathbb{E}[\|\delta\|] &\leq \tfrac{8\|A\|^2}{\alpha_X\mu_h N(N+1)}\left[2\|y_*-y_0\|+\sqrt{\tfrac{\alpha_X\mu_h}{2\|A\|^2}(\tfrac{6M^2}{\alpha_X}2N+\tfrac{N(N+1)}{2}\bar{\epsilon})}\right]\\
&\leq \tfrac{16\|A\|^2\|y_*-y_0\|}{N(N+1)\alpha_X\mu_h}+\tfrac{8\sqrt{6}\|A\|M}{\alpha_X\mu_h N^{3/2}}+\tfrac{4\|A\|\sqrt{\bar{\epsilon}}}{(N+1)\sqrt{\alpha_X\mu_h}},\\
\mathbb{E}[\|y_*-\bar{y}_N\|^2] &\leq \|y_*-y_0\|^2+\tfrac{2}{N(N+1)}\textstyle\sum_{k=1}^N\tfrac{k\alpha_X\mu_h}{2\|A\|^2}\left(2N\tfrac{12M^2}{\mu_h}+\tfrac{N(N+1)}{2}\bar{\epsilon}\right)\\
&= \|y_*-y_0\|^2+\tfrac{12M^2\alpha_X N}{\|A\|^2}+\tfrac{N(N+1)\alpha_X\mu_h}{2\|A\|^2}\bar{\epsilon}.
\end{aligned}$$

■

Corollary 40 *If*

$$w_k = k,\ \tau_k = \tfrac{k-1}{2}\mu_h \ \textit{and}\ \ \eta_k = \tfrac{4\|A\|^2N}{k\alpha_X\mu_h}, \tag{4.85}$$

then for any $N \geq 1$, we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_N)] \leq \tfrac{8\|A\|^2\|y_0-y_*\|^2+24M^2}{\alpha_X\mu_h(N+1)}+\bar{\epsilon}, \tag{4.86}$$

$$\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N)] \leq \tfrac{8\|A\|^2\|y_0\|^2+24M^2}{\alpha_X\mu_h(N+1)}+\bar{\epsilon}, \tag{4.87}$$

$$\mathbb{E}[\|\delta\|] \leq \tfrac{16\|A\|^2\|y_*-y_0\|}{(N+1)\alpha_X\mu_h+16\sqrt{3}\|A\|M}+\tfrac{4\|A\|\sqrt{\bar{\epsilon}}}{\sqrt{(N+1)\alpha_X\mu_h}}, \tag{4.88}$$

$$\mathbb{E}[\|y_*-\bar{y}_N\|^2] \leq \|y_*-y_0\|^2+\tfrac{24M^2\alpha_X}{\|A\|^2}+\tfrac{(N+1)\alpha_X\mu_h}{2\|A\|^2}\bar{\epsilon}. \tag{4.89}$$

Proof. The proofs of (4.86)-(4.89) are similar to Corollary 39 and hence the details are skipped. ■

4.3.2 Convergence analysis for DSA under strong convexity

Our goal in this subsection is to establish the complexity of the DSA algorithm for solving problem 4.9 under the strong convex assumption about h^i, $i = 1, 2, 3$, i.e., $\exists \mu_i > 0$ s.t.

$$h^i(x_1, c) - h^i(x_2, c) - \langle (h^i)'(x_2, c), x_1 - x_2 \rangle \geq \mu_i P_{X^i}(x_2, x_1),\ \forall x_1, x_2 \in X^i. \quad (4.90)$$

We describe some convergence properties for the innermost and middle loop of the DSA algorithm under the strong convexity assumptions in (4.90) in Lemma 41 and 42, respectively. The proofs for these results are similar to those for Lemma 34 and 35.

Lemma 41 below describes the convergence properties for the innermost loop of the DSA algorithm.

Lemma 41 *If the parameters $\{w_k^3\}$, $\{\tau_k^3\}$ and $\{\eta_k^3\}$ are set to* (4.80) *(with $M = 0$ and $A = A_j^3$) and*

$$N_3 \equiv N_{3,j} := \frac{2\sqrt{6}\|A_j^3\|\|y_{*,j}^3 - y_0^3\|}{\sqrt{\alpha_3 \mu_3 \epsilon}}, \quad (4.91)$$

then $B_j^3 \bar{y}_j^3$ is a stochastic $(\epsilon/3)$-subgradient of the value function v^3 at x_{j-1}^2. Moreover, there exists a constant $M_3 \geq 0$ such that $\|v^3(x_1, \xi^{[2]}) - v^3(x_2, \xi^{[2]})\| \leq M_3 \|x_1 - x_2\|, \forall x_1, x_2 \in X^3$ and

$$\mathbb{E}[\|B_j^3 \bar{y}_j^3\|_*^2 | \xi^{[2]}] \leq M_3. \quad (4.92)$$

In addition, there exists a vector $\delta \in \mathbb{R}^{m^3}$ s.t.

$$\begin{aligned} &\mathbb{E}[h^3(\bar{x}^3, c^3) - V^2(\bar{x}^2, \xi^{[3]})] \leq \epsilon/3, \\ &A^3 \bar{x}^3 - B^3 \bar{x}^2 - b^3 - \delta \in K^3 \text{ a.s.}, \\ &\mathbb{E}[\|\delta\| | \xi^{[2]}] \leq \epsilon/3. \end{aligned}$$

Proof. In view of Corollary 39 (with $M = 0$ and $\bar{\epsilon} = 0$) and the definition of N_3 in (4.91), we have

$$\mathbb{E}[\text{gap}_*(\bar{z}_j^3) | \xi^{[2]}] \ \leq \frac{8\|A_j^3\|^2 \|y_0^3 - y_*^3\|^2}{\alpha_3 \mu_3 (N_3 + 1) N_3} \leq \frac{\epsilon}{3}.$$

This observation, in view of Lemma 25, then implies that $B_j^3 \bar{y}_j^3$ is a stochastic $(\epsilon/3)$-subgradient of v^3 at x_{j-1}^2. Moreover, it follows from (4.84) (with $M = 0$ and $\bar{\epsilon} = 0$) that $\mathbb{E}[\|y_{*,j}^3 - \bar{y}_j^3\||\xi^{[2]}] \leq \|y_{*,j}^3 - y_0^3\|$. This inequality, in view of the selection of N_3 in (4.91), the assumption that $y_{*,j}^3$ is well-defined, and (4.61), then implies the latter part of our result. The techniques are similar to the proof of Lemma 34 and the details are skipped.

■

Lemma 41 below describes the convergence properties for the middle loop of the DSA algorithm.

Lemma 42 *Assume that the parameters for the innermost loop are set according to Lemma 41. If the parameters $\{w_j^2\}$, $\{\tau_j^2\}$ and $\{\eta_j^2\}$ for the middle loop are set to* (4.85) *(with $M = M_3$ and $A = A_i^2$) and*

$$N_2 \equiv N_{2,i} := \frac{24\|A_i^2\|^2\|y_0^2 - y_{*,i}^2\|^2 + 72M_3^2}{\alpha_2\mu_2\epsilon}, \tag{4.93}$$

then $B_i^2\bar{y}_i^2$ is a stochastic $(2\epsilon/3)$-subgradient of the value function v^2 at x_{i-1}^1. Moreover, there exists a constant $M_2 \geq 0$ such that $\|v^2(x_1, \xi^{[1]}) - v^2(x_2, \xi^{[1]})\| \leq M_2\|x_1 - x_2\|, \forall x_1, x_2 \in X^2$ and

$$\mathbb{E}[\|B_i^2\bar{y}_i^2\|_*^2|\xi^{[1]}] \leq M_2. \tag{4.94}$$

In addition, there exists a vector $\delta \in \mathbb{R}^{m^2}$ s.t.

$$\begin{aligned}
&\mathbb{E}[h^2(\bar{x}^2, c^2) + v^3(\bar{x}^2|\xi^2) - V^2(\bar{x}^1, \xi^{[2]})|\xi^{[1]}] \leq 2\epsilon/3,\\
&A^2\bar{x}^2 - B^2\bar{x}^1 - b^2 - \delta \in K^2 \text{ a.s.},\\
&\mathbb{E}[\|\delta\||\xi^{[1]}] \leq 2\epsilon/3.
\end{aligned}$$

Proof. By Lemma 41, the stochastic subgradients of v^3 are computed by the innermost loop with tolerance $\bar{\epsilon} = \epsilon/3$. In view of Corollary 40 (with $M = M_3$ and $\bar{\epsilon} = \epsilon/3$) and the definition of N_2 in (4.93), we have

$$\mathbb{E}[\text{gap}_*(\bar{z}_i^2)|\xi^{[1]}] \leq \frac{8\|A_i^2\|^2\|y_0^2 - y_{*,i}^2\|^2 + 24M_3^2}{\alpha_2\mu_2(N_2+1)} + \bar{\epsilon} \leq \frac{2\epsilon}{3}.$$

This observation, in view of Lemma 25, then implies that $B_i^2\bar{y}_i^2$ is a stochastic $(2\epsilon/3)$-subgradient v^2 at x_{i-1}^1. Moreover, it follows from (4.89) (with $M = M_3$ and $\bar{\epsilon} = \epsilon/3$) that

$$\mathbb{E}[\|y_{*,i}^2 - \bar{y}_i^2\|^2|\xi^{[1]}] \leq \|y_{*,i}^2 - y_0^2\|^2 + \tfrac{24M_3^2\alpha_2}{\|A_i^2\|^2} + \tfrac{(N_2+1)\alpha_2\mu_2}{6\|A_i^2\|^2}\epsilon.$$

This inequality, in view of the selection of N_2 in (4.93), the assumption that $y_{*,i}^2$ is well-defined, and (4.61), then implies the latter part of our result. The techniques are similar to the proof of Lemma 35 and the details are skipped. ■

We are now ready to state the main convergence properties of the DSA algorithm for solving strongly convex three-stage stochastic optimization problems.

Theorem 43 *Suppose that the parameters for the innermost and middle loop in the DSA algorithm are set according to Lemma 41 and Lemma 42, respectively. If the parameters* $\{w_i\}$, $\{\tau_i\}$ *and* $\{\eta_i\}$ *for the outer loop are set to (4.80) (with* $M = M_2$ *and* $A = A^1$*) and*

$$\begin{aligned} N_1 := \max\Big\{ & \tfrac{4\sqrt{3}\|A^1\|\|y_0^1\|}{\sqrt{\alpha_1\mu_1\epsilon}} + \tfrac{4(6M_2)^2}{\alpha_1\mu_1\epsilon}, \\ & \tfrac{4\sqrt{3}\|A^1\|(\sqrt{\|y_*^1 - y_0^1\|}+\sqrt{2})}{\sqrt{\alpha_1\mu_1\epsilon}} + \left(\tfrac{24\sqrt{6}\|A^1\|M_2}{\alpha_1\mu_1\epsilon}\right)^{2/3} \Big\}, \end{aligned} \tag{4.95}$$

then we will find a solution $\bar{x}^1 \in X^1$ *and a vector* $\delta \in \mathbb{R}^{m^1}$ *s.t.*

$$\begin{aligned} & \mathbb{E}[h(\bar{x}^1, c^1) + v^2(\bar{x}^1, \xi^1) - (h(x^*, c^1) + v^2(x^*, \xi^1))] \leq \epsilon, \\ & A\bar{x}^1 - b - \delta \in K^1, a.s., \\ & \mathbb{E}[\|\delta\|] \leq \epsilon, \end{aligned}$$

where x^* *denotes the optimal solution of problem 4.9.*

Proof. By Lemma 42, the stochastic subgradients of v^2 are computed by the middle loop with tolerance $\bar{\epsilon} = 2\epsilon/3$. In view of Corollary 39 (with $M = M_2$ and $\bar{\epsilon} = 2\epsilon/3$) and

the definition of N_1 in (4.95), we conclude that there exist $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\begin{aligned}\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N^1)] &\le \tfrac{8\|A^1\|^2\|y_0^1\|^2}{\alpha_1\mu_1(N_1+1)N_1} + \tfrac{24M_2^2}{\alpha_1\mu_1(N_1+1)} + \tfrac{2\epsilon}{3} \le \epsilon,\\ \mathbb{E}[\|\delta\|] &\le \tfrac{16\|A^1\|^2\|y_*^1-y_0^1\|}{N_1(N_1+1)\alpha_1\mu_1} + \tfrac{8\sqrt{6}\|A^1\|M_2}{\alpha_1\mu_1 N_1^{3/2}} + \tfrac{4\|A^1\|\sqrt{2\epsilon}}{(N_1+1)\sqrt{3\alpha_1\mu_1}} \le \epsilon,\end{aligned}$$

which together with Lemma 32 then imply our result. ■

In view of Lemma 42 and Theorem 43, the number of random samples ξ_2 and ξ_3 will be bounded by N_1 and $N_1 \times N_2$, i.e., $\mathcal{O}(1/\epsilon)$ and $\mathcal{O}(1/\epsilon^2)$, respectively, under the assumption that the random variables appearing in the definition of N_2 (i.e., A_i^2 and $y_{*,i}^2$) are bounded.

4.4 DSA for general multi-stage stochastic optimization

In this section, we consider a multi-stage stochastic optimization problem given by

$$\begin{aligned}&\min\ \{h^1(x^1,c^1) + v^2(x^1,\xi^1)\}\\ &\text{s.t. } A^1x^1 - b^1 \in K^1,\\ &\qquad x^1 \in X^1,\end{aligned} \tag{4.96}$$

where the value factions v^t, $t = 2,\ldots,T$, are recursively defined by

$$\begin{aligned}v^t(x^{t-1},\xi^{[t-1]}) &:= F^{t-1}(x^{t-1},p^{t-1}) + \mathbb{E}[V^t(x^{t-1},\xi^{[t]})|\xi^{[t-1]}],\ \ t = 2,\ldots,T-1,\\ V^t(x^{t-1},\xi^{[t]}) &:= \min\ \{h^t(x^t,c^t) + v^{t+1}(x^t)\}\\ &\qquad \text{s.t. } A^tx^t - b^t - B^tx^{t-1} \in K^t,\\ &\qquad\quad x^t \in X^t,\end{aligned} \tag{4.97}$$

and

$$\begin{aligned} v^T(x^{T-1}, \xi^{[T-1]}) &:= \mathbb{E}_{\xi^T}[V^T(x^{T-1}, \xi^{[T]})|\xi^{[T-1]}], \\ V^T(x^{T-1}, \xi^{[T]}) &:= \min\ h^T(x^T, c^T) \\ &\quad \text{s.t. } A^T x^T - b^T - B^T x^{T-1} \in K^T, \\ &\quad x^T \in X^T. \end{aligned} \tag{4.98}$$

Here $\xi^t := (A^t, b^t, B^t, c^t, p^t)$ are random variables, $h^t(\cdot, c^t)$ are relatively simple functions, $F^t(\cdot, p^t)$ are general (not necessarily simple) Lipschitz continuous convex functions and K^t are convex cones, $\forall t = 1, \ldots, T$. We also assume that one can compute the subgradient $F'(x^t, p^t)$ of function $F^t(x^t, p^t)$ at any point $x^t \in X^t$ for a given parameter p^t.

Problem (4.96) is more general than problem (4.6) (or equivalently problem (4.9)) in the following sense. First, we are dealing with a more complicated multi-stage stochastic optimization problem where the number of stages T (4.96) can be greater than three. Second, the value function $v^t(x^{t-1}, \xi^{[t-1]})$ in (4.97) is defined as the summation of $F^{t-1}(x^{t-1}, p^{t-1})$ and $\mathbb{E}[V^t(x^{t-1}, \xi^{[t]})|\xi^{[t-1]}]$, where F^{t-1} is not necessarily simple. We intend to generalize the DSA algorithm in Sections 4.2 and 4.3 for solving problem (4.96). More specifically, we show how to compute a stochastic ϵ-subgradient of v^{t+1} at x^t, $t = 1, \ldots, T-2$, in a recursive manner until we obtain the ϵ-subgradient of v^T at x^{T-1}.

We are now ready to formally state the DSA algorithm for solving the multi-stage stochastic optimization problem in (4.96). Observe that the following notations will be used in the algorithm:

- N_t is the number of iterations for stage t subproblem and k_t is the corresponding index, i.e., $k_t = 1, \ldots, N_t$.
- $\xi^t_{k_{t-1}} = (A^t_{k_{t-1}}, b^t_{k_{t-1}}, B^t_{k_{t-1}}, c^t_{k_{t-1}}, p^t_{k_{t-1}})$ is the k_{t-1} th random scenarios in stage t subproblem, $(x^t_{k_t}, y^t_{k_t})$ are the k_t th iterates in stage t subproblem.
- For simplicity, we denote $\xi^t_{k_{t-1}}$ as ξ^t_k, $(x^t_{k_t}, y^t_{k_t})$ as (x^t_k, y^t_k).

Algorithm 8 DSA for multi-stage stochastic programs

Input: initial points $\{x_0^t\}$, $k_t = 1, \forall t$, iteration number N_t and stepsize strategy $\{w_k\}$.

Start with procedure $\mathrm{DSA}(1, 0)$.

procedure: $\mathrm{DSA}(t, u)$

for $k_t = 1, \ldots, N_t$ **do**

if $t < T$ **then**

Generate random scenarios ξ_k^{t+1}.

$(\bar{x}^{t+1}, \bar{y}^{t+1}) = \mathrm{DSA}(t+1, x_k^t)$ and $G(x_{k-1}^t, \xi_k^{t+1}) = (B_k^{t+1})^T \bar{y}^{t+1}$.

else

$G(x_{k-1}^T, \xi_k^{T+1}) = 0$.

end if

$(x_k^t, y_k^t) = \mathrm{SPDT}(x_{k-1}^t, y_{k-1}^t, y_{k-2}^t, G(x_{k-1}^t, \xi_k^{t+1}), u, \xi_{k-1}^t, h^t, X^t, K_*^t, \theta_k^t, \tau_k^t, \eta_k^t)$.

end for

return: $\bar{z}^t = \sum_{k=1}^{N_t} w_k z_k^t / \sum_{k=1}^{N_t} w_k$.

In order to show the convergence of the above DSA algorithm, we need the following assumption on the boundedness of the operators B^t:

$$\|B^t\| \leq \mathcal{B}_t, \ \forall t = 2, \cdots, T. \tag{4.99}$$

Lemma 44 below establishes some convergence properties of the DSA algorithm for solving the last stage problem.

Lemma 44 *Suppose that the algorithmic parameters in the DSA algorithm applied to problem 4.96 are chosen as follows.*

a) *For a general convex problem,* $\{w_k^T\}$, $\{\tau_k^T\}$ *and* $\{\eta_k^T\}$ *are set to* (4.48) *(with* $M = 0$ *and* $A = A_k^T$*) and*

$$N_T \equiv N_{T,k} := \frac{T\sqrt{2}\|A_k^T\|[2(\Omega_T)^2 + \|y_{*,k}^T - y_0^T\|^2]}{\sqrt{\alpha_T}\epsilon}. \tag{4.100}$$

b) *Under the strongly convex assumption* (4.90)*,* $\{w_k^T\}$*,* $\{\tau_k^T\}$ *and* $\{\eta_k^T\}$ *are set to* (4.80) *(with* $M = 0$ *and* $A = A_k^T$*) and*

$$N_T \equiv N_{T,k} := \frac{\sqrt{8T}\|A_k^T\|\|y_{*,k}^T - y_0^T\|}{\sqrt{\alpha_T \mu_T \epsilon}}. \tag{4.101}$$

Then $B_k^T \bar{y}_k^T$ *is a stochastic* (ϵ/T)*-subgradient of the value function* v^T *at* x_{k-1}^{T-1}*. Moreover, there exists a constant* $M_T \geq 0$ *such that* $\|v^T(x_1, \xi^{[T-1]}) - v^T(x_2, \xi^{[T-1]})\| \leq M_T \|x_1 - x_2\|, \forall x_1, x_2 \in X^T$ *and*

$$\mathbb{E}_{\xi^T}[\|B_k^T \bar{y}_k^T\|_*^2] \leq M_T. \tag{4.102}$$

Proof. The innermost loop of the DSA algorithm is equivalent to the application of Algorithm 2 to the last stage saddle point problem in (4.12). Note that for this problem, we do not have any subsequent stages and hence $\tilde{v} = 0$. In other words, the subgradients of $\tilde{v}$ are exact. To show part a), in view of Corollary 30 (with $M = 0$ and $\bar{\epsilon} = 0$) and the definition of N_T in (4.100), we have

$$\mathbb{E}[\text{gap}_*(\bar{z}_k^T)|\xi^{[T-1]}] \leq \frac{\sqrt{2}\|A_k^T\|[2(\Omega_T)^2 + \|y_*^T - y_0^T\|^2]}{\sqrt{\alpha_T} N_T} \leq \frac{\epsilon}{T}.$$

This observation, in view of Lemma 25, then implies that $B_k^T \bar{y}_k^T$ is a stochastic (ϵ/T)-subgradient of v^T at x_{j-1}^{T-1}. Moreover, it follows from (4.52) (with $M = 0$ and $\bar{\epsilon} = 0$) that

$$\mathbb{E}[\|y_{*,k}^T - \bar{y}_k^T\|^2|\xi^{[T-1]}] \leq \|y_{*,k}^T - y_0^T\|^2 + 4(\Omega_T)^2 + \frac{(N_T+1)\epsilon}{2}.$$

This inequality, in view of the selection of N_T in (4.100), the assumption that $y_{*,k}^T$ is well-defined, and (4.99), then implies the latter part of our result. Similarly, the result in (4.101) follows from Corollary 39 (with $M = 0$ and $\bar{\epsilon} = 0$) and the definition of N_T in (4.101). ■

We show in Lemma 45 some convergence properties of the middle loops of the DSA algorithm.

Lemma 45 *Assume that the parameters for the innermost loop are set according to Lemma 44. Moreover, suppose that the algorithmic parameters for the middle loops are chosen as follows.*

a) *For general convex problem, the parameters* $\{w_k^t\}$, $\{\tau_k^t\}$ *and* $\{\eta_k^t\}$ *for the middle loops* ($t = 2, \ldots, T-1$) *are set to* (4.53) *(with* $M = M_{t+1}$ *and* $A = A_k^t$*) and*

$$N_t \equiv N_{t,k} := \left(\frac{4\sqrt{2}T\|A_k^t\|\Omega_t}{\sqrt{\alpha_t}\epsilon}\right)^{\frac{2}{3}} + \left[\frac{2T\left(\|A_k^t\|\|y_{*,k}^t - y_0^t\|^2 + 4\sqrt{3}M_{t+1}\Omega_t\right)}{\sqrt{\alpha_t}\epsilon}\right]^2. \tag{4.103}$$

b) *Under strongly convex assumption* (4.90)*, the parameters* $\{w_k^t\}$, $\{\tau_k^t\}$ *and* $\{\eta_k^t\}$ *for the middle loops* ($t = 2, \ldots, T-1$) *are set to* (4.85) *(with* $M = M_{t+1}$ *and* $A = A_k^t$*) and*

$$N_t \equiv N_{t,k} := \frac{8T\|A_k^t\|^2\|y_0^t - y_{*,k}^t\|^2 + 24TM_{t+1}^2}{\alpha_t\mu_t\epsilon}. \tag{4.104}$$

Then $B_k^t\bar{y}_k^t$ *is a stochastic* $((T+1-t)\epsilon/T)$*-subgradient of the value function* v^t *at* x_{k-1}^{t-1}*. Moreover, there exists a constant* $M_t \geq 0$ *such that* $\|v^t(x_1, \xi^{[t-1]}) - v^t(x_2, \xi^{[t-1]})\| \leq M_t\|x_1 - x_2\|, \forall x_1, x_2 \in X^t$ *and*

$$\mathbb{E}[\|B_k^t\bar{y}_k^t\|_*^2|\xi^{[t-1]}] \leq M_t. \tag{4.105}$$

Proof. The middle loops ($t = 2, \ldots, T-1$) of the DSA algorithm applied to multistage stochastic optimization is equivalent to the application of Algorithm 2 to the second stage saddle point problem in (4.11). Note that for this problem, we have $\tilde{v} = v^{t+1}$. Moreover, by Lemma 44, the stochastic subgradients of v^T are computed by the innermost loop with tolerance $\bar{\epsilon} = \epsilon/T$. To show part a), in view of Corollary 31 (with $M = M_{t+1}$ and $\bar{\epsilon} = (T-t)\epsilon/T$) and the definition of N_t in (4.103), we have

$$\mathbb{E}[\mathrm{gap}_*(\bar{z}_k^t)|\xi^{[t-1]}] \leq \frac{2\sqrt{2}\|A_k^t\|\Omega_t}{N_t\sqrt{\alpha_t N_t}} + \frac{\|A_k^t\|\|y_{*,k}^t - y_0^t\|^2 + 4\sqrt{3}M_{t+1}\Omega_t}{\sqrt{\alpha_t N_t}} + \bar{\epsilon} \leq \frac{(T+1-t)\epsilon}{T}.$$

This observation, in view of Lemma 25, then implies that $B_k^t \bar{y}_k^t$ is a stochastic $((T+1-t)\epsilon/T)$-subgradient v^t at x_{k-1}^{t-1}. Moreover, it follows from (4.57) (with $M = M_{t+1}$ and $\bar{\epsilon} = (T-t)\epsilon/T$) that

$$\mathbb{E}[\|y_{*,k}^t - \bar{y}_k^t\|^2 | \xi^{[t-1]}] \leq \|y_{*,k}^t - y_0^t\|^2 + \tfrac{2\Omega_t^2}{N_t} + \tfrac{\sqrt{6}(1+\alpha_t)M_{t+1}\Omega_t}{\|A_k^t\|} + \tfrac{\sqrt{\alpha_t N_t}\epsilon}{3\sqrt{2}\|A_k^t\|}.$$

This inequality, in view of the selection of N_t in (4.103), the assumption that $y_{*,k}^t$ is well-defined, and (4.99), then implies the latter part of our result. Similarly, in view of Corollary 40, we have part b). ∎

We are now ready to establish the main convergence properties of the DSA algorithm for solving general multi-stage stochastic optimization problems with $T \geq 3$.

Theorem 46 *Suppose that the parameters for the inner loops in the DSA algorithm are set according to Lemma 44 and Lemma 45. Moreover, assume that the algorithmic parameters in the outer loop of the DSA algorithm are chosen as follows.*

a) *For general convex problem, the parameters $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$ for the outer loop are set to (4.48) (with $M = M_2$ and $A = A^1$) and*

$$N_1 := \max\left\{ \tfrac{2\sqrt{2}T\|A^1\|[2(\Omega_1)^2 + \|y_0^1\|^2]}{\sqrt{\alpha_1}\epsilon} + \left(\tfrac{8\sqrt{3}TM_2\Omega_1}{\sqrt{\alpha_1}\epsilon}\right)^2, \right.$$
$$\left. \tfrac{6T\|A^1\|(\sqrt{2\alpha_1}\|y_*^1 - y_0^1\| + 2\Omega_1) + 27(T-1)\sqrt{\alpha_1}\|A^1\|}{\alpha_1 T\epsilon} + \left(\tfrac{6\sqrt{3}M_2(\sqrt{2}\|A^1\| + \sqrt{\alpha_1})}{\alpha_1\epsilon}\right)^2 \right\}. \tag{4.106}$$

b) *Under strongly convex assumption* (4.90)*, the parameters $\{w_k\}$, $\{\tau_k\}$ and $\{\eta_k\}$ for the outer loop are set to (4.80) (with $M = M_2$ and $A = A^1$) and*

$$N_1 := \max\left\{ \tfrac{4\sqrt{T}\|A^1\|\|y_0^1\|}{\sqrt{\alpha_1\mu_1\epsilon}} + \tfrac{24TM_2^2}{\alpha_1\mu_1\epsilon}, \right.$$
$$\left. \tfrac{4\sqrt{3}\|A^1\|\sqrt{\|y_*^1 - y_0^1\|}}{\sqrt{\alpha_1\mu_1\epsilon}} + \left(\tfrac{24\sqrt{6}\|A^1\|M_2}{\alpha_1\mu_1\epsilon}\right)^{2/3} + \tfrac{12\|A^1\|\sqrt{T-1}}{\sqrt{\alpha_1\mu_1 T\epsilon}} \right\}. \tag{4.107}$$

Then we will find a solution $\bar{x}^1 \in X^1$ and a vector $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\begin{aligned}
&\mathbb{E}[h(\bar{x}^1, c) + v^2(\bar{x}^1, \xi^1) - (h(x^*, c) + v^2(x^*, \xi^1))] \leq \epsilon,\\
&A\bar{x}^1 - b - \delta \in K^1, a.s.,\\
&\mathbb{E}[\|\delta\|] \leq \epsilon,
\end{aligned}$$

where x^* denotes the optimal solution of problem 4.9.

Proof. The outer loop of the DSA algorithm is equivalent to the application of Algorithm 2 to the first stage saddle point problem in (4.14). Note that for this problem, we have $\tilde{v} = v^2$. Moreover, by Lemma 45, the stochastic subgradients of v^2 are computed by the middle loop with tolerance $\bar{\epsilon} = (T-1)\epsilon/T$. To show part a), in view of Corollary 30 (with $M = M_2$ and $\bar{\epsilon} = (T-1)\epsilon/T$) and the definition of N_1 in (4.106), we conclude that there exist $\delta \in \mathbb{R}^{m^1}$ s.t.

$$\begin{aligned}
\mathbb{E}[\mathrm{gap}_\delta(\bar{z}_N^1)] &\leq \tfrac{\sqrt{2}\|A^1\|(2\Omega_1^2 + \|y_0^1\|^2)}{\sqrt{\alpha_1 N_1}} + \tfrac{4\sqrt{3}M_2\Omega_1}{\sqrt{\alpha_1 N_1}} + \tfrac{(T-1)\epsilon}{T} \leq \epsilon,\\
\mathbb{E}[\|\delta\|] &\leq \tfrac{2\sqrt{2\alpha_1}\|A^1\|\|y_*^1 - y_0^1\| + 4\Omega_1\|A^1\|}{\alpha_1 N_1} + \tfrac{2M_2(\sqrt{6}\|A^1\| + \sqrt{3\alpha_1})}{\alpha_1\sqrt{N_1}} + \sqrt{\tfrac{3\|A^1\|(T-1)\epsilon}{N_1 T\sqrt{\alpha_1}}} \leq \epsilon,
\end{aligned}$$

which together with Lemma 32 then imply our result. Similarly, in view of Corollary 39, we have part b). ∎

In view of the results stated in Lemma 44, Lemma 45 and Theorem 46, the total number of scenarios required to find an ϵ-solution of (4.96) is given by $N_2 \times N_3 \times \ldots N_T$, and hence will grow exponentially with respect to T, no matter the objective functions are strongly convex or not. These sampling complexity bounds match well with those in [54, 55], implying that multi-stage stochastic optimization problems are essentially intractable for $T \geq 5$ and a moderate target accuracy. Hence, it is reasonable to use the DSA algorithm only for multi-stage stochastic optimization problems with T relatively small and ϵ relatively large. However, it is interesting to point out that the DSA algorithm only needs to go through the scenario tree once and hence its memory requirement increases only

linearly with respect to T. Moreover, the development of the complexity bounds of multi-stage stochastic optimization in terms of their dependence on various problem parameters may help us to further explore the structure of the problems and to identify special classes of problems possibly admitting faster solution methods.

It is also interesting to compare the DSA method with some other decomposition type algorithms. As discussed in Section 1, in the sample average approximation approach, we can apply a few different decomposition methods for solving the deterministic counterpart of the multi-stage stochastic optimization problem. These methods need to go through the whole scenario tree many times and hence it is necessary to store the scenario tree first. One widely used decomposition method is the stochastic dual dynamic programming (SDDP). Under the stage-wise independence assumption, SDDP iteratively builds cutting plane models to approximate the value functions starting from the last stage T until the first stage (backward iteration), and then generates feasible solutions starting from the first stage to the last stage (forward iteration). On the other hand, as a common drawback for cutting plane methods, SDDP converges slowly as the number of decision variables in each stage increases [76]. Improvement of cutting plane methods, e.g., based on the bundle-level method, however, can only be applied to two-stage problems only (see [101] and references therein). Moreover, the rate of convergence of SDDP, i.e., how many number of forward and backward iterations it will take to achieve a certain accurate solution, still remains unknown for multi-stage problems with $T \geq 3$, although its asymptotic convergence has been established for multi-stage linear programming [57] .

4.5 Numerical experiment

Our goal in this section is to report the results from our preliminary numerical experiments conducted to test the efficiency of the DSA method applied to a class of multi-stage asset allocation problems.

We consider a classic multistage asset allocation problem due to Dantzig and Infanger

[**dantzig1993multi**] given by

$$
\begin{array}{lll}
\min_{x^0,p^1,q^1} & \mathbb{E}\left[\min -u(\sum_{i=1}^{n+1} x_i^1)\right. & \left. + \ldots + \mathbb{E}\left[\min -u(\sum_{i=1}^{n+1} x_i^T)\right]\right] \\
\text{s.t.} 0 \le p_i^1 \le \bar{p}^1, & \text{s.t. } x_i^1 = R_i^1(x_i^0 - p_i^1 + q_i^1), & \text{s.t. } x_i^T = R_i^T(x_i^{T-1} - p_i^T + q_i^T),\ \ i = 1, \ldots, n, \\
0 \le q_i^1 \le \bar{q}^1, & x_{n+1}^1 = x_{n+1}^0 + \sum_{i=1}^n (1-\hat{p}_i)p_i^1 & x_{n+1}^T = x_{n+1}^{T-1} + \sum_{i=1}^n (1-\hat{p}_i)p_i^T, \\
& \quad - \sum_{i=1}^n (1+\hat{q}_i)q_i^1, & \quad - \sum_{i=1}^n (1+\hat{q}_i)q_i^T. \\
\sum_{i=1}^{n+1} x_i^0 = w_0, & 0 \le p_i^2 \le \bar{p}^2, & \\
x_i^0 \ge 0, & 0 \le q_i^2 \le \bar{q}^2, &
\end{array}
\tag{4.108}
$$

Here p_i^t and q_i^t, respectively, denote the amount of asset i that will be sold and purchased in period t, $\hat{p}_i$ and $\hat{q}_i$, respectively, denote the transaction costs for selling and purchasing one unit of asset i, and R_i^t represent the factor of random return for asset i from time t to time $t+1$. Moreover, the utility function $u(\cdot)$ describes the investor's risk preference. In particular, a linear utility function $u(\cdot)$ describes risk neutrality while a concave utility function models risk averseness. At the initial time period 0 the decision maker has a total amount of wealth w_0 in assets $i = 1, \ldots, n$ and in cash (indexed as asset $n+1$ for notational convenience). The dollar values of these initially available assets are denoted by $x_i^0,\ i = 1, \ldots, n+1$. In each period of time, short-selling of assets and borrowing of cash are allowed when $x_i < 0$, but there exist upper bounds $\bar{p}$ and $\bar{q}$ on the selling and buying amount, respectively. The goal of the decision maker is to maximize the expected utility $\mathbb{E}[u(\sum_{i=1}^{n+1} x_i^T)]$ for the portfolio over T periods of time.

4.5.1 Stagewise dependent random return

Our goal in this subsection is to demonstrate that the DSA method does not require the stage-wise independence assumption for the random returns. In this set of experiments, we model the correlation between asset returns using a factor model

$$
R^t = FV^t, \tag{4.109}
$$

which relates the asset returns $R^t = (R_1^t, \ldots, R_n^t)'$ to factors $V^t = (v_1^t, \ldots, v_h^t)'$ through a factor matrix $F \in \mathbb{R}^{n \times h}$. This factor model will allow us to consider the stage-wise dependence, e.g., given by

$$v_i^t = v_i^{t-1} + \epsilon_i^t,\ i = 1, \ldots, h, \tag{4.110}$$

where ϵ_i^t denote the independent random variation of the factor v_i in time t. We collected the data of weekly returns for $1,887$ assets from Thomson Reuters Datastream (`http://financial.thomsonreuters.com/`), and use these data to fit the random return model. We assume that the investor is risk averse with the utility function $u(\cdot)$ defined as the classic concave quadratic utility function [102], i.e., $u(W) = W - bW^2$ with $W = \sum_{i=1}^{n+1} x_i$. The value of $b = 1/(3W_0)$ is chosen according to [102], where W_0 is the initial total wealth. We generate three instances (Inst 1, Inst 2 and Inst 3) which have a fixed number of stages 3, but with different number of assets (5, 200 and 400).

When implementing the DSA algorithm, we consider every outer mostest loop as one iteration and run the algorithm for 100 iterations. For the sake of convenience, we set $N_1 = \ldots = N_T = 100$. Note that in order to estimate the function values for an output solution, we generate N realizations for the random vector $\{\epsilon^t\}, t = 1, \ldots, T-1$, and form a scenario tree consisting of N^{T-1} random returns $R^{j,t}$ at level $t\ \forall t = 1, \ldots, T, j = 1, \ldots, N^{T-1}$ according to (4.109) and (4.110). Then we will find a prefixed control policy $\{x^0, \bar{p}^t, \bar{q}^t\}, t = 1, \ldots, T-1$ based on the the output of the algorithm, and calculate other state variables according to

$$x_i^{j,t} = R_i^{j,t}(x_i^{t-1} - \bar{p}_i^t + \bar{q}_i^t), \forall i = 1, \ldots, n. \tag{4.111}$$

In other words, at stage 1, we will get N feasible $\{x^{j,1}\}, \forall j = 1, \ldots, N$ by (4.111), and at stage 2, we will get total N^2 feasible $\{x^{i,2}|x^{j,1}, R^{i,2}\}, \forall i = 1, \ldots, N^2$ by (4.111) and so on.

Table 4.1: Problem parameters for stagewise dependent return

	n	h	w_0	$\bar{p} = \bar{q}$	$\hat{p} = \hat{q}$	T
Inst 1	5	3	3	0.1	0.05	3
Inst 2	200	70	500	1	0.05	3
Inst 3	400	240	1,000	1	0.05	3

Table 4.2: Numerical results for DSA with stagewise dependent return

	#. of Iter.	0	10	20	60	100
Inst 1	FV	-4.0812	-4.1047	-4.1186	-4.1704	-4.2352
	Time(s)	0	1.96	4.02	12.37	21.00
Inst 2	FV	-665.79	-665.99	-666.13	-672.38	-675.80
	Time(s)	0	12.38	24.77	77.40	126.55
Inst 3	FV	-1.3326*e+3	-1.3334*e+3	-1.3337*e+3	-1.3414*e+3	-1.3493*e+3
	Time(s)	0	56.65	114.64	339.21	565.73

Then we estimate the function value by

$$FV = \frac{1}{N}\sum_{j=1}^{N}\left[-u(x^{j,1}) + \frac{1}{N}\sum_{i=N(j-1)+1}^{Nj}[-u(x^{i,2}) + \ldots]\right]. \tag{4.112}$$

It is worth noting that FV estimates an upper bound on the objective value at $\{x^0\}$. Nevertheless, our experimental results reported in Table 4.2 indicates that DSA does converge for these problems with stagewise dependent return.

4.5.2 Stagewise independent return

Our goal in the second set of experiments is to compare DSA with SDDP for solving problem (4.108). Since SDDP cannot be directly applied for solving problems with stagewise dependent return, in order to compare these two algorithms, we assume the random returns are stagewise independent given by

$$R^t = \mu + \epsilon^t, \forall t = 1, \ldots, T, \tag{4.113}$$

where $\mu \sim$ Uniform$[0.8, 1.2]$, and $\epsilon^t \sim$ Normal$(0, \sigma^2)$. Given starting point $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_T)$ and approximation of value function $\mathfrak{Q}_t$ for $t = 1, \ldots, T$, each iteration of the SDDP algo-

rithm consists of a forward step and a backward step to update the feasible solutions and the approximate value functions, respectively. We implemented the standard SDDP algorithm as described in [57]. In the forward step, for $t = 1$ to T, we randomly generate a sample $\{R^{tj}\}, j = 1 \ldots, M$ with size $M = 20$, and call the convex optimization solver CVX in Matlab 2017b to solve the subproblem

$$\begin{aligned}
\min_{x^{tj},p^{t+1,j},q^{t+1,j},s^{tj}} \quad & -u(\textstyle\sum_{i=1}^{n+1} x_i^{tj}) + \mathfrak{Q}_{t+1}(x^{tj}, p^{t+1,j}, q^{t+1,j}, s^{tj}) \\
\text{s.t. } & x_i^{tj} = R_i^{tj}(\bar{x}_i^{t-1} - \bar{p}_i^t + \bar{q}_i^t),\ \forall i = 1, \ldots, n, \\
& x_{n+1}^{tj} = \bar{x}_{n+1}^{t-1} + \textstyle\sum_{i=1}^{n}(1 - \hat{p}_i)\bar{p}_i^t - \sum_{i=1}^{n}(1 + \hat{q}_i)\bar{q}_i^t, \\
& x_i^{tj} - p_i^{t+1,j} + q_i^{t+1,j} - s_i^{tj} = 0,\ \forall i = 1, \ldots, n, \\
& x_{n+1}^{tj} + \textstyle\sum_{i=1}^{n}(1 - \hat{p}_i)p_i^{t+1,j} - \sum_{i=1}^{n}(1 + \hat{q}_i)q_i^{t+1,j} - s_{n+1}^{tj} = 0, \\
& x^{tj}, p^{t+1,j}, q^{t+1,j}, s^{tj} \geq 0,
\end{aligned}$$

where $\mathfrak{Q}_{t+1}$ denotes the current approximation for the value function at the $(t+1)$-th stage. Denoting $y^{tj} = (x^{tj}, p^{t+1,j}, q^{t+1,j}, s^{tj})$, we then compute the solution $\bar{y}^t \equiv (\bar{x}^t, \bar{p}^{t+1}, \bar{q}^{t+1}, \bar{s}^t) = \frac{1}{M}\sum_{j=1}^{M} y^{tj}$. In the backward step, for $t = T$ to 1, we call the CVX solver to solve the subproblem

$$\begin{aligned}
\min_{x^{tj},p^{t+1,j},q^{t+1,j},s^{tj}} \quad & -u(\textstyle\sum_{i=1}^{n+1} x_i^{tj}) + \mathfrak{Q}_{t+1}(x^{tj}, p^{t+1,j}, q^{t+1,j}, s^{tj}) \\
s.t. \ & \pi_i^{x,t,j} :\ x_i^{tj} = \tilde{R}_i^{tj}(\bar{x}_i^{t-1} - \bar{p}_i^t + \bar{q}_i^t),\ \forall i = 1, \ldots, n, \\
& \pi_{n+1}^{x,t,j} :\ x_{n+1}^{tj} = \bar{x}_{n+1}^{t-1} + \textstyle\sum_{i=1}^{n}(1 - \hat{p}_i)\bar{p}_i^t - \sum_{i=1}^{n}(1 + \hat{q}_i)\bar{q}_i^t, \\
& \pi_i^{s,t,j} :\ x_i^{tj} - p_i^{t+1,j} + q_i^{t+1,j} - s_i^{tj} = 0,\ \forall i = 1, \ldots, n, \\
& \pi_{n+1}^{s,t,j} :\ x_{n+1}^{tj} + \textstyle\sum_{i=1}^{n}(1 - \hat{p}_i)p_i^{t+1,j} - \sum_{i=1}^{n}(1 + \hat{q}_i)q_i^{t+1,j} - s_{n+1}^{tj} = 0, \\
& x^{tj}, p^{t+1,j}, q^{t+1,j}, s^{tj} \geq 0,
\end{aligned}$$

Table 4.3: Problem parameters for stagewise independent data

	n	w_0	$\bar{p} = \bar{q}$	$\hat{p} = \hat{q}$	T	σ
Inst 4	5	3	0.1	0.05	3	0.05
Inst 5	200	500	1	0.05	3	0.1
Inst 6	400	1,000	1	0.05	3	0.2
Inst 7	5	3	0.1	0.05	4	0.1
Inst 8	5	3	0.1	0.05	5	0.1

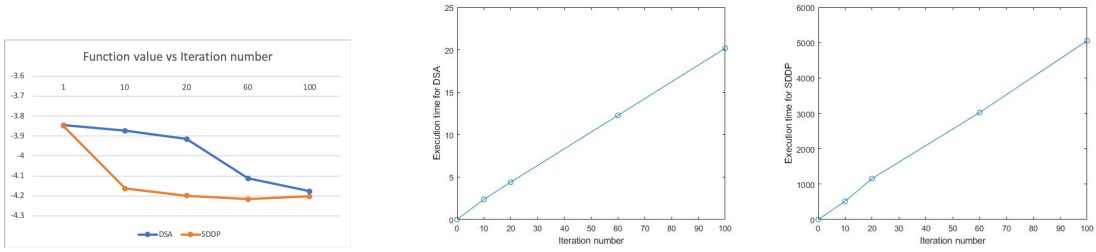
for the fixed set of randomly generated scenarios $\tilde{R}^{tj}, j = 1, \ldots, N_t$, with $N_t = 100$, and compute the optimal primal solution $\hat{y}^{tj} \equiv (\hat{x}^{tj}, \hat{p}^{t+1,j}, \hat{q}^{t+1,j}, \hat{s}^{tj})$ and dual solution $(\hat{\pi}^{x,t,j}, \hat{\pi}^{s,t,j})$. We then update the cutting plane model $\mathfrak{Q}_t(\cdot) = \max\{\mathfrak{Q}_t(\cdot), l_t(\cdot)\}$, where $l_t(y^{t-1}) := \tilde{Q}_t(\bar{y}^{t-1}) + \tilde{g}_t^T(y^{t-1} - \bar{y}^{t-1})$ with $\tilde{Q}_t(\bar{y}^{t-1}) = \frac{1}{N_t}\sum_{j=1}^{N_t}[-u(\sum_{i=1}^{n+1}\hat{x}_i^{tj}) + \mathfrak{Q}_{t+1}(\hat{x}^{tj}, \hat{p}^{t+1,j}, \hat{q}^{t+1,j}, \hat{s}^{tj})]$ and

$$(\tilde{g}_t)_i = \begin{cases} \frac{1}{N_t}\sum_{j=1}^{N_t} \tilde{R}_i^{tj}\hat{\pi}_i^{x,t,j}, & 1 \le i \le n; \\ \frac{1}{N_t}\sum_{j=1}^{N_t} \hat{\pi}_{n+1}^{x,t,j}, & i = n+1; \\ \frac{1}{N_t}\sum_{j=1}^{N_t} -R_i^{tj}\hat{\pi}_i^{x,t,j} + (1-\hat{p}_i)\hat{\pi}_{n+1}^{x,t,j}, & n+2 \le i \le 2n+1; \\ \frac{1}{N_t}\sum_{j=1}^{N_t} R_i^{tj}\hat{\pi}_i^{x,t,j} - (1+\hat{q}_i)\hat{\pi}_{n+1}^{x,t,j}, & 2n+2 \le i \le 3n+1; \\ 0, & 3n+2 \le i \le 4n+2. \end{cases}$$

We apply both DSA and SDDP to solve a few different problem instances of (4.108) with parameters given in Table 4.3. In particular, we consider two groups of instances. The first group (Inst 4, Inst 5 and Inst 6) has a fixed number of stages 3, but with different number of assets (5, 200 and 400), while the second group (Inst 4, Inst 7 and Inst 8) has the same parameter setting except that the number of stages changes from 3 to 4 or 5. Our hypothesis is that the DSA method can scale up with the dimension of the problem (i.e., the number of assets), while SDDP can handle problems with a larger number of stages.

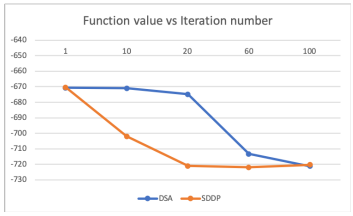
We first report the estimated function values in Figure 4.1a, 4.2a and 4.3a for the first group of instances. Note that in order to estimate the function values for a generated solution, we generate N sequences of random variables $\{\epsilon^t\}, t = 1, \ldots, T$, and com-

pute the random returns $R_j^t \in \mathbb{R}^n, \forall t = 1, \ldots, T, j = 1, \ldots, N$ according to (4.113), then we compute feasible solution by (4.111) and estimate the function value by $FV = \frac{1}{N} \sum_{j=1}^{N} \sum_{t=1}^{T-1} -u(\sum_{i=1}^{n+1} x_i^{j,t})$ with $N = 1000$. We observe from Figures 4.1a, 4.2a and 4.3a that SDDP generates slightly better solution quality, and that DSA significantly outperforms SDDP in terms of computation time for instances with a small number of stages (e.g., $T = 3$) by comparing Figure 4.1b with 4.1c, and similarly Figure 4.2b with 4.2c, and Figure 4.3b with 4.3c. Moreover, by comparing Figures 4.1b, 4.4b, and 4.5b, with 4.1c, 4.4c, and 4.5c, we can see that for problem instances with a small number of assets (i.e., Inst 4, Inst 7 and Inst 8), as the number of stages varies from 3, 4 to 5, the execution time for DSA algorithm changes from $20, 2,000$ to $190,000$ seconds in 100 iterations, while the one for SDDP only changes from $5,000$, $6,000$ to $7,000$ seconds. From these preliminary numerical results, we indeed confirm that DSA can be used to handle multi-stage stochastic optimization problems with a large number of decision (or state) variables, but a relatively smaller number of stages. On the other hand, SDDP type algorithms can be used to solve problems with a larger number of stages but smaller number of decision (or state) variables. These two types of algorithms seem to be complimentary to each other for solving multi-stage stochastic optimization problems.

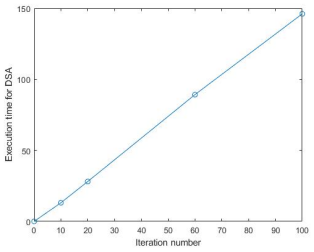


(a) Function value vs iter (b) Execution time for DSA (c) Execution time for SDDP

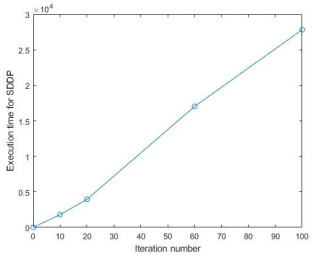
Figure 4.1: Comparison for Inst 4



(a) Function value vs iter

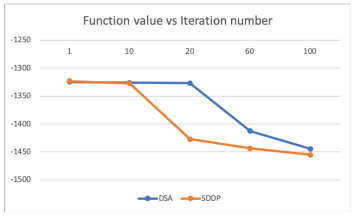


(b) Execution time for DSA

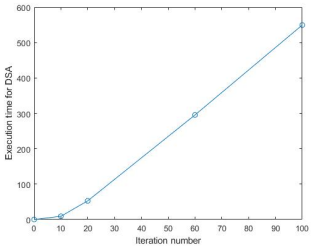


(c) Execution time for SDDP

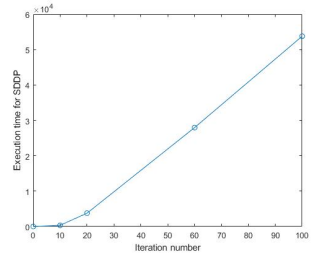
Figure 4.2: Comparison for Inst 5



(a) Function value vs iter



(b) Execution time for DSA

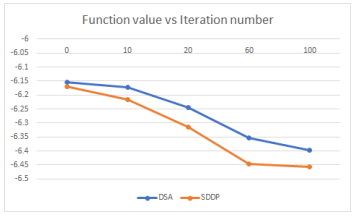


(c) Execution time for SDDP

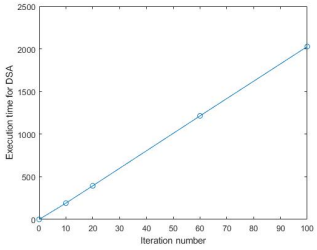
Figure 4.3: Comparison for Inst 6

4.6 Conclusion

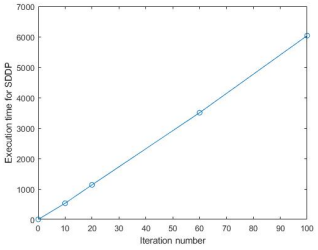
In this chapter, we present a new class of stochastic approximation algorithms, i.e., dynamic stochastic approximation (DSA), for solving multi-stage stochastic optimization problems. This algorithm is developed by reformulating the optimization problem in each stage as a saddle point problem and then recursively applying an inexact primal-dual stochastic approximation algorithm to compute an approximate stochastic subgradient of the previous stage. We establish the convergence of this algorithm by carefully bounding the bias and variance associated with these approximation errors. For a three-stage stochastic optimization problem, we show that the total number of required scenarios to find an ϵ-solution is bounded by $\mathcal{O}(1/\epsilon^4)$ and $\mathcal{O}(1/\epsilon^2)$, respectively, for general convex and strongly convex cases. These bounds are essentially not improvable in terms of their dependence on the target accuracy. We also generalize DSA for solving multi-stage stochastic optimization problems with the number of stages $T > 3$. To the best of our knowledge, this is the first



(a) Function value vs iter

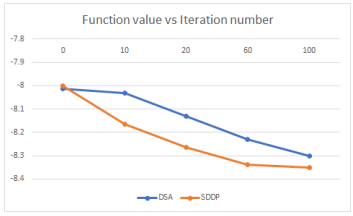


(b) Execution time for DSA

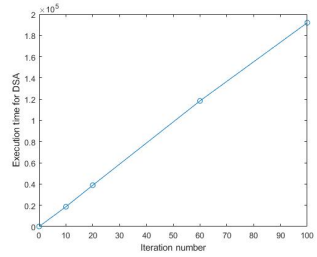


(c) Execution time for SDDP

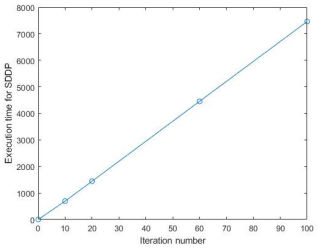
Figure 4.4: Comparison for Inst 7



(a) Function value vs iter



(b) Execution time for DSA



(c) Execution time for SDDP

Figure 4.5: Comparison for Inst 8

time that stochastic approximation methods have been developed and their complexity is established for multi-stage stochastic optimization.

From the preliminary numerical results, we can see the DSA method is efficient for solving high dimensional problems with a relatively smaller number of stages. However, as the number of stages increase, the computing time would increase exponentially even though it can handle the case when random variable are stage-wise dependent. Further improvement on the practical performance of this method should be pursed along the directions of better estimating problem parameters especially those related to the size of subgradients and dual multipliers. It would be interesting to study whether one can estimate these parameters in an online fashion while running these methods, and whether one can further improve the convergence of DSA in terms of its dependence on these problem parameters, e.g., by using accelerated SA methods and some other algorithmic schemes.

It is worth noting that there exist a class of alternative approaches based on linear decision rule models for solving multi-stage stochastic optimization problems. In these meth-

ods we assume that the decisions linearly depend on the decisions previously made and the realization of random variables that have been observed so far. Using this approach, one can reformulate a multi-stage stochastic optimization problem into a two-stage problem, and hence can significantly reduce the computational cost. In comparison with the exact methods we focus on in this chapter, using linear decision rule models can only generate suboptimal solutions for the original multi-stage stochastic optimization problems in general.

REFERENCES

[1] R. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *The Journal of Risk*, vol. 2, pp. 21–41, 2000.

[2] A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 969–996, 2006.

[3] O. Chapelle, B. Scholkopf, and A. Zien, *Semi-supervised learning*. Cambridge, Mass.: MIT Press, 2006.

[4] A. Shapiro, "Monte carlo sampling methods," in *Stochastic Programming*, A. Ruszczyński and A. Shapiro, Eds., Amsterdam: North-Holland Publishing Company, 2003.

[5] A. J. Kleywegt, A. Shapiro, and T. H. de Mello, "The sample average approximation method for stochastic discrete optimization," vol. 12, pp. 479–502, 2001.

[6] W. Wang and S. Ahmed, "Sample average approximation of expected value constrained stochastic programs," *Operations Research Letters*, vol. 36, no. 5, pp. 515–519, 2008.

[7] H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Mathematical Statistics*, vol. 22, pp. 400–407, 1951.

[8] A. Benveniste, M. Métivier, and P. Priouret, *Algorithmes adaptatifs et approximations stochastiques*. Masson, 1987, English translation: *Adaptive Algorithms and Stochastic Approximations*, Springer Verlag (1993).

[9] Y. Ermoliev, "Stochastic quasigradient methods and their application to system optimization," *Stochastics*, vol. 9, pp. 1–36, 1983.

[10] A. Gaivoronski, "Nonstationary stochastic programming problems," *Kybernetika*, vol. 4, pp. 89–92, 1978.

[11] G. Pflug, "Optimization of stochastic models," in *The Interface Between Simulation and Optimization*, Boston: Kluwer, 1996.

[12] A. Ruszczyński and W. Sysk, "A method of aggregate stochastic subgradients with on-line stepsize rules for convex stochastic programming problems," *Mathematical Programming Study*, vol. 28, pp. 113–131, 1986.

[13] J. C. Spall, *Introduction to stochastic search and optimization: estimation, simulation, and control*. John Wiley & Sons, 2005, vol. 65.

[14] B. Polyak, "New stochastic approximation type procedures," *Automat. i Telemekh.*, vol. 7, pp. 98–107, 1990.

[15] B. Polyak and A. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM J. Control and Optimization*, vol. 30, pp. 838–855, 1992.

[16] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[17] G. Lan, "An optimal method for stochastic composite optimization," *Mathematical Programming*, vol. 133(1), pp. 365–397, 2012.

[18] S. Ghadimi and G. Lan, "Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, I: A generic algorithmic framework," vol. 22, pp. 1469–1492, 2012.

[19] ——, "Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: Shrinking procedures and optimal algorithms," vol. 23, pp. 2061–2089, 2013.

[20] J. C. Duchi, S. Shalev-shwartz, Y. Singer, and A. Tewari, "Composite objective mirror descent," *In Proceedings of the Twenty Third Annual Conference on Computational Learning Theory*, 2010.

[21] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," *J. Mach. Learn. Res.*, pp. 2543–2596, 2010.

[22] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic optimization," Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA, Technical Report, June 2013.

[23] A. Nedic and S. Lee, *On stochastic subgradient mirror-descent algorithm with weighted averaging*, 2014.

[24] M. Schmidt, N. L. Roux, and F. Bach, "Minimizing finite sums with the stochastic average gradient," Tech. Rep., 2013.

[25] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, pp. 95–110, 1956.

[26] M. Jaggi, "Revisiting frank-wolfe: Projection-free sparse convex optimization," in *the 30th International Conference on Machine Learning*, 2013.

[27] M. Jaggi and M. Sulovský, "A simple algorithm for nuclear norm regularized problems," in *the 27th International Conference on Machine Learning*, 2010.

[28] G. Lan, "The complexity of large-scale convex programming under a linear optimization oracle," *arXiv preprint arXiv:1309.5550*, 2013.

[29] R. M. Freund and P. Grigas, "New Analysis and Results for the Frank-Wolfe Method," *ArXiv e-prints*, 2013. arXiv: 1307.0873.

[30] Z. Harchaoui, A. Juditsky, and A. S. Nemirovski, "Conditional gradient algorithms for machine learning," NIPS OPT Workshop, 2012.

[31] C. Guzmán and A. Nemirovski, "On lower complexity bounds for large-scale smooth convex optimization," *Journal of Complexity*, vol. 31, no. 1, pp. 1–14, 2015.

[32] G. Lan and Y. Zhou, "Conditional gradient sliding for convex optimization," Technical Report, Tech. Rep., 2014.

[33] S. Ahipasaoglu and M. Todd, "A modified frank-wolfe algorithm for computing minimum-area enclosing ellipsoidal cylinders: Theory and algorithms," *Computational Geometry*, vol. 46, pp. 494–519, 2013.

[34] F. Bach, S. Lacoste-Julien, and G. Obozinski, "On the equivalence between herding and conditional gradient algorithms," in *the 29th International Conference on Machine Learning*, 2012.

[35] A. Beck and M. Teboulle, "A conditional gradient method with linear rate of convergence for solving convex linear systems," *Math. Methods Oper. Res.*, vol. 59, pp. 235–247, 2004.

[36] K. L. Clarkson, "Coresets, sparse greedy approximation, and the frank-wolfe algorithm," *ACM Trans. Algorithms*, vol. 6, no. 4, 63:1–63:30, Sep. 2010.

[37] E. Hazan, "Sparse approximate solutions to semidefinite programs," in *LATIN 2008: Theoretical Informatics*, ser. Lecture Notes in Computer Science, E. Laber, C. Bornstein, L. Nogueira, and L. Faria, Eds., vol. 4957, Springer Berlin Heidelberg, 2008, pp. 306–316, ISBN: 978-3-540-78772-3.

[38] R. Luss and M. Teboulle, "Conditional gradient algorithms for rank one matrix approximations with a sparsity constraint," *SIAM Review*, vol. 55, pp. 65–98, 2013.

[39] A. G. S. Shalev-Shwartz and O. Shamir, "Large-scale convex minimization with a low rank constraint," in *the 28th International Conference on Machine Learning*, 2011.

[40] C. Shen, J. Kim, L. Wang, and A. van den Hengel, "Positive semidefinite metric learning using boosting-like algorithms," *Journal of Machine Learning Research*, vol. 13, pp. 1007–1036, 2012.

[41] B. Jiang, T. Lin, S. Ma, and S. Zhang, "Structured nonconvex and nonsmooth optimization: Algorithms and iteration complexity analysis," *Computational Optimization and Applications*, vol. 72, no. 1, pp. 115–157, 2019.

[42] G. Braun, S. Pokutta, and D. Zink, "Lazifying conditional gradient algorithms.," in *ICML*, 2017, pp. 566–575.

[43] M. L. Gonçalves and J. G. Melo, "A newton conditional gradient method for constrained nonlinear systems," *Journal of Computational and Applied Mathematics*, vol. 311, pp. 473–483, 2017.

[44] G. Lan, *First-order and Stochastic Optimization Methods for Machine Learning*. Switzerland AG: Springer Nature, 2020.

[45] R. Drzymala, R. Mohan, L Brewster, J Chu, M. Goitein, W Harms, and M Urie, "Dose-volume histograms," *International Journal of Radiation Oncology* Biology* Physics*, vol. 21, no. 1, pp. 71–78, 1991.

[46] P. Mayles, A. Nahum, and J.-C. Rosenwald, *Handbook of radiotherapy physics: theory and practice*. CRC Press, 2007.

[47] H. E. Romeijn, R. K. Ahuja, J. F. Dempsey, and A. Kumar, "A column generation approach to radiation therapy treatment planning using aperture modulation," *SIAM Journal on Optimization*, vol. 15, no. 3, pp. 838–862, 2005.

[48] M. Goitein, *Radiation oncology: a physicist's-eye view*. Springer Science & Business Media, 2007.

[49] C. Men, X. Gu, D. Choi, A. Majumdar, Z. Zheng, K. Mueller, and S. B. Jiang, "Gpu-based ultrafast imrt plan optimization," *Physics in Medicine & Biology*, vol. 54, no. 21, p. 6565, 2009.

[50] C. Men, H. E. Romeijn, X. Jia, and S. B. Jiang, "Ultrafast treatment plan optimization for volumetric modulated arc therapy (vmat)," *Medical physics*, vol. 37, no. 11, pp. 5787–5791, 2010.

[51] C. Men, H. E. Romeijn, Z. C. Taşkın, and J. F. Dempsey, "An exact approach to direct aperture optimization in imrt treatment planning," *Physics in Medicine & Biology*, vol. 52, no. 24, p. 7333, 2007.

[52] J. Birge and F. Louveaux, *Introduction to Stochastic Programming*. New York: Springer, 1997.

[53] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory*. Philadelphia: SIAM, 2009.

[54] A. Shapiro and A. Nemirovski, "On complexity of stochastic programming problems," *E-print available at: http://www.optimization-online.org*, 2004.

[55] A. Shapiro, "On complexity of multistage stochastic programs," *Operations Research Letters*, vol. 34, pp. 1–8, 2006.

[56] M. V. Pereira and L. M. Pinto, "Multi-stage stochastic optimization applied to energy planning," *Mathematical programming*, vol. 52, no. 1-3, pp. 359–375, 1991.

[57] A. Shapiro, "Analysis of stochastic dual dynamic programming method," *European Journal of Operational Research*, vol. 209, pp. 63–72, 2011.

[58] A. Philpott, V. d. Matos, and E. Finardi, "On solving multistage stochastic programs with coherent risk measures," *Operations Research*, vol. 61, pp. 957–970, 2013.

[59] C. J. Donohue and J. R. Birge, "The abridged nested decomposition method for multistage stochastic linear programs with relatively complete recourse," *Algorithmic Operations Research*, vol. 1, no. 1, 2006.

[60] M. Hindsberger and A. Philpott, "Resa: A method for solving multistage stochastic linear programs," *Journal of Applied Operational Research*, vol. 6, no. 1, pp. 2–15, 2014.

[61] V. Kozmík and D. P. Morton, "Evaluating policies in risk-averse multi-stage stochastic programming," *Mathematical Programming*, vol. 152, no. 1-2, pp. 275–300, 2015.

[62] R. T. Rockafellar and R. J.-B. Wets, "Scenarios and policy aggregation in optimization under uncertainty," *Mathematics of operations research*, vol. 16, no. 1, pp. 119–147, 1991.

[63] J. Higle and S. Sen, "Stochastic decomposition: An algorithm for two-stage linear programs with recourse," *Mathematics of Operations Research*, vol. 16, pp. 650–669, 1991.

[64] A. S. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," vol. 19, pp. 1574–1609, 2009.

[65] G. Lan, A. S. Nemirovski, and A. Shapiro, "Validation analysis of mirror descent stochastic approximation method," *Mathematical Programming*, vol. 134(2), pp. 425–458, 2012.

[66] A. S. Nemirovski and D. Yudin, *Problem complexity and method efficiency in optimization*, ser. Wiley-Interscience Series in Discrete Mathematics. John Wiley, XV, 1983.

[67] S. Ghadimi and G. Lan, "Stochastic first- and zeroth-order methods for nonconvex stochastic programming," vol. 23(4), pp. 2341–2368, 2013.

[68] A. Nedić, "On stochastic subgradient mirror-descent algorithm with weighted averaging," 2012.

[69] S. Ghadimi, G. Lan, and H. Zhang, "Mini-batch stochastic approximation methods for constrained nonconvex stochastic programming," vol. 155, pp. 267–305, 2016.

[70] M. Wang, E. X. Fang, and H. Liu, "Stochastic compositional gradient descent: Algorithms for minimizing compositions of expected-value functions," *Mathematical Programming*, 2016.

[71] X. Wang, S. Ma, D. GOLDFARB, and W. Liu, "Stochastic quasi-newton methods for nonconvex stochastic optimization," *SIAM Journal on Optimization*, vol. 27, pp. 235–247, 2017.

[72] B. Dai, N. He, Y. Pan, B. Boots, and L. Song, "Learning from conditional distributions via dual embeddings," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1458–1467, 2017.

[73] Y. E. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$," *Doklady AN SSSR*, vol. 269, pp. 543–547, 1983.

[74] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," vol. 156, pp. 59–99, 2016.

[75] B. T. Polyak, "A general method of solving extremum problems," *Doklady Akademii Nauk SSSR*, vol. 174, no. 1, p. 33, 1967.

[76] Y. E. Nesterov, *Introductory Lectures on Convex Optimization: a basic course*. Massachusetts: Kluwer Academic Publishers, 2004.

[77] Y. Chen, G. Lan, and Y. Ouyang, "Optimal primal-dual methods for a class of saddle point problems," vol. 24(4), pp. 1779–1814, 2014.

[78] G. Lan, Z. Lu, and R. D. C. Monteiro, "Primal-dual first-order methods with $\mathcal{O}(1/\epsilon)$ iteration-complexity for cone programming," vol. 126, pp. 1–29, 2011.

[79] H. Jiang and U. V. Shanbhag, "On the solution of stochastic optimization and variational problems in imperfect information regimes," *arXiv preprint arXiv:1402.1457*, 2014.

[80] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR computational mathematics and mathematical physics*, vol. 7, no. 3, pp. 200–217, 1967.

[81] A. Auslender and M. Teboulle, "Interior gradient and proximal methods for convex and conic optimization," *SIAM Journal on Optimization*, vol. 16, no. 3, pp. 697–725, 2006.

[82] H. Bauschke, J. Borwein, and P. Combettes, "Bregman monotone optimization algorithms," *SIAM Journal on Controal and Optimization*, vol. 42, pp. 596–636, 2003.

[83] M. Teboulle, "Convergence of proximal-like algorithms," vol. 7, pp. 1069–1083, 1997.

[84] A. Beck, A. Ben-Tal, N. Guttmann-Beck, and L. Tetruashvili, "The comirror algorithm for solving nonsmooth constrained convex problems," *Operations Research Letters*, vol. 38, no. 6, pp. 493–498, 2010.

[85] D. Goldfarb and G. Iyengar, "Robust portfolio selection problems," *Mathematics of Operations Research*, vol. 28, no. 1, pp. 1–38, 2003.

[86] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal estimated sub-gradient solver for svm," in *In ICML*, 2007, pp. 807–814.

[87] J. C. Duchi, P. L. Bartlett, and M. J. Wainwright, "Randomized smoothing for stochastic optimization," vol. 22, pp. 674–701, 2012.

[88] D. Boob, Q. Deng, and G. Lan, "Stochastic first-order methods for convex and nonconvex function constrained optimization," *arXiv*, 2019, 1908.02734.

[89] Y. E. Nesterov, "Smooth minimization of nonsmooth functions," *Mathematical Programming*, vol. 103, pp. 127–152, 2005.

[90] A. S. Nemirovski, "Prox-method with rate of convergence $o(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems," vol. 15, pp. 229–251, 2005.

[91] E. Y. Hamedani and N. S. Aybat, "A primal-dual algorithm for general convex-concave saddle point problems," *arXiv preprint arXiv:1803.01401*, 2018.

[92] A. Shapiro, D. Dentcheva, and A. Ruszczyński.

[93] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ: Princeton University Press, 1970.

[94] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imaging Vision*, vol. 40, pp. 120–145, 2011.

[95] G. Lan and Y. Zhou, "An optimal randomized incremental gradient method," *Mathematical programming*,

[96] K. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Non-linear Programming*, ser. Stanford Mathematical Studies in the Social Sciences. Stanford University Press, 1958.

[97] R. Monteiro and B. Svaiter, "On the complexity of the hybrid proximal projection method for the iterates and the ergodic mean," vol. 20, pp. 2755–2787, 2010.

[98] B. He and X. Yuan, "On the o(1/n) convergence rate of the douglas-rachford alternating direction method," *SIAM Journal on Numerical Analysis*, vol. 50, no. 2, pp. 700–709, 2012.

[99] Y. Ouyang, Y. Chen, G. Lan, and E. Pasiliao, "An accelerated linearized alternating direction method of multipliers," *SIAM Journal on Imaging Sciences*, 2014, to appear.

[100] M. Rudelson and R. Vershynin, *Non-asymptotic theory of random matrices: Extreme singular values*, 2010.

[101] G. Lan, "Bundle-level type methods uniformly optimal for smooth and non-smooth convex optimization," vol. 149(1), 1–45, 2015.

[102] C. S. Pedersen and S. E. Satchell, "Utility functions whose parameters depend on initial wealth," *Bulletin of Economic Research*, vol. 55, no. 4, pp. 357–371, 2003.