

RISK NEUTRAL AND RISK AVERSE STOCHASTIC OPTIMIZATION

A Dissertation
Presented to
The Academic Faculty

By

Yi Cheng

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

Georgia Institute of Technology

December 2022

© Yi Cheng 2022

RISK NEUTRAL AND RISK AVERSE STOCHASTIC OPTIMIZATION

Thesis committee:

Dr. Alexander Shapiro
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Lauren N. Steimle
School of Industrial and Systems Engineering
Georgia Institute of Technology

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

Dr. Enlu Zhou
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Alp Muharremoglu

Amazon.com

Date approved: November 23, 2022

It was the best of times, it was the worst of times.

Charles Dickens

To my parents

ACKNOWLEDGMENTS

Throughout the years at Georgia Tech, I have received a lot of help, support and enlightenment from many people. In the paragraphs that follow, I would like to point out a few of them that I am particularly indebted to.

First and foremost, I want to express my deepest gratitude to my advisors Alexander Shapiro and Guanghui (George) Lan, who “push” me to the critical point of intellectual drive and ignite the engine for my academic and personal edification. The first conversation with Alex was about human history, scattered with universe evolution. I found myself inadvertently keep asking questions (may be quite stupid as I barely knew those), leaving him more constraints to stay on the seat for a long time. He did not complain. On the contrary, he was so patient and generously gave his understandings and answers in an amiable tone. Later, Alex introduced to me the research area of “Stochastic Programming” where I spot great interest and determined to unfold my Ph.D. journey with exploring its related topics. During the years of study, I enjoyed those “randomized” meetings with Alex (during the pandemic, switched on by the musical Skype notification sound), where he often inspired me to test new ideas, encouraged me to envision broader horizons and gave constructive feedback. I have been so fortunate to be advised by him and nourished from his insights and critical thoughts. The greatest lessons I have learned from George are relentless pursuit of high-quality research and insistence on qualified details. The first few meetings with George were often fused together discussion of first order methods that were of “highest order of confusion” (to me), my sudden clicked with the problems after his hints and his wholehearted encouragement at the end of the meetings. Under his guidance, I have overcome a lot of obstacles and gradually established a systematic way of conducting research and presenting results. I am so grateful that I can learn from George, a role model that endlessly fuels me to fight for higher standards and continuous progress. I would like to thank the rest of my committee members, Alp Muharremoglu, Lauren Steimle and

Enlu Zhou for always being very supportive to me and providing insightful and constructive comments to my thesis. I would like to thank my collaborators, Edwin Romeijn and Vincent Guigues. Edwin introduced me the challenging healthcare problem which motivated me to develop the research in projection-free methods and provided a new perspective to evaluate related results. Vincent is a very pragmatic and rigorous researcher who has unique and interesting ideas in the area of stochastic programming.

I would like to thank Jiachen Shi, Hairong Wang and Buse Eylul Oruc Aglar for their friendship. They keep encouraging me whenever there are ups or downs and accompany me for the best and worst of the times. There are many other friends that I owe gratitude to. The list includes Jimmy Zhang, Lingquan Ding, Georgios Kotsalis, Ray Liu, Zhiqiang Zhou, Di Wu, Shixuan Zhang, who are always offering their help in research and job seeking, providing me with valuable perspectives of life and beyond. I would like to thank Akane Fujimoto, Prakirt Jhunhunwala, Sajad Khodadadian, Chungjae Lee, Yan Li, Shancong Mou, Cindy Azuero Pedraza, Yuyang Shi, Zilong Wang, Shaowu Yuchi, Keyu Zhu and other fellow students in the cohort that shared the joys (and of course sorrows) in the days and nights at ISyE Main 341. I would like to thank Judy, Yihan, April, Kim, Yongxin and Haotian who have been very helpful during my times in Seattle. I would like to thank Yiguo, Ke, Ci, Zaiwei, Chenghao and Scott, who add to my experience a museum-like and dishes-and-champagne-style layer (thanks to many get-together dinners, trips with them), that weave in both lasting laughter and lucid thoughts. Never would I forget my friends Peilin, Wenhao and Yunru, who have been standing by me for more than ten years.

Last but not least, my greatest gratitude goes to my family. I am deeply indebted to my partner Yijiang, for his trust, respect and unconditional love. He is like starry nights that accompany me through the journey. "It's amazing how he can speak right to my heart. Without saying a word, he can light up the dark." I would like to thank my parents, who show me the greatest virtues of mankind, help me comprehend the creed of freedom and shape my independence to live, strong will to thrive.

TABLE OF CONTENTS

Acknowledgments	v
List of Tables	xi
List of Figures	xiv
Summary	xv
Chapter 1: Introduction	1
1.1 Multistage Stochastic Program	1
1.1.1 Risk Neutral Stochastic Program	1
1.1.2 Risk Averse Stochastic Program	4
1.2 Computational Methods for Solving Stochastic Program	8
1.2.1 Stochastic Dual Dynamic Programming (SDDP)	8
1.2.2 Projection-free Methods for Risk Averse Single-stage Problem	12
Chapter 2: Upper Bounds for Risk Neutral Multistage Stochastic Program	15
2.1 Overview	15
2.2 Dual Bounds for Finite-horizon Multistage Stochastic Program	16
2.2.1 Dynamic Programming Equations of the Dual	17
2.2.2 Dynamics of Lagrangian Multipliers	21

2.2.3	Sensitivity Analysis	23
2.3	Dual SDDP	24
2.3.1	Dual SDDP for Problems with Uncertainty in b_t and B_t	24
2.3.2	Dual SDDP for Problems with Uncertainty in All Parameters	29
2.3.3	Dual SDDP for Problems with Interstage Dependent Cost Coefficients	30
2.4	Dual Bounds for Periodical Multistage Stochastic Program	32
2.5	Periodical Dual SDDP	37
2.5.1	Trial Points Selection	39
2.5.2	Cutting Plane Algorithm	39
2.5.3	Trust-bound Strategy	40
2.6	Numerical Results	41
2.6.1	Inventory Model	41
2.6.2	Hydro-thermal Generation Problem	52
2.7	Proofs of Auxiliary Results	62
2.7.1	Properties of the Multiplicative Autoregressive Process.	62
2.7.2	Proof of Lemma 2.3.1.	63
2.7.3	Proof of Proposition 2.3.1.	65
2.7.4	Proof of Theorem 2.3.1	69
Chapter 3: Upper Bounds for Risk Averse Multistage Stochastic Program		70
3.1	Overview	70
3.2	Risk-neutral Stochastic Optimal Control	72
3.3	Risk-averse Stochastic Optimal Control	78

3.3.1	Risk-averse Setting	78
3.3.2	Statistical Upper Bounds on the Value of the Policy	80
3.3.3	Q -factor Approach	83
3.4	Numerical Experiments	85
Chapter 4: Sample Complexity of Stationary Stochastic Programs		90
4.1	Overview	90
4.2	Stationary Stochastic Programs	91
4.3	Sample Complexity Analysis	92
4.3.1	Risk Averse Case	96
4.4	Inventory Model	97
4.4.1	Risk Averse Case	99
4.5	Numerical Illustration	100
4.5.1	Test Cases and Experimental Settings	100
4.5.2	Risk Neutral Case	102
4.5.3	Risk Averse Case	104
Chapter 5: Projection-free Methods for Convex Functional Constrained Opti- mization		112
5.1	Overview	112
5.2	Level Conditional Gradient Method	117
5.3	Outer Loop of LCG	118
5.4	Conditional Gradient Oracle	122
5.4.1	CGO for Smooth Functions	123

5.4.2	CGO for Structured Nonsmooth Functions	132
5.5	Overall Complexity	142
5.6	Modified Level Conditional Gradient Method	147
5.6.1	Outer Loop Iteration Complexity of MLCG	149
5.7	Auxiliary Lemmas	151
Chapter 6:	Projection-free Methods for Nonconvex Functional Constrained Op-	
	timization	153
6.1	Overview	153
6.2	Nonconvex Functional Constrained Optimization Problem	154
6.3	Proximal Point Methods for Nonconvex Functional Constrained Problem	155
6.3.1	Exact Proximal Point Method	155
6.3.2	Inexact Proximal Point Method	158
6.4	Direct Nonconvex Conditional Gradients Method	163
6.5	Numerical Experiments	169
6.5.1	Portfolio Selection	169
6.5.2	IMRT Treatment Planning	176
References	187

LIST OF TABLES

2.1	Lower bound Lb and upper bound Ub computed by Primal SDDP and upper bounds computed by Dual SDDP penalties $v_{t,k} = 1000$ along iterations. . .	44
2.2	Time needed (in seconds) to obtain a solution of relative accuracy ε with Primal SDDP and Dual SDDP with penalties $v_{t,k} = 1000$	44
2.3	Convergence results of MC-primal and TS-dual	46
2.4	Upper and lower bounds at the last iteration of Primal SDDP.	48
2.5	Comparison between optimal Lagrange multipliers from Primal SDDP and Dual SDDP with penalties.	49
2.6	Sensitivity of the optimal value with respect to ϕ and μ by the two methods.	49
2.7	Inventory problem: evolution of bounds of primal and dual periodical programs.	51
2.8	Lower bound Lb and upper bound Ub computed by Primal SDDP and upper bounds computed by variants of Dual SDDP along iterations. All costs have been divided by 10^6	57
2.9	Relative error as a function of the number of iterations for Primal SDDP, Dual SDDP 1, and Dual SDDP 2.	57
2.10	CPU time (in seconds) needed to obtain a solution of relative accuracy ε with Primal SDDP and variants of Dual SDDP.	59
2.11	Hydro-thermal problem with $\gamma = 0.8$: deterministic bounds of primal and dual periodical programs.	60
2.12	Hydro-thermal problem with $\gamma = 0.9906$: deterministic bounds of primal and dual periodical programs.	62

3.1	Convergence of convex combination of expectation and AV@R problem for different λ	88
3.2	Convergence of KL-divergence problem for different ϵ	89
4.1	Risk neutral case: convergence of solving SAA problems.	103
4.2	Risk neutral case: sample standard deviations of optimal values of $M = 100$ SAA problems.	105
4.3	Risk neutral case: theoretical standard deviation of the optimal value function for the inventory problem.	106
4.4	Risk averse case: convergence of solving SAA problems.	108
4.5	Risk averse case: sample standard deviations of optimal values of $M = 100$ SAA problems.	110
4.6	Risk averse case: theoretical standard deviation of the optimal value function for the inventory problem.	110
6.1	Features of the stock market dataset.	173
6.2	Results of solving model (Card-Free-Convex) by LCG.	174
6.3	Results of solving model (Card-Free-Nonconvex) by DNCG and IPP-LCG.	174
6.4	Results of solving model (Card-Convex) by LCG.	175
6.5	Results of solving model (Card-Nonconvex-1) by IPP-LCG.	175
6.6	Results of solving model (Card-Nonconvex-2) by DNCG.	175
6.7	Results of solving model (Card-Convex) by CoexDurCG.	176
6.8	Features of the synthetic dataset	181
6.9	Results of applying LCG on the synthetic dataset at iteration 1000.	182
6.10	Results of applying CoexDurCG on synthetic data with $\Phi = 0.005$ at iteration 1000.	183
6.11	Treatment plans constructed by LCG on Prostate dataset with different Φ	184

6.12 Results of applying LCG on Prostate dataset.	185
6.13 Treatment plans constructed by DNCG with different initial conditions on Prostate dataset.	186
6.14 Results of applying DNCG and LCG initial+ DNCG on Prostate dataset.	186

LIST OF FIGURES

2.1	Graph of V_2 and of V_2^γ for $\gamma = 1, 100, 1000$	28
2.2	Cumulative CPU time along iterations of Primal SDDP and Dual SDDP with penalizations $v_{t,k} = 1000$	45
2.3	Evolution of primal and dual bounds for interstage dependent cost process .	46
2.4	Top left: upper and lower bounds computed by Primal SDDP and upper bounds computed by Dual SDDP 1, Dual SDDP 2, and Dual SDDP 3, for the first 20 iterations; Top right: same outputs for iterations 21, . . . , 150; Bottom: same outputs for iterations 151, . . . , 1000.	56
2.5	Cumulative CPU time for Primal SDDP, Dual SDDP 1, Dual SDDP 2, and Dual SDDP 3.	58
2.6	Hydro-thermal problem with $\gamma = 0.9906$: evolution of deterministic bounds of primal and dual periodical multistage stochastic programs. The orange line is obtained by smoothing the dual bounds (in blue) to exhibit the descending trend.	61
3.1	Evolution of lower and upper bounds for convex combination of expectation and AV@R problem when $\lambda = 0.5$	88
4.1	Normal probability plot (Q-Q plot) for the risk neutral hydro-thermal problem	107
4.2	Risk averse case: bounds evolution of the SAA problems of the inventory problem.	109
4.3	Normal probability plot (Q-Q plot) for the risk averse hydro-thermal problem	111

SUMMARY

In this thesis, we focus on the modeling, computational methods and applications of multistage/single-stage stochastic optimization, which entail risk aversion under certain circumstances. Chapters 2-4 concentrate on multistage stochastic programming while Chapter 5-6 deal with a class of single-stage functional constrained stochastic optimization problems.

First, we investigate the deterministic upper bound of a Multistage Stochastic Linear Program (MSLP). We first present the Dual SDDP algorithm, which solves the Dynamic Programming equations for the dual and computes a sequence of nonincreasing deterministic upper bounds for the optimal value of the problem, even without the presence of Relatively Complete Recourse (RCR) condition. We show that optimal dual solutions can be obtained using Primal SDDP when computing the duals of the subproblems in the backward pass. As a byproduct, we study the sensitivity of the optimal value as a function of the involved problem parameters. In particular, we provide formulas for the derivatives of the value function with respect to the parameters and illustrate their application on an inventory problem. Next, we extend to the infinite-horizon MSLP and show how to construct a deterministic upper bound (dual bound) via the proposed Periodical Dual SDDP. Finally, as a proof of concept of the developed tools, we present the numerical results of (1) the sensitivity of the optimal value as a function of the demand process parameters; (2) conduct Dual SDDP on the inventory and the Brazilian hydro-thermal planning problems under both finite-horizon and infinite-horizon settings.

Second, we propose a construction of the statistical upper bound for the optimal value of risk-averse Stochastic Optimal Control (SOC) problems. This outlines an approach to a solution of a long standing problem in that area of research. The bound holds for a large class of convex and monotone conditional risk mappings. We show the validity of the statistical upper bound to solve a real-world stochastic hydro-thermal planning problem.

Third, we discuss sample complexity of solving stationary stochastic programs by the Sample Average Approximation (SAA) method. We investigate this in the framework of Stochastic Optimal Control (in discrete time) setting. In particular we derive a Central Limit Theorem type asymptotics for the optimal values of the SAA problems. The main conclusion is that the sample size, required to attain a given relative error of the SAA solution, is not sensitive to the discount factor, even if the discount factor is very close to one. We consider the risk neutral and risk averse settings. The presented numerical experiments confirm the theoretical analysis.

Fourth, we propose a novel projection-free method, referred to as Level Conditional Gradient (LCG) method, for solving convex functional constrained optimization. Different from the constraint-extrapolated conditional gradient type methods (CoexCG and CoexDurCG), LCG, as a primal method, does not assume the existence of an optimal dual solution, thus improving the convergence rate of CoexCG/CoexDurCG by eliminating the dependence on the magnitude of the optimal dual solution. Similar to existing level-set methods, LCG uses an approximate Newton method to solve a root-finding problem. In each approximate Newton update, LCG calls a conditional gradient oracle (CGO) to solve a saddle point subproblem. The CGO developed herein employs easily computable lower and upper bounds on these saddle point problems. We establish the iteration complexity of the CGO for solving a general class of saddle point optimization. Using these results, we show that the overall iteration complexity of the proposed LCG method is $\mathcal{O}\left(\frac{1}{\epsilon^2} \log\left(\frac{1}{\epsilon}\right)\right)$ for finding an ϵ -optimal and ϵ -feasible solution of the considered problem. To the best of our knowledge, LCG is the first primal conditional gradient method for solving convex functional constrained optimization. For the subsequently developed nonconvex algorithms in this thesis, LCG can also serve as a subroutine or provide high-quality starting points that expedites the solution process.

Last, to cope with the nonconvex functional constrained optimization problems, we develop three approaches: the Level Exact Proximal Point (EPP-LCG) method, the Level

Inexact Proximal Point (IPP-LCG) method and the Direct Nonconvex Conditional Gradient (DNCG) method. The proposed EPP-LCG and IPP-LCG methods utilize the proximal point framework and solve a series of convex subproblems. By solving each subproblem, they leverage the proposed LCG method, thus averting the effect from large Lagrangian multipliers. We show that the iteration complexity of the algorithms is bounded by $\mathcal{O}\left(\frac{1}{\epsilon^3} \log\left(\frac{1}{\epsilon}\right)\right)$ in order to obtain an (approximate) KKT point. However, the proximal-point type methods have triple-layer structure and may not be easily implementable. To alleviate the issue, we also propose the DNCG method, which is the first single-loop projection-free algorithm for solving nonconvex functional constrained problem in the literature. This algorithm provides a drastically simpler framework as it only contains three updates in one loop. We show that the iteration complexity to find an ϵ -Wolfe point is bounded by $\mathcal{O}(1/\epsilon^4)$. To the best of our knowledge, all these developments are new for projection-free methods for nonconvex optimization. We demonstrate the effectiveness of the proposed nonconvex projection-free methods on a portfolio selection problem and the intensity modulated radiation therapy treatment planning problem. Moreover, we compare the results with the LCG method proposed in Chapter 6. The outcome of the numerical study shows all methods are efficient in jointly minimizing risk while promoting sparsity in a rather short computational time for the real-world and large-scale datasets.

Some of the contents of the thesis can be found in [1, 2, 3, 4, 5].

CHAPTER 1

INTRODUCTION

In this chapter, we introduce the background and discuss the motivation for the thesis. Specifically, we present the mathematical formulations for multistage stochastic programs under various settings and extend our discussion to the risk averse case for both single-stage and multistage models in Section 1.1. In Section 1.2, we review existing methodologies and challenges for solving both multistage and single-stage programs.

1.1 Multistage Stochastic Program

Multistage stochastic program (MSP) provides a framework for making decisions under uncertainty where the decision space is typically high dimensional and the uncertainty is modeled by general stochastic processes.

1.1.1 Risk Neutral Stochastic Program

We start with the mathematical formulation of a multistage stochastic linear program with T stages:

$$\begin{aligned} \min_{x_t \in \mathcal{X}_t} \quad & \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} c_t^\top x_t \right] \\ \text{s.t.} \quad & A_1 x_1 = b_1, \\ & B_t x_{t-1} + A_t x_t = b_t, \quad t = 2, \dots, T. \end{aligned} \tag{1.1}$$

Here vectors $c_t = c_t(\xi_t) \in \mathbb{R}^{n_t}$, $b_t = b_t(\xi_t) \in \mathbb{R}^{m_t}$ and matrices $B_t = B_t(\xi_t)$, $A_t = A_t(\xi_t)$ are functions of random process $\xi_t \in \mathbb{R}^{d_t}$, $t = 1, \dots, T$, and $\gamma \in (0, 1)$ is the discount factor. We denote by $\xi_{[t]} = (\xi_1, \dots, \xi_t)$ the history of the data process up to time t and by $\mathbb{E}_{|\xi_{[t]}}$ the corresponding conditional expectation. The optimization in (1.1) is performed over functions (policies) $x_t = x_t(\xi_{[t]}) \in \mathbb{R}^{n_t}$, $t = 1, \dots, T$, of the data process satisfying

the feasibility constraints. Vector ξ_1 and the first stage solution x_1 are deterministic, i.e., the first stage decision is made before knowing (observing) realizations of the data process ξ_2, \dots, ξ_T .

We review two cases regarding problem (1.1) where T is finite and infinite. In particular, when $T = 1$, (1.1) becomes a single-stage stochastic model. For multistage stochastic program under both settings, we make the following assumptions throughout the thesis unless stated otherwise.

(A0) We assume that the random data process $\{\xi_t\}$ is stagewise independent and that the probability distribution of $\xi_t, t = 2, \dots, T$ do not depend on our decisions.

(A1) (Relatively Complete Recourse) For every $x_{t-1} \geq 0$ the set $\{x_t : B_t(\xi_t)x_{t-1} + A_t(\xi_t)x_t = b_t(\xi_t), x_t \geq 0\}$ is nonempty for all $\xi_t \in \Xi_t$ and $t \geq 2$.

(A2) (i) There exist bounded sets $\mathcal{X}_t \subset \mathbb{R}^{n_t}$ such that adding the constraints $x_t \in \mathcal{X}_t, t = 1, \dots, T$, to the problem (1.1) does not change its optimal value; (ii) the cost functions $c_t : \Xi_t \rightarrow \mathbb{R}, t = 1, \dots, T$, are bounded.

In applications the sets \mathcal{X}_t typically are sufficiently large boxes containing the considered decision variables. Of course, if functions $c_t(\cdot)$ are constants, the boundedness condition (A2)(ii) holds automatically.

Problem (1.1) leads to optimization over implementable policy. Under stagewise dependence assumption on the data process $\{\xi_t\}$ and some regularity condition (to justify the interchange for the expectation and the infimum/minimum), problem (1.1) is equivalent to the following nested formulation:

$$\min_{x_1 \in \bar{\mathcal{X}}_1} c_1^\top x_1 + \mathbb{E} \left[\inf_{x_2 \in \bar{\mathcal{X}}_2} \gamma c_2^\top x_2 + \mathbb{E} \left[\dots + \mathbb{E} \left[\inf_{x_T \in \bar{\mathcal{X}}_T} \gamma^{T-1} c_T^\top x_T \right] \right] \right], \quad (1.2)$$

where $\bar{\mathcal{X}}_1 := \mathcal{X}_1 \cup \{x_1 : B_1 x_1 = b_1\}$. When the number T of stages is finite, based on problem (2.2), it is possible to write the following dynamic programming equations (cf.,

[6, Remark 3, Chapter 3.1.1]). At stage $t = T, \dots, 2$, the value function $Q_t(x_{t-1}, \xi_t)$ is given by the optimal value of the problem

$$\begin{aligned} \min_{x_t \in \mathcal{X}_t} \quad & c_t^\top x_t + \gamma Q_{t+1}(x_t) \\ \text{s.t.} \quad & B_t x_{t-1} + A_t x_t = b_t, \end{aligned} \tag{1.3}$$

with

$$Q_{t+1}(x_t) = \mathbb{E}[Q_{t+1}(x_t, \xi_{t+1})] \tag{1.4}$$

and $Q_{T+1}(\cdot) \equiv 0$. At the first stage the following problem should be solved

$$\begin{aligned} \min_{x_1 \in \mathcal{X}_1} \quad & c_1^\top x_1 + \gamma Q_2(x_1) \\ \text{s.t.} \quad & A_1 x_1 = b_1. \end{aligned} \tag{1.5}$$

When $T = 1$, the multistage formulation regarding (1.1) collapses to a single-stage model, which can be written, in a more generic way, as

$$\begin{aligned} \min_{x \in X} \quad & f(x) := \mathbb{E}[F(x, \xi)] \\ \text{s.t.} \quad & h_i(x) \leq 0, \quad i = 1, \dots, m. \end{aligned} \tag{1.6}$$

Here $f : X \rightarrow \mathbb{R}$ is proper lower semicontinuous function (not necessarily convex), $h := (h_1; \dots; h_m)$, $h_i : X \rightarrow \mathbb{R}, i = 1, \dots, m$ are proper lower semicontinuous and convex functions, $X \subseteq \mathbb{R}^n$ is a nonempty compact convex set. We call problem (1.6) either convex or nonconvex functional constrained optimization depending on whether f is convex or not. For the convex case, the objective function f is not necessarily differentiable. On the other hand, we assume f to be a differentiable function with Lipschitz continuous gradients for the nonconvex setting. Throughout the thesis, we assume that the distribution of the random variable ξ for the single-stage model (1.6) is discrete with finite number of realizations. In this way, the corresponding model can be transformed into a deterministic one and we focus our attention on developing efficient algorithms to solve such model.

1.1.2 Risk Averse Stochastic Program

So far we have introduced stochastic optimization problems that optimize over the *expected value*, that is, risk-neutral optimization problem. One downside of the approach is that, when it comes with high fluctuations of specific realizations of the stochastic process, it fails to manage the heavy quantiles, also interpreted as risk. This drives us to develop optimization models and methods that consider risk aversion.

Risk Measures. Let (Ω, \mathcal{F}, P) be a probability space and let \mathcal{Z} be a linear space of \mathcal{F} -measurable functions (random variables) $Z : \Omega \rightarrow \mathbb{R}$. A risk measure is a function $\mathcal{R} : \mathcal{Z} \rightarrow \mathbb{R}$ which assigns to a random variable Z a real number representing its risk. Typical example of the linear space \mathcal{Z} is the space of random variables with finite p -th order moments, denoted $L_p(\Omega, \mathcal{F}, P)$, $p \in [1, \infty)$. It is said that risk measure \mathcal{R} is *convex* if it possesses the properties of convexity, monotonicity, and translation equivariance. If moreover it is positively homogeneous, then it is said that risk measure \mathcal{R} is *coherent* (coherent risk measures were introduced in [7]). We can refer to [8] and [9] for a thorough discussion of risk measures.

In this thesis we consider a class of convex risk measures which can be represented in the following parametric form:

$$\mathcal{R}(Z) = \inf_{\theta \in \Theta} \mathbb{E}_P[\Psi(Z, \theta)], \quad (1.7)$$

where $\Theta \subset \mathbb{R}^k$ and $\Psi : \mathbb{R} \times \Theta \rightarrow \mathbb{R}$ is a real valued function. The notation \mathbb{E}_P in (1.7) emphasizes that the expectation is taken with respect to the probability measure (distribution) P of random variable Z . We consider risk measures of the form (1.7) for every time period. That is, for every $t = 1, \dots, T$, we consider a probability space $(\Omega_t, \mathcal{F}_t, P_t)$, and risk

measure¹

$$\mathcal{R}_t(Z_t) = \inf_{\theta_t \in \Theta} \mathbb{E}_{P_t}[\Psi(Z_t, \theta_t)], \quad Z_t \in \mathcal{Z}_t, \quad (1.8)$$

defined on the respective linear space of random variables, say $\mathcal{Z}_t := L_p(\Omega_t, \mathcal{F}_t, P_t)$.

We make the following assumptions.

- (C) (i) The set Θ is a nonempty closed convex. (ii) For every $Z_t \in \mathcal{Z}_t, t = 1, \dots, T$, the expectation in the right hand side of (1.8) is well defined and the infimum is finite valued. (iii) The function $\Psi(z, \theta)$ is convex in $(z, \theta) \in \mathbb{R} \times \Theta$. (iv) For every $\theta \in \Theta$, the function $\Psi(\cdot, \theta)$ is monotone nondecreasing, i.e., if $z_1 \leq z_2$ then $\Psi(z_1, \theta) \leq \Psi(z_2, \theta)$. (v) For every $z, a \in \mathbb{R}$,

$$\inf_{\theta \in \Theta} \Psi(z + a, \theta) = a + \inf_{\theta \in \Theta} \Psi(z, \theta).$$

Extended polyhedral risk measures, introduced in [10], are also of form (1.7).

Proposition 1.1.1. *Assumptions (B) and (C) imply that the functional \mathcal{R} , defined in (1.7), satisfies the axioms of convex risk measures².*

Proof. It follows from assumption (C)(iii) that $\mathbb{E}[\Psi(Z, \theta)]$ is convex in $(Z, \theta) \in \mathcal{Z} \times \Theta$, and hence its minimum over convex set Θ is convex. That is, the functional $\mathcal{R} : \mathcal{Z} \rightarrow \mathbb{R}$ is convex. By Assumptions (B) and (C)(iv) the functional \mathcal{R} is monotone, i.e., if $Z, Z' \in \mathcal{Z}$ are such that $Z \geq Z'$ almost surely (a.s.), with respect to the measure P , then $\mathcal{R}(Z) \geq \mathcal{R}(Z')$. Assumptions (B) and (C)(v) imply the translation equivariance property, i.e., $\mathcal{R}(Z + a) = \mathcal{R}(Z) + a$ for any $Z \in \mathcal{Z}$ and $a \in \mathbb{R}$. \square

Moreover, Assumption (C)(iv) implies that \mathcal{R} is consistent with the stop-loss order meaning that if $Z_1 \leq_{\text{icx}} Z_2$ then $\mathcal{R}(Z_1) \leq \mathcal{R}(Z_2)$ where the relation \leq_{icx} between random

¹It is possible to consider different parametric sets Θ_t and different functions Ψ_t for different time periods. For the sake of simplicity, we consider the same set Θ and function Ψ , this is in line with the examples below. On the other hand, the probability distributions P_t could be different for different time periods.

²In fact, in our construction of the statistical upper bound we do not need the translation equivariance property. We assume this property in order for \mathcal{R} to satisfy the standard axioms of convex risk measures.

variables (representing losses) is given by

$$Z_1 \leq_{\text{icx}} Z_2 \iff \mathbb{E}[f(Z_1)] \leq \mathbb{E}[f(Z_2)]$$

for any nondecreasing convex function f such that the respective expectations are well defined (see [11, 12]). For random variables representing incomes, the stop-loss order is replaced with the second order stochastic dominance, see for instance [11, 12, 10].

Recall that $Z, Z' \in \mathcal{Z}$ are said to be distributionally equivalent (with respect to the reference measure P) if $P(Z \leq z) = P(Z' \leq z)$ for all $z \in \mathbb{R}$. It is said that a functional $\mathcal{R} : \mathcal{Z} \rightarrow \mathbb{R}$ is *law invariant* if $\mathcal{R}(Z) = \mathcal{R}(Z')$ for any distributionally equivalent $Z, Z' \in \mathcal{Z}$. It follows immediately from the definition (1.8) that \mathcal{R}_t is a function of its cdf $F_t(z) = P_t(Z_t \leq z)$, and hence is law invariant. For every t , consider the direct product $P_1 \times \dots \times P_t$ of probability measures and the corresponding space $\mathcal{Z}_1 \times \dots \times \mathcal{Z}_t$. Conditional mapping $\mathcal{R}_{t|\xi_{[t-1]}} : \mathcal{Z}_t \rightarrow \mathbb{R}$ is defined as a counterpart of the law invariant functional \mathcal{R}_t , $t = 1, \dots, T$. Since ξ_0 is deterministic, $\mathcal{R}_{1|\xi_0} = \mathcal{R}$. The associated nested functional is defined in the composite form

$$\mathfrak{R}(\cdot) := \mathcal{R}_{1|\xi_0} \left(\mathcal{R}_{2|\xi_{[1]}} \left(\dots \mathcal{R}_{T|\xi_{[T-1]}}(\cdot) \right) \right). \quad (1.9)$$

We refer to [9, section 7.6] for a detailed discussion of constructions of such conditional mappings and nested functionals. Note that in this framework the process ξ_1, \dots, ξ_T , viewed as a random process with respect to the reference probability distributions, is *stagewise independent* with P_t being the marginal distribution of ξ_t .

There is a large class of risk measures which can be represented in the parametric form (1.7).

Example 1.1.1. *The Average Value-at-Risk measure*

$$\text{AV@R}_\alpha(Z) = \inf_{\theta \in \mathbb{R}} \mathbb{E} [\theta + \alpha^{-1}[Z - \theta]_+], \quad \alpha \in (0, 1), \quad (1.10)$$

is of form (1.7) with $\Psi(z, \theta) = \theta + \alpha^{-1}[z - \theta]_+$, and $\Theta = \mathbb{R}$, $\mathcal{Z} = L_1(\Omega, \mathcal{F}, P)$.

Example 1.1.2. A convex combination of the expectation and of Average Value-at-Risk measures given by

$$\mathcal{R}(Z) := \lambda_0 \mathbb{E}[Z] + \sum_{i=1}^k \lambda_i \text{AV@R}_{\alpha_i}(Z),$$

where λ_i are positive numbers with $\sum_{i=0}^k \lambda_i = 1$, and $\alpha_i \in (0, 1)$. Here \mathcal{R} is of form (1.7) with $\Theta = \mathbb{R}^k$, $\mathcal{Z} = L_1(\Omega, \mathcal{F}, P)$, and $\Psi(z, \theta) = \lambda_0 z + \sum_{i=1}^k \lambda_i (\theta_i + \alpha_i^{-1}[z - \theta_i]_+)$.

Example 1.1.3 (ϕ -divergence). Another example is risk measures constructed from ϕ -divergence ambiguity sets (cf., [13],[14],[9, section 7.2.2]). Let $\phi : \mathbb{R} \rightarrow \mathbb{R}_+ \cup \{+\infty\}$ be a convex lower semicontinuous function such that $\phi(1) = 0$ and $\phi(x) = +\infty$ for $x < 0$. By duality arguments the distributionally robust functional associated with the ambiguity set determined by the respective ϕ -divergence constraint with level $\epsilon > 0$ can be written in the form (1.7) with

$$\mathcal{R}_\epsilon(Z) = \inf_{\mu, \lambda > 0} \{ \lambda \epsilon + \mu + \lambda \mathbb{E}_P[\phi^*((Z - \mu)/\lambda)] \}, \quad (1.11)$$

$\theta = (\mu, \lambda)$, $\lambda > 0$, and $\Psi(z, \theta) = \lambda \epsilon + \mu + \lambda \phi^*((Z - \mu)/\lambda)$, where ϕ^* is the conjugate of ϕ . In particular for the Kullback-Leibler (KL)-divergence, $\phi(x) = x \ln x - x + 1$, $x \geq 0$, and

$$\mathcal{R}_\epsilon(Z) = \inf_{\mu, \lambda > 0} \{ \lambda \epsilon + \mu + \lambda e^{-\mu/\lambda} \mathbb{E}_P[e^{Z/\lambda}] - \lambda \}. \quad (1.12)$$

Given $\lambda > 0$ the minimizer over μ in (1.12) is given by $\mu = \lambda \ln \mathbb{E}_P[e^{Z/\lambda}]$ and hence

$$\mathcal{R}_\epsilon(Z) = \inf_{\lambda > 0} \{ \lambda \epsilon + \lambda \ln \mathbb{E}_P[e^{Z/\lambda}] \}. \quad (1.13)$$

Risk measures in the above examples are positively homogeneous, and hence are coherent.

Example 1.1.4. Let $u : \mathbb{R} \rightarrow [-\infty, +\infty)$ be a proper closed concave and nondecreasing

utility function with nonempty domain. The functional

$$\mathcal{R}(Z) := \inf_{\theta \in \mathbb{R}} \{\theta - \mathbb{E}[u(Z + \theta)]\},$$

is of form (1.7) with $\Theta = \mathbb{R}$ and $\Psi(z, \theta) = \theta - u(z + \theta)$. This risk measure is convex, but is not necessarily positively homogeneous. It can be viewed as the opposite of the OCE (Optimized Certainty Equivalent (see [15])).

Utilizing the nested functional (1.9), the risk averse multistage programming problem can be written as

$$\min_{x_1 \in \bar{\mathcal{X}}_1} c_1^\top x_1 + \mathcal{R}_2 \left(\inf_{x_2 \in \bar{\mathcal{X}}_2} c_2^\top x_2 + \mathcal{R}_t \left(\cdots + \mathcal{R}_T \left(\inf_{x_T \in \bar{\mathcal{X}}_T} c_T^\top x_T \right) \right) \right), \quad (1.14)$$

where $\bar{\mathcal{X}}_1 := \mathcal{X}_1 \cup \{x_t : B_t x_{t-1} + A_t x_t = b_t\}$. Based on (1.14), the risk averse counterpart of dynamic programming equations (1.3) can be written as (recall assumption (A0))

$$Q_t(x_{t-1}, \xi_t) = \inf_{x_t \in \mathcal{X}_t} \{c_t^\top x_t + \gamma Q_{t+1}(x_t) : B_t x_{t-1} + A_t x_t = b_t\}, \quad (1.15)$$

for $t = T, \dots, 1$ with $Q_t(x_{t-1}) = \mathcal{R}_t(Q_t(x_{t-1}, \xi_t))$, $Q_{T+1} \equiv 0$.

1.2 Computational Methods for Solving Stochastic Program

1.2.1 Stochastic Dual Dynamic Programming (SDDP)

For a multistage stochastic program, in order to make the problem solvable, we need to first discretize the underlying stochastic process if it has very large/infinite number of realizations. For stagewise independent data process, we discretize it using the Sample Average Approximation (SAA) [16], which generates an independent identically distributed (i.i.d.) random sample of a random vector ξ_t at stage t and employs the sample average to approx-

imate the expectation for each stage. We refer to the problem with discretized stochastic process by the SAA approach as the SAA problem. If the distribution of ξ_t has finite support, we naturally obtain the sample average formulation of the model in terms of finite number of realizations. The dynamic programming equations for the SAA problem can be written as

$$\hat{Q}_{tj}(x_{t-1}) = \inf_{x_t \in \mathcal{X}_t} \{ \hat{c}_{tj}^\top x_t + \gamma \hat{Q}_{t+1}(x_t) : \hat{B}_{tj} x_{t-1} + \hat{A}_{tj} x_t = \hat{b}_{tj} \}, j = 1, \dots, N_t \quad (1.16)$$

with

$$\hat{Q}_{t+1}(x_t) = \frac{1}{N_{t+1}} \sum_{j=1}^{N_{t+1}} \hat{Q}_{t+1,j}(x_t), \quad (1.17)$$

$t = 2, \dots, T$ and $\hat{Q}_{T+1}(\cdot) \equiv 0$. At the first stage, we solve

$$\min_{x_1 \in \mathcal{X}_1} \{ c_1^\top x_1 + \gamma \hat{Q}_2(x_1) : A_1 x_1 = b_1 \}, \quad (1.18)$$

which gives the optimal value of the SAA problem. Here $\hat{\xi}_{tj} := (\hat{c}_{tj}, \hat{A}_{tj}, \hat{B}_{tj}, \hat{b}_{tj})$, $j = 1, \dots, N_t$ is a random sample of ξ_t . We denote this random sample for the SAA construction by \mathcal{S}_N .

We will discuss applying the SDDP method to the SAA problem rather than the original problem if the embedded stochastic process does not have finite number of realizations. In the discussion of the SDDP method, we will concentrate on the computational properties of applying it to a finitely generated problem. The statistical properties of the sample average approximation can be addressed separately. In brief, it is pointed out in [17] that under mild regularity conditions, the sample complexity of the SAA approach is $\mathcal{O}(\epsilon^{-2(T-1)})$, which is the order of the number of scenarios (i.e. $\prod_{t=2}^T N_t$) required for a first stage solution of the SAA problem to be ϵ -optimal for the true problem. In Chapter 3, we will discuss the sample complexity in particular for the discounted stationary stochastic program.

SDDP algorithm was first suggested in [18] to solve the dynamic programming equa-

tions, which is based on the nested cutting plane method of [19] and consists with a forward step and a backward step at each run. Recent developments in [20] have shown that the complexity of the method is linearly dependent on the number of the stages T . However, when the number of state variables is large, it is still challenging to solve the dynamic programming equations of the SAA problem.

To illustrate, in the backward step of the algorithm, at stage t , given trial solution \bar{x}_t and the current approximation $\mathfrak{Q}_t(\cdot)$ of $\hat{Q}_t(\cdot)$, the optimal value and subgradient are computed and participate in constructing a new supporting plane at the corresponding trial point. The cutting plane approximation of the value function is then updated by the maximum of the supporting planes obtained so far. In the forward step of the algorithm, for a generated sample path $\hat{\xi}_t$ of realizations of the random data process, starting with x_0 the trial points are generated by computing a minimizer \bar{x}_t of the right-hand side of (1.16) with $\hat{Q}_t(\cdot)$ replaced by $\mathfrak{Q}_t(\cdot)$.

According to the assumption of the relatively complete recourse (A1), $\bar{x}_t = \bar{x}(\xi_{[t]}), t = 1, \dots, T$ is a feasible implementable policy for the considered SAA problem. As a result, the expected value $\mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} c_t^\top \bar{x}_t \right]$ gives an upper bound for the optimal value of the SAA problem. In practice, a subsample \mathcal{S}_M of \mathcal{S}_N is generated with replacement in the forward step, where $\mathcal{S}_M := \{\hat{\xi}_{tj}, t = 2, \dots, T, j = 1, \dots, M\}$ (recall that SAA problem is constructed from a random sample \mathcal{S}_N) and the optimal values $v_j := \sum_{t=1}^T \hat{c}_{tj}^\top \bar{x}_{tj}, j = 1, \dots, M$ are computed accordingly. Then

$$\left[\bar{v} - z_{\alpha/2} \hat{\sigma} / \sqrt{M}, \bar{v} + z_{\alpha/2} \hat{\sigma} / \sqrt{M} \right] \quad (1.19)$$

constitutes an approximate $100(1 - \alpha)$ confidence interval for the optimal value of the SAA problem. Here $\bar{v} := \frac{1}{M} \sum_{j=1}^M v_j, \hat{\sigma} := \sqrt{\frac{1}{M-1} \sum_{j=1}^M (v_j - \bar{v})^2}$ and z_α denotes the $(1 - \alpha)$ -quantile of the standard normal distribution. In addition, the upper end of the confidence interval in (1.19) gives an upper bound of the optimal value of the SAA problem. In this way, a

relative gap is computed so as to guide the stopping of the SDDP algorithm. Specifically, let \hat{Q}_1 be the optimal value of the first stage model computed in the backward step, which is the lower bound of the optimal value of the SAA problem. Then

$$gap = \frac{\bar{v} + z_{\alpha/2}\hat{\sigma}/\sqrt{M} - \hat{Q}_1}{\hat{Q}_1} \times 100\%. \quad (1.20)$$

If the gap is below some given precision ϵ , the SDDP algorithm terminates, which indicates that the SAA problem is solved with accuracy ϵ with confidence of about $1 - \alpha$. We refer to [21] for further analysis of the method and to [22] for implementation details.

For the multistage stochastic programs that inherit periodical stochastic processes and with discount factor very close to 1, the computational effort to reduce the gap becomes prohibitive. This motivates us to develop the deterministic upper bounds based on the dual formulation and will be detailed in Chapter 2.

So far, we have presented a framework of the SDDP method for the risk neutral multistage stochastic programs. As for the risk averse case, it can be handled with an SDDP type method with additional but simple modifications on top. To illustrate, in the backward step, given trial points \bar{x}_{t-1} and current lower approximation $\mathfrak{Q}_{t+1}(\cdot)$, we can obtain the optimal values by solving the corresponding minimization problem given below

$$\underline{Q}_t(\bar{x}_{t-1}, \xi_t^j) = \min_{x_t \in \mathcal{X}} \{ \hat{c}_{tj}^\top x_t + \gamma \mathfrak{Q}_{t+1}(x_t) : \hat{B}_{tj} \bar{x}_{t-1} + \hat{A}_{tj} x_t = \hat{b}_{tj} \}, j = 1, \dots, N_t \quad (1.21)$$

as well as the subgradients g_t^j of $\underline{Q}_t(\cdot, \xi_t^j)$ at \bar{x}_{t-1} . The approximation \mathfrak{Q}_t is updated as follow. First, we obtain the value of the current cost-to-go function as $\mathcal{R}_t(\underline{Q}_t(\bar{x}_{t-1}, \xi_t))$.

Take the risk measure $(1 - \lambda)\mathbb{E}[\cdot] + \lambda\text{CVaR}_\alpha(\cdot)$ as an example,

$$\begin{aligned} \mathcal{R}_t(\underline{Q}_t(\bar{x}_{t-1}, \xi_t)) &= \frac{1 - \lambda}{N} \sum_{j=1}^N \underline{Q}_t(\bar{x}_{t-1}, \xi_t^j) + \lambda \underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(\kappa)}) \\ &+ \frac{\lambda}{\alpha N} \sum_{j=\kappa+1}^N \left(\underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(j)}) - \underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(\kappa)}) \right), \end{aligned} \quad (1.22)$$

where $\underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(j)})$ are ordered statistics such that $\underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(1)}) \leq \dots \leq \underline{Q}_t(\bar{x}_{t-1}, \xi_t^{(N)})$ and $\kappa := \lceil (1 - \alpha)N \rceil$. Then we compute the gradients under the same measure of \mathcal{R}_t and compute the supporting planes accordingly at the trial points in a similar manner as in SDDP for risk neutral problem. We present a more detailed SDDP type algorithm for risk-averse SOC problem in Chapter 3.

1.2.2 Projection-free Methods for Risk Averse Single-stage Problem

In spite of their importance in a variety of different applications, the algorithmic studies for solving single-stage risk averse functional constrained problems are still limited. When the dimension of the decision variables x is large, one natural choice for solving problem (1.6) would be first-order methods. These methods only require first-order information of the objective and constraint functions, and have been widely used in large-scale data analysis and machine learning applications due to their scalability. Most of these first-order methods require the projection over the feasible set X (see [23]), which results in two significant limitations when applied to sparse optimization. First, the projection step often destroys the sparsity requirement in the sense that they cannot guarantee a sparse solution trajectory. Second, projection-based methods often require the computation of full gradients, which can be computationally expensive, and sometimes is not even possible.

To avoid these issues associated with projection-based algorithms, a common practice is to opt for the projection-free (a.k.a. conditional gradient) methods, which were pioneered in the work [24] and subsequently developed in [25, 26, 27, 28, 29, 30]. We refer

to [31] for a more comprehensive review of the method. Such algorithms eschew projections in favor of linear optimization. To be more specific, at each step, given the current iterate, these algorithms move towards an extreme point of a feasible set when optimizing a linear approximation of the objective function, and then update the iterate as a convex combination of the selected extreme points. As a consequence, each solution generated by these algorithms possesses sparse or low-rank properties. The generation of a sparse solution trajectory is one of the crucial properties that make projection-free methods stand out in sparse optimization, since this will enable the practitioners to choose one sparse solution from the trajectory. Another appealing property is that these algorithms only require the computation of one gradient component rather than the full gradient in many sparse optimization problems.

Unfortunately, the simplicity of existing projection-free methods comes with two major limitations. The first one is that it only demonstrates efficiency in convex problems with simple feasible set without functional constraints. To address this issue, [31] proposed several novel constraint-extrapolated conditional gradient type methods (CoexCG and CoexDurCG) that handle the convex optimization with more involved functional constraints. However, these algorithms assume the existence of an optimal dual solution and the iteration complexity of the algorithms depends on the magnitude of a possibly large optimal dual solution associated with the function constraints. The second significant limitation is the lack of efficient projection-free methods in dealing with nonconvex optimization problems with functional constraints.

Methodologies developed for convex functional constrained problem lie in several lines. One research direction has been directed to exact/quadratic penalty and augmented Lagrangian methods [32, 33, 34, 35], etc., which require to solve the penalty subproblems and obtain the solutions therein. Instead of solving a more complicated penalty problem, a saddle-point reformulation of the original convex functional constrained problem is tackled by primal-dual type methods [36, 37, 38, 39, 40]. However, these methods require the pro-

jection over X and depend on the dual space, whose diameter may be large. Alternatively, the convex problem can be reformulated as a root finding problem and are suggested to be solved by level-set methods [41, 42, 43, 44, 45, 29, 46, 47]. In particular, [46] developed a projection-based level-set method that maintains a feasible solution at each iteration. However, their method relies on a relatively strong assumption of feasibility guarantee and requires an estimate of the optimality gap of the problem. Other variants in solving the convex subproblems of the root finding problem include accelerated gradient descent methods [48] and bundle methods [49, 50, 51], which resort to either an complicated quadratic program or a costly projection onto the feasible set.

Nonconvex problems have attracted much attention due to their empirical merits in important applications (see, e.g., [52, 53]). Algorithms are developed mainly in two different ways. One is to solve the problems indirectly within the framework of proximal point methods [54, 55, 56, 57, 58, 39], which approximate the problems with convex subproblems and may have nested structure that impedes efficient implementation. Direct approaches for solving nonconvex problems have also been studied in parallel (see e.g., [59, 60, 61, 62, 63]). However, these methods mainly focus on solving unconstrained problems or problems with simple feasible sets, which are not applicable to our setting. Existing methods for nonconvex optimization with function constraints all require projections (see, e.g., [64] and references therein), whereas current projection-free methods can only handle simple feasible sets for nonconvex optimization (see, e.g., [65] and [23, Section 7.1]).

Using ideas from conditional gradient sliding methods [66, 67, 68], one can possibly leverage projection-based scheme to solve convex and nonconvex functional constrained problems. Although the conditional gradient sliding type methods might improve the number of gradients evaluation and maintains the optimal number of calls of linear optimization (LO) oracle, they would require to compute and store the full gradients, which is not applicable in our case.

CHAPTER 2
UPPER BOUNDS FOR RISK NEUTRAL MULTISTAGE STOCHASTIC
PROGRAM

2.1 Overview

Duality plays a key role in optimization. For generic optimization problems, weak duality allows to bound the optimal value, even when there is a duality gap between the primal and dual optimal values. In the context of multistage stochastic programs, duality was studied in [69], see also [6] for a review. More recently, the sensitivity analysis of multistage stochastic programs was discussed in [70] and [71], where the sensitivity analysis is based on Lagrange multipliers associated with the value functions.

In this chapter, we study the dual of the risk neutral multistage stochastic linear program (MSLP) and introduce Dual SDDP exercised on the dual formulation for computing a sequence of convergent (deterministic) upper bounds for the optimal value (see [1]). The developed approach proved to be especially useful in the infinite horizon setting when the discount factor is very close to one. On the contrary, the conventional statistical upper bounds fails (cf., [2]). This is important for evaluating accuracy of the obtained solutions and stopping criteria. In particular, by running Primal SDDP and Dual SDDP in parallel, we are able to compute a deterministic lower bound (LB) and a deterministic upper bound (UB) of the MSLP. When the relative gap between the LB and UB is close enough (within some precision), it provides a valid stopping criteria, which implies that the algorithm (Parallel Primal SDDP and Dual SDDP) solves for an approximately optimal primal solution of the MSLP and such solution can be obtained from the forward pass of Primal SDDP. As a byproduct, we apply the developed methodology to sensitivity analysis of the optimal values.

The rest of this chapter is organized as follows. In Section 2.2 we describe the construction of the dual bounds for MSLP in finite-horizon setting. We also study the dynamics of Lagrange multipliers which is of important application in the sensitivity analysis. Then we present Dual SDDP and its variants in Section 2.3. In Section 2.4 and Section 2.5, we extend our analysis and methodology to the infinite-horizon setting, where the duality of periodical MSLP and the corresponding SDDP-type algorithm will be studied. Finally in Section 2.6 we report numerical results on testing the proposed algorithms on an inventory problem and the Brazilian hydro-thermal planning problem.

2.2 Dual Bounds for Finite-horizon Multistage Stochastic Program

Consider problem (1.1) with $x_t \in \mathcal{X}_t$ replaced by $x_t \geq 0$ and the discount factor $\gamma = 1$. The Lagrangian of problem (1.1) is

$$L(x, \pi) = \mathbb{E} \left[\sum_{t=1}^T c_t^\top x_t + \pi_t^\top (b_t - B_t x_{t-1} - A_t x_t) \right] \quad (2.1)$$

in variables¹ $x = (x_1(\xi_{[1]}), \dots, x_T(\xi_{[T]}))$ and $\pi = (\pi_1(\xi_{[1]}), \dots, \pi_T(\xi_{[T]}))$ with the convention that $x_0 = 0$. Dualization of the feasibility constraints leads to the following dual of problem (1.1) [6, Chapter 3.2.3]:

$$\begin{aligned} \max_{\pi} \quad & \mathbb{E} \left[\sum_{t=1}^T b_t^\top \pi_t \right] \\ \text{s.t.} \quad & A_T^\top \pi_T \leq c_T, \\ & A_{t-1}^\top \pi_{t-1} + \mathbb{E}_{|\xi_{[t-1]}} [B_t^\top \pi_t] \leq c_{t-1}, \quad t = 2, \dots, T. \end{aligned} \quad (2.2)$$

The optimization in (2.2) is over policies $\pi_t = \pi_t(\xi_{[t]})$, $t = 1, \dots, T$.

Suppose further the process ξ_1, \dots, ξ_T is stagewise independent (i.e., random vector ξ_{t+1} is independent of $\xi_{[t]}$, $t = 1, \dots, T - 1$), and distribution of ξ_t has a finite support, $\{\xi_{t,1}, \dots, \xi_{t,N_t}\}$ with respective probabilities $p_{t,j}$, $j = 1, \dots, N_t$, $t = 2, \dots, T$. We denote by

¹Note that since ξ_1 is deterministic, the first-stage decision x_1 is also deterministic; we write it as $x_1(\xi_{[1]})$ for uniformity of notation, and similarly for π_1 .

$A_{t,j}, B_{t,j}, c_{t,j}, b_{t,j}$ the respective scenarios corresponding to $\xi_{t,j}$.

Since the random process $\xi_t, t = 1, \dots, T$, has a *finite* number of realizations (scenarios), problem (1.1) can be viewed as a large linear program and (2.2) as its dual. By the standard theory of linear programming we have the following.

Proposition 2.2.1. *Suppose that problem (1.1) has a finite optimal value. Then the optimal values of problems (1.1) and (2.2) are equal to each other and both problems have optimal solutions.*

2.2.1 Dynamic Programming Equations of the Dual

We can write the following DP equations for the dual problem (2.2). At the last stage $t = T$, given π_{T-1} and $\xi_{[T-1]}$, we need to solve the following problem with respect to π_T :

$$\begin{aligned} \max_{\pi_T} \quad & \mathbb{E}[b_T^\top \pi_T] \\ \text{s.t.} \quad & A_T^\top \pi_T \leq c_T, \\ & A_{T-1}^\top \pi_{T-1} + \mathbb{E}[B_T^\top \pi_T] \leq c_{T-1}. \end{aligned} \tag{2.3}$$

Since ξ_T is independent of $\xi_{[T-1]}$, the random parameters in (2.3) are functions of the marginal distribution of ξ_T , and are independent of $\xi_{[T-1]}$. Also it is assumed that ξ_T has a finite support. Therefore problem (2.3) can be written in terms of scenarios, corresponding to the marginal distribution of ξ_T , as follows

$$\begin{aligned} \max_{\pi_{T,1}, \dots, \pi_{T,N_T}} \quad & \sum_{j=1}^{N_T} p_{T,j} b_{T,j}^\top \pi_{T,j} \\ \text{s.t.} \quad & A_{T,j}^\top \pi_{T,j} \leq c_{T,j}, \quad j = 1, \dots, N_T, \\ & A_{T-1}^\top \pi_{T-1} + \sum_{j=1}^{N_T} p_{T,j} B_{T,j}^\top \pi_{T,j} \leq c_{T-1}. \end{aligned} \tag{2.4}$$

The optimal value $V_T(\pi_{T-1}, \xi_{T-1})$ and an optimal solution² $(\bar{\pi}_{T,1}, \dots, \bar{\pi}_{T,N_T})$ of prob-

²Note that problem (2.4) may have more than one optimal solution. In case of finite number of scenarios the considered linear program always has a solution provided its optimal value is finite.

lem (2.4) are functions of vectors π_{T-1} and c_{T-1} and matrix A_{T-1} . And so on going backward in time, using the stagewise independence assumption, we can write the respective dynamic programming equations for $t = T - 1, \dots, 2$, as

$$\begin{aligned} \max_{\pi_{t,1}, \dots, \pi_{t,N_t}} \quad & \sum_{j=1}^{N_t} p_{t,j} [b_{t,j}^\top \pi_{t,j} + V_{t+1}(\pi_{t,j}, \xi_{t,j})] \\ \text{s.t.} \quad & A_{t-1}^\top \pi_{t-1} + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1}, \end{aligned} \quad (2.5)$$

with $V_t(\pi_{t-1}, \xi_{t-1})$ being the optimal value of problem (2.5). Finally at the first stage the following problem should be solved

$$\max_{\pi_1} b_1^\top \pi_1 + V_2(\pi_1, \xi_1). \quad (2.6)$$

These dynamic programming equations can be compared with the dynamic programming equations for primal problem (1.1), where the respective value function $Q_t(x_{t-1}, \xi_{t,j})$, $j = 1, \dots, N_t$, is given by the optimal value of

$$\begin{aligned} \min_{x_t \geq 0} \quad & c_{t,j}^\top x_t + Q_{t+1}(x_t) \\ \text{s.t.} \quad & B_{t,j} x_{t-1} + A_{t,j} x_t = b_{t,j}, \end{aligned} \quad (2.7)$$

with $Q_{T+1}(\cdot) \equiv 0$, and for $t = T - 1, \dots, 1$,

$$Q_{t+1}(x_t) := \mathbb{E}[Q_{t+1}(x_t, \xi_{t+1})] = \sum_{j=1}^{N_t} p_{t+1,j} Q_{t+1}(x_t, \xi_{t+1,j}).$$

Let us make the following observations about the dual problem: (1) unlike in the primal problem, the optimization (maximization) problems (2.4) and (2.5) do not decompose into separate problems with respect to each $\pi_{t,j}$ and should be solved as one linear program with respect to $(\pi_{t,1}, \dots, \pi_{t,N_t})$; (2), the value function $V_t(\pi_{t-1}, \xi_{t-1})$ is a concave function of π_{t-1} ; (3) if A_t and c_t , $t = 2, \dots, T$, are deterministic, then $V_t(\pi_{t-1})$ is only a function of

π_{t-1} .

The following definition of Relatively Complete Recourse (RCR) is applied to the dual problem. Recall that we assume that the set of possible realizations (scenarios) of the data process is finite.

Definition 2.2.1. *We say that a sequence $\bar{\pi}_t, t = 1, \dots, T$, is generated by the forward step if $\bar{\pi}_1 \in \mathbb{R}^{m_1}$ and $\bar{\pi}_t$ coincides with some $\pi_{t,j}, j = 1, \dots, N_t, t = 2, \dots, T$, where $\pi_{t,1}, \dots, \pi_{t,N_t}$ is a feasible solution of the respective dynamic program (2.5) for $t = 2, \dots, T-1$, and (2.4) for $t = T$. We say that the dual problem (2.2) has Relatively Complete Recourse (RCR) if at every stage $t = 2, \dots, T$, for any generated π_{t-1} by the forward step, the respective dynamic program has a feasible solution at stage t for every realization of the random data.*

Without RCR, the value of $V_t(\pi_{t-1}, \xi_{t-1})$ could go to infinity for a generated π_{t-1} and $\xi_{t-1} = \xi_{t-1,j}$. Unfortunately, it could happen that the dual problem does not have the RCR property even if the primal problem does, even in the two-stage case. Indeed, the infeasibility of problem (2.4) could be due to its last constraint, but not because of the remaining ones since the primal problem is feasible and bounded below. Therefore by the LP duality theorem, the dual problem is feasible and has a finite optimal value. A toy example below illustrates this case.

Example 2.2.1. *Even both the controls and the states are bounded above and below in the primal model, it is possible that the dual model does not satisfy RCR. Consider the two-stage inventory problem in (2.57) with bounded x_t, y_t , namely*

$$\begin{aligned} \underline{x}_t &\leq x_t \leq \bar{x}_t, \\ \underline{y}_t &\leq y_t \leq \bar{y}_t. \end{aligned} \tag{2.8}$$

In the model, $c_t = 1.5 + \cos(\frac{\pi t}{6}), d_{tj} = (5 + 0.5t)(1.5 + 0.1z_{tj})$, where $\{z_{tj}\}_{j=1}^{M=5}$ is a sample from the standard Gaussian distribution. The boundaries and initial values are set

as $(\underline{x}_t, \bar{x}_t) = (-100, 100), (\underline{y}_t, \bar{y}_t) = (-100, 100), x_0 = 10$. The corresponding dual is given by the following

$$\begin{aligned}
-c_1 x_0 + \max \mathbb{E} \left[\sum_{t=1}^{T=2} d_t \pi_t - d_t \mu_t^1 + d_t \mu_t^2 + \bar{x}_t u_t^x - \underline{x}_t l_t^x + \bar{y}_t u_t^y - \underline{y}_t l_t^y \right] - x_0 \mu_1^3 \\
-\pi_{t-1} + u_{t-1}^x - l_{t-1}^x + \mathbb{E}[\mu_t^3] = -a_t, \quad t = 2 \\
\pi_t - \mu_t^3 - \mu_t^1 + \mu_t^2 + u_t^y - l_t^y = a_t, \quad t = 1, 2 \\
-\pi_T + u_T^x - l_T^x = 0 \\
\mu_t^1 \geq -g_t, \quad t = 1, 2 \\
\mu_t^2 \geq -h_t, \quad t = 1, 2 \\
\mu_t^1, \mu_t^2, \mu_t^3, u_t^x, l_t^x, u_t^y, l_t^y \leq 0, \quad t = 1, 2
\end{aligned} \tag{2.9}$$

From simple simulation, we observe that given a feasible first stage solution $(\pi_1, u_1^x, l_1^x) = (199.818, 0, 0)$, the second stage problem is infeasible. Therefore, in this instance, the second stage dual problem is not always feasible given any feasible first stage solutions. Hence it does not satisfy RCR.

The RCR is crucial for an implementation of the SDDP algorithm. One way to deal with the problem of absence of RCR in numerical procedures is to use the following penalty approach. Consider the following relaxation of problem (2.4):

$$\begin{aligned}
\tilde{V}_T(\pi_{T-1}, \xi_{T-1}) = \max_{\pi_{T,1}, \dots, \pi_{T,N_T}, \zeta_T \geq 0} \sum_{j=1}^{N_T} p_{T,j} b_{T,j}^\top \pi_{T,j} - v_T^\top \zeta_T \\
\text{s.t.} \quad A_{T,j}^\top \pi_{T,j} \leq c_{T,j}, \quad j = 1, \dots, N_T, \\
A_{T-1}^\top \pi_{T-1} + \sum_{j=1}^{N_T} p_{T,j} B_{T,j}^\top \pi_{T,j} \leq c_{T-1} + \zeta_T,
\end{aligned} \tag{2.10}$$

where v_T is a vector with positive components. For ζ_T large enough, the last constraint of problem (2.10) is satisfied. Consequently problem (2.10) is always feasible and hence its

optimal value $\tilde{V}_T(\pi_{T-1}, \xi_{T-1}) > -\infty$. We also have that

$$\tilde{V}_T(\pi_{T-1}, \xi_{T-1}) \geq V_T(\pi_{T-1}, \xi_{T-1}), \quad (2.11)$$

with the equality holding if $\zeta_T = 0$ in the optimal solution of (2.10). If $V_T(\pi_{T-1}, \xi_{T-1})$ is finite, this equality holds if the components of vector v_T are large enough (see Lemma 2.7.1).

Similarly, problems (2.5) can be relaxed to

$$\begin{aligned} \max_{\pi_{t,1}, \dots, \pi_{t,N_t}, \zeta_t \geq 0} \quad & \sum_{j=1}^{N_t} p_{t,j} \left[b_{t,j}^\top \pi_{t,j} + \tilde{V}_{t+1}(\pi_{t,j}, \xi_{t,j}) \right] - v_t^\top \zeta_t \\ \text{s.t.} \quad & A_{t-1}^\top \pi_{t-1} + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1} + \zeta_t, \end{aligned} \quad (2.12)$$

with vector v_t having positive components. In that way, the infeasibility problem is avoided and by (2.11) the obtained value gives an upper bound for the optimal value of the dual problem. Note that for sufficiently large vectors v_t this upper bound coincides with the optimal value of the dual problem (Lemma 2.7.1).

2.2.2 Dynamics of Lagrangian Multipliers

Let us consider for the moment the two-stage setting, i.e., $T = 2$. The primal problem can be written as

$$\min_{x_1 \geq 0} c_1^\top x_1 + \mathbb{E} [Q(x_1, \xi_2)] \quad \text{s.t.} \quad A_1 x_1 = b_1, \quad (2.13)$$

where $Q(x_1, \xi_2)$ is the optimal value of the second-stage problem

$$\min_{x_2 \geq 0} c_2(\xi_2)^\top x_2 \quad \text{s.t.} \quad B_2(\xi_2)x_1 + A_2(\xi_2)x_2 = b_2(\xi_2). \quad (2.14)$$

The Lagrangian of problem (2.14) is

$$L(x_1, x_2, \lambda, \xi_2) = c_2(\xi_2)^\top x_2 + \lambda^\top (b_2(\xi_2) - B_2(\xi_2)x_1 - A_2(\xi_2)x_2).$$

In the dual form, $Q(x_1, \xi_{2,j})$ is given by the optimal value of the problem

$$\max_{\lambda_{2,j}} (b_{2,j} - B_{2,j}x_1)^\top \lambda_{2,j} \text{ s.t. } c_{2,j} - A_{2,j}^\top \lambda_{2,j} \geq 0. \quad (2.15)$$

We have that if $x_1 = \bar{x}_1$ is an optimal solution of the first stage problem, then optimal Lagrange multipliers $\pi_{2,j}$ are given by the optimal solution $\bar{\lambda}_{2,j}$ of problem (2.15).

This can be extended to the multistage setting of problem (1.1) (recall that the stagewise independence condition is assumed). At the last stage $t = T$, given optimal solution \bar{x}_{T-1} , the following problem should be solved

$$\min_{x_T \geq 0} c_T(\xi_T)^\top x_T \text{ s.t. } B_T(\xi_T)\bar{x}_{T-1} + A_T(\xi_T)x_T = b_T(\xi_T). \quad (2.16)$$

For a realization $\xi_T = \xi_{T,j}$, the dual of problem (2.16) reads

$$\max_{\lambda_{T,j}} (b_{T,j} - B_{T,j}\bar{x}_{T-1})^\top \lambda_{T,j} \text{ s.t. } c_{T,j} - A_{T,j}^\top \lambda_{T,j} \geq 0. \quad (2.17)$$

We then have that $\pi_{T,j}$ are given by the optimal solution $\bar{\lambda}_{T,j}$ of problem (2.17). At stage $t = T - 1$, given optimal solution \bar{x}_{T-2} , the following problem is supposed to be solved (see (2.7))

$$\begin{aligned} \min_{x_{T-1} \geq 0} \quad & c_{T-1}(\xi_{T-1})^\top x_{T-1} + Q_T(x_{T-1}) \\ \text{s.t.} \quad & A_{T-1}(\xi_{T-1})x_{T-1} = b_{T-1}(\xi_{T-1}) - B_{T-1}(\xi_{T-1})\bar{x}_{T-2}. \end{aligned} \quad (2.18)$$

We have that $Q_T(\cdot)$ is a convex piecewise linear function. Therefore for every realization $\xi_{T-1} = \xi_{T-1,j}$ it is possible to represent (2.18) as a linear program and hence to write its dual. The optimal Lagrange multipliers of that dual give the corresponding Lagrange multipliers $\pi_{T-1,j}$. And so on for other stages going backward in time. That is, we have the following.

Remark 2.2.1. *If $(\bar{x}_1, \dots, \bar{x}_T(\xi_{[T]}))$ is an optimal solution of the primal problem, then for*

$x_{t-1} = \bar{x}_{t-1}$ the Lagrange multiplier $\pi_{t,j}$ is given by the respective Lagrange multiplier of problem (2.7).

2.2.3 Sensitivity Analysis

In this part, we discuss an application of the duality analysis to a study of sensitivity of the optimal value to small perturbations of the involved parameters. Suppose now that the data $c_t(\xi_t, \theta)$, $b_t(\xi_t, \theta)$, $B_t(\xi_t, \theta)$, $A_t(\xi_t, \theta)$ of problem (1.1) also depend on parameter vector $\theta \in \mathbb{R}^k$. Denote by $\vartheta(\theta)$ the optimal value of the parameterized problem (1.1) considered as a function of θ , and by $\mathfrak{S}(\theta)$ and $\mathfrak{D}(\theta)$ the sets of optimal solutions of the respective primal and dual problems. Recall that the sets $\mathfrak{S}(\theta)$ and $\mathfrak{D}(\theta)$ are nonempty provided the optimal value $\vartheta(\theta)$ is finite. Consider the directional derivative

$$\vartheta'(\theta, h) = \lim_{\tau \downarrow 0} \frac{\vartheta(\theta + \tau h) - \vartheta(\theta)}{\tau}$$

of $\vartheta(\cdot)$ at θ in direction h . Recall that $\vartheta(\cdot)$ is (Gâteaux) differentiable at θ iff $\vartheta'(\theta, h)$ exists for all $h \in \mathbb{R}^k$ and is linear in h , in which case $\vartheta'(\theta, h) = h^\top \nabla \vartheta(\theta)$.

Let $L(x, \pi, \theta)$ be the corresponding Lagrangian (see (2.38)) considered as a function of θ . Then we have the following formula for the directional derivatives of the optimal value function (e.g., [72, Proposition 4.27]).

Proposition 2.2.2. *Suppose that the data functions are continuously differentiable functions of θ , and for a given $\theta = \bar{\theta}$ the optimal value $\vartheta(\bar{\theta})$ is finite and the sets $\mathfrak{S}(\bar{\theta})$ and $\mathfrak{D}(\bar{\theta})$ of optimal solutions are bounded. Then*

$$\vartheta'(\bar{\theta}, h) = \max_{\pi \in \mathfrak{D}(\bar{\theta})} \min_{x \in \mathfrak{S}(\bar{\theta})} h^\top \nabla_{\theta} L(x, \pi, \bar{\theta}). \quad (2.19)$$

In particular if $\mathfrak{S}(\bar{\theta}) = \{\bar{x}\}$ and $\mathfrak{D}(\bar{\theta}) = \{\bar{\pi}\}$ are singletons, then $\vartheta(\cdot)$ is differentiable at $\bar{\theta}$ and

$$\nabla \vartheta(\bar{\theta}) = \nabla_{\theta} L(\bar{x}, \bar{\pi}, \bar{\theta}). \quad (2.20)$$

2.3 Dual SDDP

In this section, using the results of 2.2, we discuss an adaptation of the cutting planes approach for the approximation of the value functions of the dual problem, similar to the standard SDDP method. More specifically, we develop Dual SDDP and its variants for cases with different underlying uncertainty.

2.3.1 Dual SDDP for Problems with Uncertainty in b_t and B_t

In this part, we consider the case where only b_t and B_t are functions of ξ_t , and hence are random. We first state in the following the result that under some mild condition, the optimal value of (2.5) (resp. (2.6)) does not change by adding the box constraints $\underline{\pi}_t \leq \pi_{t,j} \leq \bar{\pi}_t$ (resp. $\underline{\pi}_1 \leq \pi_1 \leq \bar{\pi}_1$).

Lemma 2.3.1. *Suppose that the optimal value of primal problem (1.1) is finite and that there exists feasible $\hat{x} > 0$. Then for every $t = 1, \dots, T$, there exist $\underline{\pi}_t, \bar{\pi}_t \in \mathbb{R}^{m_t}$ such that adding box constraints $\underline{\pi}_t \leq \pi_t \leq \bar{\pi}_t$ the dual problem (2.5) is unchanged (i.e., has the same optimal value)*

Consequently, we can reformulate the dual problem by adding the box constraints on the dual variables.

Recall that it is assumed that the number of scenarios is finite and hence problem (1.1) can be viewed as a large linear program. The assumption of existence of feasible $\hat{x} > 0$ means that problem (1.1) possesses a feasible solution with all components being strictly positive. If, the equality constraints of problem (1.1) are linearly independent, then the aforementioned strict feasibility condition implies that the set of optimal solutions of the dual problem (i.e., the set of Lagrange multipliers) is bounded, see for instance [73]. On the other hand, in the above lemma the linear independence condition is not assumed. A proof of Lemma 2.3.1 and a way to obtain the corresponding bounds $\underline{\pi}_t, \bar{\pi}_t$ can be found in 2.7.

As mentioned earlier, a difficulty to solve the dual problem with an SDDP type method is that RCR may not be satisfied by the dual problem, even if RCR holds for the primal. We propose Dual SDDP with penalizations to deal with the issue.

Dual SDDP with penalizations solves the dual problem (2.4)-(2.6) by directly exercising on the approximation problem (2.10) and (2.12) when $t > 1$ with box constraints on the dual variables according to Lemma (2.3.1). To illustrate, it contains a forward pass which generates trial points and a backward pass which computes cutting planes for the value functions. On top of that, it introduces slack variables in the constraints which may become infeasible for some past decisions in the subproblems solved in the forward passes. Slack variables are penalized in the objective function with positive penalizing coefficients $v_{t,k}$. Therefore, at each iteration, all subproblems solved in forward and backward passes are always feasible. At iteration k , concave value functions $V_t, t = 2, \dots, T$, are approximated by polyhedral upper bounding functions V_t^k such that:

$$V_t^k(\pi_{t-1}) = \min_{0 \leq i \leq k} \bar{\theta}_t^i + \langle \bar{\beta}_t^i, \pi_{t-1} \rangle \quad (2.21)$$

where $\bar{\theta}_t^i$ and $\bar{\beta}_t^i$ are respectively real numbers and vectors. In the following, we present the detailed scheme of Dual SDDP.

Initialization. For $t = 2, \dots, T$, take V_t^0 as an affine upper bounding function for V_t and $V_{T+1}^0 \equiv 0$. Set iteration counter k to 1.

Step 1: forward pass of iteration k (computation of dual trial points). For the first stage of the forward pass, we compute an optimal solution π_1^k of

$$V_1^k = \max_{\pi_1} b_1^\top \pi_1 + V_2^{k-1}(\pi_1), \text{ s.t. } \underline{\pi}_1 \leq \pi_1 \leq \bar{\pi}_1. \quad (2.22)$$

For stage $t = 2, \dots, T - 1$, generate a realization $\xi_{t,j_t(k)}$ from ξ_t . Given π_{t-1}^k , and

compute

$$\begin{aligned}
V_t^k &= \max_{\pi_{t,1}, \dots, \pi_{t,N_t}, \zeta_t \geq 0} \sum_{j=1}^{N_t} p_{t,j} [b_{t,j}^\top \pi_{t,j} + V_{t+1}^{k-1}(\pi_{t,j})] - v_{t,k}^\top \zeta_t \\
\text{s.t.} \quad & A_{t-1}^\top \pi_{t-1}^k + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1} + \zeta_t, \\
& \underline{\pi}_t \leq \pi_{t,j} \leq \bar{\pi}_t, \quad j = 1, \dots, N_t.
\end{aligned} \tag{2.23}$$

An optimal solution of the problem above has N_t components $(\pi_{t,1}, \pi_{t,2}, \dots, \pi_{t,N_t})$ for π_t . Take π_t^k according to the probability mass function from $\pi_{t,1}, \dots, \pi_{t,N_t}$.

Step 2: backward pass of iteration k (computation of new cuts). For $t = T$, let $(x_T, x_{T-1}, \bar{\Psi}, \underline{\Psi})$ be an optimal solution of the dual of (2.23) as follows³:

$$\begin{aligned}
\bar{V}_T^k(\pi_{T-1}^k) &= \min_{x_T, x_{T-1}, \bar{\Psi}, \underline{\Psi}} x_{T-1}^\top (c_{T-1} - A_{T-1}^\top \pi_{T-1}^k) + \mathbb{E}[c_T^\top x_T + \bar{\Psi}^\top \bar{\pi}_T - \underline{\Psi}^\top \underline{\pi}_T] \\
\text{s.t.} \quad & A_T x_T + B_T x_{T-1} + \bar{\Psi} - \underline{\Psi} = b_T, \\
& 0 \leq x_{T-1} \leq v_{T,k}, x_T, \underline{\Psi}, \bar{\Psi} \geq 0
\end{aligned} \tag{2.24}$$

The new cut coefficients for V_T is given by

$$\bar{\beta}_T^k = -A_{T-1} x_{T-1}, \quad \bar{\theta}_T^k = \bar{V}_T^k(\pi_{T-1}^k) - \langle \bar{\beta}_T^k, \pi_{T-1}^k \rangle.$$

For $t = T - 1, \dots, 2$, compute an optimal solution $(x_{t-1}, \nu, \bar{\Psi}, \underline{\Psi})$ of

$$\begin{aligned}
\bar{V}_t^k(\pi_{t-1}^k) &= \min_{x_{t-1}, \nu, \bar{\Psi}, \underline{\Psi}} x_{t-1}^\top [c_{t-1} - A_{t-1}^\top \pi_{t-1}^k] + \mathbb{E} \left[\sum_{i=0}^k \nu_i \bar{\theta}_{t+1}^i + \bar{\Psi}^\top \bar{\pi}_t - \underline{\Psi}^\top \underline{\pi}_t \right] \\
\text{s.t.} \quad & B_t x_{t-1} - \sum_{i=0}^k \nu_i \bar{\beta}_{t+1}^i - \underline{\Psi} + \bar{\Psi} = b_t, \\
& \sum_{i=0}^k \nu_i = 1, \underline{\Psi}, \bar{\Psi} \geq 0, \\
& \nu_0, \dots, \nu_k \geq 0, 0 \leq x_{t-1} \leq v_{t,k},
\end{aligned} \tag{2.25}$$

³We suppressed the dependence of the optimal solution on T and k to alleviate notation.

and

$$\bar{\beta}_t^k = -A_{t-1}x_{t-1}, \quad \bar{\theta}_t^k = \bar{V}_t^k(\pi_{t-1}^k) - \langle \bar{\beta}_t^k, \pi_{t-1}^k \rangle.$$

Step 3: Do $k \leftarrow k + 1$ and go to Step 1.

The validity of the cuts computed in the backward pass of Dual SDDP with penalizations is shown in Proposition 2.3.1.

Proposition 2.3.1. *Consider Dual SDDP algorithm with penalizations $v_{t,k} \geq 0$. Let Assumptions (A1) and (A2) hold. Then for every $t = 2, \dots, T$, the sequence V_t^k is a non-increasing sequence of upper bounding functions for V_t , i.e., for every $k \geq 1$ we have $V_t \leq V_t^k \leq V_t^{k-1}$ and therefore (V^k) (recall that V^{k-1} is the optimal value of (2.22)) is a nonincreasing sequence of deterministic upper bounds on the optimal value of (1.1).*

To understand the effect of the sequence of penalizing parameters $(v_{t,k})$ on Dual SDDP with penalizations, we define the following DP equations (see also Lemma 2.7.1 in Section 2.7):

$$V_T^\gamma(\pi_{T-1}) = \begin{cases} \max_{\pi_{T,1}, \dots, \pi_{T,N_T}, \zeta_T \geq 0} & \sum_{j=1}^{N_T} p_{T,j} b_{T,j}^\top \pi_{T,j} - \gamma \mathbf{e}^\top \zeta_T \\ \text{s.t.} & A_{T,j}^\top \pi_{T,j} \leq c_{T,j}, \quad j = 1, \dots, N_T, \\ & A_{T-1}^\top \pi_{T-1} + \sum_{j=1}^{N_T} p_{T,j} B_{T,j}^\top \pi_{T,j} \leq c_{T-1} + \zeta_T, \end{cases} \quad (2.26)$$

for $t = 2, \dots, T - 1$:

$$V_t^\gamma(\pi_{t-1}) = \begin{cases} \max_{\pi_{t,1}, \dots, \pi_{t,N_t}, \zeta_t \geq 0} & \sum_{j=1}^{N_t} p_{t,j} [b_{t,j}^\top \pi_{t,j} + V_{t+1}^\gamma(\pi_{t,j})] - \gamma \mathbf{e}^\top \zeta_t \\ \text{s.t.} & A_{t-1}^\top \pi_{t-1} + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1} + \zeta_t, \end{cases} \quad (2.27)$$

and we define the first-stage problem

$$\max_{\pi_1} \pi_1^\top b_1 + V_2^\gamma(\pi_1), \quad (2.28)$$

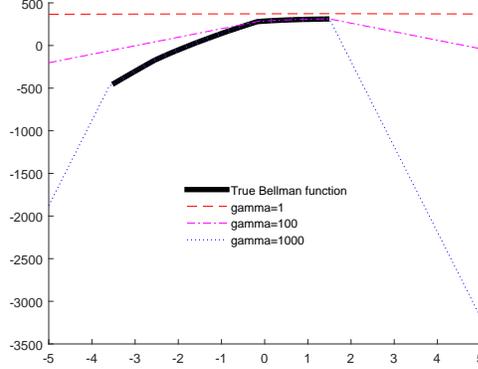


Figure 2.1: Graph of V_2 and of V_2^γ for $\gamma = 1, 100, 1000$.

where \mathbf{e} is a vector of ones and γ is a positive real number. As we will see below, V_t^γ can be seen as an upper bounding concave approximation of V_t which gets “closer” to V_t when γ increases. Therefore, Dynamic Programming can be used to solve these DP equations and obtain good approximations of functions V_t and V_t^γ . To obtain these approximations, we need to obtain approximations of the domains of functions V_t and compute approximations of these functions on a set of points in that domain.

To observe the impact of penalizing term γ on V_t^γ , we run Dynamic Programming both on DP equations (2.4), (2.5), (2.6) and on DP equations (2.26), (2.27), (2.28) for $\gamma = 1, 100, 1000$, on an instance of the inventory problem with $T = 20$ and $N_t = 20$. The corresponding graphs of V_2 (bold dark solid line) and of V_2^γ for $\gamma = 1, 100, 1000$, are represented in 2.1. We observe that all functions V_2^γ are, as expected, concave upper bounding functions for V_2 finite everywhere. We also see that on the domain of V_2 , V_2^γ gets closer to V_2 when γ increases and eventually coincides with V_2 on this domain when γ is sufficiently large. Similar graphs were observed for remaining functions $V_t, V_t^\gamma, t = 3, \dots, T$. Therefore, convergence of Dual SDDP with penalizations requires the coefficients $v_{t,k}$ to become arbitrarily large. Proof of the following theorem is given in Section 2.7.

Theorem 2.3.1. *Consider optimization problem (1.1) and Dual SDDP with penalizations applied to the dual of this problem. Let Assumptions (A1) and (A2) hold. Assume that samples $\xi_t^\ell, t = 2, \dots, T, \ell \geq 1$, in the forward passes are independent, that $v_{t,k+1} \geq v_{t,k}$*

for all t, k , and that $\lim_{k \rightarrow +\infty} v_{t,k} = +\infty$ for all stage t . Then the sequence V^k is a deterministic sequence of upper bounds on the optimal value of (1.1) which converges almost surely to the optimal value of this problem.

The “deterministic” upper bounds V^k are functions of the randomly generated samples and as such can be viewed as random variables. By the standard theory of SDDP, these bounds converge almost surely to the optimal value of the dual problem, and hence, of the primal problem.

We refer to Section 2.6 for examples of sequences $v_{t,k}$ used to solve an hydro-thermal and an inventory problem.

2.3.2 Dual SDDP for Problems with Uncertainty in All Parameters

We have seen in Section 2.2 on how to write DP equations for the dual problem of a MSLP when all data (A_t, B_t, c_t, b_t) in (ξ_t) is random. In this situation, cost-to-go functions V_t are functions $V_t(\pi_{t-1}, \xi_{t-1})$ of both past decision π_{t-1} and past value ξ_{t-1} of process (ξ_t) . Also recall that functions $V_t(\cdot, \xi_{t-1})$ are concave for all ξ_{t-1} . Therefore, Dual SDDP with penalizations from the previous section must be modified as follows. For each stage $t = 2, \dots, T$, instead of computing just one approximation of a single function (function V_t), we now need to compute approximations of N_t functions, namely concave value functions $V_t(\cdot, \xi_{t-1,j})$, $j = 1, \dots, N_t$. The approximation $V_{t,j}^k$ computed for $V_t(\cdot, \xi_{t-1,j})$ at iteration k is a polyhedral function $V_{t,j}^k$ given by:

$$V_{t,j}^k(\pi_{t-1}) = \min_{0 \leq i \leq k} \bar{\theta}_{t,j}^i + \langle \bar{\beta}_{t,j}^i, \pi_{t-1} \rangle.$$

Therefore more computational effort is needed. However, the adaptations of the method can be easily written. More specifically, at iteration k , in the forward pass, dual trial points are obtained replacing $V_t(\cdot, \xi_{t-1,j})$ by $V_{t,j}^{k-1}$ and in the backward pass a cut is computed at stage t for $V_t(\cdot, \xi_{t-1,j_k})$ with j_k satisfying $\xi_{t-1,j_k} = \tilde{\xi}_{t-1}^k$ where $\tilde{\xi}_{t-1}^k$ is the sampled value of

ξ_{t-1} at iteration k .

2.3.3 Dual SDDP for Problems with Interstage Dependent Cost Coefficients

We consider problems of form (1.1) where costs c_t affinely depend on their past while b_t are stagewise independent. Specifically, suppose that c_t follow a multiplicative vector autoregressive process of form

$$c_t = \varepsilon_t \circ \left(\sum_{j=1}^p \Phi_{t,j} c_{t-j} + \mu_t \right), \quad (2.29)$$

with $(x \circ y)_i = x_i y_i$ denoting the componentwise product, and where matrices $\Phi_{t,j}$ and vectors $\mu_t \geq 0$ as well as $c_1, \dots, c_{2-p} \geq 0$ are given.

We assume that the process (b_t, ε_t) is stagewise independent and that the support of b_t, ε_t is the finite set

$$\{(b_{t,1}, \varepsilon_{t,1}), \dots, (b_{t,N_t}, \varepsilon_{t,N_t})\},$$

with $\varepsilon_{t,i} > 0$ and $p_{t,i} = \mathbb{P}\{(b_t, \varepsilon_t) = (b_{t,i}, \varepsilon_{t,i})\}$, $i = 1, \dots, N_t$. For some values of $\Phi_{t,j}$ (for instance for matrices with nonnegative entries), this guarantees that all realizations of the price process $\{c_t\}$ are positive. The developments which follow can be easily extended to other linear models for $\{c_t\}$, for instance ARIMA or AR models, see ([74]) for the definition of state vectors of minimal size for generalized linear models.

Using the notation $c_{t_1:t_2} = (c_{t_1}, c_{t_1+1}, \dots, c_{t_2-1}, c_{t_2})$ for $t_1 \leq t_2$ integer, for the corresponding primal problem (of the form (1.1)), we can write the following DP equations: define $Q_{T+1} \equiv 0$ and for $t = 2, \dots, T$,

$$Q_t(x_{t-1}, c_{t-p:t-1}) = \mathbb{E}_{b_t, \varepsilon_t} \left[Q_t(x_{t-1}, c_{t-p:t-1}, b_t, \varepsilon_t) \right] \quad (2.30)$$

where $Q_t(x_{t-1}, c_{t-p:t-1}, b_t, \varepsilon_t)$ is given by

$$\begin{aligned} \min_{x_t \geq 0} & \left[\varepsilon_t \circ \left(\sum_{j=1}^p \Phi_{t,j} c_{t-j} + \mu_t \right) \right]^\top x_t + Q_{t+1} \left(x_t, c_{t+1-p:t-1}, \varepsilon_t \circ \left(\sum_{j=1}^p \Phi_{t,j} c_{t-j} + \mu_t \right) \right) \\ & A_t x_t + B_t x_{t-1} = b_t, \end{aligned} \quad (2.31)$$

while the first-stage problem is

$$\begin{aligned} \min_{x_1 \geq 0} & c_1^\top x_1 + Q_2(x_1, c_{2-p:1}) \\ & A_1 x_1 = b_1. \end{aligned}$$

Standard SDDP does not apply directly to solve DP equations (2.30)-(2.31) because functions Q_t given by (2.30)-(2.31) are not convex. Nevertheless, we can use the Markov Chain discretization variant of SDDP (computational details can be found in [22]) to solve DP equations (2.30)-(2.31). On the other hand, as pointed above, it is possible to apply SDDP for the dual problem with the added state variables. Along the lines of 2.2.1, we can write DP equations for the dual, now with function V_t depending on $\pi_{t-1}, c_{t-1}, \dots, c_{t-p}$.

The following DP equations for the dual of (1.1) with (c_t) of form (2.29) can be written. For the last stage T , we have to solve the problem:

$$\begin{aligned} \max_{\pi_{T1}, \dots, \pi_{TN_T}} & \sum_{j=1}^{N_T} p_{Tj} \pi_{Tj}^\top b_{Tj} \\ & A_T^\top \pi_{Tj} \leq \varepsilon_{Tj} \circ \left(\mu_T + \sum_{\ell=1}^p \Phi_{T\ell} c_{T-\ell} \right), \quad j = 1, \dots, N_T, \\ & \sum_{j=1}^{N_T} p_{Tj} B_T^\top \pi_{Tj} \leq c_{T-1} - A_{T-1}^\top \pi_{T-1}, \end{aligned} \quad (2.32)$$

with optimal value $V_T(\pi_{T-1}, c_{T-p:T-1})$.

Next for stage $t = 2, \dots, T-1$, given V_{t+1} , we need to solve the problem

$$\begin{aligned} \max_{\pi_{t1}, \dots, \pi_{tN_t}} & \sum_{j=1}^{N_t} p_{tj} \left(\pi_{tj}^\top b_{tj} + V_{t+1} \left(\pi_{tj}, c_{t+1-p:t-1}, \varepsilon_{tj} \circ \left(\mu_t + \sum_{\ell=1}^p \Phi_{t\ell} c_{t-\ell} \right) \right) \right) \\ \text{s.t.} & \sum_{j=1}^{N_t} p_{tj} B_t^\top \pi_{tj} \leq c_{t-1} - A_{t-1}^\top \pi_{t-1}, \end{aligned} \quad (2.33)$$

while the first stage problem is

$$\max_{\pi_1} \pi_1^\top b_1 + V_2(\pi_1, c_{2-p:1}). \quad (2.34)$$

These functions are concave and therefore we can apply Dual SDDP with penalizations to these DP equations to build polyhedral approximations of these functions V_t of form

$$V_t^k(\pi_{t-1}, c_{t-1}, \dots, c_{t-p}) = \min_{0 \leq i \leq k} \theta_t^i + \langle \beta_{t,0}^i, \pi_{t-1} \rangle + \sum_{j=1}^p \langle \beta_{t,j}^i, c_{t-j} \rangle \quad (2.35)$$

at iteration k .

We conclude this section highlighting some advantages and disadvantages of Dual SDDP compared to Primal SDDP. For Dual SDDP, a stage t subproblem is coupled across scenarios $1, \dots, N_t$ and hence is larger. If c_t and A_t are random then with Dual SDDP we must store N_T , rather than one, sets of cuts at each stage. As a result of the last two issues, the computational effort per iteration is larger for Dual SDDP but fewer iterations are required. On the other hand, Dual SDDP computes deterministic valid upper bounds and provides a feasible dual policy. Also, similarly to primal SDDP which provides statistical upper bounds (for a minimization problem) on the optimal value of the Multistage Stochastic Program, Dual SDDP provides statistical lower bounds (for a maximization problem).

2.4 Dual Bounds for Periodical Multistage Stochastic Program

Consider (1.1) for the infinite horizon setting of $T = \infty$. On top of Assumptions (A0)-(A2), we make the following assumptions about the periodical behavior:

(A3) The random vectors ξ_t and ξ_{t+m} have the same distribution for $t \geq 2$ (recall that ξ_1 is deterministic).

(A4) The sequence of functions $b_t(\cdot)$, $B_t(\cdot)$, $A_t(\cdot)$ and $c_t(\cdot)$ has period m , i.e., these functions are the same for $t = \tau$ and $t = \tau + m$, $t \geq 2$.

Of course, if functions $c_t(\cdot)$ are constants, the boundness condition (A4)(ii) holds automatically. Under assumptions (A3) and (A4), the value functions $Q_t(\cdot, \cdot)$ and $Q_{t+m}(\cdot, \cdot)$, and the expected value functions $\mathcal{Q}_t(\cdot)$ and $\mathcal{Q}_{t+m}(\cdot)$, are the same for all $t \geq 2$, with $\mathcal{Q}_{m+2}(\cdot)$ set to $\mathcal{Q}_2(\cdot)$. This leads to the following periodical variant of Wald-Bellman (WB) equations for the value functions (cf., [75]):

$$\mathcal{Q}_\tau(x_{\tau-1}) = \mathbb{E}[Q_\tau(x_{\tau-1}, \xi_\tau)], \quad (2.36)$$

with

$$Q_\tau(x_{\tau-1}, \xi_\tau) = \inf_{x_\tau \geq 0} \{c_\tau^\top x_\tau + \gamma \mathcal{Q}_{\tau+1}(x_\tau) : B_\tau x_{\tau-1} + A_\tau x_\tau = b_\tau\}, \quad (2.37)$$

for $\tau = 2, \dots, m+1$, and \mathcal{Q}_{m+2} replaced by \mathcal{Q}_2 for $\tau = m+1$. It is possible to show that there exists a unique set of value functions $Q_\tau(x_{\tau-1}, \xi_\tau)$, $\mathcal{Q}_\tau(x_{\tau-1})$, $\tau = 2, \dots, m+1$, satisfying these WB equations and that these value functions are convex in $x_{\tau-1}$ (cf., [76]). The first stage solution is obtained by solving problem (1.5) with $\mathcal{Q}_2(\cdot)$ being the solution of these WB equations.

In order to solve the WB equations (2.36) - (2.37), a cutting plane algorithm was suggested in [76]. That algorithm can be viewed as a variant of the Stochastic Dual Dynamic Programming (SDDP) method. An upper bound for the optimal value in that algorithm is based on a statistical estimate of the value of the current iterate approximation of the optimal policy. When the discount factor γ is close to one, the convergence is slow and the computational effort to reduce the optimality gap becomes prohibitive. This motivates development of dual upper bounds as discussed below.

To construct the upper bound, we first set the stage number of T to be finite and then we pass to a limit. The Lagrangian of problem (1.1) is

$$L(x, \pi) = \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} c_t^\top x_t + \pi_t^\top (b_t - B_t x_{t-1} - A_t x_t) \right] \quad (2.38)$$

in variables $x = (x_1(\xi_{[1]}), \dots, x_T(\xi_{[T]}))$ and $\pi = (\pi_1(\xi_{[1]}), \dots, \pi_T(\xi_{[T]}))$ with the convention that $x_0 = 0$. Dualization of the feasibility constraints leads to the following dual of problem (1.1) (cf., [6, Section 3.2.3]):

$$\begin{aligned}
\max_{\pi} \quad & \mathbb{E} \left[\sum_{t=1}^T b_t^\top \pi_t \right] \\
\text{s.t.} \quad & A_T^\top \pi_T \leq \gamma^{T-1} c_T, \\
& A_{t-1}^\top \pi_{t-1} + \mathbb{E}_{|\xi_{[t-1]}} [B_t^\top \pi_t] \leq \gamma^{t-2} c_{t-1}, \quad t = 2, \dots, T.
\end{aligned} \tag{2.39}$$

The optimization in (2.39) is over policies $\pi_t = \pi_t(\xi_{[t]})$, $t = 1, \dots, T$. Note that in the considered framework of finite number of scenarios, problem (1.1) can be viewed as a large linear program and problem (2.2) as its dual. By the theory of linear programming we have that the optimal values of primal problem (1.1) and its dual (2.2) are equal to each other.

It is convenient for the subsequent analysis to make change of variables $\lambda_t = \gamma^{-(t-1)} \pi_t$, $t = 1, \dots, T$, in order to remove the powers of γ in the right hand sides of the feasibility constraints. In terms of variables λ_t problem (2.39) can be written as

$$\begin{aligned}
\max_{\lambda} \quad & \left\{ \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} b_t^\top \lambda_t \right] = b_1^\top \lambda_1 + \gamma \mathbb{E}_{|\xi_1} [b_2^\top \lambda_2 + \dots + \gamma \mathbb{E}_{|\xi_{[T-1]}} [b_T^\top \lambda_T]] \right\} \\
\text{s.t.} \quad & A_T^\top \lambda_T \leq c_T, \\
& A_{t-1}^\top \lambda_{t-1} + \gamma \mathbb{E}_{|\xi_{[t-1]}} [B_t^\top \lambda_t] \leq c_{t-1}, \quad t = 2, \dots, T.
\end{aligned} \tag{2.40}$$

Recall that the process ξ_1, \dots, ξ_T is assumed to be stagewise independent, and distribution of ξ_t has a finite support, $\Xi_t = \{\xi_t^1, \dots, \xi_t^{N_t}\}$, with respective probabilities p_{tj} , $j = 1, \dots, N_t$, $t = 2, \dots, T$. We denote by $A_t^j, B_t^j, c_t^j, b_t^j$ the respective scenarios corresponding to ξ_t^j .

We can write the following DP equations for the dual problem (2.40) (cf. Section 2.2). At the last stage $t = T$, given λ_{T-1} and $\xi_{[T-1]}$, we need to solve the following problem with

respect to λ_T :

$$\begin{aligned}
& \max_{\lambda_T} \mathbb{E}[b_T^\top \lambda_T] \\
& \text{s.t.} \quad A_T^\top \lambda_T \leq c_T, \\
& \quad \quad A_{T-1}^\top \lambda_{T-1} + \gamma \mathbb{E}[B_T^\top \lambda_T] \leq c_{T-1}.
\end{aligned} \tag{2.41}$$

In terms of scenarios the above problem can be written as

$$\begin{aligned}
& \max_{\lambda_{T1}, \dots, \lambda_{TN_T}} \sum_{j=1}^{N_T} p_{Tj} (b_T^j)^\top \lambda_{Tj} \\
& \text{s.t.} \quad A_{Tj}^\top \lambda_{Tj} \leq c_{Tj}, \quad j = 1, \dots, N_T, \\
& \quad \quad A_{T-1}^\top \lambda_{T-1} + \gamma \sum_{j=1}^{N_T} p_{Tj} (B_T^j)^\top \lambda_{Tj} \leq c_{T-1}.
\end{aligned} \tag{2.42}$$

The optimal value $V_T(\lambda_{T-1}, \xi_{T-1})$ and an optimal solution $(\bar{\lambda}_{T1}, \dots, \bar{\lambda}_{TN_T})$ of problem (2.42) are functions of vectors λ_{T-1} and c_{T-1} and matrix A_{T-1} . And so on going backward in time, using the stagewise independence assumption, we can write the respective dynamic programming equations for $t = T - 1, \dots, 2$, as

$$\begin{aligned}
& \max_{\lambda_{t1}, \dots, \lambda_{tN_t}} \sum_{j=1}^{N_t} p_{tj} [(b_t^j)^\top \lambda_{tj} + \gamma V_{t+1}(\lambda_{tj}, \xi_{tj})] \\
& \text{s.t.} \quad A_{t-1}^\top \lambda_{t-1} + \gamma \sum_{j=1}^{N_t} p_{tj} (B_t^j)^\top \lambda_{tj} \leq c_{t-1},
\end{aligned} \tag{2.43}$$

with $V_t(\lambda_{t-1}, \xi_{t-1})$ being the optimal value of problem (2.43). Note that unlike the primal problem, the dynamic equations of the dual problem do not decompose into individual scenarios - the optimization problem (2.43) is formulated jointly with respect to the dual variables $\lambda_{t1}, \dots, \lambda_{tN_t}$.

Finally at the first stage the following problem should be solved

$$\max_{\lambda_1} b_1^\top \lambda_1 + \gamma V_2(\lambda_1). \tag{2.44}$$

Note that if A_t and c_t are deterministic, then $V_{t+1}(\lambda_t)$ does not depend on ξ_t .

Now consider the infinite horizon case $T = \infty$. Consider first when $m = 1$. In that

case the random process ξ_t is i.i.d. with the corresponding scenarios $\xi^j = (A^j, B^j, c^j, b^j)$, $j = 1, \dots, N$, which do not depend on $t \geq 2$. The WB equations for the value function $V(\lambda_j, \xi^j)$ of the dual problem then become

$$V(\lambda, \xi^j) = \sup_{\lambda_1, \dots, \lambda_N} \left\{ \sum_{k=1}^N p_k [(b^k)^\top \lambda_k + \gamma V(\lambda_k, \xi^k)] : (A^j)^\top \lambda + \gamma \sum_{k=1}^N p_k (B^k)^\top \lambda_k \leq c^j \right\}, \quad (2.45)$$

$j = 1, \dots, N$. Note that solution $V(\lambda, \xi^j)$ of this equation is concave in λ , and if $A^j \equiv A$ and $c^j \equiv c$ are deterministic, then $V(\lambda)$ does not depend on ξ^j and is given by the equation

$$V(\lambda) = \sup_{\lambda_1, \dots, \lambda_N} \left\{ \sum_{k=1}^N p_k [(b^k)^\top \lambda_k + \gamma V(\lambda_k)] : A^\top \lambda + \gamma \sum_{k=1}^N p_k (B^k)^\top \lambda_k \leq c \right\}. \quad (2.46)$$

Consider the general case of $m \geq 1$. Then the WB equations for the value functions of the dual problem are

$$V_\tau(\lambda_{\tau-1}, \xi_{\tau-1}^j) = \sup_{\lambda_{\tau 1}, \dots, \lambda_{\tau N_\tau}} \left\{ \sum_{k=1}^{N_\tau} p_{\tau k} [(b_\tau^k)^\top \lambda_{\tau k} + \gamma V_{\tau+1}(\lambda_{\tau k}, \xi_\tau^k)] : \right. \\ \left. (A_{\tau-1}^j)^\top \lambda_{\tau-1} + \gamma \sum_{k=1}^{N_\tau} p_{\tau k} (B_\tau^k)^\top \lambda_{\tau k} \leq c_{\tau-1}^j \right\}, \quad (2.47)$$

for $\tau = 2, \dots, m+1$, and V_{m+2} replaced by V_2 . If $A_\tau^j \equiv A_\tau$ and $c_\tau^j \equiv c_\tau$, $\tau = 2, \dots, m+1$, are deterministic, then value functions do not depend on A_τ and c_τ and are given by equations

$$V_\tau(\lambda_{\tau-1}) = \sup_{\lambda_{\tau 1}, \dots, \lambda_{\tau N_\tau}} \left\{ \sum_{k=1}^{N_\tau} p_{\tau k} [(b_\tau^k)^\top \lambda_{\tau k} + \gamma V_{\tau+1}(\lambda_{\tau k})] : A_{\tau-1}^\top \lambda_{\tau-1} + \gamma \sum_{k=1}^{N_\tau} p_{\tau k} (B_\tau^k)^\top \lambda_{\tau k} \leq c_{\tau-1} \right\}, \quad (2.48)$$

for $\tau = 2, \dots, m+1$, and V_{m+2} replaced by V_2 .

Remark 2.4.1. Under the specified assumptions, there is no duality gap between the primal and dual problems also in the case of the infinite number of stages (when $T = \infty$). Indeed, by assumption (A2)(i) the considered decision variables $x_t(\cdot)$ can be restricted to the respective bounded sets \mathcal{X}_t . Together with the periodical assumption this implies that $x_t(\cdot)$ can be viewed as bounded for all t and almost every realization of the random data

process. By the boundedness of the cost functions (assumption (A2)(ii)), this implies that there is a constant $\kappa > 0$ such that $|c_t(\cdot)x_t(\cdot)| \leq \kappa$ for all t and almost every realization of the random data process. Thus we can bound the difference between the optimal values of problem (1.1) for the infinite and finite number T of stages as follows

$$\left| \mathbb{E} \left[\sum_{t=T+1}^{\infty} \gamma^{t-1} c_t^\top x_t \right] \right| \leq \sum_{t=T+1}^{\infty} \gamma^{t-1} \kappa = \kappa \gamma^T / (1 - \gamma). \quad (2.49)$$

Let ϑ_T be the optimal value of the primal problem (1.1). By (2.49) we have for $T < T'$ that

$$|\vartheta_{T'} - \vartheta_T| \leq \kappa \gamma^T / (1 - \gamma). \quad (2.50)$$

It follows that ϑ_T is a Cauchy sequence, and hence tends to a finite limit ϑ_∞ as $T \rightarrow \infty$.

By invoking the no duality gap property of (finite dimensional) linear programs we have that ϑ_T is also the optimal value of the dual problem for any finite T . It follows that the optimal value of the dual problem tends to the same ϑ_∞ as $T \rightarrow \infty$.

2.5 Periodical Dual SDDP

In this section we provide a detailed discussion of the periodical Dual SDDP method with period $m \geq 1$. In particular, we elaborate on the trial point selection in the forward step and cutting plane method in the backward step of the algorithm. We refer to 1 for the scheme of Periodical Dual SDDP.

Algorithm 1 Periodical Dual SDDP with penalization

- 1: Given sample size N_τ and discretizations $\{u_\tau^j, v_\tau^j, b_\tau^j, A_\tau^j, B_\tau^j, C_\tau^j\}_{j=1}^{N_\tau}$, for $\tau = 2, \dots, m+1$.
 - 2: Initialization of cutting planes: $\mathfrak{R}_\tau^0 = \text{LargeBound}$, $\tau = 2, \dots, m+1$, $\mathfrak{R}_{m+2}^0 = \mathfrak{R}_2^0$.
 - 3: **for** $k = 1, 2, \dots$ **do**
 - 4: **for** $t = 1, \dots, T$ **do** ▷ Forward Pass
 - 5: **if** $t = 1$ **then** $\tau = 1$
 - 6: **else** $\tau \equiv (t \bmod m)$
 - 7: **end if**
 - 8: $\{\lambda_{tj}\}_{j=1}^{N_\tau} = \arg \max \left\{ \sum_{j=1}^{N_\tau} p_{\tau j} b_\tau^j \lambda_{tj} + \gamma \mathfrak{R}_{\tau+1}^{k-1}(\lambda_t) - r_\tau^k \nu_\tau : \right.$
 - 9:
$$A_{\tau-1}^\top \bar{\lambda}_{t-1} + \gamma \sum_{j=1}^{N_\tau} p_{\tau j} B_\tau^j \lambda_{tj} - \nu_\tau \leq c_{\tau-1} \left. \right\}$$
 - 9: Select forward solutions $\lambda'_t \leftarrow \lambda_{t\hat{j}}$
 - 10: **end for**
 - 11: Trial points selection: $(\bar{\lambda}_1, \dots, \bar{\lambda}_m) \leftarrow (\lambda'_\ell, \lambda'_{\ell+1}, \dots, \lambda'_{\ell+m-1})$
 - 12: **for** $\tau = m+1, \dots, 2$ **do** ▷ Backward Pass
 - 13: $(\bar{V}_\tau(\bar{\lambda}_{\tau-1}), g_\tau) = \max \left\{ \sum_{j=1}^{N_\tau} p_{\tau j} b_\tau^j \lambda_{\tau j} + \gamma \mathfrak{R}_{\tau+1}^k(\lambda_\tau) - r_\tau^k \nu_\tau : \right.$
 - 13:
$$A_{\tau-1}^\top \bar{\lambda}_{\tau-1} + \gamma \sum_{j=1}^{N_\tau} p_{\tau j} B_\tau^j \lambda_{\tau j} - \nu_\tau \leq c_{\tau-1} \left. \right\}$$
 - 14: Update cutting planes: $\mathfrak{R}_\tau^k \leftarrow \left\{ \alpha \in \mathfrak{R}_\tau^{k-1} : \alpha \leq \sum_{j=1}^{N_{\tau-1}} p_{\tau-1j} \psi_{\tau j}(\lambda_{\tau-1j}), \right.$
 - 14:
$$\psi_{\tau j}(\lambda_{\tau-1j}) = g_\tau^\top (\lambda_{\tau-1j} - \bar{\lambda}_{\tau-1}) + \bar{V}_\tau(\bar{\lambda}_{\tau-1}) \left. \right\}$$
 - 15: **if** $\tau = 2$ **then** update cutting planes: $\mathfrak{R}_{m+2}^k \leftarrow \left\{ \alpha \in \mathfrak{R}_{m+2}^{k-1} : \alpha \leq \sum_{j=1}^{N_{m+1}} p_{m+1j} \psi_{2j}(\lambda_{m+1j}), \right.$
 - 15:
$$\psi_{2j}(\lambda_{m+1j}) = g_2^\top (\lambda_{m+1j} - \bar{\lambda}_1) + \bar{V}_2(\bar{\lambda}_1) \left. \right\}$$
 - 16: **end if**
 - 17: **end for**
 - 18: Deterministic bound $V_1^k = \max \{ b_1^\top \lambda_1 + \gamma \mathfrak{R}_2^k(\lambda_1) \}$
 - 19: **end for**
-

2.5.1 Trial Points Selection

In the forward step, we fix a finite value for the number of stages T , and solve the corresponding T -stage problems. The obtained feasible solutions of the dual problem are state variables $\{\lambda_{tj}\}$. Note that in the dual setting each optimization problem at stage $t = 2, \dots, T$, is not separable with respect to solution corresponding to each (discretized) sample. For each stage $t \geq 2$, by selecting $\lambda'_t := \lambda_{t\hat{j}}$ with probability $p_{\tau\hat{j}}$, $\hat{j} \in \{1, \dots, N_\tau\}$, where $\tau = t \pmod{m}$, we construct a set of solutions $\{\lambda'_t : t = 2, \dots, T\}$. Next, consider T stages divided into consecutive groups with period m , that is, $\{(\ell, \ell + 1, \dots, \ell + m - 1) : \ell = 1, m + 1, 2m + 1, 3m + 1, \dots, \ell + m - 1 \leq T\}$. Randomly select ℓ from $\{1, m + 1, 2m + 1, \dots, \ell + m - 1 \leq T\}$, then we construct a group of trial points $\{\bar{\lambda}_{\ell+\tau-1} : \tau = 1, \dots, m\}$.

2.5.2 Cutting Plane Algorithm

To deal with the issue of possible violation of the Relatively Complete Recourse (RCR), a penalty term ν is introduced for the dual problem with objective coefficient r . With the penalty term, the value function at each stage τ is finite valued on a compact set formed by linear constraints, thus maximum is attainable. The WB equations can be written as, for $\tau = m + 1, \dots, 2$,

$$\begin{aligned} V_\tau(\lambda_{\tau-1}) = & \max_{\lambda_\tau, \nu_\tau \geq 0} \sum_{j=1}^{N_\tau} p_{\tau j} \left(b_\tau^j \top \lambda_{\tau j} + \gamma V_{\tau+1}(\lambda_{\tau j}) \right) - r_\tau \top \nu_\tau \\ \text{s.t.} \quad & A_{\tau-1} \top \lambda_{\tau-1} + \gamma \sum_{j=1}^{N_\tau} p_{\tau j} B_\tau^j \top \lambda_{\tau j} - \nu_\tau \leq c_{\tau-1}, \end{aligned} \quad (2.51)$$

where $V_{m+2}(\cdot)$ is replaced by $V_2(\cdot)$. For $\tau = 1$, the problem is deterministic and can be written as

$$V_1 = \max_{\lambda_1} b_1 \top \lambda_1 + \gamma V_2(\lambda_1). \quad (2.52)$$

For each stage $\tau \in \{1, 2, \dots, m + 1\}$, given a current upper approximation $\mathfrak{V}_{\tau+1}(\cdot)$

of the value function $V_{\tau+1}(\cdot)$ and a trial point $\bar{\lambda}_{\tau-1}$, new cuts $\{\psi_{\tau j}(\cdot)\}$ are constructed by computing the (sub)gradient g_τ at $\bar{\lambda}_{\tau-1}$ of the current estimate of the value function. That is,

$$\psi_{\tau j}(\lambda_{\tau-1j}) := g_\tau^\top (\lambda_{\tau-1j} - \bar{\lambda}_{\tau-1}) + \bar{V}_\tau(\bar{\lambda}_{\tau-1}), \quad j = 1, \dots, N_{\tau-1}, \quad (2.53)$$

and a new supporting plane for $V_\tau(\cdot)$ at $\bar{\lambda}_{\tau-1}$ is generated by $\{\psi_{\tau j}\}$ as

$$l_\tau(\lambda_{\tau-1}) := \sum_{j=1}^{N_{\tau-1}} p_{\tau-1j} \psi_{\tau j}(\lambda_{\tau-1j}), \quad (2.54)$$

where

$$\begin{aligned} \bar{V}_\tau(\bar{\lambda}_{\tau-1}) = \max_{\lambda_\tau, \nu_\tau \geq 0} & \sum_{j=1}^{N_\tau} p_{\tau j} b_\tau^j \lambda_{\tau j} + \gamma \mathfrak{V}_{\tau+1}(\lambda_\tau) - r_\tau^\top \nu_\tau \\ \text{s.t.} & A_{\tau-1}^\top \bar{\lambda}_{\tau-1} + \gamma \sum_{j=1}^{N_\tau} p_{\tau j} B_\tau^j \lambda_{\tau j} - \nu_\tau \leq c_{\tau-1}, \end{aligned} \quad (2.55)$$

with $\lambda_\tau = [\lambda_{\tau 1}, \dots, \lambda_{\tau j}, \dots, \lambda_{\tau N_\tau}]$. Then the collection of supporting planes of $\mathfrak{V}_\tau(\cdot)$ is updated by $\mathfrak{V}_\tau(\cdot) \leftarrow \min\{\mathfrak{V}_\tau(\cdot), l_\tau(\cdot)\}$. Specifically, cutting plane approximation for value function at stage $m+2$ is equal to the approximation at stage 2 by the periodic property.

2.5.3 Trust-bound Strategy

Note that for each $\tau \in \{2, \dots, m+1\}$, $\mathfrak{V}_\tau(\cdot)$ is formed by the minimum of piecewise linear functions, and hence problem (2.55) can be formulated as a linear programming problem. Suppose at iteration k in the backward step, we are solving the following problem at stage τ :

$$\begin{aligned} \max & \sum_{j=1}^{N_\tau} p_{\tau j} b_\tau^j \lambda_{\tau j} + \gamma \alpha_{\tau+1} - r_\tau^\top \nu_\tau \\ \text{s.t.} & A_{\tau-1}^\top \bar{\lambda}_{\tau-1} + \gamma \sum_{j=1}^{N_\tau} p_{\tau j} B_\tau^j \lambda_{\tau j} - \nu_\tau \leq c_{\tau-1}, \\ & \alpha_{\tau+1} \leq l_{\tau+1}^s(\lambda_\tau), \quad s = 1, \dots, k, \end{aligned} \quad (2.56)$$

where $l_{\tau+1}^s(\cdot)$ denotes the supporting plane generated at iteration $s \leq k$. Consequently, we obtain optimal value of $\alpha_{\tau+1}$, denoted by $\tilde{\alpha}_{\tau+1}^k$, as the best upper approximation for value function at stage $\tau + 1$ for current iteration.

If the algorithm restarts after the k -th iteration, then all the generated cuts are eliminated and for each stage $\tau, \tau = 2, \dots, m + 1$, value function \bar{V}_τ is set to $\tilde{\alpha}_\tau^k$. The algorithmic scheme is the same after each restart.

2.6 Numerical Results

In this section, we report numerical results obtained by applying Dual SDDP and its periodical counterpart to the inventory problem and to the Brazilian interconnected power system problem. In addition, we present numerical sensitivity analysis on a parameterized inventory problem. For problems with periodical data process, we test the algorithms with different discount factors. In both models, convergence is measured by the relative gap computed by deterministic upper bound of the dual problem and deterministic lower bound of the primal problem unless stated otherwise.

2.6.1 Inventory Model

Consider the classical inventory model (cf., [77])

$$\begin{aligned} \min_{y_t \geq x_{t-1}} \quad & \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} \left(c_t(y_t - x_{t-1}) + b_t[D_t - y_t]_+ + h_t[y_t - D_t]_+ \right) \right] \\ \text{s.t.} \quad & x_t = y_t - D_t, \quad t = 1, \dots, T, \end{aligned} \quad (2.57)$$

where c_t, b_t, h_t are ordering cost, backorder penalty cost and holding cost per unit, respectively and $[\cdot]_+ := \max\{\cdot, 0\}$. Here x_t denotes the current inventory level, in particular x_0 denotes the initial level, $y_t - x_{t-1}$ represents the order quantity at stage t , and D_1, \dots, D_T is the demand process.

In order to formulate model (2.57) as a linear program and hence to construct its dual,

we proceed as follows. An equivalent formulation of (2.57) is to replace $[D_t - y_t]_+$ and $[y_t - D_t]_+$ with $w_t \geq 0$ and $v_t \geq 0$, respectively, and simultaneously to add feasibility constraints $y_t + w_t \geq D_t$ and $y_t - v_t \leq D_t$. The Lagrangian of this problem then becomes

$$L(p, d) = \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} \left(c_t(y_t - x_{t-1}) + b_t w_t + h_t v_t + \Psi_t(p, d) \right) \right], \quad (2.58)$$

where

$$\Psi_t(p, d) := \pi_t(D_t + x_t - y_t) + \mu_t(y_t - x_{t-1}) + u_t(y_t + w_t - D_t) + s_t(D_t + v_t - y_t),$$

$p := (p_1(\xi_1), \dots, p_T(\xi_T))$, $d := (d_1(\xi_1), \dots, d_T(\xi_T))$ with $p_t(\xi_t) := (x_t(\xi_t), y_t(\xi_t), w_t(\xi_t), v_t(\xi_t))$ and $d_t(\xi_t) := (\pi_t(\xi_t), \mu_t(\xi_t), u_t(\xi_t), s_t(\xi_t))$. Dualization of feasibility constraints and change of variables $d_t \leftarrow \gamma^{-(t-1)} d_t$ result in the following periodical multistage linear stochastic inventory dual model

$$\begin{aligned} \max_{\pi, \mu, u, s} \quad & \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} D_t (\pi_t - u_t + s_t) \right] - c_1 x_0 - \mu_1 x_0 \\ \text{s.t.} \quad & \pi_t - \mu_t - u_t + s_t = c_t, \\ & -u_t \leq b_t, \quad -s_t \leq h_t, \quad \mu_t, u_t, s_t \leq 0, \quad t = 1, \dots, T, \\ & -\gamma^{-1} \pi_{t-1} + \mathbb{E}[\mu_t] = -c_t, \quad t = 2, \dots, T. \end{aligned} \quad (2.59)$$

Next we present results for finite-horizon setting (including sensitivity analysis) and infinite-horizon setting, which are based on various assumption and modeling of the underlying demand process. For all experiments, we set $c_t = \cos(\frac{t}{6}) + 1.5$, $b_t = 2.8$, $h_t = 0.2$, $\forall t$ and an initial condition $x_0 = 10$.

Dual SDDP under Finite-horizon Setting

We assume the following settings for the finite-horizon setting. The demands are discrete random variables such that at the first stage, D_1 is assumed to be deterministic with $D_1 =$

5.5 and $D_t^j = \alpha + \beta \xi_t^j$, $t = 2, \dots, T$, where $\alpha = 9.0, \beta = 0.6$. Values $\xi_t^j, j = 1, \dots, 50$, are generated by taking random samples of size 50 from the uniform distribution on the interval $[0, 1]$ independently for each $t = 2, \dots, T$. The assigned probabilities $p_{tj} = 0.02$ are the same for all $t = 2, \dots, T$ and j . Moreover, under this setting the discount factor γ is 1.

We run Primal and Dual SDDP on a larger instance with $T = 100$ and $N_t = 100$ for 600 iterations. We terminate the Primal SDDP when the gap is < 0.1 , where the gap is defined as $\frac{Ub-Lb}{Ub}$ with Ub and Lb corresponding to upper and lower bounds computed by the Primal SDDP along iterations. Specifically, the lower bound Lb is the optimal value of the first stage problem and the upper bound Ub is the upper end of a 97.5%-one-sided confidence interval on the optimal value obtained using the sample of total costs computed by all previous forward passes.

The evolution of the upper bounds computed along the iterations of Dual SDDP (with penalizations $v_{t,k} = 1000$) and of the upper and lower bounds computed by Primal SDDP are reported in Table 2.1 for iterations 2, 3, 5, 10, 50, 100, 200, 300, 400, 500, and 600. We observe that for the first 100 iterations, the upper bound decreases more quickly with the Dual SDDP. More precisely, we fix confidence levels $\varepsilon = 0.2, 0.15, 0.1, 0.05, 0.01$, and for each confidence level, we compute the time needed, running Primal and Dual SDDP in parallel, to obtain a solution with relative accuracy ε stopping the algorithm when the upper bound Ub_D computed by Dual SDDP and the lower bound Lb , computed by Primal SDDP, satisfies $(Ub_D-Lb)/Ub_D < \varepsilon$. The results are reported in Table 2.2. In this table, we also report the time needed to obtain a solution of relative accuracy ε using only the information provided by Primal SDDP, stopping the algorithm when $(Ub-Lb)/Ub < \varepsilon$.

We observe that when ε is small (0.05 and 0.01) the smallest CPU time is obtained combining Primal SDDP with Dual SDDP with penalizations. For $\varepsilon = 0.05$ and 0.01, 600 iterations are even not enough to get a solution of relative accuracy ε using Primal SDDP.

In Figure 2.2, we report the cumulative CPU time along iterations of all methods. We

Table 2.1: Lower bound Lb and upper bound Ub computed by Primal SDDP and upper bounds computed by Dual SDDP penalties $v_{t,k} = 1000$ along iterations.

Iteration	Primal SDDP Lb	Primal SDDP Ub	Dual SDDP with penalties
2	656.4	25 443	20 015
3	713.1	19 340	20 012
5	3361.8	14 800	19 993
10	5330.1	10 662	16 452
50	5483.1	6 594.5	5500.9
100	5483.5	6 039.2	5484.8
200	5483.6	5 762.4	5484.2
300	5483.7	5 671.0	5484.0
400	5483.7	5 625.3	5483.9
500	5483.7	5 597.9	5483.8
600	5483.7	5 579.9	5483.8

Table 2.2: Time needed (in seconds) to obtain a solution of relative accuracy ε with Primal SDDP and Dual SDDP with penalties $v_{t,k} = 1000$.

ε	Primal SDDP	Dual SDDP with penalties $v_{t,k} = 1000$
0.2	300.2	35.8
0.15	459.8	41.2
0.1	825.6	48.3
0.05	2366.2	61.5
0.01	-	103.2

see that each iteration requires a similar computational bulk and the CPU time increases exponentially with the number of iterations.

Dual SDDP for Inventory Problem with Interstage Dependent Cost Coefficients

We consider a variant of the inventory model (2.57) with interstage dependent cost coefficients. The cost c_t is modeled by a Markovian random process such as

$$c_t = \epsilon_t \cdot (\theta c_{t-1} + \mu), \quad t = 1, \dots, T, \quad (2.60)$$

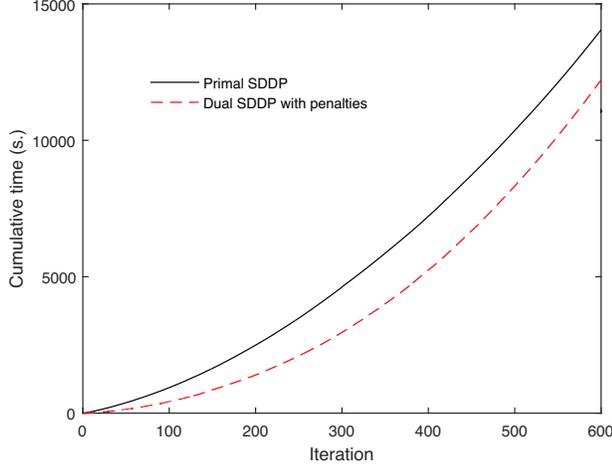


Figure 2.2: Cumulative CPU time along iterations of Primal SDDP and Dual SDDP with penalizations $v_{t,k} = 1000$.

where ϵ_t follows the log-normal distribution with mean 1.0 and variance 2.0 for all t and $\theta \in (0, 1)$, $\mu > 0$. In this model, we assume all other parameters, b_t , h_t and the demand D_t are deterministic. As mentioned in 2.3.3, the dual formulation of the inventory problem brings the random cost coefficients into the right-hand side of the model. By treating the random costs as state variables, the value functions become concave and can be directly solved by Dual SDDP with penalizations. This is in contrast to the primal problem where such approach destroys convexity of the respective value functions.

To illustrate the convergence of Dual SDDP applied to this problem with interstage dependent random process, we ran the algorithm on the $T = 12$ -stage inventory problem with self-generated data under parameter $(\theta, \mu) = (0.001, 0.1)$. Moreover, we also present numerical results obtained solving the primal model with a Markovian approximation variant of SDDP, as described in ([22]).

We discretize the random process in both the primal and dual models with $N_t = 100$ for $t = 2, \dots, T$. For the dual model, $\{\epsilon_t^j\}_{t,j}$ are generated in a stage-wise independent manner from the lognormal distribution. To apply Markovian SDDP on the primal, we deploy 100,000 sample paths to train the transition matrix with 100 Markov states.

Figure 2.3 shows the evolution of the deterministic primal (lower) bound and the dual

Table 2.3: Convergence results of MC-primal and TS-dual

Iter.	Primal LB.	Dual UB.	Gap(%)
10	53.4104	73.6759	27.51
100	56.8763	57.5192	1.11
180	56.8763	57.4734	1.04

(upper) bound while 2.3 details the values of the bounds and the relative gaps throughout different iterations. It can be noticed that after some iterations, the Markovian primal problem and the dual problem converge with relatively small gap, which confirms the viability of the dual approach. In addition, our dual approach provides a valid upper bound for problems with interstage dependent cost processes, while for Markovian SDDP, it is still unclear how to compute a valid upper bound for such problems.

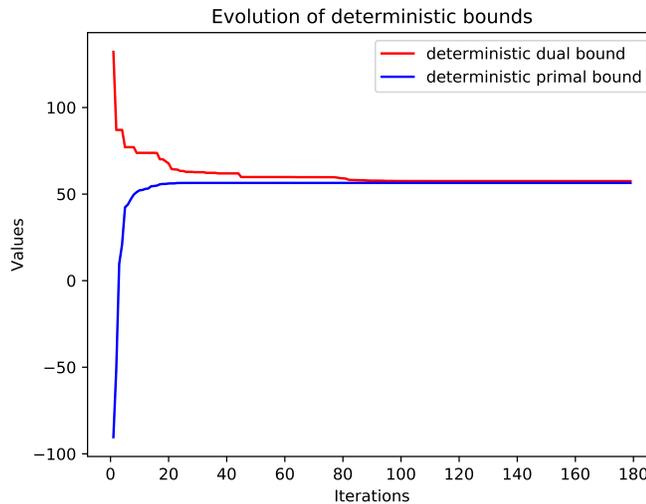


Figure 2.3: Evolution of primal and dual bounds for interstage dependent cost process

Sensitivity Analysis

In the classical setting the demand process is assumed to be stagewise independent, i.e., D_{t+1} is assumed to be independent of $D_{[t]} = (D_1, \dots, D_t)$ for $t = 1, \dots, T - 1$. In order to capture the autocorrelation structure of the demand process it is tempting to model it as, say first order, autoregressive process $D_t = \mu + \phi D_{t-1} + \epsilon_t$, where errors ϵ_t are assumed to be a sequence i.i.d (independent identically distributed) random variables. However this

approach may result in some of the realizations of the demand process to be negative, which of course does not make sense. One way to deal with this is to make the transformation $Y_t := \log D_t$ and to model Y_t as an autoregressive process. A problem with this approach is that it leads to nonlinear equations for the original process D_t , which makes it difficult to use in the numerical algorithms discussed below.

We assume that the demand is modeled as the following multiplicative autoregressive process

$$D_t = \epsilon_t(\phi D_{t-1} + \mu), \quad t = 1, \dots, T, \quad (2.61)$$

where $\phi \in (0, 1)$, $\mu \geq 0$ are parameters and $D_0 \geq 0$ is given. The errors ϵ_t are i.i.d with log-normal distributions having means and standard deviations given by $\mathbb{E}[\epsilon_t] = 1$ and $\text{Var}(\epsilon_t) = \sigma^2 > 1$, respectively. This guarantees that all realizations of the demand process are positive. It is possible to view (2.61) as a linearization of the log-transformed process $\log D_t$, with parameters ϕ and μ are estimated from the data (cf., [78]). See in Section 2.7 for a discussion of statistical properties of the process (2.61).

The process (2.61) involves parameters ϕ and μ which are estimated from the data. As such, these parameters are subject to estimation errors. This raises the question of the sensitivity of the optimal value $\vartheta(\theta) = \vartheta(\phi, \mu)$ of the corresponding problem (??) viewed as a function of $\theta = (\phi, \mu)$. Using Proposition 2.2.2, derivatives $\partial\vartheta(\phi, \mu)/\partial\phi$ and $\partial\vartheta(\phi, \mu)/\partial\mu$ are given by

$$\partial\vartheta(\phi, \mu)/\partial\phi = \partial L(\bar{x}, \bar{y}, \bar{\pi})/\partial\phi = \mathbb{E} \left[\sum_{t=1}^T \bar{\pi}_t \epsilon_t D_{t-1} \right], \quad (2.62)$$

$$\partial\vartheta(\phi, \mu)/\partial\mu = \partial L(\bar{x}, \bar{y}, \bar{\pi})/\partial\mu = \mathbb{E} \left[\sum_{t=1}^T \bar{\pi}_t \epsilon_t \right], \quad (2.63)$$

where (\bar{x}, \bar{y}) is an optimal solution of the primal problem and $\bar{\pi}$ are the corresponding Lagrange multipliers. With these derivatives at hand, asymptotic distributions of the estimates of ϕ and μ can be translated into the asymptotics of the optimal value in a straightforward way by application of the Delta Theorem (cf., [79]).

Consider an instance of the model with $T = 10$ stages and with optimal value $\vartheta(\theta)$ for the two-dimensional parameter vector $\theta = (\phi, \mu)$. Our goal is to compute derivatives (2.62) and (2.63) solving the primal and dual problems by respectively Primal and Dual SDDP.

We consider 4 instances with $(\phi, \mu) = (0.01, 0.1), (0.01, 3.0), (0.001, 0.1),$ and $(0.001, 3.0)$. The remaining parameters of these instances are those from the previous section. We discretize both the primal and dual problem into $N_t = 100$ samples for each stage $t = 2, \dots, 10$. We take the relative error $\varepsilon = 0.01$ for the stopping criterion and use 10 000 Monte Carlo simulations to estimate the expectations in (2.62), (2.63). For Primal SDDP, the upper bound Ub and lower bound Lb at termination are given in Table 2.4 for the four instances.

Table 2.4: Upper and lower bounds at the last iteration of Primal SDDP.

Bound	Instance 1	Instance 2	Instance 3	Instance 4
Ub	17.9176	478.687	15.3940	404.242
Lb	17.9163	475.017	15.3927	402.913

The optimal mean values of Lagrangian multipliers for the demand constraints computed, for a given stage $t \geq 2$, averaging over the 10 000 values obtained simulating 10 000 forward passes after termination, are given in 2.5. In this table, `LM` stands for the multipliers obtained using Primal SDDP as explained in Remark 2.2.1 and `Dual` stands for the multipliers obtained using Dual SDDP with penalties. The fact that the multipliers obtained are close from both methods coins the validity of the two alternatives to compute derivatives of the value function of a multistage stochastic linear program.

With optimal dual solutions $\{\bar{\pi}_t\}$ and the realizations of $\{D_t\}$ and $\{\epsilon_t\}$ at hand, we are able to compute the sensitivity of the optimal value with respect to ϕ and μ , using (2.62) and (2.63), with expectations estimated for 10 000 Monte Carlo simulations. We benchmark our method against the finite-difference method. Specifically, for value function ϑ , the finite-difference method approximates the derivative with respect to u_0 by $v'(u_0) \approx \frac{v(u_0+\delta)-v(u_0-\delta)}{2\delta}$ for some small δ .

Table 2.5: Comparison between optimal Lagrange multipliers from Primal SDDP and Dual SDDP with penalties.

Stage	Instance 1		Instance 2		Instance 3		Instance 4	
	LM	Dual	LM	Dual	LM	Dual	LM	Dual
2	0.2465	0.2373	1.6701	1.66959	0.0444	0.0328	1.666	1.666
3	0.3218	0.31095	1.4098	1.4120	0.1421	0.1340	1.406	1.409
4	0.3268	0.3221	0.9862	0.9861	0.19439	0.18974	0.984	0.984
5	0.3086	0.3058	0.6330	0.6329	0.2145	0.2128	0.6327	0.6327
6	0.3408	0.3412	0.49998	0.499897	0.2708	0.2717	0.4999	0.4998
7	0.5026	0.5051	0.63397	0.63397	0.4378	0.4418	0.6339	0.6339
8	0.7047	0.7049	0.8348	0.8340	0.6404	0.6413	0.8349	0.8334
9	0.8985	0.9032	1.0322	1.0343	0.83501	0.8401	1.0315	1.0343
10	1.1022	1.1037	1.2302	1.2365	1.03926	1.04091	1.23	1.23

The sensitivity of the optimal value of the inventory problem with respect to (ϕ, μ) is displayed in 2.6. In this table, $S-\phi$ and $S-\mu$ denote the derivatives with respect to ϕ and μ computed by our method, and $fd-\phi$, $fd-\mu$ denote the derivatives computed by the finite-difference method. In order to measure the difference between the two methods, we also compute $S\text{-gap-}\phi$ and $S\text{-gap-}\mu$, where $S\text{-gap-}\phi := \frac{|fd-\phi - S-\phi|}{|fd-\phi|} \times 100\%$ and $S\text{-gap-}\mu := \frac{|fd-\mu - S-\mu|}{|fd-\mu|} \times 100\%$.

Table 2.6: Sensitivity of the optimal value with respect to ϕ and μ by the two methods.

Instance	$fd-\phi$	$S-\phi$	$S\text{-gap-}\phi(\%)$	$fd-\mu$	$S-\mu$	$S\text{-gap-}\mu(\%)$
1	403.604	401.094	0.622	164.578	164.158	0.255
2	10716.111	10671.262	0.419	185.346	184.847	0.270
3	269.514	269.443	0.026	134.646	134.463	0.136
4	7780.570	7770.274	0.132	158.017	158.001	0.0101

We observe that the derivatives obtained by both methods are close to each other, especially when ϕ and μ are small. This is because small ϕ and μ gives rise to less variability in the demand. Note also that the finite-difference method is more time consuming since it requires computing the optimal value twice. Instead, our method only needs to solve the model once. Moreover, computing the Lagrange multipliers does not significantly consume CPU time, as they are generated as a by-product of Primal SDDP. Alternatively, as discussed above, one can compute the optimal multipliers using Dual SDDP with penalties.

Another drawback of the finite-difference method lies in its numerical instability. Indeed, the method is more accurate when δ is very small. However, the division by a very small number generates bias while our approach is more stable.

Periodical Dual SDDP

Consider model (2.59) with infinite horizon $T = \infty$ and period $m = 12$. At the first stage, D_1 is assumed to be deterministic with $D_1 = 5.5$. At stages $\tau = 2, \dots, m + 1$, we assume the following setting. The demands are discrete random variables such that $D_\tau^j = \alpha + \beta \xi_\tau^j$, where $\alpha = 9.0, \beta = 0.6$, and values $\xi_\tau^j, j = 1, \dots, 50$, are generated by taking random samples of size 50 from the uniform distribution on the interval $[0, 1]$ independently for each $\tau = 2, \dots, m + 1$. The assigned probabilities $p_{\tau j} = 0.02$ are the same for all τ and j . $c_\tau = \cos(\frac{\pi}{6}\tau) + 1.5$ for $\tau = 2, \dots, m + 1$. For $t \geq m + 2$ the above setting is repeated periodically with period $m = 12$. Optimization of (2.59) is performed over the respective policies satisfying the feasibility constraints.

We conduct experiments with the following values of the discount factor: $\gamma = 0.8, \gamma = 0.9906, \gamma = 0.9990, \gamma = 0.9999$. These settings aim at investigating the rate of convergence in an empirical sense when discount factor approaches one. To solve the dual problem we apply the periodical Dual SDDP algorithm with penalization (see 1 in Section 2.5), equipped with penalty parameter sequence: $r_t^k = 10^4, t = 1, 2, \dots$, for every iteration k .

In 2.7, we use ‘Primal-PSDDP’ and ‘Dual-PSDDP’ to denote the periodical Primal SDDP and Dual SDDP algorithms, respectively. Deterministic (upper) bounds of the dual and deterministic (lower) bound of the primal problem are represented by (D.-UB.) and (D.-LB), respectively. For example, Dual-PSDDP(D.-UB.) refers to the deterministic (upper) bound output from periodical Dual SDDP. Gap(%) is computed by

$$\frac{\text{Dual-PSDDP(D.-UB.)} - \text{Primal-PSDDP(D.-LB)}}{\text{Dual-PSDDP(D.-UB.)}} \times 100\%. \quad (2.64)$$

Different rows of the table are associated with different discount factors. At each row, we display deterministic bounds of the primal and dual problems when the algorithm stabilizes. The results in the table suggest that as the discount factor approaches one, the convergence slows down. This is not surprising and such effect is well known. On the other hand, results in Table 2.7 show that when the algorithm stabilizes, the optimality gap does not differ much in scale even when the discount factor is close to one. It can also be seen that the optimal value of the problem is almost proportional to $(1 - \gamma)^{-1}$. This of course is in accordance with the geometric series view of the considered problem (2.57).

Table 2.7: Inventory problem: evolution of bounds of primal and dual periodical programs.

γ	Dual-PSDDP(D.-UB.)	Primal-PSDDP(D.-LB.)	Gap(%)
0.8	43.783186	43.782698	1.115×10^{-3}
0.9906	1173.345945	1173.204425	1.206×10^{-2}
0.9990	11059.03217	11051.86157	6.485×10^{-2}
0.9999	110590.2919	110514.1413	6.886×10^{-2}

When the discount factor γ approaches one (e.g., $\gamma = 0.999$, $\gamma = 0.9999$), our experiments indicate that the convergence is much slower than for smaller discount factors. In order to deal with this we apply the trust-bound strategy to the periodical Dual SDDP algorithm, as it drastically saves CPU time and yields a faster convergence (see Remark 2.6.1 below and 2.5.3).

Remark 2.6.1 (trust-bound strategy). The Periodical Dual SDDP algorithm starts by setting an initial (constant) upper bound for the value functions, and proceeds by adding cuts during the iterations. In order to make sure that this initial upper bound is bigger than the respective optimal values, the corresponding constant is taken to be sufficiently large. After a significant number of cuts are generated, large linear programs should be solved at consecutive iterations, and this slows down the progress of the numerical procedure. An idea is to restart the algorithm after a certain number of iterations by removing all generated cuts and setting the current upper bounds of the value functions at each stage of the optimization

problem. This strategy worked quite well especially when the discount factor was close to one.

Remark 2.6.2. When the discount factor is very close to one, it becomes very challenging to compute the classical statistical upper bound for the optimal value especially of large-scale problems. To illustrate this, consider for instance the inventory model (2.59) with $\gamma = 0.999$ and period $m = 12$, and its statistical upper bound (with 95% confidence level). When the algorithm for solving the primal model stabilizes, we evaluate value of the constructed policy on the discretized model using Monte Carlo simulation with number of simulations equal to 3000.

Note that when $\gamma = 0.999$, the error of a finite horizon approximation is of order $O(\gamma^T/(1-\gamma))$ (cf., [76]), which is small enough (≈ 0.045) only when $T \geq 10000$. The CPU time to compute the statistical upper bound using $T = 10000$ exceeds 24 hours. If we decrease T to 5000, the CPU time to compute the statistical bound is around 18.7 hours. However, the obtained statistical bound turns out to be smaller than the primal deterministic bound, which indicates that such T is too small to provide a valid upper bound. On the other hand, a valid (deterministic) upper bound obtained by solving the dual problem employing the periodical Dual SDDP method with trust bound strategy, only consumes CPU time 3040 seconds, which is less than 1 hour with the corresponding relative gap less than 0.1%.

2.6.2 Hydro-thermal Generation Problem

In this section we consider the Brazilian Inter-connected Power System operation planning problem discussed in [80]. This problem is much larger than the inventory problem considered in the previous part. The original problem has $T = 120$ stages corresponding to 10 years of monthly planning with the discount factor $\gamma = 0.9906$ (this discount factor corresponds to the annual discount rate of 12%), and 4 state variables representing energy equivalent reservoirs of four interconnected main regions. The random data process is represented by the respective 4-dimensional vectors of monthly inflows. We assume

that the monthly inflows are stagewise independent and are sampled from 4-dimensional log-normal distributions calibrated by the historical data.

Similar to the illustration in the inventory case, we present results of applying Dual SDDP and Periodical Dual SDDP on the problem with finite horizon and infinite horizon, respectively.

Primal and Dual Optimization Models

The explicit primal model of the infinite-horizon problem with discount factor $\gamma = 0.9906$ is the following:

$$\begin{aligned}
& \min \sum_{t=1}^T \gamma^{t-1} \left[\sum_{i=1}^4 b_i s_{i,t} + \sum_{i=1}^4 \sum_{j=1}^4 e_j \text{df}_{i,j,t} + \sum_{i=1}^4 u_i \sum_{k \in \Omega_i} g_{i,k,t} + \sum_{i=1}^5 \sum_{j=1}^5 c_{j,i} \text{ex}_{j \rightarrow i,t} \right] \\
& \text{s.t. for } t = 1, 2, \dots, T, \\
& \sum_{k \in \Omega_i} g_{i,k,t} + q_{i,t} + \sum_{j=1}^4 \text{df}_{i,j,t} - \sum_{j=1}^5 \text{ex}_{i \rightarrow j,t} + \sum_{j=1}^5 \text{ex}_{j \rightarrow i,t} = d_{i,t}, \quad i = 1, \dots, 4, \\
& \sum_{j=1}^5 \text{ex}_{j \rightarrow 5,t} - \sum_{j=1}^5 \text{ex}_{5 \rightarrow j,t} = 0, \\
& q_{i,t} + s_{i,t} + v_{i,t} - v_{i,t-1} = a_{i,t} \quad i = 1, \dots, 4, \\
& s_{i,t} \geq 0, \quad i = 1, \dots, 4, \\
& 0 \leq v_{i,t} \leq \bar{v}_i, \quad i = 1, \dots, 4, \\
& 0 \leq q_{i,t} \leq \bar{q}_i, \quad i = 1, \dots, 4, \\
& 0 \leq \text{df}_{i,j,t} \leq \bar{\text{df}}_{i,j}, \quad i, j = 1, \dots, 4, \\
& 0 \leq \text{ex}_{i \rightarrow j,t} \leq \bar{\text{ex}}_{i,j}, \quad i, j = 1, \dots, 4, \\
& \underline{g}_i \leq g_{i,k,t} \leq \bar{g}_i, \quad i = 1, \dots, 4, \quad k \in \Omega_i.
\end{aligned} \tag{2.65}$$

We refer to [80] and [76] for the details of the primal model and variables/parameters notations correspondingly.

By writing the Lagrangian of (2.65) and dualization of the feasibility constraints, the dual model can be written as

$$\begin{aligned} \max \quad & \sum_{t=1}^T \gamma^{t-1} \left[\sum_{i=1}^4 \left(d_{i,t} \lambda_{i,t} + \bar{v}_i x_{i,t} + \bar{q}_i o_{i,t} + \sum_{j=1}^4 \bar{d}f_j h_{i,j,t} + \sum_{k \in \Omega_i} (\bar{g}_i z_{i,k,t} + \underline{g}_i \omega_{i,k,t}) + a_{i,t} \mu_{i,t} \right) \right. \\ & \left. + \sum_{i=1}^5 \sum_{j=1}^5 \bar{e}x_{i,j} f_{i,j,t} \right] + \sum_{i=1}^4 a_{i,1} \mu_{i,1} + \sum_{i=1}^4 v_{i,0} y_{i,1} \end{aligned}$$

s.t. for $t = 1, \dots, T$,

$$\mu_{t,i} \leq b_i, \quad i = 1, \dots, 4,$$

$$\lambda_{i,t} + \mu_{i,t} + o_{i,t} \leq 0, \quad i = 1, \dots, 4,$$

$$\lambda_{i,t} + h_{i,j,t} \leq e_j, \quad i, j = 1, \dots, 4,$$

$$\lambda_{i,t} + z_{i,k,t} + \omega_{i,k,t} = u_i, \quad i = 1, \dots, 4, \quad k \in \Omega_i,$$

$$\text{for } i \in \{1, \dots, 5\}, j \in \{1, \dots, 5\},$$

$$\text{if } i = j : f_{i,j,t} \leq c_{i,j},$$

$$\text{if } i \neq j, (i, j) \leq 4 : -\lambda_{i,t} + \lambda_{j,t} + f_{i,j,t} \leq c_{i,j},$$

$$\text{if } i \neq j, i = 5, j < 5 : \lambda_{j,t} - \eta_t + f_{i,j,t} \leq c_{i,j},$$

$$\text{if } i \neq j, i < 5, j = 5 : \lambda_{i,t} + \eta_t + f_{i,j,t} \leq c_{i,j},$$

$$x_{i,t} \leq 0, \quad i = 1, \dots, 4,$$

$$o_{i,t} \leq 0, \quad i = 1, \dots, 4,$$

$$h_{i,j,t} \leq 0 \quad i, j = 1, \dots, 4,$$

$$z_{i,k,t} \leq 0, \quad i = 1, \dots, 4, \quad k \in \Omega_i,$$

$$\omega_{i,k,t} \geq 0, \quad i = 1, \dots, 4, \quad k \in \Omega_i,$$

$$f_{i,j,t} \leq 0, \quad i, j = 1, \dots, 5,$$

for $t = 2, \dots, T$,

$$\mu_{i,t-1} + x_{i,t-1} - \gamma \mathbb{E} [\mu_{i,t}] \leq 0, \quad i = 1, \dots, 4.$$

(2.66)

Here in (2.66), we denote the states as $\{x_{i,t}, i = 1, \dots, 4\}$ and $\{\mu_{i,t}, i = 1, \dots, 4\}$ for $t = 1, 2, \dots, T$. Control variables are denoted by $\{\lambda_{i,t}\}, \{o_{i,t}\}, \{h_{i,j,t}\}, \{z_{i,k,t}\}, \{\omega_{i,k,t}\}, \{f_{i,j,t}\}$ and $\{y_{i,1}\}$. In both models, initial stored energy $v_{i,0}$ and initial inflow $a_{i,1}, i = 1, \dots, 4$ are given and inflow $\mathbf{a}_t := (a_{1,t}, \dots, a_{4,t}), t = 2, \dots, T$ is periodical and modeled as stagewise independent stochastic process, such that $\mathbf{a}_t \sim \text{lognormal}(\boldsymbol{\mu}_\tau, \boldsymbol{\Sigma}_\tau)$ for each t if $t \bmod m = \tau, \tau = 2, \dots, m + 1$, where $\boldsymbol{\mu}_\tau$ and $\boldsymbol{\Sigma}_\tau$ is the mean and covariance matrix of log of the historical inflow data for each month, respectively.

Dual SDDP under Finite-horizon Setting

We solve this problem using Dual SDDP and Primal SDDP for comparison. For Dual SDDP, a general procedure to define sequences of penalizations $(v_{t,k})$ ensuring convergence of the corresponding Dual SDDP method is to take $v_{t,k} = \gamma_0 \alpha^{k-1} \mathbf{e}, k \geq 1, t = 2, \dots, T$, with $\alpha > 1, \gamma_0 > 0$. For numerical reasons, we also take a large upper bound U for these sequences and use

$$v_{t,k} = \min(U, \gamma_0 \alpha^{k-1}) \mathbf{e}, k \geq 1, t = 2, \dots, T. \quad (2.67)$$

Three variants of Dual SDDP are considered: for the first variant, denoted by Dual SDDP 1, $v_{t,k}$ are as in (2.67) with $\gamma_0 = 10^4, \alpha = 1.3, U = 10^{10}$. To illustrate the fact that for constant sequences $v_{t,k} = \gamma_0$, Dual SDDP converges (resp. does not converge) for sufficiently large constants γ_0 (resp. sufficiently small constants γ_0) we also define two other variants corresponding to $U = +\infty, \gamma_0 = 10^9, \alpha = 1$, and $U = +\infty, \gamma_0 = 10^6, \alpha = 1$, in (2.67), respectively denoted by Dual SDDP 2 and Dual SDDP 3.

We run Dual SDDP for 1000 iterations and Primal SDDP for 3000 iterations. The evolution of the upper and lower bounds computed by the methods for the first 1000 iterations is given in Figure 2.4.⁴

Further, the values of these bounds for different iterations are reported in Table 2.8.

⁴The upper bounds for Primal SDDP are computed as explained in 2.6.1

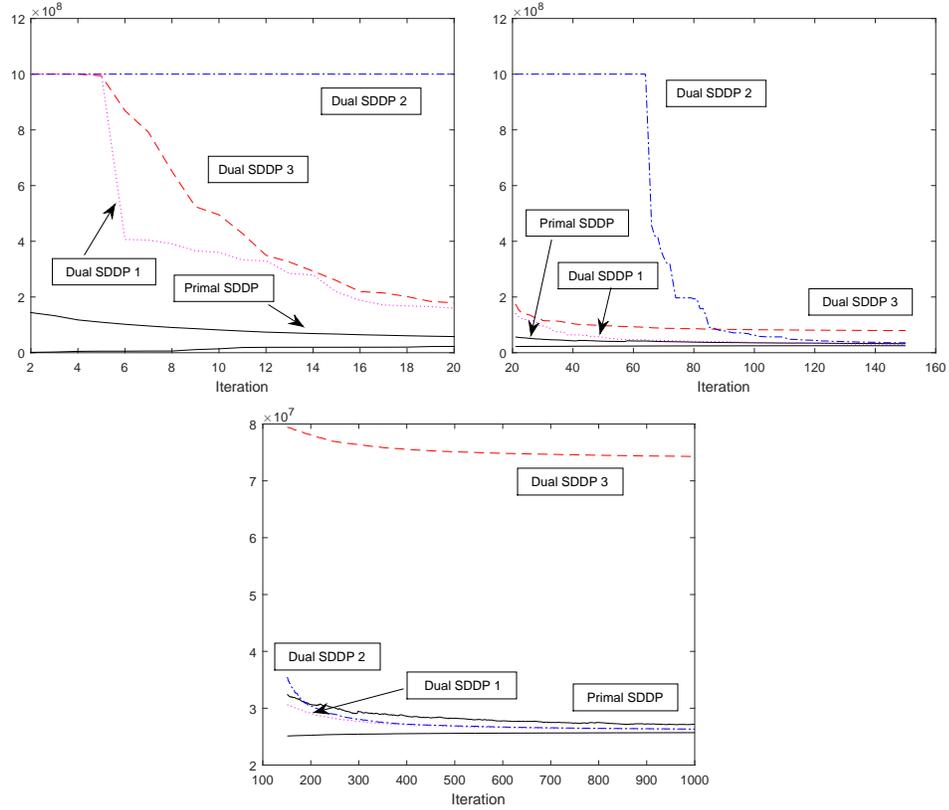


Figure 2.4: Top left: upper and lower bounds computed by Primal SDDP and upper bounds computed by Dual SDDP 1, Dual SDDP 2, and Dual SDDP 3, for the first 20 iterations; Top right: same outputs for iterations 21, . . . , 150; Bottom: same outputs for iterations 151, . . . , 1000.

We observe that parameter γ_0 for Dual SDDP 3 is too small to allow this method to converge to the optimal value of the problem whereas the other two variants Dual SDDP 1 and Dual SDDP 2 of Dual SDDP converge. Naturally, these methods start with large upper bounds but after a few tens of iterations the upper bounds with Dual SDDP 1 and Dual SDDP 2 are better than the upper bound computed by Primal SDDP. In particular, it is interesting to notice that the best (lowest) upper bounds are obtained with the variant of Dual SDDP that uses adaptive penalizations, i.e., penalizations that increase with the number of iterations before reaching value U in (2.67).

We also report in Table 2.9 the relative error $\frac{\text{Upper}_M(i) - \text{Lower}_{\text{SDDP}}(i)}{\text{Upper}_M(i)}$ for iterations $i = 100, 200, 300, 400, 500, 800,$ and 1000 for all methods M where $\text{Upper}_M(i)$ and $\text{Lower}_{\text{SDDP}}(i)$ are respectively the upper bound computed by method M at iteration i and the lower bound

Table 2.8: Lower bound Lb and upper bound Ub computed by Primal SDDP and upper bounds computed by variants of Dual SDDP along iterations. All costs have been divided by 10^6 .

Iteration	Primal SDDP Lb	Primal SDDP Ub	Dual SDDP 1	Dual SDDP 2	Dual SDDP 3
2	1.317	143.98	1000.2	1000.2	1000.2
5	5.5588	109.36	1000.2	1000.2	994.04
10	14.032	81.728	360.40	1000.2	495.08
50	23.670	41.346	54.999	1000.2	96.720
100	24.787	35.502	36.322	64.072	82.494
150	25.111	32.447	30.685	35.595	79.465
200	25.249	30.672	29.076	30.404	78.059
250	25.374	30.079	28.215	28.943	76.917
300	25.436	29.434	27.710	28.030	76.344
350	25.477	29.014	27.309	27.532	75.852
400	25.526	28.626	27.110	27.188	75.526
1000	25.703	27.175	26.304	26.335	74.292
3000	25.798	26.883	-	-	

computed by Primal SDDP at iteration i . For iterations 300 and forwards, the relative error is much smaller with variants of Dual SDDP, meaning that Primal SDDP overestimates the optimality gap.

However, each iteration of Dual SDDP takes more time as can be seen in Figure 2.5 which reports the cumulative CPU time for all methods. To illustrate, running Dual and Primal SDDP in parallel, we can compute the time needed to obtain a solution of relative accuracy ε using the standard stopping criterion for Primal SDDP (see ([81])) or using the

Table 2.9: Relative error as a function of the number of iterations for Primal SDDP, Dual SDDP 1, and Dual SDDP 2.

Iteration	Primal SDDP	Dual SDDP 1	Dual SDDP 2
100	0.30	0.32	0.61
200	0.18	0.13	0.17
300	0.14	0.08	0.09
400	0.11	0.06	0.06
500	0.09	0.05	0.05
800	0.07	0.03	0.03
1000	0.05	0.02	0.02

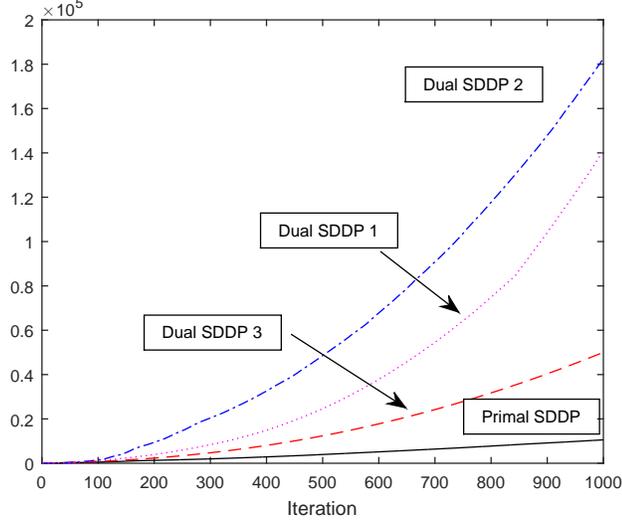


Figure 2.5: Cumulative CPU time for Primal SDDP, Dual SDDP 1, Dual SDDP 2, and Dual SDDP 3.

lower bound from Primal SDDP and the upper bound from Dual SDDP, and computing the relative error obtained with these bounds each time a new bound (either lower bound or upper bound) is computed. The results are reported in Table 2.10. We see that due to the fact that Dual SDDP iterations are more time consuming, for all relative accuracies but one, the use of the stopping criterion based on Dual SDDP upper bounds is more computationally burdened. From this experiment, performed on a larger problem (in terms of size of the state vector and number of control variables for each stage) than the inventory problem, it seems that the use of Dual SDDP for a stopping criterion of Primal SDDP will decrease the overall computational bulk only for small problems (which have a limited to small number of controls, state variables, and scenarios).

Finally, as an evidence of the fact that RCR does not hold for the dual of the inventory and the hydro-thermal problem, we observed that the maximal and mean values of $\|\zeta_t^k\|_1$ along iterations, where ζ_t^k is an optimal value of ζ_t in (2.23) for iteration k , are positive for some stages.

Table 2.10: CPU time (in seconds) needed to obtain a solution of relative accuracy ε with Primal SDDP and variants of Dual SDDP.

ε	Primal SDDP	Dual SDDP 1	Dual SDDP 2
0.3	515	1 042	4 133
0.2	1 167	1 895	7 446
0.15	1 659	2 910	9 882
0.1	3 168	5 114	16 387
0.075	5 359	8 003	22 457
0.05	11 124	15 738	35 113
0.04	45 391	23 449	51 381

Periodical Dual SDDP

Here we follow the periodical variant of this problem discussed in [76] with period $m = 12$ corresponding to the monthly cycle of one year.

We apply the periodical Dual SDDP (1) to solve the SAA of the dual problem, with 50 samples per stage. In order to approximate the infinite horizon setting, we run $T = 120$ stages in the forward pass. The error of that finite horizon approximation is of order $O(\gamma^T/(1-\gamma))$ (cf., [76]). By exploring the periodical behavior, we only need to perform optimization on m stages in the backward pass to approximate the value functions. Objective coefficients of the penalty terms in the algorithm are chosen as $\{r_\tau^k\} = 1 \times 10^9$ throughout all stages and all iterations. The initial upper bounds of value functions approximation is set as 1×10^9 for all stages.

Empirical results are reported for two cases: $\gamma = 0.8$ and $\gamma = 0.9906$. We solve the first model without applying trust-bound strategy. It can be observed that the periodical Dual SDDP method, with the trust-bound strategy, signifies fast convergence in the dual problem, especially when the discount factor is close to one. As it was discussed in the previous section, for $\gamma = 0.9906$ in order to employ the classical statistical upper bound procedure, the corresponding time horizon T should be so large that makes it computationally infeasible.

Table 2.11 reports deterministic bounds and relative gaps of primal and dual problems

Table 2.11: Hydro-thermal problem with $\gamma = 0.8$: deterministic bounds of primal and dual periodical programs.

Iter.	Dual-PSDDP (D.-UB.)($\times 10^6$)	Primal-PSDDP (D.-LB.)($\times 10^6$)	Gap(%)
100	20.454	6.261	69.39
200	11.959	6.589	44.90
300	9.499	6.739	29.06
400	8.602	6.824	20.67
500	8.182	6.851	16.26
800	7.616	6.897	9.43
1000	7.477	6.915	7.51
1500	7.328	6.941	5.28

with $\gamma = 0.8$ for iterations 100, 200, 300, 400, 500, 800, 1000, 1500. We use same notations here as in Table 2.7.

In Figure 2.6 we demonstrate evolution of deterministic primal and dual bounds produced by the algorithm in solving the hydro-thermal problem with discount factor $\gamma = 0.9906$. To solve the dual problem, we utilize the trust-bound strategy by restarting the algorithm every 100 iterations and run the algorithm for 1900 iterations in total when the gap is smaller than given precision 6%.

In view of the evolution of the dual bounds displayed by the figure, we add a few remarks. First, it can be observed that the dual bounds are monotonically decreasing in each epoch (between two consecutive restarts). Such property is not maintained by consecutive restarts, that is, at the beginning of current restart, the dual bound may be larger than the one at the end of the last restart. A reason for this is that the re-initialized bounds of the value functions are still larger than the potential tightest upper bounds of the problem. It should be noticed that the displayed dual bound is the optimal value of the value function at the first stage while at each restart only value functions from stage 2 and onwards are initialized using the information from the last iteration. Therefore, at the beginning of each restart, the multistage problem is re-optimized and a new optimal value of the first stage problem is computed. Secondly, it can be seen from the figure that the algorithm converges

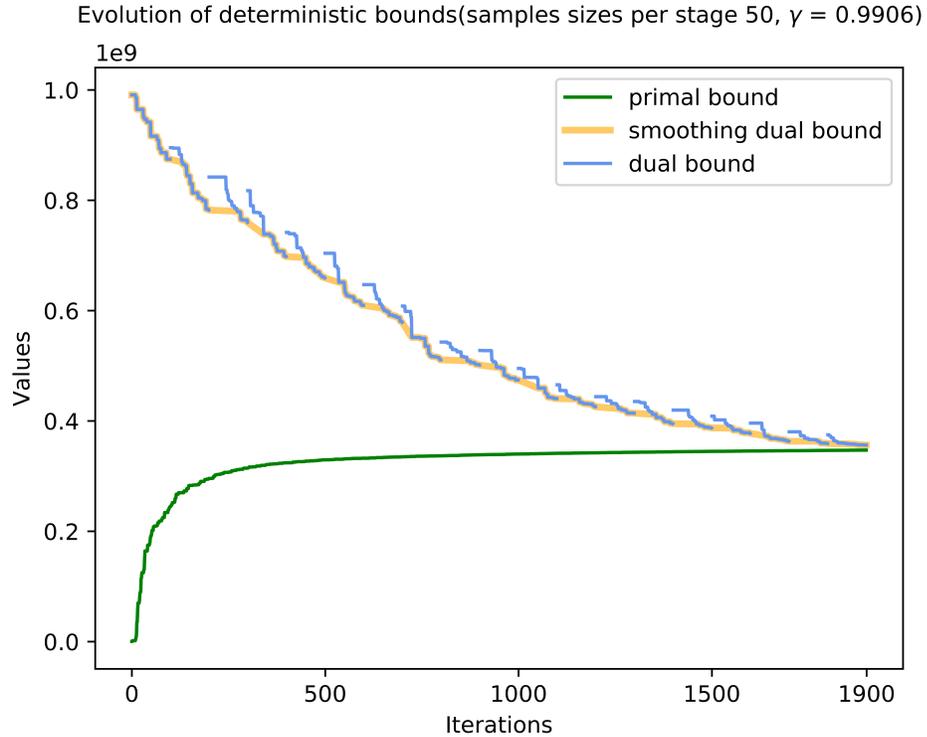


Figure 2.6: Hydro-thermal problem with $\gamma = 0.9906$: evolution of deterministic bounds of primal and dual periodical multistage stochastic programs. The orange line is obtained by smoothing the dual bounds (in blue) to exhibit the descending trend.

faster in the first few restarts and becomes slower afterwards. The algorithm stabilizes and precludes the dual bound from descending below the primal lower bound.

In Table 2.12, we present results of the values of both deterministic bounds and the relative gaps at iterations 100, 500, 1000, 1200, 1500, 1700, 1900 and the algorithm is terminated when the gap is smaller than a given precision 6%. It could be observed that the computed gaps are significantly better than the ones in Table 2.11, even with a larger discount factor. This is due to the employed trust-bound strategy.

Table 2.12: Hydro-thermal problem with $\gamma = 0.9906$: deterministic bounds of primal and dual periodical programs.

Iter.	Dual-PSDDP (D.-UB.)($\times 10^8$)	Primal-PSDDP (D.-LB.)($\times 10^8$)	Gap(%)
100	8.7443	2.4405	72.09
500	6.5912	3.2941	52.61
1000	4.7428	3.3995	28.32
1200	4.2559	3.4213	19.61
1500	3.8719	3.448	10.95
1700	3.6315	3.4601	4.72
1900	3.5621	3.4698	2.59

2.7 Proofs of Auxiliary Results

In this section, we discuss some statistical properties of the multiplicative autoregressive process (2.61), and prove Lemma 2.3.1, Proposition 2.3.1, and 2.3.1.

2.7.1 Properties of the Multiplicative Autoregressive Process.

Consider the multiplicative autoregressive process (2.61). Note that under the specified conditions the demand process is not stationary. Indeed, since the errors ϵ_t are i.i.d. and $\mathbb{E}[\epsilon_t] = 1$ we have that $\mathbb{E}[\mathcal{D}_t] = \phi\mathbb{E}[\mathcal{D}_{t-1}] + \mu$ and

$$\begin{aligned}
 \text{Var}(\mathcal{D}_t) &= \mathbb{E}[\text{Var}(\epsilon_t(\phi\mathcal{D}_{t-1} + \mu)|\mathcal{D}_{t-1})] + \text{Var}[\mathbb{E}(\epsilon_t(\phi\mathcal{D}_{t-1} + \mu)|\mathcal{D}_{t-1})] \\
 &= \mathbb{E}[\sigma^2(\phi\mathcal{D}_{t-1} + \mu)^2] + \text{Var}(\phi\mathcal{D}_{t-1} + \mu) \\
 &= \sigma^2\mathbb{E}[(\phi\mathcal{D}_{t-1} + \mu)^2] + \phi^2\text{Var}(\mathcal{D}_{t-1}).
 \end{aligned} \tag{2.68}$$

It follows that $\mathbb{E}[\mathcal{D}_t]$ converges to $\mu/(1 - \phi)$ as $t \rightarrow \infty$. Suppose, for example, that $\mu = 0$. Then $\mathcal{D}_t = \epsilon_t\phi\mathcal{D}_{t-1} = \mathcal{D}_0\phi^t \prod_{\tau=1}^t \epsilon_\tau$, $t = 1, \dots, T$, $\mathbb{E}[\mathcal{D}_t] = \mathcal{D}_0\phi^t \rightarrow 0$, and

$\text{Var}(\mathcal{D}_t) = \mathcal{D}_0^2 \phi^{2t} [(1 + \sigma^2)^t - 1]$. Therefore if $\phi^2(1 + \sigma^2) < 1$, then $\text{Var}(\mathcal{D}_t) \rightarrow 0$; and if $\phi^2(1 + \sigma^2) > 1$, then $\text{Var}(\mathcal{D}_t) \rightarrow \infty$ provided $\mathcal{D}_0 > 0$.

We need more notations to proceed to the proofs of the mentioned theoretical results. We introduce the sequence of functions for $t = 2, \dots, T$, with the first constraint $A_T^\top \pi_{T,j} \leq c_T$ omitted for $t < T$,

$$\bar{V}_t^k(\pi_{t-1}) := \begin{cases} \max_{\pi_{t,1}, \dots, \pi_{t,N_t}, \zeta_t} & \sum_{j=1}^{N_t} p_{t,j} b_{t,j}^\top \pi_{t,j} - v_{t,k}^\top \zeta_t \\ \text{s.t.} & A_t^\top \pi_{t,j} \leq c_t, \quad j = 1, \dots, N_t, \\ & A_{t-1}^\top \pi_{t-1} + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1} + \zeta_t, \\ & \zeta_t \geq 0, \quad \underline{\pi}_t \leq \pi_{t,j} \leq \bar{\pi}_t, \quad j = 1, \dots, N_t. \end{cases} \quad (2.69)$$

Due to the finite support assumption of the distribution of $\xi_t, t = 2, \dots, T$, we can represent the scenarios for $\xi_1, \xi_2, \dots, \xi_T$, by a scenario tree of depth $T + 1$ where the root node n_0 associated to stage 0 (with decision x_0 taken at that node) has one child node n_1 associated to the first stage. We denote by \mathcal{N} the set of nodes and for a node n of the tree, by $F(n)$ the parent node, by (x_n, π_n) a primal-dual pair at that node and by ξ_n the realization of process (ξ_t) at node n (this realization ξ_n contains in particular the realizations c_n of c_t , b_n of b_t , A_n of A_t , and B_n of B_t).

2.7.2 Proof of Lemma 2.3.1.

If the optimal value of primal problem (1.1) is finite then the optimal value of the corresponding dual problem is finite which implies that there is a bounded dual solution and ensures the existence of $\underline{\pi}_t$ and $\bar{\pi}_t$.

Now assume there is $\hat{x} > 0$ feasible for (1.1). Let $1 \leq t \leq T$ and let us fix a node m of stage t . Let \bar{A}_m such that constraints $A_m x_m + B_m x_{F(m)} = b_m$ are rewritten in the compact form $\bar{A}_m x = b_m$ in terms of vector $x = (x_n)_{n \in \mathcal{N}}$ of decision variables in the scenario tree.

The dual function obtained dualizing the coupling constraints of node m is given by

$$\theta(\pi_m) = \min \mathbb{E}[c^\top x] + \pi_m^\top (A_m x_m + B_m x_{F(m)} - b_m), \quad x \in \mathcal{S}_m,$$

for $\mathcal{S}_m = \{x = (x_n)_{n \in \mathcal{N}} : x \geq 0\} \cap \mathcal{A}_m$ where

$$\mathcal{A}_m = \{x = (x_n)_{n \in \mathcal{N}} : A_n x_n + B_n x_{F(n)} = b_n, \forall n \neq m, n \in \mathcal{N}\}.$$

By Linear Programming Duality, the optimal value $\mathcal{Q}_1(x_0)$ of primal problem (1.1) is the optimal value of the dual problem

$$\max_{\pi_m \in \mathbb{R}^{m_t}} \theta(\pi_m), \quad (2.70)$$

which can clearly be written as

$$\mathcal{Q}_1(x_0) = \max_{\pi_m} \{\theta(\pi_m) : \pi_m = \bar{A}_m x - b_m, x \in \text{Aff}(\mathcal{S}_m)\}, \quad (2.71)$$

where $\text{Aff}(\mathcal{S}_m)$ is the affine hull of \mathcal{S}_m . We now bound the optimal solutions of dual problem (2.71). Since (2.70) and (2.71) have the same optimal values, adding these bounds as constraints on π_m in (2.70) does not change its optimal value. Since $\hat{x} > 0$ there is $r > 0$ such that

$$\mathbb{B}(\hat{x}, r) \subseteq \{x \geq 0\}. \quad (2.72)$$

We argue that $\text{Aff}(\mathcal{S}_m) = \mathcal{A}_m$. Indeed, the inclusion $\text{Aff}(\mathcal{S}_m) \subseteq \mathcal{A}_m$ is clear. Now if $x \in \mathcal{A}_m$ then if $x = \hat{x}$ we have that $x \in \mathcal{S}_m \subseteq \text{Aff}(\mathcal{S}_m)$ and if $x \neq \hat{x}$, recalling that $\hat{x} \in \mathcal{A}_m$ satisfies (2.72) we have that

$$y := \hat{x} + \frac{r}{2} \frac{x - \hat{x}}{\|x - \hat{x}\|} \in \mathcal{A}_m \cap \mathbb{B}(\hat{x}, r) \subseteq \mathcal{S}_m.$$

Therefore x belongs to the line that contains y and \hat{x} with y, \hat{x} belonging to \mathcal{S}_m which implies $x \in \text{Aff}(\mathcal{S}_m)$ and $\text{Aff}(\mathcal{S}_m) = \mathcal{A}_m$.

It follows that

$$\mathbb{B}(\hat{x}, r) \cap \text{Aff}(\mathcal{S}_m) = \mathbb{B}(\hat{x}, r) \cap \mathcal{A}_m \subseteq \mathcal{S}_m$$

and that there is $\rho_*(m) > 0$ such that

$$\mathbb{B}(0, \rho_*) \cap (\bar{A}_m \mathcal{A}_m - b_m) \subseteq \bar{A}_m (\mathbb{B}(\hat{x}, r) \cap \mathcal{A}_m) - b_m.$$

Let $\bar{\pi}_m$ be an optimal solution of problem (2.71) and let $z = 0$ if $\bar{\pi}_m = 0$ and $z = -\frac{\bar{\pi}_m}{\|\bar{\pi}_m\|_2} \rho_*$ otherwise. Observe that $z \in \mathbb{B}(0, \rho_*) \cap (\bar{A}_m \mathcal{A}_m - b_m)$ and therefore $z \in \bar{A}_m (\mathbb{B}(\hat{x}, r) \cap \mathcal{A}_m) - b_m \subseteq \bar{A}_m \mathcal{S}_m - b_m$ and z can be written $z = \bar{A}_m \tilde{x} - b_m$ for $\tilde{x} \in \mathbb{B}(\hat{x}, r) \cap \mathcal{S}_m$. Denoting by \underline{V} any finite lower bound on the optimal value $\mathcal{Q}_1(x_0)$, it follows that

$$\begin{aligned} \underline{V} \leq \mathcal{Q}_1(x_0) = \theta(\bar{\pi}_m) &\leq \mathbb{E}[c^\top \hat{x}] + \bar{\pi}_m^\top (\bar{A}_m \tilde{x} - b_m) \\ &\leq \mathbb{E}[c^\top \hat{x}] + r \sum_{t=1}^T \mathbb{E}[\|c_t\|_2] + \bar{\pi}_m^\top z \\ &= \mathbb{E}[c^\top \hat{x}] + r \sum_{t=1}^T \mathbb{E}[\|c_t\|_2] - \rho_*(m) \|\bar{\pi}_m\|_2 \end{aligned}$$

which gives for every node n of stage t that

$$\|\bar{\pi}_n\|_2 \leq \max_{m \in \text{Nodes}(t)} \frac{\mathbb{E}[c^\top \hat{x}] - \underline{V} + r \sum_{t=1}^T \mathbb{E}[\|c_t\|_2]}{\rho_*(m)}$$

with corresponding box constraints $\underline{\pi}_t, \bar{\pi}_t$ where $\text{Nodes}(t)$ are the nodes of stage t . \square

2.7.3 Proof of Proposition 2.3.1.

We show by induction on k that $V_t \leq V_t^k$ for $t = 2, \dots, T$. For $k = 0$ these relations hold by definition. Assume that for some $k \geq 1$ we have $V_t \leq V_t^{k-1}$ for $t = 2, \dots, T$. We show by backward induction on t that $V_t \leq V_t^k$ for $t = 2, \dots, T$. Observe that for any π_{T-1} , optimization problem (2.69) with optimal value $\bar{V}_T^k(\pi_{T-1})$ is feasible. Indeed, since

primal problem (1.1) is feasible and has a finite optimal value, the corresponding dual problem is feasible which implies that there is $\pi_{T,1}, \dots, \pi_{T,N_T}$ satisfying $A_T^\top \pi_{T,j} \leq c_T$, $\underline{\pi}_T \leq \pi_{T,j} \leq \bar{\pi}_T$, $j = 1, \dots, N_T$, and for every such points we can find $\zeta_T \geq 0$ satisfying the remaining constraints in (2.69). Therefore $\bar{V}_T^k(\pi_{T-1})$ is finite for every π_{T-1} and is the optimal value of the corresponding dual optimization problem, i.e., for any π_{T-1} we get

$$\bar{V}_T^k(\pi_{T-1}) = \begin{cases} \min_{\alpha, \delta, \underline{\Psi}, \bar{\Psi}} & \delta^\top (c_{T-1} - A_{T-1}^\top \pi_{T-1}) + c_T^\top \sum_{j=1}^{N_T} \alpha_j + \sum_{j=1}^{N_T} \bar{\Psi}_j^\top \bar{\pi}_T - \sum_{j=1}^{N_T} \underline{\Psi}_j^\top \underline{\pi}_T \\ \text{s.t.} & A_T \alpha_j + p_{T,j} B_{T,j} \delta - \underline{\Psi}_j + \bar{\Psi}_j = p_{T,j} b_{T,j}, \quad j = 1, \dots, N_T, \\ & 0 \leq \delta \leq v_{T,k}, \alpha_j, \underline{\Psi}_j, \bar{\Psi}_j \geq 0, \quad j = 1, \dots, N_T. \end{cases}$$

Using this dual representation and the definition of $\bar{\theta}_T^k, \bar{\beta}_T^k$, we get for every π_{T-1} :

$$\bar{\theta}_T^k + \langle \bar{\beta}_T^k, \pi_{T-1} \rangle \geq \bar{V}_T^k(\pi_{T-1}). \quad (2.73)$$

Recalling representation (2.69) for $\bar{V}_T^k(\pi_{T-1})$, observe that for every $\pi_{T-1} \in \text{dom}(V_T)$ we have $\bar{V}_T^k(\pi_{T-1}) \geq V_T(\pi_{T-1})$ whereas for $\pi_{T-1} \notin \text{dom}(V_T)$ we have $V_T(\pi_{T-1}) = -\infty$ while $\bar{V}_T^k(\pi_{T-1})$ is finite, which shows that for every π_{T-1} we have $\bar{V}_T^k(\pi_{T-1}) \geq V_T(\pi_{T-1})$, which, combined with (2.73) and the induction hypothesis, gives

$$V_T^k(\pi_{T-1}) \geq V_T(\pi_{T-1})$$

for every π_{T-1} .

Now assume that $V_{t+1}^k(\pi_t) \geq V_{t+1}(\pi_t)$ for all π_t for some $t \in \{2, \dots, T-1\}$. We want to show that $V_t^k(\pi_{t-1}) \geq V_t(\pi_{t-1})$ for all π_{t-1} . First observe that for every π_{t-1} , linear program (2.69) with optimal value $\bar{V}_t^k(\pi_{t-1})$ is feasible and has a finite optimal value. Therefore we can express $\bar{V}_t^k(\pi_{t-1})$ as the optimal value of the corresponding dual problem

given by

$$\begin{aligned}
& \min_{\delta, \nu, \underline{\Psi}, \bar{\Psi}} \delta^\top \left[c_{t-1} - A_{t-1}^\top \pi_{t-1} \right] + \sum_{i=0}^k \bar{\theta}_{t+1}^i \sum_{j=1}^{N_t} \nu_i(j) + \sum_{j=1}^{N_t} \bar{\Psi}_j^\top \bar{\pi}_t - \sum_{j=1}^{N_t} \underline{\Psi}_j^\top \pi_t \\
& \text{s.t.} \quad p_{t,j} B_{t,j} \delta - \sum_{i=0}^k \nu_i(j) \bar{\beta}_{t+1}^i - \underline{\Psi}_j + \bar{\Psi}_j = p_{t,j} b_{t,j}, \quad j = 1, \dots, N_t, \\
& \quad \sum_{i=0}^k \nu_i(j) = p_{t,j}, \quad \underline{\Psi}_j, \bar{\Psi}_j \geq 0, \quad j = 1, \dots, N_t, \\
& \quad \nu_0, \dots, \nu_k \geq 0, \quad 0 \leq \delta \leq v_{t,k}.
\end{aligned} \tag{2.74}$$

Using this representation of \bar{V}_t^k and the definition of $\bar{\theta}_t^k, \bar{\beta}_t^k$, we obtain for every π_{t-1} :

$$\bar{\theta}_t^k + \langle \bar{\beta}_t^k, \pi_{t-1} \rangle \geq \bar{V}_t^k(\pi_{t-1}). \tag{2.75}$$

Next, recalling representation (2.69) for $\bar{V}_t^k(\pi_{t-1})$ and the induction hypothesis, we get

$$\bar{V}_t^k(\pi_{t-1}) \geq \widehat{V}_t^k(\pi_{t-1}) \tag{2.76}$$

where

$$\widehat{V}_t^k(\pi_{t-1}) := \begin{cases} \max_{\pi_{t,1}, \dots, \pi_{t,N_t}, \zeta_t} & \sum_{j=1}^{N_t} p_{t,j} (b_{t,j}^\top \pi_{t,j} + V_{t+1}(\pi_{t,j})) - v_{t,k}^\top \zeta_t \\ \text{s.t.} & A_{t-1}^\top \pi_{t-1} + \sum_{j=1}^{N_t} p_{t,j} B_{t,j}^\top \pi_{t,j} \leq c_{t-1} + \zeta_t, \\ & \zeta_t \geq 0, \underline{\pi}_t \leq \pi_{t,j} \leq \bar{\pi}_t, \quad j = 1, \dots, N_t. \end{cases}$$

Similarly to the induction step $t = T$, for every π_{t-1} , we have

$$\widehat{V}_t^k(\pi_{t-1}) \geq V_t(\pi_{t-1}). \tag{2.77}$$

Combining (2.75), (2.76), and (2.77) with the induction hypothesis, we obtain $V_t^k(\pi_{t-1}) \geq V_t(\pi_{t-1})$ for all π_{t-1} which achieves the proof of the induction step t .

In particular $V_2^{k-1} \geq V_2$ which implies that V^{k-1} is greater than or equal to the optimal value of dual problem (2.2) which is also, by linear programming duality, the optimal value of primal problem (1.1). \square

The proof of 2.3.1 is based on the following lemma:

Lemma 2.7.1. *Suppose that the multistage problem (1.1) has a finite optimal value. Then for sufficiently large values of the components of vectors v_t , in the dynamic equations (2.12), the optimal value of the multistage problem defined by these dynamic equations coincides with the optimal value of the original problem (1.1).*

Proof. As it was already mentioned, since it is assumed that the number of scenarios is finite, we can view problem (1.1) as a large linear program (deterministic equivalent) written under the form

$$\min_x c^\top x \quad \text{s.t. } \mathcal{A}x = b, \quad x \geq 0. \quad (2.78)$$

Also since (1.1) has a finite optimal value, it has a nonempty set of optimal solutions and there is a bounded optimal solution of (2.78). Let us fix such an optimal solution \bar{x} . We have that problem (2.78) can be written

$$\min_x c^\top x \quad \text{s.t. } \mathcal{A}x = b, \quad 0 \leq x \leq \bar{x}. \quad (2.79)$$

The dynamic programming equations (2.4) - (2.6) represent the standard dual of (1.1). We can also think about that dual as a large linear programming problem of the form (this is the dual of (2.78)):

$$\max_\pi b^\top \pi \quad \text{s.t. } \mathcal{A}^\top \pi \leq c. \quad (2.80)$$

Similarly the deterministic equivalent of penalized dynamic equations (2.12) can be written as:

$$\max_{\pi, \zeta} b^\top \pi - v^\top \zeta \quad \text{s.t. } \mathcal{A}^\top \pi \leq c + \zeta, \quad \zeta \geq 0. \quad (2.81)$$

Next, from optimality conditions of linear programs, (x, π) is an optimal primal-dual pair

for (2.78)-(2.80) if and only if

$$x^\top(\mathcal{A}^\top\pi - c) = 0, \mathcal{A}x = b, x \geq 0, \mathcal{A}^\top\pi \leq c. \quad (2.82)$$

The corresponding optimality conditions for (2.81) are

$$x^\top(\mathcal{A}^\top\pi - c - \zeta) - \zeta^\top\gamma = 0, \mathcal{A}^\top\pi \leq c + \zeta, \zeta \geq 0, \mathcal{A}x = b, x \geq 0, \gamma \geq 0, x = v - \gamma. \quad (2.83)$$

Now let $\bar{\pi}$ be an optimal dual solution, i.e., an optimal solution of (2.80). Then (2.82) is satisfied with $(x, \pi) = (\bar{x}, \bar{\pi})$. It follows that if $v \geq \bar{x}$, then $(x, \pi, \zeta, \gamma) = (\bar{x}, \bar{\pi}, 0, v - \bar{x})$ with $\zeta = 0$ satisfies (2.83), and hence $(\bar{\pi}, \bar{\zeta}) = (\bar{\pi}, 0)$ is an optimal solution of (2.81) showing that the optimal value of (2.81) is $b^\top\bar{\pi} = c^\top\bar{x}$, i.e., the optimal value of (2.78). We obtain that for $v \geq \bar{x}$, the optimal values of problems (2.80) and (2.81) do coincide. Observe that the dual of (2.81) is given by

$$\min_x c^\top x \text{ s.t. } \mathcal{A}x = b, 0 \leq x \leq v,$$

and for $v \geq \bar{x}$, this linear program has the same optimal value as (2.79), which, as we have seen, is equivalent to primal problem (1.1). \square

2.7.4 Proof of Theorem 2.3.1

Dual SDDP with penalizations is SDDP applied to Dynamic Programming equations corresponding to a linear program with finite optimal value, satisfying relatively complete recourse with discrete uncertainties of finite support. Since samples $\tilde{\xi}_t^k$ in Dual SDDP with penalizations are independent, we can follow the convergence proof of SDDP for linear programs from [82] to obtain that V^k converges to the optimal value of the penalized linear programs, which, by Lemma 2.7.1 (observe that the Lemma can be applied since $\lim_{k \rightarrow +\infty} v_{tk} = +\infty$), is the optimal value of (1.1). \square

CHAPTER 3

UPPER BOUNDS FOR RISK AVERSE MULTISTAGE STOCHASTIC PROGRAM

3.1 Overview

For risk-neutral problems and a discrete finite sample space, a standard stopping criterion for SDDP is based on a deterministic lower bound and a statistical upper bound on the optimal value of the problem, computed at each iteration of the method. For nested risk-averse problems, a deterministic lower bound can be computed as in the risk-neutral case, but to the best of our knowledge, no computationally feasible *statistical* upper bound has been proposed so far for SDDP.

Of course, in theory the value of the constructed approximate policy can be computed by evaluating the risk at each node of the scenario tree. However, this computation rapidly becomes prohibitive with increase of the number of stages and the resulting in exponential growth of the number of possible realizations of the stochastic data process. A deterministic upper bound on the value of the approximate risk-averse policy was proposed in [83] on the basis of inner approximations of the value functions. which is a natural extension of similar constructions for two stage programs (e.g., [84, section 9.5]). The bounds in [85, 86] were developed for risk-neutral problems, and recently extended to risk-averse problems in [87]. However, the computational bulk required to compute the deterministic bounds from [83] and [87] for risk-averse problems increases rapidly with increase of the number of stages, the number of realizations of the stochastic data per stage, and the dimension of the state vectors. The goal of this chapter is to fill this gap proposing an efficiently computable statistical upper bound for SDDP applied to nested-risk averse stochastic problems. This will be possible for a large class of monotone convex risk measures that will be studied.

Our developments will be derived for Stochastic Optimal Control (SOC) modeling,

instead of the Multistage Stochastic Programming approach often used in the SDDP and related methods. The SOC is classical with applications documented in a large number of publications (e.g., [88]). We would like to emphasize that many problems discussed in the Stochastic Programming (SP) literature, can be formulated in the SOC framework. One such example is the classical inventory model (it is presented from both points of view in Sections 1.2.3 and 7.6.3 in [9]). Another such example is the hydro-thermal planning problem. One modification in applying an SDDP type algorithm to SOC problems is the fact that it is not necessary anymore to solve the dual problems to compute the required subgradients of the cost-to-go functions. More importantly, from the point of view of the SDDP type algorithms, applied to risk-averse problems, there is an important difference between the SOC modeling, as compared with the SP approach. A straightforward attempt for computation of statistical upper bounds in the SP framework resulted in an exponential growth of the involved bias with increase of the number of stages, which made it practically useless (cf., [89]). On the other hand, we are going to demonstrate that in the SOC framework it is possible to construct such statistical upper bound in a computationally feasible way.

The outline of the chapter is the following. In Section 3.2, we present the class of risk-neutral SOC problems and describe the SDDP type approach for solving this class of problems. In Section 3.3, we present the risk-averse SOC problem and describe the SDDP algorithm for this problem. In Section 3.3.2, we derive our statistical upper bound. Finally, in Section 3.4 we present numerical results where our upper bound is computed along iterations of SDDP type algorithm to solve a risk-averse real-life hydro-thermal planning problem.

We use the following notation. By $\xi_{[t]} := (\xi_1, \dots, \xi_t)$ we denote the history of a process (ξ_t) up to time t . By $\mathbb{I}_A(x)$ we denote the indicator function of a set A , i.e., $\mathbb{I}_A(x) = 0$ if $x \in A$, and $\mathbb{I}_A(x) = +\infty$ otherwise. For $a \in \mathbb{R}$, $[a]_+ := \max\{a, 0\}$.

3.2 Risk-neutral Stochastic Optimal Control

Consider the classical Stochastic Optimal Control (SOC) (discrete time, finite horizon) model (e.g., [88]):

$$\min_{\pi \in \Pi} \mathbb{E}^\pi \left[\sum_{t=1}^T c_t(x_t, u_t, \xi_t) + c_{T+1}(x_{T+1}) \right], \quad (3.1)$$

where Π is the set of policies satisfying the constraints

$$\Pi = \left\{ \pi_t = \pi_t(\xi_{[t-1]}) : \pi_t = (x_t, u_t), u_t \in \mathcal{U}_t, x_{t+1} = F_t(x_t, u_t, \xi_t), t = 1, \dots, T \right\}. \quad (3.2)$$

Here variables $x_t \in \mathbb{R}^{n_t}$, $t = 1, \dots, T + 1$, represent the state of the system, $u_t \in \mathbb{R}^{m_t}$, $t = 1, \dots, T$, are controls, $\xi_t \in \mathbb{R}^{d_t}$, $t = 1, \dots, T$, are random vectors, $c_t : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \times \mathbb{R}^{d_t} \rightarrow \mathbb{R}$, $t = 1, \dots, T$, are cost functions, $c_{T+1}(x_{T+1})$ is a final cost function, $F_t : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \times \mathbb{R}^{d_t} \rightarrow \mathbb{R}^{n_{t+1}}$ are (measurable) mappings and \mathcal{U}_t is a (nonempty) subset of \mathbb{R}^{m_t} . Values x_1 and ξ_0 are deterministic (initial conditions); it is also possible to view x_1 as random with a given distribution, this is not essential for the following discussion. The optimization in (3.1) is performed over policies $\pi \in \Pi$ determined by decisions u_t and state variables x_t considered as functions of $\xi_{[t-1]} = (\xi_1, \dots, \xi_{t-1})$, $t = 1, \dots, T$, and satisfying the feasibility constraints (3.2). For the sake of simplicity, in order not to distract from the main message of the paper, we assume that the control sets \mathcal{U}_t do not depend on x_t .

It is possible to extend the analysis to the general case, where the control sets are functions of the state variables. Consider the setting where the control set depends on the state variables. That is, consider the extension of problem (3.1) - (3.2), where the feasibility constraints $u_t \in \mathcal{U}_t$ are replaced by $u_t \in \mathcal{U}_t(x_t)$ with $\mathcal{U}_t : \mathbb{R}^{n_t} \rightrightarrows \mathbb{R}^{m_t}$ being a (measurable) point to set mapping, $t = 1, \dots, T$. By changing the cost functions to $\bar{c}_t(x_t, u_t, \xi_t) := c_t(x_t, u_t, \xi_t) + \mathbb{I}_{\mathcal{U}_t(x_t)}(u_t)$, we can write the corresponding problem in the

following form

$$\min_{\pi} \mathbb{E}^{\pi} \left[\sum_{t=1}^T \bar{c}_t(x_t, u_t, \xi_t) + c_{T+1}(x_{T+1}) \right], \quad (3.3)$$

$$\text{s.t. } u_t = \pi_t(\xi_{[t-1]}), u_t \in \mathbb{R}^{m_t} \text{ and } x_{t+1} = F_t(x_t, u_t, \xi_t), t = 1, \dots, T. \quad (3.4)$$

In order to maintain convexity of the value functions, we need to verify convexity in (x_t, u_t) of the cost functions $\bar{c}_t(x_t, u_t, \xi_t)$, i.e., to verify convexity of the indicator functions $\psi_t(x_t, u_t) := \mathbb{I}_{\mathcal{U}_t(x_t)}(u_t)$. Note that $\psi_t(x_t, u_t) = 0$ if $u_t \in \mathcal{U}_t(x_t)$, and $\psi_t(x_t, u_t) = +\infty$ otherwise, i.e., $\psi_t(\cdot, \cdot)$ is the indicator function of the set $A_t := \{(x_t, u_t) : u_t \in \mathcal{U}_t(x_t)\}$. Therefore $\psi_t(x_t, u_t)$ is convex iff the set A_t is a convex subset of $\mathbb{R}^{n_t} \times \mathbb{R}^{m_t}$. In particular, suppose that $\mathcal{U}_t(x_t) := \{u_t : g_{ti}(x_t, u_t) \leq 0, i = 1, \dots, k\}$ for given functions $g_{ti} : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \rightarrow \mathbb{R}$. Then the set A_t is convex if the functions $g_{ti}(\cdot, \cdot)$ are convex. We also should be able to compute a subgradient of $\psi_t(x_t, u_t)$ with respect to x_t . Assuming that the set A_t is convex and closed, the subdifferential of $\psi_t(x_t, u_t)$ at a point $(\bar{x}_t, \bar{u}_t) \in A_t$ is given by the normal cone to A_t at the point (\bar{x}_t, \bar{u}_t) .

With some abuse of the notation we use the same notation for x_t and u_t , and later for θ_t , when considered as functions of the random process ξ_t , and when considered as vector variables, e.g., when writing the respective dynamic programming equations. The particular meaning will be clear from the context.

It is said that the process ξ_t is *stagewise independent* if the probability distribution of ξ_t does not depend on $\xi_{[t-1]}$ for $t = 1, \dots, T$. We make the following basic assumption.

- (A) The random data process ξ_1, \dots, ξ_T is stagewise independent, and its probability distribution does not depend on our decisions.

Since it is assumed that the data process is stagewise independent, it suffices to consider policies of the form $u_t = u_t(x_t), t = 1, \dots, T$.

We can consider problem (3.1)-(3.2) in the framework of Stochastic Programming (SP) if we view $y_t = (x_t, u_t)$ as decision variables. In various applications it is possible to

approach the *same* problem using either the SOC or SP formulations. For example, the classical inventory model can be treated in both frameworks (e.g., [9, sections 1.2.3 and 7.6.3]). Another such example is discussed in Section 3.4 below.

However, there are essential differences between the SOC and SP modeling approaches. In the SOC there is a clear separation between the state and control variables. At every stage (time period) t the optimization is performed over feasible controls (also called actions) u_t and consequently the state at the next stage is determined by the state equation $x_{t+1} = F_t(x_t, u_t, \xi_t)$. This has important implications for the SDDP algorithm, especially in the risk averse setting.

Quite often the same optimization problem can be alternatively formulated either in the SOC or SP framework. In both cases the decision should be based on information available at time of the decision, this is the so-called nonanticipativity principle. There are various ways how the information available at time t can be represented. Here we assume that it is defined by history of the random (data) process ξ_t . We label the available history at time t as $\xi_{[t-1]} = (\xi_0, \xi_1, \dots, \xi_{t-1})$, with ξ_0 being given (deterministic). Of course, shifting the time label we can write this as $\xi_{[t]} = (\xi_1, \dots, \xi_t)$ with now ξ_1 being deterministic representing the initial conditions, which is more common in the SP formulations. What is important that in both cases our decisions are functions of the observed realizations of the data process at time of the decision. It also could be noted that we need to consider only policies which are functions of the data process alone because of the basic assumption that the distribution of the random process ξ_t does not depend on our decisions.

The dynamic programming equations can be written as follows. At the last stage, the value function $V_{T+1}(x_{T+1}) = c_{T+1}(x_{T+1})$ and, going backward in time for $t = T, \dots, 1$, the value functions

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \mathbb{E} [c_t(x_t, u_t, \xi_t) + V_{t+1}(F_t(x_t, u_t, \xi_t))] , \quad (3.5)$$

where the expectation is taken with respect to the (marginal) distribution of ξ_t , The optimal policy is defined by the optimal controls $\bar{u}_t(x_t) \in \mathcal{U}_t^*(x_t)$, where

$$\mathcal{U}_t^*(x_t) := \arg \min_{u_t \in \mathcal{U}_t} \mathbb{E} [c_t(x_t, u_t, \xi_t) + V_{t+1}(F_t(x_t, u_t, \xi_t))] . \quad (3.6)$$

The optimal value of the SOC problem (3.1)-(3.2) is given by the first stage value function $V_1(x_1)$, and can be viewed as a function of the initial conditions x_1 .

We assume that the sets $\mathcal{U}_t^*(x_t)$, $t = 1, \dots, T$, are *nonempty* for every possible realization of state variables. This holds under standard regularity conditions, e.g., if the sets \mathcal{U}_t are compact and the objective function in the right hand side of (3.6) is continuous in $u_t \in \mathcal{U}_t$.

We consider the convex case, by making the following assumption.

- (B) For $t = 1, \dots, T$: (i) the sets \mathcal{U}_t are closed, convex, (ii) the cost functions $c_t(x_t, u_t, \xi_t)$ are convex in (x_t, u_t) , and

$$F_t(x_t, u_t, \xi_t) := A_t x_t + B_t u_t + b_t, \quad (3.7)$$

with matrices $A_t = A_t(\xi_t)$, $B_t = B_t(\xi_t)$ and vectors $b_t = b_t(\xi_t)$ being functions of ξ_t .

It follows that the value functions $V_t(\cdot)$ are convex.

Suppose further that random vector ξ_t has a finite number of realizations ξ_{ti} with respective probabilities p_{ti} , $i = 1, \dots, N$, $t = 1, \dots, T$ (for the sake of simplicity assume that the cardinality N is the same for every time t). Denote $c_{ti}(x_t, u_t) := c_t(x_t, u_t, \xi_{ti})$ and $A_{ti} = A_t(\xi_{ti})$, $B_{ti} = B_t(\xi_{ti})$, $b_{ti} = b_t(\xi_{ti})$, $i = 1, \dots, N$, the respective values of the parameters. In that case, the dynamic programming equations (3.5) can be written as

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \underbrace{\sum_{i=1}^N p_{ti} [c_{ti}(x_t, u_t) + V_{t+1}(A_{ti}x_t + B_{ti}u_t + b_{ti})]}_{\mathbb{E}[c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)]} . \quad (3.8)$$

The subdifferentials of the value functions are obtained from the dynamic programming equations (3.8). That is, consider function

$$Q_t(x_t, u_t) := \mathbb{E} [c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)].$$

Since $c_t(x_t, u_t, \xi_t)$ is convex in (x_t, u_t) and V_{t+1} is convex, $Q_t(x_t, u_t)$ is convex. By (3.8) we have that

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} Q_t(x_t, u_t) = \inf_{u_t \in \mathbb{R}^{m_t}} \{Q_t(x_t, u_t) + \mathbb{I}_{\mathcal{U}_t}(u_t)\}. \quad (3.9)$$

Consequently we have the following formula for the subdifferential of $V_t(\cdot)$ (cf., [90, Theorem 24(a)]):

$$\partial V_t(x_t) = \{\gamma_t : (\gamma_t, 0) \in \partial[Q_t(x_t, u_t) + \mathbb{I}_{\mathcal{U}_t}(u_t)]\} = \{\gamma_t : (\gamma_t, 0) \in \partial Q_t(x_t, \bar{u}_t)\}, \quad (3.10)$$

where¹ \bar{u}_t is any point of $\mathcal{U}_t^*(x_t)$. Since the expectation here is a finite sum, we have that

$$\underbrace{\sum_{i=1}^N p_{ti} [\partial(c_{ti}(x_t, u_t) + V_{t+1}(A_{ti}x_t + B_{ti}u_t + b_{ti}))]}_{\mathbb{E}[\partial(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t))]} \subset \partial Q_t(x_t, \bar{u}_t). \quad (3.11)$$

Actually under mild regularity conditions the equality in (3.11) holds, in particular the equality holds if the value functions are real valued. Note that the subdifferentials in (3.11) are taken jointly in x_t and u_t . It follows that any subgradient in the left side of (3.11) is also a subgradient in the right side of (3.11). Consequently a subgradient $\nabla V_t(\cdot)$ is given by

$$\nabla V_t(x_t) = \sum_{i=1}^N p_{ti} [\nabla c_{ti}(x_t, \bar{u}_t) + A_{ti}^\top \nabla V_{t+1}(A_{ti}x_t + B_{ti}\bar{u}_t + b_{ti})], \quad (3.12)$$

for any $\bar{u}_t \in \mathcal{U}_t^*(x_t)$, where $\nabla c_t(x_t, \bar{u}_t, \xi_t)$ is a subgradient of $c_t(\cdot, \bar{u}_t, \xi_t)$ at x_t .

¹The indicator function can be removed in the last term of (3.10) since the second component of $(\gamma_t, 0)$ is 0.

Now suppose that value functions $V_t(\cdot)$ are approximated by (lower bounding) piecewise affine functions

$$\underline{V}_t(x_t) = \max_{j=1, \dots, M} \ell_{tj}(x_t), \quad (3.13)$$

where $\ell_{tj}(x_t) = a_{tj}^\top x_t + h_{tj}$, $j = 1, \dots, M$. Then $\nabla \underline{V}_t(x_t) = a_{t\nu}$, where $\nu \in \{1, \dots, M\}$ is such that $\nu \in \arg \max_{j=1, \dots, M} \ell_{tj}(x_t)$. This suggests a way for computing a subgradient of a current approximation of the value functions in a cutting planes type algorithm discussed below. There is no need to solve dual problems as in the classical SDDP method.

A cutting planes (SDDP type) algorithm for the SOC problem can be described as follows. In the forward step at iteration k of the algorithm, for given convex piecewise affine lower bounding approximations \underline{V}_t^k of the value functions and for a generated sample path (scenario) $\hat{\xi}_1, \dots, \hat{\xi}_T$ of realizations of the random data process, starting with the initial value $\hat{x}_1 = x_1$, compute a minimizer in the right hand side of (3.8) for the current approximation of the value function, that is

$$\hat{u}_t \in \arg \min_{u_t \in \mathcal{U}_t} \sum_{i=1}^N p_{ti} [c_{ti}(x_t, u_t) + \underline{V}_{t+1}^k(A_{ti}x_t + B_{ti}u_t + b_{ti})], \quad (3.14)$$

for $x_t = \hat{x}_t$, and set $\hat{x}_{t+1} = F_t(\hat{x}_t, \hat{u}_t, \hat{\xi}_t)$. If the set \mathcal{U}_t is polyhedral, the cost functions $c_{ti}(x_t, u_t)$ are piecewise affine functions of u_t , this minimization problem can be written as a linear programming problem. In the next backward step of the algorithm, the cutting planes approximation of the value functions are updated going backwards in time by adding the cuts at the computed trial points \hat{x}_t . In computing the cuts use subgradients (at the trial points) of the current approximations of the value functions.

3.3 Risk-averse Stochastic Optimal Control

3.3.1 Risk-averse Setting

Consider the risk averse setting in the nested form. That is, the expectation operator in the risk neutral formulation (3.1) - (3.2) is replaced by the nested risk measure \mathfrak{R} , defined in (1.9) under the assumption that the data process is stagewise independent with respect to the reference distributions. Suppose further that the state equations are affine of the form (3.7). This leads to the following risk averse problem² (in the nested form)

$$\min \mathcal{R}_{1|\xi_0} \left(\mathbf{c}_1 + \mathcal{R}_{2|\xi_{[1]}} \left(\mathbf{c}_2 + \cdots + \mathcal{R}_{T|\xi_{[T-1]}} (\mathbf{c}_T) \right) + \mathbf{c}_{T+1} \right), \quad (3.15)$$

$$\text{s.t. } u_t \in \mathcal{U}_t \text{ and } x_{t+1} = A_t x_t + B_t u_t + b_t, \quad t = 1, \dots, T, \quad (3.16)$$

where we use notation $\mathbf{c}_t := c_t(x_t, u_t, \xi_t)$, $t = 1, \dots, T$, and $\mathbf{c}_{T+1} := c_{T+1}(x_{T+1})$. Recall that the optimization (minimization) in (3.15) is over policies which are functions of the data process and subject to the feasibility constraints (3.16). To alleviate notation, we will use (x_t, u_t) instead of $(x_t(\xi_{[t-1]}), u_t(\xi_{[t-1]}))$. The constraints in the above problem should be satisfied with probability one with respect to the reference measures.

The risk averse counterpart of dynamic equations (3.8) can be written as $V_{T+1}(x_{T+1}) = c_{T+1}(x_{T+1})$ and for $t = T, \dots, 1$,

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \mathcal{R}_t \left(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t) \right) \quad (3.17)$$

$$= \inf_{u_t \in \mathcal{U}_t, \theta_t \in \Theta} \mathbb{E}_{P_t} \left[\Psi \left(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t), \theta_t \right) \right], \quad (3.18)$$

where formulation (3.18) is obtained by applying definition (1.8) of \mathcal{R}_t . Note that it is possible to write dynamic equations (3.17) in terms of the (static) risk measures \mathcal{R}_t because of the basic assumption of stagewise independence of the process ξ_t (with respect to the reference measures) (e.g., [9, section 6.5.4, Remark 39]). The respective optimal policy is

²Recall that $\mathcal{R}_{1|\xi_0} = \mathcal{R}$.

defined by the optimal controls

$$\bar{u}_t(x_t) \in \arg \min_{u_t \in \mathcal{U}_t} \mathcal{R}_t(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)). \quad (3.19)$$

As in the risk neutral setting, we assume that the set of minimizers in the right hand side of (3.19) is *nonempty* for all possible realizations of state variables.

The developments of Section 3.2 can be adapted to this risk-averse framework. Under the convexity assumption (B), the value functions $V_t(\cdot)$ are convex in the risk averse setting as well. There are explicit formulas how to compute a subgradient of the functional $\mathcal{R} : \mathcal{Z} \rightarrow \mathbb{R}$ for various examples of risk measures (cf., [9, section 6.3.2]).

Recall definition (1.8) of risk measure \mathcal{R}_t . For x_t and the optimal control $\bar{u}_t = \bar{u}_t(x_t)$, determined by (3.19), consider a minimizer

$$\bar{\theta}_t \in \arg \min_{\theta_t \in \Theta} \mathbb{E}_{P_t} [\Psi(c_t(x_t, \bar{u}_t, \xi_t) + V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t), \theta_t)]. \quad (3.20)$$

Then, similar to (3.12) and using the Chain rule, a subgradient $\nabla V_t(x_t)$ of the value function V_t at x_t can be computed as

$$\nabla V_t(x_t) = \mathbb{E}_{P_t} \left[\Psi'(y_t, \bar{\theta}_t) \left(\nabla c_t(x_t, \bar{u}_t, \xi_t) + A_t^\top \nabla V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t) \right) \right], \quad (3.21)$$

where $\Psi'(y_t, \bar{\theta}_t)$ is a subgradient³ of $\Psi(\cdot, \bar{\theta}_t)$ at y_t , $\nabla c_t(x_t, \bar{u}_t, \xi_t)$ is a subgradient of $c_t(\cdot, \bar{u}_t, \xi_t)$ at x_t , $\nabla V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t)$ is a subgradient of V_{t+1} at $A_t x_t + B_t \bar{u}_t + b_t$, and $y_t := c_t(x_t, \bar{u}_t, \xi_t) + V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t)$.

As a special case, consider Example 1.1.1 of the Average Value-at-Risk measure. In that case the minimizer $\bar{\theta}$ in the right hand side of (1.10) is given by the $(1 - \alpha)$ -quantile of the considered distribution. That is, suppose that the reference distribution P_t has a finite number of N realizations with equal probabilities $1/N$. Then $\bar{\theta}_t$ can be computed by ar-

³If $\Psi(\cdot, \bar{\theta}_t)$ is differentiable at y_t , then $\Psi'(y_t, \bar{\theta}_t)$ is given by the derivative of $\Psi(\cdot, \bar{\theta}_t)$ at y_t .

ranging values $c_{ti}(x_t, \bar{u}_t) + V_{t+1}(A_{ti}x_t + B_{ti}\bar{u}_t + b_{ti})$, $i = 1, \dots, N$, in the increasing order and taking the respective empirical $(1 - \alpha)$ -quantile. Consequently, the required subgradient of the current approximation of the value function can be computed in a straightforward way (cf., [91]).

One important difference between the SOC and SP modeling is that in the SOC approach there is a clear separation between the states and controls. Because of the stagewise independence assumption, the value functions $V_t(x_t)$ are functions of the state variables only. The controls u_t and the corresponding values θ_t of the parameter vector are computed (estimated) simultaneously based on equation (3.18). That is, the estimated values of θ_t are functions of state x_t and optimal controls \bar{u}_t , based on a current approximation of the value function (see eq. (3.20)). This makes the computed estimates of θ_t to be consistent for the generated discretization (sample) of the marginal distribution of ξ_t . This is in contrast to the SP approach where the bias of the corresponding estimates of θ_t explodes exponentially with increase of the number of stages (cf., [89]).

3.3.2 Statistical Upper Bounds on the Value of the Policy

In this section, we discuss the construction of a statistical upper bound on the optimal value of the risk averse problem. As before, all probabilistic statements and expectations are taken with respect to the *reference distributions*. Let $\underline{V}_t(x_t)$, $t = 1, \dots, T$, be current approximations of the value functions. This defines the corresponding (approximate) policy (\hat{x}_t, \hat{u}_t) with

$$\hat{u}_t \in \arg \min_{u_t \in \mathcal{U}_t} \mathcal{R}_t(c_t(\hat{x}_t, u_t, \xi_t) + \underline{V}_{t+1}(A_t \hat{x}_t + B_t u_t + b_t)), \quad (3.22)$$

with value $\underline{V}_1(x_1)$ giving a lower bound for the optimal value of the problem.

For a given realization (scenario) ξ_1, \dots, ξ_T of the data process, \hat{x}_t and \hat{u}_t are computed in the forward step of the SDDP algorithm, and can be viewed as functions $\hat{x}_t = \hat{x}_t(\xi_{[t-1]})$

and $\hat{u}_t = \hat{u}_t(\xi_{[t-1]})$. When each reference probability distribution has a finite support (of N points), i.e., for the discretized version of the problem, these values are computable.

Now let $\hat{\theta}_t \in \Theta$ be a specified function of the data process, $\hat{\theta}_t = \hat{\theta}_t(\xi_{[t-1]})$, $t = 1, \dots, T$. Note that $\hat{\theta}_t$ is non-anticipative in the sense that it does not depend on unobserved values ξ_t, \dots, ξ_T at time t . Denote $\hat{c}_t := c_t(\hat{x}_t, \hat{u}_t, \xi_t)$, $t = 1, \dots, T$, and $\hat{c}_{T+1} := c_{T+1}(\hat{x}_{T+1})$. Consider the following sequence of random variables (functions of the data process) defined iteratively going backward in time: $\mathbf{v}_{T+1} := \hat{c}_{T+1}$ and

$$\mathbf{v}_t := \Psi(\hat{c}_t + \mathbf{v}_{t+1}, \hat{\theta}_t), \quad t = T, \dots, 1. \quad (3.23)$$

Of course, values \mathbf{v}_t depend on a choice of parameters $\hat{\theta}_t$. We will discuss an appropriate choice of $\hat{\theta}_t$ later. Our statistical upper bound on the value of a risk-averse approximate policy is given in the following proposition.

Proposition 3.3.1. *Consider the risk-averse problem (3.15) - (3.16). Let \mathbf{v}_t be the sequence of random variables (defined iteratively by (3.23)) associated with current approximations of the value functions. Then for $t = 1, \dots, T$,*

$$\mathcal{R}_{t|\xi_{[t-1]}}(\hat{c}_t + \dots + \mathcal{R}_{T|\xi_{[T-1]}}(\hat{c}_T + \hat{c}_{T+1})) \leq \mathbb{E}_{|\xi_{[t-1]}}[\mathbf{v}_t], \quad w.p.1. \quad (3.24)$$

In particular, $\mathbb{E}[\mathbf{v}_1]$ is greater than or equal to the value of the policy defined by the considered approximate value functions, and is an upper bound on the optimal value of the risk-averse problem.

Proof. For $t = T$, using the definition of \hat{u}_T and since $\hat{\theta}_T \in \Theta$, we get

$$\begin{aligned} \mathcal{R}_{T|\xi_{[T-1]}}(\hat{c}_T + \hat{c}_{T+1}) &= \inf_{u_T \in \mathcal{U}_T} \mathcal{R}_T \left(c_T(\hat{x}_T, u_T, \xi_T) + \hat{V}_{T+1}(A_T \hat{x}_T + B_T u_T + b_T) \right) \\ &\leq \mathbb{E}_{|\xi_{[T-1]}} \left[\Psi \left(c_T(\hat{x}_T, \hat{u}_T, \xi_T) + c_{T+1}(A_T \hat{x}_T + B_T \hat{u}_T + b_T), \hat{\theta}_T \right) \right] \\ &= \mathbb{E}_{|\xi_{[T-1]}}[\mathbf{v}_T]. \end{aligned}$$

We now use induction in t going backward in time. For $t - 1$ we have

$$\begin{aligned}
& \mathcal{R}_{t-1|\xi_{[t-2]}} \left(\hat{c}_{t-1} + \mathcal{R}_{t|\xi_{[t-1]}} \left(\hat{c}_t + \dots + \mathcal{R}_{T|\xi_{[T-1]}} (\hat{c}_T + c_{T+1}(\hat{x}_{T+1})) \right) \right) \\
& \leq \mathcal{R}_{t-1|\xi_{[t-2]}} \left(\hat{c}_{t-1} + \mathbb{E}_{|\xi_{[t-1]}}[\mathbf{v}_t] \right) \quad (\text{monotonicity and induction step}) \\
& \leq \mathbb{E}_{|\xi_{[t-2]}} \left[\Psi \left(\hat{c}_{t-1} + \mathbb{E}_{|\xi_{[t-1]}}[\mathbf{v}_t], \hat{\theta}_{t-1} \right) \right] \quad (\text{because } \hat{\theta}_{t-1} \in \Theta) \\
& = \mathbb{E}_{|\xi_{[t-2]}} \left[\Psi \left(\mathbb{E}_{|\xi_{[t-1]}}[\hat{c}_{t-1} + \mathbf{v}_t], \hat{\theta}_{t-1} \right) \right] \quad (\text{since } \hat{c}_{t-1} \text{ is a function of } \xi_{[t-1]}) \quad (3.25) \\
& \leq \mathbb{E}_{|\xi_{[t-2]}} \mathbb{E}_{|\xi_{[t-1]}} \left[\Psi \left(\hat{c}_{t-1} + \mathbf{v}_t, \hat{\theta}_{t-1} \right) \right] \quad (\text{by Jensen's inequality}) \\
& = \mathbb{E}_{|\xi_{[t-2]}} \left[\Psi \left(\hat{c}_{t-1} + \mathbf{v}_t, \hat{\theta}_{t-1} \right) \right] \\
& = \mathbb{E}_{|\xi_{[t-2]}} [\mathbf{v}_{t-1}].
\end{aligned}$$

This completes the induction step. □

Therefore, for a sample path (scenario) of the data process, an unbiased point estimate of an upper bound on the corresponding policy value can be computed recursively starting with $\mathbf{v}_{T+1} = c_{T+1}(\hat{x}_{T+1})$ and going backward in time using the iteration procedure (3.23). Finally \mathbf{v}_1 gives a point estimate of an upper bound on the corresponding value of the policy. Therefore by generating a sample of scenarios, of the random data process, and averaging the corresponding point estimates it is possible to construct the respective statistical upper bound for the optimal value of the risk averse problem.

The quality of such statistical bound depends on the choice of the parameter value function $\hat{\theta}_t$. It is natural to use the corresponding minimizer of the form (3.20). That is, to take

$$\hat{\theta}_t \in \arg \min_{\theta_t \in \Theta} \mathbb{E} \left[\Psi \left(c_t(\hat{x}_t, \hat{u}_t, \xi_t) + \underline{V}_{t+1}(A_t \hat{x}_t + B_t \hat{u}_t + b_t), \theta_t \right) \right]. \quad (3.26)$$

The so defined $\hat{\theta}_t$ is a function of \hat{x}_t and \hat{u}_t , which in turn are functions of $\xi_{[t-1]}$. For example, as it was pointed at the end of Section 3.3.1, in case of the Average Value-at-Risk measure such $\hat{\theta}_t$ can be easily computed by using the respective quantile. Note that even for $\hat{\theta}_t$ of the form (3.26) the inequality (3.24) can be strict. This is because Jensen's inequality

was used in derivations (3.25). Nevertheless, this approach performed well in the numerical experiments discussed in the next section.

Remark 3.3.1. We would like to point to the important difference between the corresponding SOC and SP approaches to construction of the statistical upper bound for the risk averse problems. Computation of the parameter $\hat{\theta}_t$ in (3.26) is based on the distribution of random vector ξ_t . When ξ_t has a finite number of realizations $\xi_{ti}, i = 1, \dots, N$, the parameter $\hat{\theta}_t$ is a function of all corresponding costs \hat{c}_{ti} and $A_{ti}, B_{ti}, b_{ti}, i = 1, \dots, N$, and therefore in a sense is a consistent estimate of $\bar{\theta}_t$ defined in (3.20). On the other hand, in the SP setting it was not possible to construct a computationally feasible consistent estimate of the respective parameter of the risk measure. As a result a straightforward attempt for computation of such statistical upper bound in the SP framework resulted in an exponential growth of the involved bias with increase of the number of stages, which made it practically useless (cf., [89]).

We close this section by presenting Algorithm 2 for computing the statistical upper bound for a T -stage SOC problem.

3.3.3 Q -factor Approach

When the function Ψ is not polyhedral, as for instance in the setting of ϕ -divergence example, the procedure requires solving nonlinear optimization programs. This could be inconvenient since nonlinear optimization solvers should be used, which are known to be less efficient than linear solvers. In the considered example of KL-divergence, this requires solving one-dimensional nonlinear programs, which does not pose a significant problem. In general, in order to keep the procedure to linear programming solvers, the Q -factor approach, discussed below, can be used. Note however that the Q -factor approach involves increasing the state space which could significantly slow down the convergence of the algorithm.

Algorithm 2 SDDP-type Algorithm for SOC Problem

- 1: Inputs: stage-wise independent samples $\xi_t := \{\xi_{tj}\}_{1 \leq j \leq N_t}, t = 1, \dots, T$, initializations of $V_t(\cdot) : \underline{V}_t^0(\cdot), t = 1, \dots, T$, initial point \hat{x}_1
 - 2: **for** $k = 1, 2, \dots, K$ **do**
 - 3: $\underline{V}_{T+1}^{k-1}(\cdot) = V_{T+1}$
 - 4: **for** $t = 1, \dots, T$ **do** ▷ Forward Step
 - 5: $\hat{u}_t = \arg \min_{u_t \in \mathcal{U}_t} \mathcal{R}_t(c_t(\hat{x}_t, u_t, \xi_t) + \underline{V}_{t+1}^{k-1}(A_t \hat{x}_t + B_t u_t + b_t))$
 - 6: Draw a sample $(\hat{A}_t, \hat{B}_t, \hat{b}_t)$ from $\{\xi_t\}$
 - 7: $\hat{x}_{t+1} = \hat{A}_t \hat{x}_t + \hat{B}_t \hat{u}_t + \hat{b}_t$
 - 8: **end for**
 - 9: **for** $t = T, \dots, 1$ **do** ▷ Backward Step
 - 10: $\hat{\theta}_t = \arg \min_{\theta_t \in \Theta} \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi(c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}^{k-1}(A_{tj} \hat{x}_t + B_{tj} \hat{u}_t + b_{tj}), \theta_t)$
 - 11: $v_t = \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi(c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}^{k-1}(A_{tj} \hat{x}_t + B_{tj} \hat{u}_t + b_{tj}), \hat{\theta}_t)$
 - 12: $g_t = \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi'(y_{tj}, \hat{\theta}_t) (\nabla c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + A_{tj}^\top \nabla \underline{V}_{t+1}^{k-1}(A_{tj} \hat{x}_t + B_{tj} \hat{u}_t + b_{tj}))$
- where
- 13: $y_{tj} := c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}^{k-1}(A_{tj} \hat{x}_t + B_{tj} \hat{u}_t + b_{tj})$
 - 14: $\underline{V}_t^k(x_t) = \max(\underline{V}_t^{k-1}(x_t), g_t^T(x_t - \hat{x}_t) + v_t)$
 - 15: **end for**
 - 16: Lower bound: $L_k = \underline{V}_1^k(\hat{x}_1)$
 - 17: **end for**
 - 18: Generate S sample paths $\xi'_j: \{\xi'_{tj}\}_{1 \leq t \leq T}, j = 1, \dots, S$, run forward step for each sample path $\xi'_j, j = 1, \dots, S$, obtain $(\hat{u}_{tj}, \hat{x}_{tj})_{1 \leq t \leq T}$ and $\hat{x}_{T+1,j}, j = 1, \dots, S$ ▷ Evaluation
 - 19: Set $\mathbf{v}_{T+1,j} = c_{T+1}(\hat{x}_{T+1,j}), j = 1, \dots, S$
 - 20: **for** $t = T, \dots, 1$ **do**
 - 21: $\hat{\theta}_t = \arg \min_{\theta_t \in \Theta} \frac{1}{S} \sum_{j=1}^S \Psi(c_t(\hat{x}_t, \hat{u}_t, \xi'_{tj}) + \underline{V}_{t+1}^K(A'_{tj} \hat{x}_t + B'_{tj} \hat{u}_t + b'_{tj}), \theta_t)$
 - 22: **for** $j = 1, \dots, S$ **do**
 - 23: $\mathbf{v}_{tj} = \Psi(c_t(\hat{x}_{tj}, \hat{u}_{tj}, \xi'_{tj}) + \mathbf{v}_{t+1,j}, \hat{\theta}_t)$
 - 24: **end for**
 - 25: **end for**
 - 26: $\bar{\mathbf{v}}_1 = \frac{1}{S} \sum_{j=1}^S \mathbf{v}_{1j}, \sigma^2 = \frac{1}{S-1} \sum_{j=1}^S (\mathbf{v}_{1j} - \bar{\mathbf{v}}_1)^2$
 - 27: Statistical upper bound: $U_S = \bar{\mathbf{v}}_1 + z_{1-\alpha} \sigma / \sqrt{S}$, where $\mathbb{P}(Z \geq z_{1-\alpha}) = \alpha, Z \sim \mathcal{N}(0, 1)$.
-

The following is a counterpart of the Q -factor approach popular in the SOC applications. Consider the dynamic equations (3.18) and define

$$Q_t(x_t, u_t, \theta_t) := \mathbb{E}_{P_t} \left[\Psi(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t), \theta_t) \right]. \quad (3.27)$$

We have that

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t, \theta_t \in \Theta} Q_t(x_t, u_t, \theta_t),$$

and hence the dynamic equations (3.18) can be written in terms of $Q_t(x_t, u_t, \theta_t)$ as

$$Q_t(x_t, u_t, \theta_t) = \mathbb{E}_{P_t} \left[\Psi \left(c_t(x_t, u_t, \xi_t) + \inf_{u_{t+1} \in \mathcal{U}_{t+1}, \theta_{t+1} \in \Theta} Q_{t+1}(A_t x_t + B_t u_t + b_t, u_{t+1}, \theta_{t+1}), \theta_t \right) \right]. \quad (3.28)$$

The cutting planes, SDDP type, algorithm can be applied directly to functions $Q_t(x_t, u_t, \theta_t)$ rather than to the value functions $V_t(x_t)$. In the backward step of the algorithm, subgradients with respect to x_t, u_t and θ_t , of the current approximations of the functions $Q_t(x_t, u_t, \theta_t)$, should be computed. An advantage of that approach is that the calculation of these subgradients does not require solving *nonlinear optimization* programs even if the function Ψ is not polyhedral⁴. On the other hand, this Q -factor approach involves increasing the state space from x_t to (x_t, u_t, θ_t) , which could make the convergence of the algorithm considerably slower.

3.4 Numerical Experiments

In this section numerical experiments are performed on the Brazilian Inter-connected Power System problem (we refer to [91] for more details on the problem description). All experiments were run using Python 3.8.5 under Ubuntu 20.04.1 LTS operating system with a 4.20 GHz Intel Core i7 processor and 32Gb RAM. We extended the `MSPPY` solver⁵[92]

⁴The function Ψ is not polyhedral, for example, in the ϕ -divergence case. In that case the SDDP algorithm, applied to the value functions $V_t(x_t)$, requires solving nonlinear programs of the form (1.11).

⁵<https://github.com/lingquant/msppy>

for the SDDP algorithm solving for the SOC problem. We report numerical results of the convergence guided by the deterministic lower bound and the statistical upper bound of the risk averse stochastic optimal control problem.

The hydro-thermal planning problem is a large-scale problem with $T = 120$ planning horizon stages and four state variables related to the energy reservoirs in four interconnected regions. The monthly energy inflows define the stochastic data process in the model. For the sake of simplicity, it is assumed in the experiments below that the random inflow process is stagewise independent. The (discretization) samples are generated from log-normal distributions (with 100 realizations at each stage) estimated from the historical data. Previous attempts to define a statistical upper bound have shown some of the challenges of this task. For example, the numerical results in [89] show that by formulating the problem as a risk-averse multistage stochastic program, the scale of the statistical upper bounds starts to explode when the number of stages T is more than 10.

We aim to demonstrate via the hydro-thermal planning problem, the effectiveness of the construction of the statistical upper bound proposed in Section 3.3. This suggests first to formulate the problem as a risk-averse optimal control model, and then to solve it by a variant of the SDDP algorithm, while preserving the number of stages, the states, and the data process in the original problem. More specifically, we construct the upper bound as explained in Section 3.3.2, detailed in Algorithm 2. We conduct experiments for risk measures of convex combination of expectation and AV@R and KL-divergence, as described in Examples 1.1.2 and 1.1.3, respectively. We solve both problems, and compute the corresponding statistical upper bounds, by an SDDP-type algorithm as described in Algorithm 2.

Implementation Details.

1. Convex combination of expectation and AV@R (Example 1.1.2): $((1 - \lambda)\mathbb{E}[\cdot] + \lambda\text{AV@R}_\alpha(\cdot))$. For this risk measure, we perform tests with $\alpha = 0.05$ and $\lambda \in$

$\{0, 0.5, 1\}$. When $\lambda = 0$, the problem becomes risk neutral, while $\lambda = 1$ corresponds to an extreme risk aversion.

In this setting, at each backward step and in the evaluation procedure (line 10 and line 21 in Algorithm 2), $\hat{\theta}_t$ can be computed by arranging values $c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj}), j = 1, \dots, N_t$, in the increasing order and taking the respective empirical $(1 - \alpha)$ -quantile. Moreover, in order to obtain a fast converging deterministic lower bound, we adopt the biased-sampling technique proposed in [93].

2. KL-divergence (Example 1.1.3). For this risk measure, we conduct experiments for $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-8}, 10^{-12}\}$, which corresponds to problems with different levels of risk aversion. In particular, when $\epsilon = 10^{-12}$, the problem is essentially a risk neutral problem, up to some numerical error.

In this case, at steps indicated by line 10 and line 21 in Algorithm 2, the following (one-dimensional) convex program:

$$\hat{\lambda}_t = \arg \min_{\lambda_t > 0} \{ \lambda_t \epsilon + \lambda_t \ln \mathbb{E}_{P_t} [e^{Z_t/\lambda_t}] \}, \quad (3.29)$$

where $Z_t := \{c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj})\}_{1 \leq j \leq N_t}$, was solved using `Scipy` solver.

Results. For risk measure $(1 - \lambda)\mathbb{E}[\cdot] + \lambda AV@R_\alpha(\cdot)$, with $\lambda = 0.5$, in order to examine the trend of the statistical upper bound, we compute the upper bound for the problem at every 10 iterations with a sample of size $S = 10$, by running 10 forward passes in parallel. Figure 3.1 displays the evolution of the deterministic lower bounds and the statistical upper bounds for the hydro-thermal planning problem for 3000 iterations. We can see from the figure that the statistical upper bound oscillates significantly for the first 500 iterations and then gradually stabilizes within narrow fluctuations. Table 3.1 reports, for different choices of λ , the statistical upper bounds obtained from Monte Carlo simulation using 3000 sam-

ples, along with the deterministic lower bounds and the relative gap ($\frac{\text{upper bound} - \text{lower bound}}{\text{lower bound}}$) at iteration 3000. From the results, it seems that the relative gap of the problem is not very sensitive to the level of risk aversion.

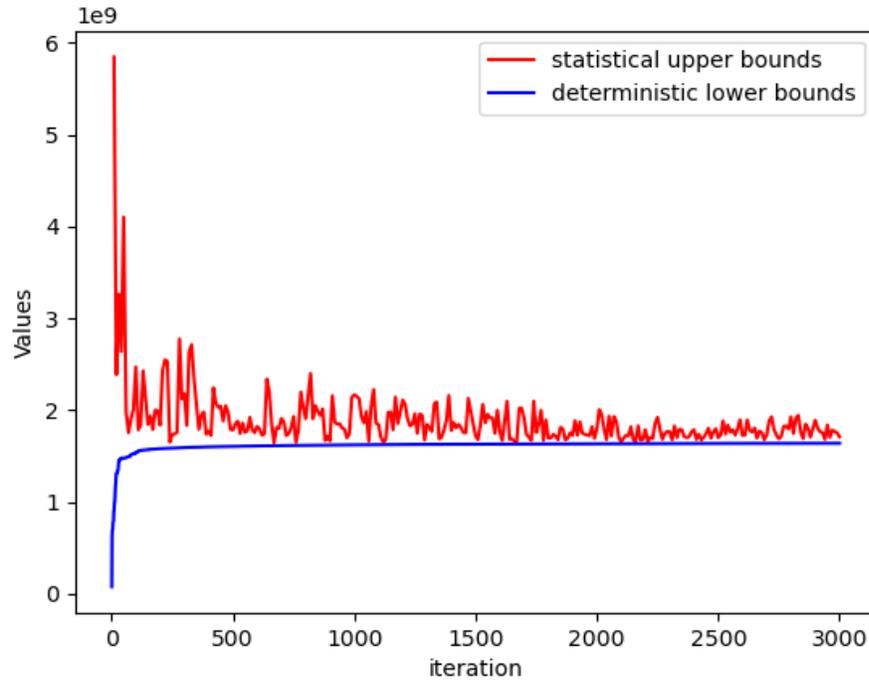


Figure 3.1: Evolution of lower and upper bounds for convex combination of expectation and AV@R problem when $\lambda = 0.5$.

Table 3.1: Convergence of convex combination of expectation and AV@R problem for different λ .

$(1 - \lambda)\mathbb{E}[\cdot] + \lambda\text{AV@R}_\alpha(\cdot)$			
λ	Deterministic lower bound ($\times 10^9$)	Statistical upper bound ($\times 10^9$)	Gap(%)
0.0	0.345	0.348	0.97
0.5	1.640	1.672	1.93
1.0	6.669	7.003	5.02

Table 3.2 reports results for the KL-divergence problem. The statistical upper bounds are computed by Monte Carlo simulation using $S = 3000$ samples, the lower bound and

the relative gap, are computed as well for difference values of ϵ . All results in the table are obtained when the problems are solved for 3000 iterations. We observe that when ϵ increases, the relative gap becomes larger.

Table 3.2: Convergence of KL-divergence problem for different ϵ .

KL-divergence			
ϵ	Deterministic lower bound ($\times 10^9$)	Statistical upper bound ($\times 10^9$)	Gap(%)
10^{-1}	4.894	5.959	21.76
10^{-2}	4.202	4.659	10.89
10^{-3}	3.991	4.306	7.88
10^{-8}	3.246	3.324	2.42
10^{-12}	0.339	0.342	1.03

There are two somewhat different reasons for the gap between the considered statistical upper and deterministic lower bounds. One reason is the optimality gap similar to the risk neutral case. The additional gap, as compared to the risk neutral setting, appears because Jensen's inequality is employed in derivations (3.25). This gap tends to increase as the function $\Psi(\cdot, \theta)$ becomes more "nonlinear". This can be clearly seen in Table 3.2, the gap increases with increase of ϵ , and also in Table 3.1 as the problem becomes more risk-averse.

CHAPTER 4

SAMPLE COMPLEXITY OF STATIONARY STOCHASTIC PROGRAMS

4.1 Overview

In this chapter we discuss the sample complexity of solving stationary stochastic programs by the Sample Average Approximation (SAA) method. We investigate this in the framework of Optimal Control (in discrete time) setting. In particular we derive a Central Limit Theorem type asymptotics for the optimal values of the SAA problems. The main conclusion is that the sample size, required to attain a given relative error of the SAA solution, is not sensitive to the discount factor, even if the discount factor is very close to one.

We demonstrate that the standard error (standard deviation) of the distribution of the optimal value of the SAA grows more or less at the same rate $O((1-\gamma)^{-1})$ as the respective optimal value. This supports the evidence of numerical experiments that variability of the sample error of the optimal values, measured in terms of the relative error, is not sensitive to the increase of the discount factor, even when the discount factor is very close to one. This is somewhat surprising since as is well known, it is becoming more difficult to solve the problem with increase of the discount factor. We investigate both the risk neutral and risk averse settings. The presented numerical experiments confirm the theoretical analysis.

The chapter is organized as follows. We first introduce the concept of the stationary stochastic programs in Section 4.2 and discuss the motivation therefrom. In Section 4.3 we present the theoretical analysis of sample complexity for risk neutral and risk averse problems. In particular, we show how the statistical upper bound of the SDDP algorithm can be constructed in the risk averse case. In Section 4.4 we discuss in detail the classical inventory model. Finally in Section 4.5 we present results of numerical experiments.

We use the following notation throughout the chapter. For a point ξ we denote by δ_ξ the

measure of mass one at ξ . For $a \in \mathbb{R}$, $[a]_+ := \max\{0, a\}$.

4.2 Stationary Stochastic Programs

Consider the following optimal control (in discrete time) infinite horizon problem

$$\begin{aligned} \min_{u_t \in \mathcal{U}} \quad & \mathbb{E}_P \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, u_t, \xi_t) \right] \\ \text{s.t.} \quad & x_{t+1} = F(x_t, u_t, \xi_t). \end{aligned} \tag{4.1}$$

Variables $x_t \in \mathbb{R}^n$ represent state of the system, $u_t \in \mathbb{R}^m$ are controls, $\xi_t \in \mathbb{R}^d$, $t = 0, \dots$, is a sequence of independent identically distributed (i.i.d.) random vectors (random noise or disturbances) with probability distribution P of ξ_t supported on set $\Xi \subset \mathbb{R}^d$, $c : \mathcal{X} \times \mathbb{R}^m \times \Xi \rightarrow \mathbb{R}$ is the cost function, $F : \mathcal{X} \times \mathbb{R}^m \times \Xi \rightarrow \mathcal{X}$ is a measurable mapping, $\mathcal{U} \subset \mathbb{R}^m$ and $\mathcal{X} \subset \mathbb{R}^n$ are nonempty closed sets, and $\gamma \in (0, 1)$ is the discount factor. Value x_0 is given (initial conditions). The notation \mathbb{E}_P emphasizes that the expectation is taken with respect to the probability distribution P of ξ_t . In such setting, problem (4.1) is the classical formulation of *stationary* optimal control (in discrete time) problem (e.g., [75]).

Problem (4.1) can be also considered in the framework of stochastic programming by viewing $y_t = (x_t, u_t)$ as decision variables (e.g., [6]). In case the problem is convex, it is possible to apply a Stochastic Dual Dynamic Programming (SDDP) cutting plane type algorithm for a numerical solution. For periodical infinite horizon stochastic programming problems such algorithms were discussed in [76], problem (4.1) can be viewed as a particular case of the periodical setting with the period of one. In order to solve (4.1) numerically the (generally continuous) distribution of the random process ξ_t should be discretized. The so-called Sample Average Approximation (SAA) method approaches this by generating a random sample of the (marginal) distribution of ξ_t by using Monte Carlo sampling techniques.

This raises the question of the involved sample complexity, i.e., how large should be the sample size N in order for the SAA problem to give an accurate approximation of

the original problem. In some applications the discount factor γ is very close to one. It is well known that as the discount factor approaches one, it becomes more difficult to solve problem (4.1). For a given $\gamma \in (0, 1)$, the sample complexity of the discretization is discussed in [76], with the derived upper bound on the sample size N being of order $O((1 - \gamma)^{-3}\epsilon^{-2})$ as a function of the discount factor γ and the error level $\epsilon > 0$. Since the optimal value of problem (4.1) increases at the rate of $O((1 - \gamma)^{-1})$ as γ approaches one, in terms of the relative error $(1 - \gamma)^{-1}\epsilon$, this would imply the required sample size is of order $O((1 - \gamma)^{-1})$ as a function of γ . This suggests that increasing γ from 0.99 to 0.999 would require to increase the sample size by the factor of 10 in order to achieve more or less the same relative accuracy of the SAA method. However, the above is just an *upper* bound and some numerical experiments indicate that the relative error of the SAA approach is not much sensitive to increase of the discount factor even when it is very close to one.

4.3 Sample Complexity Analysis

The (classical) Bellman equation for the value function, associated with problem (4.1), can be written as

$$V(x) = \inf_{u \in \mathcal{U}} \mathbb{E}_P [c(x, u, \xi) + \gamma V(F(x, u, \xi))], \quad x \in \mathcal{X}. \quad (4.2)$$

Consider the following assumptions.

(A1) The cost function is *bounded*, i.e., there is a constant $\kappa > 0$ such that $|c(x, u, \xi)| \leq \kappa$ for all $(x, u, \xi) \in \mathcal{X} \times \mathcal{U} \times \Xi$.

(A2) The function $c(\cdot, \cdot, \cdot)$ and the mapping $F(\cdot, \cdot, \cdot)$ are continuous on the set $\mathcal{X} \times \mathcal{U} \times \Xi$.

Let $\mathbb{B}(\mathcal{X})$ be the space of bounded functions $g : \mathcal{X} \rightarrow \mathbb{R}$ equipped with the sup-norm $\|g\|_\infty = \sup_{x \in \mathcal{X}} |g(x)|$. Then, under the assumption (A1), $V(\cdot)$ is the fixed point of mapping $\mathcal{T} : \mathbb{B}(\mathcal{X}) \rightarrow \mathbb{B}(\mathcal{X})$ defined as

$$\mathcal{T}(g)(x) := \inf_{u \in \mathcal{U}} \mathbb{E}_P [c(x, u, \xi) + \gamma g(F(x, u, \xi))], \quad g \in \mathbb{B}(\mathcal{X}). \quad (4.3)$$

As is well known, the mapping \mathcal{T} is a contraction mapping for $\gamma < 1$. Thus equations (4.2) have unique solution \bar{V} (e.g., [75]). For $x = x_0$, the corresponding optimal policy is given by $\bar{u}_t = \pi(x_t)$, $t = 0, \dots$, with

$$\pi(x) \in \arg \min_{u \in \mathcal{U}} \mathbb{E}_P [c(x, u, \xi) + \gamma \bar{V}(F(x, u, \xi))]. \quad (4.4)$$

For a given $x = x_0$ consider $\vartheta(P) := \bar{V}(x)$ viewed as a function of the probability measure P . Given a sample ξ^j , $j = 1, \dots, N$, of the random vector ξ , consider the corresponding empirical measure $\hat{P}_N = N^{-1} \sum_{j=1}^N \delta_{\xi^j}$. We are interested in the asymptotics of the value function $\hat{V}_N(x) = \vartheta(\hat{P}_N)$ of the corresponding SAA problem. That is, we would like to derive a Central Limit Theorem for $N^{1/2}(\hat{V}_N(x) - \bar{V}(x))$ for a fixed point $x \in \mathcal{X}$.

We can approach this problem in the following way. For a probability measure Q and $\tau \in [0, 1]$, consider probability measure $(1 - \tau)P + \tau Q = P + \tau(Q - P)$, and the directional derivative (if it exists)

$$\vartheta'(P, Q - P) := \lim_{\tau \downarrow 0} \frac{\vartheta(P + \tau(Q - P)) - \vartheta(P)}{\tau}. \quad (4.5)$$

Then we can use the approximation

$$\vartheta(\hat{P}_N) - \vartheta(P) \approx \vartheta'(P, \hat{P}_N - P). \quad (4.6)$$

This is the approach of Von Mises statistical functionals. It requires to compute the directional derivative (4.5), and consequently uses approximation (4.6) to derive the asymptotics. Even if this directional derivative does exist, the approximation (4.6) is a heuristic (this approach is routinely used in Statistics). In order to justify the obtained asymptotics in a rigorous way often the functional Delta Theorem is employed, we will discuss this later.

To compute the directional derivative (4.5) we proceed as follows. Consider the set of

optimal policies

$$\mathfrak{S}(x) := \arg \min_{u \in \mathcal{U}} \mathbb{E}_P [c(x, u, \xi) + \gamma \bar{V}(F(x, u, \xi))], \quad (4.7)$$

where $\bar{V}(\cdot)$ is the solution of Bellman equation (4.2). Under the assumptions (A1) and (A2) the value function $\bar{V}(\cdot)$ is continuous, and hence the set $\mathfrak{S}(x)$ is nonempty, provided the set \mathcal{U} is compact.

Note that for any $\pi(x) \in \mathfrak{S}(x)$, the value function of the true problem can be written as

$$\bar{V}(x) = \mathbb{E}_P [\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t)], \quad (4.8)$$

with $x_{t+1} = F_t(x_t, \pi(x_t), \xi_t)$, $x_0 = x$, $t \geq 0$. Consider the following formula for the directional derivative (4.5),

$$\vartheta'(P, Q - P) = \inf_{\pi(x) \in \mathfrak{S}(x)} \mathbb{E}_{Q-P} [\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t)], \quad x = x_0. \quad (4.9)$$

We will give a proof of formula (4.9) in some cases and discuss difficulties associated with a rigorous derivation of (4.9) for a general setting.

Since

$$\vartheta(P) = \mathbb{E}_P [\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t)], \quad \text{for } \pi(x) \in \mathfrak{S}(x), \quad (4.10)$$

by (4.6) this leads to the approximation

$$\vartheta(\hat{P}_N) - \vartheta(P) \approx \inf_{\pi(x) \in \mathfrak{S}(x)} \mathbb{E}_{\hat{P}_N - P} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right] \quad (4.11)$$

$$= \inf_{\pi(x) \in \mathfrak{S}(x)} \mathbb{E}_{\hat{P}_N} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right] - \vartheta(P). \quad (4.12)$$

- In particular if $\mathfrak{S}(x) = \{\bar{\pi}(x)\}$ is a singleton, then by the CLT the approximation (4.12) suggests that $N^{1/2}(\vartheta(\hat{P}_N) - \vartheta(P))$ converges in distribution to normal

$\mathcal{N}(0, \sigma^2(x_0))$ with

$$\sigma^2(x) = \text{Var} \left(\sum_{t=0}^{\infty} \gamma^t c(x_t, \bar{\pi}(x_t), \xi_t) \right). \quad (4.13)$$

Note that in the approximation (4.12) the set of optimal policies $\pi(x)$ is computed with respect to the distribution P of ξ_t , and that the variance in (4.13) is taken with respect to the distribution P as well.

Consider an optimal policy $\pi(x) \in \mathfrak{S}(x)$ (for the true problem). Since this policy is feasible we have that

$$\vartheta(P + \tau(Q - P)) \leq \mathbb{E}_{P + \tau(Q - P)} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right], \quad \tau \in [0, 1].$$

Together with (4.10) this implies

$$\limsup_{\tau \downarrow 0} \frac{\vartheta(P + \tau(Q - P)) - \vartheta(P)}{\tau} \leq \inf_{\pi(x) \in \mathfrak{S}(x)} \mathbb{E}_{Q - P} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right]. \quad (4.14)$$

This gives the upper bound for the directional derivative. In order to derive the respective lower bound there is a need for some type of compactness condition.

Consider the set \mathfrak{P} of measurable mappings $\pi : \mathcal{X} \rightarrow \mathcal{U}$. Equipped with the distance

$$d(\pi_1, \pi_2) := \sup_{x \in \mathcal{X}} \|\pi_1(x) - \pi_2(x)\|,$$

the set \mathfrak{P} becomes a metric space. We can view any $\pi \in \mathfrak{P}$ as a policy for the considered infinite horizon problem. For a given distribution P , the optimal policy is obtained by choosing $\pi \in \mathfrak{P}$ which minimizes the right hand side of (4.4). Suppose that we can choose a subset $\mathfrak{P}^* \subset \mathfrak{P}$ such that by restricting the optimization to $\pi \in \mathfrak{P}^*$ the corresponding optimal value does not change for all probability measures of the form $P + \tau(Q - P)$, $\tau \in [0, 1]$. We refer to such set \mathfrak{P}^* as the *restricted set*, and to the corresponding metric

space (\mathfrak{P}^*, d) as the restricted metric space. Of course choice of the restricted set \mathfrak{P}^* is associated with the probability measures P and Q . If we can choose the restricted the metric space (\mathfrak{P}^*, d) to be compact, then we can proceed to the following proof.

Proposition 4.3.1. *Suppose that the assumptions (A1) and (A2) are satisfied and there exists the restricted compact metric space (\mathfrak{P}^*, d) . Then formula (4.9) holds.*

Proof. For $\tau \in [0, 1]$ and policy $\pi \in \mathfrak{P}^*$ consider function

$$h(\tau, \pi) := \mathbb{E}_{P+\tau(Q-P)} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right],$$

with $x = x_0$ and $x_{t+1} = F(x_t, \pi(x_t), \xi_t)$ for $t \geq 0$. We have that

$$\frac{\partial h(\tau, \pi)}{\partial \tau} = \mathbb{E}_{Q-P} \left[\sum_{t=0}^{\infty} \gamma^t c(x_t, \pi(x_t), \xi_t) \right]. \quad (4.15)$$

By the Lebesgue dominated convergence theorem, the right hand side of (4.15) is continuous with respect to $\pi \in \mathfrak{P}^*$. Formula (4.9) now follows by Danskin's theorem (e.g., [70, Theorem 4.13]) applied to the function $h(\tau, \pi)$.

The main technical difficulty in applying the above proposition is verification of existence of the restricted *compact* metric space (\mathfrak{P}^*, d) . Note that the metric space (\mathfrak{P}, d) is compact if either the set \mathcal{X} is finite and the set \mathcal{U} is compact, or the set \mathcal{U} is finite. In such cases we can take $\mathfrak{P}^* = \mathfrak{P}$.

4.3.1 Risk Averse Case

Let \mathcal{U} be a law invariant coherent risk measure (cf., [94]), and consider the corresponding nested formulation of stationary inventory model. In that case Bellman equation can be written, similar to (4.21), as (e.g., [76])

$$V(x) = \inf_{u \in \mathcal{U}} \mathcal{U} \left[c(x, u, \xi) + \gamma V(F(x, u, \xi)) \right], \quad x \in \mathcal{X}. \quad (4.16)$$

For example we can consider the Average Value-at-Risk measure (also called Conditional Value-at-Risk, Expected Shortfall, Expected Tail Loss)

$$\text{CVaR}_\alpha(Z) = \inf_{\eta \in \mathbb{R}} \mathbb{E}_P \{ \eta + \alpha^{-1} [Z - \eta]_+ \}, \quad \alpha \in (0, 1).$$

Then equation (4.16) takes the form

$$V(x) = \inf_{u \in \mathcal{U}, \eta \in \mathbb{R}} \mathbb{E}_P \{ \eta + \alpha^{-1} [c(x, u, \xi) + \gamma V(F(x, u, \xi)) - \eta]_+ \}. \quad (4.17)$$

Let $(\bar{\pi}(x), \bar{\eta}(x))$ be an optimal solution of (4.17). Then the optimal value of the corresponding nested infinite horizon problem is given by

$$\mathbb{E}_P \left[\sum_{t=0}^{\infty} \gamma^t \left(\bar{\eta}(x_t) + \alpha^{-1} [c(x_t, \bar{\pi}(x_t), \xi_t) - \bar{\eta}(x_t)]_+ \right) \right]. \quad (4.18)$$

Suppose that the optimal solution $(\bar{\pi}(x), \bar{\eta}(x))$ is unique. By derivations similar to the risk neutral (expected value) case, this suggests that $N^{1/2}(\vartheta(\hat{P}_N) - \vartheta(P))$ converges in distribution to normal $\mathcal{N}(0, \sigma^2(x))$ with

$$\sigma^2(x) = \text{Var} \left(\sum_{t=0}^{\infty} \gamma^t \left(\bar{\eta}(x_t) + \alpha^{-1} [c(x_t, \bar{\pi}(x_t), \xi_t) - \bar{\eta}(x_t)]_+ \right) \right). \quad (4.19)$$

4.4 Inventory Model

Consider the stationary inventory model (cf., [77])

$$\begin{aligned} \min_{u_t \geq 0} \quad & \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t (cu_t + b[D_t - (x_t + u_t)]_+ + h[x_t + u_t - D_t]_+) \right] \\ \text{s.t.} \quad & x_{t+1} = x_t + u_t - D_t, \end{aligned} \quad (4.20)$$

where $c, b, h \in \mathbb{R}_+$ are the ordering cost, backorder penalty cost and holding cost per unit, respectively (with $b > c \geq 0$), x_t is the current inventory level, u_t is the order quantity,

and $D_t \in \mathbb{R}_+$ is the demand at time t which is a random iid process. Then the optimal policy is myopic basestock policy $\bar{\pi}(x) = [x^* - x]_+$, where $x^* = F^{-1}\left(\frac{b-(1-\gamma)c}{b+h}\right)$ with $F(x) = P(D \leq x)$ being the cdf of the demand (e.g., [77]). The optimal (basestock) policy is $\bar{u}_t = [x^* - x_t]_+$, and $x_{t+1} = x_t + \bar{u}_t - D_t$. That is $\bar{u}_t = x^* - x_t$ if $x_t \leq x^*$, and $\bar{u}_t = 0$ if $x_t \geq x^*$. Consequently $x_{t+1} = x^* - D_t$ if $x_t \leq x^*$, and $x_{t+1} = x_t - D_t$ if $x_t \geq x^*$.

The corresponding Bellman equation can be written as

$$V(x) = \inf_{u \geq 0} \mathbb{E}_P[cu + \psi(x + u, D) + \gamma V(x + u - D)], \quad x \in \mathbb{R}, \quad (4.21)$$

with $D \sim P$ and

$$\psi(x, D) := b[D - x]_+ + h[x - D]_+.$$

Substituting $\bar{u}(x) = [x^* - x]_+$ into the right hand side of (4.21) we obtain,

$$V(x) = -cx + cx^* + \mathbb{E}_P[\psi(x^*, D) + \gamma V(x^* - D)], \quad \text{for } x \leq x^*, \quad (4.22)$$

$$V(x) = \mathbb{E}_P[\psi(x, D) + \gamma V(x - D)], \quad \text{for } x \geq x^*. \quad (4.23)$$

Since D is nonnegative we have that $x^* - D \leq x^*$, and hence by (4.22) that

$$V(x^* - D) = cD + \mathbb{E}_P[\psi(x^*, D) + \gamma V(x^* - D)].$$

It follows that for $x \leq x^*$,

$$V(x) = -cx + cx^* + \mathbb{E}_P[\gamma cD + \psi(x^*, D) + \gamma \psi(x^*, D) + \gamma^2 V(x^* - D)].$$

By continuing this process we obtain for $x \leq x^*$,

$$V(x) = -cx + (1 - \gamma)^{-1} \mathbb{E}_P[\gamma cD + (1 - \gamma)cx^* + \psi(x^*, D)]. \quad (4.24)$$

Note that $x^* \in \mathfrak{X}$, where

$$\mathfrak{X} := \arg \min_{x \in \mathbb{R}} \mathbb{E}_P [(1 - \gamma)cx + \psi(x, D)].$$

Then by [6, Theorem 5.7] we have the following result.

Theorem 4.4.1. *For $x \leq x^*$ it holds that*

$$\hat{V}_N(x) = -cx + (1 - \gamma)^{-1} \inf_{x \in \mathfrak{X}} \mathbb{E}_{\hat{P}_N} [\gamma cD + (1 - \gamma)cx + \psi(x, D)] + o_p(N^{-1/2}). \quad (4.25)$$

In particular if the set \mathfrak{X} is the singleton, i.e. the quantile x^ is unique, then $N^{1/2}(\hat{V}_N(x) - V(x))$ converges in distribution to normal $\mathcal{N}(0, \sigma^2)$ with*

$$\sigma^2 = (1 - \gamma)^{-2} \text{Var}(\gamma cD + \psi(x^*, D)). \quad (4.26)$$

The variance in (4.26) is taken with respect to the distribution P of the demand. In the present case it was possible to derive the corresponding asymptotics of the form (4.12) in the rigorous way.

4.4.1 Risk Averse Case

Let \mathcal{R} be a law invariant coherent risk measure and consider the corresponding nested formulation of stationary inventory model. In that case Bellman equation can be written, similar to (4.21), as

$$V(x) = \inf_{u \geq 0} \mathcal{R} [\psi(x, u, D) + \gamma V(x + u - D)], \quad x \in \mathbb{R}. \quad (4.27)$$

For example we can consider the Average Value-at-Risk measure $\mathcal{R}(\cdot) := \text{CVaR}_\alpha(\cdot)$. The base stock policy is optimal here as well with

$$x^* \in \arg \min_{x \in \mathbb{R}} \mathcal{R}(cx + \psi(x, D) + \gamma V(x - D)). \quad (4.28)$$

Counterparts of equations (4.22) and (4.23) follow here with the expectation \mathbb{E}_P replaced by the risk measure \mathcal{R} .

4.5 Numerical Illustration

In this section, we present numerical illustration of the relationship between standard deviation of the optimal value functions and discount factors for the stationary control problems. Numerical experiments are performed on the stationary inventory problem and the Brazilian Inter-connected Power System problem with risk neutral and risk averse formulations. The second test problem is reformulated to have stationary data structure.

4.5.1 Test Cases and Experimental Settings

Inventory Problem. The stationary inventory problem has stagewise independent uncertain demands D and deterministic ordering cost c , holding cost h and backlogging cost b , following the description in section 4.4. For the numerical test, the first stage is set to be deterministic with $D_1 = 5.5$ and initial state $x_0 = 10.0$. For the second stage and onwards, the model is stationary with $h = 0.2$, $b = 2.8$, $c = \cos(\frac{\pi}{3}) + 1.5$ and the demand is predicted by

$$D = d + \phi \cdot \xi, \quad (4.29)$$

where $d = 9.0$, $\phi = 0.6$ and ξ is uniformly distributed in the interval $[0, 1]$.

Hydro-thermal Planning Problem. The hydro-thermal planning problem has larger scale than the inventory problem. The original problem has total number of stages $T = 120$ and

4 state variables corresponding to the energy equivalent reservoirs of 4 interconnected regions. The random data process is characterized by the underlying stochastic monthly energy inflows. Specifically, the monthly inflows are sampled from a log-normal distributions trained from the historical data and is assumed to be stagewise independent. We refer to [80] for more details of the problem. For illustration purpose, the model assumes that the energy inflows have period 1, that is, the distribution of the inflows from the second stage and onwards (first stage is deterministic) are the same. In this way, the model has stationary data structure.

For each problem, we discretize the continuous random variables at each stage into N realizations, and approximate the true problem by its SAA counterpart. To illustrate the sample complexity for the stationary programs, we consider different sample sizes for discretization: $N = 10, 50, 100$. Besides, we perform numerical tests with different discount factors $\gamma = 0.8, 0.9, 0.9906$ and 0.999 .

Note that each SAA problem is a function of a sample with size N . By randomizing SAA problems for M times, we obtain M optimal values of the SAA problems corresponding to different samples. If M is sufficiently large, these samples then approximately follow normal distribution. Therefore, the variability of the optimal value functions can be measured by the sample standard deviation.

To elaborate, let $\hat{V}^{(r)}$ denote the optimal value (up to some precision) of the SAA problem related to the r -th sample, for $1 \leq r \leq M$. Then the sample standard deviation is computed by

$$\hat{\sigma} = \sqrt{\frac{1}{M-1} \sum_{r=1}^M [\bar{V}_M - \hat{V}^{(r)}]^2}, \quad (4.30)$$

where $\bar{V}_M = \frac{1}{M} \sum_{r=1}^M \hat{V}^{(r)}$.

In the numerical tests, we choose $M = 100$ in order to achieve high significance in the normality test. We will present more details of how to compute $\hat{V}^{(r)}$ later in this section.

For each risk measure, a test instance is determined by selections of N and γ . We

conduct the numerical experiments for each test instance in the following three steps:

1. Run the Periodical SDDP type algorithm to solve M SAA problems and obtain primal bounds $\hat{V}^{(r)}$, $r = 1, \dots, M$.
2. Construct upper bounds for the SAA problems and compare with $\hat{V}^{(r)}$ to check convergence. For risk neutral formulations, dual bounds are accessible for all sample sizes N and discount factors γ . The dual bounds were constructed according to [2] with period equal to 1. For risk averse formulations, only statistical upper bounds are available for discount factors $\gamma = 0.8, 0.9$ and all sample sizes N .
3. Compute sample standard deviation of the optimal values of all SAA problems according to (4.30).
4. For inventory problem, compute theoretical standard deviation for risk neutral case by (4.26) and risk averse case by (4.28). Compare the results with those from step 3.

All implementations were written in Python 3 using the MSPPY solver described in [22] and the `dualsddp` described in [2].

4.5.2 Risk Neutral Case

In this section, we report numerical results for the risk neutral formulation of the stationary inventory problem and the hydro-thermal planning problem.

In 4.1, we provide a summary of solving the SAA problem of the stationary inventory problem and the hydro-thermal planning problem for different test instances. The first two columns represent the identity (discretized sample size and discount factor) of the test case. Column 3 and 4 give the deterministic lower bounds (primal bounds) and upper bounds (dual bounds) of the problems. The last column reports the relative gap calculated by $\frac{UB-LB}{LB} \times 100\%$. Observe that for each sample size N , the gaps for different discount factors remain in low level. This shows that increasing the discount factor from γ_1 to γ_2

does not require to increase the sample size by the factor of $(1 - \gamma_2)^{-1}/(1 - \gamma_1)^{-1}$ in order to achieve similar convergence in solving the SAA analogue of the true problem.

Table 4.1: Risk neutral case: convergence of solving SAA problems.

Inventory problem				
N	γ	LB	UB	Gap(%)
10	0.8	67.210	67.238	4.17×10^{-2}
	0.9	158.196	158.283	5.5×10^{-2}
	0.9906	1928.66	1933.93	0.27
	0.999	18227.69	18408.19	0.993
50	0.8	67.941	67.98	5.74×10^{-2}
	0.9	159.84	159.93	5.63×10^{-2}
	0.9906	1947.93	1953.19	0.27
	0.999	18409.09	18629.29	1.19
100	0.8	68.032	68.06	4.12×10^{-2}
	0.9	160.05	160.13	4.998×10^{-2}
	0.9906	1950.32	1956.33	0.31
	0.999	18431.61	18675.72	1.32
Hydro-thermal planning problem				
N	γ	LB ($\cdot 10^6$)	UB ($\cdot 10^6$)	Gap(%)
50	0.8	1.2259	1.2287	0.23
	0.9	2.4518	2.4961	1.77
	0.9906	26.0858	26.2726	1.03
	0.999	243.5701	257.7629	5.5
100	0.8	1.2259	1.2276	0.14
	0.9	2.4519	2.5591	4.19
	0.9906	26.0863	26.3817	1.12
	0.999	243.5745	257.3031	5.33

In 4.2, we present sample standard deviation computed from $M = 100$ optimal values of the SAA problems. The first two columns of the table account for the identity of the test instances. The third column displays the sample standard deviation of 100 optimal values of the SAA problems for each test instance according to (4.30). The last column is the result of multiplying the sample standard deviation and the value $(1 - \gamma)$. Additionally in table 4.3, we report for the inventory problem the theoretical standard deviation of the

optimal value functions for each discount factor, which is computed according to (4.26). We make the following observations. First, the sample standard deviations of the optimal values of the SAA problems almost proportional to the factor $(1 - \gamma)^{-1}$. Evidence can be found in the last column of 4.2, which demonstrates that for each N , the values of $\hat{\sigma}_N \cdot (1 - \gamma)$ resemble each other for different discount factors. This is also the case for the theoretical standard deviations derived from the inventory model (see third column of 4.3). Second, the sample standard deviations are close to the theoretical ones. For the inventory problem, comparisons between $\hat{\sigma}_N \cdot (1 - \gamma)$ in 4.2 and $\sigma \cdot (1 - \gamma)/\sqrt{N}$ in 4.3 for each N and γ supports such claim. For the hydro-thermal planning problem, the closed form of standard deviation of the optimal value function is not known, thus we only report the sample standard deviation. However, the closeness of the sample standard deviation and the theoretical one of the inventory problem sheds some light on same conjecture for other stationary problems.

As the empirical results suggest that the standard deviations of different discount values are proportional to the factor $(1 - \gamma)^{-1}$, it is not surprising to see that convergence of the risk neutral SAA problems with different discount factors do not vary much.

4.5.3 Risk Averse Case

Numerical experiments for the risk averse case adopt the risk measure of weighted sum of expectation and the Conditional Value-at-Risk ($\text{CVaR}_\alpha(\cdot)$) with parameter λ (the weighted sum parameter) and α (the confidence level). For the inventory problem, we choose $\lambda = 0.2$, $\alpha = 0.05$; for the hydro-thermal planning problem, $\lambda = 0.5$, $\alpha = 0.05$. For the selected risk measure, the analogue of formula (4.18) is given by

$$\mathbb{E}_P \left[\sum_{t=0}^{\infty} \gamma^t \left((1 - \lambda)c(x_t, \bar{\pi}(x_t), \xi_t) + \lambda \left(\bar{v}(x_t) + \alpha^{-1} [c(x_t, \bar{\pi}(x_t), \xi_t) - \bar{v}(x_t)]_+ \right) \right) \right]. \quad (4.31)$$

We apply the risk averse SDDP algorithm with biased sampling (see [95]) to solve the

Table 4.2: Risk neutral case: sample standard deviations of optimal values of $M = 100$ SAA problems.

Inventory problem			
N	γ	$\hat{\sigma}_N$	$\hat{\sigma}_N \cdot (1 - \gamma)$
10	0.8	0.52779	0.10556
	0.9	1.05557	0.10556
	0.9906	11.2296	0.10556
	0.999	104.849	0.104849
50	0.8	0.25295	0.05059
	0.9	0.50590	0.05059
	0.9906	5.3819	0.05059
	0.999	50.2507	0.05025
100	0.8	0.16361	0.03272
	0.9	0.32722	0.03272
	0.9906	3.48112	0.03272
	11 0.999	32.5026	0.032503
Hydro-thermal planning problem			
N	γ	$\hat{\sigma}_N$	$\hat{\sigma}_N \cdot (1 - \gamma)$
50	0.8	30.9749	6.195
	0.9	63.0804	6.3089
	0.9906	701.624	6.5959
	0.999	6552.6559	6.553
100	0.8	22.9013	4.5803
	0.9	45.3019	4.5302
	0.9906	516.5621	4.8557
	0.999	4852.4276	4.8524

SAA problems. To construct the upper bounds, we compute the statistical upper bounds for the expected policy value in (4.31). Specifically, we replace ∞ with a large value of T in (4.31) to approximate the true policy value. Here, we choose $T = 120$ for the numerical experiments. For discount factors very close to 1 (e.g. $\gamma = 0.9906, 0.999$), it is very challenging to compute a valid statistical upper bound (see [2]). For this reason, we only provide statistical upper bounds for SAA problems with discount factors $\gamma = 0.8$ and $\gamma = 0.9$ in the risk averse case. When solving SAA problems with larger discounts ($\gamma = 0.9906, 0.999$), we adopt the stopping criteria as when the deterministic bounds (primal

Table 4.3: Risk neutral case: theoretical standard deviation of the optimal value function for the inventory problem.

γ	σ	$\sigma \cdot (1 - \gamma)$	$\sigma \cdot (1 - \gamma) / \sqrt{10}$	$\sigma \cdot (1 - \gamma) / \sqrt{50}$	$\sigma \cdot (1 - \gamma) / \sqrt{100}$
0.8	1.4596	0.2919	0.0923	0.0412	0.02919
0.9	3.1812	0.3181	0.1006	0.04499	0.03497
0.9906	36.9038	0.3469	0.1097	0.04906	0.03469
0.999	349.7119	0.3497	0.1106	0.04946	0.03497

lower bounds) become stablized.

4.4 presents the lower bounds and 95% confidence intervals for the SAA problems(if applicable). The confidence intervals are computed based on the policy values evaluated on the policy by generating 1000 sample paths. Gaps are computed via $\frac{UB-LB}{LB} \times 100\%$ where UB denotes the upper end of the confidence interval. Additionally, in the spirit of understanding the evolution of the upper bounds and the lower bounds, we compute for each iteration the lower bounds and confidence intervals based on the policy values obtained from 6 forward passes. The results (for the SAA problems of the inventory problem with $\gamma = 0.8$ and different sample sizes N) demonstrated in 4.3 show that the upper ends of the confidence intervals are larger than the lower bounds. Besides, for both problems with relatively small discount factors, the gaps are evident to show convergence.

Similar to 4.2, 4.5 reports the sample standard deviations of the optimal values collected from solving $M = 100$ risk averse SAA problems for each test instance. Likewise, by solving $M = 100$ risk averse SAA problems for each test instance, we obtain the sample standard deviations of the optimal values, denoted by $\hat{\sigma}$, which are almost proportional to the factor $(1 - \gamma)^{-1}$, similar to the results displayed in 4.2. For the inventory problem, following the formula in (4.28), we can also compute the closed form of the standard deviation of the optimal value function under the risk measure mentioned above, where the base stock policy is still optimal. 4.6 shows such theoretical standard deviation. By comparing values of $\hat{\sigma}_N \cdot (1 - \gamma)$ in 4.5 and $\sigma \cdot (1 - \gamma) / \sqrt{N}$ for each (γ, N) for the inventory problem, our numerical results suggest that such theoretical standard deviations are aligned with the

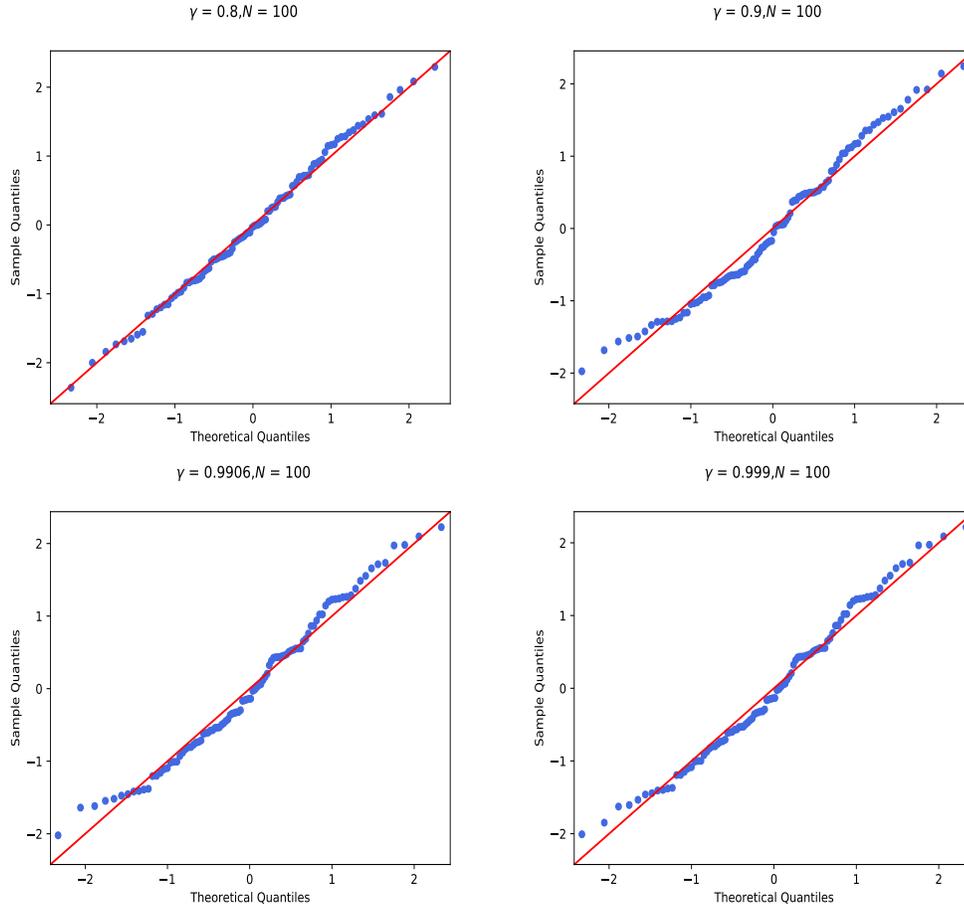


Figure 4.1: Normal probability plot (Q-Q plot) for the risk neutral hydro-thermal problem

sampled ones by comparing $\hat{\sigma}_N \cdot (1 - \gamma)$ and $\sigma \cdot (1 - \gamma) / \sqrt{N}$ for each (γ, N) . We come to the same conclusion as in 4.5.2 that the standard deviations of the optimal value function with discount γ are almost proportional to the factor $(1 - \gamma)^{-1}$.

Table 4.4: Risk averse case: convergence of solving SAA problems.

Inventory problem				
N	γ	LB	CI	Gap(%)
10	0.8	67.599	[67.58,67.62]	0.023
	0.9	158.995	[158.96,159.02]	0.014
	0.9906	1938.91	-	-
	0.999	18324.308	-	-
50	0.8	68.429	[68.41, 68.46]	0.045
	0.9	160.843	[160.81,160.89]	0.029
	0.9906	1962.44	-	-
	0.999	18529.947	-	-
100	0.8	68.493	[68.47,68.51]	0.029
	0.9	160.993	[160.95, 161.03]	0.023
	0.9906	1950.32	-	-
	0.999	18545.915	-	-
Hydro-thermal planning problem				
N	γ	LB ($\cdot 10^6$)	CI ($\cdot 10^6$)	Gap(%)
50	0.8	1.2259	[1.2268,1.2269]	0.08
	0.9	2.452	[2.453,2.454]	0.1
	0.9906	26.0902	-	-
	0.999	243.6119	-	-
100	0.8	1.226	[1.2272,1.2273]	0.11
	0.9	2.452	[2.454,2.455]	0.1
	0.9906	26.0926	-	-
	0.999	243.6346	-	-

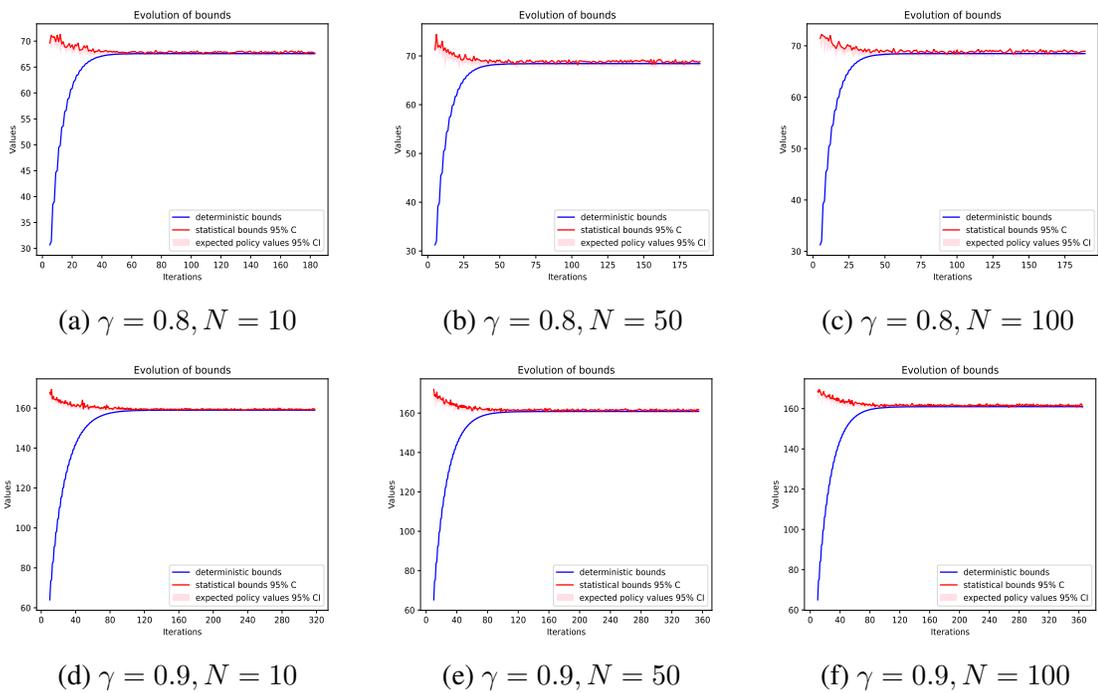


Figure 4.2: Risk averse case: bounds evolution of the SAA problems of the inventory problem.

Table 4.5: Risk averse case: sample standard deviations of optimal values of $M = 100$ SAA problems.

Inventory problem			
N	γ	$\hat{\sigma}_N$	$\hat{\sigma}_N \cdot (1 - \gamma)$
10	0.8	0.512	0.1024
	0.9	1.024	0.1024
	0.9906	10.894	0.1024
	0.999	101.721	0.1017
50	0.8	0.212	0.0424
	0.9	0.424	0.0424
	0.9906	4.508	0.0424
	0.999	42.089	0.0421
100	0.8	0.1354	0.02707
	0.9	0.2707	0.02707
	0.9906	2.8802	0.02707
	0.999	26.8921	0.0269
Hydro-thermal planning problem			
N	γ	$\hat{\sigma}_N$	$\hat{\sigma}_N \cdot (1 - \gamma)$
50	0.8	74.856	14.971
	0.9	130.065	13.007
	0.9906	1664.465	15.646
	0.999	15542.136	15.542
100	0.8	50.097	10.019
	0.9	106.75	10.675
	0.9906	1145.777	10.77
	0.999	10698.883	10.7

Table 4.6: Risk averse case: theoretical standard deviation of the optimal value function for the inventory problem.

γ	σ	$\sigma \cdot (1 - \gamma)$	$\sigma \cdot (1 - \gamma) / \sqrt{10}$	$\sigma \cdot (1 - \gamma) / \sqrt{50}$	$\sigma \cdot (1 - \gamma) / \sqrt{100}$
0.8	1.399	0.279	0.09	0.04	0.028
0.9	2.811	0.281	0.09	0.04	0.028
0.9906	36.122	0.339	0.107	0.048	0.034
0.999	340.041	0.34	0.107	0.048	0.034

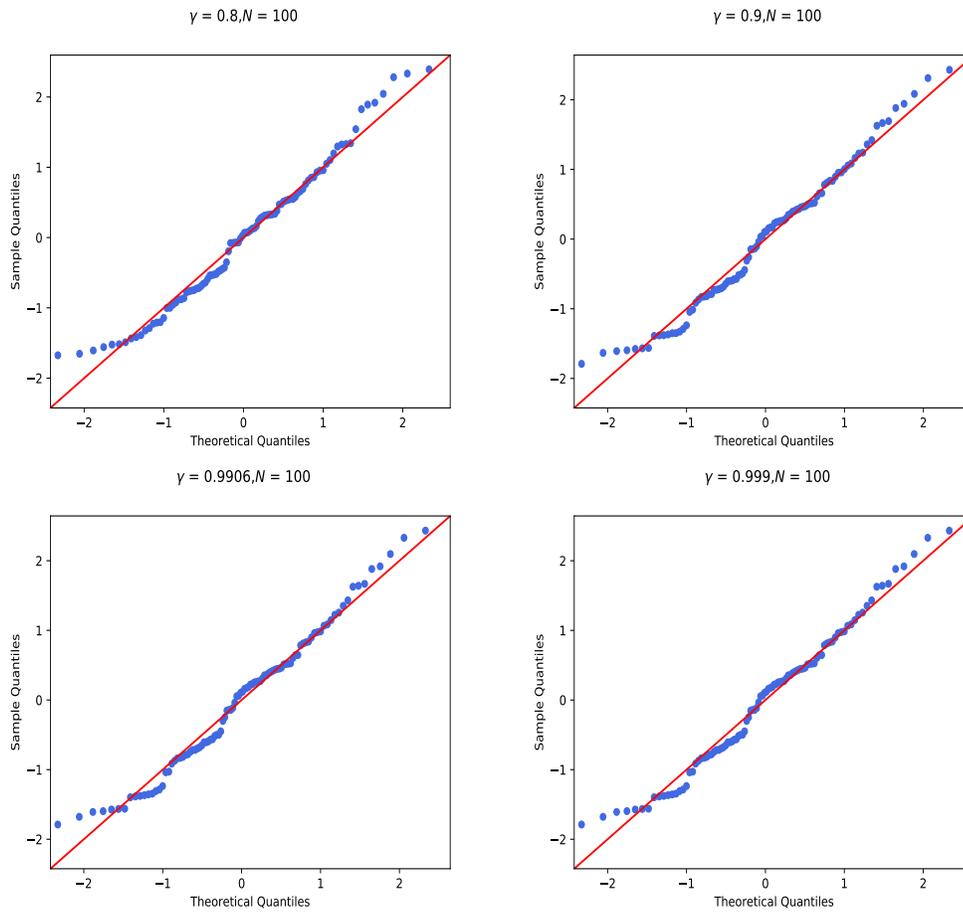


Figure 4.3: Normal probability plot (Q-Q plot) for the risk averse hydro-thermal problem

CHAPTER 5
PROJECTION-FREE METHODS FOR CONVEX FUNCTIONAL
CONSTRAINED OPTIMIZATION

5.1 Overview

In previous chapters, we have seen both theoretical and computational developments around the risk-neutral multistage stochastic linear programs. Henceforth, we will focus on discussion of a class of risk averse convex problems under static-stage setting. In this chapter, we primarily consider a class of convex functional constrained problems with requirement of sparsity in solutions.

Indeed, making decisions from the point of view of risk aversion arises widely in many important applications, such as financial engineering [96, 97, 98, 99], radiation therapy treatment planning [100, 101], supply chain management [102] and power system operations [103]. Risk averse optimization provides a framework for managing fluctuations of specific realizations of the underlying random process, which is critically important especially when tail-probability event relates to the failure or catastrophic disruption of the system being optimized. For example, consider a portfolio selection problem of maximizing the expected return, the optimal strategy suggests directing all investment to the asset with the highest expected return, which may result in losing all or a large amount of the invested principal when the *realized* return of the asset is very low. Furthermore, in many applications of risk averse optimization, solution sparsity is desirable. This happens, for example, in portfolio selection when the number of selected assets is capped and in assortment planning when number of items in the assortment is limited. In addition, sparse solutions are often easier to store and actuate. For instance, in signal processing, sparse approximate solutions are sought after as they can be processed, stored and transmitted in

an efficient fashion. As such, sparse optimization finds rich applications in compressed sensing [104, 105, 106, 107], sparse learning [108, 109] and matrix completion [110, 111].

In risk averse optimization, risk aversion is often manifested by risk measures such as Value-at-Risk (VaR) [94] and Conditional Value-at-Risk (CVaR) [112, 97]. To be specific, given a probability level $\alpha \in (0, 1)$, VaR is defined as the left-side α -quantile of a random variable while CVaR represents the expected value of the α -quantile distribution and is a convex approximation of (VaR). Other widely adopted risk measures include entropic, mean-variance and mean-upper-semideviation risk measure (see Section 6, [6]), to name a few. In some applications (e.g. distributionally robust optimization), risk is also coined by probabilistic form (e.g. chance constraints). Optimizing over the risk measure, either as a constraint or an objective, is commonly used to construct risk averse policies. While convex risk measures result in tractable formulations, nonconvex risk measures such as Value-at-Risk or chance constraints are more appropriate to model and control the risk in some situation.

In sparse optimization, a sparse formulation often aims to find an approximate minimizer (maximizer) that follows the cardinality constraint modeled by ℓ_0 -norm, i.e. the number of nonzeros within the solution is less than a given level. As a convex surrogate of the ℓ_0 -norm, ℓ_1 -norm is also shown to promote solution sparsity. In many scenarios when solution structure can not be attained by the simple sparsity formulation as mentioned above, group sparsity (e.g., sum of a group of ℓ_p norm, $p > 0$) is used to select or deselect the elements in the decision vector at the group level. On top of that, the nuclear norm $\|X\|_*$ (sum of singular values) is often exerted to induce low rank structure, such as in matrix completion. Similar to risk aversion, these aforementioned sparsity requirements can be incorporated either as a regularization term in the objective or a constraint in defining the feasible set.

Risk averse optimization and sparse optimization have been studied separately in most existing literature. These requirements are sometimes conflicting, for example, a diversified

selection of portfolio can reduce the risk but may lead to the violation of the cardinality constraint on the number of assets. Therefore, joint consideration of risk aversion and sparsity appears to be very important in a wide range of applications, e.g., cardinality-constrained assortment planning, cardinality-constrained portfolio selection, power grid optimization and radiation therapy planning. This motivate us to consider a class of functional constrained optimization problems that can be used to model jointly sparsity requirement and risk aversion.

As mentioned earlier, one notable example that is carried out with risk aversion and solution sparsity is portfolio selection with cardinality requirement. A set of risk efficient portfolios constructed from *all* available assets (see e.g., [113]), however, raised the question of whether such an ideal policy is attainable. Indeed, due to various kinds of market friction such as transaction costs, taxes, regulations and asset indivisibility, common practice is to invest on a limited number of assets in a more realistic setting. To this end, cardinality requirement is imposed on the portfolio selection model with the goal to minimize the risk induced by a loss function $\Psi(\cdot)$. One such formulation is give by

$$\begin{aligned}
 & \min \text{CVaR} [\Psi(x)] \\
 & \text{s.t. } \psi(x) \leq c, \\
 & \quad x \in X,
 \end{aligned} \tag{5.1}$$

where $\psi(\cdot)$ is a certain convex surrogate of cardinality constraint and c is the desired number of selected assets. Moreover, the problem in (5.1) can also be formulated as a *nonconvex* problem with convex constraint, by replacing CVaR with VaR in the objective. Alternatively, the cardinality requirement can be modeled directly by the ℓ_0 -norm and participates in the objective function. To meet the cardinality constraint while minimizing risk is a long-standing challenge in the area. Models and methodologies for cardinality constrained portfolio selection optimization have been developed in [114, 115, 116, 117, 118]. How-

ever, these integer programming oriented approaches are computationally inefficient when dealing with large-scale problems, although they may return exact solutions for smaller problems.

Another important application of risk averse sparse optimization can be found in intensity modulated radiation therapy (IMRT) treatment plan in the area of healthcare analytics. This problem can be cast as a jointly sparse and risk averse optimization. In particular, the objective function of the optimization problem is formulated as a VaR, which represents a set of clinical criteria to avoid overdose (resp. underdose) to healthy (resp. tumor) tissues. In addition to a simplex constraint to induce a smaller number of apertures, it consists of a functional constraint, namely, a group sparsity constraint to enforce sparse angle/aperture selection in order to reduce the operation time and the radiation exposure to the patient (see e.g., [100, 101, 31] for more details of the problem description). Due to the huge dimensionality of the decision variable (e.g., the number of apertures), existing approaches suggest to approximate the risk averse requirement by some convex surrogate functions (e.g., quadratic penalty or CVaR). However, these methods rarely return a solution that satisfies all clinical criteria, and often require a lot of fine-tuning of problem formulation (e.g., the penalty parameters). This motivates us to model the clinical criteria from a probabilistic perspective by employing the VaR measure as it is closer to the original clinical criteria (in terms of mathematical formula and interpretation) and to develop efficient algorithms to deal with such nonconvex model.

In this chapter, we propose a novel projection-free method, referred to as Level Conditional Gradient (LCG) method, for solving convex functional constrained optimization. Different from the constraint-extrapolated conditional gradient type methods (CoexCG and CoexDurCG) developed in [31], LCG, as a primal method, does not assume the existence of an optimal dual solution, thus improving the convergence rate of CoexCG/CoexDurCG by eliminating the dependence on the magnitude of the optimal dual solution. Similar to existing level-set methods, LCG uses an approximate Newton method to solve a root-

finding problem. In each approximate Newton update, LCG calls a conditional gradient oracle (CGO) to solve a saddle point subproblem. The CGO developed herein employs easily computable lower and upper bounds on these saddle point problems. We establish the iteration complexity of the CGO for solving a general class of saddle point optimization. Using these results, we show that the overall iteration complexity of the proposed LCG method is $\mathcal{O}\left(\frac{1}{\epsilon^2} \log\left(\frac{1}{\epsilon}\right)\right)$ for finding an ϵ -optimal and ϵ -feasible solution of problem (5.2). To the best of our knowledge, LCG is the first primal conditional gradient method for solving convex functional constrained optimization. For the subsequently developed nonconvex algorithms in this paper, LCG can also serve as a subroutine or provide high-quality starting points that expedites the solution process.

The rest of the chapter is organized as follow. We first describe a class of convex functional constraints problems and introduce the Level Conditional Gradient (LCG) method in Section 5.2. Then in Section 5.3, we provide the design and analysis of the outer loop of LCG. Next in Section 5.4, we present the Conditional Gradient Oracle (CGO), as an inner oracle of LCG, or as a general oracle for convex saddle point problem, for solving both the smooth and nonsmooth settings. Finally in Section 5.5, we summarize the overall iteration complexity of the proposed LCG method. For modeling and numerical results of the portfolio selection and IMRT planning problem, we will show them in 6 which covers the methodology for the nonconvex problems, in order to achieve a more complete illustration and comparison among the proposed algorithms for solving either convex and nonconvex functional constrained problems.

The following notations will be used throughout the paper.

- Without specific mention, $\|\cdot\|$ denotes arbitrary norm (not necessarily associated with the inner product) in the Euclidean space and $\|\cdot\|_*$ denotes its conjugate.
- For a closed convex set $X \subset \mathbb{R}^n$, the set $N_X(x)$ denotes the normal cone at $x \in X$ and $N_X(x) := \{g \in \mathbb{R}^n \mid \forall z \in X : g^\top(z - x) \leq 0\}$.

- A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is L_f -smooth if $\|\nabla f(x_1) - \nabla f(x_2)\|_* \leq L_f \|x_1 - x_2\|$, $\forall x_1, x_2 \in X$.
- A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is M_f -Lipschitz continuous if $|f(x_1) - f(x_2)| \leq M_f \|x_1 - x_2\|$, $\forall x_1, x_2 \in X$.
- Suppose x^* is an optimal solution of (5.2). \bar{x} is an ϵ -optimal and ϵ -feasible solution (or ϵ -solution) of (5.2) if $\bar{x} \in X$, $f(\bar{x}) - f(x^*) \leq \epsilon$ and $\|[h(\bar{x})]_+\|_\infty \leq \epsilon$.

5.2 Level Conditional Gradient Method

The main problem of interest in this chapter is given in the form of

$$\begin{aligned}
 f^* &:= \min f(x) \\
 \text{s.t. } &h_i(x) \leq 0, \quad i = 1, \dots, m, \\
 &x \in X,
 \end{aligned} \tag{5.2}$$

where $f : X \rightarrow \mathbb{R}$ is proper lower semicontinuous function (not necessarily convex), $h := (h_1; \dots; h_m)$, $h_i : X \rightarrow \mathbb{R}, i = 1, \dots, m$ are proper lower semicontinuous and convex functions, $X \subseteq \mathbb{R}^n$ is a nonempty compact convex set. We call problem (5.2) either convex or nonconvex functional constrained optimization depending on whether f is convex or not. For the convex case, the objective function f is not necessarily differentiable. On the other hand, we assume f to be a differentiable function with Lipschitz continuous gradients for the nonconvex setting. The functional constrained problem in (5.2) can be used not only for the joint optimization of sparsity and risk aversion, but also for other potential applications that require the trade-off of different requirements.

It is well-known that problem (5.2) can be reduced to a root finding problem. For a

given level estimate $l \in \mathbb{R}$, let us define

$$\begin{aligned}\phi(l) &:= \min_{x \in X} \max \{f(x) - l, h_1(x), \dots, h_m(x)\} \\ &= \min_{x \in X} \max_{(\gamma, z) \in Z} \gamma[f(x) - l] + \sum_{i=1}^m z_i h_i(x).\end{aligned}\tag{5.3}$$

Here $Z := \{(\gamma, z) \in \mathbb{R}^{m+1} : \gamma + \sum_{i=1}^m z_i = 1, \gamma, z_i \geq 0\}$ denotes the standard simplex.

We can easily verify that: (a) $\phi(l)$ is monotonically non-increasing and convex w.r.t. l ; (b) $\phi(f^*) = 0$; (c) $\phi(l) \geq 0$ for any $l \leq f^*$ and $\phi(l) \leq 0$ for any $l \geq f^*$. Therefore, problem (5.2) is equivalent to finding the root of $\phi(l) = 0$.

We propose to solve (5.2) by LCG (see Algorithm 3), which consists of an *outer loop* that updates the level estimate l (i.e., the estimation of f^*), and an *inner loop* that calls a specialized conditional gradient oracle (CGO) to solve the saddle point problem in (5.3) given a level estimate l .

5.3 Outer Loop of LCG

The basic idea of the LCG method is to apply an approximate Newton's method to solve $\phi(l) = 0$. Assume for the moment that problem (5.3) can be solved exactly for a given l ($l = l_k$). Then one can compute the function value $\phi(l_k)$, a subgradient $\phi'(l_k)$. Solving the following linear equation

$$\phi(l_k) + \phi'(l_k)(l - l_k) = 0$$

gives us an updated iterate l_{k+1} as

$$l_{k+1} = l_k - \frac{\phi(l_k)}{\phi'(l_k)}.$$

Since $\phi(l_k)$ cannot be computed exactly, we suggest to use a computable lower bound and an approximate subgradient in place of $\phi(l_k)$ and $\phi'(l_k)$ in the above equation, respectively. Started with an initial level estimate $l_1 \leq f^*$, we call CGO to compute a lower bound

L_k , an upper bound U_k of $\phi(l_k)$ and an approximate pair of solutions $(x_k; (\gamma_k, z_k)) \in X \times Z$ of problem(5.3) at the k -th iteration (see Algorithm 3). A gap defined by these bounds (i.e. $U_k - L_k$) indicates how accurately problem (5.3) (with $l = l_k$) is solved. Whenever the upper bound $U_k \leq \epsilon$, LCG terminates since an approximate root of $\phi(l) = 0$ has been found due to $\phi(l_k) \leq U_k \leq \epsilon$ and $\phi(l_k) \geq 0$. Otherwise, the algorithm updates the level estimate l_k . More specifically, we define the following linear function as a lower approximation of $\phi(l), \forall l \in \mathbb{R}$:

$$\mathcal{L}_k(l) := L_k - \gamma_k(l - l_k). \quad (5.4)$$

Intuitively, $\mathcal{L}_k(l)$ underestimates $\phi(l)$ since $-\gamma_k$ and L_k respectively serve as an approximate subgradient and a lower bound for $\phi(\cdot)$ at $l = l_k$. To perform the approximate Newton's step as mentioned earlier, we solve $\mathcal{L}_k(l) = 0$ and obtain the following update of the level estimate

$$l_{k+1} = l_k + \frac{1}{\gamma_k} L_k. \quad (5.5)$$

We note that the LCG method provides a general framework for solving the root finding problem in (5.3) and is not restricted to a particular inner oracle, as long as the output (γ_k, L_k, U_k) of the inner oracle (e.g., CGO) satisfies the following conditions:

$$\gamma_k > 0, \quad (5.6)$$

$$L_k \leq \phi(l_k) \leq U_k, \quad (5.7)$$

$$\mathcal{L}_k(l) \leq \phi(l), \forall l. \quad (5.8)$$

The following lemma states an important property of the sequence of the level estimates $(l_k)_{k \geq 1}$ generated in the outer loop of the algorithm. Such property will be used in establishing the number of outer loops required by the LCG method.

Lemma 5.3.1. *At iteration k , if Algorithm 3 does not terminate, then $L_k > 0$. Moreover,*

Algorithm 3 Level Conditional Gradient Method (LCG)

- 1: **Inputs:** $\epsilon > 0, \mu \in (\frac{1}{2}, 1)$.
 - 2: **Initialization:** $x_0 \in X, l_1 = \{\min f(x_0) + \langle \nabla f(x_0), x - x_0 \rangle : x \in X\}$.
 - 3: **for** $k = 1, 2, \dots$ **do**
 - 4: Call CGO with input l_k and obtain approximate solutions $(x_k; (\gamma_k, z_k)) \in X \times Z$, lower bound L_k , upper bound U_k such that $U_k - L_k \leq (1 - \mu)\epsilon$.
 - 5: **if** $U_k \leq \epsilon$ **then**
 - 6: Terminate and return x_k .
 - 7: **end if**
 - 8: $l_{k+1} = l_k + \frac{1}{\gamma_k} L_k$.
 - 9: **end for**
-

the sequence of the level estimates satisfies $l_1 < \dots < l_k < l_{k+1} < \dots \leq f^*, k \geq 1$.

Consequently, $\phi(l_{k+1}) \geq \phi(l_k) \geq \dots \geq \phi(f^*) = 0$.

Proof. We first show that at iteration k , if the algorithm does not terminate, then $L_k > 0$. Indeed, if, on the opposite, $L_k \leq 0$, since CGO stops at $U_k - L_k \leq (1 - \mu)\epsilon$, then $U_k \leq (1 - \mu)\epsilon + L_k \leq (1 - \mu)\epsilon \leq \epsilon$, leading to the termination of the algorithm. Therefore, together by the requirement that $\gamma_k > 0$ returned by CGO, at each update of l , we have $l_{k+1} - l_k \geq \frac{1}{\gamma_k} L_k > 0$. In addition, noting that $\mathcal{L}_k(l_{k+1}) = 0$ by the origin of l_{k+1} in (5.5), and that $\phi(l_{k+1}) \geq \mathcal{L}_k(l_{k+1})$ since \mathcal{L} underestimates ϕ , we have $\phi(l_{k+1}) \geq \mathcal{L}_k(l_{k+1}) = 0 = \phi(f^*)$, which, in view of the fact that ϕ is nonincreasing, implies that $l_{k+1} \leq f^*, k \geq 1$. By the definition of l_1 , we have $l_1 \leq f^*$. Finally by the monotonicity non-increasing property of ϕ and $\phi(f^*)$, we have $\phi(l_{k+1}) \geq \phi(l_k) \geq \dots \geq \phi(f^*) = 0$. \square

In the theorem below, we establish the iteration complexity of reaching “ $U_k \leq \epsilon$ ”, which is essentially the outer loop iteration complexity of solving (5.2) by Algorithm 3.

Theorem 5.3.1. *For all $k \geq 1$, we have*

$$U_k \leq (f^* - l_1) \frac{1}{\mu} \left(\frac{1}{2\mu} \right)^k, \quad (5.9)$$

where $\mu \in (\frac{1}{2}, 1)$, l_1 is the initial estimate of the optimal value of (5.2) such that $l_1 \leq f^*$. Moreover, given precision ϵ , at the termination of LCG when $U_k \leq \epsilon$, the algorithm yields

an ϵ -optimal and ϵ -feasible solution x_k of problem (5.2).

Proof. By the linearity of $\mathcal{L}_k(\cdot)$ and the relation that $l_{k-1} < l_k < l_{k+1}$ according to Lemma 5.3.1, we have

$$\frac{\mathcal{L}_k(l_{k-1}) - \mathcal{L}_k(l_k)}{l_k - l_{k-1}} = \frac{\mathcal{L}_k(l_k) - \mathcal{L}_k(l_{k+1})}{l_{k+1} - l_k},$$

which together with the fact $\mathcal{L}_k(l_{k+1}) = 0$ and the simple relation $a + b \geq 2\sqrt{ab}$, $a, b \in \mathbb{R}^+$ imply that

$$\begin{aligned} (l_{k+1} - l_k)\mathcal{L}_k(l_{k-1}) &\geq (l_{k+1} - l_k + l_k - l_{k-1})\mathcal{L}_k(l_k) \\ &\geq 2\sqrt{l_{k+1} - l_k}\sqrt{l_k - l_{k-1}}\mathcal{L}_k(l_k). \end{aligned}$$

Rearranging the terms, we obtain

$$\frac{\mathcal{L}_k(l_{k-1})}{\sqrt{l_k - l_{k-1}}} \geq \frac{2\mathcal{L}_k(l_k)}{\sqrt{l_{k+1} - l_k}}. \quad (5.10)$$

Observe that $U_k - L_k \leq (1 - \mu)\epsilon$ and $U_k > \epsilon$ when the algorithm does not terminate at iteration k . Therefore, we have $U_k - L_k \leq (1 - \mu)U_k$, and thus $L_k/U_k \geq \mu$. Using this observation and the fact $\mathcal{L}_k(l_k) = L_k$, we obtain $\mathcal{L}_k(l_k) \geq \mu U_k$. Note also $U_{k-1} \geq \phi(l_{k-1}) \geq \mathcal{L}_k(l_{k-1})$. Using this bound and the one in (5.10), we have

$$\frac{U_{k-1}}{\sqrt{l_k - l_{k-1}}} \geq \frac{2\mu U_k}{\sqrt{l_{k+1} - l_k}}. \quad (5.11)$$

Applying the above relation recursively and the facts that $U_1 \leq \frac{1}{\mu}L_1$, $L_1 \leq \phi(l_1)$, $l_{k+1} -$

$l_k \leq f^* - l_1$ yields

$$\begin{aligned}
U_k &\leq \frac{1}{2\mu} \sqrt{\frac{l_{k+1} - l_k}{l_k - l_{k-1}}} U_{k-1} \\
&\leq \left(\frac{1}{2\mu}\right)^{k-1} \sqrt{\frac{l_{k+1} - l_k}{l_2 - l_1}} U_1 \\
&\leq \frac{1}{\mu} \left(\frac{1}{2\mu}\right)^{k-1} \sqrt{(f^* - l_1)\phi(l_1)} \sqrt{\frac{L_1}{l_2 - l_1}}.
\end{aligned} \tag{5.12}$$

Note also it can be easily verified that: $\phi(l) - \phi(l + \delta) \leq \delta$, $l \in \mathbb{R}$, for any $\delta \geq 0$ (same for $\mathcal{L}_1(\cdot)$), which leads to $\phi(l_1) \leq f^* - l_1$ and $\mathcal{L}_1(l_1) \leq l_2 - l_1$ as $\mathcal{L}_1(l_2) = 0$ and $\phi(f^*) = 0$. Consequently, following from the relation in (5.12), we attain

$$U_k \leq \frac{1}{\mu} \left(\frac{1}{2\mu}\right)^{k-1} (f^* - l_1).$$

At the termination of LCG when $U_k \leq \epsilon$, the algorithm yields an ϵ -optimal and ϵ -feasible solution x_k of problem (5.2) since $f(x_k) - f^* \leq f(x_k) - l_k \leq U_k \leq \epsilon$ and $\max_{i=1, \dots, m} \{h_i(x_k)\} \leq U_k \leq \epsilon$. \square

5.4 Conditional Gradient Oracle

In this section, we introduce the Conditional Gradient Oracle (CGO) for solving the following saddle point problem

$$\bar{\phi} := \min_{x \in \bar{X}} \max_{z \in \bar{Z}} \bar{f}(x) + \sum_{i=1}^m z_i \bar{h}_i(x). \tag{5.13}$$

Here, $\bar{f} : \bar{X} \rightarrow \mathbb{R}$ and $\bar{h} : \bar{X} \rightarrow \mathbb{R}^{\bar{m}}$ are proper lower semicontinuous convex functions, $\bar{X} \subseteq \mathbb{R}^{\bar{n}}$ is a nonempty compact convex set, and $\bar{Z} \subseteq \mathbb{R}^{\bar{m}}$ is a general compact set. Under these assumptions an optimal pair of solutions $(x^*, z^*) \in \bar{X} \times \bar{Z}$ of problem (5.13) must exist. Clearly, the subproblem in (5.3) can be viewed as a special case of problem (5.13) with $\bar{f} = 0$, $\bar{X} = X$, $\bar{Z} = Z$ and $\bar{h} = (f - l, h)$.

5.4.1 CGO for Smooth Functions

We assume in this subsection that \bar{f} is an $L_{\bar{f}}$ -smooth, $M_{\bar{f}}$ -Lipschitz continuous function and \bar{h}_i is an $L_{\bar{h}_i}$ -smooth, $M_{\bar{h}_i}$ -Lipschitz continuous function, $i = 1, \dots, \bar{m}$.

Let $\nu : \bar{Z} \rightarrow \mathbb{R}$ be a 1-strongly convex and L_ν -smooth distance generating function and define the proximal function at point $z' \in \bar{Z}$ as $V(z', z) := \nu(z) - \nu(z') - \langle \nabla \nu(z'), z - z' \rangle, z \in \bar{Z}$. Further, denote the linear approximation of \bar{f} and \bar{h} at x' as

$$\begin{aligned} \ell_{\bar{f}}(x', x) &:= \bar{f}(x') + \langle \nabla \bar{f}(x'), x - x' \rangle, \\ \ell_{\bar{h}_i}(x', x) &:= \bar{h}_i(x') + \langle \nabla \bar{h}_i(x'), x - x' \rangle, \quad i = 1, \dots, \bar{m}. \end{aligned}$$

The algorithmic scheme of CGO is stated in Algorithm 4. Through step (5.14) - (5.18), it first extrapolates the linear approximation of the convex functions \bar{h} controlled by the weight λ_t , then updates the dual variable r_t based on the extrapolated value \tilde{h}_t and the proximal function V . En route, CGO computes the primal variable p_t by minimizing the a linear function over \bar{X} and determines the solution x_t by taking the convex combination of p_t and x_{t-1} . Then it recursively computes the lower bounding functions $\underline{f}_t(\cdot)$ and $\underline{h}_t(\cdot)$ of $\bar{f}(\cdot)$ and $\bar{h}(\cdot)$, respectively in (5.19) and (5.20), with $\underline{f}_0(\cdot)$ and $\underline{h}_0(\cdot)$ respectively initialized as lower linear approximation of $\bar{f}(\cdot)$ and $\bar{h}(\cdot)$ at the initial point (see Lemma 5.4.1 for a formal proof). Finally in (5.21) and (5.22), CGO generates a lower bound L_t and an upper bound U_t of (5.13) by solving simple linear programs.

It is worth mentioning the relationship between CGO and the CoexCG/CoexDurCG algorithm in [31]. Both CoexCG and CGO share a similar routine of updating the primal and dual variables from the perspective of applying projection-free technique. The main differences of these two algorithms lie in the following several fronts. First, CGO computes the lower and upper bounds of the saddle point problem and terminates when these two bounds are close enough. CGO neither requires the knowledge of the total number of iterations as it is the case for CoexCG, nor does it need to perform additional regularization

Algorithm 4 Conditional Gradient Oracle (CGO)

Parameters: $\lambda_t \geq 0, \tau_t \geq 0, \alpha_t \in [0, 1], \alpha_1 = 1, \epsilon > 0, \mu \in (\frac{1}{2}, 1)$.

Initialization: $x_{-2} = x_{-1} = x_0 \in \bar{X}, p_{-1} = p_0 \in \bar{X}, z_0 = r_0 \in \bar{Z}, \underline{f}_0(x) \leq \bar{f}(x), \underline{h}_0(x) \leq \bar{h}(x)$.

for $t = 1, 2, \dots$ **do**

 Compute z_t, x_t, L_t and U_t according to

$$\tilde{h}_t = \ell_{\bar{h}}(x_{t-2}, p_{t-1}) + \lambda_t[\ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2})], \quad (5.14)$$

$$r_t = \arg \min_{z \in \bar{Z}} \langle -\tilde{h}_t, z \rangle + \tau_t V(r_{t-1}, z), \quad (5.15)$$

$$z_t = (1 - \alpha_t)z_{t-1} + \alpha_t r_t, \quad (5.16)$$

$$p_t = \arg \min_{x \in \bar{X}} \ell_{\bar{f}}(x_{t-1}, x) + \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle, \quad (5.17)$$

$$x_t = (1 - \alpha_t)x_{t-1} + \alpha_t p_t, \quad (5.18)$$

$$\underline{f}_t(x) = (1 - \alpha_t)\underline{f}_{t-1}(x) + \alpha_t \ell_{\bar{f}}(x_{t-1}, x), \quad (5.19)$$

$$\underline{h}_t(x) = (1 - \alpha_t)\underline{h}_{t-1}(x) + \alpha_t \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle, \quad (5.20)$$

$$L_t = \min_{x \in \bar{X}} \underline{f}_t(x) + \underline{h}_t(x), \quad (5.21)$$

$$U_t = \max_{z \in \bar{Z}} \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle. \quad (5.22)$$

if $U_t - L_t \leq (1 - \mu)\epsilon$ **then**

 Terminate and return x_t, z_t, L_t, U_t .

end if

end for

in the dual update, as it is the case for CoexDurCG. Second, CoexCG is designed to solve the functional constrained problem while CGO aims for the saddle point problem. In the special case of solving the subproblem (5.3), the dual space Z that CGO operates on is a simplex so that its convergence rate is not affected by large Lagrangian multipliers, as opposed to CoexCG. Third, as an inner oracle in solving the convex constrained problem, CGO outputs the dual solution z_t , which participates in the update of level estimate l in the outer loop. In CoexCG, the dual variable is created merely as a tool for the convergence analysis.

In the remaining part of this subsection, we discuss the convergence properties of CGO.

The following lemma shows that L_t and $U_t, t = 1, 2, \dots$, are valid lower bounds and upper bounds of (5.13), respectively.

Lemma 5.4.1. *Let $\bar{\phi}, L_t$ and U_t be defined in (5.13), (5.21) and (5.22), respectively. Also*

let z_t be defined in (5.16). Then we have

$$\underline{f}_t(x) \leq \bar{f}(x), \quad (5.23)$$

$$\underline{h}_t(x) \leq \langle \bar{h}(x), z_t \rangle, \quad (5.24)$$

$$L_t \leq \bar{\phi} \leq U_t, \quad (5.25)$$

for any $t \geq 1$.

Proof. The relation $\underline{f}_t(x) \leq \bar{f}(x)$ immediately follows from the initial condition $\underline{f}_0(x) \leq \bar{f}(x)$ and the fact that $\underline{f}_t(x)$ is the convex combinations of two lower bounding functions of $\bar{f}(x)$ by (5.19).

Let r_t be defined in (5.15). Using the relation of $1 - \alpha_t = \frac{\Gamma_t}{\Gamma_{t-1}}$ and dividing both sides of (5.20) by Γ_t , we have

$$\begin{aligned} \frac{1}{\Gamma_t} \underline{h}_t(x) &= \frac{1}{\Gamma_{t-1}} \underline{h}_{t-1}(x) + \frac{\alpha_t}{\Gamma_t} \langle \bar{\ell}_h(x_{t-1}, x), r_t \rangle \\ &\leq \frac{1}{\Gamma_{t-1}} \underline{h}_{t-1}(x) + \frac{\alpha_t}{\Gamma_t} \langle \bar{h}(x), r_t \rangle \\ &= \sum_{j=1}^t \frac{\alpha_j}{\Gamma_j} \langle \bar{h}(x), r_j \rangle, \end{aligned} \quad (5.26)$$

where the second inequality follows from $\bar{\ell}_h(x_t, x) \leq \bar{h}(x)$, the third inequality is due to the recursive deduction. Multiplying both sides of (5.26) with Γ_t , along with the initial condition that $\underline{h}_0(x) \leq \bar{h}(x)$, we conclude that

$$\underline{h}_t(x) \leq \langle \bar{h}(x), \sum_{j=1}^t \theta_j r_j \rangle = \langle \bar{h}(x), z_t \rangle,$$

where $\theta_j := \Gamma_t \frac{\alpha_j}{\Gamma_j}$ with $\sum_{j=1}^t \theta_j = 1$ and $z_t = \sum_{j=1}^t \theta_j r_j$. It then follows $\underline{f}_t(x) + \underline{h}_t(x) \leq \bar{f}(x) + \langle \bar{h}(x), z_t \rangle$, $\forall x \in \bar{X}$. By the definition of L_t in (5.21), we have $L_t \leq \min_{x \in \bar{X}} \max_{z \in \bar{Z}} \bar{f}(x) + \langle \bar{h}(x), z \rangle$, which shows that L_t is a valid lower bound of problem (5.13). Moreover, using

the fact that

$$\forall x \in \bar{X}, \max_{z \in \bar{Z}} \bar{f}(x) + \langle \bar{h}(x), z \rangle \geq \min_{x \in \bar{X}} \max_{z \in \bar{Z}} \bar{f}(x) + \langle \bar{h}(x), z \rangle,$$

and the definition of U_t in (5.22) and $x_{t-1}, p_t \in \bar{X}$, we obtain

$$U_t = \max_{z \in \bar{Z}} \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle \geq \min_{x \in \bar{X}} \max_{z \in \bar{Z}} \bar{f}(x) + \langle \bar{h}(x), z \rangle.$$

□

In view of Lemma 5.4.1, $(\underline{f}_t + \underline{h}_t)(\cdot)$ provides a lower bound for the objective of (5.13), i.e., $(\underline{f}_t + \underline{h}_t)(\cdot) \leq \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle$. This motivates us to define the gap function for problem (5.13) as

$$\bar{Q}_t(w_t, w) := \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle - \underline{f}_t(x) - \underline{h}_t(x), \quad (5.27)$$

where $w_t := (x_t, z_t)$, $w := (x, z)$. Also, by the definition of L_t and U_t , we can easily see that

$$\max_{w \in \bar{X} \times \bar{Z}} \bar{Q}_t(w_t, w) = U_t - L_t.$$

It is worth mentioning here that the gap function in (5.27) is different from those used in the existing literature, given by

$$\tilde{Q}_t(w_t, w) := \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle - \bar{f}(x) - \langle \bar{h}(x), z_t \rangle.$$

As a consequence, these algorithms require the solution of $\min_{x \in \bar{X}} (\bar{f}(x) + \langle \bar{h}(x), z_t \rangle)$ to compute a lower bound on $\bar{\phi}$, which can be computationally expensive unless both \bar{f} and \bar{h} are simple enough (e.g., linear functions). On the other hand, the computation of the lower bound L_t in CGO only requires one call to the linear optimization oracle. In addition, since $\bar{Q}_t(w_t, w) \geq \tilde{Q}_t(w_t, w)$, we obtain stronger convergence guarantees for the developed

algorithm by using $\bar{Q}_t(w_t, w)$ instead of $\tilde{Q}_t(w_t, w)$ as the error measure.

The following proposition establishes the recursion of the gap function (5.27) for CGO, which is an important intermediate step to the proof of the main theorem.

Proposition 5.4.1. *At iteration $t > 1$, we have*

$$\begin{aligned} \bar{Q}_t(w_t, w) &\leq (1 - \alpha_t)\bar{Q}_{t-1}(w_{t-1}, w) + \frac{(L_{\bar{f}} + z^\top L_{\bar{h}})\alpha_t^2}{2}D_{\bar{X}}^2 + \frac{9\alpha_t\lambda_t^2\bar{M}^2D_{\bar{X}}^2}{2\tau_t} \\ &\quad + \alpha_t [\langle \ell_{\bar{h}}(x_{t-1}, p_t) - \ell_{\bar{h}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle] \\ &\quad + \alpha_t [\tau_t V(r_{t-1}, z) - \tau_t V(r_t, z)]. \end{aligned}$$

Proof. Together by the definition of $\bar{Q}_t(w_t, w)$, $\underline{f}_t, \underline{h}_t$ respectively in (5.27), (5.19) and (5.20) and Lemma 5.7.3, for any $w \in \bar{X} \times \bar{Z}$, we have

$$\begin{aligned} \bar{Q}_t(w_t, w) &\leq (1 - \alpha_t)\bar{f}(x_{t-1}) + \alpha_t \ell_{\bar{f}}(x_{t-1}, p_t) + \frac{L_{\bar{f}}\alpha_t^2}{2}\|p_t - x_{t-1}\|^2 \\ &\quad + (1 - \alpha_t)\langle \bar{h}(x_{t-1}), z \rangle + \alpha_t \langle \ell_{\bar{h}}(x_{t-1}, p_t), z \rangle + \frac{z^\top L_{\bar{h}}\alpha_t^2}{2}\|p_t - x_{t-1}\|^2 \\ &\quad - (1 - \alpha_t)\underline{f}_{t-1}(x) - \alpha_t \ell_{\bar{f}}(x_{t-1}, x) - (1 - \alpha_t)\underline{h}_{t-1}(x) - \alpha_t \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle \\ &\leq (1 - \alpha_t) \left[\bar{f}(x_{t-1}) + \langle \bar{h}(x_{t-1}), z \rangle - \underline{f}_{t-1}(x) - \underline{h}_{t-1}(x) \right] + \frac{(L_{\bar{f}} + z^\top L_{\bar{h}})\alpha_t^2}{2}D_{\bar{X}}^2 \\ &\quad + \alpha_t \left[\ell_{\bar{f}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}}(x_{t-1}, x) - \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle \right] \\ &= (1 - \alpha_t)\bar{Q}_{t-1}(w_{t-1}, w) + \frac{(L_{\bar{f}} + z^\top L_{\bar{h}})\alpha_t^2}{2}D_{\bar{X}}^2 \\ &\quad + \alpha_t \left[\ell_{\bar{f}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}}(x_{t-1}, x) - \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle \right]. \end{aligned} \tag{5.28}$$

The primal update (5.17) implies that

$$\ell_{\bar{f}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}}(x_{t-1}, p_t), r_t \rangle \leq \ell_{\bar{f}}(x_{t-1}, x) + \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle, \quad \forall x \in \bar{X}.$$

Rearranging the terms in the above inequality, we have

$$\ell_{\bar{f}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}}(x_{t-1}, x) - \langle \ell_{\bar{h}}(x_{t-1}, x), r_t \rangle \leq \langle \ell_{\bar{h}}(x_{t-1}, p_t), z - r_t \rangle. \quad (5.29)$$

Meanwhile, by Lemma 5.7.1, we obtain

$$\langle \ell_{\bar{h}}(x_{t-1}, p_t), z - r_t \rangle \leq \langle \ell_{\bar{h}}(x_{t-1}, p_t) - \tilde{h}_t, z - r_t \rangle + \tau_t V(r_{t-1}, z) - \tau_t V(r_t, z) - \tau_t V(r_{t-1}, r_t). \quad (5.30)$$

In addition,

$$\begin{aligned} & \langle \ell_{\bar{h}}(x_{t-1}, p_t) - \tilde{h}_t, z - r_t \rangle - \tau_t V(r_{t-1}, r_t) \\ &= \langle \ell_{\bar{h}}(x_{t-1}, p_t) - \ell_{\bar{h}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \\ & \quad + \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), r_t - r_{t-1} \rangle - \tau_t V(r_{t-1}, r_t) \\ & \leq \langle \ell_{\bar{h}}(x_{t-1}, p_t) - \ell_{\bar{h}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \\ & \quad + \frac{9\lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t}, \end{aligned} \quad (5.31)$$

where the first equality follows from the definition of \tilde{h}_t in (5.14) and the last inequality is due to

$$\begin{aligned} & \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), r_t - r_{t-1} \rangle - \tau_t V(r_{t-1}, r_t) \\ & \leq \frac{\lambda_t^2}{2\tau_t} \|\ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2})\|^2 \\ & = \frac{\lambda_t^2}{2\tau_t} \|\bar{h}(x_{t-2}) - \bar{h}(x_{t-3}) + \langle \nabla \bar{h}(x_{t-2}), p_{t-1} - x_{t-2} \rangle + \langle \nabla \bar{h}(x_{t-3}), x_{t-3} - p_{t-2} \rangle\|^2 \\ & \leq \frac{\lambda_t^2}{2\tau_t} \|\bar{h}(x_{t-2}) - \bar{h}(x_{t-3}) + \bar{h}(p_{t-1}) - \bar{h}(x_{t-2}) + \bar{h}(p_{t-2}) - \bar{h}(x_{t-3})\|^2 \\ & \leq \frac{9\lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t}. \end{aligned} \quad (5.32)$$

Plugging (5.29), (5.30) and (5.31) into (5.28), we prove the result. \square

Proposition 5.4.2. *Suppose for $t \geq 2$, parameters $\{\alpha_t\}$, $\{\lambda_t\}$ and $\{\tau_t\}$ in Algorithm 4 satisfy*

$$\alpha_1 = 1, \quad \frac{\lambda_t \alpha_t}{\Gamma_t} = \frac{\alpha_{t-1}}{\Gamma_{t-1}} \text{ and } \frac{\alpha_t \tau_t}{\Gamma_t} \geq \frac{\alpha_{t-1} \tau_{t-1}}{\Gamma_{t-1}}. \quad (5.33)$$

Then for the general saddle point problem (5.13), for $T \geq 1$, we have

$$\bar{Q}_T(w_T, w) \leq \Gamma_T \sum_{t=1}^T \left[\frac{(L_{\bar{f}} + z^\top L_{\bar{h}}) \alpha_t^2}{2\Gamma_t} D_{\bar{X}}^2 + \frac{9\alpha_t \lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t \Gamma_t} \right] + \frac{9\alpha_T \bar{M}^2 D_{\bar{X}}^2}{2\tau_T} + \alpha_T \tau_T \bar{V}, \quad (5.34)$$

$\forall w \in \bar{X} \times \bar{Z}$.

Proof. In view of Lemma 5.7.2 and Proposition 5.4.1 as well as $1 - \alpha_1 = 0$, we have

$$\begin{aligned} \frac{\bar{Q}_T(w_T, w)}{\Gamma_T} &\leq \sum_{t=1}^T \left[\frac{(L_{\bar{f}} + z^\top L_{\bar{h}}) \alpha_t^2}{2\Gamma_t} D_{\bar{X}}^2 + \frac{9\alpha_t \lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t \Gamma_t} \right] \\ &\quad + \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} [\langle \ell_{\bar{h}}(x_{t-1}, p_t) - \ell_{\bar{h}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle] \\ &\quad + \sum_{t=1}^T \frac{\alpha_t \tau_t}{\Gamma_t} [V(r_{t-1}, z) - V(r_t, z)], \forall w \in \bar{X} \times \bar{Z}. \end{aligned}$$

The second equation in (5.33) indicates that summing up the extrapolated linear function values from 1 to T cancels out intermediate terms, such that

$$\begin{aligned} &\sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} [\langle \ell_{\bar{h}}(x_{t-1}, p_t) - \ell_{\bar{h}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \langle \ell_{\bar{h}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle] \\ &= \alpha_T \langle \ell_{\bar{h}}(x_{T-1}, p_T) - \ell_{\bar{h}}(x_{T-2}, p_{T-1}), z - r_T \rangle - \lambda_1 \langle \ell_{\bar{h}}(x_{-1}, p_0) - \ell_{\bar{h}}(x_{-2}, p_{-1}), z - r_0 \rangle. \end{aligned}$$

Besides, the third inequality in (5.33) implies

$$\begin{aligned}
& \sum_{t=1}^T \frac{\alpha_t \tau_t}{\Gamma_t} [V(r_{t-1}, z) - V(r_t, z)] \\
&= \frac{\alpha_1 \tau_1}{\Gamma_1} V(r_0, z) + \sum_{t=2}^T \left(\frac{\alpha_t \tau_t}{\Gamma_t} - \frac{\alpha_{t-1} \tau_{t-1}}{\Gamma_{t-1}} \right) V(r_{t-1}, z) - \frac{\alpha_T \tau_T}{\Gamma_T} V(r_T, z) \\
&\leq \frac{\alpha_1 \tau_1}{\Gamma_1} \bar{V} + \sum_{t=2}^T \left(\frac{\alpha_t \tau_t}{\Gamma_t} - \frac{\alpha_{t-1} \tau_{t-1}}{\Gamma_{t-1}} \right) \bar{V} - \frac{\alpha_T \tau_T}{\Gamma_T} V(r_{T-1}, z) \\
&\leq \frac{\alpha_T \tau_T}{\Gamma_T} \bar{V} - \frac{\alpha_T \tau_T}{\Gamma_T} V(r_T, z).
\end{aligned} \tag{5.35}$$

Using the above relations, together with the initial condition $x_{-2} = x_{-1}$, $p_{-1} = p_0$ which gives $\ell_{\bar{h}}(x_{-1}, p_0) - \ell_{\bar{h}}(x_{-2}, p_{-1}) = 0$ and the relation that

$$\alpha_T \langle \ell_{\bar{h}}(x_{T-1}, p_T) - \ell_{\bar{h}}(x_{T-2}, p_{T-1}), z - r_T \rangle - \alpha_T \tau_T V(r_T, z) \leq \frac{9\bar{M}^2 \alpha_T D_{\bar{X}}^2}{2\tau_T},$$

we conclude that $\forall w \in \bar{X} \times \bar{Z}$,

$$\begin{aligned}
\bar{Q}_T(w_T, w) &\leq \Gamma_T \sum_{t=1}^T \left[\frac{(L_{\bar{f}} + z^\top L_{\bar{h}}) \alpha_t^2}{2\Gamma_t} D_{\bar{X}}^2 + \frac{9\alpha_t \lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t \Gamma_t} \right] \\
&\quad + \alpha_T \langle \ell_{\bar{h}}(x_{T-1}, p_T) - \ell_{\bar{h}}(x_{T-2}, p_{T-1}), z - r_T \rangle - \alpha_T \tau_T V(r_T, z) \\
&\quad - \lambda_1 \langle \ell_{\bar{h}}(x_{-1}, p_0) - \ell_{\bar{h}}(x_{-2}, p_{-1}), z - r_0 \rangle + \alpha_T \tau_T \bar{V} \\
&\leq \Gamma_T \sum_{t=1}^T \left[\frac{(L_{\bar{f}} + z^\top L_{\bar{h}}) \alpha_t^2}{2\Gamma_t} D_{\bar{X}}^2 + \frac{9\alpha_t \lambda_t^2 \bar{M}^2 D_{\bar{X}}^2}{2\tau_t \Gamma_t} \right] + \frac{9\bar{M}^2 \alpha_T D_{\bar{X}}^2}{2\tau_T} + \alpha_T \tau_T \bar{V}.
\end{aligned}$$

□

Theorem 5.4.1 below states the main convergence properties for CGO. We need to use the following quantities for this result:

$$\bar{M} := \left(\sum_{i=1}^{\bar{m}} M_{\bar{h}_i}^2 \right)^{1/2}, \quad D_{\bar{X}} := \max_{x_1, x_2 \in \bar{X}} \|x_1 - x_2\| \quad \text{and} \quad \bar{V} := \max_{z_1, z_2 \in \bar{Z}} V(z_1, z_2).$$

Theorem 5.4.1. *Suppose that the algorithmic parameters in CGO are set to*

$$\alpha_t = \frac{2}{t+1}, \quad \lambda_t = \frac{t-1}{t}, \quad \tau_t = 9\sqrt{t}\bar{M}D_{\bar{X}}, \quad t \geq 1. \quad (5.36)$$

Then for any $t \geq 1$,

$$\bar{Q}_t(w_t, w) \leq \frac{2(L_{\bar{f}} + z^\top L_{\bar{h}})D_{\bar{X}}^2}{t+1} + \frac{\bar{M}D_{\bar{X}}}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right] \quad \forall w \in (\bar{X}, \bar{Z}). \quad (5.37)$$

Proof. It is easy to verify that the identities in (5.36) satisfy the conditions in (5.33). By definition of $\{\Gamma_t\}$ and (5.36), we have $\Gamma_t = \frac{2}{t(t+1)}$ and $\alpha_t/\Gamma_t = t$, so that for any $T \geq 1$

$$\begin{aligned} \Gamma_T \sum_{t=1}^T \frac{\alpha_t^2}{2\Gamma_t} &= \Gamma_T \sum_{t=1}^T \frac{t}{t+1} \leq \frac{2}{T+1}, \\ \Gamma_T \sum_{t=1}^T \frac{9\alpha_t\lambda_t^2}{2\tau_t\Gamma_t} &\leq \frac{2\sqrt{T}}{3(T+1)\bar{M}D_{\bar{X}}}. \end{aligned}$$

Plugging the above relations in (5.34), we obtain

$$\begin{aligned} \bar{Q}_T(w_T, w) &\leq \frac{2(L_{\bar{f}} + z^\top L_{\bar{h}})D_{\bar{X}}^2}{T+1} + \frac{2\sqrt{T}\bar{M}D_{\bar{X}}}{3(T+1)} + \frac{\bar{M}D_{\bar{X}}}{\sqrt{T}(T+1)} + \frac{18\sqrt{T}\bar{M}D_{\bar{X}}\bar{V}}{(T+1)} \\ &= \frac{2(L_{\bar{f}} + z^\top L_{\bar{h}})D_{\bar{X}}^2}{T+1} + \bar{M}D_{\bar{X}} \left[\frac{2\sqrt{T}}{3(T+1)} + \frac{1}{\sqrt{T}(T+1)} + \frac{18\sqrt{T}\bar{V}}{T+1} \right] \\ &\leq \frac{2(L_{\bar{f}} + z^\top L_{\bar{h}})D_{\bar{X}}^2}{T+1} + \frac{\bar{M}D_{\bar{X}}}{\sqrt{T+1}} \left[18\bar{V} + \frac{7}{6} \right], \quad \forall w \in (\bar{X}, \bar{Z}). \end{aligned} \quad (5.38)$$

In this way, we show the conclusion in (5.37). □

5.4.2 CGO for Structured Nonsmooth Functions

In this section, we focus on problem (5.13) where $\bar{f}(\cdot)$ and $\bar{h}_i(\cdot)$, $i = 1, \dots, \bar{m}$ are structured nonsmooth functions represented by the following form (see also [119]):

$$\begin{aligned}\bar{f}(x) &= \max_{y \in Y_0} \{ \langle B_0 x, y \rangle - \hat{f}(y) \}, \\ \bar{h}_i(x) &= \max_{y \in Y_i} \{ \langle B_i x, y \rangle - \hat{h}_i(y) \}, i = 1, \dots, \bar{m},\end{aligned}$$

where Y_i , $i = 0, 1, \dots, \bar{m}$ are closed convex sets, \hat{f} and \hat{h}_i are simple (continuous and differentiable) convex functions, possibly ω_i -strongly convex $i = 0, 1, \dots, \bar{m}$. Let $u_i : Y_i \rightarrow \mathbb{R}$ be a 1-strongly convex distance generating function. Define the proximal function U_i as $U_i(x) := u_i(y) - u_i(y_{u_i}) - \langle \nabla u_i(y_{u_i}), y - y_{u_i} \rangle$, $y \in Y_i$, where $y_{u_i} := \arg \min_{y \in Y_i} u_i(y)$. Further let η_i , $i = 0, \dots, \bar{m}$ be the smoothing parameters that can vary or stay static over iterations.

To generalize CGO to solve problems with structured nonsmooth functions, we need to leverage the Nesterov smoothing scheme [119] to approximate the possibly nonsmooth functions \bar{f} and \bar{h}_i by \bar{f}_{η_0} and \bar{h}_{i,η_i} stated below:

$$\bar{f}_{\eta_0}(x) := \max_{y \in Y_0} \{ \langle B_0 x, y \rangle - \hat{f}(y) - \eta_0 U_0(y) \}, \quad (5.39)$$

$$\bar{h}_{i,\eta_i}(x) := \max_{y \in Y_i} \{ \langle B_i x, y \rangle - \hat{h}_i(y) - \eta_i U_i(y) \}, i = 1, \dots, \bar{m}. \quad (5.40)$$

It can be shown that (see [119]), \bar{f}_{η_0} and \bar{h}_{i,η_i} are differentiable with Lipschitz constants $L_{\bar{f},\eta} := \frac{\|B_0\|^2}{\omega_0 + \eta_0}$ and $L_{\bar{h}_i,\eta} := \frac{\|B_i\|^2}{\omega_i + \eta_i}$. Suppose Y_i , $i = 1, \dots, \bar{m}$ are compact, then \bar{h}_{i,η_i} have bounded gradients such as $\|\nabla \bar{h}_{i,\eta_i}(x)\|_\infty \leq \bar{M}_{B_i, U_i}$, where $\bar{M}_{B_i, U_i} := \|B_i\| (\|y_{u_i}\| + \sqrt{2} D_{U_i})$, $i = 1, \dots, \bar{m}$, $D_{U_i} := \left(\max_{y \in Y_i} U_i(y) \right)^{1/2}$. Moreover, the relation between the original func-

tions and the smoothing counterparts are characterized by

$$\begin{aligned} \bar{f}_{\eta_0}(x) &\leq \bar{f}(x) \leq \bar{f}_{\eta_0}(x) + \eta_0 D_{U_0}^2, \\ \bar{h}_{i,\eta_i}(x) &\leq \bar{h}_i(x) \leq \bar{h}_{i,\eta_i}(x) + \eta_i D_{U_i}^2, \quad i = 1, \dots, \bar{m}. \end{aligned} \quad (5.41)$$

In this part, we focus on the case where the smoothing parameters η_i , $i = 0, 1, \dots, \bar{m}$ are adapted over iterations such as

$$\eta_i^0 \geq \eta_i^1 \geq \dots \geq \eta_i^t, \quad i = 0, 1, \dots, \bar{m}. \quad (5.42)$$

In this case, at each iteration t , the approximations of \bar{f} and \bar{h} are $\bar{f}_{\eta_0^t}$ and $\bar{h}_{\eta_i^t}$. Accordingly, their Lipschitz constants are changed to $L_{\bar{f}}^t \equiv L_{\bar{f},\eta_0^t} := \frac{\|B_0\|^2}{\omega_0 + \eta_0^t}$ and $L_{\bar{h}_i}^t \equiv L_{\bar{h}_i,\eta_i^t} := \frac{\|B_i\|^2}{\omega_i + \eta_i^t}$. Nevertheless, the relation in (5.41) still holds for each $\bar{f}_{\eta_0^t}$ and $\bar{h}_{\eta_i^t}$ at iteration t . Moreover, similar to [31], it can be shown that the sequences $\{\bar{f}_{\eta_0^t}\}_t$ and $\{\bar{h}_{i,\eta_i^t}\}_t$ satisfy:

$$\begin{aligned} \bar{f}_{\eta_0^{t-1}} &\leq \bar{f}_{\eta_0^t} \leq \bar{f}_{\eta_0^{t-1}} + (\eta_0^{t-1} - \eta_0^t) D_{U_0}^2, \\ \bar{h}_{i,\eta_i^{t-1}} &\leq \bar{h}_{i,\eta_i^t} \leq \bar{h}_{i,\eta_i^{t-1}} + (\eta_i^{t-1} - \eta_i^t) D_{U_i}^2, \quad i = 1, \dots, \bar{m}. \end{aligned} \quad (5.43)$$

The algorithm (see Algorithm 5) of solving the general structured nonsmooth problems (with \bar{f} and \bar{h} respectively approximated by $\bar{f}_{\eta_0^t}$ and \bar{h}_{i,η_i^t}) is similar to Algorithm 4, except that the linear approximations of the objective function and constraint are replaced by $\ell_{\bar{f}_{\eta_0^t}}(x', x) := \bar{f}_{\eta_0^t}(x') + \langle \nabla \bar{f}_{\eta_0^t}(x'), x - x' \rangle$ and $\ell_{\bar{h}_{i,\eta_i^t}}(x', x) := \bar{h}_{i,\eta_i^t}(x') + \langle \nabla \bar{h}_{i,\eta_i^t}(x'), x - x' \rangle$, $i = 1, \dots, \bar{m}$, respectively. If the original functions are smooth, then the parameters η_i^t simply reduces to constant zero.

Algorithm 5 CGO for Structured Nonsmooth Problems

The algorithm is modified from Algorithm 4 by replacing step (5.14) with

$$\tilde{h}_t = \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) + \lambda_t [\ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2})], \quad (5.44)$$

and primal update (5.17) with

$$p_t = \arg \min_{x \in \bar{X}} \ell_{\bar{f}_{\eta^t}}(x_{t-1}, x) + \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, x), r_t \rangle, \quad (5.45)$$

and update of lower bound functionals (5.19) and (5.20) with

$$\underline{f}_t(x) = (1 - \alpha_t) \underline{f}_{t-1}(x) + \alpha_t \ell_{\bar{f}_{\eta^t}}(x_{t-1}, x), \quad (5.46)$$

$$\underline{h}_t(x) = (1 - \alpha_t) \underline{h}_{t-1}(x) + \alpha_t \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, x), r_t \rangle. \quad (5.47)$$

For the original nonsmooth problem, the gap function is defined by

$$\bar{Q}_t(w_t, w) := \bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle - \bar{f}(x) - \langle \bar{h}(x), z_t \rangle, \forall w \in \bar{X} \times \bar{Z}.$$

In view of Lemma 5.4.1, we can show that $(\underline{f}_t + \underline{h}_t)(\cdot)$ computed from (5.46) and (5.47) is a lower bounding function of both the original objective $\bar{f}(x_t) + \langle \bar{h}(x_t), z \rangle$ and the smoothing approximation $\bar{f}_{\eta_0^t}(x_t) + \langle \bar{h}_{\eta_0^t}(x_t), z \rangle$. The gap function of the approximated problem is hereby defined as

$$\bar{Q}_t^\eta(w_t, w) := \bar{f}_{\eta_0^t}(x_t) + \langle \bar{h}_{\eta_0^t}(x_t), z \rangle - \underline{f}_t(x) - \underline{h}_t(x), \forall w \in \bar{X} \times \bar{Z}, \quad (5.48)$$

for $w_t := (x_t, z_t)$. Following from (5.41), it is easy to see that

$$\bar{Q}_t(w_t, w) \leq \bar{Q}_t^\eta(w_t, w) + \eta_0^t D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^t D_{U_i}^2, \quad k \geq 1, \forall w \in \bar{X} \times \bar{Z}. \quad (5.49)$$

We will show in Theorem 5.4.2 the iteration complexity of solving the nonsmooth problem is bounded by $\mathcal{O}(1/\epsilon^2)$. To this end, we start by identifying an important recursion relation of the gap function (5.48) in Proposition 5.4.3. Then in Proposition 5.4.4, we state the convergence property under general parameter setup. In the subsequent analysis, we use the following notations: $\bar{M}_{B,U} := \sqrt{\sum_{i=1}^{\bar{m}} M_{B_i, U_i}^2}$, $L_{\bar{h}}^t := (L_{\bar{h}_1, \eta^t}, \dots, L_{\bar{h}_{\bar{m}}, \eta^t})$ and $\bar{h}_{\eta^t} := (\bar{h}_{1, \eta_1^t}, \dots, \bar{h}_{\bar{m}, \eta_{\bar{m}}^t})$.

Proposition 5.4.3. *At iteration $t > 1$, we have*

$$\begin{aligned} \bar{Q}_t^\eta(w_t, w) &\leq (1 - \alpha_t) \bar{Q}_{t-1}^\eta(w_{t-1}, w) + \frac{(L_{\bar{f}}^t + z^\top L_{\bar{h}}^t) \alpha_t^2 D_{\bar{X}}^2}{2} \\ &\quad + (1 - \alpha_t) \left[(\eta_0^{t-1} - \eta_0^t) D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i (\eta_i^{t-1} - \eta_i^t) D_{U_i}^2 \right] \\ &\quad + \frac{6\alpha_t \lambda_t^2 M_{B,U}^2 D_{\bar{X}}^2}{\tau_t} + \frac{3\alpha_t \lambda_t^2}{\tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \\ &\quad + \alpha_t \left[\langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}), z - r_t \rangle \right. \\ &\quad \quad - \lambda_t \langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \\ &\quad \quad \left. + \tau_t V(r_{t-1}, z) - \tau_t V(r_t, z) \right], \quad \forall w \in \bar{X} \times \bar{Z}. \end{aligned} \quad (5.50)$$

Proof. Using the relation in (5.43), the updates in (5.46), (5.47) and applying Lemma 5.7.3

on $\bar{f}_{\eta_0^t}, \bar{h}_{i, \eta_i^t}$, we have

$$\begin{aligned}
\bar{Q}_t^\eta(w_t, w) &\leq (1 - \alpha_t) \left[\bar{f}_{\eta_0^t}(x_{t-1}) + \langle \bar{h}_{\eta^t}(x_{t-1}), z \rangle - \underline{f}_{t-1}(x) - \underline{h}_{t-1}(x) \right] + \frac{\left(L_{\bar{f}}^t + z^\top L_{\bar{h}} \right) \alpha_t^2 D_X^2}{2} \\
&\quad + \alpha_t \left[\ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, x) - \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, x), r_t \rangle \right] \\
&\leq (1 - \alpha_t) \left[\bar{Q}_{t-1}^\eta(w_{t-1}, w) + (\eta_0^{t-1} - \eta_0^t) D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i (\eta_i^{t-1} - \eta_i^t) D_{U_i}^2 \right] \\
&\quad + \frac{\left(L_{\bar{f}}^t + z^\top L_{\bar{h}} \right) \alpha_t^2 D_X^2}{2} \\
&\quad + \alpha_t \left[\ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, x) - \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, x), r_t \rangle \right].
\end{aligned} \tag{5.51}$$

Note that the last line in (5.51) has the following relation:

$$\begin{aligned}
&\ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, p_t) + \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t), z \rangle - \ell_{\bar{f}_{\eta_0^t}}(x_{t-1}, x) - \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, x), r_t \rangle \\
&\leq \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t), z - r_t \rangle \\
&\leq \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \tilde{h}_t, z - r_t \rangle + \tau_t V(r_{t-1}, z) - \tau_t V(r_t, z) - \tau_t V(r_{t-1}, r_t) \\
&= \langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}), z - r_t \rangle - \lambda_t \left(\langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \right) \\
&\quad + \tau_t V(r_{t-1}, z) - \tau_t V(r_t, z) \\
&\quad - \lambda_t \left(\langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), r_{t-1} - r_t \rangle \right) - \tau_t V(r_{t-1}, r_t),
\end{aligned} \tag{5.52}$$

where the first inequality follows from the primal update in (5.45), the second inequality is the result of the dual update and Lemma 5.7.1 and the last equality is by the definition of \tilde{h}_t in (5.44).

Moreover, the last line in (5.52) can be bounded by

$$\begin{aligned}
& -\lambda_t \left(\langle \ell_{\bar{h}_{\eta^{t-1}}}^-(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}^-(x_{t-3}, p_{t-2}), r_{t-1} - r_t \rangle \right) - \tau_t V(r_{t-1}, r_t) \\
& \leq \frac{\lambda_t^2}{2\tau_t} \sum_{i=1}^{\bar{m}} \left(\ell_{\bar{h}_{i, \eta_i^{t-1}}}^-(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{i, \eta_i^{t-2}}}^-(x_{t-3}, p_{t-2}) \right)^2 \\
& = \frac{\lambda_t^2}{2\tau_t} \sum_{i=1}^{\bar{m}} \left(\bar{h}_{i, \eta_i^{t-1}}(x_{t-2}) - \bar{h}_{i, \eta_i^{t-2}}(x_{t-3}) + \langle \nabla \bar{h}_{i, \eta_i^{t-1}}(x_{t-2}), p_{t-1} - x_{t-2} \rangle \right. \\
& \quad \left. + \langle \nabla \bar{h}_{i, \eta_i^{t-2}}(x_{t-3}), x_{t-3} - p_{t-2} \rangle \right)^2 \\
& \leq \frac{3\lambda_t^2}{2\tau_t} \sum_{i=1}^{\bar{m}} \left[\left(\bar{h}_{i, \eta_i^{t-1}}(x_{t-2}) - \bar{h}_{i, \eta_i^{t-2}}(x_{t-3}) \right)^2 + 2\bar{M}_{B_i, U_i}^2 D_{\bar{X}}^2 \right] \\
& \leq \frac{3\lambda_t^2}{2\tau_t} \sum_{i=1}^{\bar{m}} \left[\left(\bar{h}_{i, \eta_i^{t-2}}(x_{t-2}) - \bar{h}_{i, \eta_i^{t-2}}(x_{t-3}) + (\eta_i^{t-2} - \eta_i^{t-1}) D_{U_i}^2 \right)^2 + 2\bar{M}_{B_i, U_i}^2 D_{\bar{X}}^2 \right] \\
& \leq \frac{3\lambda_t^2}{2\tau_t} \sum_{i=1}^{\bar{m}} \left[2 \left(\langle \nabla \bar{h}_{i, \eta_i^{t-2}}(x_{t-2}), x_{t-2} - x_{t-3} \rangle \right)^2 + 2(\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 + 2\bar{M}_{B_i, U_i}^2 D_{\bar{X}}^2 \right] \\
& \leq \frac{6\lambda_t^2 M_{B, U}^2 D_{\bar{X}}^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4.
\end{aligned} \tag{5.53}$$

The result of (5.50) follows from plugging relations (5.52) and (5.53) into (5.51). \square

Proposition 5.4.4. *Suppose that parameters $\alpha_t, \lambda_t, \tau_t$ in Algorithm 5 satisfy (5.33), the smoothing parameters η_i^t satisfy the relation in (5.42). Then for any $T \geq 1$,*

$$\begin{aligned}
\bar{Q}_T(w_T, w) & \leq \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\frac{\alpha_t \left(L_{\bar{f}}^t + z^\top L_{\bar{h}}^t \right) D_{\bar{X}}^2}{2} + \eta_0^t D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^t D_{U_i}^2 \right. \\
& \quad \left. + \frac{6\lambda_t^2 M_{B, U}^2 D_{\bar{X}}^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \right] \\
& \quad + \frac{6\alpha_T \lambda_T^2 M_{B, U}^2 D_{\bar{X}}^2}{\tau_T} + \frac{3\alpha_T \lambda_T^2}{\tau_T} \sum_{i=1}^{\bar{m}} (\eta_i^{T-1} - \eta_i^T)^2 D_{U_i}^4 + \alpha_T \tau_T \bar{V} \\
& \quad + \eta_0^T D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^T D_{U_i}^2,
\end{aligned} \tag{5.54}$$

$\forall w \in \bar{X} \times \bar{Z}$.

Proof. Applying Lemma 5.7.2 and Proposition 5.4.3, we obtain

$$\begin{aligned}
\bar{Q}_T^\eta(w_T, w) &\leq \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\frac{\alpha_t \left(L_f^t + z^\top L_h^t D_{\bar{X}}^2 \right)}{2} + \frac{6\lambda_t^2 M_{B,U}^2 D_{\bar{X}}^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \right] \\
&\quad + \sum_{t=1}^T \frac{1 - \alpha_t}{\Gamma_t} \left((\eta_0^{t-1} - \eta_0^t) D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i (\eta_i^{t-1} - \eta_i^t) D_{U_i}^2 \right) \\
&\quad + \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}), z - r_t \rangle \right. \\
&\quad \quad \left. - \lambda_t \langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \right] \\
&\quad + \Gamma_T \sum_{t=1}^T \frac{\alpha_t \tau_t}{\Gamma_t} (V(r_{t-1}, z) - V(r_t, z)) \\
&= \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\frac{\alpha_t \left(L_f^t + z^\top L_h^t D_{\bar{X}}^2 \right)}{2} + \eta_0^t D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^t D_{U_i}^2 \right. \\
&\quad \quad \left. + \frac{6\lambda_t^2 M_{B,U}^2 D_{\bar{X}}^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \right] \\
&\quad + \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}), z - r_t \rangle \right. \\
&\quad \quad \left. - \lambda_t \langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \right] \\
&\quad + \Gamma_T \sum_{t=1}^T \frac{\alpha_t \tau_t}{\Gamma_t} (V(r_{t-1}, z) - V(r_t, z)),
\end{aligned} \tag{5.55}$$

where the second equality follows from

$$\sum_{t=1}^T \frac{1 - \alpha_t}{\Gamma_t} \left((\eta_0^{t-1} - \eta_0^t) D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i (\eta_i^{t-1} - \eta_i^t) D_{U_i}^2 \right) = \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left(\eta_0^t D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^t D_{U_i}^2 \right).$$

Similar to the derivation in (5.35), the last line in (5.55) follows

$$\Gamma_T \sum_{t=1}^T \frac{\alpha_t \tau_t}{\Gamma_t} (V(r_{t-1}, z) - V(r_t, z)) \leq \alpha_T \tau_T (\bar{V} - V(r_T, z)). \quad (5.56)$$

Next, using the relation $\frac{\lambda_t \alpha_t}{\Gamma_t} = \frac{\alpha_{t-1}}{\Gamma_{t-1}}$ and relation (5.56), we have

$$\begin{aligned} & \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\langle \ell_{\bar{h}_{\eta^t}}(x_{t-1}, p_t) - \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}), z - r_t \rangle \right. \\ & \quad \left. - \lambda_t \langle \ell_{\bar{h}_{\eta^{t-1}}}(x_{t-2}, p_{t-1}) - \ell_{\bar{h}_{\eta^{t-2}}}(x_{t-3}, p_{t-2}), z - r_{t-1} \rangle \right] - \alpha_T \tau_T V(r_T, z) \\ &= \frac{\alpha_T}{\Gamma_T} \langle \ell_{\bar{h}_{\eta^T}}(x_{T-1}, p_T) - \ell_{\bar{h}_{\eta^{T-1}}}(x_{T-2}, p_{T-1}), z - r_T \rangle - \alpha_T \tau_T V(r_T, z) \\ & \quad - \lambda_1 \langle \ell_{\bar{h}_{\eta^0}}(x_{-1}, p_0) - \ell_{\bar{h}_{\eta^{-1}}}(x_{-2}, p_{-1}), z - r_0 \rangle \\ &\leq \frac{6\alpha_T \lambda_T^2 M_{B,U}^2 D_X^2}{\tau_T} + \frac{3\alpha_T \lambda_T^2}{\tau_T} \sum_{i=1}^{\bar{m}} (\eta_i^{T-1} - \eta_i^T)^2 D_{U_i}^4, \end{aligned} \quad (5.57)$$

where the last inequality follows from Young's inequality and the initialization condition $x_0 = x_{-1} = x_{-2}, p_0 = p_{-1}$.

Finally, by the inequality (5.49) with $t = T$ and the results in (5.55), (5.56) and (5.57), we reach the conclusion in (5.54). \square

Proposition 5.4.4 can be easily extended to the convex constrained problem (5.3). More explicitly,

$$\begin{aligned} Q_T(w_T, w) &\leq \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\frac{\alpha_t z^\top L_H^t D_X^2}{2} + \sum_{i=0}^m z_i \eta_i^t D_{U_i}^2 \right. \\ & \quad \left. + \frac{6\lambda_t^2 M_{B,U}^2 D_X^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=0}^m (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \right] \\ & \quad + \frac{6\alpha_T \lambda_T^2 M_{B,U}^2 D_X^2}{\tau_T} + \frac{3\alpha_T \lambda_T^2}{\tau_T} \sum_{i=0}^m (\eta_i^{T-1} - \eta_i^T)^2 D_{U_i}^4 + \alpha_T \tau_T V \\ & \quad + \eta_0^T D_{U_0}^2 + \sum_{i=0}^m z_i \eta_i^T D_{U_i}^2. \end{aligned}$$

and

$$\begin{aligned}
\|H(x_T, f^*)\|_\infty &\leq \Gamma_T \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left[\frac{\alpha_t \max_{i=1, \dots, \bar{m}} L_{H_i}^t D_X^2}{2} + \max_{i=1, \dots, \bar{m}} \eta_i^T D_{U_i}^2 \right. \\
&\quad \left. + \frac{6\lambda_t^2 M_{B,U}^2 D_X^2}{\tau_t} + \frac{3\lambda_t^2}{\tau_t} \sum_{i=0}^m (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \right] \\
&\quad + \frac{6\alpha_T \lambda_T^2 M_{B,U}^2 D_X^2}{\tau_T} + \frac{3\alpha_T \lambda_T^2}{\tau_T} \sum_{i=0}^m (\eta_i^{T-1} - \eta_i^T)^2 D_{U_i}^4 + \alpha_T \tau_T V \\
&\quad + \eta_0^T D_{U_0}^2 + \max_{i=1, \dots, \bar{m}} \eta_i^T D_{U_i}^2.
\end{aligned}$$

Theorem 5.4.2 below demonstrates convergence rate of Algorithm 5.

Theorem 5.4.2. *Suppose parameters α_t, λ_t and τ_t are specified according to (5.36), with \bar{M} replaced $\bar{M}_{B,U}$ and*

$$\eta_i^t = \frac{\|B_i\| D_{\bar{X}}}{\sqrt{t} D_{U_i}}, i = 0, 1, \dots, m,$$

then for $t \geq 1$, we have

$$\begin{aligned}
\bar{Q}_t(w_t, w) &\leq \frac{8D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right)}{3\sqrt{t+1}} + \frac{4D_{\bar{X}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2}{3\bar{M}_{B,U} t(t+1)} + \frac{8\bar{M}_{B,U} D_{\bar{X}}}{9\sqrt{t+1}} \\
&\quad + \frac{4\bar{M}_{B,U} D_{\bar{X}}}{3(t+1)\sqrt{t}} + \frac{2D_{\bar{X}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2}{3t^2(t+1)\sqrt{t}\bar{M}_{B,U}} + \frac{18\bar{M}_{B,U} D_{\bar{X}} \bar{V}}{\sqrt{t+1}} + \frac{D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right)}{\sqrt{t}}.
\end{aligned} \tag{5.58}$$

Proof. Note first, since $\sqrt{t-1} \geq \sqrt{t-2}$, then

$$\begin{aligned}
(\eta_i^{t-2} - \eta_i^{t-1})^2 &= \frac{\|B_i\|^2 D_{\bar{X}}^2}{D_{U_i}^2} \left(\frac{1}{\sqrt{t-2}} - \frac{1}{\sqrt{t-1}} \right)^2 \\
&\leq \frac{\|B_i\|^2 D_{\bar{X}}^2}{D_{U_i}^2} \frac{1}{(t-1)(t-2)}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
& \sum_{t=3}^T \frac{3\alpha_t \lambda_t^2}{\Gamma_t \tau_t} \sum_{i=1}^{\bar{m}} (\eta_i^{t-2} - \eta_i^{t-1})^2 D_{U_i}^4 \\
& \leq \frac{D_{\bar{X}}}{3\bar{M}_{B,U}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2 \sum_{t=3}^T \frac{(t-1)^2}{\sqrt{tt}(t-1)(t-2)} \\
& \leq \frac{2D_{\bar{X}}}{3\bar{M}_{B,U}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2,
\end{aligned} \tag{5.59}$$

where the last inequality follows from the relation

$$\sum_{t=3}^T \frac{(t-1)^2}{\sqrt{tt}(t-1)(t-2)} \leq \sum_{t=3}^T \frac{1}{(t-2)^{3/2}} \leq \int_3^T \frac{1}{(t-2)^{3/2}} \leq 2.$$

Besides,

$$\begin{aligned}
& \sum_{t=1}^T \frac{\alpha_t}{\Gamma_t} \left(\eta_0^t D_{U_0}^2 + \sum_{i=1}^{\bar{m}} z_i \eta_i^t D_{U_i}^2 \right) \\
& = D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right) \sum_{t=1}^T \sqrt{t} \\
& \leq \frac{2D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right)}{3} T\sqrt{T}.
\end{aligned} \tag{5.60}$$

Moreover, since $\omega_i \geq 0$, $i = 0, 1, \dots, \bar{m}$, then $L_{\bar{f}}^t = \frac{\|B_0\|^2}{\eta_0^t + \omega_0} \leq \frac{\|B_0\|^2}{\eta_0^t}$, $L_{\bar{h}}^t = \frac{\|B_i\|^2}{\omega_i + \eta_i^t} \leq \frac{\|B_i\|^2}{\eta_i^t}$, $i = 1, \dots, \bar{m}$ and

$$\begin{aligned}
& \sum_{t=1}^T \frac{\alpha_t^2}{2\Gamma_t} \left(L_{\bar{f}}^t + z^\top L_{\eta}^t \right) D_{\bar{X}}^2 \\
& \leq D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right) \sum_{t=1}^T \frac{t\sqrt{t}}{t+1} \\
& \leq \frac{2D_{\bar{X}} \left(\|B_0\| D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i} \right)}{3} T\sqrt{T},
\end{aligned} \tag{5.61}$$

where the last inequality is due to $\sum_{t=1}^T \frac{t\sqrt{t}}{t+1} \leq \frac{2}{3}T\sqrt{T}$.

Using the relations in (5.59), (5.60), (5.61), similar to (5.38), we can show that

$$\begin{aligned} \bar{Q}_T^\eta(w_T, w) &\leq \frac{8D_{\bar{X}} \left(\|B_0\|D_{U_0} + \sum_{i=1}^{\bar{m}} z_i \|B_i\|D_{U_i} \right)}{3\sqrt{T+1}} + \frac{4D_{\bar{X}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2}{3\bar{M}_{B,U}T(T+1)} + \frac{8\bar{M}_{B,U}D_{\bar{X}}}{9\sqrt{T+1}} \\ &\quad + \frac{4\bar{M}_{B,U}D_{\bar{X}}}{3(T+1)\sqrt{T}} + \frac{2D_{\bar{X}} \sum_{i=1}^{\bar{m}} \|B_i\|^2 D_{U_i}^2}{3T^2(T+1)\sqrt{T}\bar{M}_{B,U}} + \frac{18\bar{M}_{B,U}D_{\bar{X}}\bar{V}}{\sqrt{T+1}}. \end{aligned}$$

We conclude the result in (5.58) by noting that $\eta_0^T D_{U_0}^2 + \sum_{i=0}^{\bar{m}} z_i \eta_i^T D_{U_i}^2 \leq \frac{D_{\bar{X}} \|B_0\| D_{U_0}}{\sqrt{T}} + \frac{D_{\bar{X}} \sum_{i=1}^{\bar{m}} z_i \|B_i\| D_{U_i}}{\sqrt{T}}$. \square

Similarly, for problem (5.3) with structured nonsmooth functions, we conclude that,

$$\begin{aligned} Q_t(w_t, w) &\leq \frac{8D_X \left(\sum_{i=0}^{m+1} z_i \|B_i\| D_{U_i} \right)}{3\sqrt{t+1}} + \frac{4D_X \sum_{i=0}^{m+1} \|B_i\|^2 D_{U_i}^2}{3M_{B,U}t(t+1)} + \frac{8M_{B,U}D_X}{9\sqrt{t+1}} \\ &\quad + \frac{4M_{B,U}D_X}{3(t+1)\sqrt{t}} + \frac{2D_X \sum_{i=0}^{m+1} \|B_i\|^2 D_{U_i}^2}{3t^2(t+1)\sqrt{t}M_{B,U}} + \frac{18M_{B,U}D_X V}{\sqrt{t+1}} + \frac{D_X \left(\sum_{i=0}^{m+1} z_i \|B_i\| D_{U_i} \right)}{\sqrt{t}}, \end{aligned}$$

$$\begin{aligned} \|\bar{h}(x_T, f^*)\|_\infty &\leq \frac{8D_X \left(\max_{i=0, \dots, m+1} \|B_i\| D_{U_i} \right)}{3\sqrt{t+1}} + \frac{4D_X \sum_{i=0}^{m+1} \|B_i\|^2 D_{U_i}^2}{3M_{B,U}t(t+1)} + \frac{8M_{B,U}D_X}{9\sqrt{t+1}} \\ &\quad + \frac{4M_{B,U}D_X}{3(t+1)\sqrt{t}} + \frac{2D_X \sum_{i=0}^m \|B_i\|^2 D_{U_i}^2}{3t^2(t+1)\sqrt{t}M_{B,U}} + \frac{18M_{B,U}D_X V'}{\sqrt{t+1}} + \frac{D_X \left(\max_{i=0, \dots, m+1} \|B_i\| D_{U_i} \right)}{\sqrt{t}}. \end{aligned}$$

5.5 Overall Complexity

In this section, we present the overall iteration complexity of the LCG method applied to the convex functional constrained problem in (5.2) with subproblem (5.3) solved by CGO.

As mentioned in Section 5.4.1, when we apply CGO for solving subproblem (5.3), we have $\bar{f}(x) = 0$, $\bar{h}(x) \equiv \bar{h}(x; l) = (f(x) - l, h(x))$ for a given level estimate l , $\bar{X} = X$ and $\bar{Z} = Z$. In the following lemma we show that the output of CGO satisfies the conditions (5.6)-(5.8) to guarantee the convergence of the outer loop of LCG.

Lemma 5.5.1. *When LCG does not terminate at iteration k , the output (γ_k, L_k, U_k) of CGO satisfies (5.6)-(5.8).*

Proof. First define the sequence $\{\beta^t\}$, across inner iteration (CGO iteration) $t \geq 1$ as follow: $\beta^t := (\beta_1^t, \dots, \beta_t^t)$, where $\beta_j^t = \begin{cases} \alpha_t, & \text{if } j = t, \\ (1 - \alpha_t)\beta_j^{t-1}, & \text{if } j \neq t, \end{cases}$ with $\beta_1^1 = \alpha_1$. Denote $r_j := (r_{j,0}, \dots, r_{j,m})$, where $r_{j,i}$ is the i -th element of vector r_j at iteration j , $j \leq t$. According to Algorithm 4, $\underline{h}_t(x; l)$ can be explicitly written as:

$$\underline{h}_t(x; l) = -\gamma_t l + \sum_{j=1}^t \beta_j^t \left[r_{j,0} \ell_f(x_{j-1}, x) + \sum_{i=1}^m r_{j,i} \ell_{h_i}(x_{j-1}, x) \right], \quad (5.62)$$

where $\gamma_t = \sum_{j=1}^t \beta_j^t r_{j,0}$ and we have $z_{t,i} = \sum_{j=1}^t \beta_j^t r_{j,i}$, $z_{t,i}$ is the i -th element of vector z_t at iteration t .

At each CGO iteration under outer iteration k , by the definition in (5.4), we have $\mathcal{L}_t(l) = L_t - \gamma_t(l - l_k)$. Arranging the terms and using the relation in (5.21) and (5.62), for each $l \in \mathbb{R}$, we have

$$\begin{aligned} \mathcal{L}_t(l) &= -\gamma_t l + (L_t + \gamma_t l_k) \\ &= -\gamma_t l + \min_{x \in X} \underline{h}_t(x; l_k) + \gamma_t l_k \\ &= \min_{x \in X} \underline{h}_t(x; l). \end{aligned}$$

Moreover, according to Lemma 5.4.1, it can be shown that $\underline{h}_t(x; l) \leq \gamma_t[f(x) - l] + \langle h(x), z_t \rangle, \forall x \in X$. Hence, $\mathcal{L}_t(l) = \min_{x \in X} \underline{h}_t(x; l) \leq \min_{x \in X} \gamma_t[f(x) - l] + \langle h(x), z_t \rangle \leq$

$\min_{x \in X} \max_{(\gamma, z) \in Z} \gamma[f(x) - l] + \langle h(x), z \rangle \equiv \phi(l)$. Immediately, we obtain the relation in (5.8). In view of Lemma 5.4.1 and the relation in (5.8), for a given level estimate $l \in \mathbb{R}$, L_k, U_k are the lower bound and upper bound of $\phi(l)$, respectively. Therefore, (5.7) is satisfied.

Now, we show (5.6). By (5.62), we have

$$\mathcal{L}_t(l) = -\gamma_t l + \min_{x \in X} \sum_{j=1}^t \beta_j^t \left[r_{j,0} \ell_f(x_{j-1}, x) + \sum_{i=1}^m r_{j,i} \ell_{h_i}(x_{j-1}, x) \right].$$

In the case where LCG is not terminated at outer iteration k , suppose CGO runs $t(k)$ iterations. According to Lemma 5.3.1, it produces a lower bound such that $L_{t(k)} > 0$.

Here $\gamma_k \equiv \gamma_{t(k)}$, $z_k \equiv z_{t(k)}$, $L_k \equiv L_{t(k)}$ and $\mathcal{L}_k(l_k) \equiv \mathcal{L}_{t(k)}(l_k)$. If $\gamma_k = 0$, recall that $\gamma_k = \sum_{j=1}^{t(k)} \beta_j^t r_{j,0}$, then we must have $r_{j,0} = 0, \forall j \leq t(k)$. This implies that $L_k = \mathcal{L}_k(l_k) = \min_{x \in X} \sum_{j=1}^{t(k)} \beta_j^{t(k)} \left[\sum_{i=1}^m r_{j,i} \ell_{h_i}(x_{j-1}, x) \right] \leq \min_{x \in X} \langle z_k, h(x) \rangle \leq 0$, which leads to contradiction. This shows that CGO returns γ_k such that $\gamma_k > 0$. \square

Corollary 5.5.1. *Suppose the algorithmic parameters of CGO are set to (5.36). Then for any $t \geq 1$ and $\forall w \in (X, Z)$,*

$$\begin{aligned} Q_t(w_t, w) &\leq \frac{2z^\top L_{\bar{h}} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right], \\ f(x_t) - f^* &\leq \frac{2 \max_{i=1, \dots, m+1} L_{\bar{h}_i} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right], \\ \|h(x_t)\|_\infty &\leq \frac{2 \max_{i=1, \dots, m+1} L_{\bar{h}_i} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right]. \end{aligned}$$

Proof. The convergence analysis on the gap function $Q_t(w_t, w)$ is similar to the one on the general case when treating $\bar{f}(\cdot) = 0$ (see Theorem 5.4.1). We conclude that

$$Q_t(w_t, w) \leq \frac{2z^\top L_{\bar{h}} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right], \forall w := (x, z) \in X \times Z.$$

Now we analyze the bound of $\|\bar{h}(x_t; f^*)\|_\infty$, thus the bounds of $f(x_t) - f^*$ and of $\|h(x_t)\|_*$. Suppose there exists at least one element of $\bar{h}(x_t; f^*)$ is positive, otherwise, we arrive trivially at $\|\bar{h}(x_t; f^*)\|_* \leq 0$. Define $w' := (x^*, z')$, where x^* is the optimal primal solution. z' is defined as follows: $z'_j = 1$ if j is one of the indices such that $j \in \arg \max_{i=1, \dots, m} \bar{h}_i(x_t; f^*)$ and $z'_i = 0$ otherwise. By the definition of z' and the relation $\underline{h}_t(x^*; f^*) \leq \langle \bar{h}(x^*; f^*), z_t \rangle \leq 0$, we have

$$\begin{aligned} Q_t(w_t, w') &= \langle \bar{h}(x_t; f^*), z' \rangle - \underline{h}_t(x^*; f^*) \\ &\geq \|\bar{h}(x_t; f^*)\|_\infty. \end{aligned} \tag{5.63}$$

Note also,

$$\|\bar{h}(x_t; f^*)\|_\infty = \max\{f(x_t) - f^*, h(x_t)\}$$

Then we have $\forall w \in (X, Z)$,

$$\begin{aligned} f(x_t) - f^* &\leq \frac{2 \max_{i=1, \dots, m+1} L_{\bar{h}_i} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right], \\ \|h(x_t)\|_\infty &\leq \frac{2 \max_{i=1, \dots, m+1} L_{\bar{h}_i} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right]. \end{aligned}$$

□

We are now ready to establish the overall iteration complexity of the LCG method.

Theorem 5.5.1. *Suppose that the algorithmic parameters of CGO are set to (5.36). Then the total number of CGO iterations required to find an ϵ -solution $\bar{x} \in X$ of (5.2) can be bounded by $\mathcal{O}\left(\frac{1}{\epsilon^2} \log\left(\frac{1}{\epsilon}\right)\right)$.*

Proof. Using the result in Corollary 5.5.1 and the fact that $U_t - L_t = \max_{w \in X \times Z} Q_t(w_t, w)$, we

immediately obtain

$$U_t - L_t \leq \frac{2 \max_{i=1, \dots, m+1} L_{\bar{h}_i} D_X^2}{t+1} + \frac{\bar{M} D_X}{\sqrt{t+1}} \left[18\bar{V} + \frac{7}{6} \right].$$

Consequently, given precision $\epsilon > 0$, to attain $U_k - L_k \leq \epsilon$ at each call of CGO, the number of iterations is bounded by $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$. Furthermore, in view of Theorem 5.3.1, the required number of outer loop iterations to obtain $U_k \leq \epsilon$ and thus $f(x_k) - f^* \leq \epsilon$, $\|h(x_k)\|_\infty \leq \epsilon$ is bounded by $\mathcal{O}\left(\log \frac{1}{\epsilon}\right)$. Combining these two results, the overall iteration complexity of LCG solving the convex constrained problem (5.2) is $\mathcal{O}\left(\frac{1}{\epsilon^2} \log\left(\frac{1}{\epsilon}\right)\right)$. \square

It is worth mentioning here that as the output solution x_k may not be a feasible solution such that $h(x_k) \leq 0$, we develop a lower bound for $f(x_k) - f^*$, which is presented in Lemma 5.5.2 below.

Lemma 5.5.2. *Let $(x^*, y^*) \in \mathbb{R}_n \times \mathbb{R}_+^m$ be the saddle point of the convex constrained problem (5.2). Let (γ^*, z^*) be the optimal dual solution of the root finding problem*

$$\min_{x \in X} \max_{(\gamma, z) \in Z} L(x, (\gamma, z)) := \gamma[f(x) - f^*] + \langle h(x), z \rangle \quad (5.64)$$

(i.e. problem (5.3) with $l = f^*$). Denote $[\cdot]_+ := \max\{0, \cdot\}$. Then $\forall x \in X$, $f(x) - f^*$ is lower bounded such that

$$f(x) - f^* \geq -\min\{\|y^*\|, \frac{\|z^*\|}{\gamma^*}\} \| [h(x)]_+ \|. \quad (5.65)$$

Proof. According to the result in [34, Corollary 2], we have

$$f(x) - f^* \geq -\|y^*\| \| [h(x)]_+ \|. \quad (5.66)$$

Since $(x^*, (\gamma^*, z^*))$ is a pair of saddle point of (5.64), thus by the saddle point theorem, we

have

$$L(x^*, (\gamma, z)) \leq L(x^*, (\gamma^*, z^*)) \leq L(x, (\gamma^*, z^*)), \forall x \in X, (\gamma, z) \in Z.$$

Using the above relation and the facts that $L(x^*, (\gamma^*, z^*)) = 0$, $\langle [h(x)]_+ - h(x), z^* \rangle \geq 0$, we have $\forall x \in X$:

$$\gamma^*[f(x) - f^*] = L(x, (\gamma^*, z^*)) - \langle h(x), z^* \rangle \geq L(x^*, (\gamma^*, z^*)) - \langle h(x), z^* \rangle \geq -\|[h(x)]_+\| \|z^*\| \quad (5.67)$$

Combining (5.66) and (5.67), we obtain $\forall x \in X$:

$$f(x) - f^* \geq -\min\{\|y^*\|, \frac{\|z^*\|}{\gamma^*}\} \|[h(x)]_+\|. \quad (5.68)$$

□

5.6 Modified Level Conditional Gradient Method

In LCG, we require that the initial level estimate satisfies $l_1 \leq f^*$. Otherwise, the algorithm terminates at the first outer iteration. To see this, if $l_1 \geq f^*$, then $L_1 \leq \phi(l_1) \leq \phi(f^*) = 0$, which holds because L_1 is the lower bound of $\phi(l_1)$ and ϕ is a non-increasing function. Hence by the stopping criteria of CGO, $U_1 \leq L_1 + (1 - \mu)\epsilon \leq \epsilon$, which results in the termination of the algorithm. It should be noted that, by slightly modifying the outer scheme of LCG, it is also possible to approximate f^* from above starting with an initial level $l_1 \geq f^*$.

In this section, we focus on the modified version of LCG (MLCG), which generates a sequence of decreasing level estimates that approximate from above the optimal value of the problem. More specifically, we present the algorithmic framework of the MLCG method and convergence analysis of the outer loop. Remarkably, CGO requires no change in MLCG and remains as what it is in LCG.

To approximate the optimal value f^* by a sequence of decreasing level estimate from above, the original LCG (Algorithm 3, Section 5.3) mainly takes changes in the following

steps: (1) In the initialization step, l_1 is required to overestimate f^* . One plausible option is letting $l_1 = f(x_0)$, s.t. $x_0 \in X, h_i(x_0) \leq 0, i = 1, \dots, m$; (2) The update rule of the level estimate is changed to $l_{k+1} = l_k + U_k$; (3) The algorithm terminal condition is changed to $L_k \geq -\epsilon\tilde{\kappa}$. Here $\tilde{\kappa}$ is a lower approximation of the condition number κ , where

$$\kappa := -\frac{\phi(l_1)}{l_1 - f^*}.$$

It can be easily verified that $\kappa \leq 1$. We can choose $\tilde{\kappa} = -\frac{U_1}{l_1 - \tilde{f}}$, where $\tilde{f} := \min\{f(x_0) + \langle \nabla f(x_0), x - x_0 \rangle : x \in X\}$. In this way, $\tilde{f} \leq f^*$ and $\tilde{\kappa} \leq \kappa$. Similar to [46], the outer loop iteration complexity relies on κ and we will demonstrate later how the terminal condition $L_k \geq -\epsilon\tilde{\kappa}$ implies the convergence of the outer loop. The MLCG method is summarized in Algorithm 6.

Algorithm 6 Modified Level Conditional Gradient Method (MLCG)

- 1: Inputs: $\epsilon > 0, \mu \in (0, 1)$.
 - 2: Initialization: $x_0 \in X, h_i(x_0) \leq 0, i = 1, \dots, m, l_1 = f(x_0)$.
 - 3: **for** $k = 1, 2, \dots$ **do**
 - 4: Call CGO with input l_k and obtain approximate solutions $(x_k; (\gamma_k, z_k)) \in X \times Z$, lower bound L_k , upper bound U_k and if $k = 1$, obtain $\tilde{\kappa} = -\frac{U_1}{l_1 - \tilde{f}}$ such that $U_k - L_k \leq (1 - \mu)\tilde{\kappa}\epsilon$.
 - 5: **if** $L_k \geq -\epsilon\tilde{\kappa}$ **then**
 - 6: Terminate and return x_k .
 - 7: **end if**
 - 8: $l_{k+1} = l_k + U_k$.
 - 9: **end for**
-

Remark 5.6.1. *There is one edge case such that $l_1 = f^*$ leading to termination of the algorithm at iteration $k = 1$, thus we will not consider this situation in the convergence analysis. In this case, $U_1 \geq \phi(l_1) = \phi(f^*) = 0$. Let $\tilde{\kappa} = \epsilon$ (instead of $-\frac{U_1}{l_1 - \tilde{f}}$). In view of CGO terminal condition, we have $L_1 \geq U_1 - (1 - \mu)\tilde{\kappa}\epsilon \geq -(1 - \mu)\tilde{\kappa}\epsilon \geq -\tilde{\kappa}\epsilon$ and the algorithm terminates. Since $l_1 = f^*$, the condition number κ does not exist. In fact, the algorithm terminates at $k = 1$, thus the iteration complexity in this case will not be affected by κ .*

5.6.1 Outer Loop Iteration Complexity of MLCG

We start with several lemmas that are of important use in the outer loop convergence analysis.

Lemma 5.6.1. *When MLCG is not terminated at iteration k ($k \geq 1$), then $U_k \leq \mu L_k$.*

Proof. We first demonstrate that $U_k \geq 0$ then the algorithm terminates at iteration k . Indeed, at $k > 1$, if $U_k \geq 0$, then by the stopping criteria of CGO, $L_k \geq U_k - (1 - \mu)\tilde{\kappa}\epsilon \geq -(1 - \mu)\tilde{\kappa}\epsilon \geq -\tilde{\kappa}\epsilon$. Consequently, the algorithm terminates. At iteration k , since the algorithm is not terminated, then $L_k < -\epsilon\tilde{\kappa}$. By the stopping criteria of CGO such as $U_k - L_k \leq (1 - \mu)\tilde{\kappa}\epsilon$, we have $U_k - L_k \leq (1 - \mu)\tilde{\kappa}(-L_k/\tilde{\kappa}) = (1 - \mu)(-L_k)$. Rearranging the terms in the above inequality, we arrive at $U_k \leq \mu L_k$. □

Lemma 5.6.2. $l_1 > \dots > l_k > \dots \geq f^*$.

Proof. We have discussed the case when $l_1 = f^*$ in Remark 5.6.1. We will focus on the case where $l_1 > f^*$. We show $l_{k+1} - f^* \geq 0$ by induction. Suppose $l_k - f^* \geq 0$, then

$$\begin{aligned} l_{k+1} - f^* &= l_k + U_k - f^* \\ &\geq l_k + \phi(l_k) - f^* \\ &\geq l_k - f^* - (l_k - f^*) \\ &= 0. \end{aligned}$$

where the last inequality is due to $\phi(f^*) - \phi(l_k) \leq l_k - f^*$ with $l_k - f^* \geq 0$ (induction assumption) and $\phi(f^*) = 0$. Next, we show that $l_{k+1} < l_k$. By Lemma 5.6.1, we have $U_k \leq \mu L_k < 0$ when the algorithm does not terminate, then $l_{k+1} - l_k = U_k < 0$. □

The theorem below shows the outer iteration complexity of MLCG and demonstrates that when it terminates it outputs an ϵ -optimal and ϵ -feasible solution.

Theorem 5.6.1. *For $k \geq 1$, Algorithm 6 generates l_k that satisfies*

$$l_k - f^* \leq (1 - \kappa\mu)^{k-1}(l_1 - f^*),$$

where $\mu \in (0, 1)$, $\kappa \in (0, 1]$ and l_1 is the initial level estimate of f^* such that $l_1 > f^*$. Moreover, at the termination of Algorithm 6, it returns an ϵ -optimal and ϵ -feasible solution.

Proof. We have the following relation

$$\begin{aligned} l_{k+1} - f^* &= l_k + U_k - f^* \\ &\leq l_k - f^* + \mu L_k \\ &\leq l_k - f^* + \mu\phi(l_k) \\ &\leq l_k - f^* - \mu\kappa(l_k - f^*) \\ &= (1 - \mu\kappa)(l_k - f^*). \end{aligned}$$

The first inequality follows from Lemma 5.6.1, the second inequality from $L_k \leq \phi(l_k)$, the third is due to $\frac{\phi(l_k) - \phi(f^*)}{l_k - f^*} \leq \frac{\phi(l_1) - \phi(f^*)}{l_1 - f^*} \equiv -\kappa$, which holds because of the convexity of $\phi(\cdot)$ and the relation that $l_k \leq l_1$, $k \geq 1$. Using induction on k , we have $l_{k+1} - f^* \leq (1 - \mu\kappa)^k(l_1 - f^*)$.

Next, we will show that given precision ϵ , the terminal condition $L_k \geq -\epsilon\tilde{\kappa}$ implies $l_k - f^* \leq \epsilon$. Note first $-\phi(l_k) \leq -L_k$ and $-\phi(l_k) \geq \kappa(l_k - f^*)$. Using these inequalities and the relation that $\tilde{\kappa} \leq \kappa$, we obtain $l_k - f^* \leq -L_k/\kappa \leq \frac{\tilde{\kappa}}{\kappa}\epsilon \leq \epsilon$. Moreover, in view of Lemma 5.6.2, we have $L_k \leq \phi(l_k) \leq \phi(f^*) = 0$, then $U_k \leq L_k + (1 - \mu)\tilde{\kappa}\epsilon \leq (1 - \mu)\tilde{\kappa}\epsilon$. This implies that $f(x_k) - f^* = (f(x_k) - l_k) + (l_k - f^*) \leq U_k + \frac{\tilde{\kappa}}{\kappa}\epsilon \leq ((1 - \mu)\tilde{\kappa} + \frac{\tilde{\kappa}}{\kappa})\epsilon$ and $\max_{i=1, \dots, m} h_i(x_k) \leq U_k \leq (1 - \mu)\tilde{\kappa}\epsilon \leq \epsilon$. \square

5.7 Auxiliary Lemmas

To establish the convergence for CGO, we tap into the following three well-studied results. Throughout the analysis, we need to use the following notation: let α_t be defined in

Algorithm 4, define the sequence $\{\Gamma_t\}$ as $\Gamma_t := \begin{cases} 1, & \text{if } t = 1, \\ (1 - \alpha_t)\Gamma_{t-1}, & \text{if } t > 1, \end{cases} \quad t = 1, 2, \dots$

The first lemma is the so-called ‘‘three-point’’ lemma which characterizes the optimality condition of the dual update in (5.15).

Lemma 5.7.1. [23, Lemma 3.1] *Let r_t be defined in (5.15). Then*

$$\langle -\tilde{h}_t, r_t - z \rangle + \tau_t V(r_{t-1}, r_t) \leq \tau_t V(r_{t-1}, z) - \tau_t V(r_t, z), \forall z \in \bar{Z}.$$

The second lemma deals with telescoping sums.

Lemma 5.7.2. [23, Lemma 3.17] *Let $\{R_t\}$ be some given sequence. A sequence $\{S_t\}$ such that*

$$S_t \leq (1 - \alpha_t)S_{t-1} + R_t, \quad t = 1, 2, \dots,$$

satisfies

$$\frac{S_t}{\Gamma_t} \leq (1 - \alpha_1)S_0 + \sum_{j=1}^t \frac{R_j}{\Gamma_j}.$$

We utilize the following properties for smooth functions.

Lemma 5.7.3. [23, Lemma 3.2]: *Let p_t, x_t be defined in Algorithm 4. If \bar{f} and \bar{h} are smooth functions such that $\forall x_1, x_2 \in X$, $\|\nabla \bar{f}(x_1) - \nabla \bar{f}(x_2)\| \leq L_{\bar{f}}\|x_1 - x_2\|$ and $\|\nabla \bar{h}_i(x_1) - \nabla \bar{h}_i(x_2)\| \leq L_{\bar{h}_i}\|x_1 - x_2\|, i = 1, \dots, \bar{m}$, then the following conditions hold:*

$$\begin{aligned}\bar{f}(x_t) &\leq (1 - \alpha_t)\bar{f}(x_{t-1}) + \alpha_t \ell_{\bar{f}}(x_{t-1}, p_t) + \frac{L_{\bar{f}}\alpha_t^2}{2}\|p_t - x_{t-1}\|^2, \\ \bar{h}_i(x_t) &\leq (1 - \alpha_t)\bar{h}_i(x_{t-1}) + \alpha_t \ell_{\bar{h}_i}(x_{t-1}, p_t) + \frac{L_{\bar{h}_i}\alpha_t^2}{2}\|p_t - x_{t-1}\|^2, i = 1, \dots, \bar{m}.\end{aligned}$$

CHAPTER 6
PROJECTION-FREE METHODS FOR NONCONVEX FUNCTIONAL
CONSTRAINED OPTIMIZATION

6.1 Overview

In the last chapter, we have developed a projection-free method guided by level-set update for a class of convex constrained problems in the context of risk averse and sparsity. In sparse optimization, nonconvex formulations have attracted much attentions due to its empirical merits (see e.g. [53, 52]). In a substantial amount of literature (e.g. [44, 110]), sparsity was modeled by ℓ_p -regularized problem ($p > 0$), where in some cases a Lasso approach [108] is deployed. Alternatively, nonconvex approximations such as DC were adopted (see, e.g. [52, 53, 118, 120]).

In this chapter, to cope with the nonconvex functional constrained optimization problems (when f is nonconvex in (5.2)), we develop three approaches: the Exact/Inexact Proximal Point Level Conditional Gradient (EPP-LCG/IPP-LCG) methods and the Direct Nonconvex Conditional Gradient (DNCG) method. The proposed EPP-LCG/IPP-LCG methods utilize the proximal point framework and solve a series of convex subproblems. By solving each subproblem, it leverages the proposed LCG method, thus averting the effect from large Lagrangian multipliers. We show that the iteration complexity of the algorithms is bounded by $\mathcal{O}\left(\frac{1}{\epsilon^3} \log\left(\frac{1}{\epsilon}\right)\right)$ in order to obtain an (approximate) KKT point. However, the proximal-point type method has triple-layer structure and may not be easily implementable. To alleviate the issue, we also propose the DNCG method, which is the first single-loop projection-free algorithm for solving nonconvex functional constrained problem in the literature. This algorithm provides a drastically simpler framework as it only contains three updates in one loop. We show that the iteration complexity to find an ϵ -Wolfe point is

bounded by $\mathcal{O}(1/\epsilon^4)$. To the best of our knowledge, all these developments are new for projection-free methods for nonconvex optimization.

Finally, we present novel convex and nonconvex functional constrained models that are well-suited to risk averse sparse optimization problems in portfolio selection and IMRT treatment planning. These models incorporate different types of risk aversion and sparsity requirements and can be solved efficiently by our proposed algorithms. For the portfolio selection problem with cardinality requirement, our numerical experiments show that all algorithms (LCG, IPP-LCG, DNCG) are efficient in jointly minimizing risk while lowering cardinality of the selected assets in a rather short computational time for real-world and large-scale datasets. For the IMRT application, the proposed DNCG method, equipped with initial points output from LCG, satisfies the requirement of meeting a set of very challenging clinical criteria and selecting sparse angles in order to reduce the radiation time and the treatment delivery time, which accounts for a desirable treatment plan. It is worth mentioning that such requirements could not be satisfied by using any existing methods developed in the literature (e.g., [100, 101, 31]).

The rest of the chapter is organized as follow. We first describe a class of nonconvex functional constraints problems in 6.2. Then in 6.3, we introduce the proximal point methods targeted on a class of convex constrained problems with exact and inexact versions. Next in 6.4, we present the Nonconvex Conditional Gradient method. Lastly, we apply these methods to the portfolio selection problem and IMRT treatment planning problem and report numerical results accordingly in 6.5. Importantly, we also display the results of applying convex methods discussed in 5 for a more complete illustration and comparison with the nonconvex algorithms.

6.2 Nonconvex Functional Constrained Optimization Problem

In this section, we focus on the nonconvex functional constrained problem (5.2), where f is nonconvex and $h_i, i = 1, \dots, m$ are convex. Due to the difficulty of solving the nonconvex

functional constrained optimization to global optimality (even to local optimality), we seek an approximate stationary points of problem (5.2). We introduce two methods to solve this problem: Inexact Proximal Point Level Conditional Gradient (IPP-LCG) and Direct Nonconvex Conditional Gradient (DNCG).

Throughout the section, we make the following assumptions: (a) f is L_f -smooth and M_f -Lipschitz continuous; (b) h_i is L_{h_i} -smooth and M_{h_i} -Lipschitz continuous; (c) f satisfies a lower curvature condition such that

$$f(x) - f(y) - \langle \nabla f(y), x - y \rangle \geq -\frac{L_f}{2} \|x - y\|^2, \forall x, y \in X. \quad (6.1)$$

Here we assume that $\|\cdot\|$ is an inner product norm in this section for the sake of simplicity.

6.3 Proximal Point Methods for Nonconvex Functional Constrained Problem

In this part, we present the exact proximal point method and an inexact variant for solving the problem with nonconvex structure by extending the LCG method discussed in Chapter 5. Given the current iterate $x_{j-1} \in X$, we solve the following convex subproblem:

$$\begin{aligned} \min \quad & f(x; x_{j-1}) := f(x) + \frac{L_f}{2} \|x - x_{j-1}\|^2 \\ \text{s.t.} \quad & h_i(x) \leq 0, \quad i = 1, \dots, m, \\ & x \in X. \end{aligned} \quad (6.2)$$

We assume that the Slater conditions holds for (6.2), i.e., $\exists \tilde{x} \in X$ such that $h_i(\tilde{x}) < 0$, $i = 1, \dots, m$, and use (x_j^*, y_j^*) to denote a pair of its primal and dual solutions.

6.3.1 Exact Proximal Point Method

The idea of the exact proximal point method is to solve the convex subproblems iteratively as described in Algorithm 7.

Algorithm 7 Exact Proximal Point Algorithm

- 1: Parameter: precision ϵ .
- 2: Initialization: a feasible point x_0 .
- 3: **for** $j = 1, 2, \dots, J$ **do**
- 4:

$$\begin{aligned} x_j &= \arg \min_{x \in X} f(x; x_{j-1}) \\ \text{s.t. } & h_i(x) \leq 0, \quad i = 1, 2, \dots, m. \end{aligned} \tag{6.3}$$

- 5: **if** $\|x_{j-1} - x_j\| \leq \epsilon$ **then**
 - 6: Terminate and return x_j .
 - 7: **end if**
 - 8: **end for**
-

We utilize the LCG method to solve the subproblems (6.3). We assume that for the subproblem (6.3) for each $j \geq 1$, there exists a strictly feasible x'_j such that $h_i(x'_j) < 0, i = 1, \dots, m$, respectively. Therefore, the existence of KKT points for each subproblem is verified by this Slater condition.

The exact proximal point method aims to seek an ϵ -KKT points of problem (5.2) defined below.

Definition 6.3.1. $x \in X$ is an ϵ -KKT point of problem (5.2) if $h_i(x) \leq 0, i = 1, \dots, m$ and there exists y such that $y \geq 0$ and

$$\begin{aligned} \sum_{i=1}^m |y^i h_i(x)| &\leq \epsilon, \\ d \left(\nabla f(x) + \sum_{i=1}^m y^i \nabla h_i(x), -N_X(x) \right)^2 &\leq \epsilon. \end{aligned} \tag{6.4}$$

Theorem 6.3.1. *If the Slater condition holds for each subproblem in 7, then for j^* :*

$\arg \min_{1 \leq j \leq J} \|x_j - x_{j-1}\|^2$, x_{j^*} is an ϵ_J -KKT point with

$$\epsilon_J = \frac{8\underline{L}_f L_v [f(x_0) - f^*]}{3J}. \quad (6.5)$$

Proof. According to the strong convexity of $f(\cdot; x_{j-1})$, and optimality of x_j for solving subproblem (??) at the j -th iteration, we have for all feasible x that

$$\begin{aligned} f(x, x_{j-1}) &\geq f(x_j; x_{j-1}) + \langle \nabla f(x_j; x_{j-1}), x - x_j \rangle + \frac{\underline{L}_f}{2} \|x - x_j\|^2 \\ &\geq f(x_j; x_{j-1}) + \frac{\underline{L}_f}{2} \|x - x_j\|^2 \end{aligned} \quad (6.6)$$

and

$$h_i(x_j) \leq 0. \quad (6.7)$$

Replacing x in (6.6) with x_{j-1} and using the definition of $f(\cdot; x_{j-1})$, we obtain

$$\|x_{j-1} - x_j\|^2 \leq \frac{2}{3\underline{L}_f} [f(x_{j-1}) - f(x_j)]. \quad (6.8)$$

Summing up (6.8) by $j = 1, 2, \dots, J$ yields

$$\sum_{j=1}^J \|x_{j-1} - x_j\|^2 \leq \frac{2}{3\underline{L}_f} [f(x_0) - f(x_J)] \leq \frac{2}{3\underline{L}_f} [f(x_0) - f^*]. \quad (6.9)$$

Finally, by the definition of j^* , we have $J\|x_{j^*} - x_{j^*-1}\|^2 \leq \sum_{j=1}^J \|x_{j-1} - x_j\|^2$. Therefore

$$\|x_{j^*} - x_{j^*-1}\|^2 \leq \frac{2}{3J\underline{L}_f} [f(x_0) - f^*]. \quad (6.10)$$

Now let $\{y_{j^*}^i\}$ denote the dual solution corresponds to x_{j^*} , from the KKT condition for the

j^* -th subproblem we have

$$\sum_{i=1}^m y_{j^*}^i h_i(x_{j^*}) = 0, \quad y_{j^*}^i \geq 0, \quad (6.11)$$

and

$$\begin{aligned} d \left(\nabla f(x_{j^*}) + \sum_{i=1}^m y_{j^*}^i \nabla h_i(x_{j^*}), -N_X(x_{j^*}) \right)^2 &\leq 4\underline{L}_f^2 L_v \|x_{j^*} - x_{j^*-1}\|^2 \\ &\leq \frac{8\underline{L}_f L_v}{3J} [f(x_0) - f^*]. \end{aligned} \quad (6.12)$$

Combining (6.7), (6.11) and (6.12), we conclude that x_{j^*} is an ϵ_J -KKT point of problem (5.2), where ϵ_J is defined in (6.5). \square

Theorem 6.3.1 states that in order to achieve ϵ -KKT point of problem (5.2), the exact proximal point method needs to perform $\mathcal{O}(1/\epsilon)$ iterations.

6.3.2 Inexact Proximal Point Method

The main idea of the IPP-LCG method is to leverage the LCG method (see Section 5.3) to *inexactly* solve a sequence of convex subproblems (6.2) that approximate the original non-convex problem. As described in Algorithm 8, at the j -th iteration the IPP-LCG method calls LCG to solve subproblem (6.2) to obtain a (δ^f, δ^h) -optimal solution x_j s.t.

$$f(x_j; x_{j-1}) - f(x_j^*; x_{j-1}) \leq \delta^f; \quad \|[h(x_j)]_+\| \leq \delta^h.$$

Among all the candidate solutions across iterations, the method picks $x_{\hat{j}}$ such that $\hat{j} \in \arg \min_{j=1, \dots, J} \{f(x_{j-1}) - f(x_j)\}$ as the output solution.

We use the following criterion to measure the progress of the IPP-LCG method.

Definition 6.3.2. For problem (5.2),

(i) x' is an (ϵ, δ) -KKT point if $x' \in X$ and there exists (x, y) such that $h_i(x) \leq 0, x \in$

Algorithm 8 Inexact Proximal Point Level Conditional Gradient Method (IPP-LCG)

Initialization: $x_0 \in X$.

for $j = 1, 2, \dots, J$ **do**

 Call LCG to solve (6.2) and return a (δ^f, δ^h) -optimal solution x_j .

end for

Select \hat{j} such that $\hat{j} \in \arg \min_{j=1, \dots, J} \{f(x_{j-1}) - f(x_j)\}$.

Terminate and return $x_{\hat{j}}$.

$X, y_i \geq 0, i = 1, \dots, m$ and

$$\begin{aligned} \sum_{i=1}^m |y_i h_i(x)| &\leq \epsilon, \\ \left[d \left(\nabla f(x) + \sum_{i=1}^m y_i \nabla h_i(x), -N_X(x) \right) \right]^2 &\leq \epsilon, \\ \|x' - x\|^2 &\leq \delta, \end{aligned} \quad (6.13)$$

where $\epsilon, \delta > 0$, $d(\cdot, \cdot)$ denotes the distance between two sets A and B such that

$$d(A, B) := \min_{a \in A, b \in B} \|a - b\|;$$

(ii) x is an ϵ -KKT point (paired with y) if it satisfies the first two criteria in (6.13) with

$$h_i(x) \leq 0, x \in X, y_i \geq 0, i = 1, \dots, m.$$

The following two lemmas serve as building blocks for establishing the convergence rate of IPP-LCG. In particular, Lemma 6.3.1 characterizes an important property of the optimal solution of the subproblem (6.2) and Lemma 6.3.2 states that the Slater condition enforces uniform boundness on y_j^* .

Lemma 6.3.1. *If x_j^* is a KKT point (paired with y_j^*) of the subproblem (6.2), then $\forall x \in X$,*

$$f(x; x_{j-1}) - f(x_j^*; x_{j-1}) + \langle y_j^*, h(x) \rangle \geq \frac{L_f}{2} \|x_j^* - x\|^2.$$

Proof. By (6.1), we have that that $f(\cdot; x')$ is strongly convex. Together by the strong convexity of $f(\cdot; x_{j-1})$ and convexity of $h_i(\cdot)$ as well as the fact that $y_j^* \geq 0$, we have

$$\begin{aligned}
f(x; x_{j-1}) + \langle y_j^*, h(x) \rangle &\geq f(x_j^*; x_{j-1}) + \langle \nabla f(x_j^*; x_{j-1}), x - x_j^* \rangle \\
&\quad + \frac{\underline{L}_f}{2} \|x - x_j^*\|^2 + \sum_{i=1}^m y_{j,i}^* (h_i(x_j^*) + \langle \nabla h_i(x_j^*), x - x_j^* \rangle) \\
&\geq f(x_j^*; x_{j-1}) + \frac{\underline{L}_f}{2} \|x - x_j^*\|^2, \forall x \in X,
\end{aligned} \tag{6.14}$$

where the last inequality follows from properties of the KKT point that $\sum_{i=1}^m y_{j,i}^* h_i(x_j^*) = 0$ and $\nabla f(x_j^*; x_{j-1}) + \sum_{i=1}^m y_{j,i}^* \nabla h_i(x_j^*)$ belongs to the normal cone of X . \square

Lemma 6.3.2. *Suppose there exists $\tilde{x} \in X$ such that $h_i(\tilde{x}) < 0$, $i = 1, \dots, m$, then the dual solution y_j^* is uniformly bounded such that*

$$\|y_j^*\|_1 \leq \frac{f(\tilde{x}) - f^* + \underline{L}_f \bar{V}}{\|h(\tilde{x})\|_\infty}, \tag{6.15}$$

where $\|\cdot\|_1$ denote the ℓ_1 norm and $\bar{V} := \max_{x,y \in X} \frac{1}{2} \|x - y\|^2$.

Proof. Using Lemma 6.3.1 and replacing x with \tilde{x} , we have

$$\begin{aligned}
-\langle y_j^*, h(\tilde{x}) \rangle &\leq f(\tilde{x}; x_{j-1}) - f(x_j^*; x_{j-1}) - \frac{\underline{L}_f}{2} \|\tilde{x} - x_j^*\|^2 \\
&= f(\tilde{x}) - f(x_j^*) + \underline{L}_f [\|\tilde{x} - x_{j-1}\|^2 - \|x_j^* - x_{j-1}\|^2] - \frac{\underline{L}_f}{2} \|\tilde{x} - x_j^*\|^2 \\
&\leq f(\tilde{x}) - f(x_j^*) + \frac{\underline{L}_f}{2} \|\tilde{x} - x_{j-1}\|^2 \\
&\leq f(\tilde{x}) - f(x_j^*) + \underline{L}_f \bar{V}.
\end{aligned} \tag{6.16}$$

Note that $h_i(\tilde{x}) \leq -\|h(\tilde{x})\|_\infty, i = 1, \dots, m$, then $-\langle y_j^*, h(\tilde{x}) \rangle \geq \|y_j^*\|_1 \|h(\tilde{x})\|_\infty$. \square

We are now ready to present the convergence result for the IPP-LCG method.

Theorem 6.3.2. *The total number of CGO iterations performed by the IPP-LCG method to compute an (ϵ, ϵ) -KKT point of problem (5.2) is bounded by $\mathcal{O}\left(\frac{1}{\epsilon^3} \log\left(\frac{1}{\epsilon}\right)\right)$.*

Proof. From Algorithm 8, we have $\delta^f \geq f(x_j; x_{j-1}) - f(x_j^*; x_{j-1})$ and $\delta^h \geq \|h(x_j)\|_\infty$. Define $B := \frac{f(\bar{x}) - f^* + \underline{L}_f \bar{V}}{\|h(\bar{x})\|_\infty}$, $\epsilon_J := \frac{2}{\underline{L}_f}(\delta^f + B\delta^h)$, $\epsilon'_J := \frac{8\underline{L}_f}{J}[f(x_0) - f(x_J)] + \delta^f + B\delta^h$. Note first, since (x_j^*, y_j^*) is a pair of optimal solution of subproblem (6.2), then the complementary slackness condition in (6.3.2) automatically holds, i.e. $\sum_{i=1}^m |y_{j,i}^* h_i(x_j^*)| = 0$, where $y_{j,i}^*$ is the i -th element of y_j^* .

By Lemma 6.3.1 (replace x with $x_{\hat{j}}$), we have

$$\begin{aligned} \|x_{\hat{j}}^* - x_{\hat{j}}\|^2 &\leq \frac{2}{\underline{L}_f} \left[f(x_{\hat{j}}; x_{\hat{j}-1}) - f(x_{\hat{j}}^*; x_{\hat{j}-1}) + \langle y_{\hat{j}}^*, h(x_{\hat{j}}) \rangle \right] \\ &\leq \frac{2}{\underline{L}_f} (\delta^f + B\delta^h). \end{aligned} \quad (6.17)$$

Replacing x with $x_{\hat{j}-1}$ in Lemma 6.3.1 and using the relations that $f(x_{\hat{j}}; x_{\hat{j}-1}) - f(x_{\hat{j}}^*; x_{\hat{j}-1}) \leq \delta^f$ as well as $h(x_{\hat{j}}) \leq \delta^h$, we obtain

$$\begin{aligned} \frac{\underline{L}_f}{2} \|x_{\hat{j}}^* - x_{\hat{j}-1}\|^2 &\leq f(x_{\hat{j}-1}) - f(x_{\hat{j}}^*; x_{\hat{j}-1}) + \langle y_{\hat{j}}^*, h(x_{\hat{j}-1}) \rangle \\ &= f(x_{\hat{j}-1}) - f(x_{\hat{j}}) - \underline{L}_f \|x_{\hat{j}} - x_{\hat{j}-1}\|^2 + \delta^f + B\delta^h \\ &\leq f(x_{\hat{j}-1}) - f(x_{\hat{j}}) + \delta^f + B\delta^h \\ &\leq \frac{1}{J} \sum_{j=1, \dots, J} f(x_{j-1}) - f(x_j) + \delta^f + B\delta^h \\ &= \frac{1}{J} [f(x_0) - f(x_J)] + \delta^f + B\delta^h, \end{aligned} \quad (6.18)$$

where the third inequality follows from the selection of \hat{j} in Algorithm 8. Using the above

equality, and the KKT condition applied for (6.2), we arrive at

$$\begin{aligned} \left[d \left(\nabla f(x_{\hat{j}}^*) + \sum_{i=1}^m y_{\hat{j}}^{*i} \nabla h_i(x_{\hat{j}}^*), -N_X(x_{\hat{j}}^*) \right) \right]^2 &= 4\underline{L}_f^2 \|x_{\hat{j}}^* - x_{\hat{j}-1}\|^2 \\ &\leq \frac{8\underline{L}_f}{J} [f(x_0) - f(x_J)] + \delta^f + B\delta^h. \end{aligned} \tag{6.19}$$

Combining (6.17) and (6.19), we reach the conclusion that $x_{\hat{j}}$ is an $(\epsilon'_J, \epsilon_J)$ -KKT point of problem (5.2). Consequently, given precision ϵ , combining the result in Theorem 5.5.1, the overall iteration complexity of IPP-LCG solving for an (ϵ, ϵ) -KKT point is bounded by $\mathcal{O}(\frac{1}{\epsilon^3} \log(\frac{1}{\epsilon}))$. \square

Note that given a target accuracy $\epsilon > 0$, δ^f and δ^h can be selected as in the order of ϵ such that $\delta^f = \mathcal{O}(\epsilon)$ and $\delta^h = \mathcal{O}(\epsilon)$. Moreover, the iteration number J can be fixed to $\mathcal{O}(1/\epsilon)$.

It is worth noting that IPP-LCG can also be generalized to solve structured nonsmooth problems by applying LCG on the nonsmooth convex subproblems. The convergence analysis in this case is more or less the same as that shown in Theorem 6.3.2.

Observe that IPP-LCG is a triple-layer algorithm as we add an extra proximal point approximation loop on top of LCG, which already contains one inner oracle and an outer loop. Hence, it is not very convenient to implement this algorithm. In the next subsection, we present a more concise and easily implementable algorithm to solve problem (5.2) in the nonconvex setting.

6.4 Direct Nonconvex Conditional Gradients Method

To tackle the nonconvex functional constrained optimization (5.2), one alternative is to solve its Lagrangian dual given by

$$\min_{x \in X} \max_{y \in \mathbb{R}_+^m} \{F(x, y) := f(x) + \sum_{i=1}^m y_i h_i(x)\}. \quad (6.20)$$

In general, F in (6.20) is nonsmooth in x and can be approximated by a smooth function

$$\tilde{F}(x) := f(x) + \sum_{i=1}^m y_i(x) h_i(x) - \frac{c}{2} \|y(x)\|^2, \text{ with } y(x) := \arg \max_{y(x) \in \mathbb{R}_+^m} \tilde{F}(x) \quad (6.21)$$

as shown in Lemma 6.4.1 below.

Lemma 6.4.1. $\tilde{F}(\cdot)$ is a smooth function such that $\|\nabla \tilde{F}(x_1) - \nabla \tilde{F}(x_2)\| \leq L_c \|x_1 - x_2\|$, $\forall x_1, x_2 \in X$, where $L_c := L_f + \frac{\|M_h\| \|L_h\| D_X}{c} + \frac{\|M_h\|^2}{c}$, $L_h := (L_{h_1}, \dots, L_{h_m})$, $M_h := (M_{h_1}, \dots, M_{h_m})$ and $c > 0$.

Proof. Applying the first order optimality on $\max_{y \in \mathbb{R}_+^m} f(x) + \langle h(x), y \rangle - \frac{c}{2} \|y\|^2$ at y_{k-1} and y_k , respectively, we obtain $\forall y \in \mathbb{R}_+^m$,

$$\langle h(x_{k-1}) - cy_{k-1}, y - y_{k-1} \rangle \leq 0, \quad (6.22)$$

$$\langle h(x_k) - cy_k, y - y_k \rangle \leq 0. \quad (6.23)$$

Furthermore,

$$\begin{aligned} c \|y_{k-1} - y_k\|^2 &\leq \langle h(x_{k-1}) - h(x_k), y_{k-1} - y_k \rangle \\ &\leq \|h(x_{k-1}) - h(x_k)\| \|y_{k-1} - y_k\| \\ &\leq \|M_h\| \|x_{k-1} - x_k\| \|y_{k-1} - y_k\|, \end{aligned}$$

where the first inequality is by summing up the two inequalities above with y replaced by

y_k in (6.22) and y_{k-1} in (6.23); the second inequality follows from the Cauchy Schwarz inequality; the third one follows by the Lipschitz continuity of h . This gives

$$\|y_{k-1} - y_k\| \leq \frac{1}{c} \|M_h\| \|x_{k-1} - x_k\|.$$

Next, we derive a bound for $\|y_k\|$. Suppose x^* is an optimal solution of (5.2), then $h(x^*) \leq 0$. By (6.27) and the Lipschitz continuity of $h(\cdot)$, we have

$$\|y_k\| \leq \left\| \max \left\{ \frac{h(x_k) - h(x^*)}{c}, 0 \right\} \right\| \leq \frac{\|h(x_k) - h(x^*)\|}{c} \leq \frac{\|M_h\| \|x_k - x^*\|}{c} \leq \frac{\|M_h\| D_X}{c}.$$

Using the above inequality, the smoothness of f and h , we arrive at

$$\begin{aligned} & \|\nabla \tilde{F}(x_k) - \nabla \tilde{F}(x_{k-1})\| \\ &= \|\nabla f(x_k) - \nabla f(x_{k-1}) + \langle y_k, \nabla h(x_k) \rangle - \langle y_k, \nabla h(x_{k-1}) \rangle + \langle y_k - y_{k-1}, \nabla h(x_{k-1}) \rangle\| \\ &\leq \left(L_f + y_k^\top L_h + \frac{\|M_h\|^2}{c} \right) \|x_{k-1} - x_k\| \\ &\leq \left(L_f + \frac{\|M_h\| \|L_h\| D_X}{c} + \frac{\|M_h\|^2}{c} \right) \|x_{k-1} - x_k\|. \end{aligned}$$

□

Note that, given x , we can obtain the closed form solution of $y(x)$ in (6.21) such that $y(x) = \max \left\{ \frac{h(x)}{c}, 0 \right\}$. The DNCG method (detailed in Algorithm 9) directly applies the conditional gradient method on the following approximation problem:

$$\min_{x \in X} \tilde{F}(x). \tag{6.24}$$

More specifically, DNCG takes x_0 as input and calculates y_0 using the closed form. Then in each iteration it computes the primal solution x_k by calling the linear optimization oracle in (6.25) and performing convex combination in (6.26). Finally it updates the dual solution y_k in (6.27).

Algorithm 9 Direct Nonconvex Conditional Gradient Method (DNCG)

Inputs: $c > 0$.

Initialization: $x_0 \in X, y_0 = \max \left\{ \frac{h(x_0)}{c}, 0 \right\}$.

for $k = 1, 2, \dots, K$ **do**

$$p_k = \arg \min_{x \in X} \langle \nabla \tilde{F}(x_{k-1}), x - x_{k-1} \rangle, \quad (6.25)$$

$$x_k = (1 - \alpha_k)x_{k-1} + \alpha_k p_k, \quad (6.26)$$

$$y_k = \max \left\{ \frac{h(x_k)}{c}, 0 \right\}. \quad (6.27)$$

end for

To evaluate the efficiency of the DNCG method applied on problem (5.2) at $\bar{x} \in X$, we use the following error measures:

Definition 6.4.1. *Given a target accuracy $\epsilon > 0$, $\bar{x} \in X$ is an ϵ -Wolfe point if*

$$\begin{aligned} Q(\bar{x}) &:= \max_{x \in X} \langle \nabla \tilde{F}(\bar{x}), \bar{x} - x \rangle \leq \epsilon, \\ \|[h(\bar{x})]_+\|^2 &\leq \epsilon. \end{aligned} \quad (6.28)$$

The function $Q(\bar{x})$ in (6.28), often referred to as the Wolfe gap in projection-free methods, corresponds to the first-order optimality condition of problem (6.24). This explains why we call $\bar{x} \in X$ satisfying (6.28) an ϵ -Wolfe point. By the definition of \tilde{F} , we have $\nabla \tilde{F}(\bar{x}) = \nabla f(\bar{x}) + \sum_{i=1}^m y_i(\bar{x}) \nabla h_i(\bar{x})$. Hence this first criterion in (6.28) also tells us how the stationarity of the KKT condition of problem (5.2) is satisfied for a given pair of primal and dual solution $(\bar{x}, y(\bar{x})) \in X \times \mathbb{R}_+^m$. The second criteria characterizes the constraint violation at \bar{x} . Note that the ϵ -Wolfe point defined above provides no guarantee of complementary slackness for the KKT condition of (5.2).

It is worth pointing out here the relationship between the convergence criteria used by DNCG (see (6.28)) and the one by IPP-LCG (see (6.13)). If x' is an ϵ -KKT point (see Definition 6.3.2 (ii).), then $h(x') \leq 0$, so that $\|[h(x')]_+\|^2 \leq 0$, which implies the second condition of the ϵ -Wolfe point. For some $y \geq 0$, let $r = \nabla f(x') + \sum_{i=1}^m y_i \nabla h_i(x')$. Since $[d(r, -N_X(x'))]^2 \leq \epsilon$, then we can find some $g \in -N_X(x')$ such that $\|g - r\|^2 = \epsilon$ and

$\langle g, x - x' \rangle \geq 0, \forall x \in X$. Consequently, $\forall x \in X$,

$$\langle g - r, x' - x \rangle + \langle r, x' - x \rangle \leq 0. \quad (6.29)$$

Let $z' \in \arg \max_{x \in X} \langle r, x' - x \rangle$. From (6.29), we have

$$\max_{x \in X} \langle r, x' - x \rangle \leq \langle g - r, z' - x' \rangle. \quad (6.30)$$

Taking the square of both sides in (6.30), we obtain

$$\begin{aligned} \left(\max_{x \in X} \langle r, x' - x \rangle \right)^2 &\leq \|g - r\|^2 \|z' - x'\|^2 \\ &\leq \epsilon D_X^2. \end{aligned} \quad (6.31)$$

The result in (6.31) implies that $(Q(x'))^2 \leq \epsilon D_X^2$, and thus $Q(x') \leq \sqrt{\epsilon} D_X$.

Note that an ϵ -KKT point inherits complimentary slackness, which is not a condition for an ϵ -constrained Wolfe point. However, such ϵ -KKT point is not explicitly computed by IPP-LCG and it is only used to be measured against the output solution under some distance, while an ϵ -Wolfe point directly associates with the computed solution of DNCG.

We are now ready to analyze the convergence rate of the DNCG algorithm based on the criteria in (6.28).

Theorem 6.4.1. *The total number of iterations required to compute an approximate solution \bar{x} such that $Q(\bar{x}) \leq \epsilon$ and $\|[h(\bar{x})]_+\|^2 \leq \epsilon$ is bounded by $\mathcal{O}(1/\epsilon^4)$.*

Proof. Proof. Suppose $\{x_k\}$ is generated by Algorithm 9. Let $\tilde{F}^* := \min_{x \in X} \tilde{F}(x)$, $\hat{k} := \arg \min_{0 \leq k \leq K-1} Q(x_k)$, $c = \frac{1}{K^{1/4}}$ and $\alpha_k = \frac{1}{\sqrt{K}}$, where K is a known priori. By Lemma 6.4.1, we have

$$\tilde{F}(x_k) - \tilde{F}(x_{k-1}) \leq \langle \nabla \tilde{F}(x_{k-1}), x_k - x_{k-1} \rangle + \left(L_f + \frac{\|M_h\| \|L_h\| D_X}{c} + \frac{\|M_h\|^2}{c} \right) \|x_{k-1} - x_k\|^2.$$

Since by the definition of $Q(\cdot)$ and (6.26),

$$\langle \nabla \tilde{F}(x_{k-1}), x_k - x_{k-1} \rangle = \alpha_k \langle \nabla \tilde{F}(x_{k-1}), p_k - x_{k-1} \rangle = -\alpha_k Q(x_{k-1}),$$

then we have

$$\alpha_k Q(x_{k-1}) \leq -\tilde{F}(x_k) + \tilde{F}(x_{k-1}) + \left(L_f + \frac{\|M_h\| \|L_h\| D_X}{c} + \frac{\|M_h\|^2}{c} \right) \alpha_k^2 \|x_{k-1} - p_k\|^2.$$

Summing up the above inequality from $k = 1$ to K and using the fact that $\tilde{F}^* \leq \tilde{F}(x_K)$ result in

$$\left(\sum_{k=1}^K \alpha_k \right) \min_{1 \leq k \leq K} Q(x_{k-1}) \leq \tilde{F}(x_0) - \tilde{F}^* + \left(L_f + \frac{\|M_h\| \|L_h\| D_X}{c} + \frac{\|M_h\|^2}{c} \right) D_X^2 \sum_{1 \leq k \leq K} \alpha_k^2. \quad (6.32)$$

Dividing both sides of (6.32) by $\sum_{k=1}^K \alpha_k$, we obtain

$$Q(x_{\hat{k}}) \leq \frac{1}{\sqrt{K}} \left[\tilde{F}(x_0) - \tilde{F}^* + \frac{L_f}{2} D_X^2 \right] + \frac{1}{K^{1/4}} \left(\frac{\|M_h\|^2 D_X^2}{2} + \frac{\|M_h\| \|L_h\| D_X^3}{2} \right). \quad (6.33)$$

Next, we derive a bound for $\|[h(x_{\hat{k}})]_+\|^2$. Let y_{ki} be the i -th element of the vector y_k at iteration $k = 1, \dots, K$. Note first, if $h(x_{\hat{k}}) \leq 0$, then $\|[h(x_{\hat{k}})]_+\|^2 = 0$. The analysis below focuses on the case where $h(x_{\hat{k}}) > 0$. Consequently, by (6.27), we have $y_{\hat{k}i} = \frac{h(x_{\hat{k}})}{c}$, $i = 1, \dots, m$ and

$$\sum_{i=1}^m y_{\hat{k}i} h_i(x_{\hat{k}}) = \sum_{i=1}^m \frac{1}{c} (h_i(x_{\hat{k}}))^2. \quad (6.34)$$

Note also, using the Lipschitz continuity and the lower curvature property of f , it can be

easily verified that

$$\langle \nabla f(y), x - y \rangle \leq \frac{L_f}{2} D_X^2 + M_f D_X, \forall x, y \in X. \quad (6.35)$$

Suppose x^* is the optimal solution of (5.2), then $h(x^*) \leq 0$. By convexity of $h(\cdot)$ and the definition of $Q(x_{\hat{k}})$ in (6.28), we obtain

$$\begin{aligned} \sum_{i=1}^m y_{\hat{k}i} h_i(x_{\hat{k}}) &\leq \sum_{i=1}^m y_{\hat{k}i} (h_i(x^*) + \langle \nabla h_i(x_{\hat{k}}), x_{\hat{k}} - x^* \rangle) \\ &\leq \sum_{i=1}^m y_{\hat{k}i} \langle \nabla h_i(x_{\hat{k}}), x_{\hat{k}} - x^* \rangle \\ &\leq Q(x_{\hat{k}}) + \langle \nabla f(x_{\hat{k}}), x^* - x_{\hat{k}} \rangle \\ &\leq Q(x_{\hat{k}}) + \frac{L_f}{2} D_X^2 + M_f D_X, \end{aligned} \quad (6.36)$$

where the first inequality is because of the convexity of $h(\cdot)$, the second inequality is due to $h(x^*) \leq 0$ and $y_{\hat{k}} \geq 0$, the third inequality is by the definition of $Q(x_{\hat{k}})$ and the last inequality follows from (6.35). Combining (6.34) and (6.36), we have

$$\begin{aligned} \|[h(x_{\hat{k}})]_+\|^2 &\leq c \left(Q(x_{\hat{k}}) + \frac{L_f}{2} \|x^* - x_{\hat{k}}\|^2 + f(x^*) - f(x_{\hat{k}}) \right) \\ &\leq \frac{1}{K^{1/4}} \left(Q(x_{\hat{k}}) + \frac{L_f}{2} D_X^2 + M_f D_X \right), \end{aligned}$$

which implies

$$\|[h(x_{\hat{k}})]_+\|^2 \leq \frac{1}{K^{3/4}} \left[\tilde{F}(x_0) - \tilde{F}^* + \frac{L_f}{2} D_X^2 \right] + \frac{1}{\sqrt{K}} \left(\frac{\|M_h\|^2 D_X^2}{2} + \frac{\|M_h\| \|L_h\| D_X^3}{2} \right) + \frac{1}{K^{1/4}} \frac{L_f}{2} D_X^2. \quad (6.37)$$

Combining (6.33) and (6.37), given target accuracy $\epsilon > 0$, the iteration complexity of DNCG of solving for \bar{x} such that $Q(\bar{x}) \leq \epsilon$ and $\|[h(\bar{x})]_+\|^2 \leq \epsilon$ is bounded by $\mathcal{O}(1/\epsilon^4)$.

□

Remark 6.4.1. *In establishing the convergence rate of the DNCG method, we assume that*

$h(\cdot)$ is a smooth function. Consider now when $h(\cdot)$ is nonsmooth and inherits special structure as described in (5.40). Similar to Appendix 5.4.2, we can apply Nesterov smoothing scheme and construct $\{h_{i,\eta_i}\}$ such as

$$h_{i,\eta_i}(x) := \max_{z \in Z_i} \left\{ \langle B_i x, z \rangle - \hat{h}_i(z) - \eta_i U_i(z) \right\}, i = 1, \dots, \bar{m}. \quad (6.38)$$

In this way, $h_{i,\eta_i}(x)$ is a L_{h_i,η_i} -smooth function with $L_{h_i,\eta_i} = \frac{\|B_i\|^2}{\omega_i + \eta_i}$. We thereby define the gap function as $Q_\eta(\bar{x}) := \max_{x \in X} \langle \nabla \tilde{F}_\eta(\bar{x}), \bar{x} - x \rangle$, where $\tilde{F}_\eta(x) = f(x) + \sum_{i=1}^m y_i h_{i,\eta_i}(x) - \frac{c}{2} \|y\|^2$. Let $\eta_i = \frac{\|M_h\| D_X^3}{K^{1/8}}$, $\forall i = 1, \dots, m$. Then by Theorem 6.4.1, we have that $Q_\eta(\bar{x})$ is upper bounded by $\mathcal{O}(1/K^8)$. By the second relation in (5.41), we have that $\|[h(x_{\hat{k}})]_+\|^2$ is upper bounded by $\mathcal{O}(1/K^4)$.

We can also generalize DNCG to solve problems with nonsmooth $h(\cdot)$ by using the Nesterov smoothing scheme.

6.5 Numerical Experiments

In this section, we demonstrate the efficiency of the proposed algorithms (LCG, IPP-LCG and DNCG) in two important applications: portfolio selection and the intensity modulated radiation therapy (IMRT) treatment planning. Numerical comparison with CoexDurCG are also provided. All experiments are run using Python 3.8.5 under the Ubuntu 20.04.1 LTS operating system with a 4.20 GHz Intel Core i7 processor and 32Gb RAM.

6.5.1 Portfolio Selection

In this section, we first introduce the portfolio selection problem with and without cardinality constraint, and then apply LCG, IPP-LCG and DNCG to solve the formulated convex and nonconvex models using the real-world stock market dataset.

Models

Consider selecting portfolio among N risky assets with random return $r_i, i = 1, \dots, N$ and random target return R (a.k.a. market index). Let x_i be the decision variable that determines the weight of the i -th asset to be chosen, $i = 1, \dots, N$, such that $\sum_{i=1}^N x_i \leq 1$. The goal is to minimize the risk that the overall return is below the target return in expectation, i.e., $\mathbb{E} \left[\mathbb{1} \left\{ R - \sum_{i=1}^N r_i x_i > 0 \right\} \right]$, or, equivalently, $P \left(R - \sum_{i=1}^N r_i x_i > 0 \right)$, where $\mathbb{1}\{x > 0\} = 1$ if $x > 0$ and 0 otherwise. Given K samples of R and $r_i, i = 1, \dots, N$, the sample average of the risk can be written as

$$\frac{1}{K} \sum_{k=1}^K \mathbb{1} \left\{ R_k - \sum_{i=1}^N r_{ik} x_i > 0 \right\}. \quad (6.39)$$

Cardinality-free Models. Let the function in (6.39) be the objective function. We can formulate the following cardinality-free *nonconvex* model:

$$\begin{aligned} \min_x f(x) &:= \frac{1}{K} \sum_{k=1}^K \mathbb{1} \left\{ R_k - \sum_{i=1}^N r_{ik} x_i > 0 \right\} \\ \text{s.t.} & \sum_{i=1}^N x_i \leq 1, \\ & x_i \geq 0, \quad i = 1, \dots, N. \end{aligned} \quad (\text{Card-Free-Nonconvex})$$

The objective function in model (Card-Free-Nonconvex) is a step function, which is discontinuous and nonconvex. To implement the proposed algorithms solving the model (Card-Free-Nonconvex), we employ a nonconvex smooth approximation of $f(x)$ parameterized with θ such as $\tilde{f}_\theta(x) = \frac{1}{K} \sum_{k=1}^K \frac{1}{1 + \exp\{(-R_k + \sum_{i \in I} r_{ik} z_i)/\theta\}}$. Clearly, $\tilde{f}_\theta \rightarrow f$ when $\theta \rightarrow 0$.

Now we discuss how to approximate the nonconvex model (Card-Free-Nonconvex) using the convex formulation. Define $\phi(x) := [1 + x]_+$. Since for some $t > 0$, $\phi(tx) \geq 1$,

then for a random variable X , we have

$$\inf_{t>0} \mathbb{E}[\phi(tX)] \geq \mathbb{E}[\mathbb{1}_{\{X>0\}}]. \quad (6.40)$$

Consequently, instead of minimizing $\mathbb{E}[\mathbb{1}_{\{X>0\}}]$, we minimize its upper bound $\inf_{t>0} \mathbb{E}[\phi(tX)]$, which is equivalent to minimizing $\inf_{t>0} \mathbb{E}[\phi(tX)] - \alpha$, where α can be regarded as some confidence level such that $\alpha > 0$. Note that

$$\inf_{t>0} \mathbb{E}[\phi(tX)] - \alpha = \inf_{u \in \mathbb{R}} \{u + \alpha^{-1} \mathbb{E}[X - u]_+\}$$

and the minimum of the right-hand-side of the above inequality falls in $[\underline{u}, \bar{u}]$, where \underline{u} and \bar{u} are the respective left and right side $1 - \alpha$ quantile of the distribution of X . Coincidentally, $\inf_{u \in \mathbb{R}} \{u + \alpha^{-1} \mathbb{E}[X - u]_+\}$ is the Conditional-Value-at-Risk (CVaR) measure that is convex. Leveraging such approximation, we arrive at the convex approximation of the nonconvex model in (Card-Free-Convex).

Using the Conditional-Value-at-Risk approximation as described above, we can also transform the above nonconvex model into a convex one:

$$\begin{aligned} \min_{u,x} f(x, u) &:= u + \frac{1}{\alpha K} \sum_{k=1}^K \left[-u + R_k - \sum_{i=1}^N r_{ik} x_i \right]_+ \\ \text{s.t.} \quad &\sum_{i=1}^N x_i \leq 1, \\ &x_i \geq 0, \quad i = 1, \dots, N, \\ &\underline{u} \leq u \leq \bar{u}. \end{aligned} \quad (\text{Card-Free-Convex})$$

Cardinality-constrained Models. In practice, decision makers intend to select only a portion of the available assets due to restrictions arising from transaction costs, budget constraints, etc. Such cardinality requirement can be included using the sparsity constraint $\sum_{i=1}^N \mathbb{1}\{x_i > 0\} - \Psi \leq 0$, where Ψ is a given number of allowed selected assets. We can also

derive its convex approximation as

$$\sum_{i=1}^N v + \Psi^{-1}[x_i - v]_+ \leq 0, \underline{v} \leq v \leq \bar{v}. \quad (6.41)$$

For the cardinality-constrained portfolio selection problem, we develop the following convex and nonconvex models:

1. Incorporate (6.41) in (Card-Free-Convex) (Card-Convex)
2. Incorporate constraint (6.41) in (Card-Free-Nonconvex) (Card-Nonconvex-1)
3. Add a weighted objective term $\frac{1}{\Psi} \sum_{i=1}^N \mathbb{1}\{x_i > 0\}$ in (Card-Free-Nonconvex) (Card-Nonconvex-2)

In model (Card-Nonconvex-2), to make the objective function continuous, we use the smooth approximation function $\frac{1}{\Psi} \sum_{i=1}^N \frac{1}{1+\exp\{(-x_i)/\theta\}}$ to replace the step function in implementation.

Although the aforementioned convex and nonconvex models (with and without cardinality constraint) have different objective functions, the true objective is the risk in (6.39). Therefore, to evaluate the effectiveness and compare the performance of different algorithms, we focus on the value of (6.39) and the number of selected assets (mainly for cardinality-constrained models) computed by the algorithms.

Tests on Stock Market Dataset

To evaluate the proposed algorithms, we test the cardinality-free and cardinality-constrained models from Section 6.5.1 using the historical stock data from six major stock markets provided by *Thomson Reuters Datastream* and *Fama & French Data Library*. The dataset contains weekly returns $\{r_{ik}\}$ for N assets and market indices (target level) $\{R_k\}$ across K

weeks. In the original dataset, the value of Ψ for the cardinality constraint is not available. We construct it by the following rule: $\Psi = \lfloor 0.2 * N \rfloor$ if $N \leq 100$ and $\Psi = \lfloor 0.05 * N \rfloor$ if $N > 100$. Table 6.1 lists some key information about the six datasets and we refer to [121] for more details.

Table 6.1: Features of the stock market dataset.

Instance	Description	# of assets (N)	# of weeks (K)	Cardinality (Ψ)
DJ	Dow Jones Industrial Average (USA)	28	1363	5
FF49	Fama and French 49 Industry (USA)	49	2325	9
ND100	NASDAQ 100 (USA)	82	596	16
FTSE100	FTSE 100 (UK)	83	717	16
SP500	S&P 500 (USA)	442	595	22
NDComp	NASDAQ Composite (USA)	1203	685	60

In all experiments, to implement the proposed algorithms, we employ a nonconvex smooth approximation of the step functions in the objective of (Card-Free-Nonconvex) and (Card-Nonconvex-2) (see Appendix ??). The initial values of x_i are set to zero. In the tables below, we use the following notations: (1) “ $f(x_N)$ ” stands for the objective value and “ $\|h(x_N)\|_2$ ” for the norm of the cardinality constraint violation ($|\sum_{i=1}^N x_i - \Psi|$); (2) “Risk” is the value in (6.39) (it is the same as $f(x_N)$ in the nonconvex models); (3) “# ass.” represents the number of selected assets, i.e. the number of $\{x_i\}$ that are nonzero; (4) “Card vio.” records the values of cardinality violation such that Card. vio. = $\max(\# \text{ ass.} - \Psi, 0)$; (5) “Time(s)” is the CPU time in seconds.

Results of Cardinality-free Models. Table 6.2 and Table 6.3 report the computational results of applying LCG, DNCG and IPP-LCG to solve the cardinality-free models. All algorithms are terminated when the number of iteration reaches 100. In particular, for LCG and IPP-LCG, the number of iterations is the total iterations to run CGO. To examine the sparsity of the solutions, we record the values of cardinality violation, although no cardinality control is imposed in this case. From these tables, we observe that LCG, DNCG and IPP-LCG solve the models efficiently and yield relatively small risk. Without cardinality

control, however, the numbers of selected assets returned by all these algorithms are larger than the required cardinality Ψ .

Table 6.2: Results of solving model (Card-Free-Convex) by LCG.

Instance	LCG				
	$f(x_N)$	Risk	# ass.	Card. vio.	Time (s)
DJ	0.0102	0.0168	28	23	0.0262
FF49	0.0021	0.0082	45	36	0.0485
ND100	0.0057	0.0184	51	35	0.023
FTSE100	0.0063	0.0181	46	30	0.025
SP500	0.0063	0.0185	66	44	0.0412
NDComp	0.0162	0.0292	83	23	0.141

Table 6.3: Results of solving model (Card-Free-Nonconvex) by DNCG and IPP-LCG.

Instance	DNCG				IPP-LCG			
	Risk($f(x_N)$)	# ass.	Card. vio.	Time (s)	Risk($f(x_N)$)	# ass.	Card. vio.	Time (s)
DJ	0.019	27	22	0.0159	0.0183	27	22	0.0383
FF49	0.0077	48	39	0.0367	0.0082	43	34	0.0588
ND100	0.0167	48	32	0.0179	0.0184	51	35	0.0165
FTSE100	0.0139	50	34	0.0161	0.0153	48	32	0.0255
SP500	0.0151	63	41	0.0322	0.0067	81	59	0.047
NDComp	0.0204	78	18	0.1557	0.0219	72	12	0.186

Results of Cardinality-constrained Models. For the developed cardinality-constrained models in Section 6.5.1, we solve them by LCG, DNCG and IPP-LCG, respectively, and report the numerical results in Table 6.4 - 6.6 accordingly. Intuitively, when restricting to a small pool of assets, the risk of not reaching the market index increases. As shown in Table 6.4 - 6.6, the risk (6.39) is higher than the one from the cardinality-free models (see Table 6.2 - 6.3). Nevertheless, with the cardinality constraint, all algorithms select a much smaller number of assets in a similarly efficient manner. Comparing all three algorithms, the nonconvex methods meet the cardinality requirement more strictly than their convex counterpart while the DNCG method applied on model (Card-Nonconvex-2) meet the cardinality requirement for all the instances and consumes the least CPU time for most of the

instances.

Table 6.4: Results of solving model (Card-Convex) by LCG.

Instance	LCG					
	$f(x_N)$	$\ h(x_N)\ _2$	Risk	# ass.	Card. vio.	Time (s)
DJ	0.0429	0.1029	0.1056	10	5	0.0467
FF49	0.0445	0.0717	0.248	13	4	0.0758
ND100	0.0306	0.0423	0.1208	22	6	0.0578
FTSE100	0.0223	0.0344	0.0586	22	6	0.0423
SP500	0.0253	0.0296	0.1076	28	6	0.0656
NDComp	0.0257	0	0.0175	60	0	0.266

Table 6.5: Results of solving model (Card-Nonconvex-1) by IPP-LCG.

Instance	IPP-LCG				
	Risk($f(x_N)$)	$\ h(x_N)\ _2$	# ass.	Card. vio.	Time (s)
DJ	0.1012	0.0844	8	3	0.0509
FF49	0.228	0.0559	12	3	0.0801
ND100	0.0586	0.0264	20	4	0.059
FTSE100	0.1158	0.0401	20	4	0.0443
SP500	0.1042	0.018	28	6	0.0777
NDComp	0.0146	0	59	0	0.349

Table 6.6: Results of solving model (Card-Nonconvex-2) by DNCG.

Instance	DNCG			
	Risk($f(x_N)$)	# ass.	Card. vio.	Time (s)
DJ	0.1071	5	0	0.0293
FF49	0.206	8	0	0.0361
ND100	0.0872	15	0	0.0174
FTSE100	0.0516	16	0	0.0202
SP500	0.0756	21	0	0.0547
NDComp	0.0365	58	0	0.326

Finally, we apply the CoexDurCG algorithm proposed in [31] to solve model (Card-Convex) and report the results in Table 6.7. Compared with LCG (see Table 6.4), for instances with smaller asset pool such as “DJ”, “FF49” and “ND100”, CoexDurCG returns higher risk and selects more assets than LCG; for instances with larger pool such as

“FTSE100” “SP500” and “NDComp”, CoexDurCG produces sparser solutions with less cardinality violation but the computed risk is higher and consumes more CPU time.

Table 6.7: Results of solving model (Card-Convex) by CoexDurCG.

Instance	CoexDur CG						
	Iter.	$f(x_N)$	$\ h(x_N)\ _2$	Risk	# ass.	Card. vio.	Time (s)
DJ	100	0.0569	0.0559	0.1079	13	8	0.0443
FF49	100	0.0483	0.0716	0.275	11	2	0.1189
ND100	100	0.0473	0.0295	0.1644	22	6	0.0444
FTSE100	100	0.0425	0.0198	0.1074	12	0	0.06658
	150	0.0379	0.0159	0.09903	16	0	0.08253
SP500	100	0.0396	0.0088	0.1143	9	0	0.0693
	500	0.027	0.0072	0.0891	17	0	0.4889
NDComp	100	0.0364	0.0017	0.0788	26	0	0.162
	500	0.0211	0.0021	0.0365	43	0	0.6148
	1000	0.0169	0.0018	0.0219	57	0	0.968

6.5.2 IMRT Treatment Planning

In this section, we first overview the IMRT treatment planning problem and formulate it as convex or nonconvex models. We then test the performance of LCG and DNCG for solving these models on four randomly generated data instances and one real-world dataset obtained from the Prostate database (<https://github.com/cerr/CERR/wiki>).

Models

During the radiation therapy treatment, a patient receives prescribed radiation doses from a linear accelerator (linac), which is comprised of a set of *angles* ($a \in A$) and in each angle, different *apertures* ($e \in E_a$) can be formed to determine the doses intensity. The decisions of the treatment planning problem consist of the selection of a set of angles and apertures as well as the determination of the doses intensity, in an effort to deliver a certain level of radiation to the tumor tissues and avoid overdoses on the healthy ones. To elaborate, for each patients, the target body structures are discretized into small voxels v and the

collection of all voxels is denoted by \mathcal{V} . In the linac, each angle $a \in A$ contains rectangular grids of beamlet $(l, r), l = 1, \dots, m, r = 1, \dots, n$, which can stay active or blocked. An aperture $e \in E_a$ of an angle a is then determined by the status of the beamlets. A set of binary variables $\{x_{l,r}^{a,e}\}$ are created to decide the shape of the aperture e from angle a . Specifically, $x_{l,r}^{a,e} = 1$ if beamlet (l, r) is active, and $x_{l,r}^{a,e} = 0$ if beamlet (l, r) is blocked. An additional set of variables $\{y_{a,e}\}$ are created to decide the intensity rate of the selected aperture e , where $e \in E_a$. The unit intensity delivered to voxel v from beamlet (l, r) is denoted by $D_{(l,r)v}$ in Gy. Then the total amount of radiation received by voxel v is

$$z_v = \sum_{a \in A} \sum_{e \in E_a} \sum_{l=1}^m \sum_{r=1}^n R D_{(l,r)v} x_{l,r}^{a,e} y_{a,e}, \quad \forall v \in \mathcal{V}.$$

We use k to index the underdose/overdose clinical criteria, where $k \in K_u (k \in K_o)$ denote the underdose (overdose) criterion and S_k to denote the set of structures in criterion k , where $S_k \subset \mathcal{V}, k \in K_u \cup K_o$. Additionally, we denote the number of voxels in \mathcal{V} by N_v , the number of voxel in S_k by N_k and the required quantile of criterion k by p_k .

A desirable treatment plan operates only on a small number of angles in order to reduce the operation time. To serve the purpose, a group sparsity constraint (parameterized on $\Phi > 0$) is included as proposed in [31] in the optimization model:

$$\sum_{a \in A} \max_{e \in E_a} y_{a,e} \leq \Phi, \quad (6.42)$$

where $y_{a,e}$ are decision variables of intensity rate of the selected aperture e . Moreover, it is crucial to satisfy the required clinical criteria on particular body structures (mathematically discretized into small voxels). For instance, in the Prostate dataset,

- underdose criteria “PTV68: $V68 \geq 95\%$ ”: the percentage of voxels in structure PTV68 that receive at least 68 Gy dose should be at least 95%;
- overdose criteria “PTV68: $V74.8 \leq 10\%$ ”: the percentage of voxels in structure

PTV68 that receive at least 74.8 Gy dose should not be over 10%.

A conventional way to model the clinical criteria is by risk averse constraints, in an attempt to avoid underdose (resp. overdose) to tumor (resp. healthy) structures. To be more precise, let X be the random variable that denotes the amount of radiation received by a randomly selected voxel in certain structure. For some properly chosen right hand side b , the underdose/overdose criteria can be modeled by

$$\sup\{\tau : P(X < \tau) \leq \alpha\} \geq b, \quad (\text{underdose}) \quad (6.43)$$

$$\inf\{\tau : P(X > \tau) \leq \alpha\} \leq b. \quad (\text{overdose}) \quad (6.44)$$

Convex Formulation. Note that the left hand side in both (6.43) and (6.44) are nonconvex and we use CVaR for approximation in the convex formulation stated below. Follow the description in [31], the convex model is adapted as follow:

$$\min f(z) := \frac{1}{N_v} \sum_{v \in \mathcal{V}} \underline{w}_v [\underline{T}_v - z_v]_+^2 + \bar{w}_v [z_v - \bar{T}_v]_+^2 \quad (6.45)$$

$$\text{s.t.} \quad -\tau_k + \frac{1}{p_k N_k} \sum_{v \in S_k} [\tau_k - z_v]_+ \leq -b_k, \quad \forall k \in K_u, \quad (6.46)$$

$$\tau_k + \frac{1}{p_k N_k} \sum_{v \in S_k} [z_v - \tau_k]_+ \leq b_k, \quad \forall k \in K_o, \quad (6.47)$$

$$z_v = \sum_{a \in A} \sum_{e \in E_a} \sum_{l=1}^m \sum_{r=1}^n RD_{(l,r)v} x_{lr}^{a,e} y_{a,e}, \quad \forall v \in S_k, k \in K_u \cup K_o, \quad (6.48)$$

$$\sum_{a \in A} \max_{e \in E_a} y_{a,e} \leq \Phi, \quad (6.49)$$

$$\sum_{a \in A} \sum_{e \in E_a} y_{a,e} \leq 1, \quad (6.50)$$

$$y^{a,t} \geq 0, \quad (6.51)$$

$$\tau_k \leq \bar{\tau}_k, \quad k \in K_u \cup K_o, \quad (6.52)$$

$$\tau_k \geq \underline{\tau}_k, \quad k \in K_u \cup K_o. \quad (6.53)$$

The objective function $f(z)$ in (6.45), serving as a convex surrogate of the clinical criteria, penalizes underage and overage dose of a voxel with pre-defined threshold $\underline{T}_v, \bar{T}_v$ and weights $\underline{w}_v, \bar{w}_v$, where $[\cdot]_+$ denotes $\max(\cdot, 0)$. To reinforce the clinical criteria, constraints (6.46) and (6.47) are added. To solve the convex model by the LCG method, smoothing scheme (with entropy distance generating function) is applied on all nonsmooth functionals ((6.46),(6.47) and (6.49)), which includes construction of $\{h_{i,\eta_i}\}$ as indicated in Algorithm 5 for nonsmooth underdose/overdose constraints and the group sparsity constraint.

However, in this formulation, more decision variables (e.g. τ_k) and parameters (e.g. b_k) are needed to refine the approximation, to which the solutions could be very sensitive.

Nonconvex Formulation. To alleviate the side effects caused by the convex approximation, we attempt to formulate the objective function using (6.43) and (6.44) directly. Specifically, we minimize the weighted sample average of $P(X > \tau)$ for overdose criteria and $P(X < \tau)$ for underdose criteria. The nonconvex model uses the original clinical criteria in the objective while subjecting to the group sparse constraint (6.42). The exact nonconvex formulation is described as follow.

$$\begin{aligned}
\min f(z) &:= \sum_{k \in K_u} \frac{w_k}{N_k} \sum_{v \in S_k} \mathbb{1}_{\{z_v < \tau_k\}} + \sum_{k \in K_o} \frac{w_k}{N_k} \sum_{v \in S_k} \mathbb{1}_{\{z_v > \tau_k\}} \\
\text{s.t. } z_v &= \sum_{a \in A} \sum_{e \in E_a} \sum_{l=1}^m \sum_{r=1}^n RD_{(l,r)v} x_{lr}^{a,e} y_{a,e}, \quad \forall v \in S_k, k \in K_u \cup K_o, \\
&\sum_{a \in A} \max_{e \in E_a} y_{a,e} \leq \Phi, \\
&\sum_{a \in A} \sum_{e \in E_a} y_{a,e} \leq 1, \\
&y_{a,e} \geq 0.
\end{aligned} \tag{6.54}$$

Here $\{w_k\}$ is a set of weights for underdose and overdose objective terms; $f(\cdot)$ is a step function which is nonconvex and discontinuous; $\{\tau_k\}$ are parameters given by the clinical criteria, instead of decision variables to calibrate the approximation in the case of the

convex formulation. To solve the model by the proposed algorithm, we employ a sigmoid function (parameterized on θ) to approximate the original function. Specifically, for $k \in K_u$, the approximation reads

$$\tilde{f}_\theta^k(x) = \frac{1}{N_k} \sum_{v \in S_k} \frac{1}{1 + \exp\{(z_v - \tau_k)/\theta\}}. \quad (6.55)$$

Similarly, for $k \in K_o$,

$$\tilde{f}_\theta^k(x) = \frac{1}{N_k} \sum_{v \in S_k} \frac{1}{1 + \exp\{(-z_v + \tau_k)/\theta\}}. \quad (6.56)$$

Note that when $\theta \rightarrow 0$, $\tilde{f}_\theta \rightarrow f$ and \tilde{f}_θ is nonconvex.

Tests on Synthetic Dataset

The synthetic dataset used in Section 6.5 mimics the IMRT dataset of a real patient, with each containing information of (discretized) voxels, beamlet coordinates and corresponding unit intensity (D matrix) received by each voxel. In particular, each angle pairs with a D matrix with dimension of $\#$ of voxels \times $\#$ of beamlets, and there are 180 D matrices in total. In particular, the number of beamlets is determined by the discretization granularity (beamlet unit length).

Table 6.8 describes main features of each dataset, where ‘‘Granularity’’ stands for beamlet unit length. Instance 1 and 2 (resp. instance 3 and 4) are featured in lower (resp. higher) beamlet granularity and have the same set of D matrices and voxels. Higher discretization accuracy (e.g. 0.25) results in larger number of beamlets, thus in higher dimension of D matrix. Therefore, instance 3 and 4 are in larger scale than instance 1 and 2. Among all the voxels, we randomly select two sets of tumor tissues that require radiation therapy and treat the rest as the healthy ones. For the tumor issues, we consider two underdose and one overdose constraints.

We compare the performance of CoexDurCG and LCG applied on the convex formu-

Table 6.8: Features of the synthetic dataset

Instance	# of angels	# of voxels	# of beamlets	Accuracy	b_k	p_k
1	180	4096	100	1.0	[40, 50, 100]	[0.01, 0.01, 0.05]
2	180	4096	100	1.0	[50, 60, 80]	[0.01, 0.01, 0.01]
3	180	262144	2000	0.25	[40, 50, 100]	[0.01, 0.01, 0.05]
4	180	262144	2000	0.25	[50, 60, 80]	[0.01, 0.01, 0.01]

lation ($\Phi = 0.005$ in (6.42)) using the aforementioned synthetic datasets and report the results in Table 6.10. In this table, the primal variable is denoted by x_N and the vector of constraints by $h(x_N) = (h_s; h_c)$, including the CVaR constraints h_c (for clinical criteria) and the group sparsity constraint h_s . We see that both algorithms consume similar CPU time to run 1000 iterations. This is expected as they are projection-free type algorithms and not required to compute full gradients of potentially high-dimensional decision variables. Besides, over all instances, both algorithms return similar objective values. However, CoexDurCG returns a solution that results in larger constraint violation, especially in the clinical constraints.

We also provide numerical results of running LCG on various Φ . Table 6.9 displays the results of applying LCG to solve the convex formulation (6.45) - (6.53) with various Φ . From the table, we observe that regardless of the large difference in scale among the instances, the proposed algorithm LCG exhibits comparable performance in solving all instances in view of the objective value and constraint violation at iteration 1000. By comparing instance 1 and instance 2 (namely, instance 3 and instance 4), we see that the values of $\|h(x_N)\|_2$ in instance 1 (resp. instance 3) remain lower than those in instance 2 (resp. instance 4). Such results indicate that the satisfaction of the constraints are sensitive to the choice of (b_k, p_k) thus to the decision variable τ_k , which jointly determine the CVaR approximation. An additional observation is that when Φ decreases (i.e. sparsity requirement is more stringent), the violation of the group sparsity constraint increases, which is an expected effect of Φ .

Table 6.9: Results of applying LCG on the synthetic dataset at iteration 1000.

Instance	Φ	LCG				Time (s)
		$f(x_N)$	$\ h(x_N)\ _2$	$\ h_s\ _2$	$\ h_c\ _2$	
1	1.0	0.0136	0.319	0	0.319	901
	0.5	0.0142	0.326	0	0.326	914
	0.05	0.0156	0.449	0.302	0.332	948
	0.005	0.0193	0.528	0.421	0.319	924
	0.0005	0.0174	0.576	0.499	0.288	938
2	1.0	0.0156	0.626	0	0.626	916
	0.5	0.0161	0.628	0	0.628	923
	0.05	0.0197	0.702	0.291	0.639	942
	0.005	0.019	0.763	0.402	0.649	908
	0.0005	0.0142	0.815	0.476	0.662	949
3	1.0	0.0479	0.434	0	0.434	4678
	0.5	0.0466	0.436	0	0.436	4726
	0.05	0.0514	0.451	0.087	0.442	4685
	0.005	0.047	0.476	0.169	0.445	4834
	0.0005	0.048	0.493	0.188	0.456	4842
4	1.0	0.0421	0.919	0	0.919	4766
	0.5	0.0441	0.943	0	0.943	4762
	0.05	0.0498	0.969	0.068	0.967	4813
	0.005	0.0435	0.984	0.175	0.968	4871
	0.0005	0.0433	0.975	0.201	0.954	4772

Tests on Prostate Dataset

In this part, we conduct numerical experiments on a publicly available dataset of a patient with prostate cancer. The dataset has 3,047,040 voxels and 180 angles, with the granularity of beamlets grids (beamlet unit length) equal to 1.0 for each angle. The average number of beamlets is 155. As such, the dimension of the data matrices reaches more than $3,047,040 \times 155 \times 180$. More importantly, the dataset contains 10 clinical criteria for six structures: 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: $V50 \leq 1\%$. In our numerical study, the obtained solution is evaluated by whether it satisfies all above the clinical criteria.

This IMRT problem with Prostate dataset is notoriously difficult for the following chal-

Table 6.10: Results of applying CoexDurCG on synthetic data with $\Phi = 0.005$ at iteration 1000.

Instance	CoexDurCG					LCG				
	$f(x_N)$	$\ h(x_N)\ _2$	$\ h_s\ _2$	$\ h_c\ _2$	Time (s)	$f(x_N)$	$\ h(x_N)\ _2$	$\ h_s\ _2$	$\ h_c\ _2$	Time (s)
1	0.0193	0.984	0.641	0.747	926	0.0193	0.528	0.421	0.319	924
2	0.0166	1.643	0.614	1.524	996	0.019	0.763	0.402	0.649	908
3	0.0467	1.043	0.205	1.023	4889	0.047	0.476	0.169	0.445	4834
4	0.0465	3.193	0.208	3.186	4867	0.0435	0.984	0.175	0.968	4871

lenges.

First, the aforementioned clinical criteria inherit potential contradiction. For example, the tumor structure “PTV68” and the healthy structures “Bladder” and “Rectum” are very close, but it is required that at least 95% of the tumor structure receives no less than 68 Gy dose while strict percentage cap is placed on the dose received by the healthy ones.

Second, it is difficult to meet the underdose and overdose clinical criteria simultaneously for the “PTV68” structure. To see this, the difference between the upper dose limit (74.8 Gy) and the lower dose limit (68 Gy) is very close, which implies that one necessary condition to satisfy the underdose and overdose criteria is that at least 90% of the received dose should fall in $[68, 74.8]$.

Third, in order to shorten the operation time, we need to select small number of angles with no more than 100 apertures in total. Such requirements are potentially conflicting with accomplishing the target of dose delivery. Therefore, the model and algorithms should be designed to make smart trade-offs.

Last, the dimension of the data matrices are over $3 \text{ million} \times 155 \times 180$, leading to high dimensional decision space with potential size larger than 180×45^{10} , which is quite computationally cumbersome, and prevents any methods requiring full gradient computation.

Results of the Convex Formulation. We apply the LCG algorithm to solve the convex formulation. In Table 6.11, we summarize the treatment plan (number of angles, number

of apertures) constructed by LCG and the number of iterations needed to deliver the plan. From the table, we see that when Φ is smaller, the algorithm tends to select less angles, which is an expected effect of the group sparsity constraint.

Table 6.11: Treatment plans constructed by LCG on Prostate dataset with different Φ .

Φ	# of iter.	# of angles	# of apertures
1.0	100	27	99
0.5	85	17	84
0.005	63	6	62
0.0005	78	5	77

Table 6.12 details the fulfillment of the clinical criteria for different sparsity parameters Φ . Here in the table, each column (starting from the second one) represents the clinical criteria (criterion) of particular structure. For each of them (e.g. PTV68 / V68 \geq 95%), the first line (e.g. PTV68) indicates the treated structure; in the second line (and onwards), take “V68 \geq 95%” as an instance, it means that the percentage of voxels that receive at least 68 Gy dose (V68) should be no less than 95% (\geq 95%); in the instance of “V74.8 \leq 10%”, it means that the percentage of voxels that receive at least 74.8 Gy dose (V74.8) should be no larger than 10% (\leq 10%). In each line at each cell of the table, we record such voxel percentages correspondingly computed by the algorithm. In the case of “V68 \geq 95%”, when the recorded value is no less than 0.95, then the clinical criterion “PTV68 / V68 \geq 95%” is satisfied; in the case of “V74.8 \leq 10%”, when the recorded value is no larger than 0.1, then the clinical criterion “PTV68 / V74.8 \leq 10%” is satisfied.

From the displayed results in Table 6.12, when $\Phi = 0.005$, the algorithm returns fairly good solution in terms of satisfying all clinical criteria, except for the criterion PTV68: V74.8 \leq 10%. From the clinical perspective, when it is very difficult to satisfy all the clinical criteria (e.g., trade-offs between angle sparsity and clinical criteria satisfaction), priority is to given to the underdosing criteria of the tumor structures and the overdosing ones of the healthy structures so as to guarantee the overall effectiveness of the treatment. In our situation, the only restriction we can possibly relax is the overdose criterion of the

tumor structure PTV68: $V74.8 \leq 10\%$. According to the results in Table 6.12, when $\Phi = 0.005$, since the violation of this criterion does not affect the satisfaction of other criteria, the output solution still accounts for a desirable treatment plan with respect to the number of selected angles and the satisfaction of the clinical criteria when combining the results in Table 6.11.

Table 6.12: Results of applying LCG on Prostate dataset.

Φ	PTV56 $V56 \geq 95\%$	PTV68 $V68 \geq 95\%$ $V74.8 \leq 10\%$	Rectum $V30 \leq 80\%$ $V50 \leq 50\%$ $V65 \leq 25\%$	Bladder $V40 \leq 70\%$ $V65 \leq 30\%$	Lft. femoral head $V50 \leq 1\%$	Rht. femoral head $V50 \leq 1\%$
1.0	0.9997	0.9647 0.1593	0.6825 0.2188 0.0601	0.5365 0.2179	0.0011	0.001
0.5	0.9996	0.9536 0.1423	0.7126 0.2455 0.0533	0.5239 0.2115	0.01	0.0023
0.005	0.9987	0.9544 0.1263	0.7778 0.3786 0.0561	0.552 0.2287	0.0024	0.0
0.0005	0.9994	0.9056 0.1151	0.7998 0.3804 0.1071	0.6198 0.2556	0.0003	0.0022

Results of the Nonconvex Formulation. We apply the DNCG method to solve the non-convex formulation. In this case, the sparsity parameter Φ is set to be 0.005 provided that it demonstrates the best numerical performance in the convex case. We provide two types of results:

1. DNCG: run the algorithm on a set of trivially generated initial points.
2. LCG initial + DNCG: run DNCG with a set of initial solutions computed by LCG.

For implementation of “LCG initial + DNCG”, the starting point is obtained by solving the convex model with sparsity parameter $\Phi = 0.005$ at iteration 63 (see Table 6.11 and 6.12).

In this way, the initial solution is feasible in terms of satisfying all clinical criteria except for “PTV 68: $V74.8 \leq 10\%$ ”. Results of a treatment plan (selected angles/apertures) constructed by the proposed algorithms are shown in Table 6.13. In Table 6.14, we demonstrate the fulfillment of the clinical criteria by applying DNCG with two different initialization schemes as mentioned above. With the trivial initialization, the DNCG algorithm produces a solution that meets all criteria, even for the hard criterion “PTV 68: $V74.8 \leq 10\%$ ”. One downside is that it selects more angles and consumes more number of iterations, compared to the LCG algorithm. With warm-up initialization, DNCG selects less number of angles and requires less number of iterations while satisfying all clinical criteria.

Table 6.13: Treatment plans constructed by DNCG with different initial conditions on Prostate dataset.

Type	Φ	# of iter.	# of angels	# of apertures
DNCG	0.005	96	14	83
LCG initial + DNCG	0.005	63(convex)+17(nonconvex)	9	75

Table 6.14: Results of applying DNCG and LCG initial+ DNCG on Prostate dataset.

Type	Φ	PTV56 $V56 \geq 95\%$	PTV68 $V68 \geq 95\%$ $V74.8 \leq 10\%$	Rectum $V30 \leq 80\%$ $V50 \leq 50\%$ $V65 \leq 25\%$	Bladder $V40 \leq 70\%$ $V65 \leq 30\%$	Lft. femoral head $V50 \leq 1\%$	Rht. femoral head $V50 \leq 1\%$
DNCG	0.005	0.9522	0.9549 0.0104	0.7506 0.2772 0.0743	0.5299 0.2405	0.0012	0.0
LCG initial + DNCG	0.005	0.9571	0.9503 0.0126	0.7829 0.3844 0.0884	0.5411 0.2204	0.00067	0.0

REFERENCES

- [1] V. Guigues, A. Shapiro, and Y. Cheng, “Duality and sensitivity analysis of multistage linear stochastic programs,” *European Journal of Operational Research*, 2022.
- [2] A. Shapiro and Y. Cheng, “Dual bounds for periodical stochastic programs,” *Operations Research*, 2022.
- [3] A. Shapiro and Y. Cheng, “Central limit theorem and sample complexity of stationary stochastic programs,” *Operations Research Letters*, vol. 49, pp. 676–681, 2021.
- [4] V. Guigues, A. Shapiro, and Y. Cheng, “Risk-averse stochastic optimal control: An efficiently computable statistical upper bound,” *Operations Research Letters*, 2022.
- [5] Y. Cheng, G. Lan, and H. E. Romeijn, “Functional constrained optimization for risk aversion and sparsity control,” *arXiv preprint arXiv:2210.05108*, 2022.
- [6] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.
- [7] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, “Coherent measures of risk,” *Mathematical Finance*, vol. 9, pp. 203–228, 1999.
- [8] H. Föllmer and A. Schied, *Stochastic Finance: An Introduction in Discrete Time*, 2nd. Walter de Gruyter, Berlin, 2004.
- [9] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory*, third. Philadelphia: SIAM, 2021.
- [10] V. Guigues and W. Römisich, “Sampling-based decomposition methods for multistage stochastic programs based on extended polyhedral risk measures,” *SIAM Journal on Optimization*, vol. 22, pp. 286–312, 2 2012.
- [11] M. Shaked and J. Shanthikumar, *Stochastic Orders and their Applications*. Probability and Mathematical Statistics. Academic Press Inc., Boston, MA, 1994.
- [12] A. Muller and D. Stoyan, *Comparison Methods for Stochastic Models and Risks*. Wiley Series in Probability and Statistics. John Wiley Sons Ltd., Chichester, 2002.
- [13] G. Bayraksan and D. K. Love, “Data-driven stochastic programming using phidivergences,” *Tutorials in Operations Research, INFORMS*, pp. 1563–1581, 2015.

- [14] A. Ben-Tal and M. Teboulle, “Penalty functions and duality in stochastic programming via phi-divergence functionals,” *Mathematics of Operations Research*, vol. 12, pp. 224–240, 1987.
- [15] B.-T. A. and T. M., “An old-new concept of convex risk measures: The optimized certainty equivalent,” *Mathematical Finance*, vol. 17, pp. 449–476, 3 2007.
- [16] A. J. Kleywegt, A. Shapiro, and T. Homem-de-Mello, “The sample average approximation method for stochastic discrete optimization,” *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2002.
- [17] A. Shapiro, “On complexity of multistage stochastic programs,” *Operations Research Letters*, vol. 34, no. 1, pp. 1–8, 2006.
- [18] M. V. Pereira and L. M. Pinto, “Multi-stage stochastic optimization applied to energy planning,” *Mathematical programming*, vol. 52, no. 1, pp. 359–375, 1991.
- [19] J. R. Birge, “Decomposition and partitioning methods for multistage stochastic linear programs,” *Operations research*, vol. 33, no. 5, pp. 989–1007, 1985.
- [20] G. Lan, “Complexity of stochastic dual dynamic programming,” *Mathematical Programming*, pp. 1–38, 2020.
- [21] A. Shapiro, “Analysis of stochastic dual dynamic programming method,” *European Journal of Operational Research*, vol. 209, no. 1, pp. 63–72, 2011.
- [22] L. Ding, S. Ahmed, and A. Shapiro, “A python package for multi-stage stochastic programming,” *Optimization online*, pp. 1–41, 2019.
- [23] G. Lan, *First-order and Stochastic Optimization Methods for Machine Learning*. Springer.
- [24] M. Frank and P. Wolfe, “An algorithm for quadratic programming,” *Naval research logistics quarterly*, vol. 3, no. 1-2, pp. 95–110, 1956.
- [25] M. Jaggi and M. Suvovskỳ, “A simple algorithm for nuclear norm regularized problems,” in *ICML*, 2010.
- [26] Z. Harchaoui, A. Juditsky, and A. Nemirovski, “Conditional gradient algorithms for machine learning,” in *NIPS Workshop on Optimization for ML*, Citeseer, vol. 3, 2012, pp. 3–2.
- [27] M. Jaggi, “Revisiting frank-wolfe: Projection-free sparse convex optimization,” in *International Conference on Machine Learning*, PMLR, 2013, pp. 427–435.

- [28] G. Lan, “The complexity of large-scale convex programming under a linear optimization oracle,” *arXiv preprint arXiv:1309.5550*, 2013.
- [29] Z. Harchaoui, A. Juditsky, and A. Nemirovski, “Conditional gradient algorithms for norm-regularized smooth convex optimization,” *Mathematical Programming*, vol. 152, no. 1, pp. 75–112, 2015.
- [30] R. M. Freund and P. Grigas, “New analysis and results for the frank–wolfe method,” *Mathematical Programming*, vol. 155, no. 1, pp. 199–230, 2016.
- [31] G. Lan, E. Romeijn, and Z. Zhou, “Conditional gradient methods for convex optimization with function constraints,” *arXiv preprint arXiv:2007.00153*, 2020.
- [32] M. R. Hestenes, “Multiplier and gradient methods,” *Journal of optimization theory and applications*, vol. 4, no. 5, pp. 303–320, 1969.
- [33] M. J. Powell, “A method for nonlinear constraints in minimization problems,” *Optimization*, pp. 283–298, 1969.
- [34] G. Lan and R. D. Monteiro, “Iteration-complexity of first-order penalty methods for convex programming,” *Mathematical Programming*, vol. 138, no. 1, pp. 115–139, 2013.
- [35] G. Lan and R. D. Monteiro, “Iteration-complexity of first-order augmented lagrangian methods for convex programming,” *Mathematical Programming*, vol. 155, no. 1-2, pp. 511–547, 2016.
- [36] G. Korpelevich, “The extragradient method for finding saddle points and other problems,” *Ekon. Mat. Metody*, vol. 12, no. 4, pp. 747–756, 1976.
- [37] A. 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,” *SIAM Journal on Optimization*, vol. 15, no. 1, pp. 229–251, 2004.
- [38] 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.
- [39] D. Boob, Q. Deng, and G. Lan, “Stochastic first-order methods for convex and nonconvex functional constrained optimization,” *arXiv preprint arXiv:1908.02734*, 2019.

- [40] E. Y. Hamedani and N. S. Aybat, “A primal-dual algorithm with line search for general convex-concave saddle point problems,” *SIAM Journal on Optimization*, vol. 31, no. 2, pp. 1299–1329, 2021.
- [41] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov, “New variants of bundle methods,” *Mathematical programming*, vol. 69, no. 1, pp. 111–147, 1995.
- [42] Y. Nesterov, *Lectures on Convex Optimization* (Springer Optimization and Its Applications). Springer International Publishing, 2018, ISBN: 9783319915777.
- [43] E. Van Den Berg and M. P. Friedlander, “Probing the pareto frontier for basis pursuit solutions,” *SIAM Journal on Scientific Computing*, vol. 31, no. 2, pp. 890–912, 2009.
- [44] E. Van den Berg and M. P. Friedlander, “Sparse optimization with least-squares constraints,” *SIAM Journal on Optimization*, vol. 21, no. 4, pp. 1201–1229, 2011.
- [45] A. Y. Aravkin, J. V. Burke, and M. P. Friedlander, “Variational properties of value functions,” *SIAM Journal on optimization*, vol. 23, no. 3, pp. 1689–1717, 2013.
- [46] Q. Lin, S. Nadarajah, and N. Soheili, “A level-set method for convex optimization with a feasible solution path,” *SIAM Journal on Optimization*, vol. 28, no. 4, pp. 3290–3311, 2018.
- [47] A. Y. Aravkin, J. V. Burke, D. Drusvyatskiy, M. P. Friedlander, and S. Roy, “Level-set methods for convex optimization,” *Mathematical Programming*, vol. 174, no. 1, pp. 359–390, 2019.
- [48] Y. Nesterov, “A method for solving the convex programming problem with convergence rate $O(1/k^2)$,” *Proceedings of the USSR Academy of Sciences*, vol. 269, pp. 543–547, 1983.
- [49] C. Lemarechal, “An extension of davidon methods to non differentiable problems,” in *Nondifferentiable optimization*, Springer, 1975, pp. 95–109.
- [50] P. Wolfe, “A method of conjugate subgradients for minimizing nondifferentiable functions,” in *Nondifferentiable optimization*, Springer, 1975, pp. 145–173.
- [51] G. Lan, “Bundle-level type methods uniformly optimal for smooth and nonsmooth convex optimization,” *Mathematical Programming*, vol. 149, no. 1, pp. 1–45, 2015.
- [52] G. Scutari, F. Facchinei, L. Lampariello, S. Sardellitti, and P. Song, “Parallel and distributed methods for constrained nonconvex optimization-part ii: Applications in communications and machine learning,” *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1945–1960, 2016.

- [53] D. Boob, Q. Deng, G. Lan, and Y. Wang, “A feasible level proximal point method for nonconvex sparse constrained optimization,” *arXiv preprint arXiv:2010.12169*, 2020.
- [54] O. Güler, “New proximal point algorithms for convex minimization,” *SIAM Journal on Optimization*, vol. 2, no. 4, pp. 649–664, 1992.
- [55] D. Bertsekas, *Convex optimization algorithms*. Athena Scientific, 2015.
- [56] G. Lan and Y. Yang, “Accelerated stochastic algorithms for nonconvex finite-sum and multiblock optimization,” *SIAM Journal on Optimization*, vol. 29, no. 4, pp. 2753–2784, 2019.
- [57] W. Kong, J. G. Melo, and R. D. Monteiro, “Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs,” *SIAM Journal on Optimization*, vol. 29, no. 4, pp. 2566–2593, 2019.
- [58] R. Ma, Q. Lin, and T. Yang, “Proximally constrained methods for weakly convex optimization with weakly convex constraints,” *arXiv preprint arXiv:1908.01871*, 2019.
- [59] S. Ghadimi and G. Lan, “Stochastic first-and zeroth-order methods for nonconvex stochastic programming,” *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2341–2368, 2013.
- [60] S. Ghadimi and G. Lan, “Accelerated gradient methods for nonconvex nonlinear and stochastic programming,” *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, 2016.
- [61] Z. Allen-Zhu and E. Hazan, “Variance reduction for faster non-convex optimization,” in *International conference on machine learning*, PMLR, 2016, pp. 699–707.
- [62] S. J. Reddi, A. Hefny, S. Sra, B. Póczos, and A. Smola, “Stochastic variance reduction for nonconvex optimization,” in *International conference on machine learning*, PMLR, 2016, pp. 314–323.
- [63] Y. Carmon, J. C. Duchi, O. Hinder, and A. Sidford, “Accelerated methods for nonconvex optimization,” *SIAM Journal on Optimization*, vol. 28, no. 2, pp. 1751–1772, 2018.
- [64] D. Boob, Q. Deng, and G. Lan, “Level constrained first order methods for function constrained optimization,” *arXiv preprint arXiv:2205.08011*, 2022.

- [65] 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.
- [66] G. Lan and Y. Zhou, “Conditional gradient sliding for convex optimization,” *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1379–1409, 2016.
- [67] D. Garber, “Faster projection-free convex optimization over the spectrahedron,” *Advances in Neural Information Processing Systems*, vol. 29, 2016.
- [68] G. Lan, S. Pokutta, Y. Zhou, and D. Zink, “Conditional accelerated lazy stochastic gradient descent,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 1965–1974.
- [69] R. Rockafellar, “Duality and optimality in multistage stochastic programming,” *Annals of Operations Research*, vol. 85, pp. 1–19, 1999.
- [70] J. Bonnans, Z. Cen, and T. Christel, “Sensitivity analysis of energy contracts by stochastic programming techniques,” in (Numerical Methods in Finance, Springer Proceeding in Mathematics 12), R. Carmona, P. D. Moral, P. Hu, and N. Oudjane, Eds., Numerical Methods in Finance, Springer Proceeding in Mathematics 12. 2012, pp. 447–471.
- [71] G. Terca and D. Wozabal, “Envelope theorems for multistage linear stochastic optimization,” *Operations Research*, 2020.
- [72] J. F. Bonnans and A. Shapiro, *Perturbation analysis of optimization problems*. Springer Science & Business Media, 2013.
- [73] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms I and II*. Springer-Verlag, Berlin, 1993.
- [74] V. Guigues, “SDDP for some interstage dependent risk-averse problems and application to hydro-thermal planning,” *Computational Optimization and Applications*, vol. 57, pp. 167–203, 2014.
- [75] D. P. Bertsekas and S. E. Shreve, *Stochastic optimal control: the discrete-time case*. Athena Scientific, 1996, vol. 5.
- [76] A. Shapiro and L. Ding, “Periodical multistage stochastic programs,” *SIAM Journal on Optimization*, vol. 30, no. 3, pp. 2083–2102, 2020.
- [77] P. Zipkin, *Foundations of Inventory Management*. McGraw-Hill, Boston, 2000.

- [78] A. Shapiro, W. Tekaya, J. da Costa, and M. Soares, “Risk neutral and risk averse stochastic dual dynamic programming method,” *European Journal of Operational Research*, vol. 224, no. 2, pp. 375–391, 2013.
- [79] J. Doob, “The limiting distributions of certain statistics,” *Annals of Mathematical Statistics*, vol. 6, pp. 160–169, 1935.
- [80] A. Shapiro, W. Tekaya, J. P. da Costa, and M. P. Soares, “Risk neutral and risk averse stochastic dual dynamic programming method,” *European journal of operational research*, vol. 224, no. 2, pp. 375–391, 2013.
- [81] A. Shapiro, “Analysis of stochastic dual dynamic programming method,” *European Journal of Operational Research*, vol. 209, pp. 63–72, 2011.
- [82] A. B. Philpott and Z. Guan, “On the convergence of stochastic dual dynamic programming and related methods,” *Oper. Res. Lett.*, vol. 36, pp. 450–455, 2008.
- [83] A. Philpott, V. de Matos, and E. Finardi, “On solving multistage stochastic programs with coherent risk measures,” *Operations Research*, vol. 61, no. 4, pp. 957–970, 2013.
- [84] J. Birge and F. Louveaux, *Introduction to Stochastic Programming* (Springer Series in Operations Research). Springer, 2011.
- [85] V. Leclere, P. Carpentier, J.-P. Chancelier, A. Lenoir, and F. Pacaud, “Exact converging bounds for stochastic dual dynamic programming via fenchel duality,” *Siam Journal on Optimization*, vol. 30, pp. 1223–1250, 2 2020.
- [86] V. Guigues, A. Shapiro, and Y. Cheng, “Duality and sensitivity analysis of multi-stage linear stochastic programs,” *Optimization online*, 2019.
- [87] B. da Costa and V. Leclere, “Dual sddp for risk-averse multistage stochastic programs,” *arXiv*, 2021.
- [88] D. Bertsekas and S. Shreve, *Stochastic Optimal Control, The Discrete Time Case*. Academic Press, New York, 1978.
- [89] A. Shapiro and L. Ding, “Upper bound for optimal value of risk averse multistage problems,” *Technical report, Georgia Tech*, 2016.
- [90] R. T. Rockafellar, *Conjugate Duality and Optimization*. Philadelphia: Society for Industrial and Applied Mathematics, 1974.

- [91] A. Shapiro, W. Tekaya, J. da Costa, and M. P. Soares, “Risk neutral and risk averse stochastic dual dynamic programming method,” *European Journal of Operational Research*, vol. 224, pp. 375–391, 2013.
- [92] L. Ding, S. Ahmed, and A. Shapiro, “A python package for multi-stage stochastic programming,” *Optimization online*, 2019.
- [93] R. Liu and A. Shapiro, “Reformulation approach to risk averse stochastic programming,” *Risk Neutral Reformulation Approach to Risk Averse Stochastic Programming*, vol. 286, pp. 21–31, 2020.
- [94] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, “Coherent measures of risk,” *Mathematical finance*, vol. 9, no. 3, pp. 203–228, 1999.
- [95] R. P. Liu and A. Shapiro, “Risk neutral reformulation approach to risk averse stochastic programming,” *European Journal of Operational Research*, vol. 286, no. 1, pp. 21–31, 2020.
- [96] J. C. Hull, *Options futures and other derivatives*. Upper Saddle River, N.J. : Pearson/Prentice Hall, 2006.
- [97] R. T. Rockafellar and S. Uryasev, “Conditional value-at-risk for general loss distributions,” *Journal of banking & finance*, vol. 26, no. 7, pp. 1443–1471, 2002.
- [98] H. Föllmer and A. Schied, “Convex measures of risk and trading constraints,” *Finance and stochastics*, vol. 6, no. 4, pp. 429–447, 2002.
- [99] A. J. McNeil, R. Frey, and P. Embrechts, *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton university press, 2015.
- [100] 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.
- [101] H. E. Romeijn and J. F. Dempsey, “Intensity modulated radiation therapy treatment plan optimization,” *Top*, vol. 16, no. 2, pp. 215–243, 2008.
- [102] B. Tomlin, “On the value of mitigation and contingency strategies for managing supply chain disruption risks,” *Management science*, vol. 52, no. 5, pp. 639–657, 2006.
- [103] M. Carrión, A. B. Philpott, A. J. Conejo, and J. M. Arroyo, “A stochastic programming approach to electric energy procurement for large consumers,” *IEEE Transactions on Power Systems*, vol. 22, no. 2, pp. 744–754, 2007.

- [104] D. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [105] E. J. Candès and M. B. Wakin, “An introduction to compressive sampling,” *IEEE signal processing magazine*, vol. 25, no. 2, pp. 21–30, 2008.
- [106] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE Journal of selected topics in signal processing*, vol. 1, no. 4, pp. 586–597, 2007.
- [107] D. Goldfarb, S. Ma, and K. Scheinberg, “Fast alternating linearization methods for minimizing the sum of two convex functions,” *Mathematical Programming*, vol. 141, no. 1, pp. 349–382, 2013.
- [108] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [109] T. Blumensath and M. E. Davies, “Iterative thresholding for sparse approximations,” *Journal of Fourier analysis and Applications*, vol. 14, no. 5, pp. 629–654, 2008.
- [110] E. J. Candès, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted l-1 minimization,” *Journal of Fourier analysis and applications*, vol. 14, no. 5, pp. 877–905, 2008.
- [111] J.-F. Cai, E. J. Candès, and Z. Shen, “A singular value thresholding algorithm for matrix completion,” *SIAM Journal on optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [112] R. T. Rockafellar, S. Uryasev, *et al.*, “Optimization of conditional value-at-risk,” *Journal of risk*, vol. 2, pp. 21–42, 2000.
- [113] H. M. Markowitz, “Portfolio selection: Efficient diversification of investments,” *Cowles Foundation Monograph*, vol. 16, 1959.
- [114] T.-J. Chang, N. Meade, J. E. Beasley, and Y. M. Sharaiha, “Heuristics for cardinality constrained portfolio optimisation,” *Computers & Operations Research*, vol. 27, no. 13, pp. 1271–1302, 2000.
- [115] D. Li, X. Sun, and J. Wang, “Optimal lot solution to cardinality constrained mean–variance formulation for portfolio selection,” *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, vol. 16, no. 1, pp. 83–101, 2006.

- [116] D. Bertsimas and R. Shioda, “Algorithm for cardinality-constrained quadratic optimization,” *Computational Optimization and Applications*, vol. 43, no. 1, pp. 1–22, 2009.
- [117] J. Gao and D. Li, “Optimal cardinality constrained portfolio selection,” *Operations research*, vol. 61, no. 3, pp. 745–761, 2013.
- [118] X. Zheng, X. Sun, D. Li, and J. Sun, “Successive convex approximations to cardinality-constrained convex programs: A piecewise-linear dc approach,” *Computational Optimization and Applications*, vol. 59, no. 1-2, pp. 379–397, 2014.
- [119] Y. Nesterov, “Smooth minimization of non-smooth functions,” *Mathematical programming*, vol. 103, no. 1, pp. 127–152, 2005.
- [120] J.-y. Gotoh, A. Takeda, and K. Tono, “Dc formulations and algorithms for sparse optimization problems,” *Mathematical Programming*, vol. 169, no. 1, pp. 141–176, 2018.
- [121] R. Bruni, F. Cesarone, A. Scozzari, and F. Tardella, “Real-world datasets for portfolio selection and solutions of some stochastic dominance portfolio models,” *Data in brief*, vol. 8, pp. 858–862, 2016.