# SADDLE POINT TECHNIQUES IN CONVEX COMPOSITE AND ERROR-IN-MEASUREMENT OPTIMIZATION

A Thesis
Presented to
The Academic Faculty

by

Niao He

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Operations Research

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology
December 2015

Copyright © 2015 by Niao He

# SADDLE POINT TECHNIQUES IN CONVEX COMPOSITE AND ERROR-IN-MEASUREMENT OPTIMIZATION

Approved by:


Dr. Arkadi Nemirovski, Advisor
H. Milton Stewart School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Dr. Maria-Florina Balcan
School of Computer Science
*Carnegie Mellon University*

Dr. Alexander Shapiro
H. Milton Stewart School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Dr. Anton Kleywegt
H. Milton Stewart School of Industrial and Systems Engineering
*Georgia Institute of Technology*

Dr. Shabbir Ahmed
H. Milton Stewart School of Industrial and Systems Engineering
*Georgia Institute of Technology*



Date Approved: October 27, 2015

# ACKNOWLEDGEMENTS

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

My foremost and deepest gratitude goes to my advisor, Professor Arkadi Nemirovski, for his uncountable guidance and relentless support both in research and in life. I couldn't be more appreciative of his tremendous dedication in advising and nurturing me during my PhD years. He is always there for help whenever needed and always with great patience to guide me through the jungles. Apart from his unparalleled knowledge and wisdom, he is also incredibly motivating and encouraging at all times. I couldn't imagine what a better advisor could be like. Working with him is surely the most valuable part of this journey, and a priceless fortune of a lifetime. Moreover, I would like to express my deepest appreciation to my committee, Professors Alex Shapiro, Shabbir Ahmed, Nina Balcan, and Anton Kleywegt, for their generous efforts and valuable comments on the thesis, and of course I would never forget about those excellent and insightful courses they have taught me in the past.

I have been extremely fortunate to collaborate with many outstanding researchers and great minds in the area of optimization and machine learning, Professors Anatoli Juditsky, Zaid Harchaoui, Le Song, and Nina Balcan, who have inspired me with their extraordinary expertise and vision in the new frontiers of research; their enthusiasm and professional altitude as a researcher has had a profound impact on my own career pursuit. I am truely grateful for their invaluable advices and discussions, which have incented me to widen my research from various perspectives. I also had the great pleasure of working with Dr. Matthew Brand, a remarkable researcher, during my summer internship in MERL. Last but not the least, my gratitudes extend to some of my other coauthors, who I have learnt a lot from, Dr. Hua Ouyang, Dr. Yingyu Liang, Bo Dai, Bo Xie, Nan Du, and I very

much appreciate the vibrant and fruitful conversations we had. Moreover, I would also like to thank Dr. Guanghui Lan and Dr. Fatma Kilinc-Karzan for their incredible kindness, tremendous help and valuable career advices.

I must also thank many other outstanding faculty members at Georgia Tech since I have benefited so much from the graduate courses I took all these years, particularly Drs. Santanu Dey, Vladimir Koltchinskii, William Cook, Jim Dai, Greg Blekherman, Hayriye Ayhan, Sigrun Andradottir, etc. I would also like to thank all the staff of ISyE, especially Pam Morrison, Mark Reese, and Yvonne Smith for their assistance. I'd also like to give my heartfelt and special thanks to Dr. Gary Parker, for his genuine caring in my progress even after retirement.

I am also very lucky to have met so many amazing friends, classmates, officemates, and colleagues at Georgia Tech. Special thanks go to Yuan Wang, Yanling Chang, Chengliang Zhang, Qiushi Chen, Cristobal Guzman, Carlos de Andrade, Carl Morris, Javad Feizollahi, Jikai Zou, Zhi Han, Yu Zhang, Feng Qiu, Weijun Ding, Weijun Xie, Qianyi Wang, Haiyue Yu, Peng Tang, Yi Zhang, Xinchang Wang and many others.

Last but definitely not least, I would like to thank my dearest mom and sister for their understanding and company. I am deeply indebted to my boyfriend, Yuntao Li, who has taken care of a million of little things for me; I would never have gone so far without his sacrifices and love.

Finally, I would like to acknowledge the support for this research from National Science Foundation Grant CMMI-1232623, and the Kiplinger Fellowship from ISyE.

# TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

# SUMMARY

This dissertation aims to develop efficient algorithms with improved scalability and stability properties for large-scale optimization and optimization under uncertainty, and to bridge some of the gaps between modern optimization theories and recent applications emerging in the Big Data environment. To this end, the dissertation is dedicated to two important subjects – i) *Large-scale Convex Composite Optimization* and ii) *Error-in-Measurement Optimization*. In spite of the different natures of these two topics, the common denominator, to be presented, lies in their accommodation for systematic use of saddle point techniques for mathematical modeling and numerical processing. The main body can be split into three parts.

In the first part, we consider a broad class of variational inequalities with composite structures, allowing to cover the saddle point/variational analogies of the classical convex composite minimization (i.e. summation of a smooth convex function and a simple nonsmooth convex function). We develop novel composite versions of the state-of-the-art Mirror Descent and Mirror Prox algorithms aimed at solving such type of problems. We demonstrate that the algorithms inherit the favorable efficiency estimate of their prototypes when solving structured variational inequalities. Moreover, we develop several variants of the composite Mirror Prox algorithm along with their corresponding complexity bounds, allowing the algorithm to handle the case of imprecise prox mapping as well as the case when the operator is represented by an unbiased stochastic oracle.

In the second part, we investigate four general types of large-scale convex composite optimization problems, including (a) multi-term composite minimization, (b) linearly constrained composite minimization, (c) norm-regularized nonsmooth minimization, and (d) maximum likelihood Poisson imaging. We demonstrate that the composite Mirror Prox, when integrated with saddle point techniques and other algorithmic tools, can solve all these optimization problems with the best known so far rates of convergences. Our main

related contributions are as follows. Firstly, regards to problems of type (a), we develop an optimal algorithm by integrating the composite Mirror Prox with a saddle point reformulation based on exact penalty. Secondly, regards to problems of type (b), we develop a novel algorithm reducing the problem to solving a "small series" of saddle point subproblems and achieving an optimal, up to log factors, complexity bound. Thirdly, regards to problems of type (c), we develop a Semi-Proximal Mirror-Prox algorithm by leveraging the saddle point representation and linear minimization over problems' domain and attain optimality both in the numbers of calls to the first order oracle representing the objective and calls to the linear minimization oracle representing problem's domain. Lastly, regards to problem (d), we show that the composite Mirror Prox when applied to the saddle point reformulation circumvents the difficulty with non-Lipschitz continuity of the objective and exhibits better convergence rate than the typical rate for nonsmooth optimization. We conduct extensive numerical experiments and illustrate the practical potential of our algorithms in a wide spectrum of applications in machine learning and image processing.

In the third part, we examine error-in-measurement optimization, referring to decision-making problems with data subject to measurement errors; such problems arise naturally in a number of important applications, such as privacy learning, signal processing, and portfolio selection. Due to the postulated observation scheme and specific structure of the problem, straightforward application of standard stochastic optimization techniques such as Stochastic Approximation (SA) and Sample Average Approximation (SAA) are out of question. Our goal is to develop computationally efficient and, hopefully, not too conservative data-driven techniques applicable to a broad scope of problems and allowing for theoretical performance guarantees. We present two such approaches – one depending on a fully algorithmic calculus of saddle point representations of convex-concave functions and the other depending on a general approximation scheme of convex stochastic programming. Both approaches allow us to convert the problem of interests to a form amenable for SA or SAA. The latter developments are primarily focused on two important applications – affine signal processing and indirect support vector machines.

# CHAPTER I

# INTRODUCTION

## *1.1 Motivation and Goals*

In the era of Big Data, due to the massive amount and diverse sources of data, decision-making processes become very challenging and require good optimization models and problem-solving methods, particularly those with scalability and stability. To tackle these challenges, both in practice and in theory, there is a strong need for studies on designing efficient algorithms for optimization problems in high-dimensional regimes and establishing data-oriented approaches to optimization problems under uncertainty. *This dissertation aims to develop efficient algorithms with improved scalability and stability properties for large-scale optimization and optimization under uncertainty, and to bridge some of gaps between modern optimization theories and recent applications emerging in the Big Data environment.*

This dissertation is driven by and concentrates on two important subjects.

### 1.1.1 Large-scale Convex Composite Optimization

Last decade demonstrates significant and steadily growing interests in minimizing composite functions of the form:

$$\min_{x \in X} \; f(x) + h(x)$$

where $f$ is a convex, continuously differentiable function and $h$ is a convex but perhaps not differentiable function. Such problems arise ubiquitously in machine learning, signal processing, bioinformatics, computer vision, and many other fields. In these applications, $f$ usually refers to loss function, or model fitting term, measuring how well a candidate solution $x$ "fits" the available information on the true solution, and $h$ is a regularizer "promoting" desired properties of the solution we seek for (sparsity, low rank, etc.). Popular problems include the Lasso, ridge regression, trace-norm matrix completion, total variation

based image denoising, and so on. In general, nonsmoothness of the objective in a convex optimization problem slows down the achievable convergence rate; the challenge in composite minimization is to avoid this slow-down by utilizing special structure of the nonsmooth term $h$; such structure is indeed present in relevant applications.

Proximal algorithms are especially well-suited for composite minimization. It was shown in Nesterov's seminal work [63] and several subsequent papers (see, e.g., [6, 7, 22, 80, 76] and references therein) that when function $f$ is smooth, the proximal version of the fast gradient method works as if there were no nonsmooth term $h$ at all and exhibits the $O(1/t^2)$ convergence rate, which is the optimal rate attainable by first order algorithms of large-scale smooth convex optimization. These proximal algorithms (see [69] for a comprehensive survey) require computation of a composite proximal operator at each iteration, i.e. solving problems of the form

$$\min_{x \in X} \left\{ \frac{1}{2}\|x\|_2^2 + \langle \xi, x \rangle + \alpha h(x) \right\}$$

given input vector $\xi$ and positive scalar $\alpha$. We call function $h$ that admits easy-to-compute composite proximal operators, *proximal-friendly*. Typical examples of proximal-friendly functions considered in literatures include $\ell_p$ norm, trace/nuclear norm and block $\ell_1/\ell_p$ norm (group lasso). The situation when $f$ is nonsmooth has also been widely studied in the literature. Various algorithms have been developed and achieve the optimal $O(1/t)$ convergence rate, based on smoothing techniques [64] and primal-dual method [23, 25].

In another line of research, conditional gradient type algorithms have lately received an emerging interest when dealing with large scale composite minimization. In several important cases, especially in high dimensional regime, computing proximal operator can be expensive or intractable. A classical example is the nuclear norm minimization arising in low rank matrix recovery and semidefinite optimization. Here computing proximal operator boils down to singular value thresholding and thus requires computationally expensive in the large scale case full singular decomposition. In contrast to the proximal algorithms, conditional gradient type methods operate with the linear minimization oracle (LMO) at

each iteration, i.e., solving auxiliary problems of the form

$$\min_{x \in X} \{\langle \xi, x \rangle + \alpha h(x)\}$$

which can be much cheaper than computing composite proximal operators. For instance, in the case of the nuclear-norm, the LMO only requires computing the leading pair of singular vectors, which is by orders of magnitude faster than full singular value decomposition. We call function $h$ that admits easy-to-compute linear minimization oracle, *LMO-friendly*. When function $f$ is smooth, it was shown in [36] and later in[65] that the generic conditional gradient method exhibits a $O(1/t)$ rate of convergence, which is also the optimal rate attainable by LMO-based algorithms.

**Motivation.** Despite of the much success in this classical settings of composite minimization, it comes to our attention that most of these algorithms cannot be directly applied to the following situations:

1. *there are several proximal-friendly or LMO-friendly components in the objective;*
2. *the objective is separable with several proximal-friendly terms, but the corresponding blocks of variables are subject to coupling linear constraints;*
3. *there is an additive mixture of proximal-friendly and LMO-friendly components;*
4. *$f$ is nonsmooth, and $h$ is LMO-friendly;*
5. *$f$ is even not Lipschitz continuous.*

Problems of the outlined types recently emerge in a wide spectrum of application, especially in statistics, machine learning and image processing, where regularization technique plays an important role. In order to deal with massive and complex datasets, a variety of structured regularizers (sometimes called penalties) and their hybrid mixtures are introduced to promote several desired properties of the solution simultaneously, such as sparsity and low rank. There is a huge body of literature on this subject, see, e.g. [81, 4, 17] and references therein.

One motivating example is the matrix completion problem, arising in recommendation systems, where the goal is to reconstruct the original matrix $y \in \mathbf{R}^{n\times n}$, assumed to be both sparse and low-rank, given noisy observations of part of the entires. Specifically, let the observation be $b = P_\Omega y + \xi$, where $\Omega$ is a given set of cells in an $n \times n$ matrix, $P_\Omega y$ is the restriction of $y \in \mathbf{R}^{n\times n}$ onto $\Omega$, and $\xi$ is a random noise. A natural way to recover $y$ from $b$ is to solve the optimization problem

$$\min_{y\in\mathbf{R}^{n\times n}} \left\{ \frac{1}{2}\|P_\Omega y - b\|_2^2 + \lambda\|y\|_1 + \mu\|y\|_{\rm nuc} \right\}$$

where $\mu, \lambda > 0$ are regularization parameters. Here $\|y\|_2 = \sqrt{\mathrm{Tr}(y^T y)}$ is the Frobenius norm, $\|y\|_1 = \sum_{i,j=1}^n |y_{ij}|$ is the $\ell_1$-norm, and $\|y\|_{\rm nuc} = \sum_{i=1}^n \sigma_i(y)$ ($\sigma_i(y)$ are the singular values of $y$) is the nuclear norm of a matrix $y \in \mathbf{R}^{n\times n}$. The $\ell_1$-norm regularization term is used to promote sparsity and the nuclear norm term is used to promote low rank. One can see that, when the size $n$ of $y$ is "large, but not too large" (say, $n \le 2000$), both terms can be regarded as proximal-friendly. Once the dimension becomes "very large,", one can no longer treat both penalties as proximal friendly, but perhaps still can treat them as LMO-friendly. In the gray zone in between we deal with a mixture of proximal-friendly and LMO-friendly regularizers. None of these situations can be directly tackled with the existing first-order algorithms, proximal type and conditional gradient type alike.

**Goals.** While problems of the outlined types occur in a wide spectrum of real-world applications among the aforementioned fields (more examples will be provided in subsequent sections), the literature on design of scalable algorithms adjusted to the outlined problems' structures turns out to be quite limited. Our ultimate goal on this subject is i) on the theoretical side, to develop a "universal" algorithmic framework that covers a broad class of optimization problems, including composite settings of almost all problems of convex structures (convex minimization, convex-concave saddle point problems, variational inequalities, Nash equilibrium problems), and with "complications" 1– 5 listed above; ii) on the practical side, to apply the resulting algorithmic tool to four generic convex optimization problems:

(a) *Multi-Term Composite Minimization:* convex optimization problem

$$\min_{y\in Y} \sum_{k=1}^{K} [\psi_k(A_k y + b_k) + \Psi_k(A_k y + b_k)] \tag{1.1.1}$$

where $Y$ is closed convex set, for $1 \le k \le K$, $\psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex Lipschitz-continuous functions, and $\Psi_k(\cdot) : Y_k \to \mathbf{R}$ are proximal-friendly convex functions;

(b) *Linearly Constrained Composite Minimization:* multi-term composite minimization problems that are subject to linear equality constraints

$$\begin{array}{rl} \min\limits_{[y^1;\ldots;y^K]\in Y_1\times\cdots\times Y_K} & \sum_{k=1}^{K} \left[\psi_k(y^k) + \Psi_k(y^k)\right] \\ \text{s.t.} & \sum_{k=1}^{K} A_k y^k = b \end{array} \tag{1.1.2}$$

where $Y_k$ are closed convex sets and $\psi_k$ and $\Psi_k$ are as in (a);

(c) *Norm-Regularized Nonsmooth Minimization:* composite minimization

$$\min_{y\in Y} \; f(y) + h(Ay) \tag{1.1.3}$$

where $f$ is a convex Lipschitz-continuous function given by saddle point representation, and $h$ is a LMO-friendly function;

(d) *Composite Maximum Likelihood Poisson Imaging:* a particular non-Lipschitz convex minimization problem

$$\min_{x\in\mathbf{R}^n_+} L(x) + h(x), \text{ with } L(x) = s^T x - \sum_{i=1}^{m} c_i \ln(a_i^T x) \tag{1.1.4}$$

where $s, c, a_i, i = 1, \ldots m$ are given nonnegative vectors and $h$ is proximal-friendly. Specific feature of Poisson Imaging is that $L(\cdot)$ in general is not even Lipschitz continuous.

### 1.1.2 Error-in-Measurement Optimization

Besides the large scale, another ubiquitous fact in many real world problems is that problem's data is not always known exactly. Due to intrinsic physical limitations, prohibitive cost, or hard constraints, often data cannot be measured accurately or directly, and therefore are subject to measurement errors. Measurement errors take place in a wide spectrum of

applications, ranging from traditional medical tests, remote sensing, bioinformatics, chemical process engineering to more recent privacy learning, portfolio management, and electric power systems operations. Data that suffer from such errors can be loosely categorized into two classes: i) *fixed parameters*, such as characteristics of technological devices, inherent constants of reaction kinetics, proportions of components in raw materials, statistical parameters of a stochastic process ii) *random samples from a fixed distribution*, such as highly variable sensor network data, clinical trials, medical scans, etc.

**Motivation.** Many optimization problems dealing with data affected by measurement errors can be generally posed as,

$$\min_{x \in X} \Phi(x, \pi^*) \qquad (\star)$$

where $\Phi(\cdot, \pi^*)$ is convex in $x \in X$, and $\pi^* \in \Pi$ is unknown, but admits observations ("measurements") $\omega_t$, $t = 1, 2, ...$, sampled independently from a distribution $P_{\pi^*}$, where $\{P_\pi : \pi \in \Pi\}$ is a given family of distributions with the domain $\Pi$ known to contain $\pi^*$. We call such problems, *error-in-measurement optimization*.

Related problems got some attention and have been studied in different contexts in the literature including research on errors-in-variables models [20], missing-data-problems [52], online learning models with noisy data [21], robust optimization [8], misspecified optimization [2, 42]. We emphasize here that our interest is in closely related yet distinct settings, and our theoretical developments to be presented, seem to be novel.

As of now, studies on the error-in-measurement optimization in the setting we have outlined seem to be rather limited. An intuitive but naive way to address the problem might be to simply replace the unknown data by its sample estimate, This, however, could lead to highly unreliable solutions unless the size of sample is large enough. A more reliable way to solve these problems is to rely on data-driven robust optimization approaches by constructing uncertainty sets using historical observations of the random variable [11, 29, 12]. However, such approaches suffer from a) unclear guidelines on constructing uncertainty sets, b) computational deficiency in the high-dimensional regime, and c) overly-conservative solutions when amount of measurements is limited. A conceptually less conservative approach

would be to adjust the decision variables as the sampling goes on, like what Stochastic Approximation algorithm [74, 71, 72, 58] does. However, specific structures of problem $(\star)$ and of the postulated observation scheme make straightforward application of the standard Stochastic Optimization techniques (like Stochastic Approximation, or Sample Average Approximation) just impossible. Some techniques for converting $(\star)$ to a form amenable for Stochastic Approximation are proposed in [21]; these techniques, however, impose severe limitations on the objectives $\Phi(x,\pi)$ and measurement schemes which can be treated.

**Goals.** In connection with error-in-measurement optimization, our goal is to develop computationally efficient and, hopefully, not too conservative data-driven techniques applicable to a broad scope of problems $(\star)$ and allowing for theoretical performance guarantees. Our primary focus is on the following three generic scenarios:

(i) *System of convex constraints under direct noisy observations of the data:* we are interested in solving the system

$$\text{Find } x \in X: \quad F_i(x,\pi^*) \le 0, 1 \le i \le I,$$

where $F_i(x,\pi) : X \times \Pi$ is convex in $u$ and concave in $\pi$; the true data $\pi^* = \mathbf{E}_{\xi\sim P}\{\xi\}$ is unknown but we can directly sample from $P$.

(ii) *Convex minimization under indirect observations:* we are interested in solving problems in the form $(\star)$ with the data $\pi^*$ being a finite dimensional vector, and the observations $\omega_t$ are given by $\omega_t = A\pi^* + \eta_t, t = 1, 2, \ldots$, where $A$ is a given matrix, and $\eta_t$ are i.i.d. zero mean observations with known covariance matrix.

(iii) *Stochastic programming under indirect observations:* we are interested in solving problems in the form $(\star)$ with $\pi^*$ being a unknown distribution,

$$\Phi(x,\pi^*) := \mathbf{E}_{\xi\sim\pi^*}[F(x,\xi)]$$

and the observations $\omega_t$ given by $\omega_t = \xi_t + \eta_t, t = 1, 2, \ldots$, where $\xi_t$ are i.i.d. sampled from $\pi^*$, and $\eta_t$ are independent of $\xi_t$, i.i.d. zero mean observations with known covariance matrix.

## *1.2 Outline and Main Results*

The Thesis is organized as follows.

In Chapter II, we first review the basic theory of accuracy certificates, which play a central role in quantifying the accuracy of solutions to generic problems with convex structure, including convex minimization, convex-concave saddle point problem, convex Nash equilibrium problem, and variational inequalities with monotone operators. We consider a broad class of variational inequalities with composite structure, allowing to cover saddle point/variational analogies of the classical convex composite minimization. We develop novel composite versions of the state-of-the-art Mirror Descent and Mirror Prox algorithms aimed at solving such type of problems. We demonstrate that the algorithms inherit the favorable efficiency estimate of their prototypes when solving structured variational inequalities, namely, a $O(1/\epsilon^2)$ complexity bound when the monotone operator is bounded and a $O(1/\epsilon)$ complexity bound when the monotone operator is Lipschitz continuous. Moreover, we develop several variants of the composite Mirror Prox algorithm, allowing the algorithm to handle the case of imprecise prox mapping as well as the case when the operator is represented by an unbiased stochastic oracle. Main results of Chapter II are summarized in Theorem 2.4.1 - 2.4.2, Theorem 2.5.1 - 2.5.3, Corollary 2.5.1 - 2.5.4.

In Chapter III, we investigate four general types of convex composite optimization problems outlined at the end of Section 1.1.1. We show that the composite Mirror Prox algorithm, when combined with saddle point representations and some other algorithmic techniques, can solve all these optimization problems, exhibiting the best known so far rates of convergence. To be more specific,

- Section 3.2 is devoted to multi-term composite minimization. We exploit the problem's structure and develop a saddle point reformulation based on exact penalty that allows to directly apply the composite Mirror Prox algorithm. The resulting algorithm achieves the optimal, under the circumstances, $O(1/t)$ rate of convergence. We also present, highly encouraging in our opinion, results of numerical experiments for two important applications – matrix completion and image decomposition. Main results

of Section 3.2 are summarized in Proposition 3.2.1 and Corollary 3.2.1.

- Section 3.3 is devoted to linearly constrained composite minimization. We propose a sequential composite Mirror Prox algorithm which solves a sequence of saddle point subproblems. The algorithm achieves an overall $O(1/\epsilon)$ complexity bound up to some log factors. We present promising experimental results showing the potential of this algorithm for the basis pursuit application. Main results of Section 3.3 are summarized in Propostion 3.3.1 and Theorem 3.3.1.

- Section 3.4 is devoted to norm-regularized nonsmooth minimization. We propose the Semi-Proximal Mirror-Prox algorithm, which leverages the saddle point representation of one component of the objective while handling the other component via linear minimization over the problem's domain. We establish the theoretical convergence rate of Semi-Proximal Mirror-Prox, which exhibits the optimal complexity bounds in three aspects: i) $O(1/\epsilon)$ for the number of calls to first-order oracles, ii) $O(1/\epsilon^2)$ for the number of calls to linear minimization oracle, and iii) $O(1/\epsilon^2)$ for the number of calls to the stochastic oracles if under stochastic setting. We present promising experimental results illustrating the the potential of our approach as compared to several competing methods for two machine learning applications – robust collaborative filtering for movie recommendation and link prediction for social network analysis. Main results of Section 3.4 are summarized in Propositions 3.4.1 - 3.4.3.

- Section 3.5 is devoted to Maximum Likelihood Poisson Imaging. We investigate problem of minimizing Poisson-type loss (problem (d) in the end of Section 1.1.1), which has been a long-standing challenge in machine learning community due to lack of Lipschitz continuity when dealing with Poisson loss. We utilize saddle point reformulation of the problem of interest and process the resulting problem with composite Mirror Prox algorithm, thus avoiding the necessity to deal directly with a non-Lipschitz objective. We show that under favorable circumstances, the algorithm enjoys a $O(1/t)$

convergence rate in contrast to the usual $O(1/\sqrt{t})$ rate for solving nonsmooth optimization. We also demonstrate experimentally, the efficiency of the proposed algorithm as applied to Poison Emission Tomography reconstruction. The main results of Section 3.5 is summarized in Propositions 3.5.1 - 3.5.2.

Main results of Chapter II, III of the Thesis significantly improve upon our previous work in [27, 28, 31, 67] and lead to our sucessive publications in [38, 39].

In Chapter IV, we investigate error-in-measurement optimization. In Section 4.2, we focus on solving the system of convex constraints with direct noisy observations of the data (problem (i) in Section 1.1.2). We first develop a fully algorithmic calculus of saddle point representations for convex-concave functions, in analogy to the well-known Fenchel duality of convex functions. We use this calculus to convert the system of convex constraints we want to solve into convex-concave saddle point problem allowing for stochastic first-order oracles and process the resulting problem by mirror descent stochastic approximation. We provide rigorous accuracy analysis for the approximate solution yielded by the stochastic approximation procedure and propose several theoretically justified techniques for validating the quality of this solution. Main results of this section are summarized in Propositions 4.2.1 - 4.2.6. In Section 4.3, we deal with the case of indirect noisy observations (problem (ii) and (iii) in Section 1.1.2). We propose a general approximation scheme that reduces the problems to convex stochastic programming with semiinfinite constraints. We develop techniques for building safe tractable approximations of these semi-infinite problems and process them with stochastic approximation (SA) or sample average approximation (SAA) . These developments are primarily focused on two important applications – affine signal processing and indirect support vector machines. We present encouraging, albeit at this point in time very preliminary, numerical results illustrating the practical potential of our approach as applied to the affine signal processing. Main results of this section are summarized in Propositions 4.3.1 - 4.3.3.

In summary, we believe that our theoretical developments and numerical results on composite saddle point Mirror Prox based algorithms presented in Chapters II, III and

published in our papers [38, 39] clearly demonstrate high theoretical and practical significance of the approaches to Large-scale Composite Convex Optimization we are developing. As compared to this research, our studies on error-in-measurement optimization presented in Chapter III are in a less developed stage, due to natural time limitations, and we intend to carry out in-depth research along the directions outlined in Chapter IV in the future. We believe, however, that already the preliminary in their nature results of Chapter IV demonstrate novelty and broad scope of the proposed approaches and justify incorporating this material into the Thesis.

To conclude Introduction, we remark that while at the first glance the two topics of our Thesis – Large-scale Composite Convex Optimization and Error-in-Measurement Convex Optimization have not that much in common, such an impression would be wrong: as we see it, the "common denominator" of these two topics, reflected in the title of our Thesis, is the systematic use of saddle point techniques for modeling problems of interest and for their numerical processing.

# CHAPTER II

# COMPOSITE MIRROR DESCENT/PROX FOR PROBLEMS WITH CONVEX STRUCTURE

## *2.1 Overview*

In this chapter, we first review the basic theory of accuracy certificates, which plays a central role in quantifying the accuracy of solutions to generic problems with convex structure, including convex minimization, convex-concave saddle point problem, convex Nash equilibrium problem, and variational inequalities with monotone operators. We then introduce a broad class of variational inequalities with special structure, which represents saddle point/variational analogies of what is usually called composite minimization (minimizing a sum of an easy-to-handle nonsmooth and a general-type smooth convex functions as if there were no nonsmooth component at all). We develop composite versions of the state-of-ther-art Mirror Descent and Mirror Prox algorithms for solving such type of problems. We demonstrate that the algorithms inherit the favorable efficiency estimate of their prototypes when solving structured variational inequalities, namely, a $O(1/\epsilon^2)$ complexity bound when the monotone operator is bounded and a $O(1/\epsilon)$ complexity bound when the monotone operator is Lipschitz continuous. To make it even more general and flexible, we establish several variants of the composite Mirror Prox algorithm, allowing the algorithm to handle inexactness of the prox mapping as well as the case when the operator is represented by an unbiased stochastic oracle.

**Organization of the chapter.** This chapter is organized as follows. We start by discussing some required background on accuracy certificates for problems with convex structures (including convex minimization, convex-concave saddle point problems, and variational inequalities with monotone operators) in Section 2.2. In Section 2.3, we first define the notion of structured variational inequalities we are mainly interested in; we then illustrate the notion with two general applications, one is a composite Nash equilibrium problem

and the other is a composite saddle point problem. In Section 2.4, we present the composite Mirror Descent algorithm along with the theoretical developments and results when applying to the Nash equilibrium problem. In Section 2.5, we discuss theoretical aspects of the composite Mirror Prox (CoMP) algorithm. More specifically, in Section 2.5.1, we establish the theoretical convergence rate when applying CoMP to the composite saddle point problem. In Section 2.5.2, we modify the algorithm allowing for general averaging schemes. In Section 2.5.3, we discuss the inexact CoMP algorithm. In Section 2.5.4, we discuss the stochasitc CoMP algorithm. Concluding remarks are made in Section 2.6.

## *2.2 Preliminaries: Accuracy Certificates for Problems with Convex Structure*

In this section, we review the basic theory of accuracy certificates, which is often used to certify the accuracy of solutions to generic problems with convex structure (including convex minimization, convex-concave saddle point problem, convex Nash equilibrium problem, and variational inequalities with monotone operators). In the sequel, we discuss four types of problems with convex structure along with their accuracy measures and show that the accuracy certificates play a key role in those cases for generating an approximate solution and quantifying its quality.

### 2.2.1 Accuracy Certificates

**Execution protocols and accuracy certificates.** Let $X$ be a nonempty closed convex set in a Euclidean space $E$ and $F(x) : X \to E$ be a vector field.

Suppose that we process $(X, F)$ by an algorithm which generates a sequence of search points $x_t \in X$, $t = 1, 2, ...$, and computes the vectors $F(x_t)$, so that after $t$ steps we have at our disposal *t-step execution protocol* $\mathcal{I}_t = \{x_\tau, F(x_\tau)\}_{\tau=1}^t$. By definition, an *accuracy certificate* for this protocol is simply a collection $\lambda^t = \{\lambda_\tau^t\}_{\tau=1}^t$ of nonnegative reals summing up to 1. We associate with the protocol $\mathcal{I}_t$ and accuracy certificate $\lambda^t$ two quantities as follows:

- *Approximate solution* $x^t(\mathcal{I}_t, \lambda^t) := \sum_{\tau=1}^t \lambda_\tau^t x_\tau$, which is a point of $X$;

- *Resolution* $\mathrm{Res}(X'|\mathcal{I}_t, \lambda^t)$ *on a subset* $X' \neq \emptyset$ of $X$ given by

$$\mathrm{Res}(X'|\mathcal{I}_t, \lambda^t) = \sup_{x \in X'} \sum_{\tau=1}^{t} \lambda_\tau^t \langle F(x_\tau), x_\tau - x \rangle. \tag{2.2.1}$$

The role of those notions in the optimization context is explained next; our exposition follows [59].

### 2.2.2 Convex Minimization

**The problem.** Let $f$ be a Lipschitz continuous convex function on $X$. $f$ gives rise to the convex minimization problem

$$\mathrm{Opt} = \min_{x \in X} f(x) \tag{2.2.2}$$

and a vector field $F(x)$ specified (in general, non-uniquely) by $F(x) \in \partial f(x)$. It is well know that $F$ is *monotone* on its domain

$$\langle F(x) - F(y), x - y \rangle \geq 0, \ \ \forall x, y \in X.$$

Note that by definition of subgradient, we have for any $x, y$, $\langle F(x), x - y \rangle \geq f(x) - f(y)$ and similarly,$\langle F(y), y - x \rangle \geq f(y) - f(x)$. Summing up the two inequalities renders the monotonicity. In fact, $x_* \in X$ is an optimal solution to (2.2.2) if and only if

$$\langle F(y), y - x_* \rangle \geq 0 \ \forall y \in X.$$

**Accuracy measure.** We quantify the (in)accuracy of a candidate solution $x \in X$ for the convex minimization problem (2.2.2) by the *accuracy measure*

$$\epsilon_{\mathrm{opt}}(x) := f(x) - \mathrm{Opt}. \tag{2.2.3}$$

The role of accuracy certificate in convex minimization becomes clear from the following observation.

**Proposition 2.2.1.** *Let* $f : X \to \mathbf{R}$ *be a continuous convex function, and* $F$ *be the associated monotone vector field on* $X$. *Let* $\mathcal{I}_t = \{x_\tau \in X, F(x_\tau)\}_{\tau=1}^t$ *be a* $t$*-step execution protocol associated with* $(X, F)$ *and* $\lambda^t = \{\lambda_\tau^t\}_{\tau=1}^t$ *be an associated accuracy certificate. Then* $x^t := x^t(\mathcal{I}_t, \lambda^t) \in X$ *and one has*

$$\epsilon_{\mathrm{opt}}(x^t) \leq \mathrm{Res}(X|\mathcal{I}_t, \lambda^t). \tag{2.2.4}$$

*Proof.* Indeed, $x^t$ is a convex combination of the points $x_\tau \in X$ with coefficients $\lambda_\tau^t$, whence $x^t \in X$. We have

$$\begin{aligned}\forall y \in X : f(x^t) - f(y) &= f(\textstyle\sum_{\tau=1}^t \lambda_\tau^t x_\tau) - f(y) \le \sum_{\tau=1}^t \lambda_\tau^t [f(x_\tau) - f(y)] \\ &\quad [\text{by convexity and the fact that } \textstyle\sum_{\tau=1}^t \lambda_\tau^t = 1] \\ &\le \textstyle\sum_{\tau=1}^t \lambda_\tau^t \langle F(x_\tau), x_\tau - y\rangle \\ &\le \mathrm{Res}(X \big| \mathcal{I}_t, \lambda^t).\end{aligned}$$

Taking infinum over $y \in X$ in the resulting inequality, we get (2.2.4). □

### 2.2.3 Convex-Concave Saddle Point Problems

**The problem.** Now let $X = X_1 \times X_2$, where $X_i$ is a closed convex subset in Euclidean space $E_i$, $i = 1, 2$, and $E = E_1 \times E_2$, and let $\Phi(x^1, x^2) : X_1 \times X_2 \to \mathbf{R}$ be a locally Lipschitz continuous function which is convex in $x^1 \in X_1$ and concave in $x^2 \in X_2$. $X_1, X_2, \Phi$ give rise to the saddle point problem

$$\mathrm{SadVal} = \min_{x^1 \in X_1} \max_{x^2 \in X_2} \Phi(x^1, x^2), \tag{2.2.5}$$

two induced convex optimization problems

$$\begin{aligned}\mathrm{Opt}(P) &= \min_{x^1 \in X_1} \left[\overline{\Phi}(x^1) = \sup_{x^2 \in X_2} \Phi(x^1, x^2)\right] \quad (P) \\ \mathrm{Opt}(D) &= \max_{x^2 \in X_2} \left[\underline{\Phi}(x^2) = \inf_{x^1 \in X_1} \Phi(x^1, x^2)\right] \quad (D)\end{aligned} \tag{2.2.6}$$

and a vector field $F(x = [x^1, x^2]) = [F_1(x^1, x^2); F_2(x^1, x^2)]$ specified (in general, non-uniquely) by the relations

$$\forall (x^1, x^2) \in X_1 \times X_2 : F_1(x^1, x^2) \in \partial_{x^1} \Phi(x^1, x^2),\ F_2(x^1, x^2) \in \partial_{x^2}[-\Phi(x^1, x^2)].$$

It is well known that $F$ is monotone on $X$, and that saddle points $x_* = (x_*^1, x_*^2)$ of $\Phi$ on $X_1 \times X_2$ are exactly points $x^* \in X$ satisfying the relation

$$\langle F(y), y - x_* \rangle \ge 0 \ \forall y \in X.$$

Saddle points exist if and only if $(P)$ and $(D)$ are solvable with equal optimal values, in which case the saddle points are exactly the pairs $(x_*^1, x_*^2)$ comprised by optimal solutions to $(P)$ and $(D)$. In general, $\mathrm{Opt}(P) \ge \mathrm{Opt}(D)$, with equality definitely taking place when

at least one of the sets $X_1, X_2$ is bounded; if both are bounded, saddle points do exist. To avoid unnecessary complications, from now on, when speaking about a convex-concave saddle point problem, we assume that the problem is *proper*, meaning that $\mathrm{Opt}(P)$ and $\mathrm{Opt}(D)$ are reals; this definitely is the case when $X$ is bounded.

**Accuracy measure.** A natural (in)accuracy measure for a candidate $x = [x^1; x^2] \in X_1 \times X_2$ to the role of a saddle point of $\Phi$ is the quantity

$$\begin{aligned}\epsilon_{\mathrm{Sad}}(x\big|X_1, X_2, \Phi) &= \overline{\Phi}(x^1) - \underline{\Phi}(x^2) \\ &= [\overline{\Phi}(x^1) - \mathrm{Opt}(P)] + [\mathrm{Opt}(D) - \underline{\Phi}(x^2)] + \underbrace{[\mathrm{Opt}(P) - \mathrm{Opt}(D)]}_{\geq 0}\end{aligned} \tag{2.2.7}$$

This inaccuracy is nonnegative and is the sum of the duality gap $\mathrm{Opt}(P) - \mathrm{Opt}(D)$ (always nonnegative and vanishing when one of the sets $X_1, X_2$ is bounded) and the inaccuracies, in terms of respective objectives, of $x^1$ as a candidate solution to $(P)$ and $x^2$ as a candidate solution to $(D)$.

The role of accuracy certificates in convex-concave saddle point problems stems from the following observation:

**Proposition 2.2.2.** *Let $X_1, X_2$ be nonempty closed convex sets, $\Phi : X := X_1 \times X_2 \to \mathbf{R}$ be a locally Lipschitz continuous convex-concave function, and $F$ be the associated monotone vector field on $X$. Let $\mathcal{I}_t = \{x_\tau = [x_\tau^1; x_\tau^2] \in X, F(x_\tau)\}_{\tau=1}^t$ be a $t$-step execution protocol associated with $(X, F)$ and $\lambda^t = \{\lambda_\tau^t\}_{\tau=1}^t$ be an associated accuracy certificate. Then $x^t := x^t(\mathcal{I}_t, \lambda^t) = [x^{1,t}; x^{2,t}] \in X$.*

*Assume, further, that $X_1' \subset X_1$ and $X_2' \subset X_2$ are closed convex sets such that*

$$x^t \in X' := X_1' \times X_2'. \tag{2.2.8}$$

*Then*

$$\epsilon_{\mathrm{Sad}}(x^t\big|X_1', X_2', \Phi) = \sup_{x^2 \in X_2'} \Phi(x^{1,t}, x^2) - \inf_{x^1 \in X_1'} \Phi(x^1, x^{2,t}) \leq \mathrm{Res}(X'\big|\mathcal{I}_t, \lambda^t). \tag{2.2.9}$$

*In addition, setting $\widetilde{\Phi}(x^1) = \sup_{x^2 \in X_2'} \Phi(x^1, x^2)$, for every $\bar{x}^1 \in X_1'$ we have*

$$\widetilde{\Phi}(x^{1,t}) - \widetilde{\Phi}(\bar{x}^1) \leq \widetilde{\Phi}(x^{1,t}) - \Phi(\bar{x}^1, x^{2,t}) \leq \mathrm{Res}(\{\bar{x}^1\} \times X_2'\big|\mathcal{I}_t, \lambda^t). \tag{2.2.10}$$

*In particular, when the problem* $\mathrm{Opt} = \min_{x^1 \in X_1'} \widetilde{\Phi}(x^1)$ *is solvable with an optimal solution* $x_*^1$*, we have*

$$\widetilde{\Phi}(x^{1,t}) - \mathrm{Opt} \le \mathrm{Res}(\{x_*^1\} \times X_2' | \mathcal{I}_t, \lambda^t). \tag{2.2.11}$$

*Proof.* The inclusion $x^t \in X$ is evident. For every set $Y \subset X$ we have $\forall [p;q] \in Y$ :

$$\begin{aligned}
&\mathrm{Res}(Y|\mathcal{I}_t, \lambda^t) \ge \textstyle\sum_{\tau=1}^t \lambda_\tau^t \left[\langle F_1(x_\tau^1), x_\tau^1 - p\rangle + \langle F_2(x_\tau^2), x_\tau^2 - q\rangle\right] \\
&\ge \textstyle\sum_{\tau=1}^t \lambda_\tau^t \left[[\Phi(x_\tau^1, x_\tau^2) - \Phi(p, x_\tau^2)] + [\Phi(x_\tau^1, q) - \Phi(x_\tau^1, x_\tau^2)]\right] \\
&\text{[by the origin of } F \text{ and since } \Phi \text{ is convex-concave]} \\
&= \textstyle\sum_{\tau=1}^t \lambda_\tau^t \left[\Phi(x_\tau^1, q) - \Phi(p, x_\tau^2)\right] \ge \Phi(x^{1,t}, q) - \Phi(p, x^{2,t}) \\
&\qquad\qquad \text{[by origin of } x^t \text{ and since } \Phi \text{ is convex-concave]}
\end{aligned}$$

Thus, for every $Y \subset X$ we have

$$\sup_{[p;q] \in Y} \left[\Phi(x^{1,t}, q) - \Phi(p, x^{2,t})\right] \le \mathrm{Res}(Y|\mathcal{I}_t, \lambda^t). \tag{2.2.12}$$

Now assume that (2.2.8) takes place. Setting $Y = X' := X_1' \times X_2'$ and recalling what $\epsilon_{\mathrm{Sad}}$ is, (2.2.12) yields (2.2.9). With $Y = \{\bar{x}^1\} \times X_2'$, (2.2.12) yields the second inequality in (2.2.10); the first inequality in (2.2.10) is evident due to $x^{2,t} \in X_2'$. □

### 2.2.4 Convex Nash Equilibrium Problem

**The problem.** Now let $X = X_1 \times X_2 \times \cdots \times X_K$, where $X_k, 1 \le k \le K$ are closed and bounded convex sets in the respective Euclidean spaces $E_k, 1 \le k \le K$. A *convex Nash equilibrium problem* on $X$ is specified by a collection of $K$ Lipschitz continuous functions $f_k(x) : X \to \mathbf{R}$, $k = 1, ..., K$, such that for every $k$ $f_k = f_k(x[1], ..., x[K])$ is convex in $x[k] \in X_k$ and concave in

$$[x]^k = [x[1]; ...; x[k-1]; x[k+1]; ...; x[K]] \in X^k = X_1 \times ... \times X_{k-1} \times X_{k+1} \times ... \times X_K,$$

and besides this, the function $f(x) := \sum_{k=1}^K f_k(x)$ is convex in $x \in X$. The Nash equilibrium problem is to find a point $x_* \in X$ such that for every $k$ the function $f_k(x_*[1], ..., x_*[k-1], x_k, x_*[k+1], ..., x_*[K])$ attains its minimum over $x_k \in X_k$ at $x_k = x_*[k]$. A convex Nash equilibrium problem gives rise to the *Nash operator*, i.e. a vector field $F(x)$

$$F(x) = [F_1(x), F_2(x), \ldots, F_K(x)]$$

where $F_k(x) \in \partial_{x_k} f_k(x)$. It is well known (see, e.g., [59])that $F$ is monotone on the domain $X$ and the Nash equilibria are exactly the points $x_* \in X$ satisfying

$$\langle F(y), y - x_* \rangle \geq 0 \ \forall y \in X.$$

**Accuracy measure.** A natural way to quantify the inaccuracy of a point $x \in X$ as an approximate Nash equilibrium is given by the measure

$$\epsilon_{\text{Nash}}(x) := \sum_{k=1}^{K} \left[ f_k(x) - \min_{x_k \in X_k} f_k(x_k, [x]^k). \right]$$

In fact, the convex minimization problem with convex objective $f$ can be considered as a special case of convex Nash equilibrium problem with $K = 1$ and $f_1(x) = f(x)$, which results in $\epsilon_{\text{Nash}}(x) = \epsilon_{\text{opt}}(x)$. Similarly, the convex-concave saddle point problem given by $\Phi(x^1, x^2)$ can be regarded as a special case when of convex Nash equilibrium problem where $x[i] = x^i$, $i = 1, 2$, and $f_1(x) = \Phi(x)$, $f_2(x) = -\Phi(x)$; moreover, in this case, we have $\epsilon_{\text{Nash}}(x) = \epsilon_{\text{Sad}}(x \big| X_1, X_2, \Phi)$. The following result therefore is a natural generalization of Propositions 2.2.1 and 2.2.2:

**Proposition 2.2.3.** *Let a convex Nash equilibrium problem be as described above and $F$ be the associated Nash operator on $X$. Let $\mathcal{I}_t = \{x_\tau \in X, F(x_\tau)\}_{\tau=1}^t$ be a $t$-step execution protocol associated with $(X, F)$ and $\lambda^t = \{\lambda_\tau^t\}_{\tau=1}^t$ be an associated accuracy certificate. Then $x^t := x^t(\mathcal{I}_t, \lambda^t) \in X$ and one has*

$$\epsilon_{\text{Nash}}(x^t) \leq \text{Res}(X \big| \mathcal{I}_t, \lambda^t). \tag{2.2.13}$$

*Proof.* The inclusion $x^t \in X$ is evident. We have $\forall y \in X$,

$$\begin{aligned}
\textstyle\sum_{k=1}^K \left[ f_k(x^t) - f_k(y_k, [x^t]^k) \right] &= \textstyle\sum_{k=1}^K \left[ f_k(\sum_{\tau=1}^t \lambda_\tau^t x_\tau) - f_k(y_k, \sum_{\tau=1}^t \lambda_\tau^t [x_\tau]^k) \right] \\
&\leq \textstyle\sum_{k=1}^K \sum_{\tau=1}^t \lambda_\tau^t \left[ f_k(x_\tau) - f_k(y_k, [x_\tau]^k) \right] \\
&\quad \text{[by convexity of } f \text{ and the concavity of } f_k(y_k, \cdot)] \\
&\leq \textstyle\sum_{k=1}^K \sum_{\tau=1}^t \lambda_\tau^t \langle F_k(x_\tau), x_\tau[k] - y_k \rangle \\
&\quad \text{[by convexity of } f_k(\cdot, [x_\tau]^k) \text{ and origin of } F_k] \\
&\leq \textstyle\sum_{\tau=1}^t \lambda_\tau^t \left[ \sum_{k=1}^K \langle F_k(x_\tau), x_\tau[k] - y_k \rangle \right] \\
&\leq \textstyle\sum_{\tau=1}^t \lambda_\tau^t \langle F(x_\tau), x_\tau - y \rangle \quad \text{[by the origin of } F] \\
&\leq \text{Res}(X \big| \mathcal{I}_t, \lambda^t).
\end{aligned}$$

Hence, $\epsilon_{\text{Nash}}(x^t) \leq \text{Res}(X\big|\mathcal{I}_t, \lambda^t)$. □

### 2.2.5 Variational Inequalities with Monotone Operators

The three types of optimization problems considered so far (convex minimization, convex-concave saddle point, and convex Nash equilibrium) are special cases of variational inequalities with monotone operators.

**Variational inequality with monotone operator.** Let $X$ be a closed and convex set and vector field $F$ be monotone on $X$, i.e.,

$$\langle F(x) - F(y), x - y\rangle \geq 0, \ \ \forall x, y \in X \tag{2.2.14}$$

The *variational inequality* problem associated with $(X, F)$, denoted as $\text{VI}(X, F)$, is to find $x_* \in X$ such that

$$\langle F(y), y - x_*\rangle \geq 0 \ \forall y \in X; \tag{2.2.15}$$

these $x_*$ are called weak solutions to the variational inequality. In contrast, a strong solution is a point $x_* \in X$ such that $\langle F(x_*), y - x_*\rangle \geq 0 \ \forall y \in X$. Note that for variational inequality with monotone operators, a strong solution is also a weak solution. The inverse is true under mild regularity assumption, e.g. when $F$ is continuous. Finally, when $X$ is convex and compact and $F$ is monotone, weak solutions to $\text{VI}(X, F)$ always exist.

**Accuracy measure.** A natural (in)accuracy measure of a point $x \in X$ to $\text{VI}(X, F)$ as a candidate weak solution is the *dual gap function*

$$\epsilon_{\text{VI}}(x\big|X, F) = \sup_{y \in X} \langle F(y), x - y\rangle \tag{2.2.16}$$

This inaccuracy is a convex nonnegative function which vanishes exactly at the set of weak solutions to the $\text{VI}(X, F)$ .

**Proposition 2.2.4.** *Let VI$(X, F)$ be the variational inequality with monotone operator $F$ and closed convex set $X$. For every $t$, every execution protocol $\mathcal{I}_t = \{x_\tau \in X, F(x_\tau)\}_{\tau=1}^t$ and every accuracy certificate $\lambda^t$ one has $x^t := x^t(\mathcal{I}_t, \lambda^t) \in X$. For every closed convex set*

*$X' \subset X$ such that $x^t \in X'$ one has*

$$\epsilon_{\mathrm{VI}}(x^t \big| X', F) \le \mathrm{Res}(X' \big| \mathcal{I}_t, \lambda^t). \tag{2.2.17}$$

*Proof.* Indeed, $x^t$ is a convex combination of the points $x_\tau \in X$ with coefficients $\lambda_\tau^t$, whence $x^t \in X$. With $X'$ as in the premise of Proposition, we have

$$\forall y \in X' : \langle F(y), x^t - y\rangle = \sum_{\tau=1}^t \lambda_\tau^t \langle F(y), x_\tau - y\rangle \le \sum_{\tau=1}^t \lambda_\tau^t \langle F(x_\tau), x_\tau - y\rangle \le \mathrm{Res}(X' \big| \mathcal{I}_t, \lambda^t),$$

where the first $\le$ is due to monotonicity of $F$. □

To summarize, throughout this section, we have associated the four types of problems with convex structure – convex minimization, convex-concave saddle points, convex Nash equilibrium problems, and variational inequalities with monotone operators, with respective accuracy measures. We have also associated with every one of these problems a monotone vector field $F$ on problem's domain $X$ and have seen that exact solutions to the problems are nothing but weak solutions of the resulting $\mathrm{VI}(X, F)$. Moreover, we have shown that for every execution protocol for $(X, F)$, an accuracy certificate for the protocol induces a feasible solution to the problem of interest, and the resolution of this protocol upper-bounds the respective inaccuracy of this solution.

## *2.3 Problems with Special Convex Structure*

### 2.3.1 The Situation

**The situation.** Let $U$ be a nonempty closed convex domain in a Euclidean space $E_u$, $E_v$ be a Euclidean space, and $X$ be a nonempty closed convex domain in $E = E_u \times E_v$. We denote vectors from $E$ by $x = [u; v]$ with blocks $u, v$ belonging to $E_u$ and $E_v$, respectively.

We assume that

(**A**.1): $E_u$ is equipped with a norm $\|\cdot\|$, the conjugate norm being $\|\cdot\|_*$, and $U$ is equipped with a *distance-generating function* (d.g.f.) $\omega(\cdot)$ (that is, with a continuously differentiable convex function $\omega(\cdot) : U \to \mathbf{R}$) which is *compatible* with $\|\cdot\|$, meaning that $\omega$ is strongly convex, modulus 1, w.r.t. $\|\cdot\|$.

Note that d.g.f. $\omega$ defines the *Bregman distance*

$$V_u(w) := \omega(w) - \omega(u) - \langle \omega'(u), w - u \rangle \geq \frac{1}{2}\|w - u\|^2,\ u, w \in U, \tag{2.3.1}$$

where the concluding inequality follows from strong convexity, modulus 1, of the d.g.f. w.r.t. $\|\cdot\|$.

In the sequel, we refer to the pair $\|\cdot\|$, $\omega(\cdot)$ as to *proximal setup* for $U$.

(**A**.2): the image $PX$ of $X$ under the projection $x = [u; v] \mapsto Px := u$ is contained in $U$.

(**A**.3): we are given a vector field $F(u, v) : X \to E$ on $X$ of the special structure as follows:

$$F(u, v) = [F_u(u); F_v],$$

with $F_u(u) \in E_u$ and $F_v \in E_v$. Note that $F$ is independent of $v$.

We assume also that

$$\forall u, u' \in U : \|F_u(u) - F_u(u')\|_* \leq L\|u - u'\| + M \tag{2.3.2}$$

with some $L < \infty$, $M < \infty$.

(**A**.4): the linear form $\langle F_v, v \rangle$ of $[u; v] \in E$ is bounded from below on $X$ and is coercive on $X$ w.r.t. $v$: whenever $[u_t; v_t] \in X$, $t = 1, 2, ...$ is a sequence such that $\{u_t\}_{t=1}^{\infty}$ is bounded and $\|v_t\|_2 \to \infty$ as $t \to \infty$, we have $\langle F_v, v_t \rangle \to \infty$, $t \to \infty$.

**Our goal** in this chapter is to show that *in the situation in question, proximal type processing* $F$ (say, $F$ is monotone on $X$, and we want to solve the variational inequality given by $F$ and $X$) *can be implemented "as if" there were no* $v$*-components in the domain and in* $F$.

### 2.3.2 Example I: Composite Nash Equilibrium Problem

Consider the case when $u$ is split into $K$ consecutive blocks: $u = [u[1]; ...; u[K]]$, and similarly for $v$: $v = [v[1]; ...; v[K]]$. For $x = [u, v]$, let us set $x[k] = [u[k]; v[k]]$.

Let

$$X = \{x = [u; v] : x[k] := [u[k]; v[k]] \in X_k, 1 \leq k \leq K\},$$

where $X_k$, $1 \le k \le K$, are closed convex sets in the respective orthogonal to each other linear subspaces $E_k$ of $E = \mathbf{R}^{n_u+n_v}$. Consider a *convex Nash equilibrium problem* given by $K$ Lipschitz continuous functions $f_k(x) : X \to \mathbf{R}$, $k = 1, ..., K$, such that for every $k$ $f_k = f_k(x[1], ..., x[K])$ is convex in $x[k] \in X_k$ and concave in

$$[x]^k = [x[1]; ...; x[k-1]; x[k+1]; ...; x[K]] \in X^k = X_1 \times ... \times X_{k-1} \times X_{k+1} \times ... \times X_K,$$

and besides this, the function $f(x) := \sum_{k=1}^K f_k(x)$ is convex in $x \in X$. In the sequel we will denote by $[u]^k$, $[v]^k$ entities obtained from $u$, resp., $v$ in exactly the same way as $[x]^k$ is obtained from $x$.

Recall that the Nash equilibrium problem is to find $x \in X$ such that for every $k$ the function $f_k(x[1], ..., x[k-1], x_k, x[k+1], ..., x[K])$ attains its minimum over $x_k \in X_k$ at $x_k = x[k]$. A natural way to quantify the inaccuracy of a point $x \in X$ as an approximate Nash equilibrium is given by the measure

$$\epsilon_{\text{Nash}}(x) := \sum_{k=1}^K \left[ f_k(x) - \min_{x_k \in X_k} f(x_k, [x]^k). \right]$$

Assume that the functions $f_k$ in the Nash equilibrium problem possess the following specific structure:

$$f_k(x = [u; v]) = \phi_k(u) + \sum_{\ell=1}^K \langle b_\ell^k, v[\ell] \rangle.$$

Here $\phi_k(u)$ are Lipschitz continuous functions on the set $PX = (P_1X_1) \times ... \times (P_KX_K)$, where $P_k \cdot [u[k]; v[k]] = u[k]$. Similarly to the case of functions $f_k$, we will use the notation $\phi_k(u[k], [u]^k)$ as an equivalent form of $\phi_k(u)$. Let $\phi_k'(u)$ be a subgradient of $\phi_k(u = [u[1]; ...; u[K]])$ with respect to $u[k]$, and let us set

$$F_k(x = [u; v]) = [\phi_k'(u); b_k^k],\ k = 1, ..., K.$$

We defined the Nash operator $F(x = [u, v])$ on $X$ given the collection $F_k(\cdot)$, $k = 1, ..., K$, by $(F(x))[k] = F_k(x)$, $1 \le k \le K$, and this field clearly is of the form

$$F(x) = [F_u(u); F_v].$$

Assuming that $F_u(\cdot)$ is bounded on $PX$: $\|F_u(u)\|_* \le M < \infty, \forall u \in PX$ and that the linear function $\langle F_v, v \rangle$ of $x = [u; v]$ is below bounded on $X$, then we are exactly in the situation

described in Section 2.3.1 with assumption (**A**.3) satisfied by $L = 0$. In this case, our goal hence, is to solve the above Nash equilibrium problem as if there were no linear terms in the functions.

### 2.3.3 Example II: Composite Saddle Point Problem

Consider the "composite" saddle point problem

$$\mathrm{SadVal} = \min_{u_1 \in U_1} \max_{u_2 \in U_2} \left[\phi(u_1, u_2) + \Psi_1(u_1) - \Psi_2(u_2)\right], \tag{2.3.3}$$

where

- $U_1 \subset E_1$ and $U_2 \subset E_2$ are nonempty closed convex sets in Euclidean spaces $E_1, E_2$
- $\phi$ is a smooth (with Lipschitz continuous gradient) convex-concave function on $U_1 \times U_2$
- $\Psi_1 : U_1 \to \mathbf{R}$ and $\Psi_2 : U_2 \to \mathbf{R}$ are convex functions, perhaps nonsmooth, but "fitting" the domains $U_1$, $U_2$ in the following sense: for $i = 1, 2$, we can equip $E_i$ with a norm $\|\cdot\|_{(i)}$, and $U_i$ - with a compatible with this norm d.g.f. $\omega_i(\cdot)$ in such a way that optimization problems of the form

$$\min_{u_i \in U_i} \left[\alpha\omega_i(u_i) + \beta\Psi_i(u_i) + \langle \xi, u_i\rangle\right] \qquad [\alpha > 0, \beta > 0] \tag{2.3.4}$$

  are easy to solve.

We act as follows:

- For $i = 1, 2$, we set $X_i = \{x_i = [u_i; v_i] \in E_i \times \mathbf{R} : u_i \in U_i, v_i \geq \Psi_i(u_i)\}$ and set

$$U := U_1 \times U_2 \subset E_u := E_1 \times E_2, E_v = \mathbf{R}^2,$$
$$X = \{x = [u = [u_1; u_2]; v = [v_1; v_2]] : u_i \in U_i, v_i \geq \Psi_i(u_i), i = 1, 2\} \subset E_u \times E_v,$$

  thus ensuring that $PX \subset U$, where $P[u; v] = u$;

- We rewrite the problem of interest equivalently as

$$\mathrm{SadVal} = \min_{x^1 = [u_1; v_1] \in X_1} \max_{x^2 = [u_2; v_2] \in X_2} [\Phi(u_1, v_1; u_2, v_2) = \phi(u_1, u_2) + v_1 - v_2] \tag{2.3.5}$$

Note that $\Phi$ is convex-concave and smooth. The associated monotone operator is

$$F(u=[u_1;u_2],v=[v_1;v_2])=[F_u(u)=[\nabla_{u_1}\phi(u_1,u_2);-\nabla_{u_2}\phi(u_1,u_2)];F_v=[1;1]]$$

and is of the structure required in (**A**.3). Note that $F$ is Lipschitz continuous, so that (2.3.2) is satisfied with properly selected $L$ and with $M=0$.

Hence, we are exactly in the situation described in Section 2.3.1 with assumption (**A**.3) satisfied by $M=0$. In this case, our goal, is to solve the above composite saddle point problem as if there were no (perhaps) nonsmooth terms $\Psi_i$.

**Remark.** We intend to process the reformulated saddle point problem (2.3.5) with a properly modified state-of-the-art Mirror Prox (MP) algorithm [56]. In its basic version and as applied to a variational inequality with Lipschitz continuous monotone operator (in particular, to a convex-concave saddle point problem with smooth cost function), this algorithm exhibits $O(1/t)$ rate of convergence, which is the best rate achievable with First Order saddle point algorithms as applied to large-scale saddle point problems (even those with bilinear cost function). However, the basic MP would require to equip the domain $X=X_1\times X_2$ of (2.3.5) with a d.g.f. $\omega(x_1,x_2)$ resulting in auxiliary problems of the form

$$\min_{x=[u_1;u_2;v_1;v_2]\in X}\left[\omega(x)+\langle\xi,x\rangle\right]. \tag{2.3.6}$$

This would require to account in $\omega$, in a nonlinear fashion, for the $v$-variables (since $\omega$ should be a strongly convex in both $u$- and $v$-variables). While it is easy to construct $\omega$ from our postulated "building blocks" $\omega_1$, $\omega_2$ leading to easy-to-solve problems (2.3.4), this construction results in auxiliary problems (2.3.6) somehow more complicated than problems (2.3.4). To overcome this difficulty, below we develop a "composite" Mirror Prox algorithm taking advantage of the special structure of $F$, as expressed in (**A**.3), and preserving the favorable efficiency estimates of the prototype. The modified algorithm operates with the auxiliary problems of the form

$$\min_{x=[u_1;u_2;v_1;v_2]\in X_1\times X_2}\sum_{i=1}^{2}\left[\alpha_i\omega_i(u_i)+\beta_i v_i+\langle\xi_i,u_i\rangle\right], \qquad [\alpha_i>0,\beta_i>0]$$

that is, with pairs of uncoupled problems

$$\min_{[u_i;v_i]\in X_i} \left[\alpha_i\omega_i(u_i) + \beta_i v_i + \langle \xi_i, u_i\rangle\right],\ i = 1, 2;$$

recalling that $X_i = \{[u_i; v_i] : u_i \in U_i, v_i \geq \Psi_i(u_i)\}$, these problems are nothing but the easy-to-solve problems (2.3.4).

## *2.4 Composite Mirror Descent*

In the rest of this chapter, unless otherwise is stated explicitly, we stay in the situation described in Section 2.3.1, with Assumptions (**A**.1) – (**A**.4) in force.

In this section, we first focus on the case when $L = 0$ in Assumption (**A**.3), namely, the vector field $F$ is only assumed to be bounded. Our goal is to develop a composite version of Mirror Descent algorithm, which works as if there were no $v$-component and still enjoys the usual efficiency estimate.

**Prox-mapping.** Given the situation described in previous section, we define the associated *prox-mapping*: for $\xi = [\eta; \zeta] \in E$ and $x = [u; v] \in X$,

$$\begin{aligned} P_x(\xi) &\in \operatorname*{Argmin}_{[s;w]\in X} \left\{\langle \eta - \omega'(u), s\rangle + \langle \zeta, w\rangle + \omega(s)\right\} \\ &\equiv \operatorname*{Argmin}_{[s;w]\in X} \left\{\langle \eta, s\rangle + \langle \zeta, w\rangle + V_u(s)\right\} \end{aligned} \tag{2.4.1}$$

Observe that $P_x([\eta; \gamma F_v])$ is well defined whenever $\gamma > 0$ – the required Argmin is nonempty due to the strong convexity of $\omega$ on $U$ and assumption (**A**.4). We verify this below.

**Lemma 2.4.1.** *For any $x = [u; v] \in X$ and $\xi = [\eta; \zeta] \in E$, the prox-mapping $P_x([\eta; \gamma F_v])$ is well-defined, provided $\gamma > 0$.*

*Proof.* All we need is to show that whenever $u \in U$, $\eta \in E_u$, $\gamma > 0$ and $[w_t; s_t] \in X$, $t = 1, 2, ...$, are such that $\|w_t\|_2 + \|s_t\|_2 \to \infty$ as $t \to \infty$, we have

$$r_t := \underbrace{\langle \eta - \omega'(u), w_t\rangle + \omega(w_t)}_{a_t} + \underbrace{\gamma\langle F_v, s_t\rangle}_{b_t} \to \infty,\ t \to \infty.$$

Indeed, assuming the opposite and passing to a subsequence, we make the sequence $r_t$ bounded. Since $\omega(\cdot)$ is strongly convex, modulus 1, w.r.t. $\|\cdot\|$, and the linear function

$\langle F_v, s\rangle$ of $[w; s]$ is below bounded on $X$ by (**A**.4), boundedness of the sequence $\{r_t\}$ implies boundedness of the sequence $\{w_t\}$, and since $\|[w_t; s_t]\|_2 \to \infty$ as $t \to \infty$, we get $\|s_t\|_2 \to \infty$ as $t \to \infty$. Since $\langle F_v, s\rangle$ is coercive in $s$ on $X$ by (**A**.4), and $\gamma > 0$, we conclude that $b_t \to \infty$, $t \to \infty$, while the sequence $\{a_t\}$ is bounded since the sequence $\{w_t \in U\}$ is so and $\omega$ is continuously differentiable. Thus, $\{a_t\}$ is bounded, $b_t \to \infty$, $t \to \infty$, implying that $r_t \to \infty$, $t \to \infty$, which is the desired contradiction. □

**Composite Mirror Descent algorithm** is as follows.

---

**Algorithm 1** Composite Mirror Descent Algorithm for VI$(X, F)$

---

**Input:** stepsizes $\gamma_\tau > 0$, inexactness $\epsilon_\tau \geq 0$, $\tau = 1, 2, \ldots$

Initialize $x_1 = [u_1; v_1] \in X$

**for** $\tau = 1, 2, \ldots, t$ **do**

$$x_{\tau+1} := [u_{\tau+1}; v_{\tau+1}] \in P_{x_\tau}(\gamma_\tau F(x_\tau)) = P_{x_\tau}(\gamma_\tau [F_u(u_\tau); F_v]) \tag{2.4.2}$$

**end for**

**Output:** $x^{t+1} := [u^{t+1}; v^{t+1}] = \left(\sum_{\tau=1}^t \gamma_\tau\right)^{-1} \sum_{\tau=1}^t \gamma_\tau x_{\tau+1}$

---

Note that since $\gamma_\tau > 0$, the recurrence in (2.4.2) is well-defined by Lemma 2.4.1. Also, by construction, $x_\tau \in X$ for all $t$, whence, the output $x^{t+1} \in X$ for all $t$ as well. The following lemma is a simple consequence of the optimality condition of the problem (2.4.1).

**Lemma 2.4.2.** *For any $x = [u; v] \in X$ and $\xi = [\eta; \zeta] \in E$, let $[u'; v'] = P_x(\xi)$, we have for all $[s; w] \in X$,*

$$\langle \eta, u' - s\rangle + \langle \zeta, v' - w\rangle \leq V_u(s) - V_{u'}(s) - V_u(u'). \tag{2.4.3}$$

*Proof.* Recall the well-known identity [24]: for all $u, u', w \in U$ one has

$$\langle V_u'(u'), w - u'\rangle = V_u(w) - V_{u'}(w) - V_u(u'). \tag{2.4.4}$$

Indeed, the right hand side is

$$[\omega(w) - \omega(u) - \langle \omega'(u), w - u\rangle] - [\omega(w) - \omega(u') - \langle \omega'(u'), w - u'\rangle] - [\omega(u') - \omega(u) - \langle \omega'(u), u' - u\rangle]$$

$$= \langle \omega'(u), u - w\rangle + \langle \omega'(u), u' - u\rangle + \langle \omega'(u'), w - u'\rangle = \langle \omega'(u') - \omega'(u), w - u'\rangle = \langle V_u'(u'), w - u'\rangle.$$

For $x = [u; v] \in X$, $\xi = [\eta; \zeta]$, let $P_x(\xi) = [u'; v'] \in X$. By the optimality condition for the problem (2.4.1), for all $[s; w] \in X$,

$$\langle \eta + V_u'(u'), u' - s\rangle + \langle \zeta, v' - w\rangle \le 0,$$

which by (2.4.4) implies that

$$\langle \eta, u' - s\rangle + \langle \zeta, v' - w\rangle \le \langle V_u'(u'), s - u'\rangle = V_u(s) - V_{u'}(s) - V_u(u').$$

□

**Theorem 2.4.1.** *Assue we are in the situation of Section 2.3.1 and under assumptions (**A**.1) –(**A**.4) and $L = 0$ in (**A**.3), i.e. $\|F_u(u)\|_* \le M < \infty, \forall u \in PX$. In the case when $F_u$ is monotone operator, we have*

$$\epsilon_{\mathrm{VI}}(x^{t+1}\big|X, F) \le \left[\sum_{\tau=1}^{t} \gamma_\tau\right]^{-1} \left[\max_{u\in PX} V_{u_1}(u) + 2M^2 \sum_{\tau=1}^{t} \gamma_\tau^2\right]. \tag{2.4.5}$$

*Proof.* Assume that $F_u(u)$ is monotone on $PX$, so that $F$ is monotone on $X$. When applying Lemma 2.4.2 with $[u; v] = [u_\tau; v_\tau]$, $[\eta; \zeta] = [\gamma_\tau F_u(u_\tau); \gamma_\tau F_v]$ and $[u'; v'] = [u_{\tau+1}; v_{\tau+1}]$, we obtain for any $z = [s; w] \in X$

$$\gamma_\tau \left[\langle F_u(u_\tau), u_{\tau+1} - s\rangle + \langle F_v, v_{\tau+1} - w\rangle\right] \le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - V_{u_\tau}(u_{\tau+1}) \tag{2.4.6}$$

Taking into account strong convexity of $\omega(\cdot)$ and monotonicity of $F$, we end up with

$$\begin{aligned}
&\gamma_\tau \left[\langle F_u(s), u_\tau - s\rangle + \langle F_v, v_{\tau+1} - w\rangle\right] \\
&\qquad \le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + \gamma_\tau \langle F_u(u_\tau), u_\tau - u_{\tau+1}\rangle,
\end{aligned}$$

whence

$$\begin{aligned}
&\gamma_\tau \langle F(z), x_{\tau+1} - z\rangle = \gamma_\tau \left[\langle F_u(s), u_{\tau+1} - s\rangle + \langle F_v, v_{\tau+1} - w\rangle\right] \\
&\le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + \gamma_\tau \langle F_u(s) - F_u(u_\tau), u_{\tau+1} - u_\tau\rangle \\
&\le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + 2\gamma_\tau M \|u_{\tau+1} - u_\tau\| \\
&\le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + 2\gamma_\tau^2 M^2,
\end{aligned}$$

whence

$$\gamma_\tau \langle F(z), x_{\tau+1} - z\rangle \le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + 2\gamma_\tau^2 M^2. \tag{2.4.7}$$

Summing up inequalities (2.4.7) over $t$, we conclude that for every $z = [s; w] \in X$ it holds

$$\langle F(z), x^{t+1} - z\rangle \leq \left[\sum_{\tau=1}^{t} \gamma_\tau\right]^{-1} \left[\max_{u\in PX} V_{u_1}(u) + 2M^2 \sum_{\tau=1}^{t} \gamma_\tau^2\right].$$

Taking maximum over $z \in X$, we end up with the desired bound. □

**Remark.** The composite Mirror Descent algorithm inherits the efficiency estimate of its prototype. In particular, when the stepsize is set to be $\gamma_\tau = \frac{\Omega}{\sqrt{2}M\sqrt{t}}$ where $\Omega^2 = \max_{u\in PX} V_{u_1}(u)$, then $\epsilon_{\mathrm{VI}}(x^{t+1}\big|X, F) \leq \frac{\sqrt{2}\Omega M}{\sqrt{t}}$. Similar results can be obtained for Nash equilibrium problems.

**Nash equilibrium problem** Recall that the Nash equilibrium problem described in Section 2.3.2 and the induced variational inequality. We can therefore apply to the problem the above algorithm.

**Theorem 2.4.2.** *Let the Nash equilibrium problem be as described in Section 2.3.2. Assume that $\phi_k(u)$ is $M_k$-Lipschitz continuous on $PX$ for $k = 1, \ldots, K$ w.r.t. $\|\cdot\|$. Assume that $\|F_u(u)\|_* \leq M < \infty, \forall u \in PX$. The candidate solution $x^{t+1}$ provided by the above Mirror Descent algorithm satisfies the efficiency estimate*

$$\epsilon_{\mathrm{Nash}}(x^{t+1}) \leq \frac{\max_{u\in PX} V_{u_1}(u) + \frac{1}{2}\mathcal{M}^2 \sum_{\tau=1}^{t} \gamma_\tau^2}{\sum_{\tau=1}^{t} \gamma_\tau}, \tag{2.4.8}$$

*where $\mathcal{M} = M + \sum_{k=1}^{K} 2M_k < \infty$.*

*Proof.* The relation (2.4.6) which now reads, for all $\bar{x} := [\bar{u}; \bar{v}] \in X$

$$\begin{aligned}
&\gamma_\tau \textstyle\sum_{k=1}^{K} \left[\langle \phi_k'(u_\tau), u_{\tau+1}[k] - \bar{u}[k]\rangle + \langle b_k^k, v_{\tau+1}[k] - \bar{v}[k]\rangle\right] \leq V_{u_\tau}(\bar{u}) - V_{u_{\tau+1}}(\bar{u}) - V_{u_\tau}(u_{\tau+1}) \\
&\Rightarrow \gamma_\tau \textstyle\sum_{k=1}^{K} \left[\langle \phi_k'(u_\tau), u_\tau[k] - \bar{u}[k]\rangle + \langle b_k^k, v_{\tau+1}[k] - \bar{v}[k]\rangle\right] \\
&\qquad \leq V_{u_\tau}(\bar{u}) - V_{u_{\tau+1}}(\bar{u}) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + \gamma_\tau \langle F_u(u_\tau), u_\tau - u_{\tau+1}\rangle.
\end{aligned} \tag{2.4.9}$$

Note that the convexity-concavity properties of $f_k$ imply that the function $\phi_k(u) \equiv \phi_k(u[k], [u]^k)$, is convex in $u[k]$ and concave in $[u]^k$. By the former fact, we have

$$\langle \phi_k'(u_\tau), u_\tau[k] - \bar{u}[k]\rangle \geq \phi_k(u_\tau) - \phi_k(\bar{u}[k], [u_\tau]^k),$$

and thus, due to Lipschitz continuity of $\phi_k$ on $PX$,

$$\langle \phi_k'(u_\tau), u_\tau[k] - \bar{u}[k] \rangle \geq \phi_k(u_{\tau+1}) - \phi_k(\bar{u}[k], [u_{\tau+1}]^k) - 2M_k \|u_\tau - u_{\tau+1}\|$$

with properly defined $M_k < \infty$. This combines with (2.4.9) to imply that

$$\begin{aligned}
&\gamma_\tau \textstyle\sum_{k=1}^K \left[ f_k(x_{\tau+1}) - f_k(\bar{x}[k], [x_{\tau+1}]^k) \right] \\
&= \gamma_\tau \textstyle\sum_{k=1}^K \left[ \phi_k(u_{\tau+1}) - \phi_k(\bar{u}[k], [u_{\tau+1}]^k) + \langle b_k^k, v_{\tau+1}[k] - \bar{v}[k] \rangle \right] \\
&\leq V_{u_\tau}(\bar{u}) - V_{u_{\tau+1}}(\bar{u}) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + \gamma_\tau \langle F_u(u_\tau), u_\tau - u_{\tau+1} \rangle + \gamma_\tau \textstyle\sum_{k=1}^K 2M_k \|u_\tau - u_{\tau+1}\| \\
&\leq V_{u_\tau}(\bar{u}) - V_{u_{\tau+1}}(\bar{u}) - \tfrac{1}{2}\|u_{\tau+1} - u_\tau\|^2 + \gamma_\tau [M + \textstyle\sum_{k=1}^K 2M_k] \|u_\tau - u_{\tau+1}\| \\
&\leq V_{u_\tau}(\bar{u}) - V_{u_{\tau+1}}(\bar{u}) + \tfrac{1}{2}\gamma_\tau^2 \mathcal{M}^2.
\end{aligned}$$

Summing up the resulting inequalities over $\tau$, we get

$$\sum_{\tau=1}^t \gamma_\tau \sum_{k=1}^K \left[ f_k(x_{\tau+1}) - f_k(\bar{x}[k], [x_{\tau+1}]^k) \right] \leq \max_{u \in PX} V_{u_1}(u) + \frac{1}{2}\mathcal{M}^2 \sum_{\tau=1}^t \gamma_\tau^2.$$

Recalling that $f(x) = \sum_{k=1}^K f_k(x)$ is convex on $X$, while $f_k(x[k], [x]^k)$ is concave in $[x]^k$, we have

$$\sum_{\tau=1}^t \gamma_\tau \sum_{k=1}^K \left[ f_k(x_{\tau+1}) - f_k(\bar{x}[k], [x_{\tau+1}]^k) \right] \geq \left[ \sum_{\tau=1}^t \gamma_\tau \right] \sum_{k=1}^K \left[ f_k(x^{t+1}) - f_k(\bar{x}[k], [x^{t+1}]^k) \right],$$

hence we get

$$\sum_{k=1}^K \left[ f_k(x^{t+1}) - f_k(\bar{x}[k], [x^{t+1}]^k) \right] \leq \frac{\max_{u \in PX} V_{u_1}(u) + \frac{1}{2}\mathcal{M}^2 \sum_{\tau=1}^t \gamma_\tau^2}{\sum_{\tau=1}^t \gamma_\tau}.$$

Taking maximum of the left hand side in $\bar{x} \in X$, we finally get (2.4.8). □

**Remark.** When applied to the above convex Nash equilibrium problem, the composite Mirror Descent algorithm inherits with properly selected stepsizes the $O(1/\epsilon^2)$ efficiency estimate of its prototype.

## *2.5 Composite Mirror Prox*

### 2.5.1 Composite Mirror Prox: basic algorithm

In the following sections, we focus on the general case when $L \neq 0$ in assumption (**A**.3) and develop a composite version of Mirror Prox algorithm. The algorithm is as follows:

**Algorithm 2** Composite Mirror Prox Algorithm (CoMP) for VI$(X, F)$

**Input:** stepsizes $\gamma_\tau > 0$, $\tau = 1, 2, \ldots$

Initialize $x_1 = [u_1; v_1] \in X$

**for** $\tau = 1, 2, \ldots, t$ **do**

$$\begin{array}{rcl} y_\tau := [u'_\tau; v'_\tau] & = & P_{x_\tau}(\gamma_\tau F(x_\tau)) = P_{x_\tau}(\gamma_\tau [F_u(u_\tau); F_v]) \\ x_{\tau+1} := [u_{\tau+1}; v_{\tau+1}] & = & P_{x_\tau}(\gamma_\tau F(y_\tau)) = P_{x_\tau}(\gamma_\tau [F_u(u'_\tau); F_v]) \end{array} \tag{2.5.1}$$

**end for**

**Output:** $x^t := [u^t; v^t] = \left(\sum_{\tau=1}^t \gamma_\tau\right)^{-1} \sum_{\tau=1}^t \gamma_\tau y_\tau$

Observe that the process is well defined by Lemma 2.4.1. From now on, for a subset $X'$ of $X$ we set

$$\Theta[X'] = \sup_{[u;v] \in X'} V_{u_1}(u). \tag{2.5.2}$$

We arrive at the following results.

**Theorem 2.5.1.** *In the setting of Section 2.3.1, assuming that (**A**.1)–(**A**.4) hold, consider the recurrence (2.5.1) with stepsizes $\gamma_\tau > 0$, $\tau = 1, 2, \ldots$ satisfying the relation:*

$$\delta_\tau := \gamma_\tau \langle F_u(u'_\tau) - F_u(u_\tau), u'_\tau - u_{\tau+1} \rangle - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) \le \gamma_\tau^2 M^2. \tag{2.5.3}$$

*Then the corresponding execution protocol $\mathcal{I}_t = \{y_\tau, F(y_\tau)\}_{\tau=1}^t$ admits accuracy certificate $\lambda^t = \{\lambda_\tau^t = \gamma_\tau / \sum_{i=1}^t \gamma_i\}$ such that for every $X' \subset X$ it holds*

$$\mathrm{Res}(X' | \mathcal{I}_t, \lambda^t) \le \frac{\Theta[X'] + M^2 \sum_{\tau=1}^t \gamma_\tau^2}{\sum_{\tau=1}^t \gamma_\tau}. \tag{2.5.4}$$

*Relation* (2.5.3) *is definitely satisfied when $0 < \gamma_\tau \le (\sqrt{2}L)^{-1}$, or, in the case of $M = 0$, when $\gamma_\tau \le L^{-1}$.*

*Proof.* When applying Lemma 2.4.2 with $[u; v] = [u_\tau; v_\tau] = x_\tau$, $\xi = \gamma_\tau F(x_\tau) = [\gamma_\tau F_u(u_\tau); \gamma_\tau F_v]$, $[u'; v'] = [u'_\tau; v'_\tau] = y_\tau$, and $[s; w] = [u_{\tau+1}; v_{\tau+1}] = x_{\tau+1}$ we obtain:

$$\gamma_\tau [\langle F_u(u_\tau), u'_\tau - u_{\tau+1} \rangle + \langle F_v, v'_\tau - v_{\tau+1} \rangle] \le V_{u_\tau}(u_{\tau+1}) - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) \tag{2.5.5}$$

and applying Lemma 2.4.2 with $[u; v] = x_\tau$, $\xi = \gamma_\tau F(y_\tau)$, $[u'; v'] = x_{\tau+1}$, and $[s; w] = z \in X$ we get:

$$\gamma_\tau [\langle F_u(u'_\tau), u_{\tau+1} - s \rangle + \langle F_v, v_{\tau+1} - w \rangle] \le V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - V_{u_\tau}(u_{\tau+1}). \tag{2.5.6}$$

Adding (2.5.6) to (2.5.5) we obtain for every $z = [s; w] \in X$

$$\begin{aligned}
\gamma_\tau \langle F(y_\tau), y_\tau - z\rangle &= \gamma_\tau[\langle F_u(u'_\tau), u'_\tau - s\rangle + \langle F_v, v'_\tau - w\rangle] \\
&\leq V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + \gamma_\tau \langle F_u(u'_\tau) - F_u(u_\tau), u'_\tau - u_{\tau+1}\rangle - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) \\
&= V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + \delta_\tau. \qquad (2.5.7)
\end{aligned}$$

Due to the strong convexity, modulus 1, of $V_u(\cdot)$ w.r.t. $\|\cdot\|$, $V_u(u') \geq \frac{1}{2}\|u - u'\|^2$ for all $u, u'$. Therefore,

$$\begin{aligned}
\delta_\tau &\leq \gamma_\tau \|F_u(u'_\tau) - F_u(u_\tau)\|_* \|u'_\tau - u_{\tau+1}\| - \tfrac{1}{2}\|u'_\tau - u_{\tau+1}\|^2 - \tfrac{1}{2}\|u_\tau - u'_\tau\|^2 \\
&\leq \tfrac{1}{2}\left[\gamma_\tau^2 \|F_u(u'_\tau) - F_u(u_\tau)\|_*^2 - \|u_\tau - u'_\tau\|^2\right] \\
&\leq \tfrac{1}{2}\left[\gamma_\tau^2 [M + L\|u'_\tau - u_\tau\|]^2 - \|u_\tau - u'_\tau\|^2\right],
\end{aligned}$$

where the last inequality is due to (2.3.2). Note that $\gamma_\tau L < 1$ implies that

$$\gamma_\tau^2 [M + L\|u'_\tau - u_\tau\|]^2 - \|u'_\tau - u_\tau\|^2 \leq \max_r \left[\gamma_\tau^2 [M + Lr]^2 - r^2\right] = \frac{\gamma_\tau^2 M^2}{1 - \gamma_\tau^2 L^2}.$$

Let us assume that the stepsizes $\gamma_\tau > 0$ ensure that (2.5.3) holds, meaning that $\delta_\tau \leq \gamma_\tau^2 M^2$ (which, by the above analysis, is definitely the case when $0 < \gamma_\tau \leq \frac{1}{\sqrt{2}L}$; when $M = 0$, we can take also $\gamma_\tau \leq \frac{1}{L}$). When summing up inequalities (2.5.7) over $\tau = 1, 2, ..., t$ and taking into account that $V_{u_{t+1}}(s) \geq 0$, we conclude that for all $z = [s; w] \in X$,

$$\sum_{\tau=1}^t \lambda_\tau^t \langle F(y_\tau), y_\tau - z\rangle \leq \frac{V_{u_1}(s) + \sum_{\tau=1}^t \delta_\tau}{\sum_{\tau=1}^t \gamma_\tau} \leq \frac{V_{u_1}(s) + M^2 \sum_{\tau=1}^t \gamma_\tau^2}{\sum_{\tau=1}^t \gamma_\tau}, \quad \lambda_\tau^t = \gamma_\tau / \sum_{i=1}^t \gamma_i.$$

□

Invoking Propositions 2.2.4, 2.2.2, we arrive at the following

**Corollary 2.5.1.** *Under the premise of Theorem 2.5.1, for every $t = 1, 2, ...,$ setting*

$$x^t = [u^t; v^t] = \frac{1}{\sum_{\tau=1}^t \gamma_\tau} \sum_{\tau=1}^t \gamma_\tau y_\tau.$$

*we ensure that $x^t \in X$ and that*

(i) *In the case when $F$ is monotone on $X$, we have*

$$\epsilon_{\mathrm{VI}}(x^t | X, F) \leq \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1} \left[\Theta[X] + M^2 \sum_{\tau=1}^t \gamma_\tau^2\right]. \qquad (2.5.8)$$

(ii) *Let $X = X_1 \times X_2$, and let $F$ be the monotone vector field associated with the saddle point problem* (2.2.5) *with convex-concave locally Lipschitz continuous cost function $\Phi$. Then*

$$\epsilon_{\mathrm{Sad}}(x^t | X_1, X_2, \Phi) \leq \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1} \left[\Theta[X] + M^2 \sum_{\tau=1}^t \gamma_\tau^2\right]. \tag{2.5.9}$$

*In addition, assuming that problem $(P)$ in* (2.2.6) *is solvable with optimal solution $x_*^1$ and denoting by $x^{1,t}$ the projection of $x^t \in X = X_1 \times X_2$ onto $X_1$, we have*

$$\overline{\Phi}(x^{1,t}) - \mathrm{Opt}(P) \leq \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1} \left[\Theta[\{x_*^1\} \times X_2] + M^2 \sum_{\tau=1}^t \gamma_\tau^2\right]. \tag{2.5.10}$$

(iii) *Let $X = X_1 \times \cdots \times X_K$, and let $F$ be the Nash operator associated with the convex Nash equilibrium problem described in Section 2.3.2. Then*

$$\epsilon_{\mathrm{Nash}}(x^t) \leq \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1} \left[\Theta[X] + M^2 \sum_{\tau=1}^t \gamma_\tau^2\right]. \tag{2.5.11}$$

**Stepsize policy and covergence rate** Assuming $PX'$ is bounded, $\Theta[X']$ is finite. In the case when $F$ is bounded, (that is, (2.3.2) holds true with $L = 0$ and some $M = 0$), the relation (2.5.3) holds true for any stepsizes $\gamma_\tau \geq 0$. A good stepsize policy in this case is to set $\gamma_\tau \equiv \frac{\sqrt{\Theta[X]}}{M\sqrt{t}}, \tau = 1, \ldots, t$ and the associated efficiency estimate in (2.5.4) becomes

$$\mathrm{Res}(X' | \mathcal{I}_t, \lambda^t) \leq \frac{2\sqrt{\Theta[X']}M}{\sqrt{t}}. \tag{2.5.12}$$

As a result, when $U$ is bounded and $F$ is uniformly bounded, the CoMP algorithm achieves a $O(1/\sqrt{t})$ convergence rate when solving all the problems with convex structure, including the variational inequality $\mathrm{VI}(X, F)$, the saddle point problem and convex Nash equilibrium problem.

In the case when $F$ is Lipschitz continuous (that is, (2.3.2) holds true with some $L > 0$ and $M = 0$), the requirements on the stepsizes imposed in the premise of Theorem 2.5.1 reduce to $\delta_\tau \leq 0$ for all $\tau$ and are definitely satisfied with the constant stepsizes $\gamma_\tau = 1/L$. Thus, in the case under consideration we can assume w.l.o.g. that $\gamma_\tau \geq 1/L$, thus efficiency estimate in (2.5.4) becomes

$$\mathrm{Res}(X' | \mathcal{I}_t, \lambda^t) \leq \frac{\Theta[X']L}{t}.$$

and therefore (2.5.10) becomes

$$\overline{\Phi}(x^{1,t}) - \mathrm{Opt}(P) \leq \frac{\Theta[\{x_*^1\} \times X_2]L}{t}. \tag{2.5.13}$$

As a result, when $U$ is bounded and $F$ is Lipschitz continuous, the CoMP algorithm achieves a $O(1/t)$ convergence rate when solving all the aforementioned problems with convex structure.

**Composite saddle point problem.** Recall the composite saddle point problem described in Section 2.3.3,

$$\mathrm{SadVal} = \min_{u_1 \in U_1} \max_{u_2 \in U_2} \left[\phi(u_1, u_2) + \Psi_1(u_1) - \Psi_2(u_2)\right],$$

We can apply to the problem the above algorithm. Assume that we have at our disposal nonnegative constants $L_{11}, L_{22}, L_{12}$ such that

$$\begin{aligned} \|\nabla_{u_1}\phi(u_1, u_2) - \nabla_{u_1}\phi(u_1', u_2)\|_{(1,*)} &\le L_{11}\|u_1 - u_1'\|_{(1)}, \\ \|\nabla_{u_1}\phi(u_1, u_2) - \nabla_{u_1}\phi(u_1, u_2')\|_{(1,*)} &\le L_{12}\|u_2 - u_2'\|_{(2)}, \\ \|\nabla_{u_2}\phi(u_1, u_2) - \nabla_{u_2}\phi(u_1, u_2')\|_{(2,*)} &\le L_{22}\|u_2 - u_2'\|_{(2)}. \end{aligned} \tag{2.5.14}$$

For "symmetry", we also have $\|\nabla_{u_2}\phi(u_1, u_2) - \nabla_{u_2}\phi(u_1', u_2)\|_{(2,*)} \le L_{12}\|u - 1 - u_1'\|_{(1)}$. Let $\Omega_i = \max_{U_i}\omega_i(u_i) - \min_{U_i}\omega_i(u_i), i = 1, 2$ and let $\mathcal{L} = L_{11}\Omega_1 + L_{22}\Omega_2 + 2L_{12}\sqrt{\Omega_1\Omega_2}$. We can equip $U = U_1 \times U_2$ with the aggregated distance generating function

$$\omega(u = [u_1; u_2]) = \alpha_1\omega_1(u_1) + \alpha_2\omega_2(u_2),$$

where

$$\alpha_1 = \frac{L_{11}\Omega_1 + L_{12}\sqrt{\Omega_1\Omega_2}}{\mathcal{L}\Omega_1}, \alpha_2 = \frac{L_{22}\Omega_2 + L_{12}\sqrt{\Omega_1\Omega_2}}{\mathcal{L}\Omega_2}.$$

Note that $\omega(u)$ is a distance generating function on $U$ compatible with the following norm

$$\|u = [u_1; u_2]\| = \sqrt{\alpha_1\|u_1\|_{(1)}^2 + \alpha_2\|u_2\|_{(2)}^2},$$

and also in this case, $\Theta[X] \le \max_U \omega(u) - \min_U \omega(u) \le 1$.

**Corollary 2.5.2.** *Let the composite saddle point problem be as described in Section 2.3.3 with Lipschitz parameters given as above. The candidate solution $x^{t+1}$ provided by the composite Mirror Prox algorithm using the above proximal setup along with stepsize $\gamma_\tau = \frac{1}{\mathcal{L}}$, leads to the efficiency estimate*

$$\left[\phi(u_1^t, u_2^t) + \Psi_1(u_1^t) - \Psi_2(u_2^t)\right] - \mathrm{SadVal} \le \frac{\mathcal{L}}{t} = \frac{L_{11}\Omega_1 + L_{22}\Omega_2 + 2L_{12}\sqrt{\Omega_1\Omega_2}}{t}. \tag{2.5.15}$$

**Remark.** The composite Mirror Prox algorithm when applied to the composite saddle point problem, preserves the favorable $O(1/\epsilon)$ efficiency estimate of its prototype. Note that this bound is unimprovable already in the large-scale bilinear saddle point case (see [61]). It is worthwhile to mention again that the prox mapping (all the composite Mirror Prox algorithm requires to compute) reduces to easy-to-solve and decoupled auxilliary problems of form

$$\min_{u_i \in U_i, v_i \geq \Psi_i(u_i)} \left[\alpha_i \omega_i(u_i) + \beta_i v_i + \langle \xi_i, u_i \rangle\right], \; i = 1, 2,$$

which is essentially the favorable situation in lots of applications to be discussed in subsequent chapters.

### 2.5.2 Composite Mirror Prox: general averaging schemes

In fact, the composite Mirror Prox algorithm admits some freedom in building approximate solutions, freedom which can be used to improve to some extent solutions' quality. Modifications to be presented originate from [60]. We assume that we are in the situation described in Section 2.3.1, and assumptions (**A**.1) – (**A**.4) are in force. In addition, we assume that

(**A**.5): The vector field $F$ described in (**A**.3) is monotone, and the variational inequality given by $(X, F)$ has a weak solution:

$$\exists x_* = [u_*; v_*] \in X : \langle F(y), y - x_* \rangle \geq 0 \; \forall y \in X \tag{2.5.16}$$

**Lemma 2.5.1.** *In the situation from Section 2.3.1 and under assumptions (**A**.1) – (**A**.5), for any $R \geq 0$, let us set*

$$\widehat{\Theta}(R) = \max_{u, u' \in U} \left\{V_u(u') : \|u - u_1\| \leq R, \|u' - u_1\| \leq R\right\} \tag{2.5.17}$$

*(this quantity is finite since $\omega$ is continuously differentiable on $U$), and let*

$$\{x_\tau = [u_\tau; v_\tau] : \tau \leq N + 1, y_\tau : \tau \leq N\}$$

*be the trajectory of the $N$-step CoMP in Algorithm 2 with stepsizes $\gamma_\tau > 0$ which ensure* (2.5.3) *for $\tau \leq N$. Then for all $u \in U$ and $t \leq N + 1$,*

$$0 \leq V_{u_t}(u) \leq \widehat{\Theta}(\max[R_N, \|u - u_1\|]), \quad R_N := 2\left(2V_{u_1}(u_*) + M^2 \sum_{\tau=1}^{N-1} \gamma_\tau^2\right)^{1/2}, \tag{2.5.18}$$

*with $u_*$ defined in* (2.5.16).

*Proof.* All we need to verify is the second inequality in (2.5.18). To this end note that when $t = 1$, the inequality in (2.5.18) holds true by definition of $\widehat{\Theta}(\cdot)$. Now let $1 < t \leq N + 1$. Summing up the inequalities (2.5.7) over $\tau = 1, ..., t-1$, we get for every $x = [u; v] \in X$:

$$\sum_{\tau=1}^{t-1} \gamma_\tau \langle F(y_\tau), y_\tau - [u; v] \rangle \leq V_{u_1}(u) - V_{u_t}(u) + \sum_{\tau=1}^{t-1} \delta_\tau \leq V_{u_1}(u) - V_{u_t}(u) + M^2 \sum_{\tau=1}^{t-1} \gamma_\tau^2$$

(we have used (2.5.3)). When $[u; v]$ is $x_*$, the left hand side in the resulting inequality is $\geq 0$, and we arrive at

$$V_{u_t}(u_*) \leq V_{u_1}(u_*) + M^2 \sum_{\tau=1}^{t-1} \gamma_\tau^2,$$

whence

$$\frac{1}{2}\|u_t - u_*\|^2 \leq V_{u_1}(u_*) + M^2 \sum_{\tau=1}^{t-1} \gamma_\tau^2$$

whence also

$$\|u_t - u_1\|^2 \leq 2\|u_t - u_*\|^2 + 2\|u_* - u_1\|^2 \leq 4[V_{u_1}(u_*) + M^2 \sum_{\tau=1}^{t-1} \gamma_\tau^2] + 4V_{u_1}(u_*)$$

and therefore

$$\|u_t - u_1\| \leq 2\sqrt{2V_{u_1}(u_*) + M^2 \sum_{\tau=1}^{t-1} \gamma_\tau^2} = R_N, \tag{2.5.19}$$

and (2.5.18) follows. □

**Proposition 2.5.1.** *In the situation of Section 2.3.1 and under assumptions (**A**.1) – (**A**.5), let $N$ be a positive integer, and let $\mathcal{I}_N = \{y_\tau, F(y_\tau)\}_{\tau=1}^N$ be the execution protocol generated by $N$-step CoMP with stepsizes $\gamma_\tau$ ensuring* (2.5.3). *Let also $\lambda^N = \{\lambda_1, ..., \lambda_N\}$ be a collection of positive reals summing up to 1 and such that*

$$\lambda_1/\gamma_1 \leq \lambda_2/\gamma_2 \leq ... \leq \lambda_N/\gamma_N. \tag{2.5.20}$$

*Then for every $R \geq 0$, with $X_R = \{x = [u; v] \in X : \|u - u_1\| \leq R\}$ one has*

$$\mathrm{Res}(X_R|\mathcal{I}_N, \lambda^N) \leq \frac{\lambda_N}{\gamma_N}\widehat{\Theta}(\max[R_N, R]) + M^2 \sum_{\tau=1}^{N} \lambda_\tau \gamma_\tau, \tag{2.5.21}$$

*with $\widehat{\Theta}(\cdot)$ and $R_N$ defined by* (2.5.17) and (2.5.18).

*Proof.* From (2.5.7) and (2.5.3) it follows that

$$\forall(x=[u;v]\in X,\tau\le N):\lambda_\tau\langle F(y_\tau),y_\tau-x\rangle\le\frac{\lambda_\tau}{\gamma_\tau}[V_{u_\tau}(u)-V_{u_{\tau+1}}(u)]+M^2\lambda_\tau\gamma_\tau.$$

Summing up these inequalities over $\tau=1,...,N$, we get $\forall(x=[u;v]\in X)$:

$$\begin{aligned}
&\sum_{\tau=1}^N\lambda_\tau\langle F(y_\tau),y_\tau-x\rangle\\
&\le\tfrac{\lambda_1}{\gamma_1}[V_{u_1}(u)-V_{u_2}(u)]+\tfrac{\lambda_2}{\gamma_2}[V_{u_2}(u)-V_{u_3}(u)]+...+\tfrac{\lambda_N}{\gamma_N}[V_{u_N}(u)-V_{u_{N+1}}(u)]+M^2\sum_{\tau=1}^N\lambda_\tau\gamma_\tau\\
&=\tfrac{\lambda_1}{\gamma_1}V_{u_1}(u)+\left[\tfrac{\lambda_2}{\gamma_2}-\tfrac{\lambda_1}{\gamma_1}\right]V_{u_2}(u)+...+\left[\tfrac{\lambda_N}{\gamma_N}-\tfrac{\lambda_{N-1}}{\gamma_{N-1}}\right]V_{u_N}(u)-\tfrac{\lambda_N}{\gamma_N}V_{u_{N+1}}(u)+M^2\sum_{\tau=1}^N\lambda_\tau\gamma_\tau\\
&\le\tfrac{\lambda_1}{\gamma_1}\widehat{\Theta}(\max[R_N,\|u-u_1\|])+\left[\tfrac{\lambda_2}{\gamma_2}-\tfrac{\lambda_1}{\gamma_1}\right]\widehat{\Theta}(\max[R_N,\|u-u_1\|])+...\\
&\qquad\qquad+\left[\tfrac{\lambda_N}{\gamma_N}-\tfrac{\lambda_{N-1}}{\gamma_{N-1}}\right]\widehat{\Theta}(\max[R_N,\|u-u_1\|])+M^2\sum_{\tau=1}^N\lambda_\tau\gamma_\tau,\\
&=\tfrac{\lambda_N}{\gamma_N}\widehat{\Theta}(\max[R_N,\|u-u_1\|])+M^2\sum_{\tau=1}^N\lambda_\tau\gamma_\tau,
\end{aligned}$$

where the concluding inequality is due to (2.5.18), and (2.5.21) follows. □

Invoking Proposition 2.2.4 and Proposition 2.2.2, we arrive at the following modification of Corollary 2.5.1.

**Corollary 2.5.3.** *Under the premise and in the notation of Proposition 2.5.1, setting*

$$x^N=[u^N;v^N]=\sum_{\tau=1}^N\lambda_\tau y_\tau.$$

*we ensure that $x^N\in X$. Besides this,*

(i) *Let $X'$ be a closed convex subset of $X$ such that $x^N\in X'$ and the projection of $X'$ on the u-space is contained in $\|\cdot\|$-ball of radius $R$ centered at $u_1$. Then*

$$\epsilon_{\rm VI}(x^N\big|X',F)\le\frac{\lambda_N}{\gamma_N}\widehat{\Theta}(\max[R_N,R])+M^2\sum_{\tau=1}^N\lambda_\tau\gamma_\tau.\tag{2.5.22}$$

(ii) *Let $X=X_1\times X_2$ and $F$ be the monotone vector field associated with saddle point problem* (2.2.5) *with convex-concave locally Lipschitz continuous cost function $\Phi$. Let, further, $X_i'$ be closed convex subsets of $X_i$, $i=1,2$, such that $x^N\in X_1'\times X_2'$ and the projection of $X_1'\times X_2'$ onto the u-space is contained in $\|\cdot\|$-ball of radius $R$ centered at $u_1$. Then*

$$\epsilon_{\rm Sad}(x^N\big|X_1',X_2',\Phi)\le\frac{\lambda_N}{\gamma_N}\widehat{\Theta}(\max[R_N,R])+M^2\sum\nolimits_{\tau=1}^N\lambda_\tau\gamma_\tau.\tag{2.5.23}$$

**Online stepsize policy and convergence rate.** We explain below how this general averaging scheme can help to build solutions with better quality. Consider the situation when $U = PX$ is bounded and $L = 0$. Let us set $X' = X$ (or in the saddle point case, $X'_i = X_i, i = 1, 2$) and $R := \max_{u \in U} \|u - u_1\|$ and denote $\widehat{\Theta} = \max_{u,u' \in U} V_u(u')$. Note that $R$ and $\widehat{\Theta} = \widehat{\Theta}(\max[R_N, R])$ are finite. When the number of steps $N$ is not fixed in advance, it makes sense to consider varying stepsizes $\gamma_\tau$ which are not tuned to a given iteration number, e.g.,

$$\gamma_\tau = \frac{\sqrt{\widehat{\Theta}}}{M\sqrt{\tau}},\ \tau = 1, 2, \ldots, N.$$

Recall that for the usual averaging scheme, we simply adopt the weights $\lambda_\tau = \gamma_\tau / \sum_{\tau=1}^N \gamma_\tau, \tau = 1, \ldots, N$ and by Corollary 2.5.1 we obtain

$$\epsilon_{\mathrm{VI}}(x^N | X, F) \leq \frac{M\widehat{\Theta}(\ln(N) + 2)}{2(\sqrt{N+1} - 1)}. \tag{2.5.24}$$

This is derived by invoking the inequalities $\sum_{\tau=1}^N \frac{1}{\tau} \leq \ln(N)+1$ and $\sum_{\tau=1}^N \frac{1}{\sqrt{\tau}} \geq 2(\sqrt{N+1}-1)$. In contrast to constant stepsize policy, using the above varying step sizes incurs an extra log-factor in the convergence rate. However, this log factor can be avoided by using a "smarter" choice of averaging scheme. Let us consider instead the weights $\lambda_\tau = \frac{1}{N}, \tau = 1, \ldots, N$. Plugging $\gamma_\tau$ and $\lambda_\tau$ into equation (2.5.22) and using the relation $\sum_{\tau=1}^N \frac{1}{\sqrt{\tau}} \leq 2\sqrt{N} - 1$, we obtain

$$\epsilon_{\mathrm{VI}}(x^N | X, F) \leq \frac{2M\widehat{\Theta}}{\sqrt{N}}. \tag{2.5.25}$$

Essentially what it says here is that by allowing larger weights for the newer iterates, we can potentially improve the quality of the average solution; which in some sense, is also intuitively attractive.

### 2.5.3 Composite Mirror Prox: inexact prox-mappings

In this section, we extend the composite Mirror Prox algorithm to allow inexact computation of the prox-mappings. The algorithm achieves similar convergence rate as in the error-free case, provided that the errors in each iteration decrease at appropriate rates. We first introduce the notation of inexact prox-mapping with accuracy $\epsilon > 0$.

**$\epsilon$-Prox-mapping** Given $\epsilon \geq 0$ for any $\xi = [\eta; \zeta] \in E_u \times E_v$ and $x = [u; v] \in X$, let us define the subset $P_x^\epsilon(\xi)$ of $X$ as

$$P_x^\epsilon(\xi) = \{\widehat{x} = [\widehat{u}; \widehat{v}] \in X : \langle \eta + \omega'(\widehat{u}) - \omega'(u), \widehat{u} - s\rangle + \langle \zeta, \widehat{v} - w\rangle \leq \epsilon \ \forall [s; w] \in X\}.$$

When $\epsilon = 0$, this reduces to the exact prox-mapping, in the usual setting, i.e.,

$$P_x(\xi) = \operatorname*{Argmin}_{[s;w]\in X} \{\langle \eta, s\rangle + \langle \zeta, w\rangle + V_u(s)\} \ .$$

When $\epsilon > 0$, this yields our definition of an inexact prox-mapping, with inexactness parameter $\epsilon$. Note that for any $\epsilon \geq 0$, the set $P_x^\epsilon(\xi = [\eta; \gamma F_v])$ is well defined and nonempty whenever $\gamma > 0$. The Composite Mirror-Prox with inexact prox-mappings is as follows:

**Algorithm 3** Inexact CoMP Algorithm for VI$(X, F)$

**Input:** stepsizes $\gamma_\tau > 0$, inexactness $\epsilon_\tau \geq 0$, $\tau = 1, 2, \ldots$

Initialize $x_1 = [u_1; v_1] \in X$

**for** $\tau = 1, 2, \ldots, t$ **do**

$$\begin{array}{rcl} y_\tau := [u'_\tau; v'_\tau] & \in & P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau F(x_\tau)) = P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau [F_u(u_\tau); F_v]) \\ x_{\tau+1} := [u_{\tau+1}; v_{\tau+1}] & \in & P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau F(y_\tau)) = P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau [F_u(u'_\tau); F_v]) \end{array} \tag{2.5.26}$$

**end for**

**Output:** $x^t := [u^t; v^t] = \left(\sum_{\tau=1}^t \gamma_\tau\right)^{-1} \sum_{\tau=1}^t \gamma_\tau y_\tau$

We modify the analysis and establish the convergence results below. First of all, as a consequence of the $\epsilon$-optimality condition, Lemma 2.4.2 now becomes

**Lemma 2.5.2.** *For any $\epsilon \geq 0$, $x = [u; v] \in X$ and $\xi = [\eta; \zeta] \in E$, let $[u'; v'] = P_x^\epsilon(\xi)$, we have for all $[s; w] \in X$,*

$$\langle \eta, u' - s\rangle + \langle \zeta, v' - w\rangle \leq V_u(s) - V_{u'}(s) - V_u(u') + \epsilon. \tag{2.5.27}$$

**Theorem 2.5.2.** *In the setting of Section 2.3.1, assuming that (**A**.1)–(**A**.4) hold, consider the recurrence (2.5.26) with inexactness $\epsilon_\tau > 0$ and stepsizes $\gamma_\tau > 0$ , $\tau = 1, 2, ...$ satisfying the relation (2.5.3)(satisfied when $\gamma_\tau \leq (\sqrt{2}L)^{-1}$ or in the case of $M = 0$, when $\gamma_\tau \leq L^{-1}$). Then the corresponding execution protocol $\mathcal{I}_t = \{y_\tau, F(y_\tau)\}_{\tau=1}^t$ admits accuracy certificate*

$\lambda^t = \{\lambda_\tau^t = \gamma_\tau / \sum_{i=1}^t \gamma_i\}$ *such that for every* $X' \subset X$ *it holds*

$$\mathrm{Res}(X'\big|\mathcal{I}_t, \lambda^t) \leq \frac{\Theta[X'] + M^2 \sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau}{\sum_{\tau=1}^t \gamma_\tau}. \tag{2.5.28}$$

Theorem 2.5.2 generalizes the previous Theorem 2.5.1 established in Section 2.5.1 for CoMP with exact prox-mappings. When inexact prox-mappings are used, the errors due to the inexactness of the prox-mappings accumulates and is reflected in the bound (2.5.28). For completeness, we provide the proof below despite of some redundancy.

*Proof.* When applying Lemma 2.5.2 with $[u; v] = [u_\tau; v_\tau] = x_\tau$, $\xi = \gamma_\tau F(x_\tau) = [\gamma_\tau F_u(u_\tau); \gamma_\tau F_v]$, $[u'; v'] = [u'_\tau; v'_\tau] = y_\tau$, and $[s; w] = [u_{\tau+1}; v_{\tau+1}] = x_{\tau+1}$ we obtain:

$$\gamma_\tau[\langle F_u(u_\tau), u'_\tau - u_{\tau+1}\rangle + \langle F_v, v'_\tau - v_{\tau+1}\rangle] \leq V_{u_\tau}(u_{\tau+1}) - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) + \epsilon_\tau \tag{2.5.29}$$

and applying Lemma 2.5.2 with $[u; v] = x_\tau$, $\xi = \gamma_\tau F(y_\tau)$, $[u'; v'] = x_{\tau+1}$, and $[s; w] = z \in X$ we get:

$$\gamma_\tau[\langle F_u(u'_\tau), u_{\tau+1} - s\rangle + \langle F_v, v_{\tau+1} - w\rangle] \leq V_{u_\tau}(s) - V_{u_{\tau+1}}(s) - V_{u_\tau}(u_{\tau+1}) + \epsilon_\tau. \tag{2.5.30}$$

Adding (2.5.30) to (2.5.29) we obtain for every $z = [s; w] \in X$

$$\begin{aligned}
\gamma_\tau\langle F(y_\tau), y_\tau - z\rangle &= \gamma_\tau[\langle F_u(u'_\tau), u'_\tau - s\rangle + \langle F_v, v'_\tau - w\rangle] \\
&\leq V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + \gamma_\tau\langle F_u(u'_\tau) - F_u(u_\tau), u'_\tau - u_{\tau+1}\rangle - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) \\
&= V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + \delta_\tau + 2\epsilon_\tau.
\end{aligned} \tag{2.5.31}$$

Since the stepsizes $\gamma_\tau > 0$ ensure that (2.5.3) holds, meaning that $\delta_\tau \leq \gamma_\tau^2 M^2$ (which, we already know, is definitely the case when $0 < \gamma_\tau \leq \frac{1}{\sqrt{2}L}$; when $M = 0$, we can take also $\gamma_\tau \leq \frac{1}{L}$). When summing up inequalities (2.5.31) over $\tau = 1, 2, ..., t$ and taking into account that $V_{u_{t+1}}(s) \geq 0$, we conclude that for all $z = [s; w] \in X'$,

$$\sum_{\tau=1}^t \lambda_\tau^t \langle F(y_\tau), y_\tau - z\rangle \leq \frac{V_{u_1}(s) + \sum_{\tau=1}^t \delta_\tau + 2\sum_{\tau=1}^t \epsilon_\tau}{\sum_{\tau=1}^t \gamma_\tau} \leq \frac{V_{u_1}(s) + M^2 \sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau}{\sum_{\tau=1}^t \gamma_\tau}.$$

Equation (2.5.28) follows by invoking the definition of resolution. □

**Corollary 2.5.4.** *Under the premise of Theorem 2.5.1, for every* $t = 1, 2, ...,$ *setting*

$$x^t = [u^t; v^t] = \frac{1}{\sum_{\tau=1}^t \gamma_\tau} \sum_{\tau=1}^t \gamma_\tau y_\tau.$$

*we ensure that* $x^t \in X$ *and that*

(i) *In the case when* $F$ *is monotone on* $X$*, we have*

$$\epsilon_{\mathrm{VI}}(x^t|X,F) \le \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1}\left[\Theta[X] + M^2\sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau\right]. \tag{2.5.32}$$

(ii) *Let* $X = X_1 \times X_2$*, and let* $F$ *be the monotone vector field associated with the saddle point problem* (2.2.5) *with convex-concave locally Lipschitz continuous cost function* $\Phi$*. Then*

$$\epsilon_{\mathrm{Sad}}(x^t|X_1,X_2,\Phi) \le \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1}\left[\Theta[X] + M^2\sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau\right]. \tag{2.5.33}$$

*In addition, assuming that problem* $(P)$ *in* (2.2.6) *is solvable with optimal solution* $x_*^1$ *and denoting by* $x^{1,t}$ *the projection of* $x^t \in X = X_1 \times X_2$ *onto* $X_1$*, we have*

$$\overline{\Phi}(x^{1,t}) - \mathrm{Opt}(P) \le \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1}\left[\Theta[\{x_*^1\}\times X_2] + M^2\sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau\right]. \tag{2.5.34}$$

(iii) *Let* $X = X_1 \times \cdots \times X_K$*, and let* $F$ *be the Nash operator associated with the convex Nash equilibrium problem described in Section 2.3.2. Then*

$$\epsilon_{\mathrm{Nash}}(x^t) \le \left[\sum_{\tau=1}^t \gamma_\tau\right]^{-1}\left[\Theta[X] + M^2\sum_{\tau=1}^t \gamma_\tau^2 + 2\sum_{\tau=1}^t \epsilon_\tau\right]. \tag{2.5.35}$$

**Remark.** The above algorithm is a non-trivial extension of the composite Mirror Prox with *exact prox-mappings*, both from a theoretical and algorithmic point of views. Note that as long as $\{\epsilon_\tau\}$ is summable, we achieve essentially the same convergence rate as when there is no error, namely a $O(1/\sqrt{t})$ rate for bounded operators and a $O(1/t)$ rate for Lipschitz continuous operators. If $\{\epsilon_\tau\}$ decays with a rate of $O(1/\tau)$, then the overall convergence is affected by a log factor. Similar modifications as discussed in Section 2.5.2 can also be obtained when a general averaging scheme is applied.

### 2.5.4 Composite Mirror Prox: extension to stochastic setting

In this section, we further extend the previous framework to the situation where we only have access to noisy information of the operator $F$. More specifically, we assume that $F_v$ is known exactly and $u$-component of the operator $F_u(u)$ is represented by the following stochastic oracle, such that for any $u \in U$, it returns a vector $g(u,\xi)$ satisfying

(**C**.1): *Unbiasedness and bounded variance*:

$$\mathbf{E}[g(u,\xi)] = F_u(u),\ \mathbf{E}[\|g(u,\xi) - F_u(u)\|_*^2] \le \sigma^2 \tag{2.5.36}$$

where $\|\cdot\|_*$ is the dual norm same as in (**A**.3).

(**C**.2): *Light tail assumption:*

$$\mathbf{E}\left[\exp\{\|g(u,\xi) - F_u(u)\|_*^2/\sigma^2\}\right] \le \exp\{1\}. \tag{2.5.37}$$

Note that by Jensen's inequality, assumption (**C**.2) implies (**C**.1).

We assume that at $i$-th call to the oracle, the query point being $u_i$, the oracle returns $g(u_i,\xi_i)$ with i.i.d. $\xi_1,\xi_2,...$ such that (2.5.36) and (2.5.37) take place. The stochastic variant of the CoMP algorithm is as follows.

**Algorithm 4** Stochastic CoMP Algorithm for VI$(X,F)$

**Input:** stepsizes $\gamma_\tau > 0$, inexactness $\epsilon_\tau \ge 0$, $\tau = 1,2,\ldots$

Initialize $x_1 = [u_1; v_1] \in X$

**for** $\tau = 1,2,\ldots,t$ **do**

Compute $g_\tau = \frac{1}{m_\tau}\sum_{j=1}^{m_\tau} g(u_\tau,\xi_{\tau,j})$,

$$y_\tau := [u'_\tau; v'_\tau] \in P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau[g_\tau; F_v]) \tag{2.5.38}$$

Compute $\widehat{g}_\tau = \frac{1}{m_\tau}\sum_{j=m_\tau+1}^{2m_\tau} g(\widehat{u}'_\tau,\xi_{\tau,j})$ and set

$$x_{\tau+1} := [u_{\tau+1}; v_{\tau+1}] \in P_{x_\tau}^{\epsilon_\tau}(\gamma_\tau[\widehat{g}_\tau; F_v]) \tag{2.5.39}$$

**end for**

**Output:** $x^t := [u^t; v^t] = \left(\sum_{\tau=1}^t \gamma_\tau\right)^{-1}\sum_{\tau=1}^t \gamma_\tau y_\tau$

We establish below the theoretical convergence guarantee for this stochastic algorithm.

**Theorem 2.5.3.** *In the setting of Section 2.3.1, assuming that (**A**.1)–(**A**.4) hold, consider the recurrence (2.5.38) and (2.5.39) with stepsizes $\gamma_\tau > 0$ satisfying $0 < \gamma_\tau \le (\sqrt{3}L)^{-1}$ (when $M = 0$, it is enough to set $\gamma_\tau \le (\sqrt{2}L)^{-1}$). Given a sequence of inexact prox-mappings with inexactness $\epsilon_\tau \ge 0$ and batch size $m_\tau > 0$. For the corresponding execution protocol $\mathcal{I}_t = \{y_\tau, F(y_\tau)\}_{\tau=1}^t$ admits accuracy certificate $\lambda^t = \{\lambda_\tau^t = \gamma_\tau/\sum_{i=1}^t \gamma_i\}$ such that*

*for every* $X' \subset X$,

(i) *it holds under Assumption (***C***.1) that*

$$\mathbf{E}[\mathrm{Res}(X'\big|\mathcal{I}_t, \lambda^t)] \le \mathcal{M}_0(t) := \frac{2\Theta[X'] + \frac{7}{2}\sum_{\tau=1}^t \gamma_\tau^2(M^2 + \frac{2\sigma^2}{m_\tau}) + 2\sum_{\tau=1}^t \epsilon_\tau}{\sum_{\tau=1}^t \gamma_\tau}, \tag{2.5.40}$$

(ii) *it holds under Assumption (****C****.2) that for any* $\Lambda > 0$,

$$\mathrm{Prob}\left\{\mathrm{Res}(X'\big|\mathcal{I}_t, \lambda^t) \ge \mathcal{M}_0(t) + \Lambda\mathcal{M}_1(t)\right\} \le \exp\{-\Lambda^2/3\} + \exp\{-\Lambda\} \tag{2.5.41}$$

*where* $\mathcal{M}_1(t) := (\sum_{\tau=1}^t \gamma_\tau)^{-1}\left(\frac{7}{2}\sum_{\tau=1}^t \frac{\gamma_\tau^2\sigma^2}{m_\tau} + 3\Theta[X]\sqrt{\sum_{\tau=1}^t \frac{\gamma_\tau^2\sigma^2}{m_\tau}}\right)$.

**Remark.** As a corollary, we immediately have

1. when $F$ is the monotone vector field, the resulting efficiency estimates take place for the dual gap of variational inequalities;

2. when $F$ stems from a convex-concave saddle point problem, then the above efficiency estimates is inherited both by the induced primal and dual suboptimality gap.

3. when $F$ stems from Nash problem, the resulting efficiency estimates take place for Nash inaccuracy.

Let us call a random feasible solution $\bar{x}$ to the variational inequality $\mathrm{VI}(X, F)$ a stochastic $\epsilon$-solution if $\mathbf{E}[\epsilon_{\mathrm{VI}}(\bar{x}\big|X, F)] \le \epsilon$.

**Stepsize policy and convergence rate.** Assume $U = PX$ is bounded. If we set $\epsilon_\tau = \Theta[X]/t, \tau = 1, \ldots, t$, the above bound reduces to

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(x^t\big|X, F)] \le \frac{4\Theta[X] + \frac{7}{2}\sum_{\tau=1}^t \gamma_\tau^2(M^2 + \frac{2\sigma^2}{m_\tau})}{\sum_{\tau=1}^t \gamma_\tau}. \tag{2.5.42}$$

In the case when $F$ is a Lipschitz continuous monotone operator with some $L > 0$ and $M = 0$, a good choice of stepsize is $\gamma_\tau = \frac{1}{\sqrt{2}L}$. Setting $m_\tau = O(1)\gamma_\tau^2 t, \tau = 1, \ldots, t$ leads to

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(x^t\big|X, F)] \le O(1)\frac{(\Theta[X] + \sigma^2)L}{t},$$

and the the total number of stochastic oracle calls required is of order $O(t^2)$. This implies that in order to obtain an stochastic $\epsilon$-solution, the stochastic CoMP needs at most $O(\frac{1}{\epsilon^2})$ calls to the stochastic oracles.

In the case when $F$ a uniformly bounded monotone operator with some $M > 0$ and $L = 0$, a good choice of stepsize is $\gamma_\tau = \frac{\sqrt{\Theta[X]}}{(M+\sigma)\sqrt{t}}$. Setting $m_\tau = O(1), \tau = 1, \ldots, t$ leads to

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(x^t \big| X, F)] \leq O(1)\frac{\sqrt{\Theta[X]}(M+\sigma)}{\sqrt{t}},$$

and the the total number of stochastic oracle calls required is of order $O(t)$.This implies that in order to obtain an stochastic $\epsilon$-solution, the stochastic CoMP needs at most $O(\frac{1}{\epsilon^2})$ calls to the stochastic oracles. Observe that in both situations, while allowing for inexactness up to order $O(\epsilon)$ at each iteration, we achieve the same complexity bound for the stochastic oracles, which is indeed optimal (see e.g. [61]).

The proof of Theorem 2.5.3 builds upon the analysis in [45] and previous proof for Theorem 2.5.1, which we provide below for completeness.

*Proof of Theorem 2.5.3.*

**$1^0$.** First of all, by simply replacing $F_u(u_\tau)$ by $g_\tau$ and replacing $F_u(u'_\tau)$ by $\widehat{g}_\tau$, equation (2.5.7) becomes, for any $[s, w] \in X$

$$\gamma_\tau[\langle \widehat{g}_\tau, u'_\tau - s\rangle + \langle F_v, v'_\tau - w\rangle] \leq V_{u_\tau}(s) - V_{u_{\tau+1}}(s) + \sigma_\tau + 2\epsilon_\tau\ , \tag{2.5.43}$$

where $\sigma_\tau := \gamma_\tau\langle \widehat{g}_\tau - g_\tau, u'_\tau - u_{\tau+1}\rangle - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau)$. Let $\Delta_\tau = F_u(u'_\tau) - \widehat{g}_\tau$, then for any $z = [s, w] \in X$, we have

$$\sum_{\tau=1}^{t}\gamma_\tau\langle F(y_\tau), y_\tau - z\rangle \leq \Theta[X] + \sum_{\tau=1}^{t}\sigma_\tau + \sum_{\tau=1}^{t}2\epsilon_\tau + \sum_{\tau=1}^{t}\gamma_\tau\langle \Delta_\tau, u'_\tau - s\rangle \tag{2.5.44}$$

Let $e_\tau = \|g_\tau - F_u(u_\tau)\|_*$ and $\widehat{e}_\tau = \|\widehat{g}_\tau - F_u(u'_\tau)\|_* = \|\Delta_\tau\|_*$, Then we have

$$\begin{aligned}\|\widehat{g}_\tau - g_\tau\|_*^2 &= \|(\widehat{g}_\tau - F_u(u'_\tau) + (F_u(u'_\tau) - F_u(u_\tau)) + (F_u(u_\tau) - g_\tau)\|_*^2\\ &\leq (\widehat{e}_\tau + L\|u'_\tau - u_\tau\| + M + e_\tau)^2\\ &\leq 3L^2\|u'_\tau - u_\tau\|^2 + 3M^2 + 3(e_\tau + \widehat{e}_\tau)^2\end{aligned}$$

Hence,

$$\sigma_\tau \leq \frac{\gamma_\tau^2}{2}\|\widehat{g}_\tau - g_\tau\|_*^2 + \frac{1}{2}\|u'_\tau - u_{\tau+1}\|_2 - V_{u'_\tau}(u_{\tau+1}) - V_{u_\tau}(u'_\tau) \leq \frac{\gamma_\tau^2}{2}\|\widehat{g}_\tau - g_\tau\|_*^2 - \frac{1}{2}\|u'_\tau - u_\tau\|^2.$$

Since the stepsize $\gamma_\tau$ satisfy that $3\gamma_\tau^2 L \leq 1$, we further have

$$\sigma_t \leq \frac{3\gamma_\tau^2}{2}[M^2 + (e_\tau + \widehat{e}_\tau)^2]. \tag{2.5.45}$$

Define a special sequence $\widetilde{u}_\tau$ such that

$$\widetilde{u}_1 = u_1; \quad \widetilde{u}_{\tau+1} = \operatorname*{argmin}_{u \in P_u X}\{\langle \gamma_\tau \Delta_\tau, u\rangle + V_{\widetilde{u}_\tau}(u)\}, \forall \tau = 1, 2, \ldots.$$

The sequence defined above satisfies the following relation (see Corollary 2 in [45] for details): for any $z = [s, w] \in X$,

$$\sum_{\tau=1}^{t} \gamma_\tau \langle \Delta_\tau, \widetilde{u}_\tau - s\rangle \leq \Theta[X] + \sum_{\tau=1}^{t} \frac{\gamma_\tau^2}{2}\|\Delta_\tau\|_*^2 = \Theta[X] + \sum_{\tau=1}^{t} \frac{\gamma_\tau^2}{2}\widehat{e}_\tau \tag{2.5.46}$$

Combining (2.5.44), (2.5.45), (2.5.46), we end up with

$$\epsilon_{\mathrm{VI}}(x^t|X, F) \leq (\sum_{\tau=1}^{t} \gamma_\tau)^{-1} \left( 2\Theta[X] + \sum_{\tau=1}^{t} \frac{7\gamma_\tau^2}{2}[M^2 + (e_\tau^2 + \widehat{e}_\tau^2)] + \sum_{\tau=1}^{t} 2\epsilon_\tau + \sum_{\tau=1}^{t} \gamma_\tau \langle \Delta_\tau, u'_\tau - \widetilde{u}_\tau\rangle \right) \tag{2.5.47}$$

**$2^0$.** Under Assumption (**C**.1), we have

$$\mathbf{E}[\Delta_\tau|\mathcal{F}_\tau] = 0, \ \mathbf{E}[e_\tau^2|\mathcal{G}_{\tau-1}] \leq \frac{\sigma^2}{m_\tau}, \text{ and } \mathbf{E}[\widehat{e}_\tau^2|\mathcal{F}_\tau] \leq \frac{\sigma^2}{m_\tau}.$$

where $\mathcal{F}_\tau = \sigma(\xi_1^1, \ldots, \xi_{2m_\tau}^1, \ldots, \xi_1^\tau, \ldots, \xi_{m_\tau}^\tau)$ and $\mathcal{G}_\tau = \sigma(\xi_1^1, \ldots, \xi_{2m_\tau}^1, \ldots, \xi_1^\tau, \ldots, \xi_{2m_\tau}^\tau)$. One can further show that $\mathbf{E}[\langle \Delta_\tau, u'_\tau - \widetilde{u}_\tau\rangle] = 0$. It follows from (2.5.47) that

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(x^t|X, F)] \leq (\sum_{\tau=1}^{t} \gamma_\tau)^{-1} \left( 2\Theta[X] + \sum_{\tau=1}^{t} \frac{7\gamma_\tau^2}{2}[M^2 + \frac{2\sigma^2}{m_\tau}] + \sum_{\tau=1}^{t} 2\epsilon_\tau \right) \tag{2.5.48}$$

which proves the first part of the theorem.

**$3^0$.** Under Assumption (**C**.2), we have

$$\mathbf{E}[\exp\{e_\tau^2/(\sigma/\sqrt{m_\tau})^2\}] \leq \exp\{1\} \text{ and } \mathbf{E}[\exp\{\widehat{e}_\tau^2/(\sigma/\sqrt{m_\tau})^2\}] \leq \exp\{1\}.$$

Let $C_1 = \sum_{\tau=1}^{t} \frac{\gamma_\tau^2 \sigma^2}{m_\tau}$, it follows from convexity and the above equation that

$$\mathbf{E}\left[\exp\left\{\frac{1}{C_1}\sum_{\tau=1}^{t} \gamma_\tau^2(e_\tau^2 + \widehat{e}_\tau^2)\right\}\right] \leq \mathbf{E}\left[\frac{1}{C_1}\sum_{\tau=1}^{t} \frac{\gamma_\tau^2\sigma^2}{m_\tau} \exp\left\{(e_\tau^2 + \widehat{e}_\tau^2)/(\sigma/m_\tau)^2\right\}\right] \leq \exp\{2\}.$$

Applying Markov's inequality, we obtain:

$$\forall \Lambda > 0: \ \text{Prob}\left(\sum_{\tau=1}^{t} \gamma_\tau^2 (e_\tau^2 + \widehat{e}_\tau^2) \geq (2+\Lambda) C_1\right) \leq \exp\{-\Lambda\}. \tag{2.5.49}$$

Let $\zeta_\tau = \langle \Delta_\tau, u'_\tau - \widetilde{u}_\tau \rangle$. We showed earlier that $\mathbf{E}[\zeta_\tau] = 0$. since $\|u'_\tau - \widetilde{u}_\tau\| \leq 2\sqrt{2}\Theta[X]$, then we also have

$$\mathbf{E}[\exp\{\zeta_\tau^2/(2\sqrt{2}\Theta[X]\sigma/\sqrt{m_\tau})^2\}] \leq \exp\{1\}$$

Applying the relation $\exp\{x\} \leq x + \exp\{9x^2/16\}$, one has for any $s \geq 0$,

$$\mathbf{E}\left[\exp\left\{s\sum_{\tau=1}^{t}\gamma_\tau\zeta_\tau\right\}\right] \leq \mathbf{E}\left[\frac{9s^2}{16}\exp\left\{\sum_{\tau=1}^{t}\gamma_\tau^2\zeta_\tau^2\right\}\right] \leq \exp\left\{\frac{9s^2}{16}\sum_{\tau=1}^{t}\frac{8\sigma^2\Theta[X]^2\gamma_\tau^2}{m_\tau}\right\}$$

By Markov's inequality, one has

$$\forall \Lambda > 0: \ \text{Prob}\left(\sum_{\tau=1}^{t}\gamma_\tau\zeta_\tau \geq 3\Lambda\Theta[X]\sqrt{\sum_{\tau=1}^{t}\frac{\sigma^2\gamma_\tau^2}{m_\tau}}\right) \leq \exp\{-\Lambda^2/2\} \tag{2.5.50}$$

Combing equation (2.5.47), (2.5.49), and (2.5.50), we arrive at

$$\forall \Lambda > 0 \ \text{Prob}\big(\epsilon_{\text{VI}}(x^t|X,F) \geq \mathcal{M}_0(t) + \Lambda\mathcal{M}_1(t)\big) \leq \exp\{-\Lambda\} + \exp\{-\Lambda^2/2\}$$

where

$$\begin{aligned}
\mathcal{M}_0(t) &= (\textstyle\sum_{\tau=1}^t \gamma_\tau)^{-1}\left(2\Theta[X] + \sum_{\tau=1}^t \frac{7\gamma_\tau^2}{2}[M^2 + \frac{2\sigma^2}{m_\tau}] + \sum_{\tau=1}^t 2\epsilon_\tau\right), \\
\mathcal{M}_1(t) &= (\textstyle\sum_{\tau=1}^t \gamma_\tau)^{-1}\left(\frac{7}{2}\sum_{\tau=1}^t \frac{\gamma_\tau^2\sigma^2}{m_\tau} + 3\Theta[X]\sqrt{\sum_{\tau=1}^t \frac{\gamma_\tau^2\sigma^2}{m_\tau}}\right).
\end{aligned}$$

Hence, we have proved the theorem. □

## *2.6 Concluding Remarks*

In this chapter, we introduce the composite versions of Mirror Descent algorithm and Mirror Prox algorithm along with its several variants for solving convex-concave saddle point problems and monotone variational inequalities of special structures. We demonstrate that the composite Mirror Descent inherits the $O(1/\epsilon^2)$ efficiency estimate of its prototype when solving variational inequalities with bounded monotone operators. Also, the composite Mirror Prox inherits the $O(1/\epsilon)$ efficiency estimate of its prototype when when solving variational inequalities with Lipschitz continuous monotone operators. The composite Mirror Prox algorithm is extensible to situations in presence of errors, either from inexact calculation of prox mappings, or from noisy monotone operators.

# CHAPTER III

# LARGE SCALE CONVEX COMPOSITE OPTIMIZATION

## *3.1 Overview*

In this chapter, we will address the outlined four generic types of large-scale convex composite optimization problems:

(a) *Multi-Term Composite Minimization:* convex optimization problem

$$\min_{y\in Y} \sum_{k=1}^{K} [\psi_k(A_k y + b_k) + \Psi_k(A_k y + b_k)] \tag{3.1.1}$$

where $Y$ is closed convex set, for $1 \leq k \leq K$, $\psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex Lipschitz-continuous functions, and $\Psi_k(\cdot) : Y_k \to \mathbf{R}$ are proximal-friendly convex functions;

(b) *Linearly Constrained Composite Minimization:* multi-term composite minimization problems that are subject to linear equality constraints

$$\begin{array}{rl} \min\limits_{[y^1;\ldots;y^K]\in Y_1\times\cdots\times Y_K} & \sum_{k=1}^K \left[\psi_k(y^k) + \Psi_k(y^k)\right] \\ \text{s.t.} & \sum_{k=1}^K A_k y^k = b. \end{array}$$

where $Y_k$ are closed convex sets and$\psi_k$ and $\Psi_k$ are as in (a);

(c) *Norm-Regularized Nonsmooth Minimization:* composite minimization

$$\min_{y\in Y} \; f(y) + h(Ay)$$

where $f$ is a convex Lipschitz-continuous function given by saddle point representation, and $h$ is a LMO-friendly function;

(d) *Composite Maximum Likelihood Poisson Imaging:* a particular non-Lipschitz convex minimization

$$\min_{x\in\mathbf{R}^n_+} L(x) + h(x), \text{ with } L(x) = s^T x - \sum_{i=1}^{m} c_i \ln(a_i^T x)$$

where $s, c, a_i, i = 1, \ldots m$ are given nonnegative vectors and $h$ is proximal-friendly. Specific feature of Poisson Imaging is that $L(\cdot)$ in general is not even Lipschitz continuous.

Despite of their fundamental distinctions, we show that the composite Mirror Prox algorithm, when combined with saddle point representations and some other algorithmic techniques, can be applied to solve all these optimization problems with best rates of convergence, under circumstances, up to our knowledge. In the rest of this chapter, we will discuss each of these problems in details.

## *3.2 Application I: Multi-Term Composite Minimization*

### 3.2.1 Problem of Interest

What follows is inspired by the recent trend of seeking efficient ways for solving problems with hybrid regularizations or mixed penalty functions in fields such as machine learning, image restoration, signal processing and many others. We are about to present two instructive examples first (for motivations, see, e.g., [14, 4, 17]).

**Example 1. (Matrix completion)** Our first motivating example is matrix completion problem, where we want to reconstruct the original matrix $y \in \mathbf{R}^{n\times n}$, known to be both sparse and low-rank, given noisy observations of part of the entries. Specifically, our observation is $b = P_\Omega y + \xi$, where $\Omega$ is a given set of cells in an $n \times n$ matrix, $P_\Omega y$ is the restriction of $y \in \mathbf{R}^{n\times n}$ onto $\Omega$, and $\xi$ is a random noise. A natural way to recover $y$ from $b$ is to solve the optimization problem

$$\mathrm{Opt} = \min_{y\in\mathbf{R}^{n\times n}} \left\{ \frac{1}{2}\|P_\Omega y - b\|_2^2 + \lambda\|y\|_1 + \mu\|y\|_{\mathrm{nuc}} \right\} \tag{3.2.1}$$

where $\mu, \lambda > 0$ are regularization parameters. Here $\|y\|_2 = \sqrt{\mathrm{Tr}(y^T y)}$ is the Frobenius norm, $\|y\|_1 = \sum_{i,j=1}^n |y_{ij}|$ is the $\ell_1$-norm, and $\|y\|_{\mathrm{nuc}} = \sum_{i=1}^n \sigma_i(y)$ ($\sigma_i(y)$ are the singular values of $y$) is the nuclear norm of a matrix $y \in \mathbf{R}^{n\times n}$.

**Example 2. (Image recovery)** Our second motivating example is image recovery problem, where we want to recover an image $y \in \mathbf{R}^{n\times n}$ from its noisy observations $b = Ay + \xi$, where $Ay$ is a given affine mapping (e.g. the restriction operator $P_\Omega$ defined as above, or some blur operator), and $\xi$ is a random noise. Assume that the image can be decomposed as $y = y_{\mathrm{L}} + y_{\mathrm{S}} + y_{\mathrm{sm}}$ where $y_{\mathrm{L}}$ is of low rank, $y_{\mathrm{sm}}$ is the matrix of contamination by a

"smooth background signal", and $y_{\rm S}$ is a sparse matrix of "singular corruption." Under this assumption in order to recover $y$ from $b$, it is natural to solve the optimization problem

$$\mathrm{Opt} = \min_{y_{\rm L}, y_{\rm S}, y_{\rm sm} \in \mathbf{R}^{n\times n}} \{\|A(y_{\rm L} + y_{\rm S} + y_{\rm sm}) - b\|_2 + \mu_1 \|y_{\rm L}\|_{\rm nuc} + \mu_2 \|y_{\rm S}\|_1 + \mu_3 \|y_{\rm sm}\|_{\rm TV}\} \tag{3.2.2}$$

where $\mu_1, \mu_2, \mu_3 > 0$ are regularization parameters. Here $\|y\|_{\rm TV}$ is the total variation of an image $y$:

$$\begin{aligned} \|y\|_{\rm TV} &= \|\nabla_i y\|_1 + \|\nabla_j y\|_1, \\ (\nabla_i y)_{ij} &= y_{i+1,j} - y_{i,j},\ [i;j] \in \mathbf{Z}^2 :\ 1 \le i < n-1, 1 \le j < n, \\ (\nabla_j y)_{ij} &= y_{i,j+1} - y_{i,j},\ [i;j] \in \mathbf{Z}^2 :\ 1 \le i < n, 1 \le j < n-1. \end{aligned}$$

These and other examples motivate us to address the following *multi-term composite minimization problem*

$$\min_{y \in Y} \left\{ \sum_{k=1}^{K} [\psi_k(A_k y + b_k) + \Psi_k(A_k y + b_k)] \right\}. \tag{3.2.3}$$

Here for $1 \le k \le K$, $\psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex Lipschitz-continuous functions, and $\Psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex functions which are "simple and fit $Y_k$".[1] For example, to pose matrix completion problem in the form of (3.2.1), we set $K = 2$, $Y_1 = Y_2 = \mathbf{R}^{n\times n}$, $A_1$ and $A_2$ identity mapping, $b_1 = b_2 = 0$, $\psi_1(y) = \frac{1}{2}\|P_\Omega y - b\|_2^2$ , $\psi_2 = 0$, and $\Psi_1(y) = \lambda\|y\|_1$, $\Psi_2(y) = \mu\|y\|_{\rm nuc}$.

**Related work** The problem of multi-term composite minimization (3.2.3) has been considered (in a somewhat different setting) in [66] for $K = 2$. When $K = 1$, problem (3.2.3) becomes the usual composite minimization problem:

$$\min_{u \in U} \{\psi(u) + \Psi(u)\} \tag{3.2.4}$$

which is well studied in the case where $\psi(\cdot)$ is a *smooth* convex function and $\Psi(\cdot)$ is a simple non-smooth function. For instance, it was shown that the composite versions of

[1] The precise meaning of simplicity and fitting will be specified later. As of now, it suffices to give a couple of examples. When $\Psi_k$ is the $\ell_1$ norm, $Y_k$ can be the entire space, or the centered at the origin $\ell_p$-ball, $1 \le p \le 2$; when $\Psi_k$ is the nuclear norm, $Y_k$ can be the entire space, or the centered at the origin Frobenius/nuclear norm ball.

Fast Gradient Method originating in Nesterov's seminal work [63] and further developed by many authors (see, e.g., [6, 7, 22, 80, 76] and references therein), as applied to (3.2.4), work *as if* there were no nonsmooth term at all and exhibit the $O(1/t^2)$ convergence rate, which is the optimal rate attainable by first order algorithms of large-scale smooth convex optimization. Note that these algorithms cannot be directly applied to problems (3.2.3) with $K > 1$.

**Our goal and main contribution** In this section, we investigate the broad family of multi-term composite minimization problems. We consider a general situation where we do not assume the smoothness of functions $\psi_k$ in (3.2.3); instead, we assume that these functions are given by smooth saddle point representations, see below. We introduce the notion of exact penalty, which translates the original problem into an equivalent convex-concave saddle point problem. We apply to the saddle point problem our newly developed algorithmic tool, the composite Mirror Prox algorithm, which allows to achieve a $O(1/t)$ convergence rate. To our knowledge, this appears to be the best rate known, under circumstances, from the literature (and established there in essentially less general setting than the one considered below). We present promising experimental results demonstrating the potential of the approachand compare it to a number of competing methods on several interesting applications.

**Outline** The rest of this section is organized as follows. In Section 3.2.2, we elaborate the problem setting and reformulate the problem of interest as a saddle point problem with special structure, which enables us to utilize the composite Mirror Prox algorithm and provide complexity analysis of the proposed approach. In Section 3.2.4 and Section 3.2.5, we illustrate the algorithm by applying it to the aforementioned matrix completion and image decomposition problems.

### 3.2.2 Saddle Point reformulation and CoMP Algorithm

**Problem setting.** We consider the problem (3.2.3) in the situation as follows. For a nonnegative integer $K$ and $0 \leq k \leq K$ we are given

1. Euclidean spaces $E_k$ and $\overline{E}_k$ along with their nonempty closed convex subsets $Y_k$ and $Z_k$, respectively;

2. Proximal setups for $(E_k, Y_k)$ and $(\overline{E}_k, Z_k)$, that is, norms $p_k(\cdot)$ on $E_k$, norms $q_k(\cdot)$ on $\overline{E}_k$, and d.g.f.'s $\omega_k(\cdot) : Y_k \to \mathbf{R}$, $\overline{\omega}_k(\cdot) : Z_k \to \mathbf{R}$ compatible with $p_k(\cdot)$ and $q_k(\cdot)$, respectively;

3. Affine mappings $y^0 \mapsto A_k y^0 + b_k : E_0 \to E_k$, where $y^0 \mapsto A_0 y^0 + b_0$ is the identity mapping on $E_0$;

4. Lipschitz continuous convex functions $\psi_k(y^k) : Y_k \to \mathbf{R}$ along with their *saddle point representations*

$$\psi_k(y^k) = \sup_{z^k \in Z_k} [\phi_k(y^k, z^k) - \overline{\Psi}_k(z^k)], \quad 0 \le k \le K, \tag{3.2.5}$$

where $\phi_k(y^k, z^k) : Y_k \times Z_k \to \mathbf{R}$ are smooth (with Lipschitz continuous gradients) functions convex in $y^k \in Y_k$ and concave in $z^k \in Z_k$, and $\overline{\Psi}_k(z^k) : Z_k \to \mathbf{R}$ are Lipschitz continuous convex functions such that the problems of the form

$$\min_{z^k \in Z_k} \quad \left[\overline{\omega}_k(z^k) + \langle \xi^k, z^k \rangle + \alpha \overline{\Psi}_k(z^k)\right] \quad [\alpha > 0] \tag{3.2.6}$$

are easy to solve;

5. Lipschitz continuous convex functions $\Psi_k(y^k) : Y_k \to \mathbf{R}$ such that the problems of the form

$$\min_{y^k \in Y_k} \quad \left[\omega_k(y^k) + \langle \xi^k, y^k \rangle + \alpha \Psi_k(y^k)\right] \quad [\alpha > 0] \tag{3.2.7}$$

are easy to solve;

6. For $1 \le k \le K$, the norms $\pi_k^*(\cdot)$ on $E_k$ are given, with conjugate norms $\pi_k(\cdot)$, along with d.g.f.'s $\widehat{\omega}_k(\cdot) :\ W_k := \{w^k \in E_k : \pi_k(w^k) \le 1\} \to \mathbf{R}$ which are strongly convex, modulus 1, w.r.t. $\pi_k(\cdot)$ such that the problems

$$\min_{w^k \in W_k} \left[\widehat{\omega}_k(w^k) + \langle \xi^k, w^k \rangle\right] \tag{3.2.8}$$

are easy to solve.

The outlined data define the sets

$$\begin{aligned} Y_k^+ &= \{[y^k;\tau^k] :\ y^k \in Y_k, \tau^k \geq \Psi_k(y^k)\} \subset E_k^+ := E_k \times \mathbf{R},\ 0 \leq k \leq K, \\ Z_k^+ &= \{[z^k;\sigma^k] :\ z^k \in Z_k, \sigma^k \geq \overline{\Psi}_k(z^k)\} \subset \overline{E}_k^+ := \overline{E}_k \times \mathbf{R},\ 0 \leq k \leq K. \end{aligned}$$

**The problem of interest** (3.2.3) along with its saddle point reformulation in the just defined situation read

$$\mathrm{Opt} = \min_{y^0 \in Y_0} \left\{ f(y^0) := \sum_{k=0}^{K} \left[\psi_k(A_k y^0 + b_k) + \Psi_k(A_k y^0 + b_k)\right] \right\} \tag{3.2.9a}$$

$$= \min_{y^0 \in Y_0} \left\{ f(y^0) = \max_{\{z^k \in Z_k\}_{k=0}^K} \sum_{k=0}^{K} \left[\phi_k(A_k y^0 + b_k, z^k) + \Psi_k(A_k y^0 + b_k) - \overline{\Psi}_k(z^k)\right] \right\} \tag{3.2.9b}$$

which we rewrite equivalently as

$$\mathrm{Opt} = \min_{\substack{\{[y^k;\tau^k]\}_{k=0}^K \\ \in Y_0^+ \times \cdots \times Y_K^+}} \max_{\substack{\{[z^k;\sigma^k]\}_{k=0}^K \\ \in Z_0^+ \times \cdots \times Z_K^+}} \left\{ \sum_{k=0}^{K} \left[\phi_k(y^k, z^k) + \tau^k - \sigma^k\right] : y^k = A_k y^0 + b_k,\ 1 \leq k \leq K \right\}. \tag{3.2.9c}$$

From now on we make the following assumptions

(**B**.1): *We have $A_k Y_0 + b_k \subset Y_k$, $1 \leq k \leq K$;*

(**B**.2): *For $0 \leq k \leq K$, the sets $Z_k$ are bounded. Further, the functions $\Psi_k$ are below bounded on $Y_k$, and the functions $f_k = \psi_k + \Psi_k$ are coercive on $Y_k$: whenever $y_t^k \in Y_k$, $t = 1, 2, ...$, are such that $p_k(y_t^k) \to \infty$ as $t \to \infty$, we have $f_k(y_t^k) \to \infty$.*

Note that (**B**.1) and (**B**.2) imply that the saddle point problem (3.2.9c) is solvable; let $\{[y_*^k; \tau_*^k]\}_{0 \leq k \leq K}$; $\{[z_*^k; \sigma_*^k]\}_{0 \leq k \leq K}$ be the corresponding saddle point.

**Course of actions.** Given $\rho_k > 0$, $1 \le k \le K$, we approximate (3.2.9c) by the problem

$$\widehat{\mathrm{Opt}} = \min_{\substack{\{[y^k;\tau^k]\}_{k=0}^K \\ \in Y_0^+ \times \cdots \times Y_K^+}} \max_{\substack{\{[z^k;\sigma^k]\}_{k=0}^K \\ \in Z_0^+ \times \cdots \times Z_K^+}} \left\{ \sum_{k=0}^K \Big[ \phi_k(y^k, z^k) + \tau^k - \sigma^k \Big] + \sum_{k=1}^K \rho_k \pi_k^*(y^k - A_k y^0) \right\} \tag{3.2.10a}$$

$$= \min_{\substack{x^1 \in X_1 \\ := Y_0^+ \times \cdots \times Y_K^+}} \max_{\substack{x^2 \in X_2 \\ := Z_0^+ \times \cdots \times Z_K^+ \times W_1 \times \cdots W_K}} \Phi\Big( \underbrace{\{[y^k;\tau^k]\}_{k=0}^K}_{x^1}, \underbrace{\Big[\{[z^k;\sigma^k]\}_{k=0}^K; \{w^k\}_{k=1}^K\Big]}_{x^2} \Big) \tag{3.2.10b}$$

where

$$\Phi(x^1, x^2) = \sum_{k=0}^K \Big[ \phi_k(y^k, z^k) + \tau^k - \sigma^k \Big] + \sum_{k=1}^K \rho_k \langle w^k, y^k - A_k y^0 - b_k \rangle.$$

Observe that the monotone operator $F(x^1, x^2) = [F_1(x^1, x^2); F_2(x^1, x^2)]$ associated with the saddle point problem in (3.2.10b) is given by

$$\begin{array}{rcl} F_1(x^1, x^2) & = & \Big[ \nabla_{y^0} \phi_0(y^0, z^0) - \sum\limits_{k=1}^K \rho_k A_k^T w^k; 1; \ \big\{ \nabla_{y^k} \phi_k(y^k, z^k) + \rho_k w^k; 1 \big\}_{k=1}^K \Big], \\ F_2(x^1, x^2) & = & \Big[ \big\{ -\nabla_{z^k} \phi_k(y^k, z^k); 1 \big\}_{k=0}^K; \ \big\{ -\rho_k [y^k - A_k y^0 - b_k] \big\}_{k=1}^K \Big]. \end{array} \tag{3.2.11}$$

Now let us set

- $U = \left\{ \begin{array}{ll} u = [y^0; ...; y^K; z^0; ...; z^K; w^1; ...; w^K] : & y^k \in Y_k,\, z^k \in Z_k,\, 0 \le k \le K, \\ & \pi_k(w^k) \le 1,\, 1 \le k \le K \end{array} \right\}$,
- $X = \left\{ \begin{array}{ll} x = \big[u = [y^0; ...; y^K; z^1; ...; z^K; w^1; ...; w^K];\ v = [\tau^0; ...; \tau^K; \sigma^0; ...; \sigma^K]\big] : & \\ \qquad u \in U,\, \tau^k \ge \Psi_k(y^k),\, \sigma^k \ge \overline{\Psi}_k(z^k),\, 0 \le k \le K & \end{array} \right\}$,
  so that $PX \subset U$, cf. assumption (**A**.2) in Section 2.3.1.

The variational inequality associated with the saddle point problem in (3.2.10b) can be treated as the variational inequality on the domain $X$ with the monotone operator

$$F(x = [u; v]) = [F_u(u); F_v],$$

where

$$F_u(\underbrace{[y^0;...;y^K;\ z^0;...;z^K;\ w^1;...;w^K]}_{u}) = \left[\begin{array}{l} \nabla_y\phi_0(y^0,z^0) - \sum\limits_{k=1}^K \rho_k A_k^T w^k \\ \left\{\nabla_y\phi_k(y^k,z^k) + \rho_k w^k\right\}_{k=1}^K \\ \left\{-\nabla_z\phi_k(y^k,z^k\right\}_{k=0}^K \\ \left\{-\rho_k[y^k - A_k y^0 - b_k]\right\}_{k=1}^K \end{array}\right] \tag{3.2.12}$$

$$F_v(\underbrace{[\tau^0;...;\tau^K;\sigma^0;...;\sigma^K]}_{v}) = [1;...;1].$$

This operator meets the structural assumptions (**A**.3) and (**A**.4) from Section 2.3.1 ((**A**.4) is guaranteed by (**B**.2)). We can equip $U$ and its embedding space $E_u$ with the proximal setup $\|\cdot\|$, $\omega(\cdot)$ given by

$$\begin{array}{rcl} \|u\| & = & \sqrt{\sum_{k=0}^K[\alpha_k p_k^2(y^k) + \beta_k q_k^2(z^k)] + \sum_{k=1}^K \gamma_k \pi_k^2(w^k)}, \\ \omega(u) & = & \sum_{k=0}^K[\alpha_k \omega_k(y^k) + \beta_k \overline{\omega}_k(z^k)] + \sum_{k=1}^K \gamma_k \widehat{\omega}_k(w^k), \end{array} \tag{3.2.13}$$

where $\alpha_k, \beta_k$, $0 \le k \le K$, and $\gamma_k$, $1 \le k \le K$, are positive aggregation parameters[2]. Observe that carrying out a step of the CoMP algorithm presented in Section 2.5.1 requires computing $F$ at $O(1)$ points of $X$ and solving $O(1)$ auxiliary problems of the form

$$\begin{array}{l} \min\limits_{\substack{[y^0;...;y^K;z^0;...;z^K],\\ [;w^1;...;w^K;\tau^0;...;\tau^K;\sigma^0;...;\sigma^K]}} \left\{\sum\limits_{k=0}^K \left[a_k\omega_k(y^k) + \langle \xi_k, y^k\rangle + b_k\tau^k\right]\right. \\ \qquad \left. + \sum\limits_{k=0}^K \left[c_k\overline{\omega}_k(z^k) + \langle \eta_k, z^k\rangle + d_k\sigma^k\right] + \sum\limits_{k=1}^K \left[e_k\widehat{\omega}_k(w^k) + \langle \zeta_k, w^k\rangle\right]\right\} : \\ y^k \in Y_k, \tau^k \ge \Psi_k(y^k),\ z^k \in Z_k, \sigma^k \ge \overline{\Psi}_k(y^k),\ 0 \le k \le K,\ \pi_k(w^k) \le 1,\ 1 \le k \le K, \end{array}$$

with positive $a_k, ..., e_k$, and we have assumed that these problems are easy to solve.

### 3.2.3 Complexity Analysis

**Exact penalty.** Let us make one more assumption:

(**C**): *For $1 \le k \le K$,*

- *$\psi_k$ are Lipschitz continuous on $Y_k$ with constants $G_k$ w.r.t. $\pi_k^*(\cdot)$,*

[2] In principle, these parameters should be chosen to optimize the resulting efficiency estimates; this indeed is doable, provided that we have at our disposal upper bounds on the Lipschitz constants of the components of $F_u$ and that $U$ is bounded, see [56, Section 5] or [43, Section 6.3.3].

- $\Psi_k$ *are Lipschitz continuous on* $Y_k$ *with constants* $H_k$ *w.r.t.* $\pi_k^*(\cdot)$.

Given a feasible solution $\overline{x} = [\overline{x}^1; \overline{x}^2]$, $\overline{x}^1 := \{[\overline{y}^k; \overline{\tau}^k] \in Y_k^+\}_{k=0}^K$ to the saddle point problem (3.2.10b), let us set

$$\widehat{y}^0 = \overline{y}^0;\ \widehat{y}^k = A_k\overline{y}^0 + b_k,\ 1 \le k \le K;\ \widehat{\tau}^k = \Psi_k(\widehat{y}^k),\ 0 \le k \le K,$$

thus getting another feasible (by assumption (**B**.1)) solution $\widehat{x} = \left[\widehat{x}^1 = \{[\widehat{y}^k; \widehat{\tau}^k]\}_{k=0}^K; \overline{x}^2\right]$ to (3.2.10b). We call $\widehat{x}^1$ *correction* of $\overline{x}^1$. For $1 \le k \le K$ we clearly have

$$\begin{aligned}
\psi_k(\widehat{y}^k) &\le \psi_k(\overline{y}^k) + G_k\pi_k^*(\widehat{y}^k - \overline{y}^k) = \psi_k(\overline{y}^k) + G_k\pi_k^*(\overline{y}^k - A_k\overline{y}^0 - b_k),\\
\widehat{\tau}^k &= \Psi_k(\widehat{y}^k) \le \Psi_k(\overline{y}^k) + H_k\pi_k^*(\widehat{y}^k - \overline{y}^k) \le \overline{\tau}^k + H_k\pi_k^*(\overline{y}^k - A_k\overline{y}^0 - b_k),
\end{aligned}$$

and $\widehat{\tau}^0 = \Psi_0(\overline{y}^0) \le \overline{\tau}^0$. Hence for $\overline{\Phi}(x^1) = \max\limits_{x^2 \in X_2} \Phi(x^1, x^2)$ we have

$$\overline{\Phi}(\widehat{x}^1) \le \overline{\Phi}(\overline{x}^1) + \sum_{k=1}^K [H_k + G_k]\pi_k^*(\overline{y}^k - A_k\overline{y}^0 - b_k) - \sum_{k=1}^K \rho_k\pi_k^*(\overline{y}^k - A_k\overline{y}^0 - b_k).$$

We see that under the condition

$$\rho_k \ge G_k + H_k,\ 1 \le k \le K, \tag{3.2.14}$$

correction does not increase the value of the primal objective of (3.2.10b), whence the saddle point value $\widehat{\text{Opt}}$ of (3.2.10b) is $\ge$ the optimal value Opt in the problem of interest (3.2.9a). Since the opposite inequality is evident, we arrive at the following

**Proposition 3.2.1.** *In the situation of Section 3.2.1, let assumptions (***B***.1), (***B***.2), (***C***) and* (3.2.14) *hold true. Then*

*(i) the optimal value* $\widehat{\text{Opt}}$ *in* (3.2.10a) *coincides with the optimal value* Opt *in the problem of interest* (3.2.9a)*;*

*(ii) consequently, if* $\overline{x} = [\overline{x}^1; \overline{x}^2]$ *is a feasible solution of the saddle point problem in* (3.2.10b)*, then the correction* $\widehat{x}^1 = \{[\widehat{y}^k; \widehat{\tau}^k]\}_{k=0}^K$ *of* $\overline{x}^1$ *is a feasible solution to the problem of interest* (3.2.9c)*, and*

$$f(\widehat{y}^0) - \text{Opt} \le \epsilon_{\text{Sad}}(\overline{x}\big|X_1, X_2, \Phi), \tag{3.2.15}$$

*where* $\widehat{y}^0 (= y^0(\widehat{x}^1))$ *is the "*$y^0$*-component" of* $\widehat{x}^1$*;*

**Corollary 3.2.1.** *Under the premise of Proposition 3.2.1, when applying to the saddle point problem (3.2.10b) the CoMP algorithm induced by the above setup and passing "at no cost" from the approximate solutions $x^t = [x^{1,t}; x^{2,t}]$ generated by CoMP to the corrections $\widehat{x}^{1,t}$ of $x^{1,t}$'s, we get feasible solutions to the problem of interest (3.2.9a) satisfying the error bound*

$$f(y^0(\widehat{x}^{1,t})) - \mathrm{Opt} \leq \frac{\Theta[x^1_* \times X_2]L}{t},\ t = 1, 2, ... \tag{3.2.16}$$

*where $L$ is the Lipschitz constant of $F_u(\cdot)$ induced by the norm $\|\cdot\|$ given by (3.2.13), and $\Theta[\cdot]$ is induced by the d.g.f. given by the same (3.2.13) and the $u = [y^0; ...; y^K; z^0; ...; z^K; w^1; ...; w^K]$ -component of the starting point. Note that $W_k$ and $Z_k$ are compact, whence $\Theta[x^1_* \times X_2]$ is finite.*

**Remark.** In principle, we can use the result of Proposition 3.2.1 "as is", that is, to work from the very beginning with values of $\rho_k$ satisfying (3.2.14); this option is feasible, provided that we know in advance the corresponding Lipschitz constants and they are not too large (which indeed is the case in some applications). This being said, when our objective is to ensure the validity of the bound (3.2.15), selecting $\rho_k$'s according to (3.2.14) could be very conservative. From our experience, usually it is better to adjust the penalization coefficients $\rho_k$ on-line. Specifically, let $\overline{\Phi}(\overline{x}^1) = \sup_{x^2 \in X_2} \Phi(\overline{x}^1, x^2)$ (cf (2.2.6)). We always have $\widehat{\mathrm{Opt}} \leq \mathrm{Opt}$. It follows that independently of how $\rho_k$ are selected, we have

$$f(\widehat{y}^0) - \mathrm{Opt} \leq \underbrace{[f(\widehat{y}^0) - \overline{\Phi}(\overline{x}^1)]}_{\epsilon_1} + \underbrace{\left[\overline{\Phi}(\overline{x}^1) - \widehat{\mathrm{Opt}}\right]}_{\epsilon_2} \tag{3.2.17}$$

for every feasible solution $\overline{x}^1 = \{[\overline{y}^k; \overline{\tau}^k]\}_{k=0}^K$ to (3.2.10b) and the same inequality holds for its correction $\widehat{x}^1 = \{[\widehat{y}^k; \widehat{\tau}^k]\}_{k=0}^K$. When $\overline{x}^1$ is a component of a good (with small $\epsilon_{\mathrm{Sad}}$) approximate solution to the saddle point problem (3.2.10b), $\epsilon_2$ is small. If $\epsilon_1$ also is small, we are done; otherwise we can either increase in a fixed ratio the current values of all $\rho_k$, or only of those $\rho_k$ for which passing from $[\overline{y}^k; \overline{\tau}^k]$ to $[\widehat{y}^k; \widehat{\tau}^k]$ results in "significant" quantities

$$[\psi_k(\widehat{y}^k) + \widehat{\tau}^k] - [\psi_k(\overline{y}^k) + \overline{\tau}^k + \rho_k \pi^*_k(\overline{y}^k - A_k \overline{y}^0 - b_k)]$$

and solve the updated saddle point problem (3.2.10b).

### 3.2.4 Numerical Illustration I: Matrix Completion

**Matrix completion.** In the experiments to be reported, we applied the just outlined approach to the matrix completion problem, where we want to reconstruct the original matrix $y \in \mathbf{R}^{n\times n}$, known to be both sparse and low-rank, given noisy observations of part of the entries. Specifically, our observation is $b = P_\Omega y + \xi$, where $\Omega$ is a given set of cells in an $n \times n$ matrix, $P_\Omega y$ is the restriction of $y \in \mathbf{R}^{n\times n}$ onto $\Omega$, and $\xi$ is a random noise. A natural way to recover $y$ from $b$ is to solve the optimization problem

$$\mathrm{Opt} = \min_{y^0 \in \mathbf{R}^{n\times n}} \big[ \upsilon(y^0) = \underbrace{\frac{1}{2}\|P_\Omega y^0 - b\|_2^2}_{\psi_0(y^0)} + \underbrace{\lambda\|y^0\|_1}_{\Psi_0(y^0)} + \underbrace{\mu\|y^0\|_{\mathrm{nuc}}}_{\Psi_1(y^0)} \big] \tag{3.2.18}$$

where $\Omega$ is a given set of cells in an $n\times n$ matrix, and $P_\Omega y$ is the restriction of $y \in \mathbf{R}^{n\times n}$ onto $\Omega$; this restriction is treated as a vector from $\mathbf{R}^M$, $M = \mathrm{Card}(\Omega)$. $\mu, \lambda > 0$ are regularization parameters. Here $\|y\|_2 = \sqrt{\mathrm{Tr}(y^T y)}$ is the Frobenius norm, $\|y\|_1 = \sum_{i,j=1}^n |y_{ij}|$ is the $\ell_1$-norm, and $\|y\|_{\mathrm{nuc}} = \sum_{i=1}^n \sigma_i(y)$ ($\sigma_i(y)$ are the singular values of $y$) is the nuclear norm of a matrix $y \in \mathbf{R}^{n\times n}$. Note that (3.2.18) is a special case of (3.2.9b) with $K = 1$, $Y_0 = Y_1 = E_0 = E_1 = \mathbf{R}^{n\times n}$, the identity mapping $y^0 \mapsto A_1 y^0$, and $\phi_0(y^0, z^0) \equiv \psi_0(y^0)$, $\phi_1 \equiv 0$ (so that $Z_k$ can be defined as singletons, and $\overline{\Psi}_k(\cdot)$ set to 0, $k = 0, 1$).

**Implementing the CoMP algorithm.** When implementing the CoMP algorithm, we used the Frobenius norm $\|\cdot\|_F$ on $\mathbf{R}^{n\times n}$ in the role of $p_0(\cdot)$, $p_1(\cdot)$ and $\pi_1(\cdot)$, and the function $\frac{1}{2}\|\cdot\|_F^2$ in the role of d.g.f.'s $\omega_0(\cdot)$, $\omega_1(\cdot)$, $\widehat{\omega}_1(\cdot)$.

The aggregation weights in (3.2.13) were chosen as $\alpha_0 = \alpha_1 = 1/D$ and $\gamma_1 = 1$, where $D$ is a *guess* of the quantity $D_* := \|y_*^0\|_F$, where $y_*^0$ is the optimal solution (3.2.18). With $D = D_*$, our aggregation would roughly optimize the right hand side in (3.2.16), provided the starting point is the origin.

The coefficient $\rho_1$ in (3.2.10b) was adjusted dynamically as explained at the end of section 3.2.3. Specifically, we start with a small (0.001) value of $\rho_1$ and restart the solution process, increasing by factor 3 the previous value of $\rho_1$, each time when the $x^1$-component $\overline{x}$ of current approximate solution and its correction $\widehat{x}$ violate the inequality $\upsilon(y^0(\widehat{x})) \le (1+\kappa)\overline{\Phi}(\overline{x})$ for some small tolerance $\kappa$ (we used $\kappa =$ 1.e-4), cf. (3.2.17).

The stepsizes $\gamma_t$ in the CoMP algorithm were adjusted dynamically, specifically, as follows. At a step $\tau$, given a current guess $\gamma$ for the stepsize, we set $\gamma_\tau = \gamma$, perform the step and check whether $\delta_\tau \le 0$. If this is the case, we pass to step $\tau + 1$, the new guess for the stepsize being 1.2 times the old one. If $\delta_\tau$ is positive, we decrease $\gamma_\tau$ in a fixed proportion (in our implementation – by factor 0.8), repeat the step, and proceed in this fashion until the resulting value of $\delta_\tau$ becomes nonpositive. When it happens, we pass to step $\tau+1$, and use the value of $\gamma_\tau$ we have ended up with as our new guess for the stepsize.

In all our experiments, the starting point was given by the matrix $\bar{y} := P_\Omega^* b$ ("observations of entries in cells from $\Omega$ and zeros in all other cells") according to $y^0 = y^1 = \bar{y}$, $\tau^0 = \lambda\|\bar{y}\|_1$, $\tau^1 = \mu\|\bar{y}\|_{\rm nuc}$, $w^1 = 0$.

**Lower bounding the optimal value.** When running the CoMP algorithm, we at every step $t$ have at our disposal an approximate solution $y^{0,t}$ to the problem of interest (3.2.21); $y^{0,t}$ is nothing but the $y^0$-component of the approximate solution $x^t$ generated by CoMP as applied to the saddle point approximation of (3.2.21) corresponding to the current value of $\rho_1$, see (3.2.11). We have at our disposal also the value $v(y^{0,t})$ of the objective of (3.2.18) at $y^{0,t}$; this quantity is a byproduct of checking whether we should update the current value of $\rho_1$ [3]. As a result, we have at our disposal the best found so far value $v^t = \min_{1\le\tau\le t} v(y^{0,\tau})$, along with the corresponding value $y_*^{0,t}$ of $y^0$: $v(y_*^{0,t}) = v^t$. In order to understand how good is the best generated so far approximate solution $y_*^{0,t}$ to the problem of interest, we need to upper bound the quantity $v^t - \mathrm{Opt}$, or, which is the same, to lower bound Opt. This is a nontrivial task, since the domain of the problem of interest is unbounded, while the usual techniques for online bounding from below the optimal value in a convex minimization problem require the domain to be bounded. We are about to describe a technique for lower bounding Opt utilizing the structure of (3.2.18).

Let $y_*^0$ be an optimal solution to (3.2.18) (it clearly exists since $\psi_0 \ge 0$ and $\lambda, \mu > 0$).

[3] With our implementation, we run this test for both search points and approximate solutions generated by the algorithm

Assume that at a step $t$ we have at our disposal an upper bound $R = R_t$ on $\|y_*^0\|_1$, and let

$$R^+ = \max[R, \|y^{0,t}\|_1].$$

Let us look at the saddle point approximation of the problem of interest

$$\begin{gathered}\widehat{\mathrm{Opt}} = \min_{x^1=[y^0;\tau^0;y^1;\tau^1]\in\widehat{X}_1} \max_{x^2\in X_2} \left[\Phi(x^1,x^2) := \psi_0(y^0) + \tau^0 + \tau^1 + \rho_1\langle y^1 - y^0, x^2\rangle\right], \\ X_1 = \{[y^0;\tau^0;y^1;\tau^1] : \tau^0 \geq \lambda\|y^0\|_1, \tau^1 \geq \mu\|y^1\|_{\mathrm{nuc}}\},\ X_2 = \{x^2 : \|x^2\|_F \leq 1\}.\end{gathered} \tag{3.2.19}$$

associated with current value of $\rho_1$, and let

$$\bar{X}_1 = \{[y^0;\tau^0;y^1;\tau^1] \in X_1 : \tau^0 \leq \lambda R^+, \tau^1 \leq \mu R^+\}.$$

Observe that the point $x^{1,*} = [y_*^0; \lambda\|y_*^0\|_1; y_*^0; \mu\|y_*^0\|_{\mathrm{nuc}}]$ belongs to $\bar{X}_1$ (recall that $\|\cdot\|_{\mathrm{nuc}} \leq \|\cdot\|_1$) and that

$$\mathrm{Opt} = \upsilon(y_*^0) \geq \overline{\Phi}(x^{1,*}),\ \overline{\Phi}(x^1) = \max_{x^2\in X_2} \Phi(x^1,x^2).$$

It follows that

$$\widehat{\mathrm{Opt}} := \min_{x^1\in\bar{X}_1} \overline{\Phi}(x^1) \leq \mathrm{Opt}.$$

Further, by Proposition 2.2.2 as applied to $X_1' = \bar{X}_1$ and $X_2' = X_2$ we have[4]

$$\overline{\Phi}(x^{1,t}) - \widehat{\mathrm{Opt}} \leq \mathrm{Res}(\bar{X}_1 \times X_2 | \mathcal{I}_t, \lambda^t),$$

where $\mathcal{I}_t$ is the execution protocol generated by CoMP *as applied to the saddle point problem* (3.2.19) (i.e., since the last restart preceding step $t$ till this step), and $\lambda^t$ is the associated accuracy certificate. We conclude that

$$\ell_t := \overline{\Phi}(x^{1,t}) - \mathrm{Res}(\bar{X}_1 \times X_2 | \mathcal{I}_t, \lambda^t) \leq \widehat{\mathrm{Opt}} \leq \mathrm{Opt},$$

and $\ell_t$ is easy to compute (since the resolution is just the maximum of a readily given by $\mathcal{I}_t, \lambda^t$ affine function over $\bar{X}_1 \times X_2$). Setting $\upsilon_t = \max_{\tau\leq t} \ell_\tau$, we get nondecreasing with $t$ lower bounds on Opt. Note that this component of our lower bounding is independent of the particular structure of $\psi_0$.

[4] note that the latter relation implies that what was denoted by $\widetilde{\Phi}$ in Proposition 2.2.2 is nothing but $\overline{\Phi}$.

It remains to explain how to get an upper bound $R$ on $\|y_*^0\|_1$, and this is where the special structure of $\psi_0(y) = \frac{1}{2}\|P_\Omega y - b\|_2^2$ is used. Recalling that $b \in \mathbf{R}^M$, let us set

$$\vartheta(r) = \min_{v \in \mathbf{R}^M} \left\{ \frac{1}{2}\|v - b\|_2^2 : \|v\|_1 \le r \right\}, \ r \ge 0,$$

It is immediately seen that replacing the entries in $b$ by their magnitudes, $\vartheta(\cdot)$ remains intact, and that for $b \ge 0$ we have

$$\vartheta(r) = \min_{v \in \mathbf{R}^M} \left\{ \frac{1}{2}\|v - b\|_2^2 : v \ge 0, \sum_i v_i \le r \right\},$$

so that $\vartheta(\cdot)$ is an easy to compute nonnegative and nonincreasing convex function of $r \ge 0$. Now, by definition of $P_\Omega$, the function $\vartheta^+(\|y^0\|_1)$ where

$$\vartheta^+(r) = \lambda r + \vartheta(r)$$

is a lower bound on $v(y^0)$. As a result, given an upper bound $v^t$ on $\mathrm{Opt} = v(y_*)$, the easy-to-compute quantity

$$R_t := \max\{r : \vartheta^+(r) \le v^t\}$$

is an upper bound on $\|y_*^0\|_1$. Since $v^t$ is nonincreasing in $t$, $R_t$ is nonincreasing in $t$ as well.

**Generating the data.** In the experiments to be reported, the data of (3.2.18) were generated as follows. Given $n$, we build "true" $n \times n$ matrix $y_\# = \sum_{i=1}^k e_i f_i^T$, with $k = \lfloor n/4 \rfloor$ and vectors $e_i, f_i \in \mathbf{R}^n$ sampled, independently of each other, as follows: we draw a vector from the standard Gaussian distribution $\mathcal{N}(0, I_n)$, and then zero out part of the entries, with probability of replacing a particular entry with zero selected in such a way that the sparsity of $y_\#$ is about a desired level (in our experiments, we wanted $y_\#$ to have about 10% of nonzero entries). The set $\Omega$ of "observed cells" was built at random, with probability 0.25 for a particular cell to be in $\Omega$. Finally, $b$ was generated as $P_\Omega(y_\# + \sigma\xi)$, where the entries of $\xi \in \mathbf{R}^{n \times n}$ were independently of each other drawn from the standard Gaussian distribution, and

$$\sigma = 0.1 \frac{\sum_{i,j} |[y_\#]_{ij}|}{n^2}.$$

We used $\lambda = \mu = 10\sigma$.[5] Finally, our guess for the Frobenius norm of the optimal solution to (3.2.18) is defined as follows. Note that the quantity $\|b\|_2^2 - M\sigma^2$ is an estimate of $\|P_\Omega y_\#\|_2^2$. We define the estimate $D$ of $D_* := \|y_*\|_F$ "as if" the optimal solution were $y_\#$, and all entries of $y_\#$ were of the same order of magnitude

$$D = \sqrt{\frac{n^2}{M}\max[\|b\|_2^2 - M\sigma^2, 1]}, \ M = \mathrm{Card}(\Omega).$$

**Numerical results.** The results of the first series of experiments are presented in Table 1. The comments are as follows.

In the "small" experiment ($n = 128$, the largest $n$ where we were able to solve (3.2.18) in a reasonable time by `CVX` [35] using the state-of-the-art `mosek` [3] Interior-Point solver and thus knew the "exact" optimal value), CoMP exhibited fast convergence: relative accuracies 1.1e-3 and 6.2e-6 are achieved in 64 and 4096 steps (1.2 sec and 74.9 sec, respectively, as compared to 4756.7 sec taken by `CVX`).

In larger experiments ($n = 512$ and $n = 1024$, meaning design dimensions 262,144 and 1,048,576, respectively), the running times look moderate, and the convergence pattern of the CoMP still looks promising[6]. Note that our lower bounding, while somehow working, is very conservative: it overestimates the "optimality gap" $v^t - v_t$ by 2-3 orders of magnitude for moderate and large values of $t$ in the $128 \times 128$ experiment. More accurate performance evaluation would require a less conservative lower bounding of the optimal value (as of now, we are not aware of any alternative).

In the second series of experiments, the data of (3.2.18) were generated in such a way that the true optimal solution and optimal value to the problem were known from the very beginning. To this end we take as $\Omega$ the collection of all cells of an $n \times n$ matrix, which, via optimality conditions, allows to select $b$ making our "true" matrix $y_\#$ the optimal solution to (3.2.18). The results are presented in Table 2.

In the third series of experiments, we compared our algorithm with the basic version of

[5] If the goal of solving (3.2.18) were to recover $y_\#$, our $\lambda$ and $\mu$ would, perhaps, be too large. Our goal, however, was solving (3.2.18) as an "optimization beast," and we were interested in "meaningful" contribution of $\Psi_0$ and $\Psi_1$ to the objective of the problem, and thus in not too small $\lambda$ and $\mu$.

[6] Recall that we do not expect linear convergence, just $O(1/t)$ one.

ADMM as presented in [13]; this version is capable to handle straightforwardly the matrix completion with noisy observations of part of the entries[7]. The data in these experiments were generated in the same way as in the aforementioned experiments with known optimal solutions. The results are presented in Table 3. We see that ADMM is essentially faster than our algorithm, suggesting that ADMM, *when applicable in its basic form*, typically outperforms CoMP. However, this is not the case when ADMM is not directly applicable; we consider one example of the sort in the next section.

It should be mentioned that in these experiments the value of $\rho_1$ resulting in negligibly small, as compared to $\epsilon_2$, values of $\epsilon_1$ in (3.2.17) was found in the first 10-30 steps of the algorithm, with no restarts afterwards.

**Remarks.** For the sake of simplicity, so far we were considering problem (3.2.18), where minimization is carried out over $y^0$ running through the entire space $\mathbf{R}^{n\times n}$ of $n\times n$ matrices. What happens if we restrict $y^0$ to reside in a given closed convex domain $Y_0$?

It is immediately seen that the construction we have presented can be straightforwardly modified for the cases when $Y_0$ is a centered at the origin ball of the Frobenius or $\|\cdot\|_1$ norm, or the intersection of such a set with the space of symmetric $n\times n$ matrices. We could also handle the case when $Y_0$ is the centered at the origin nuclear norm ball (or intersection of this ball with the space of symmetric matrices, or with the cone of positive semidefinite symmetric matrices), but to this end one needs to "swap the penalties" – to write the representation (3.2.9c) of problem (3.2.18) as

$$\min_{\substack{\{y^k;\tau^k]\}_{k=0}^1\\ \in Y_0^+\times Y_1^+}}\left\{\Upsilon(y^0,y^1,\tau^0,\tau^1):=\underbrace{\frac{1}{2}\|P_\Omega y^0-b\|_2^2}_{\psi_0(y^0)}+\tau^0+\tau^1 : y^0=y^1\right\},$$

$$Y_0^+=\{[y^0;\tau^0]: y^0\in Y_0,\tau^0\geq\mu\|y^0\|_{\rm nuc}\},\ Y_1^+=\{[y^1;\tau^1]: y^1\in Y_1,\tau^1\geq\lambda\|y^1\|_1\},$$

where $Y_1\supset Y_0$ "fits" $\|\cdot\|_1$ (meaning that we can point out a d.g.f. $\omega_1(\cdot)$ for $Y_1$ which, taken along with $\Psi_1(y^1)=\lambda\|y^1\|_1$, results in easy-to-solve auxiliary problems (3.2.7)). We can take, e.g. $\omega_1(y^1)=\frac{1}{2}\|y^1\|_F^2$ and define $Y_1$ as the entire space, or a centered at the origin

[7]Note that in a more complicated matrix recovery problem, where noisy linear combinations of the matrix entries rather than just some of these entries are observed, applying ADMM becomes somehow problematic, while the proposed algorithm still is applicable "as is."

Table 1: Composite Mirror Prox algorithm on problem (3.2.18) with $n \times n$ matrices. $v^t$ are the best values of $v(\cdot)$, and $v_t$ are lower bounds on the optimal value found in course of $t$ steps. Platform: `MATLAB` on 3.40 GHz Intel Core i7-3770 desktop with 16 GB RAM, 64 bit Windows 7.

| $t$ | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|---|---|---|---|---|---|
| CPU, sec | 0.1 | 0.3 | 0.6 | 1.2 | 2.3 | 4.7 | 9.4 | 18.7 | 37.5 | 74.9 |
| $v^t - \text{Opt}$ | 2.0e-2 | 1.8e-2 | 1.8e-2 | 1.4e-2 | 5.3e-3 | 5.0e-3 | 1.3e-3 | 7.8e-4 | 3.2e-4 | 8.3e-5 |
| $v^t - v_t$ | 4.8e0 | 4.5e0 | 4.2e0 | 3.7e0 | 2.1e0 | 6.3e-1 | 2.1e-1 | 1.3e-1 | 6.0e-2 | 3.4e-2 |
| $\frac{v^t-\text{Opt}}{\text{Opt}}$ | 1.5e-3 | 1.3e-3 | 1.3e-3 | 1.1e-3 | 4.0e-4 | 3.7e-4 | 9.5e-5 | 5.8e-5 | 2.4e-5 | 6.2e-6 |
| $\frac{v^t-v_t}{v_{4096}}$ | 3.6e-1 | 3.4e-1 | 3.2e-1 | 2.8e-1 | 1.5e-1 | 4.7e-2 | 1.6e-2 | 9.4e-3 | 4.5e-3 | 2.6e-3 |
| $\frac{v^1-\text{Opt}}{v^t-\text{Opt}}$ | 4.8e1 | 5.4e1 | 5.4e1 | 6.7e1 | 1.8e2 | 1.9e2 | 7.5e2 | 1.2e3 | 2.9e3 | 1.1e4 |
| $\frac{v^1-v_1}{v^t-v_t}$ | 3.0e0 | 3.2e0 | 3.7e0 | 3.9e0 | 6.9e0 | 2.3e1 | 6.7e1 | 1.1e2 | 2.4e2 | 4.1e2 |

(a) $n = 128$, Opt = 13.28797 (CVX CPU 4756.7 sec)

| $t$ | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|---|---|---|
| CPU, sec | 3.7 | 7.5 | 15.0 | 29.9 | 59.8 | 119.6 | 239.2 | 478.4 | 992.0 |
| $v^t - v_t$ | 4.4e1 | 4.4e1 | 4.3e1 | 4.2e1 | 4.1e1 | 3.7e1 | 2.3e1 | 1.2e1 | 5.1e0 |
| $\frac{v^t-v_t}{v_{1024}}$ | 2.4e-1 | 2.4e-1 | 2.4e-1 | 2.4e-1 | 2.2e-1 | 2.0e-1 | 1.3-1 | 6.4e-2 | 2.8e-2 |
| $\frac{v^1-v_1}{v^t-v_t}$ | 4.4e0 | 4.4e0 | 4.5e0 | 4.6e0 | 4.8e0 | 5.5e0 | 8.5e0 | 1.7e1 | 3.8e1 |

(b) $n = 512$, $v_{2048} = 175.445 \le \text{Opt} \le v^{2048} = 180.503$ (CVX not tested)

| $t$ | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|---|---|---|
| CPU, sec | 23.5 | 46.9 | 93.8 | 187.6 | 375.3 | 750.6 | 1501.2 | 3002.3 |
| $v^t - v_t$ | 1.5e2 | 1.5e2 | 1.3e2 | 1.2e2 | 1.1e2 | 8.0e1 | 1.6e1 | 5.4e0 |
| $\frac{v^t-v_t}{v_{1024}}$ | 2.4e-1 | 2.2e-1 | 2.2e-1 | 1.9e-1 | 1.7e-01 | 1.2e-1 | 2.4e-2 | 8.1e-3 |
| $\frac{v^1-v_1}{v^t-v_t}$ | 4.6e0 | 4.8e0 | 5.3e0 | 5.7e0 | 6.3e0 | 8.9e0 | 4.5e1 | 1.3e2 |

(c) $n = 1024$, $v_{1024} = 655.422 \le \text{Opt} \le v^{1024} = 660.786$ (CVX not tested)

Table 2: Composite Mirror Prox algorithm on problem (3.2.18) with $n \times n$ matrices and known optimal value Opt. $v^t$ are the best values of $v(\cdot)$, and $v_t$ are lower bounds on the optimal value found in course of $t$ steps. Platform: `MATLAB` on 3.40 GHz Intel Core i7-3770 desktop with 16 GB RAM, 64 bit Windows 7.

| $t$ | 1 | 7 | 8 | 12 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|---|---|---|
| CPU, sec | 1.3 | 8.3 | 9.3 | 11.0 | 65.9 | 125.0 | 244.7 | 486.0 |
| $v^t - \text{Opt}$ | 92.9 | 1.58 | 0.30 | 0.110 | 0.095 | 0.076 | 0.069 | 0.069 |
| $v^t - v_t$ | 700.9 | 92.4 | 69.5 | 54.6 | 52.8 | 44.2 | 21.2 | 3.07 |
| $\frac{v^t-\text{Opt}}{\text{Opt}}$ | 0.153 | 2.6e-3 | 5.0e-4 | 1.8e-4 | 1.6e-4 | 1.3e-4 | 1.1e-4 | 1.1e-4 |
| $\frac{v^t-v_t}{\text{Opt}}$ | 1.153 | 0.152 | 0.114 | 0.090 | 0.087 | 0.073 | 0.035 | 0.005 |

(a) $n = 512$, Opt = 607.9854

| $t$ | 1 | 7 | 8 | 128 | 256 | 512 |
|---|---|---|---|---|---|---|
| CPU, sec | 8.9 | 48.1 | 51.9 | 392.7 | 752.1 | 1464.9 |
| $v^t - \text{Opt}$ | 371.4 | 3.48 | 0.21 | 0.21 | 0.19 | 0.16 |
| $v^t - v_t$ | 2772 | 241.7 | 201.2 | 147.3 | 146.5 | 122.9 |
| $\frac{v^t-\text{Opt}}{\text{Opt}}$ | 0.154 | 1.5e-3 | 9e-5 | 9e-5 | 8e-5 | 7e-5 |
| $\frac{v^t-v_t}{\text{Opt}}$ | 1.155 | 0.101 | 0.084 | 0.061 | 0.061 | 0.051 |

(b) $n = 1024$, Opt = 2401.168

Table 3: Number of steps and CPU time for Composite Mirror Prox algorithm and ADMM algorithm to achieve relative error $\epsilon = 10^{-4}$ on problem (3.2.18). Platform: `MATLAB` on Intel i5-2400S @2.5GHz CPU with 4GB RAM, 64-bit Windows 7.

| $n \times n$ | Composite Mirror Prox | | ADMM | |
|---|---|---|---|---|
| | step | CPU,sec | step | CPU,sec |
| $128 \times 128$ | 34 | 0.77 | 11 | 0.13 |
| $256 \times 256$ | 94 | 8.02 | 9 | 0.37 |
| $512 \times 512$ | 38 | 15.06 | 9 | 1.42 |
| $1024 \times 1024$ | 34 | 81.76 | 8 | 8.74 |

Frobenius/$\|\cdot\|_1$ norm ball large enough to contain $Y_0$.

### 3.2.5 Numerical Illustration II: Image Decomposition

**Image decomposition.** Consider image recovery problem, where we want to recover an image $y \in \mathbf{R}^{n\times n}$ from its noisy observations $b = Ay + \xi$, where $Ay$ is a given affine mapping (e.g. the restriction operator $P_\Omega$ defined as above, or some blur operator), and $\xi$ is a random noise. Assume that the image can be decomposed as $y = y_{\rm L} + y_{\rm S} + y_{\rm sm}$ where $y_{\rm L}$ is of low rank, $y_{\rm sm}$ is the matrix of contamination by a "smooth background signal", and $y_{\rm S}$ is a sparse matrix of "singular corruption." Under this assumption in order to recover $y$ from $b$, it is natural to solve the optimization problem

$$\mathrm{Opt} = \min_{y_{\rm L},y_{\rm S},y_{\rm sm}\in\mathbf{R}^{n\times n}} \{\|A(y_{\rm L}+y_{\rm S}+y_{\rm sm})-b\|_2 + \mu_1\|y_{\rm L}\|_{\rm nuc} + \mu_2\|y_{\rm S}\|_1 + \mu_3\|y_{\rm sm}\|_{\rm TV}\} \tag{3.2.20}$$

where $\mu_1, \mu_2, \mu_3 > 0$ are regularization parameters. Here $\|y\|_{\rm TV}$ is the total variation of an image $y$:

$$\begin{aligned} \|y\|_{\rm TV} &= \|\nabla_i y\|_1 + \|\nabla_j y\|_1, \\ (\nabla_i y)_{ij} &= y_{i+1,j} - y_{i,j},\ [i;j] \in \mathbf{Z}^2:\ 1 \le i < n-1, 1 \le j < n, \\ (\nabla_j y)_{ij} &= y_{i,j+1} - y_{i,j},\ [i;j] \in \mathbf{Z}^2:\ 1 \le i < n, 1 \le j < n-1. \end{aligned}$$

**Problem reformulation.** We first rewrite (3.2.20) as a saddle point optimization problem

$$\begin{aligned} \mathrm{Opt} &= \min_{y^1,y^2,y^3\in\mathbf{R}^{n\times n}} \left\{\|A(y^1+y^2+y^3)-b\|_2 + \mu_1\|y^1\|_{\rm nuc} + \mu_2\|y^2\|_1 + \mu_3\|Ty^3\|_1\right\} \\ &= \min_{y^1,y^2,y^3} \max_{\|z\|_2\le 1} \left\{\langle z, A(y^1+y^2+y^3)-b\rangle + \mu_1\|y^1\|_{\rm nuc} + \mu_2\|y^2\|_1 + \mu_3\|Ty^3\|_1\right\}, \end{aligned} \tag{3.2.21}$$

where $T:\ \mathbf{R}^{n\times n}\to\mathbf{R}^{2n(n-1)}$ is the mapping $y\mapsto Ty=\left[\begin{array}{c}\{(\nabla_i y)_{n(j-1)+i}\}_{i=1,\ldots,n-1,\,j=1,\ldots,n}\\ \{(\nabla_j y)_{n(i-1)+j})\}_{i=1,\ldots,n,\,j=1,\ldots,n-1}\end{array}\right]$.

Next we rewrite (3.2.21) as a linearly constrained saddle-point problem with "simple" penalties:

$$\mathrm{Opt} \;=\; \min_{\substack{y^3\in Y_3\\ [y^k;\tau_k]\in Y_k^+,\,0\le k\le 2}}\ \max_{z\in Z}\left\{\langle z, A(y^1+y^2+y^3)-b\rangle+\tau_1+\tau_2+\tau_0,\ y^0=Ty^3\right\},$$

where

$$\begin{aligned}
Y_0^+ &= \{[y^0;\tau_0]:\ y^0\in Y_0=\mathbf{R}^{2n(n-1)}:\ \|y^0\|_1\le\tau_0/\mu_3\},\\
Y_1^+ &= \{[y^1;\tau_1]:\ y^1\in Y_1=\mathbf{R}^{n\times n}:\ \|y^1\|_{\mathrm{nuc}}\le\tau_1/\mu_1\},\\
Y_2^+ &= \{[y^2;\tau_2]:\ y^2\in Y_2=\mathbf{R}^{n\times n}:\ \|y^2\|_1\le\tau_2/\mu_2\}\\
Y_3 &= \mathbf{R}^{n\times n},\ \ Z=\{z\in\mathbf{R}^M:\ \|z\|_2\le 1\},
\end{aligned}$$

and further approximate the resulting problem with its penalized version:

$$\widehat{\mathrm{Opt}} \;=\; \min_{\substack{y^3\in Y_3\\ [y^k;\tau_k]\in Y_k^+,\,0\le k\le 2}}\ \max_{\substack{z\in Z\\ w\in W}}\left\{\begin{array}{l}\langle z, A(y^1+y^2+y^3)-b\rangle\\ +\tau_1+\tau_2+\tau_0+\rho\langle w, y^0-Ty^3\rangle\end{array}\right\},\qquad(3.2.22)$$

with

$$W=\{w\in\mathbf{R}^{2n(n-1)},\ \|w\|_2\le 1\}.$$

Note that the function $\psi(y^1,y^2,y^3):=\|A(y^1+y^2+y^3)-b\|_2=\max_{\|z\|_2\le1}\langle z,\ A(y^1+y^2+y^3)-b\rangle$ is Lipschitz continuous in $y^3$ with respect to the Euclidean norm on $\mathbf{R}^{n\times n}$ with corresponding Lipschitz constant $G=\|A\|_{2,2}$, which is the spectral norm (the principal singular value) of $A$. Further, $\Psi(y^0)=\mu_3\|y^0\|_1$ is Lipschitz-continuous in $y^0$ with respect to the Euclidean norm on $\mathbf{R}^{2n(n-1)}$ with the Lipschitz constant $H\le\mu_3\sqrt{2n(n-1)}$. With the help of the result of Proposition 3.2.1 we conclude that to ensure the "exact penalty" property it suffices to choose $\rho\ge\|A\|_{2,2}+\mu_3\sqrt{2n(n-1)}$. Let us denote

$$U=\left\{\begin{array}{c}u=[y^0;\ldots;y^3;z;w]:\ y^k\in Y^k,\,0\le k\le 3,\\ z\in\mathbf{R}^M,\ \|z\|_2\le 1,\ w\in\mathbf{R}^{2n(n-1)},\ \|w\|_2\le 1\end{array}\right\}.$$

We equip the embedding space $E_u$ of $U$ with the norm

$$\|u\| = \left( \alpha_0 \|y^0\|_2^2 + \sum_{k=1}^{3} \alpha_k \|y^k\|_2^2 + \beta \|z\|_2^2 + \gamma \|w\|_2^2 \right)^{1/2},$$

and $U$ with the proximal setup $(\|\cdot\|, \omega(\cdot))$ with

$$\omega(u) = \frac{\alpha_0}{2} \|y^0\|_2^2 + \sum_{k=1}^{3} \frac{\alpha_k}{2} \|y^k\|_2^2 + \frac{\beta}{2} \|z\|_2^2 + \frac{\gamma}{2} \|w\|_2^2.$$

**Implementing the CoMP algorithm.** When implementing the CoMP algorithm, we use the above proximal setup with adaptive aggregation parameters $\alpha_0 = \cdots = \alpha_4 = 1/D^2$ where $D$ is our guess for the upper bound of $||y_*||_2$, that is, whenever the norm of the current solution exceeds 20% of the guess value, we increase $D$ by factor 2 and update the scales accordingly. The penalty $\rho$ and stepsizes $\gamma_t$ are adjusted dynamically in the same way as explained in the Matrix Completion experiment.

**Numerical results.** In the first series of experiments, we build the $n \times n$ observation matrix $b$ by first generating a random matrix with rank $r = \lfloor \sqrt{n} \rfloor$ and another random matrix with sparsity $p = 0.01$, so that the observation matrix is a sum of these two matrices and of random noise of level $\sigma = 0.01$; we take $y \mapsto Ay$ as the identity mapping. We use $\mu_1 = 10\sigma, \mu_2 = \sigma, \mu_3 = \sigma$. The results of this series of experiments are presented in Table 4. Note that unlike the matrix completion problem, discussed in Section 3.2.4, here we are not able to generate the problem with known optimal solutions. Better performance evaluation would require good lower bounding of the true optimal value, which is however problematic due to unbounded problem domain.

In the second series of experiments, we implement the CoMP algorithm to decompose real images and extract the underlying low rank/sparse singular distortion/smooth background components. The purpose of these experiments is to illustrate how the algorithm performs with the choice of small regularization parameters which is meaningful from the point of view of applications to image recovery. Image decomposition results for two images are provided on Figures 1 and 2. In Figure 1, we present the decomposition of the observed image of size $256 \times 256$. We apply the model (3.2.21) with regularization parameters $\mu_1 = 0.03, \mu_2 = 0.001, \mu_3 = 0.005$. We run 2 000 iterations of CoMP (total

Table 4: Composite Mirror Prox algorithm on problem (3.2.20) with $n \times n$ matrices. $v^t$ are the best values of $v(\cdot)$ in course of $t$ steps. Platform: `MATLAB` on Intel i5-2400S @2.5GHz CPU with 4GB RAM, 64-bit Windows 7.

| $t$ | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|---|---|---|
| CPU, sec | 0.1 | 0.2 | 0.4 | 0.8 | 1.6 | 3.1 | 6.3 | 12.6 | 25.2 |
| $v_t - v_{2048}$ | 1.5e1 | 2.8e0 | 6.2e-1 | 2.3e-1 | 1.1e-1 | 4.2e-2 | 1.5e-2 | 4.4e-3 | 0.0e0 |
| $\frac{v_t - v_{2048}}{v_{2048}}$ | 9.5e-1 | 1.8e-1 | 4.0e-2 | 1.5e-2 | 7.0e-3 | 2.7e-3 | 9.9e-4 | 2.8e-4 | 0.0e0 |
| $v_t - \mathrm{Opt}$ | 1.5e1 | 2.8e0 | 6.2e-1 | 2.3e-1 | 1.1e-1 | 4.5e-2 | 1.8e-2 | 6.6e-3 | 2.2e-3 |
| $\frac{v_t - \mathrm{Opt}}{\mathrm{Opt}}$ | 9.5e-1 | 1.8e-1 | 4.0e-2 | 1.5e-2 | 7.1e-3 | 2.9e-3 | 1.1e-3 | 4.2e-4 | 1.4e-4 |

(a) $n = 64$, Opt $= 15.543$ (CVX CPU 4525.5 sec)

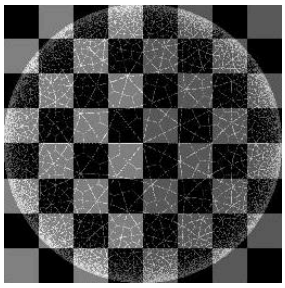
| $t$ | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|---|---|---|
| CPU, sec | 6.2 | 12.3 | 24.7 | 49.3 | 98.6 | 197.2 | 394.4 | 788.9 | 1577.8 |
| $v_t - v_{2048}$ | 1.1e2 | 5.8e1 | 2.7e1 | 1.3e1 | 6.2e0 | 2.9e0 | 1.2e0 | 3.9e-1 | 0.0e0 |
| $\frac{v_t - v_{2048}}{v_{2048}}$ | 9.0e-1 | 4.9e-1 | 2.3e-1 | 1.1e-1 | 5.2e-2 | 2.5e-2 | 1.0e-2 | 3.3e-3 | 0.0e0 |

(b) $n = 512$ (CVX not tested)

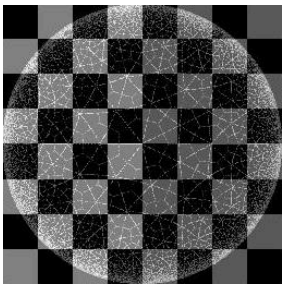
of 393.5 sec `MATLAB`, Intel i5-2400S@2.5GHz CPU). The first component $y_1$ has approximate rank $\approx 1$; the relative reconstruction error is $\|y_1 + y_2 + y_3 - b\|_2/\|b\|_2 \approx 2.8 \times 10^{-4}$. Figure 2 shows the decomposition of the observed image of size $480 \times 640$ after 1 000 iterations of CoMP (total of 873.6 sec). The regularization parameters of the problem (3.2.20) were set to $\mu_1 = 0.06, \mu_2 = 0.002, \mu_3 = 0.005$. The relative reconstruction error is $\|y_1 + y_2 + y_3 - b\|_2/\|b\|_2 \approx 8.4 \times 10^{-3}$.

In the third series of experiments, we compare the CoMP algorithm with some other first-order methods. To the best of our knowledge, a quite limited set of known methods are readily applicable to problems of the form (3.2.20), where the "observation-fitting" component in the objective is nonsmooth and the penalty terms involve different components of the observed image. As a result, we compared CoMP to just two alternatives. The first, below referred to as smoothing-APG, applies Nesterov's smoothing techniques to both the first $\|\cdot\|_2$ term and the total variation term in the objective of (3.2.20) and then uses the Accelerated Proximal Gradient method (see [62, 63] for details) to solve the resulting problem which takes the form

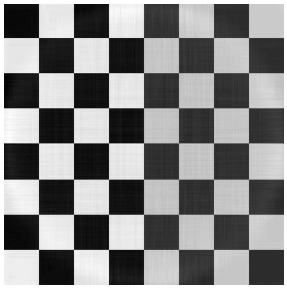
$$\min_{y^1, y^2, y^3 \in \mathbf{R}^{m \times n}} \left\{ f_{\rho_1}(y^1, y^2, y^3) + \mu_1 \|y^1\|_{\mathrm{nuc}} + \mu_2 \|y^2\|_1 + f_{\rho_2}(y^3) \right\} \tag{3.2.23}$$



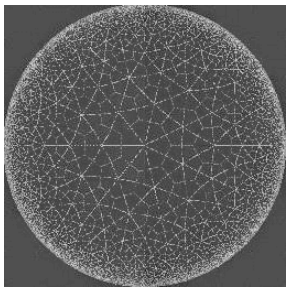
(a) observation $b$



(b) recovery $y_1 + y_2 + y_3$



(c) low-rank component



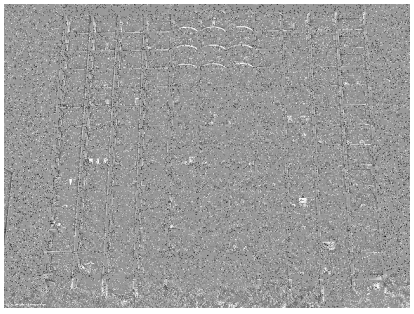
(d) sparse component

(e) smooth component

Figure 1: Observed and reconstructed images (size $256 \times 256$).

(a) observation $b$

(b) low-rank component



(c) sparse component

(d) smooth component

Figure 2: Observed and decomposed images (size $480 \times 640$)

with

$$\begin{aligned} f_{\rho_1}(y^1,y^2,y^3) &= \max_{z:\|z\|_2\le 1}\left\{\langle P_\Omega(y^1+y^2+y^3)-b,z\rangle - \tfrac{\rho_1}{2}\|z\|_2^2\right\} \\ f_{\rho_2}(y^3) &= \max_{w:\|w\|_\infty\le 1}\left\{\mu_3\langle Ty^3,w\rangle - \tfrac{\rho_2}{2}\|w\|_2^2\right\} \end{aligned}$$

where $\rho_1 > 0, \rho_2 > 0$. In the experiment, we specified the smoothing parameters as $\rho_1 = \epsilon, \rho_2 = \frac{\epsilon}{2(n-1)n}, \epsilon = 10^{-3}$.

The second alternative, referred to as smoothing-ADMM, applies smoothing technique to the first term in the objective of (3.2.20) and uses the ADMM algorithm to solve the resulting problem

$$\begin{aligned} \min_{y^1,y^2,y^3\in\mathbf{R}^{m\times n}} \quad & \left\{f_{\rho_1}(y^1,y^2,y^3)+\mu_1\|y^1\|_{\text{nuc}}+\mu_2\|y^2\|_1+\mu_3\|z\|_1\right\} \\ \text{s.t.} \quad & Ty^3 - z = 0 \end{aligned} \tag{3.2.24}$$

the associated augmented Lagrangian being

$$L_\nu(x,z;w) = \quad f_{\rho_1}(y^1,y^2,y^3)+\mu_1\|y^1\|_{\text{nuc}}+\mu_2\|y^2\|_1+\mu_3\|z\|_1+\langle w, Ty^3-z\rangle+\tfrac{\nu}{2}\|Ty^3-z\|_2^2$$

where $x = [y^1, y^2, y^3]$, $\nu > 0$ is a parameter. The basic version of ADMM would require performing alternatively $x = (y^1, y^2, y^3)$-updates and $z$-updates. Since minimizing $L_\nu$ in $x$ in a closed analytic form is impossible, we are enforced to perform $x$-update iteratively and hence inexactly. In our experiment, we used for this purpose the Accelerated Proximal Gradient method, with three implementations differing by the allowed number of inner iterations (5, 20, 50, respectively).

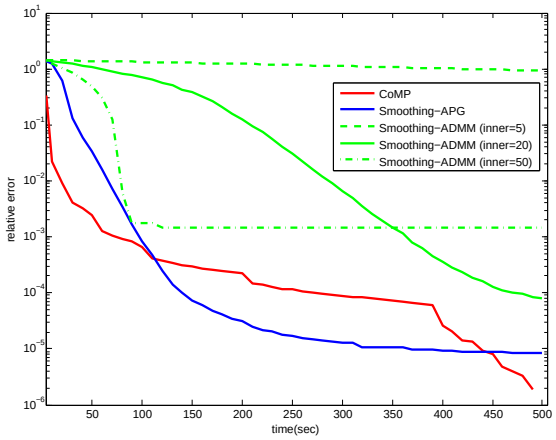



Figure 3: Comparing CoMP, smoothing-APG, and smoothing-ADMM on problem (3.2.20) with $128 \times 128$ matrix. $x$-axis: CPU time; $y$-axis: relative inaccuracy. Platform: `MATLAB` on Intel i5-2400S @2.5GHz CPU with 4GB RAM, 64-bit Windows 7.

In the experiment, we generated synthetic data in the same fashion as in the first series of experiments and compared the performances of the three algorithms (CoMP and two just described alternatives) by computing accuracies in terms of the objective achieved within a prescribed time budget. The results are presented in Figure 3. One can see that the performance of ADMM heavily depends on the allowed number of inner iterations and is not better than the performance of the Accelerated Proximal Gradient algorithm as applied to smooth approximation of the problem of interest. Our algorithm, although not consistently outperforming the Smoothing-APG approach, could still be very competitive, especially when only low accuracy is required.

#### 3.2.6 Concluding Remarks

In this section, we have investigated a particular family of problems, multi-term composite minimization, which has broad applications in many fields. We develop saddle point reformulation based on exact penalty that takes advantages of the specific problem's structure and allows us to directly apply the composite Mirror Prox algorithm. The resulting algorithm achieves the optimal $O(1/t)$ rate of convergence, which appears to be the best rate known, under circumstances, from the literature (and established there in essentially less general setting than ours). We also present, highly encouraging in our opinion, results of numerical experiments in two important applications – low-rank matrix completion and image decomposition.

### *3.3 Application II: Linearly Constrained Composite Minimization*

#### 3.3.1 Problem of Interest

Now we consider a more general (than in Section 3.2) class of convex composite minimization problems that are subject to linear equality constraints:

$$\begin{array}{rl} \min\limits_{[y^1;...;y^K]\in Y_1\times\cdots\times Y_K} & \sum_{k=1}^K \left[\psi_k(y^k)+\Psi_k(y^k)\right] \\ \text{s.t.} & \sum_{k=1}^K A_k y^k = b. \end{array} \tag{3.3.1}$$

Here for $1 \le k \le K$, $\psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex Lipschitz-continuous functions, and $\Psi_k(\cdot) : Y_k \to \mathbf{R}$ are convex functions which are simple and fit $Y_k$. We call this type

of problem, the *semi-separable problem*. One can immediately see that the above type of problems is a generalization of the multi-term composite minimization with linearly coupling constraints; now we allow for general-type linear constraints linking $y^1, ..., y^K$, while in Multi-Term Composite Minimization all $y^k$ are affinely parameterized by one of these $y^k$'s, see (3.2.9a).

A typical example that falls into this category is the basis pursuit problem, which is the following nonsmooth problem

$$\min_{x \in X} \{\|x\|_1 : \ Ax = b\} \tag{3.3.2}$$

Note that this problem can be written in the semi-separable form

$$\min_{x \in X} \left\{ \sum_{k=1}^{K} \|x_k\|_1 : \ \sum_{k=1}^{K} A_k x_k = b \right\}$$

if the data is partitioned into $K$ blocks: $x = [x_1; x_2; \ldots; x_K]$ and $A = [A_1, A_2, \ldots, A_K]$. There are also many other problems arising in signal processing, machine learning and image processing which can be naturally posed in the form of (3.3.1).

**Related work.** Problems with semi-separable structure (3.3.1) for $K = 2$, have been extensively studied using the augmented Lagrangian approach (see, e.g., [79, 13, 73, 83, 33, 34, 54, 68] and references therein). In particular, much work was carried out on the alternating directions method of multipliers (ADMM, see [13] for an overview), which optimizes the augmented Lagrangian in an alternating fashion and exhibits an overall $O(1/t)$ convergence rate. Note that the available accuracy bounds for those algorithms involve optimal values of Lagrange multipliers of the equality constraints (cf. [68]). Several variants of this method have been developed recently to adjust to the case of $K > 2$ (see, e.g.[30, 40]), however, most of these algorithms require to solve iteratively time consuming composite minimization subproblems especially when non-smooth terms in the objective are present.

A straightforward approach to solve (3.3.1) would be to rewrite it as a saddle point problem

$$\min_{[y^1; \ldots; y^K] \in Y_1 \times \cdots \times Y_K} \max_{w} \left\{ \sum_{k=1}^{K} [\psi_k(y^k) + \Psi_k(y^k)] + \langle \sum_{k=1}^{K} A_k z^k - b, w \rangle \right\} \tag{3.3.3}$$

and solve by the composite Mirror Prox algorithm from Section 2.5.1 adjusted to work with an unbounded domain $U$, or, alternatively, we could replace $\max_w$ with $\max_{w:\,\pi(w)\le R}$ with "large enough" $R$ and use the above algorithm "as is", where $\pi(\cdot)$ is some norm. The potential problem with this approach is that if the $w$-component $w^*$ of the saddle point of (3.3.3) is of large $\pi$-norm (or "large enough" $R$ is indeed large), the (theoretical) efficiency estimate would be bad since it is proportional to the magnitude of $w^*$ (resp., to $R$).

**Our goal and main contribution.** In this section, we would like to circumvent the above difficulty of unfavorable dual domains by applying to (3.3.11) a more sophisticated policy originating from [50]. We propose a sequential composite Mirror Prox algorithm, which achieves an overall $O(1/\epsilon)$ complexity bound up to log factors. We present promising experimental results showing the potential of our algorithm as compared to the simple approach described above for the basis pursuit application.

**Outline** The rest of this section is organized as follows. In Section 3.2.2, we elaborate the problem setting and reformulate the problem of interest as a saddle point problem with special structures, which enables us to utilize the composite Mirror Prox algorithm. We provide also the corresponding complexity analysis. In Section 3.2.4 and Section 3.2.5, we illustrate the algorithm when applied to the aforementioned matrix completion problem and image decomposition problem, respectively.

### 3.3.2 A Generic Algorithm for Convex Constrained Problems

Note that our problem of interest is of the generic form

$$\mathrm{Opt} = \min_{y\in Y} \{f(y) :\ g(y) \le 0\} \tag{3.3.4}$$

where $Y$ is a convex compact set in a Euclidean space $E$, $f$ and $g :\ Y \to \mathbf{R}$ are convex and Lipschitz continuous functions. For the time being, we focus on (3.3.4) and assume that the problem is feasible and thus solvable.

We intend to solve (3.3.4) by the generic algorithm presented in [50]; for our now purposes, the following description of the algorithm will do:

1. The algorithm works in *stages*. Stage $s = 1, 2, ...$ is associated with *working parameter* $\alpha_s \in (0, 1)$. We set $\alpha_1 = \frac{1}{2}$.

2. At stage $s$, we apply a first order method $\mathcal{B}$ to the problem

$$(P_s) \qquad \mathrm{Opt}_s = \min_{y \in Y} \{f_s(y) = \alpha_s f(y) + (1 - \alpha_s) g(y)\} \tag{3.3.5}$$

The only property of the algorithm $\mathcal{B}$ which matters here is its ability, when run on $(P_s)$, to produce in course of $t = 1, 2, ...$ steps iterates $y_{s,t}$, upper bounds $\overline{f}_s^t$ on $\mathrm{Opt}_s$ and lower bounds $\underline{f}_{s,t}$ on $\mathrm{Opt}_s$ in such a way that

(a) for every $t = 1, 2, ...$, the $t$-th iterate $y_{s,t}$ of $\mathcal{B}$ as applied to $(P_s)$ belongs to $Y$;

(b) the upper bounds $\overline{f}_s^t$ are nonincreasing in $t$ (this is "for free") and "are achievable," that is, they are of the form

$$\overline{f}_s^t = f_s(y^{s,t}),$$

where $y^{s,t} \in Y$ is a vector which we have at our disposal at step $t$ of stage $s$;

(c) the lower bounds $\underline{f}_{s,t}$ should be nondecreasing in t (this again is "for free");

(d) for some nonincreasing sequence $\epsilon_t \to +0$, $t \to \infty$, we should have

$$\overline{f}_s^t - \underline{f}_{s,t} \leq \epsilon_t$$

for all $t$ and $s$.

Note that since (3.3.4) is solvable, we clearly have $\mathrm{Opt}_s \leq \alpha_s \mathrm{Opt}$, implying that the quantity $\underline{f}_{s,t}/\alpha_s$ is a lower bound on Opt. Thus, at step $t$ of stage $s$ we have at our disposal a number of valid lower bounds on Opt; we denote the best (the largest) of these bounds $\underline{\mathrm{Opt}}_{s,t}$, so that

$$\mathrm{Opt} \geq \underline{\mathrm{Opt}}_{s,t} \geq \underline{f}_{s,t}/\alpha_s \tag{3.3.6}$$

for all $s, t$, and $\underline{\mathrm{Opt}}_{s,t}$ is nondecreasing in time[8].

[8] in what follows, we call a collection $a_{s,t}$ of reals nonincreasing in time, if $a_{s',t'} \leq a_{s,t}$ whenever $s' \geq s$, same as whenever $s = s'$ and $t' \geq t$. "Nondecreasing in time" is defined similarly.

3. When the First Order oracle is invoked at step $t$ of stage $s$, we get at our disposal a triple $(y_{s,t} \in Y, f(y_{s,t}), g(y_{s,t}))$. We assume that all these triples are somehow memorized. Thus, after calling First Order oracle at step $t$ of stage $s$, we have at our disposal a finite set $Q_{s,t}$ on the 2D plane such that for every point $(p,q) \in Q_{s,t}$ we have at our disposal a vector $y_{pq} \in Y$ such that $f(y_{pq}) \le p$ and $g(y_{pq}) \le q$; the set $Q_{s,t}$ (in today terminology, a *filter*) is comprised of all pairs $(f(y_{s',t'}), g(y_{s',t'}))$ generated so far. We set

$$\begin{array}{rcl} h_{s,t}(\alpha) & = & \min_{(p,q)\in Q_{s,t}} \Big[\alpha(p - \underline{\mathrm{Opt}}_{s,t}) + (1-\alpha)q\Big] : [0,1] \to \mathbf{R}, \\ \mathrm{Gap}(s,t) & = & \max\limits_{0\le\alpha\le 1} h_{s,t}(\alpha). \end{array} \tag{3.3.7}$$

4. Let $\Delta_{s,t} = \{\alpha \in [0,1] : h_{s,t}(\alpha) \ge 0\}$, so that $\Delta_{s,t}$ is a segment in $[0,1]$. Unless we have arrived at $\mathrm{Gap}(s,t) = 0$ (i.e., got an optimal solution to (3.3.4), see (3.3.8)), $\Delta_{s,t}$ is not a singleton (since otherwise $\mathrm{Gap}(s,t)$ were 0). Observe also that $\Delta_{s,t}$ are nested: $\Delta_{s',t'} \subset \Delta_{s,t}$ whenever $s' \ge s$, same as whenever $s' = s$ and $t' \ge t$.

   We continue iterations of stage $s$ while $\alpha_s$ is "well-centered" in $\Delta_{s,t}$, e.g., belongs to the mid-third of the segment. When this condition is violated, we start stage $s+1$, specifying $\alpha_{s+1}$ as the midpoint of $\Delta_{s,t}$.

The properties of the aforementioned routine are summarized in the following statement (cf. [50]).

**Proposition 3.3.1.** (i) $\mathrm{Gap}(s,t)$ *is nonincreasing in time. Furthermore, at step* $t$ *of stage* $s$*, we have at our disposal a solution* $\widehat{y}^{s,t} \in Y$ *to* (3.3.4) *such that*

$$f(\widehat{y}^{s,t}) \le \mathrm{Opt} + \mathrm{Gap}(s,t), \quad \textit{and} \quad g(\widehat{y}^{s,t}) \le \mathrm{Gap}(s,t), \tag{3.3.8}$$

*so that* $\widehat{y}^{s,t}$ *belongs to the domain* $Y$ *of problem (3.3.4) and is both* $\mathrm{Gap}(s,t)$*-feasible and* $\mathrm{Gap}(s,t)$*-optimal.*

(ii) *For every* $\epsilon > 0$*, the number* $s(\epsilon)$ *of stages until a pair* $(s,t)$ *with* $\mathrm{Gap}(s,t) \le \epsilon$ *is found obeys the bound*

$$s(\epsilon) \le \frac{\ln\left(3L\epsilon^{-1}\right)}{\ln(4/3)}, \tag{3.3.9}$$

*where $L < \infty$ is an a priori upper bound on $\max_{y\in Y}\max[|f(y)|,|g(y)|]$. Besides this, the number of steps at each stage does not exceed*

$$T(\epsilon) = \min\{t \geq 1 : \epsilon_t \leq \frac{\epsilon}{3}\} + 1. \tag{3.3.10}$$

*Proof.*

**$1^o$.** $h_{s,t}(\alpha)$ are concave piecewise linear functions on $[0,1]$ which clearly are pointwise nonincreasing in time. As a result, $\mathrm{Gap}(s,t)$ is nonincreasing in time. Further, we have

$$\begin{aligned}
\mathrm{Gap}(s,t) &= \max_{\alpha\in[0,1]}\left\{\min_{\lambda}\sum_{(p,q)\in Q_{s,t}}\lambda_{pq}[\alpha(p-\underline{\mathrm{Opt}}_{s,t})+(1-\alpha)q] :\ \lambda_{pq}\geq 0,\ \sum_{(p,q)\in Q_{s,t}}\lambda_{pq}=1\right\}\\
&= \max_{\alpha\in[0,1]}\sum_{(p,q)\in Q_{s,t}}\lambda^*_{pq}[\alpha(p-\underline{\mathrm{Opt}}_{s,t})+(1-\alpha)q]\\
&= \max\left[\sum_{(p,q)\in Q_{s,t}}\lambda^*_{pq}(p-\underline{\mathrm{Opt}}_{s,t}),\ \sum_{(p,q)\in Q_{s,t}}\lambda^*_{pq}q\right],
\end{aligned}$$

where $\lambda^*_{pq}\geq 0$ and sum up to 1. Recalling that for every $(p,q)\in Q_{s,t}$ we have at our disposal $y_{pq}\in Y$ such that $p\geq f(y_{pq})$ and $q\geq g(y_{pq})$, setting $\widehat{y}^{s,t}=\sum\limits_{(p,q)\in Q_{s,t}}\lambda^*_{pq}y_{pq}$ and invoking convexity of $f,g$, we get

$$f(\widehat{y}^{s,t})\leq\sum_{(p,q)\in Q_{s,t}}\lambda^*_{pq}p\leq\underline{\mathrm{Opt}}_{s,t}+\mathrm{Gap}(s,t),\ g(\widehat{y}^{s,t})\leq\sum_{(p,q)\in Q_{s,t}}\lambda^*_{pq}q\leq\mathrm{Gap}(s,t);$$

and (3.3.8) follows, due to $\underline{\mathrm{Opt}}_{s,t}\leq\mathrm{Opt}$.

**$2^o$.** We have $\overline{f}^t_s=\alpha_s f(y^{s,t})+(1-\alpha_s)g(y^{s,t})$ for some $y^{s,t}\in Y$ which we have at our disposal at step $t$, implying that $(\bar{p}=f(y^{s,t}),\bar{q}=g(y^{s,t}))\in Q_{s,t}$. Hence by definition of $h_{s,t}(\cdot)$ it holds

$$h_{s,t}(\alpha_s)\leq\alpha_s(\bar{p}-\underline{\mathrm{Opt}}_{s,t})+(1-\alpha_s)\bar{q}=\overline{f}^t_s-\alpha_s\underline{\mathrm{Opt}}_{s,t}\leq\overline{f}^t_s-\underline{f}_{s,t},$$

where the concluding inequality is given by (3.3.6). Thus, $h_{s,t}(\alpha_s)\leq\overline{f}^t_s-\underline{f}_{s,t}\leq\epsilon_t$. On the other hand, if stage $s$ does not terminate in course of the first $t$ steps, $\alpha_s$ is well-centered in the segment $\Delta_{s,t}$ where the concave function $h_{s,t}(\alpha)$ is nonnegative. We conclude that $0\leq\mathrm{Gap}(s,t)=\max_{0\leq\alpha\leq1}h_{s,t}(\alpha)=\max_{\alpha\in\Delta_{s,t}}h_{s,t}(\alpha)\leq 3h_{s,t}(\alpha_s)$. Thus, if a stage $s$ does

not terminate in course of the first $t$ steps, we have $\mathrm{Gap}(s,t) \le 3\epsilon_t$, which implies (3.3.10). Further, $\alpha_s$ is the midpoint of the segment $\Delta^{s-1} = \Delta_{s-1,t_{s-1}}$, where $t_r$ is the last step of stage $r$ (when $s=1$, we should define $\Delta^0$ as $[0,1]$), and $\alpha_s$ is not well-centered in the segment $\Delta^s = \Delta_{s,t_s} \subset \Delta_{s-1,t_{s-1}}$, which clearly implies that $|\Delta^s| \le \frac{3}{4}|\Delta^{s-1}|$. Thus, $|\Delta^s| \le \left(\frac{3}{4}\right)^s$ for all $s$. On the other hand, when $|\Delta_{s,t}| < 1$, we have $\mathrm{Gap}(s,t) = \max_{\alpha\in\Delta_{s,t}} h_{s,t}(\alpha) \le 3L|\Delta_{s,t}|$ (since $h_{s,t}(\cdot)$ is Lipschitz continuous with constant $3L$ [9] and $h_{s,t}(\cdot)$ vanishes at (at least) one endpoint of $\Delta_{s,t}$). Thus, the number of stages before $\mathrm{Gap}(s,t) \le \epsilon$ is reached indeed obeys the bound (3.3.9). □

### 3.3.3 Sequential Composite Mirror Prox Algorithm and Complexity

Back to our problem of interest, we want to address the following problem

$$\begin{aligned}
\mathrm{Opt} &= \min_{[y^1;\ldots;y^K]\in Y_1\times\cdots\times Y_K} \left\{ f([y^1;\ldots;y^K]) := \textstyle\sum_{k=1}^K [\psi_k(y^k) + \Psi_k(y^k)] : \ \sum_{k=1}^K A_k y^k = b \right\} \\
&= \min_{[y^1;\ldots;y^K]\in Y_1\times\cdots\times Y_K} \left\{ \sum_{k=1}^K \left[\psi_k(y^k) + \Psi_k(y^k)\right] : \ g([y^1;\ldots;y^K]) \le 0 \right\}, \qquad (3.3.11) \\
g([y^1;\ldots;y^K]) &= \pi^*\left(\sum_{k=1}^K A_k y^k - b\right) = \max_{\pi(w)\le 1} \sum_{k=1}^K \langle A_k y^k - b, w\rangle,
\end{aligned}$$

where $\pi(\cdot)$ is some norm and $\pi^*(\cdot)$ is the conjugate norm.

**Problem setting.** We consider the setting as follows. For every $k$, $1 \le k \le K$, we are given

1. Euclidean spaces $E_k$ and $\overline{E}_k$ along with their nonempty closed and bounded convex subsets $Y_k$ and $Z_k$, respectively;

2. proximal setups for $(E_k, Y_k)$ and $(\overline{E}_k, Z_k)$, that is, norms $p_k(\cdot)$ on $E_k$, norms $q_k$ on $\overline{E}_k$, and d.g.f.'s $\omega_k(\cdot) : Y_k \to \mathbf{R}$, $\overline{\omega}_k(\cdot) : Z_k \to \mathbf{R}$, which are compatible with $p_k(\cdot)$ and $q_k(\cdot)$, respectively;

3. linear mapping $y^k \mapsto A_k y^k : E_k \to E$, where $E$ is a Euclidean space;

4. Lipschitz continuous convex functions $\psi_k(y^k) : Y_k \to \mathbf{R}$ along with their *saddle point*

[9] we assume w.l.o.g. that $|\underline{\mathrm{Opt}}_{s,t}| \le L$

*representations*

$$\psi_k(y^k) = \sup_{z^k \in Z_k} [\phi_k(y^k, z^k) - \overline{\Psi}_k(z^k)], \;\; 1 \le k \le K, \tag{3.3.12}$$

where $\phi_k(y^k, z^k) : Y_k \times Z_k \to \mathbf{R}$ are smooth (with Lipschitz continuous gradients) functions convex in $y^k \in Y_k$ and concave in $z^k \in Z_k$, and $\overline{\Psi}_k(z^k) : Z_k \to \mathbf{R}$ are Lipschitz continuous convex functions such that the problems of the form

$$\min_{z^k \in Z_k} \;\; \left[\overline{\omega}_k(z^k) + \langle \xi^k, z^k \rangle + \alpha \overline{\Psi}_k(z^k)\right] \;\; [\alpha > 0] \tag{3.3.13}$$

are easy to solve;

5. Lipschitz continuous convex functions $\Psi_k(y^k) : Y_k \to \mathbf{R}$ such that the problems of the form

$$\min_{y^k \in Y_k} \;\; \left[\omega_k(y^k) + \langle \xi^k, y^k \rangle + \alpha \Psi_k(y^k)\right] \;\; [\alpha > 0]$$

are easy to solve;

6. a norm $\pi^*(\cdot)$ on $E$, with conjugate norm $\pi(\cdot)$, along with a d.g.f. $\widehat{\omega}(\cdot) : \; W := \{w \in E : \pi(w) \le 1\} \to \mathbf{R}$ compatible with $\pi(\cdot)$ and is such that problems of the form

$$\min_{w \in W} [\widehat{\omega}(w) + \langle \xi, w \rangle]$$

are easy to solve.

The outlined data define the sets

$$\begin{aligned} Y_k^+ &= \{[y^k; \tau^k] : \; y^k \in Y_k, \tau^k \ge \Psi_k(y^k)\} \subset E_k^+ := E_k \times \mathbf{R}, \; 1 \le k \le K, \\ Z_k^+ &= \{[z^k; \sigma^k] : \; z^k \in Z_k, \sigma^k \ge \overline{\Psi}_k(z^k)\} \subset \overline{E}_k^+ := \overline{E}_k \times \mathbf{R}, \; 1 \le k \le K. \end{aligned}$$

The problem of interest here is problem (3.3.11), (3.3.12):

$$\begin{aligned} \mathrm{Opt} &= \min_{[y^1;\ldots;y^K]} \max_{[z^1;\ldots;z^K]} \left\{ \sum_{k=1}^K [\phi_k(y^k, z^k) + \Psi_k(y^k) - \overline{\Psi}_k(z^k)] : \;\; \pi^*\left(\sum_{k=1}^K A_k y^k - b\right) \le 0, \right. \\ &\qquad \left. [y^1; \ldots; y^K] \in Y_1 \times \cdots \times Y_K, \;\; [z^1; \ldots; z^k] \in Z_1 \times \cdots \times Z_K \right\} \\ &= \min_{\{[y^k;\tau^k]\}_{k=1}^K} \max_{\{[z^k;\sigma^k]\}_{k=1}^K} \left\{ \sum_{k=1}^K [\phi_k(y^k, z^k) + \tau^k - \sigma^k] : \;\; \max_{w \in W} \sum_{k=1}^K \langle A_k y^k - b, w \rangle \le 0, \right. \\ &\qquad \left. \{[y^k; \tau^k] \in Y_k^+\}_{k=1}^K, \;\; \{[z^k; \sigma^k] \in Z_k^+\}_{k=1}^K, \; w \in W \right\}. \end{aligned} \tag{3.3.14}$$

**Sequential CoMP algorithm.** Using the generic algorithm described in the previous section amounts to resolving a sequence of problems $(P_s)$ as in (3.3.5) where, with a slight abuse of notation,

$$\begin{aligned} Y &= \left\{y = \{[y^k;\tau^k]\}_{k=1}^K :\ [y^k;\tau^k] \in Y_k^+,\ \tau^k \le C_k,\ 1 \le k \le K\right\}; \\ f(y) &= \max_{z=\{[z^k;\sigma^k]\}_{k=1}^K} \left\{\sum_{k=1}^K [\phi_k(y^k,z^k) + \tau^k - \sigma^k] :\ z \in Z = \{[z^k;\sigma^k] \in Z_k^+\}_{k=1}^K\right\}; \\ g(y) &= \max_w \left\{\sum_{k=1}^K \langle A_k y^k - b, w\rangle :\ w \in W\right\}. \end{aligned}$$

Here $C_k \ge \max_{y^k \in Y_k} \Psi_k(y^k)$ are finite constants introduced to make $Y$ compact, as required in the premise of Proposition 3.3.1; it is immediately seen that the magnitudes of these constants (same as their very presence) does not affect the algorithm we are about to describe.

Our sequential CoMP algorithm solves $(P_s)$ by reducing the problem to the saddle point problem

$$\begin{aligned} \overline{\mathrm{Opt}} = \min_y \max_{[z;w]} \Big\{\Phi(y,[z;w]) := \alpha \sum_{k=1}^K [\phi_k(y^k,z^k) + \tau^k - \sigma^k] + (1-\alpha)\sum_{k=1}^K \langle A_k y^k - b, w\rangle : \\ y = \{[y^k;\tau^k]\}_{k=1}^K \in Y,\ [z = \{[z^k;\sigma^k]\}_{k=1}^K \in Z;\ w \in W]\Big\}, \end{aligned} \tag{3.3.15}$$

where $\alpha = \alpha_s$. Setting

$$\begin{aligned} U &= \{u = [y^1;...;y^K;z^1;...;z^K;w] :\ y^k \in Y_k,\ z^k \in Z_k,\ 1 \le k \le K, w \in W\}, \\ X &= \{[u; v = [\tau^1;...;\tau^K;\sigma^1;...;\sigma^K]] :\ u \in U,\ \Psi_k(y^k) \le \tau^k \le C_k,\ \overline{\Psi}_k(z^k) \le \sigma^k,\ 1 \le k \le K\}, \end{aligned}$$

$X$ can be thought of as the domain of the variational inequality associated with (3.3.15), the monotone operator in question being

$$\begin{aligned} F(u,v) &= [F_u(u); F_v], \\ F_u(u) &= \begin{bmatrix} \{\alpha\nabla_y\phi_k(y^k,z^k) + (1-\alpha)A_k^T w\}_{k=1}^K \\ \{-\alpha\nabla_z\phi_k(y^k,z^k)\}_{k=1}^K \\ (1-\alpha)[b - \sum_{k=1}^K A_k y^k] \end{bmatrix}, \\ F_v &= \alpha[1;...;1]. \end{aligned} \tag{3.3.16}$$

By exactly the same reasons as in Section 3.2.2, with properly assembled norm on the embedding space of $U$ and d.g.f., (3.3.15) can be solved by the CoMP algorithm from section 2.5.1. Let us denote

$$\zeta^{s,t} = \left[\widehat{y}^{s,t} = \{[\widehat{y}^k; \widehat{\tau}^k]\}_{k=1}^K \in Y; \left[z^{s,t} \in Z; w^{s,t} \in W\right]\right]$$

the approximate solution obtained in course of $t = 1, 2, ...$ steps of CoMP when solving $(P_s)$, and let

$$\widehat{f}_s^t := \max_{z \in Z, w \in W} \Phi(\widehat{y}^{s,t}, [z; w]) = \alpha \sum_{k=1}^K [\psi_k(\widehat{y}^k) + \widehat{\tau}^k] + (1-\alpha)\pi^* \left(\sum_{k=1}^K A_k \widehat{y}^k - b\right)$$

be the corresponding value of the objective of $(P_s)$. It holds

$$\widehat{f}_s^t - \overline{\mathrm{Opt}} \leq \epsilon_{\mathrm{Sad}}(\zeta^{s,t} \big| Y, Z \times W, \Phi) \leq \epsilon_t := O(1)\mathcal{L}/t, \tag{3.3.17}$$

where $\mathcal{L} < \infty$ is explicitly given by the proximal setup we use and by the related Lipschitz constant of $F_u(\cdot)$ (note that this constant can be chosen to be independent of $\alpha \in [0, 1]$). We assume that computing the corresponding objective value is a part of step $t$ (these computations increase the complexity of a step by factor at most $O(1)$), and thus that $\overline{f}_s^t \leq \widehat{f}_s^t$. By (3.3.17), the quantity $\widehat{f}_s^t - \epsilon_t$ is a valid lower bound on the optimal value of $(P_s)$, and thus we can ensure that $\underline{f}_{s,t} \geq \widehat{f}_s^t - \epsilon_t$. The bottom line is that with the outlined implementation, we have

$$\overline{f}_s^t - \underline{f}_{s,t} \leq \epsilon_t$$

for all $s, t$, with $\epsilon_t$ given by (3.3.17). Consequently, by Proposition 3.3.1, we arrive at

**Theorem 3.3.1.** *The total number of CoMP steps needed to find a belonging to the domain of the problem of interest (3.3.11) $\epsilon$-feasible and $\epsilon$-optimal solution to this problem can be upper-bounded by*

$$O(1) \ln\left(\frac{3L}{\epsilon}\right)\left(\frac{\mathcal{L}}{\epsilon}\right),$$

*where $L$ and $\mathcal{L}$ are readily given by the smoothness parameters of $\phi_k$ and by the proximal setup we use.*

### 3.3.4 Numerical Illustrations: Basis Pursuit

**Basis pursuit** We come back to the simple example of $\ell_1$ minimization problem

$$\min_{x\in X}\{\|x\|_1 : \ Ax = b\} \tag{3.3.18}$$

where $x \in \mathbf{R}^n$, $A \in \mathbf{R}^{m\times n}$ and $m < n$.

Our main purpose here is to test the above sequential CoMP and compare it to the simple approach described in Section 3.3.1 where we directly apply CoMP to the saddle point reformulation of the problem $\min_{x\in X}\{\|x\|_1+R\|Ax-b\|_2\}$ with large enough value of $R$. For the sake of simplicity, we work with the case when $K = 1$ and $X = \{x \in \mathbf{R}^n : \|x\|_2 \le 1\}$.

**Generating the data.** In the experiments to be reported, the data of (3.3.18) were generated as follows. Given $m, n$, we first build a sparse solution $x^*$ by drawing random vector from the standard Gaussian distribution $\mathcal{N}(0, I_n)$, zeroing out part of the entries and scaling the resulting vector to enforce $x^* \in X$. We also build a dual solution $\lambda^*$ by scaling a random vector from distribution $\mathcal{N}(0, I_m)$ to satisfy $\|\lambda^*\|_2 = R_*$ for a prescribed $R_*$. Next we generate $A$ and $b$ such that $x^*$ and $\lambda^*$ are indeed the optimal primal and dual solutions to the $\ell_1$ minimization problem (3.3.18), i.e. $A^T\lambda^* \in \partial\big|_{x=x^*}\|x\|_1$ and $Ax^* = b$. To achieve this, we set

$$A = \frac{1}{\sqrt{n}}\widehat{F}_n + pq^T, \ b = Ax^*$$

where $p = \frac{\lambda^*}{\|\lambda^*\|_2^2}$, $q \in \partial\big|_{x=x^*}\|x\|_1 - \frac{1}{\sqrt{n}}\widehat{F}_n\lambda^*$, and $\widehat{F}_n$ is a $m \times n$ submatrix randomly selected from the DFT matrix $F_n$. We expect that the larger is the $\|\cdot\|_2$-norm $R_*$ of the dual solution, the harder is problem (3.3.18).

**Implementing the sequential CoMP algorithm.** When implementing the algorithm, we apply at each stage $s = 1, 2, ...$ CoMP to the saddle point problem

$$(P_s): \qquad \min_{x,\tau:\ \|x\|_2\le 1, \tau\ge\|x\|_1} \max_{w:\|w\|_2\le 1} \{\alpha_s\tau + (1-\alpha_s)\langle Ax-b, w\rangle\}.$$

The proximal setup for CoMP is given by equipping the embedding space of $U = \{u = [x; w] : x \in X, \|w\|_2 \le 1\}$ with the norm $\|u\|_2 = \sqrt{\frac{1}{2}\|x\|_2^2 + \frac{1}{2}\|w\|_2^2}$ and equipping $U$ with

Table 5: Composite Mirror Prox algorithms on problem (3.3.18). Platform: ISyE Condor Cluster

| $n$ | $m$ | $c$ | **sequential CoMP** | | **simple CoMP** | |
|---|---|---|---|---|---|---|
| | | $(R_* = c \cdot n)$ | steps | CPU(sec) | steps | CPU(sec) |
| 1024 | 512 | 1 | 7653 | 18.68 | 31645 | 67.78 |
| | | 5 | 43130 | 44.66 | 90736 | 90.67 |
| | | 10 | 48290 | 49.04 | 93989 | 93.28 |
| 4096 | 2048 | 1 | 28408 | 85.83 | 46258 | 141.10 |
| | | 5 | 45825 | 199.96 | 93483 | 387.88 |
| | | 10 | 52082 | 179.10 | 98222 | 328.31 |
| 16384 | 8192 | 1 | 43646 | 358.26 | 92441 | 815.97 |
| | | 5 | 48660 | 454.70 | 93035 | 784.05 |
| | | 10 | 55898 | 646.36 | 101881 | 1405.80 |
| 65536 | 32768 | 1 | 45153 | 3976.51 | 92036 | 4522.43 |
| | | 5 | 55684 | 4138.62 | 100341 | 8054.35 |
| | | 10 | 69745 | 6214.18 | 109551 | 9441.46 |
| 262144 | 131072 | 1 | 46418 | 6872.64 | 96044 | 14456.99 |
| | | 5 | 69638 | 10186.51 | 109735 | 16483.62 |
| | | 10 | 82365 | 12395.67 | 95756 | 13634.60 |

the d.g.f. $\omega(u) = \frac{1}{2}\|x\|_2^2 + \frac{1}{2}\|w\|_2^2$. In the sequel we refer to the resulting algorithm as *sequential* CoMP. For comparison, we solve the same problem by applying CoMP to the saddle point problem

$$(P_R): \quad \min_{x,\tau:\ \|x\|_2\le 1, \tau\ge\|x\|_1} \max_{w:\|w\|_2\le 1} \{\tau + R\langle Ax - b, w\rangle\}$$

with $R = R_*$; the resulting algorithm is referred to as *simple* CoMP. Both sequential CoMP and simple CoMP algorithms are terminated when the relative nonoptimality and constraint violation are both less than $\epsilon = 10^{-5}$, namely,

$$\epsilon(x) := \max\left\{\frac{\|x\|_1 - \|x_*\|_1}{\|x_*\|_1}, \|Ax - b\|_2\right\} \le 10^{-5}.$$

**Numerical results** are presented in Table 5. One can immediately see that to achieve the desired accuracy, the simple CoMP with $R$ set to $R_*$, i.e., to the exact magnitude of the true Lagrangian multiplier, requires almost twice as many steps as the sequential CoMP. In more realistic examples, the simple CoMP will additionally suffer from the fact that the magnitude of the optimal Lagrange multiplier is not known in advance, and the penalty $R$ in $(P_R)$ should be somehow tuned "online."

### 3.3.5 Concluding Remarks

In this section, we investigate the family of semi-separable problems, which generalizes the multi-term composite minimization problem discussed in the previous section. We propose a sequential CoMP algorithm which solves a sequence of saddle point subproblems using CoMP algorithm. The algorithm achieves an overall $O(1/\epsilon)$ complexity bound up to some log factors, which to the best of our knowledge, is nearly optimal. The framework we established here is rathere general and can be easily extended to nonlinear constraints as well.

## *3.4 Application III: Norm-Regularized Nonsmooth Minimization*

### 3.4.1 Problem of Interest

We consider the composite minimization problem

$$\text{Opt} = \min_{x \in X} F(x) := f(x) + \|\mathcal{B}x\| \tag{3.4.1}$$

where $X$ is a closed convex set in the Euclidean space $E_x$; $x \mapsto \mathcal{B}x$ is a linear mapping from $X$ to $Y(\supset \mathcal{B}X)$, where $Y$ is a closed convex set in the Euclidean space $E_y$. A wide range of machine learning and signal processing problems can be formulated in the above form. $f$ is can be either smooth, or nonsmooth yet enjoys a particular structure. The term $\|\mathcal{B}x\|$ defines a regularization penalty through a norm $\|\cdot\|$. We make two important assumptions on the function $f$ and the norm $\|\cdot\|$ defining the regularization penalty, explained below.

1. *Saddle point representation:* We assume that $f(x)$ is a perhaps non-smooth convex function given by [10]

$$f(x) = \max_{z \in Z} \Phi(x, z) \tag{3.4.2}$$

where $\Phi(x, z)$ is a smooth convex-concave function and $Z$ is a convex and compact set in the Euclidean space $E_z$. saddle point representability can be interpreted as a general form of the smoothing-favorable structure of non-smooth functions used in the Nesterov smoothing technique [62]. Representations of this type are readily available

[10] Notice that this can be relaxed a more general representation, $f(x) = \max_{z \in Z}\{\Phi(x, z) - \psi(z)\}$, where $\psi(z)$ admits easy-to-compute proximal operators or linear minimization oracles (explained in the next).

for a wide family of "well-structured" nonsmooth functions $f$ (see several examples provided below and also examples discussed in previous chapter), and actually for all empirical risk functions with convex loss in machine learning, up to our knowledge.

2. *Composite Linear Minimization Oracle (LMO):* We assume that we have at our disposal, the LMO routine which, given an input $\alpha > 0$ and $\eta \in E_y$, returns a point

$$\min_{y \in Y} \{\langle \eta, y \rangle + \alpha \|y\|\}. \tag{3.4.3}$$

Proximal-gradient-type algorithms, including the composite Mirror Prox algorithm developed in Chapter 2, require the computation of a proximal operator at each iteration, i.e.

$$\min_{y \in Y} \{\omega(y) + \langle \eta, y \rangle + \alpha \|y\|\}, \tag{3.4.4}$$

where $\omega(\cdot)$ is some distance generating function and in the usual Euclidean setup, $\omega(\cdot) = \frac{1}{2}\|\cdot\|_2^2$. For several cases of interest, described below, the computation of the proximal operator can be expensive or intractable. A classical example is the nuclear norm, whose proximal operator boils down to singular value thresholding, therefore requiring a full singular value decomposition. In contrast to the proximal operator, the composite linear minimization oracle can be much cheaper. In the case of the nuclear-norm, the LMO only requires the computation of the leading pair of singular vectors, which is by order of magnitude faster than full singular value decomposition.

**Remark.** The first option to minimize $F$ is to use the so-called Nesterov smoothing technique [62] with a conditional gradient or Frank-Wolfe algorithm to minimize the smooth approximation of $F$, based on LMO routines, see e.g. [48, 70]. Another option is to pass to the dual problem and solve by some first-order algorithm (e.g. [26]). However, both options require either a more restricted saddle point representation, often with linear-in-$x$ function $\Phi$ or good geometries of the dual domain $Z$. Moreover, neither option takes advantage of the composite structure of the objective (3.4.1) or handles the case when the linear mapping $\mathcal{B}$ is nontrivial.

**Contribution.** In this section, we propose a new algorithm, called Semi-Proximal Mirror-Prox , which is based on the inexact CoMP algorithm for solving the difficult non-smooth composite optimization problem (3.4.1). The Semi-Proximal Mirror-Prox relies upon i) saddle point representability of $f$; ii) linear minimization oracle associated with $\|\cdot\|$ in the domain $X$. While the saddle point representability of $f$ allows to handle the non-smoothness of $f$, the linear minimization over the domain $X$ allows to tackle the non-smooth regularization penalty $\|\cdot\|$. We establish the theoretical convergence rate of Semi-Proximal Mirror-Prox, which exhibits the *optimal complexity bounds*, i.e. $O(1/\epsilon^2)$, for the number of calls to linear minimization oracle. Furthermore, Semi-Proximal Mirror-Prox and its stochastic variant generalize previously proposed approaches and improve upon them in special cases:

1. Case $\mathcal{B} \equiv 0$: Semi-Proximal Mirror-Prox does not require assumptions on favorable geometry of dual domains $Z$ or simplicity of $\Phi(\cdot)$ in (3.4.2).

2. Case $\mathcal{B} = \mathbb{I}$: Semi-Proximal Mirror-Prox is competitive with previously proposed approaches [49, 70] based on smoothing techniques.

3. Case of non-trivial $\mathcal{B}$: Semi-Proximal Mirror-Prox is the first conditional-gradient-type optimization algorithm for (3.4.1).

**Related work** The Semi-Proximal Mirror-Prox algorithm belongs the family of conditional gradient algorithms, whose most basic instance is the Frank-Wolfe algorithm for constrained smooth optimization using a linear minimization oracle; see [41, 5, 10]. Recently, in [26, 44], the authors consider constrained non-smooth optimization when the domain $Z$ has a "favorable geometry", *i.e.* the domain is amenable to proximal setups (favorable geometry), and establish a complexity bound with $O(1/\epsilon^2)$ calls to the linear minimization oracle. Recently, in [49], a method called conditional gradient sliding is proposed to solve similar problems, using a smoothing technique, with a complexity bound in $O(1/\epsilon^2)$ for the calls to the linear minimization oracle (LMO) and additionally a $O(1/\epsilon)$ bound for the linear operator evaluations. Actually, this $O(1/\epsilon^2)$ bound for the LMO complexity can be

shown to be indeed *optimal* for conditional-gradient-type or LMO-based algorithms, when solving general non-smooth convex problems [48].

Conditional-gradient-type algorithms were recently proposed for composite objectives [32, 36, 85, 70, 55, 31], but cannot be applied for our problem. In [36], $f$ is smooth and $\mathcal{B}$ is identity matrix, whereas in [70], $f$ is non-smooth and $\mathcal{B}$ is also the identity matrix. The proposed Semi-Proximal Mirror-Prox can be seen as a blend of the successful components resp. of the Composite Conditional Gradient algorithm [36] and the Composite Mirror-Prox [39], that enjoys the optimal complexity bound $O(1/\epsilon^2)$ on the total number of LMO calls, yet solves a broader class of convex problems than previously considered.

**Outline** The rest of this section is organized as follows. In Section 3.4.2, we present a composite conditional gradient method tailored for smooth semi-linear problems. In Section 3.4.3, we present the conditional gradient type method based on an inexact Mirror-Prox framework for structured variational inequalities. In Section 3.4.4, we present promising experimental results showing the interest of the approach in comparison to competing methods, resp. on a collaborative filtering for movie recommendation and link prediction for social network analysis applications.

### 3.4.2 Composite Conditional Gradient

We first introduce a variant of the composite conditional gradient algorithm, denoted CCG, tailored for a particular class of problems, which we call *smooth semi-linear problems*. The composite conditional gradient algorithm was first introduced in [36] and also developed in [65]. We present an extension here which turns to be especially well suited for subproblems that will be solved in Section 3.4.3.

**Minimizing smooth semi-linear functions.** We consider the smooth semi-linear problem

$$\min_{x=[u;v]\in X} \left\{\phi^+(u,v) = \phi(u) + \langle \theta, v\rangle\right\} \tag{3.4.5}$$

represented by the pair $(X;\phi^+)$ such that the following assumptions are satisfied. We assume that

i) $X \subset E_u \times E_v$ is closed convex and its projection $PX$ on $E_x$ belongs to a convex and compact set $U$;

ii) $\phi(u) : U \to \mathbf{R}$ is a convex continuously differentiable function, and there exists $1 < \kappa \leq 2$ and $L_0 < \infty$ such that

$$\phi(u') \leq \phi(u) + \langle \nabla\phi(u), u' - u\rangle + \frac{L_0}{\kappa}\|u' - u\|^\kappa \ \forall u, u' \in U; \tag{3.4.6}$$

iii) $\theta \in E_v$ is such that every linear function on $E_u \times E_v$ of the form

$$[u; v] \mapsto \langle \eta, u\rangle + \langle \theta, v\rangle \tag{3.4.7}$$

with $\eta \in E_u$ attains its minimum on $X$ at some point $x[\eta] = [u[\eta]; v[\eta]]$; we have at our disposal a *Composite Linear Minimization Oracle* (LMO) which, given on input $\eta \in E_u$, returns $x[\eta]$.

**Algorithm 5** Composite Conditional Gradient Algorithm $\mathbf{CCG}(X, \phi(\cdot), \theta; \epsilon)$

**Input:** accuracy $\epsilon > 0$ and $\gamma_t = 2/(t+1), t = 1, 2, \ldots$
Initialize $x^1 = [u^1; v^1] \in X$ and
**for** $t = 1, 2, \ldots$ **do**
  Compute $\delta_t = \langle g_t, u^t - u^t[g_t]\rangle + \langle \theta, v^t - v^t[g_t]\rangle$, where $g_t = \nabla\phi(u^t)$;
  **if** $\delta_t \leq \epsilon$ **then**
    Return $x^t = [u^t; v^t]$
  **else**
    Find $x^{t+1} = [u^{t+1}; v^{t+1}] \in X$ such that $\phi^+(x^{t+1}) \leq \phi^+\left(x^t + \gamma_t(x^t[g_t] - x^t)\right)$
  **end if**
**end for**

Note that CCG works essentially *as if* there were no $v$-component at all. The CCG algorithm enjoys convergence rate in $O(t^{-(\kappa-1)})$ in the evaluations of the function $\phi^+$, and the optimality gap $(\delta_t)$ goes to zero at the same rate $O(t^{-(\kappa-1)})$ as well, when solving problems of type (3.4.5).

**Proposition 3.4.1.** *Denote $D$ the $\|\cdot\|$-diameter of $U$. When solving problems of type (3.4.5), the sequence of iterates $(x^t)$ of CCG satisfies*

$$\epsilon_t := \phi^+(x^t) - \min_{x \in X} \phi^+(x) \leq \frac{2L_0 D^\kappa}{\kappa(3-\kappa)}\left(\frac{2}{t+1}\right)^{\kappa-1}, \ t \geq 2 \tag{3.4.8}$$

*In addition, the optimality gap* $(\delta_t)$ *satisfy*

$$\min_{1\le s\le t}\ \delta_s \le O(1)L_0D^\kappa\left(\frac{2}{t+1}\right)^{\kappa-1},\ t\ge 2. \tag{3.4.9}$$

*Proof.*

**$1^0$.** The projection of $X$ onto $E_u$ is contained in $U$, whence

$$\|u[\nabla\phi(u^t)]-u^t\|\le D, \forall t=1,2,\dots.$$

This observation, due to the structure of $\phi^+$, implies that whenever $x,x'\in X$ and $\gamma\in[0,1]$, we have

$$\phi^+(x+\gamma(x^+-x))\le\phi^+(x)+\gamma\langle\nabla\phi^+(x),x'-x\rangle+\frac{L_0D^\kappa}{\kappa}\gamma^\kappa. \tag{3.4.10}$$

Setting $x_+^t=x^t+\gamma_t(x[\nabla\phi(u^t)]-x^t)$ and $\gamma_t=2/(t+1)$, we have

$$\begin{aligned}\epsilon_{t+1}&\le\phi^+(x_+^t)-\min_{x\in X}\phi^+(x) &(3.4.11)\\ &\le\epsilon_t+\gamma_s\langle\nabla\phi(x^t),x[\nabla\phi^+(x^t)]-x\rangle+\frac{L_0D^\kappa}{\kappa}\gamma_t^\kappa &(3.4.12)\\ &=\epsilon_t-\gamma_t\delta_t+\frac{L_0D^\kappa}{\kappa}\gamma_t^\kappa, &(3.4.13)\end{aligned}$$

whence, due to $\delta_t\ge\epsilon_t\ge 0$,

$$\begin{aligned}&(i)\quad \epsilon_{t+1}\le(1-\gamma_t)\epsilon_t+\frac{L_0D^\kappa}{\kappa}\gamma_t^\kappa,\ t=1,2,...,\\ &(ii)\quad \gamma_s\delta_s\le\epsilon_s-\epsilon_{s+1}+\frac{L_0D^\kappa}{\kappa}\gamma_s^\kappa,\ s=1,2,...\end{aligned} \tag{3.4.14}$$

**$2^0$.** Let us prove (3.4.8) by induction on $s\ge 2$. By (3.4.14.i) and due to $\gamma_1=1$ we have

$$\epsilon_2\le\frac{L_0D^\kappa}{\kappa}\ . \tag{3.4.15}$$

Whence, due to $\gamma_2=2/3$ and $1<\kappa\le 2$, we get

$$\epsilon_2\le\frac{2L_0D^\kappa}{\kappa(3-\kappa)}\gamma_2^{\kappa-1}\ . \tag{3.4.16}$$

Now, assume that,for some $t\ge 2$

$$\epsilon_t\le\frac{2L_0D^\kappa}{\kappa(3-\kappa)}\gamma_t^{\kappa-1}\ . \tag{3.4.17}$$

Then, invoking (3.4.14.i),

$$\begin{aligned}\epsilon_{t+1} &\le \frac{2L_0D^\kappa}{\kappa(3-\kappa)}\gamma_t^{\kappa-1}(1-\gamma_t)+\frac{L_0D^\kappa}{\kappa}\gamma_t^\kappa\\ &\le \frac{2L_0D^\kappa}{\kappa(3-\kappa)}\left[\gamma_t^{\kappa-1}-\frac{\kappa-1}{2}\gamma_t^\kappa\right]\\ &\le \frac{2L_0D^\kappa}{\kappa(3-\kappa)}2^{\kappa-1}\left[(t+1)^{1-\kappa}+(1-\kappa)(t+1)^{-\kappa}\right]\end{aligned}$$

Therefore, by convexity of $(t+1)^{1-\kappa}$ in $t$

$$\epsilon_{t+1} \le \frac{2L_0D^\kappa}{\kappa(3-\kappa)}2^{\kappa-1}(t+2)^{1-\kappa} = \frac{2L_0D^\kappa}{\kappa(3-\kappa)}\gamma_{t+1}^{\kappa-1}$$

The induction is completed.

**$3^0$.** To prove (3.4.9), given $s \ge 2$, let $t_- = \mathrm{Ceil}(\max[2, t/2])$. Summing up inequalities (3.4.14.ii) over $t_- \le s \le t$, we get

$$\left(\min_{1\le s\le t}\delta_s\right)\sum_{s=t_-}^t\gamma_s \le \sum_{s=t_-}^t\gamma_s\delta_s \le \epsilon_{t_-}-\epsilon_{t+1}+\frac{L_0D^\kappa}{2}\sum_{s=t_-}^t\gamma_s^\kappa \le O(1)L_0D^\kappa\gamma_t^{\kappa-1}$$

and $\sum_{s=t_-}^t\gamma_s \ge O(1)$, and (3.4.9) follows.

□

### 3.4.3 Semi-Proximal Mirror Prox Algorithm and Complexity

**Saddle Point Reformulation.** The crux of our approach for solving (3.4.1) is a smooth convex-concave saddle point reformulation. After massaging the saddle-point reformulation, we consider the variational inequality associated with the obtained saddle-point problem. We rewrite (3.4.1) in epigraph form

$$\min_{x\in X, y\in Y, \tau\ge\|y\|}\max_{z\in Z}\left\{\Phi(x,z)+\tau : y=\mathcal{B}x\right\},$$

which, with a properly selected $\rho > 0$, can be further approximated by

$$\widehat{\mathrm{Opt}} = \min_{x\in X, y\in Y, \tau\ge\|y\|}\max_{z\in Z}\left\{\Phi(x,z)+\tau+\rho\|y-\mathcal{B}x\|_2\right\} \tag{3.4.18}$$

$$= \min_{x\in X, y\in Y, \tau\ge\|y\|}\max_{z\in Z,\|w\|_2\le 1}\left\{\Phi(x,z)+\tau+\rho\langle y-\mathcal{B}x, w\rangle\right\}. \tag{3.4.19}$$

As discussed in Section 3.2.3, when $\rho$ is large enough, one can always guarantee $\widehat{\mathrm{Opt}} = \mathrm{Opt}$. It is indeed sufficient to set $\rho$ as the Lipschitz constant of $\|\cdot\|$ with respect to $\|\cdot\|_2$.

Introduce the variables $u := [x, y; z, w]$ and $v := \tau$. The variational inequality associated with the above saddle point problem is fully described by the domain

$$X_+ \quad = \quad \{x_+ = [u; v] : x \in X, y \in Y, z \in Z, \|w\|_2 \leq 1, \tau \geq \|y\|\}$$

and the monotone vector field is of the form

$$F(x_+ = [u; v]) = [F_u(u); F_v] \ ,$$

where

$$F_u\left(u = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}\right) = \begin{bmatrix} \nabla_x \Phi(x, z) - \rho \mathcal{B}^T w \\ \rho w \\ -\nabla_z \Phi(x, z) \\ \rho(\mathcal{B}x - y) \end{bmatrix}, \qquad F_v(v = \tau) = 1.$$

Notice that this essentially belongs to the family of structured variational inequalities discussed in Section 2.3.1. This implies that when computing proximal operator of $\|\cdot\|$ is easy, the problem (3.4.1) can be efficiently solved by the CoMP algorithm developed in Section 2.5.1. However, this is certainly not the case we are interested in here. In the next section, we present an efficient algorithm to solve this type of variational inequalities, in the sequel referred to as *semi-structured* ones. The family of semi-structured variational inequalities covers both cases that we discussed so far in Section 2.3.1 and 3.4.2. But most importantly, it also covers many other problems that do not fall into these two regimes and in particular, including our problem of interest (3.4.1). We are about to explain what a semi-structured variational inequality is.

**Semi-structured Variational Inequalities.** The class of semi-structured variational inequalities allows to go beyond Assumptions $(\mathbf{A}.1) - (\mathbf{A}.4)$, by assuming more structure. This structure is consistent with what we call a *semi-proximal* setup, which encompasses both the regular *proximal setup* and the regular *linear minimization setup* as special cases. Specifically, we say that $\mathrm{VI}(X, F)$ is a semi-structured variational inequality, if, in addition to Assumptions $(\mathbf{A}.1) - (\mathbf{A}.4)$, the following assumptions are satisfied:

(**S**.1) *Proximal setup for $X$*: we assume that $E_u = E_{u_1} \times E_{u_2}$, $E_v = E_{v_1} \times E_{v_2}$, and $U \subset U_1 \times U_2$, $X = X_1 \times X_2$ with $X_i \in E_{u_i} \times E_{v_i}$ and $P_i X = \{u_i : [u_i; v_i] \in X_i\} \subset U_i$ for $i = 1, 2$, where $U_1$ is convex and closed, $U_2$ is convex and compact. We also assume that $\omega(u) = \omega_1(u_1) + \omega_2(u_2)$ and $\|u\| = \|u_1\|_{E_{u_1}} + \|u_2\|_{E_{u_2}}$, with $\omega_2(\cdot) : U_2 \to \mathbf{R}$ continuously differentiable such that

$$\omega_2(u_2') \le \omega_2(u_2) + \langle \nabla\omega_2(u_2), u_2' - u_2\rangle + \frac{L_0}{\kappa}\|u_2' - u_2\|^\kappa_{E_{u_2}}, \forall u_2, u_2' \in U_2;$$

for a particular $1 < \kappa \le 2$ and $L_0 < \infty$. Furthermore, we assume that the $\|\cdot\|_{E_{u_2}}$-diameter of $U_2$ is bounded by some $D > 0$.

(**S**.2) *Partition of $F$*: the operator $F$ induced by the above partition of $X_1$ and $X_2$ can be written as

$$F(x) = [F_u(u); F_v] \text{ with } F_u(u) = [F_{u_1}(u_1, u_2); F_{u_2}(u_1, u_2)], F_v = [F_{v_1}; F_{v_2}].$$

(**S**.3) *Proximal mapping on $X_1$*: we assume that for any $\eta_1 \in E_{u_1}$ and $\alpha > 0$, we have at our disposal easy-to-compute prox-mappings of the form,

$$\mathrm{Prox}_{\omega_1}(\eta_1, \alpha) := \operatorname*{argmin}_{x_1 = [u_1; v_1] \in X_1} \{\omega_1(u_1) + \langle \eta_1, u_1\rangle + \alpha\langle F_{v_1}, v_1\rangle\}.$$

(**S**.4) *Linear minimization oracle for $X_2$*: we assume that we we have at our disposal Composite Linear Minimization Oracle (LMO), which given any input $\eta_2 \in E_{u_2}$ and $\alpha > 0$, returns an optimal solution to the minimization problem with linear form, that is,

$$\mathrm{LMO}(\eta_2, \alpha) \in \operatorname*{argmin}_{x_2 = [u_2; v_2] \in X_2} \{\langle \eta_2, u_2\rangle + \alpha\langle F_{v_2}, v_2\rangle\}.$$

**Semi-proximal setup** We denote the family of semi-structured variational inequality problems as Semi-VI$(X, F)$. On the one hand, when $U_2$ is a singleton, we get the *full-proximal setup*. On the other hand, when $U_1$ is a singleton, we get the *full linear-minimization-oracle setup* (full LMO setup). In the gray zone in between, we get the *semi-proximal setup*.

**The Semi-Proximal Mirror-Prox algorithm.** We finally present here, the Semi-Proximal Mirror-Prox algorithm, which solves the semi-structured variational inequality under assumptions $(\mathbf{A}.1) - (\mathbf{A}.4)$ and $(\mathbf{S}.1) - (\mathbf{S}.4)$. The Semi-Proximal Mirror-Prox algorithm blends both CoMP and CCG. Basically, for sub-domain $X_2$ given by LMO, instead of computing exactly the prox-mapping, we mimick inexactly the prox-mapping via a conditional gradient algorithm in the inexact CoMP algorithm discussed in Section 2.5.3. For the sub-domain $X_1$, we compute the prox-mapping as it is.

**Description of the Semi-Proximal Mirror-Prox algorithm** Basically, at step $t$, we first update $y_1^t = [\widehat{u}_1^t; \widehat{v}_1^t]$ by computing the exact prox-mapping and build $y_2^t = [\widehat{u}_2^t; \widehat{v}_2^t]$ by running the composite conditional gradient algorithm to problem (3.4.5) specifically with

$$X = X_2, \phi(\cdot) = \omega_2(\cdot) + \langle \gamma_t F_{u_2}(u_1^t, u_2^t) - \omega_2'(u_2^t), \cdot \rangle, \text{ and } \theta = \gamma_t F_{v_2},$$

until $\delta(y_2^t) = \max_{y_2 \in X_2} \langle \nabla \phi^+(y_2^t), y_2^t - y_2 \rangle \leq \epsilon_t$. We then build $x_1^{t+1} = [u_1^{t+1}; v_1^{t+1}]$ and $x_2^{t+1} = [u_2^{t+1}; v_2^{t+1}]$ similarly except this time taking the value of the operator at point $y^t$. Combining the results in Theorem 2.5.2 and Proposition 3.4.1, we arrive at the following complexity bound.

**Algorithm 6 Semi-Proximal Mirror-Prox** Algorithm for Semi-VI$(X, F)$

**Input:** stepsizes $\gamma_t > 0$, accuracies $\epsilon_t \geq 0$, $t = 1, 2, \ldots$
[1] Initialize $x^1 = [x_1^1; x_2^1] \in X$, where $x_1^1 = [u_1^1; v_1^1]; x_2^1 = [u_2^1, ; v_2^1]$.
**for** $t = 1, 2, \ldots, T$ **do**
[2] Compute $y^t = [y_1^t; y_2^t]$ according to

$$\begin{array}{rcl} y_1^t := [\widehat{u}_1^t; \widehat{v}_1^t] & = & \mathrm{Prox}_{\omega_1}(\gamma_t F_{u_1}(u_1^t, u_2^t) - \omega_1'(u_1^t), \gamma_t) \\ y_2^t := [\widehat{u}_2^t; \widehat{v}_2^t] & = & \mathbf{CCG}(X_2, \omega_2(\cdot) + \langle \gamma_t F_{u_2}(u_1^t, u_2^t) - \omega_2'(u_2^t), \cdot \rangle, \gamma_t F_{v_2}; \epsilon_t) \end{array}$$

[3] Compute $x^{t+1} = [x_1^{t+1}; x_2^{t+1}]$ according to

$$\begin{array}{rcl} x_1^{t+1} := [u_1^{t+1}; v_1^{t+1}] & = & \mathrm{Prox}_{\omega_1}(\gamma_t F_{u_1}(\widehat{u}_1^t, \widehat{u}_2^t) - \omega_1'(u_1^t), \gamma_t) \\ x_2^{t+1} := [u_2^{t+1}; v_2^{t+1}] & = & \mathbf{CCG}(X_2, \omega_2(\cdot) + \langle \gamma_t F_{u_2}(\widehat{u}_1^t, \widehat{u}_2^t) - \omega_2'(u_2^t), \cdot \rangle, \gamma_t F_{v_2}; \epsilon_t) \end{array}$$

**end for**
**Output:** $\overline{x}_T := [\bar{u}_T; \bar{v}_T] = \left(\sum_{t=1}^T \gamma_t\right)^{-1} \sum_{t=1}^T \gamma_t y^t$

**Proposition 3.4.2.** *Under the assumption* $(\boldsymbol{A}.1) - (\boldsymbol{A}.4)$ *and* $(\boldsymbol{S}.1) - (\boldsymbol{S}.4)$ *with* $M = 0$, *and choice of stepsize being* $\gamma_t = L^{-1}, t = 1, \ldots, T$, *for the outlined algorithm to return an*

*$\epsilon$-solution to the Semi-VI$(X, F)$, the total number of Mirror Prox steps required does not exceed*

$$\textit{Total number of steps} = O(1)\frac{L\Theta[X]}{\epsilon}$$

*and the total number of calls to the Linear Minimization Oracle does not exceed*

$$\mathcal{N} = O(1)\left(\frac{L_0 L^\kappa D^\kappa}{\epsilon^\kappa}\right)^{\frac{1}{\kappa-1}}\Theta[X].$$

*In particular, if we use Euclidean proximal setup on $U_2$ with $\omega_2(\cdot) = \frac{1}{2}\|x_2\|^2$, which leads to $\kappa = 2$ and $L_0 = 1$, then the number of LMO calls does not exceed $\mathcal{N} = O(1)\left(L^2D^2(\Theta[X_1] + D^2)\right)/\epsilon^2$.*

*Proof.* Let us fix $T$ as the number of Mirror prox steps, and since $M = 0$, from Theorem 2.5.2, the efficiency estimate of the variational inequality implies that

$$\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F) \le \frac{L(\Theta[X] + 2\sum_{t=1}^T \epsilon_t)}{T}.$$

Let us fix $\epsilon_t = \frac{\Theta[X]}{2T}$ for each $t = 1, \ldots, T$, then from Proposition 3.4.1, it takes at most $s = O(1)(\frac{L_0 D^\kappa T}{\Theta[X]})^{1/(\kappa-1)}$ calls to the LMO oracles to generate a point such that $\Delta_s \le \epsilon_t$. Moreover, we have

$$\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F) \le 2\frac{L\Theta[X]}{T}.$$

Therefore, to ensure $\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F) \le \epsilon$ for a given accuracy $\epsilon > 0$, the number of Mirror Prox steps $T$ is at most $O(\frac{L\Theta[X]}{\epsilon})$ and the number of LMO calls on $X_2$ needed is at most

$$\mathcal{N} = O(1)\Big(\frac{L_0 L^\kappa D^\kappa}{\epsilon^\kappa}\Big)^{1/(\kappa-1)}\Theta[X].$$

In particular, if $\kappa = 2$ and $L_0 = 1$, this becomes $\mathcal{N} = O(1)\frac{L^2D^2\Theta[X]}{\epsilon^2}$. □

**Remark.** The proposed Semi-Proximal Mirror-Prox algorithm enjoys the *optimal complexity bounds*, i.e. $O(1/\epsilon^2)$, in the number of calls to LMO; see [48] for the optimal complexity bounds for general non-smooth optimization with LMO. Consequently, when applying the algorithm to the variational reformulation of the problem of interest (3.4.1), we are able to get an $\epsilon$-optimal solution within at most $O(1/\epsilon^2)$ LMO calls.

**Stochastic extension.** The Semi-Proximal Mirror-Prox algorithm is readily extensible to the situation when we only have access to stochastic oracles on the monotone operator, as discussed in Section 2.5.4. Assume that we are under assumptions (**C**.1) and (**C**.2) from Section 2.5.4. At each iteration $t$, the stochastic oracle returns a sequence of stochastic estimates $g(u,\xi_j), j=1,\dots,2m_t$ with input being $u\in U$, where $\{\xi_j\}_{j=1}^{2m_t}$ are i.i.d. random variables. We provide below the stochastic variant of Semi-Proximal Mirror-Prox algorithm for completeness.

**Algorithm 7 Stochastic Semi-Proximal Mirror-Prox** Algorithm for Semi-VI$(X,F)$

**Input:** stepsizes $\gamma_t>0$, accuracies $\epsilon_t\geq 0$, $t=1,2,\dots$
[1] Initialize $x^1=[x_1^1;x_2^1]\in X$, where $x_1^1=[u_1^1;v_1^1]; x_2^1=[u_2^1;v_2^1]$.
**for** $t=1,2,\dots,T$ **do**
[2] Set $u^t=[u_1^t;u_2^t]$ and compute $y^t=[y_1^t;y_2^t]$ that

$$\begin{array}{rcl} y_1^t:=[\widehat{u}_1^t;\widehat{v}_1^t] & = & \mathrm{Prox}_{\omega_1}(\gamma_t g_1^t-\omega_1'(u_1^t),\gamma_t)\\ y_2^t:=[\widehat{u}_2^t;\widehat{v}_2^t] & = & \mathbf{CCG}(X_2,\omega_2(\cdot)+\langle\gamma_t g_2^t-\omega_2'(u_2^t),\cdot\rangle,\gamma_t F_{v_2};\epsilon_t)\end{array}$$

where $[g_1^t;g_2^t]=\frac{1}{m_t}\sum_{j=1}^{m_t} g(u^t,\xi_j^t)$.
[3] Set $\widehat{u}^t=[\widehat{u}_1^t;\widehat{u}_2^t]$ and compute $x^{t+1}=[x_1^{t+1};x_2^{t+1}]$ that

$$\begin{array}{rcl} x_1^{t+1}:=[u_1^{t+1};v_1^{t+1}] & = & \mathrm{Prox}_{\omega_1}(\gamma_t \widehat{g}_1^t-\omega_1'(u_1^t),\gamma_t)\\ x_2^{t+1}:=[u_2^{t+1};v_2^{t+1}] & = & \mathbf{CCG}(X_2,\omega_2(\cdot)+\langle\gamma_t \widehat{g}_2^t-\omega_2'(u_2^t),\cdot\rangle,\gamma_t F_{v_2};\epsilon_t)\end{array}$$

where $[\widehat{g}_1^t;\widehat{g}_2^t]=\frac{1}{m_t}\sum_{j=m_t+1}^{2m_t} g(\widehat{u}^t,\xi_j^t)$.
**end for**
**Output:** $\overline{x}_T:=[\bar{u}_T;\bar{v}_T]=\left(\sum_{t=1}^T\gamma_t\right)^{-1}\sum_{t=1}^T\gamma_t y^t$

The previous remark immediately leads to the following results

**Proposition 3.4.3.** *Suppose we are under assumptions* (***A***.1) − (***A***.4), (***S***.1) − (***S***.4) *with* $M=0$ *and with proximal setup on* $U_2$ *being Euclidean setup. Set stepsizes* $\gamma_t=L^{-1}, t=1,\dots,T$ *and batch size* $m_t=O(\gamma_t^2\sigma^2 T/\Theta[X])$ *for the outlined algorithm to return an stochastic* $\epsilon$*-solution to the VI*$(X,F)$ *represented by stochastic oracle satisfying the assumptions* (***C***.1) − (***C***.2)*, the total number of stochastic oracle calls required does not exceed*

$$\mathcal{N}_{SO}=O(1)\frac{\sigma^2\Theta[X]}{\epsilon^2}$$

*and the total number of calls to the Linear Minimization Oracle does not exceed*

$$\mathcal{N}_{LMO}=O(1)\frac{L^2D^2\Theta[X]}{\epsilon^2}$$

*where $\sigma^2, D, \Theta[X]$ are defined previously.*

*Proof.* Let us fix $T$ as the number of Mirror prox steps, and since $M = 0$, from Theorem 2.5.3, the efficiency estimate of the variational inequality implies that

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F)] \leq \frac{2\Theta[X] + 7\sum_{t=1}^{T}\gamma_t^2\frac{\sigma^2}{m_t} + 2\sum_{t=1}^{T}\epsilon_t}{\sum_{t=1}^{T}\gamma_t}, .$$

Let us fix $\epsilon_t = \frac{\Theta[X]}{T}$ for each $t = 1, \ldots, T$, then from Proposition 3.4.1, it takes at most $s = O(1)(\frac{L_0 D^\kappa T}{\Theta[X]})^{1/(\kappa-1)}$ calls to the LMO oracles to generate a point such that $\Delta_s \leq \epsilon_t$. Moreover, we have

$$\mathbf{E}[\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F)] \leq O(1)\frac{L\Theta[X]}{T}.$$

Therefore, to ensure $\mathbf{E}[\epsilon_{\mathrm{VI}}(\bar{x}_T|X, F)] \leq \epsilon$ for a given accuracy $\epsilon > 0$, the number of Mirror Prox steps $T$ is at most $O(\frac{L\Theta[X]}{\epsilon})$. Therefore, the number of stochastic oracle calls used is at most $\mathcal{N}_{\mathrm{SO}} = \sum_{t=1}^{T} m_t = O(\gamma_t^2 T^2\sigma^2/\Theta[X]) = O(1)\frac{\sigma^2\Theta[X]}{\epsilon^2}$. Moreover, the number of linear minimization oracle calls on $X_2$ needed is at most $\mathcal{N}_{\mathrm{LMO}} = sT = O(1)\Big(\frac{L_0 L^\kappa D^\kappa}{\epsilon^\kappa}\Big)^{1/(\kappa-1)}\Theta[X]$. In particular, if $\kappa = 2$ and $L_0 = 1$, this becomes $\mathcal{N} = O(1)\frac{L^2 D^2\Theta[X]}{\epsilon^2}$. □

**Discussion.** When solving problem (3.4.1), the above stochastic variant of Semi-Proximal Mirror-Prox algorithm enjoys the *optimal complexity bounds*, i.e. $O(1/\epsilon^2)$, both in terms of the number of calls to stochastic oracle (see [61]) and the number of calls to linear minimization oracle (see [48]). In the situation when nonsmooth $f$ in (3.4.1) does not admit saddle point representation, the above algorithms enjoys still optimal complexity bound $O(1/\epsilon^2)$ in terms of the number of calls to stochastic oracles, but that of the linear minimization oracle becomes $O(1/\epsilon^4)$. To the best of our knowledge, both results are novel.

### 3.4.4 Numerical Illustrations: Collaborative Filtering and Beyond

We present here illustrations of the proposed approach. We report the experimental results obtained with the proposed Semi-Proximal Mirror-Prox , denoted Semi-MP here, and compare them with the results obtained from state-of-the-art competing optimization algorithms. We consider three different models, all with a non-smooth loss function and a nuclear-norm regularization penalty: i) matrix completion with $\ell_2$ data fidelity term; ii)

robust collaborative filtering for movie recommendation; iii) link prediction for social network analysis. For i) & ii), we compared our results to those obtained with two competing approaches: a) Smoothing-CG; b) Semi-SPG, which will be discussed in details next. For iii), we compared our results to those obtained by Semi-LPADMM, using [68] and solving proximal mapping through conditional gradient routines.

**Matrix completion on synthetic data** We consider the matrix completion problem, with a nuclear-norm regularization penalty and an $\ell_2$ data-fidelity term. The model is given by

$$\min_x \|P_\Omega x - b\|_2 + \lambda \|x\|_{\text{nuc}}. \tag{3.4.20}$$

where $\|\cdot\|_{\text{nuc}}$ stands for the nuclear norm and $P_\Omega x$ is the restriction of $x$ onto the cells $\Omega$.

We compare the following three candidate algorithms, i) Semi-Proximal Mirror-Prox (**Semi-MP**) ; ii) conditional gradient with smoothing (**Smooth-CG**); iii) inexact accelerate proximal gradient after smoothing (**Semi-SPG**). We provide below the key steps of each algorithms.

1. **Semi-MP**: this is shorted for our Semi-Proximal Mirror-Prox algorithm, we solve the saddle point reformulation given by

$$\min_{x,v:\|x\|_{\text{nuc}} \le v} \max_{\|y\|_2 \le 1} \langle P_\Omega x - b, y\rangle + \lambda v \tag{3.4.21}$$

   which is equivalent as to the semi-structured variational inequality $\text{VI}(X, F)$ with $X = \{[u = (x; y); v] : \|x\|_{\text{nuc}} \le v, \|y\|_2 \le 1\}$ and $F = [F_u(u); F_v] = [P_\Omega^T y; b - P_\Omega x; \lambda]$. The subdomain $X_1 = \{y : \|y\|_2 \le 1\}$ is given by full-prox setup and the subdomain $X_2 = \{(x; v) : \|x\|_{\text{nuc}} \le v\}$ is given by LMO. By setting both the distance generating functions $\omega_x(x)$ and $\omega_y(y)$ to be the squared Euclidean distances,computing the $y$-component of an iterate reduces to a gradient step, and the update of $x$ follows the composite conditional gradient routine to a simple quadratic problem.

2. **Smooth-CG**: The algorithm ([70]) directly applies the generalized composite conditional gradient to the following smooth problem obtained by using the Nesterov

smoothing technique,

$$\min_{x,v:\|x\|_{\text{nuc}}\leq v} f^{\gamma}(x) + \lambda v, \text{ where } f^{\gamma}(x) = \max_{\|y\|_2\leq 1} \{\langle P_{\Omega}x - b, y\rangle - \frac{\gamma}{2}\|y\|_2^2\}. \tag{3.4.22}$$

In the full memory version, the update of $x$ at step $t$ requires solving re-optimization problem

$$\min_{\theta_1,\ldots,\theta_t} f^{\gamma}(\sum_{i=1}^{t} \theta_i u_i v_i^T) + \lambda \sum_{i=1}^{t} \theta_i \tag{3.4.23}$$

where $\{u_i, v_i\}_{i=1}^t$ are the singular vectors collected from the linear minimization oracles. Same as suggested in [70], we use the quasi-Newton solver L-BFGS-B [16] to solve the above re-optimization subproblem. Notice that in this situation, solving (3.4.23) can be relatively efficient even for large $t$ since computing the gradient of the objective in (3.4.23) does not necessarily require to compute the full matrix representation of $x = \sum_{i=1}^t \theta_i u_i v_i^T$.

3. **Semi-SPG**: The approach is to apply the accelerated proximal gradient to the smoothed composite model as in (3.4.22) and approximately compute the proximal mappings via conditional gradient routines. In fact, Semi-SPG can be considered as a direct extension of the conditional gradient sliding to the composite setting. Same as in Semi-MP, the update of $x$ is given by the composite conditional gradient routine as applied to a simple quadratic problem and additional interpolation step.

For Semi-MP and Semi-SPG, we test two different strategies for the inexact prox-mappings, a) fixed number of inner CG steps and b) decaying $\epsilon_t = c/t$ as the theory suggested. For the sake of simplicity, we generate the synthetic data such that the magnitudes of the constant factors (i.e. Frobenius norm and nuclear norm of optimal solution) are approximately of order 1, which means the accuracy is determined by the number of LMO calls. In Fig. 4, we evaluate the optimality gap of these algorithms with different parameters (e.g. number of inner steps, scaling factor c, smoothness parameter $\gamma$) and compare performances of the algorithms exhibited when the best-tuned parameters were used. As the plot shows, the Semi-MP algorithm generates a solution with $\epsilon = 10^{-3}$ accuracy within about 3000 LMO calls, which is not bad at all given the fact that the theoretical worst-case complexity is $O(1/\epsilon^2)$. Also, the plots indicate that using the strategy with $O(1/t)$

decaying inexactness provides better and more reliable performance than when fixed number of inner steps is used. Similar phenomena are observed for the Semi-SPG. One can see that these two algorithms based on inexact proximal mappings are notably faster than applying conditional gradient on the smoothed problem. Moreover, since the Smooth-CG requires additional computation and memory cost for the re-optimization procedure, the actual difference in terms of CPU time could be more significant.

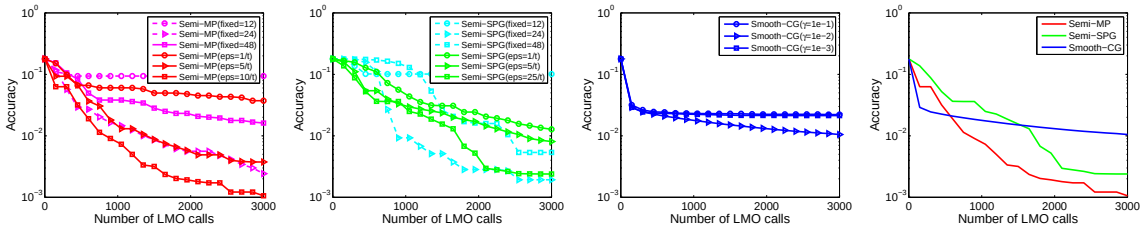



Figure 4: Matrix completion on synthetic data($1024 \times 1024$): optimality gap vs the LMO calls. From left to right: (a) Semi-MP; (b) Semi-SPG ; (c) Smooth-CG; (d) best of three.

**Robust collaborative filtering** We consider the collaborative filtering problem, with a nuclear-norm regularization penalty and an $\ell_1$-empirical risk function:

$$\min_x \frac{1}{|E|} \sum_{(i,j)\in E} |x_{ij} - b_{ij}| + \lambda \|x\|_{\text{nuc}}. \tag{3.4.24}$$

**Competing algorithms.** We compare the above three candidate algorithm. The smoothed problem for Semi-SPG and Smooth-CG in this case becomes

$$\min_{x,v:\|x\|_{\text{nuc}}\le v} f^\gamma(x) + \lambda v, \text{ where } f^\gamma(x) = \max_{\|y\|_\infty \le 1} \left\{ \frac{1}{|E|} \sum_{(i,j)\in E} (x_{ij} - b_{ij}) y_{ij} - \frac{\gamma}{2}\|y\|_2^2 \right\}. \tag{3.4.25}$$

Note that in this case, for Smooth-CG, solving the re-optimization problem in (3.4.23) at each iteration requires computing the full matrix representation for the gradient. For large $t$ and large-scale problems, the computation cost for re-optimization is no longer negligible. However, the Semi-MP and Semi-SPG do not suffer from this limitation since the conditional gradient routines are run on simple quadratic subproblems. For this particular example, we implement the Semi-MP slightly different from the above scheme. We solve the following

saddle point reformulation with properly selected $\rho$,

$$\min_{\substack{x,y,v_1,v_2:\\ v_1\geq\|x\|_{\text{nuc}},v_2\geq\|y\|_1}} \max_{\|w\|_2\leq 1} v_2 + \lambda v_1 + \rho\langle \mathcal{A}x - b - y, w\rangle \tag{3.4.26}$$

where we use $\mathcal{A}$ to denote the operator $\frac{1}{|E|}P_E$. The semi-structured variational inequality Semi-VI $(X, F)$ associated with the above saddle point problem is given by $X = \{[u = (x, y, w); v = (v_1.v_2)] : \|x\|_{\text{nuc}} \leq v_1, \|y\|_1 \leq v_2, \|w\|_2 \leq 1\}$ and $F = [F_u(u); F_v] = [\rho\mathcal{A}w; -\rho w; \rho(y - \mathcal{A}x + b); \lambda; 1]$. The subdomain $X_1 = \{(y, w, v_2) : \|y\|_1 \leq v_2, \|w\|_2 \leq 1\}$ is given by full-prox setup and the subdomain $X_2 = \{(x; v_1) : \|x\|_{\text{nuc}} \leq v_1\}$ is given by LMO. By setting both the distance generating functions to be the squared Euclidean distance,updating of the $w$-component of a iterate reduces to the gradient step, updating of the $y$-component reduces to the soft-thresholding operator, and updating of the $x$-component is given by the composite conditional gradient routine. Note that the Semi-Proximal Mirror-Prox algorithm (Semi-MP) does not require tuning of any parameter.

We run the above three algorithms on the the small and medium MovieLens datasets. The small-size dataset consists of 943 users and 1682 movies with about 100K ratings,while the medium-size dataset consists of 3952 users and 6040 movies with about 1M ratings. We follow [70] to set the regularization parameters. We randomly pick 80% of the entries to build the training dataset, and compute the normalized mean absolute error (NMAE) on the remaining test dataset. For Smooth-CG, we carry out the algorithm with different smoothing parameters, ranging in $\{1e-3, 1e-2, 1e-1, 1e0\}$ and select the one with the best performance. For the Semi-SPG algorithm, we adopt the best smoothing parameter found when running Smooth-CG. We use two different strategies to control the number of LMO calls at each iteration, i.e. the accuracy of the proximal mapping for both Semi-SPG and Semi-MP, which are a) fixed number of inner CG steps and b) decaying $\epsilon_t = c/t$ as the theory suggests. We display on Fig. 5 and Fig. 6 the performance of each algorithm under different choice of parameters and the overall comparison of objective value and NMAE on test data in Fig. 7.

In Fig. 5 and Fig. 6, we can see that using fixed inner CG steps sometimes achieves

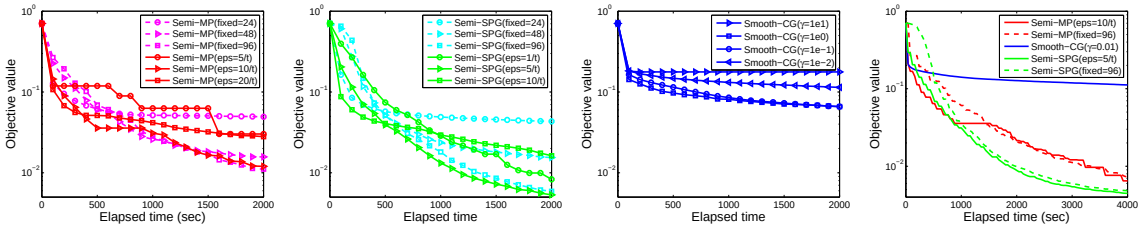



Figure 5: Robust collaborative filtering on MovieLens 100K: objective function vs elapsed time. From left to right: (a) Semi-MP; (b) Semi-SPG ; (c) Smooth-CG; (d) best of three.

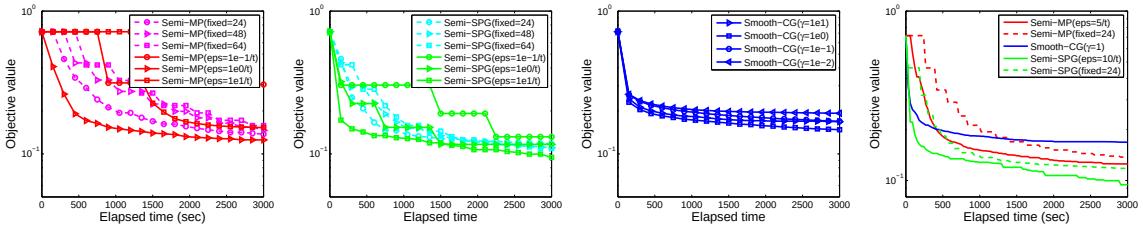



Figure 6: Robust collaborative filtering on MovieLens 1M: objective function vs elasped time. From left to right: (a) Semi-MP; (b) Semi-SPG ; (c) Smooth-CG; (d) best of three.

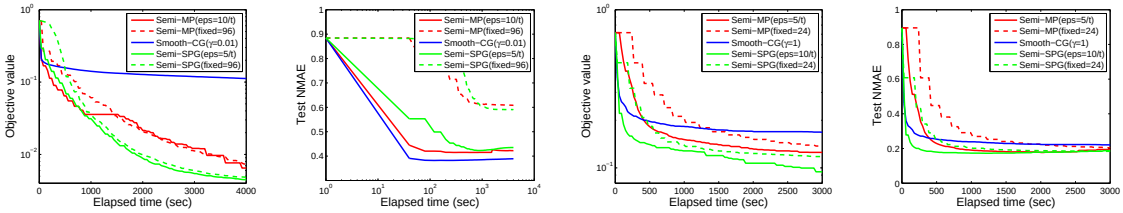



Figure 7: Robust collaborative filtering on Movie Lens: objective function and test NMAE against elapsed time. From left to right: (a) MovieLens 100K objective; (b) MovieLens 100K test NMAE; (c) MovieLens 1M objective; (d) MovieLens 1M test NMAE.

performance comparable to the one with decaying epsilon $\epsilon_t$. In Fig. 7, we can see that Semi-MP clearly outperforms Smooth-CG, and is competitive with Semi-SPG. In the large-scale setting, Semi-MP achieves better objective values as well as better test NMAE compared to Smooth-CG.

**Link prediction** We consider the following model for the link prediction problem,

$$\min_{x \in \mathbf{R}^{m \times n}} \frac{1}{|E|} \sum_{(i,j) \in E} \max\left(1 - (b_{ij} - 0.5)x_{ij}, 0\right) + \lambda_1 \|x\|_1 + \lambda_2 \|x\|_{\text{nuc}} \tag{3.4.27}$$

This example is more complicated than the previous two examples since it has not only one nonsmooth loss function but also two regularization terms. Applying the smoothing-CG or Semi-SPG would require to build two smooth approximations, one for hinge loss term and

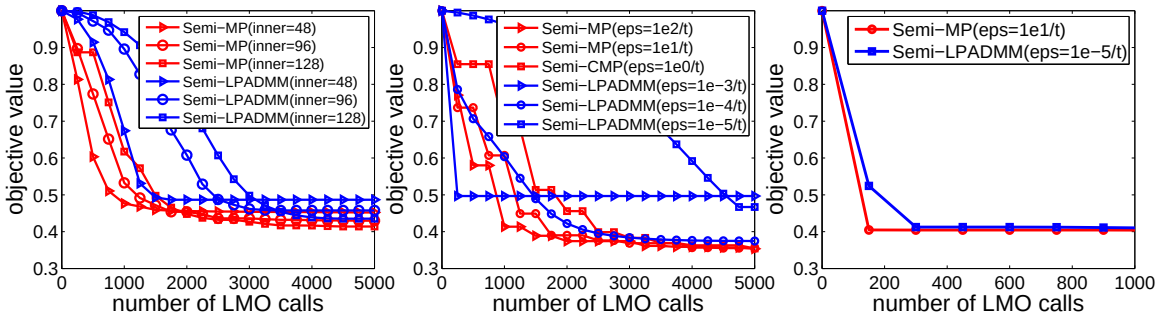



Figure 8: Link prediction on Wikivote: objective function value against the LMO calls. From left to right: (a)Wikivote(1024) with fixed inner steps; (b) Wikivote(1024) with decaying error; (c) Wikivote(full)

one for $\ell_1$ norm term. Therefore, we consider another alternative approach, Semi-LPADMM, where we apply the linearized preconditioned ADMM algorithm while computing proximal mapping through conditional gradient routines. Up to our knowledge, ADMM with early stopping is not well-analyzed in literature, but intuitively as long as the accumulated error is controlled sufficiently, the procedure converges.

We conduct experiments on a binary social graph data set called Wikivote, which consists of 7118 nodes and 103,747 edges. Since the computation cost of these two algorithms mainly come from the LMO calls, we describe in what follows the performance in terms of number of LMO calls. For the first set of experiments, we select top 1024 highest degree users from Wikivote and run the two algorithms on this small dataset with different strategies for the inner LMO calls.

In Fig. 8, we observe that the Semi-MP as compared to ADMM is less sensitive to the inaccuracies in computing prox-mappings. ADMM sometimes just stops to progress unless the prox mapping at early iterations is computed with sufficient accuracy. Another observation is that in this example,strategy with decaying $\epsilon_t$, works better in the long run than when using fixed number of inner LMOs calls. The results on the full dataset again indicate that our algorithm performs better than the semi-proximal variant of the ADMM algorithm.

### 3.4.5 Concluding Remarks

In this section, we propose a new conditional gradient type of algorithm to solve high-dimensional non-smooth composite minimization problems. The proposed Semi-Proximal Mirror-Prox, leverages the saddle point representation of one component of the objective while handling the other component via linear minimization over the problem's domain. The algorithm differs essentially from the usual proximal gradient algorithms with smoothing, which require computing precise proximal operators at each iteration and can therefore be impractical for high-dimensional problems with difficult geometry. We establish the theoretical convergence rate of Semi-Proximal Mirror-Prox, which exhibits the optimal complexity bounds, i.e. $O(1/\epsilon^2)$, for the number of calls to linear minimization oracle needed to get an $\epsilon$-solution. We present promising experimental results showing the the potential of our approach as compared to competing methods.

## *3.5 Application IV: Maximum Likelihood Based Poisson Imaging*

### 3.5.1 Problem of Interest

In a variety of applications, finding the maximum likelihood estimate in a statistical model often leads to a convex optimization problem of the following form,

$$\min_{x\in\mathbf{R}^n} L(x) + h(x) := \frac{1}{m}\sum_{i=1}^{m} \ell_i(x) + h(x) \tag{3.5.1}$$

where $L(x)$ comes from the log-likelihood, $h(x)$ is some regularization, $m$ is the number of observations. Problems of this type also arise ubiquitously in machine learning, known as empirical risk minimization, where $\ell_i(\cdot)$ corresponds to a data-driven loss function. The most typical example is the least-squares regression, where $\ell_i$ refers to the square loss, i.e. $\ell_i(x) = \frac{1}{2}(a_i^T x - b_i)^2$, given the observations $(a_i, b_i), i = 1, 2, \ldots, m$. This is widely used when the linear measurements are contaminated with Gaussian noise. In contrast, in the presence of Poisson noise, i.e.

$$b_i \sim \text{Poisson}(a_i^T x) \tag{3.5.2}$$

the loss function that forms the empirical risk minimization becomes

$$\ell_i(x) = a_i^T x - b_i \log(a_i^T x), \tag{3.5.3}$$

which will be referred to as Poisson loss. Such type of problems arise in many applications involving Poisson process or more general point process.

The most typical and well studied example is the positron emission tomography (PET) in nuclear medicine, where the event detected is triggered by the photon counts following a Poisson distribution [9, 37, 77]. The Poisson noise setting has also been considered in many other contexts, such as solar flare image reconstruction [18] and confocal microscopy image deblurring [19]. Depending on the specific applications, various choices of regularization terms can be utilized to enforce sparsity, low rank structures or smoothness. In literatures, this type of problem is sometimes called Poison compressive sensing when the true parameter is compressible.

**Background.** While there has been tremendous work on efficient first-order methods for solving the penalized least squares problem under Gaussian noise, ranging from proximal gradient methods to incremental algorithms, fewer results are known for the Poisson loss minimization. The key challenge, from a pure algorithmic point of view, lies in the fact that Poisson loss function is non-globally Lipschitz continuous/differentiable. It is well-known that when solving convex optimization problems with $L$-Lipschitz smooth objective functions, the best convergence rate of first-order method is $O(\frac{L}{t^2})$; this can be achieved by algorithms such as Nesterov's optimal gradient [64]. However, for Poisson loss minimization, there is no global Lipschitz continuity for the objective function or the gradient. Existing methods that rely on such conditions will no longer be applicable.

**Related work.** In [37], the authors biased the logarithmic term by replacing $\log(a_i^T x)$ with $\log(a_i^T x + \epsilon)$, where $\epsilon$ is a tolerance parameter of magnitude $10^{-10}$, which results a smooth problem with $L$ of order $O(1/\epsilon^2)$. One can immediately see that, this huge Lipschitz constant could significantly affect the efficiency estimate. On the contrary, in [9], the authors treated this problem as a general nonsmooth optimization and applied Mirror Descent algorithm, which avoids the dependence on Lipschitz continuity of the gradient, but in the sacrifice of having a worse rate of convergence, i.e. $O(\frac{1}{\sqrt{t}})$. Another approach

was explored in [78], where the authors consider a general setting, so-called composite self-concordant minimization, allowing to cover the Poisson loss minimization problem. They exploited the self-concordance nature of the logarithmic term and proposed a proximal gradient method with sophisticated stepsize choices and correction procedures, providing a locally linear rate of convergence as well as a $O(\frac{1}{t})$ global rate which still depends on global Lipschitz continuity constant.

**Our goal and main contribution.** Our goal in this section is to revisit the non-Lipschitz Poisson loss minimization problem with our newly developed algorithmic tools, aiming to build algorithms with reasonable computational behavior in the large-scale case. To this end, we exploit a saddle point representation of the non-Lipschitz objective, which allows us to apply the composite Mirror Prox algorithm for free. The algorithm proposed here, is free of Lipschitz continuity conditions and serves as an novel approach to address this type of non-Lipschitz optimization. As we demonstrate in the sequel, the algorithm enjoys a $O(1/t)$ convergence rate in theory and also exhibits promising performances in practice.

**Outline** The rest of this section is organized as follows. In Section 3.5.2, we reformulate the problem of interest as a composite convex-concave saddle point problem, propose a composite Mirror Prox algorithm tailored for this problem, and discuss the complexity results. In Section 3.5.3, we provide numerical illustrations of the algorithm when applied to the positron emission tomography (PET) recovery.

### 3.5.2 Saddle Point Reformulations and Complexity Analysis

**Problem restatement** We will consider the following problem with a more compact and slightly more general form:

$$\min_{x\in\mathbf{R}^n_+} L(x)+h(x),\ \text{with } L(x)=s^Tx-\sum_{i=1}^m c_i\ln(a_i^Tx) \tag{3.5.4}$$

where nonnegative coefficients $s\in\mathbf{R}^n_+, c\in\mathbf{R}^m_+$ and $a_i\in\mathbf{R}^n_+, i=1,\dots m$ are given. [11]

Throughout this section, we will assume that the regularization term $h(x)$ satisfies:

[11] To match with the loss function in equation (3.5.3), one can simply set $s=\frac{1}{m}\sum_{i=1}^m a_i$ and $c_i=b_i/m$.

- (*homogeneity*) $h(ax) = |a|h(x)$ for any $a \in \mathbf{R}$;
- (*proximal-friendliness*) proximal mapping of the following form is easy to compute,

$$\min_{x \in \mathbf{R}^n_+} \{\omega(x) + \langle \xi, x \rangle + h(x)\},$$

for some distance generating function $\omega(x) : \mathbf{R}^n_+ \to \mathbf{R}$ that is Lipschitz continuous and 1-strongly convex w.r.t. some norm $\|\cdot\|$ defined on $\mathbf{R}^n$.

Note that the above assumptions hold true for many sparsity-promoting penalty functions, e.g. $h(x) = \|x\|_1$.

**Saddle point reformulation** The crux of our method is to utilize the Fenchel representation of log function

$$\log(u) = \min_{v>0} \{uv - \log(v) - 1\}.$$

We can rewrite (3.5.4) as

$$\min_{x \in \mathbf{R}^n_+} \max_{v \in \mathbf{R}^m_{++}} s^T x + \sum_{i=1}^m [c_i \ln(v_i) - c_i v_i a_i^T x + c_i] + h(x).$$

Setting $y_i = c_i v_i$, this can be further simplified to

$$\min_{x \in \mathbf{R}^n_+} \max_{y \in \mathbf{R}^m_{++}} \Phi(x, y) := s^T x - y^T A x + \sum_{i=1}^m c_i \ln(y_i) + h(x) + c_0 \tag{3.5.5}$$

where $c_0 = \sum_{i=1}^m c_i - \sum_{i=1}^m c_i \ln(c_i)$ and $A = [a_1^T; a_2^T; \ldots; a_m^T] \in \mathbf{R}^{m \times n}_+$.

**Composite Mirror Prox** Observe that the above model can be regarded as a composite saddle point problem with two separable penalty functions: $p(y) = \sum_{j=1}^m c_i \ln(y_i)$ for variable $y$ and $h(x)$ for variable $x$. Recall that in Section 2.5.1 from Chapter II, we have developed a composite Mirror Prox algorithm which can solve such problems. To this end, we act as follows. We first move the penalty functions in the domain and reformulate the problem as a bilinear saddle point problem

$$\min_{x \in \mathbf{R}^n_+} \max_{y \in \mathbf{R}^m_{++}} \{s^T x - y^T A x + \tau + \sigma + c_0 : \sigma \geq h(x), \tau \leq \sum_{i=1}^m c_i \ln(y_i)\} \tag{3.5.6}$$

associated with monotone operator $F = [F_u(u = [x; y]); F_v(v = [\sigma; \tau])]$,

$$F_u(u) = [s - A^T y; Ax] \text{ and } F_v = [1; -1],$$

and domain

$$W = \{(u, v) : x \in \mathbf{R}^n_+, \sigma \geq h(x), y \in \mathbf{R}^m_{++}, \tau \leq \sum_{i=1}^m c_i \ln(y_i)\}.$$

We can equip the projection $U = \{u = [x, y] : x \in \mathbf{R}^n_+, y \in \mathbf{R}^m_{++}\}$ with the mixed setup

$$\omega(u) = \alpha\omega(x) + \frac{1}{2}\|y\|_2^2, \quad \|u\| = \sqrt{\alpha\|x\|_x^2 + \|y\|_2^2}$$

for some positive number $\alpha > 0$. The reason why we choose the Euclidean setup for variable $y$ stems from the following fact.

**Lemma 3.5.1.** *For any $\eta \in \mathbf{R}^m$ and $\beta > 0$, let*

$$y^+ = \operatorname*{argmin}_{y \in \mathbf{R}^m_{++}} \left\{ \frac{1}{2}\|y\|_2^2 + \langle \eta, y \rangle - \beta \sum_{i=1}^m c_i \ln(y_i) \right\},$$

*then $y^+$ is explicitly given by*

$$y_i^+ = \frac{-\eta_i + \sqrt{\eta_i^2 + 4\beta c_i}}{2}, \forall i = 1, 2, \ldots, m.$$

Before presenting the algorithm, let us introduce the composite proximal operator induced by a convex function $g$ and proximal setup $(\omega(x), \|\cdot\|)$,

$$\begin{aligned} \text{Prox}^{\omega}_{g,x_0}(\xi) &= \text{argmin}_{x \in \mathbf{R}^n_+} \{\omega(x) + \langle \xi - \omega'(x_0), x \rangle + g(x)\} \\ &= \text{argmin}_{x \in \mathbf{R}^n_+} \{V(x, x_0) + \langle \xi, x \rangle + g(x)\} \end{aligned}$$

where $V(x, x_0) := \omega(x) - \omega(x_0) - \nabla\omega(x_0)^T(x - x_0)$, is usually known as the Bregman distance. We present below in Algorithm 8 the composite Mirror Prox algorithm specifically tailored to our problem of interest as described in (3.5.4), (3.5.5), and (3.5.6).

Given any subset $X \subset \mathbf{R}^n_+$, let $Y[X] := \{y : y_i = 1/(a_i^T x), i = 1, \ldots, m, x \in X\}$. Clearly, $Y[X] \in \mathbf{R}^m_{++}$. Invoking Theorem 2.5.1 and Corollary 2.5.1, we arrive at the following results:

**Algorithm 8** Composite Mirror Prox Algorithm for Poisson Loss Minimization

0. Initialize $x^1 \in \mathbf{R}^n_+$, $y^1 \in \mathbf{R}^n_{++}$, $\alpha > 0$ and $\gamma_t > 0$,
**for** $t = 1, 2, \ldots, T$ **do**
1. Compute

$$\begin{aligned} \widehat{x}^t &= \mathrm{Prox}^{\alpha\omega}_{\gamma_t h, x^t}\big(\gamma_t(s - A^T y^t)\big) \\ \widehat{y}^t_i &= \tfrac{1}{2}\left(-\gamma_t(a_i^T x^t - y_i^t) + \sqrt{\gamma_t^2(a_i^T x^t - y_i^t)^2 + 4\gamma_t c_i}\right), i = 1, \ldots, m \end{aligned}$$

2. Compute

$$\begin{aligned} x^{t+1} &= \mathrm{Prox}^{\alpha\omega}_{\gamma_t h, x^t}\big(\gamma_t(s - A^T \widehat{y}^t)\big) \\ y^{t+1}_i &= \tfrac{1}{2}\left(-\gamma_t(a_i^T \widehat{x}^t - y_i^t) + \sqrt{\gamma_t^2(a_i^T \widehat{x}^t - y_i^t)^2 + 4\gamma_t c_i}\right), i = 1, \ldots, m \end{aligned}$$

**end for**
Output $x_T = \frac{1}{T}\sum_{t=1}^T \lambda_t x^t$

**Proposition 3.5.1.** *Assume we are given some information on the optimal solution to problem in (3.5.4): a convex compact set $X_0 \subset \mathbf{R}^n_+$ containing $x_*$ and a convex compact set $Y_0 \subset \mathbf{R}^m_{++}$ containing $Y[X_0]$. Let*

$$\mathcal{L} = \|A\|_{x\to 2} := \max_{x \in \mathbf{R}^n_+ : \|x\| \le 1} \{\|Ax\|_2\}$$

*and let stepsizes in Algorithm 8 satisfy $0 < \gamma_t \le \sqrt{\alpha}\mathcal{L}^{-1}$ for all $t > 0$. Then*

$$L(x_T) + h(x_T) - [L(x_*) + h(x_*)] \le \left[\sum\nolimits_{\tau=1}^t \gamma_\tau\right]^{-1} (\alpha\Theta[X_0] + \Theta[Y_0]), \tag{3.5.7}$$

*where $\Theta[X_0] = \max_{x\in X_0} V(x, x^1)$ and $\Theta[Y_0] = \max_{y\in Y_0} \frac{1}{2}\|y - y^1\|_2^2$. In particular, by setting $\gamma_t = \sqrt{\alpha}\mathcal{L}^{-1}$ for all $t$, one has*

$$L(x_T) + h(x_T) - [L(x_*) + h(x_*)] \le \frac{(\alpha\Theta[X_0] + \Theta[Y_0])\|A\|_{x\to 2}}{\sqrt{\alpha}T}. \tag{3.5.8}$$

**Remark I.** Note that the above algorithm works without requiring global Lipschitz continuity of the original objective function. The information set $X_0$ and $Y_0$ appear only in the efficiency estimate, but not in the algorithm itself. Nevertheless, it is not hard to obtain such information set.[12] In principle, one have at least

$$X_0 = \{x \in \mathbf{R}^n_+ : s^T x + h(x) \le \sum_{i=1}^m c_i\}.$$

[12]Knowing the geometry of such set could also help us determine favorable proximal setups.

Clearly, $X_0$ is convex and compact. The reason why $x_* \in X_0$ is due to the following observation.

**Proposition 3.5.2.** *The optimal solution $x_*$ to the problem in (3.5.4) satisfies*

$$s^T x_* + h(x_*) = \sum_{i=1}^{m} c_i. \tag{3.5.9}$$

*Proof.* This is because, for any $t > 0$, $tx_*$ is a feasible solution and the objective at this point is

$$\phi(t) := L(tx_*) + h(tx_*) = t(s^T x_* + h(x_*)) - \sum_{i=1}^{m} c_i \ln(a_i^T x_*) - \ln(t) \sum_{i=1}^{m} c_i.$$

By optimality, $\phi'(1) = 0$, i.e. $s^T x_* + h(x_*) - \sum_{i=1}^{m} c_i = 0$. □

**Remark II.** From the above proposition, one can see that the performance of the algorithm is essentially determined by the distance between the initial solution $(x^1, y^1)$ to the optimal solution $(x_*, y_*)$. In principle, if the initial solution is close enough to the optima, then one can expect the algorithm to converge very fast. In practice, the optimal choice of $\alpha = \frac{\|y^1 - y_*\|_2^2/2}{V(x^1, x_*)}$ is often unknown, one can perhaps select $\alpha$ experimentally in order to get best performance.

### 3.5.3 Numerical Illustration: Poisson Emission Tomography

**Physical background** Positron-emission tomography (PET) is a nuclear medicine, functional imaging technique that produces images, often in three dimensions, of chemical functioning and metabolic activity of internal tissues in the human body. It is heavily used for clinical diagnosis of cancer metastasis, brain and heart function. PET imaging works as follows: i) inserting radiotracer (positron-emitting radionuclides) which is tagged to a natural chemical and is transported to the organ of interest on a biologically active molecule, ii) detecting pairs of flying at opposite directions gamma quants which are emitted when a positron emitted in an act of tracer's disintegration annihilates with nearby electron, iii) reconstructing the image of tracer concentration, i.e. spatial distribution of the radioactivity within the organ, based on photon counts – numbers of pairs of gamma-quants registered

during the study by different pairs of detectors. A natural assumption for such radioactive phenomenon is that these gamma-ray photons can generated by some Poisson process.

**Poisson Maximum Likelihood** At an abstract level, let us denote by $w_1, \ldots, w_m$ as the photon counts registered by $i$-th pair of detectors. The aim of the image reconstruction in PET is to estimate the density of the tracer in the emitting object. To simplify the model, we discretize the problem and split the object into $n$ voxels (pixels in 2-D case). Denoting by $a_{ij}$ the probability that the pair of gamma-quants originating from voxel $j$ will be registered by pair of detectors $i$, we get an $m \times n$ matrix $A = [a_{ij}]$. Denoting by $x \in \mathbf{R}^n$ the vector comprised of the amounts $x_j$ of tracer in cells $j = 1, ..., n$, the measurements $w_i$ are independent across i realizations of Poisson random variables $w_i$ with parameters $[Ax]_i$:

$$w_i \sim \text{Poisson}([Ax]_i), 1 \leq i \leq m,$$

Note that the column sums in $A$ do not exceed 1; these sums are equal to 1 (i.e., $A$ is stochastic) when every pair of emitted $\gamma$-quants is registered; whether it is the case, depends on scanner's construction. For the sake of simplicity, in the sequel we assume that $A$ indeed is stochastic; extensions to the case when the column sums in $A$ are less than one are straightforward.

Maximizing the likelihood function reduces to solving the convex optimization problem

$$\text{Opt} = \min_{x \in \mathbf{R}^n_+} \sum_{i=1}^m \left[ [Ax]_i - w_i \ln([Ax]_i) \right]. \tag{3.5.10}$$

Apparently, this falls into the Poisson loss minimization described in (3.5.4). For simplicity, we will not consider penalty or regularization terms in the following.

**Saddle Point Reformulation** Invoking the optimality conditions for the above problem, we have

$$x_j \sum_{i=1}^m \left[ a_{ij} - w_i \frac{a_{ij}}{[Ax]_i} \right] = 0, \forall j = 1, \ldots, n,$$

whence, summing over $j$ and taking into account that $A$ is stochastic, we get[13]

$$\sum_{j=1}^n x_j = \sum_{i=1}^m w_i =: \theta.$$

[13]Note that this is essentially a special case revealed by Remark I in the previous section.

We loose nothing by adding to problem (3.5.10) the equality constraints $\sum_{j=1}^{n} x_j = \theta$. Invoking the saddle point reformulation in the previous section, solving the PET recovery problem (3.5.10) is equivalent to solving the convex-concave saddle point problem:

$$\min_{\substack{x\in\mathbf{R}^n_+ \\ \sum_{j=1}^n x_j=\theta}} \max_{y\in\mathbf{R}^m_{++}} \Phi(x,y) := -y^T Ax + \sum_{i=1}^{m} w_i \ln(y_i) + \widetilde{\theta} \tag{3.5.11}$$

where $\widetilde{\theta} = 2\theta - \sum_{i=1}^{m} \omega_i \ln(\omega_i)$ is a constant.

**Composite Mirror Prox algorithm for PET** Noting that the domain over $x$ is a simplex, a good choice of $\omega(x)$ is the entropy, . We present in the following the composite Mirror Prox algorithm specifically tailored to the saddle point reformulation of the PET problem as described in (3.5.11).

**Algorithm 9** Composite Mirror Prox Algorithm for PET Reconstruction

0. Initialize $x^1 \in \mathbf{R}^n_+$, $y^1 \in \mathbf{R}^n_{++}$, $\alpha > 0$ and $\gamma_t > 0$,
**for** $t = 1, 2, \ldots, T$ **do**
1. Compute

$$\begin{aligned} \widehat{x}^t_j &= x^t_j \exp(-[A^T y]_j/\alpha), j = 1, \ldots, n, \text{ then normalized to sum up to } \theta \\ \widehat{y}^t_i &= \tfrac{1}{2}\left(-\gamma_t(a_i^T x^t - y^t_i) + \sqrt{\gamma_t^2(a_i^T x^t - y^t_i)^2 + 4\gamma_t w_i}\right), i = 1, \ldots, m \end{aligned}$$

2. Compute

$$\begin{aligned} x^{t+1} &= x^t_j \exp(-[A^T \widehat{y}]_j/\alpha), j = 1, \ldots, n, \text{ then normalized to sum up to } \theta \\ y^{t+1}_i &= \tfrac{1}{2}\left(-\gamma_t(a_i^T \widehat{x}^t - y^t_i) + \sqrt{\gamma_t^2(a_i^T \widehat{x}^t - y^t_i)^2 + 4\gamma_t w_i}\right), i = 1, \ldots, m \end{aligned}$$

**end for**
Output $x_T = \frac{1}{T}\sum_{t=1}^{T} \lambda_t x^t$

**Remark.** Let $x_*$ be the true image. Note that when there is no Poisson noise, $w_i = [Ax_*]_i$ for all $i$. In this case, the optimal solution $y_*$ corresponding to the $y$-component of the saddle point problem (3.5.11) is given by $y_{*,i} = w_i/[Ax_*]_i = 1, \forall i$. Thus, we may hope that under the Poisson noise, the optimal $y_*$ is still close to 1. Assuming that this is the case, the efficiency estimate for $T$-step composite Mirror Prox algorithm in Algorithm 9 after invoking Proposition 3.5.1 and setting $\alpha = r^2 m$ for some $r > 0$, will be

$$O(1)\left(\ln(n) + \frac{1}{2r^2}\right)\frac{r\theta\sqrt{m}\|A\|_{1\to 2}}{T}.$$

Since $A$ is $m \times n$ stochastic matrix, we may hope that the Euclidean norms of columns in $A$ are of order $O(m^{1/2})$, yielding the efficiency estimate

$$O(1)\left(\ln(n) + \frac{1}{2r^2}\right)\frac{r\theta}{T}.$$

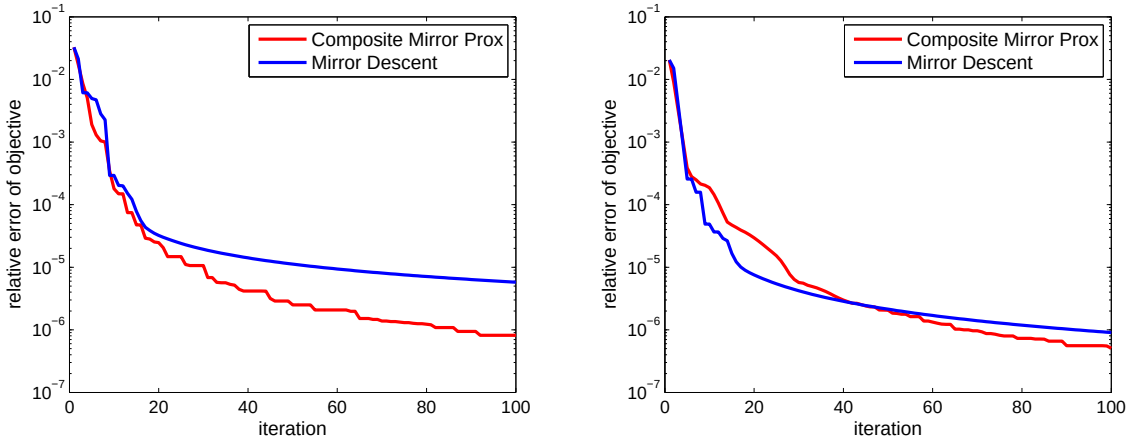
Let us look what happens in this model when $x_*$ is "uniform", i.e. all entries in $x_*$ are $\theta/n$. In this case, the optimal value is $\theta - \theta\ln(\theta) + \theta\ln(n)$, which is typically of order $O(\theta)$, implying that relative to optimal value rate of convergence is about $O(1/T)$.

**Numerical Results.** We ran experiments on several phantom images of size $256 \times 256$. We built the matrix $A$, which is of dimension $43530 \times 65536$. We first consider the noiseless situation, i.e. $w = Ax_*$, where $x_*$ refers to the true image; hence, the optimal solution and objective value are known. To demonstrate the efficiency of our algorithm, we compare our algorithm to the Mirror Descent algorithm in [9]. For both algorithms, we use the $\ell_1$ setup for the domain $X = \{x \in \mathbf{R}^n_+ : \sum_{j=1}^n x_j = \theta\}$ by setting the distance generating function to $\omega(x) = \sum_{j=1}^n x_j \ln(x_j)$.

Since the iteration cost of the two algorithms are about the same, we compare in Fig.9 their relative accuracy, i.e. $(f(x_t) - f_*)/f_*$, within the same number of iterations when applied to the Shepp-Logan phantom and the MRI brain phantom. We can see that the accuracy of composite Mirror Prox exceeds that of Mirror Descent after certain number of iterations. In Fig.10, we provide the mid-slices of our reconstructions for the MRI brain image. The experiments clearly demonstrate that our composite Mirror Prox serves as a viable alternative when solving the PET reconstruction problem and eventually produces solutions with higher accuracy compared to Mirror Descent.

### 3.5.4 Concluding Remarks

In this section, we investigate the Poisson loss minimization problem, which has been a long-standing challenge in machine learning community due to the non-Lipschitz continuity of Poisson loss. We exploit the underlying saddle point representation of the problem, allowing us to process the problem directly with the composite Mirror Prox algorithm, which no longer relies on Lipschitz continuity of the loss function. The algorithm enjoys




(a) Shepp-Logan image (b) MRI brain image

Figure 9: Convergence comparison between composite Mirror Prox and Mirror Descent.

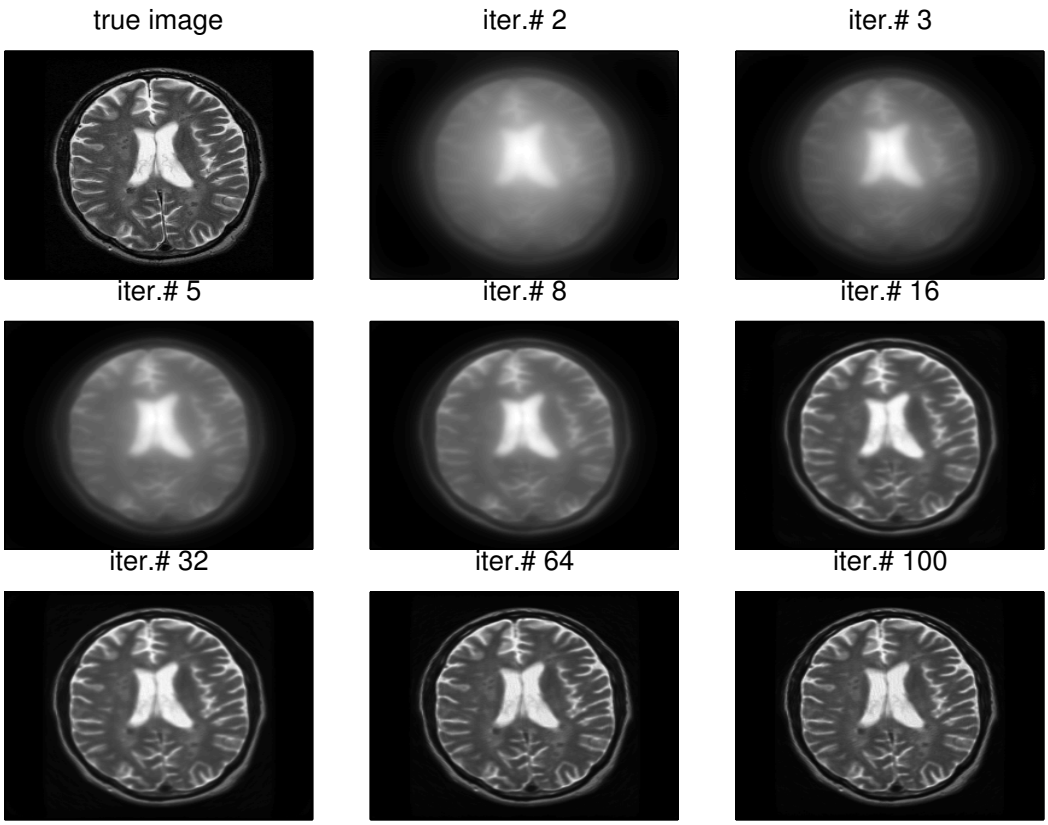



Figure 10: Performance of composite Mirror Prox on the MRI brain image

a $O(1/t)$ convergence rate in contrast to the usual $O(1/\sqrt{t})$ rate when solving nonsmooth minimization. We also demonstrate experimentally, albeit at this point in time just in a couple of experiments, the efficiency of the proposed algorithm as applied to Poison Emission Tomography reconstruction.

# CHAPTER IV

# ERROR-IN-MEASUREMENT OPTIMIZATION

## *4.1 Overview*

Our goal in this chapter is to examine the class of optimization problems with data subject to measurement errors, specifically, problems of the form

$$\min_{x\in X} \Phi(x, \pi^*) \qquad (\star)$$

where $\Phi(x, \pi)$ is a given function of the decision vector $x$ and vector of parameters $\pi$; we assume that this function is convex in $x$. In our setting, the "true value" $\pi^*$ of the parameter vector is unknown, but can be somehow "measured." Specifically, we assume that $\pi^*$ belongs to a given in advance set $\Pi$; on the top of this knowledge, we can learn $\pi^*$, by observing samples $\omega_t$, $t = 1, 2, ...$, drawn independently of each other from some distribution $P$. In the sequel, we consider two models of this type:

- *Direct Noisy Observations:* $\pi^*$ is the expectation of $P$. This case will be considered in Section 4.2.
- *Indirect Noisy Observations:* we are given in advance a parametric family $\{P_\pi : \pi \in \Pi$, of distributions with the domain $\Pi$ known to contain $\pi^*$, and the samples we observe are drawn from the distribution $P = P_{\pi^*}$. This situation will be considered in Section 4.3.

## *4.2 Convex Optimization with Direct Noisy Observations*

### 4.2.1 Error-in-measurement Optimization

Our goal is to solve systems of constraints

$$\text{Find } u \in U: \quad F_i(u, \xi_*) \le 0, 1 \le i \le I, \qquad (4.2.1)$$

where functions $F_i(u, \xi) : U \times \Xi \to \mathbf{R}$, with convex and compact $U$ and convex $\Xi$, are convex in $u \in U$ and concave in $\xi \in \Xi$, and the true vector of parameters $\xi_*$ is the expectation of

some distribution $P$; this distribution is not known in advance, but we can observe samples drawn, independently of each other, from this distribution.

**The situation** we are interested in throughout this section is when each $F_i$ is given by a *saddle point representation*. The latter notion is defined as follows. Let $f(u,\xi) : U \times \Xi \to \mathbf{R}$ be a convex-concave function. Let $\phi(u,v;y) : (U \times V) \times Y \to \mathbf{R}$ be a function that is convex in $y$ and concave in $(u,v)$, where $Y, V$ are convex sets, such that

$$f(u,\xi) = \min_{v \in V} \max_{y \in Y} \left[ a^T\xi + u^T A\xi + v^T B\xi - \phi(u,v;y) \right]. \tag{4.2.2}$$

We refer the representation of the form (4.2.2) to a *saddle-point representation* of convex-concave function $f(u,\xi)$. It is easily seen that the right hand side in (4.2.2) indeed is convex in $u$ and concave in $\xi$. Indeed, note that the right hand side can be written as $f(u,\xi) = \min_{v \in V} \left[ a^T\xi + u^T A\xi + v^T B\xi + \max_{y \in Y}[-\phi(u,v;y)] \right]$, that is, as the infimum in $v \in V$ of a convex in $(u,v)$ function depending on $\xi$ as on a parameter, so the right hand side in (4.2.2) is indeed convex in $u$. From the same representation we see that the right hand side in (4.2.2) as a function of $\xi$ is the infimum of a family of affine functions of $\xi$, and as such is concave in $\xi$.

At a first glance, convex-concave functions allowing for explicit saddle point representations seem to be a "rare commodity"; we shall see, however, that these representations admit a kind of "fully algorithmic" calculus, and that as a result, availability of such a representation is more of a rule than an exception. It should be stated that existence of saddle point representations of convex-concave functions satisfying minor regularity assumptions was established by Rockafellar in [75].

**Example 4.2.1.** *Function*

$$f(u;\Sigma) = \sqrt{u^T\Sigma u} : \mathbf{R}^n \times \mathcal{S}^n_+ \to \mathbf{R}_+$$

*where $\mathcal{S}^n_+$ is the cone of positive semidefinite symmetric $n \times n$ matrices, is a convex-concave function, that is convex in $u$ and concave in $\Sigma$, and admits a saddle-point representation:*

$$f(u,\Sigma) = \min_{S \in \mathcal{S}^n_+} \max_{y \in \mathbf{R}^n} \left[ y^T x + \langle \Sigma, S \rangle - y^T S y \right],$$

*where the inner product* $\langle \Sigma, S\rangle = \mathrm{Tr}(\Sigma S)$.

**Example 4.2.2.** *Function*

$$f(u,\xi) = \ln\left(\sum_{i=1}^{n} \xi_i e^{u_i}\right) : \mathbf{R}^n \times \mathbf{R}^n_{++} \to \mathbf{R}$$

*is a convex-concave function, and admits a saddle-point representation*

$$f(u,\xi) = \min_{v\in V}\max_{y\in Y}\left[u^T y + \xi^T v - \sum_{i=1}^{n} y_i \ln v_i - 1\right],$$

*where* $V = \{v \in \mathbf{R}^n : v > 0\}$ *and* $Y = \{y \in \mathbf{R}^n : y > 0, \sum_{i=1}^n y_i = 1\}$.

### 4.2.2 Saddle Point Representation of Convex-Concave Functions

In fact, in the developments (to be presented in next section), we need less than (4.2.2) and operate with "good" saddle point representations defined as follows.

**Definition 4.2.1 (Good saddle point representation).** *Let* $U$ *be a closed and bounded convex subset in a Euclidean space* $E_u$, *let* $\Xi \subset \mathbf{R}^m$ *be nonempty, and let* $F(u,\xi) : U\times\Xi \to \mathbf{R}$ *be a function which is convex in* $u \in U$ *for every* $\xi \in \Xi$. *Assume also that we are given closed and bounded convex sets* $X = U \times V \subset E_x := E_u \times E_v$ *and* $Y \subset E_y$, *where* $E_v$ *and* $E_y$ *are Euclidean spaces, and a function* $G_\xi(u,v;y) : X \times Y \to \mathbf{R}$, *depending on* $\xi \in \Xi$ *as a parameter, such that*

1. $G_\xi(\cdot;\cdot)$ *is continuous on* $X \times Y$ *and is convex-concave: for every* $\xi \in \Xi$, $G_\xi(u,v;y)$ *is convex in* $(u,v) \in X$ *for every fixed* $y \in Y$, *and is concave in* $y \in Y$ *for every fixed* $(u,v) \in X$;

2. $G_\xi(u,v;y)$ *is affine in* $\xi$: $G_\xi(u,v;y) = g(u,v;y) + \langle \xi, \gamma(u,v;y)\rangle$;

3. *For all* $u \in U$, $\xi \in \Xi$ *one has*

$$F(u,\xi) = \min_{v:(u,v)\in X}\max_{y\in Y} G_\xi(u,v;y).$$

*In this situation, we refer to* $(X, Y, G_\xi(\cdot;\cdot))$ *as a "good representation" of* $F(u,\xi)$.

**Definition 4.2.2** (**Simple saddle point representation**). *Given a family of regular (i.e., closed, convex, pointed and with a nonempty interior) cones $\mathcal{K}$ closed with respect to taking direct products of its elements, we call a good saddle point representation of $F(x,\xi)$ $\mathcal{K}$-simple, if*

1. *The convex compact set $Y$ in Definition 4.2.1 is of the form $Y = \{y \in \mathbf{K} : Ay \le a\}$, where $\mathbf{K} \in \mathcal{K}$;*

2. *Both $g(u,v;y)$ and $\gamma(u,v;y) = [\gamma_1(u,v;y);\ldots;\gamma_m(u,v;y)]$ are bilinear in $(u,v)$ and $y$:*

$$g(u,v;y) = \langle y, \mathcal{A}_0 u + \mathcal{B}_0 v + C_0\rangle + \langle a_0, u\rangle + \langle b_0, v\rangle + c_0,$$
$$\gamma_k(u,v;y) = \langle y, \mathcal{A}_k u + \mathcal{B}_k v + C_k\rangle + \langle a_k, u\rangle + \langle b_k, v\rangle + c_k, \forall k = 1,\ldots m,$$

   *which is essentially the same as the bilinear form of $G_\xi(u,v;y)$:*

$$G_\xi(u,v;y) = \langle p_\xi, u\rangle + \langle q_\xi, v\rangle + \langle r_\xi, y\rangle + \langle y, P_\xi u\rangle + \langle y, Q_\xi v\rangle$$

   *with $p_\xi$, $q_\xi$, $r_\xi$, $P_\xi$, $Q_\xi$ affine in $\xi$.*

In the following, we demonstrate that the above saddle point representation admit fully *algorithmic calculus*: saddle point representation of convex-concave function resulting from standard convexity-preserving operations is readily given by the saddle point representations of the operands. Such operations include taking summations with nonnegative coefficients, or direct summations, or affine substitution of variables, or taking superpositions. We list these important calculus rules below.

**Summation with positive weights.** Let $\alpha_i > 0$, $1 \le i \le I$. Let $F_i(u,\xi) : U \times \Xi \to \mathbf{R}$, $1 \le i \le I$ be given by good representations

$$F_i(u,\xi) = \min_{v^i:(u,v^i)\in X_i} \max_{y^i \in Y_i} G^i_\xi(u,v^i;y^i),\ i = 1,...,I,$$

where $G^i_\xi(u,v^i;y^i) = g^i(u,v^i;y^i) + \langle \xi, \gamma^i(u,v^i;y^i)\rangle$. Then the mapping

$$\sum_{i=1}^I \alpha_i F_i(u,\xi) : U \times \Xi \to \mathbf{R}$$

can be written as

$$\min_{v=[v^1;...;v^I]:(u,v)\in X}\ \max_{y=[y^1;...;y^I]:y\in Y}\ \sum_{i=1}^{I}\alpha_i g^i(u,v^i;y^i)+\langle\xi,\sum_{i=1}^{I}\alpha_i\gamma^i(u,v^i;y^i)\rangle:=G_\xi(u,v;y) \tag{4.2.3}$$

with $X=U\times V, V=V_1\times\ldots\times V_I, Y=Y_1\times\ldots\times Y_I$. The summation with positive weights does not affect convexity and concavity, thus $X,Y,G_\xi(u,v;y)$ is a good representation for the mapping $F(u,\xi)$.

In addition, if the representations of $F_i, 1\le i\le I$, are $\mathcal{K}$-simple, say the set $Y_i$ has the form $Y_i=\{y^i\in\mathbf{K}_i : A_iy^i\le a_i\}, \forall 1\le i\le I$, then

$$Y=\{y=[y^1;\ldots;y^I]\in\mathbf{K}:=\mathbf{K}_1\times\ldots\times\mathbf{K}_I : Ay\le a\},$$

with $A=\mathrm{diag}\{A_1,\ldots,A_I\}, a=[a_1;\ldots;a_I]$ and $K\in\mathcal{K}$, provided $\mathbf{K}_i$ are so. Also, if $g^i(u,v^i;y^i)$ and $\gamma^i(u,v^i;y^i)$ are bilinear in $(u,v^i),y^i$, then their linear combination $\sum_{i=1}^I\alpha_ig^i(u,v^i;y^i)$ and $\sum_{i=1}^I\alpha_i\gamma^i(u,v^i;y^i)$ must also be bilinear in $(u,v),y$. Thus, the representation (4.2.3) is also $\mathcal{K}$-simple. Hence, we can conclude the following proposition.

**Proposition 4.2.1.** *Good representations of $F_i(u,\xi): U\times\Xi\to\mathbf{R}, 1\le i\le I$, induce straightforwardly a good representation of the mapping*

$$F(u,\xi)=\sum_{i=1}^{I}\alpha_iF_i(u,\xi): U\times\Xi\to\mathbf{R}$$

*provided $\alpha_i>0$. Moreover, if the representations of $F_i(u,\xi)$ are $\mathcal{K}$-simple, so is the resulting representation of $F(u,\xi)$.*

**Direct summation.** Let $F_i(u^i,\xi^i): U_i\times\Xi_i\to\mathbf{R}, 1\le i\le I$, be given by good representations

$$F_i(u^i,\xi^i)=\min_{v^i:(u^i,v^i)\in X_i}\ \max_{y^i\in Y_i} G^i_{\xi^i}(u^i,v^i;y^i),\ i=1,...,I,$$

where $G^i_{\xi^i}(u^i,v^i;y^i)=g^i(u^i,v^i;y^i)+\langle\xi^i,\gamma^i(u^i,v^i;y^i)\rangle$. Then the mapping

$$F(u,\xi):=\sum_{i=1}^{I}F_i(u^i,\xi^i):\underbrace{U_1\times...\times U_I}_{U}\times\underbrace{\Xi_1\times...\times\Xi_I}_{\Xi}\to\mathbf{R},$$

with $u=[u^1;\ldots;u^I], \xi=[\xi^1;\ldots;\xi^I]$, can be written as

$$F(u,\xi)=\min_{v=[v^1;...;v^I]:(u,v)\in U\times V}\ \max_{y=[y^1;...;y^I]:y\in Y}\ \sum_{i=1}^{I}g^i(u^i,v^i;y^i)+\langle\xi,\widehat{\gamma}(u,v;y)\rangle \tag{4.2.4}$$

where $V = V_1 \times \ldots \times V_I, Y = Y_1 \times \ldots \times Y_I$, $\widehat{\gamma}(u, v; y) = [\gamma^1(u^1, v^1; y^1); \ldots; \gamma^I(u^I, v^I; y^I)]$. Note that the inner product is taken in the Euclidean space $\mathbf{R}^{m_1+\ldots+m_I}$, such that $\langle \xi, \widehat{\gamma} \rangle = \langle \xi^1, \gamma^1 \rangle + \ldots + \langle \xi^I, \gamma^I \rangle$. It is easy to see that inner function remains to be convex in $u, v$ and concave in $y$. Thus, the above provides a good representation for $F(u, \xi)$.

If the representations of $F_i, 1 \leq i \leq I$, are $\mathcal{K}$-simple, so is the representation (4.2.4). Indeed, the set $Y$ admits the simple form as previous given. Also, for each $1 \leq i \leq I$, the elements $\gamma_k^i(u^i, v^i; y^i), 1 \leq k \leq m_i$ are bilinear in $(u^i, v^i), y^i$, hence bilinear in $(u, v), y$. Thus the vector function $\widehat{\gamma}(u, v; y)$ is bilinear in $(u, v), y$. Same argument goes for the summation of $g^i(u^i, v^i; y^i)$. Hence, we can conclude the following proposition.

**Proposition 4.2.2.** *Good representations of $F_i(u^i, \xi^i) : U_i \times \Xi_i \to \mathbf{R}$, $1 \leq i \leq I$, induce straightforwardly a good representation of the mapping*

$$F(u, \xi) := \sum_{i=1}^{I} F_i(u^i, \xi^i) : \underbrace{U_1 \times ... \times U_I}_{U} \times \underbrace{\Xi_1 \times ... \times \Xi_I}_{\Xi} \to \mathbf{R},$$

*Moreover, if the representations of $F_i(u^i, \xi^i)$ are $\mathcal{K}$-simple, so is the resulting representation of $F(u, \xi)$.*

**Affine substitution of arguments.** Let a mapping $F(u, \xi) : U \times \Xi \to \mathbf{R}$ be given by a good representation:

$$F(u, \xi) = \min_{v:(u,v) \in X} \max_{y \in Y} G_\xi(u, v; y),$$

with $G_\xi(u, v; y) = g(u, v; y) + \langle \xi, \gamma(u, v; y) \rangle$. Let $w \mapsto Dw + d$, $\eta \mapsto H\eta + h$ be affine mappings taking values in the embedding spaces of $U$, $\Xi$, respectively. Then the mapping

$$\widehat{F}(w, \eta) = F(Dw + d, H\eta + h) : \underbrace{\{w : Dw + d \in U\}}_{\widehat{U}} \times \underbrace{\{\eta : H\eta + h \in \Xi\}}_{\widehat{\Xi}} \to \mathbf{R}.$$

can be written as

$$\widehat{F}(w, \eta) = \min_{v:(w,v) \in \widehat{U} \times V} \max_{y \in Y} \widetilde{g}(w, v; y) + \langle \eta, \widetilde{\gamma}(w, v; y) \rangle, \tag{4.2.5}$$

where $\widetilde{g}(w, v; y) = g(Dw + d, v; y) + \langle h, \gamma(Dw + d, v; y) \rangle$, and $\widetilde{\gamma}(w, v; y) = H^T \gamma(Dw + d, v; y)$, which indeed is a good representation.

If the representation of $F$ is $\mathcal{K}$-simple, then $\widetilde{g}(w,v;y)$ and $\widetilde{\gamma}(w,v;y)$ remain bilinear in $(w,v),y$, since the bi-linearity is not affected under affine substitution of arguments and linear transformations. Thus, the above representation is also $\mathcal{K}$-simple. Hence, we can conclude the following proposition.

**Proposition 4.2.3.** *A good representation of $F(u,\xi):U\times\Xi\to\mathbf{R}$ induces straightforwardly a good representation of the mapping*

$$\widehat{F}(w,\eta)=F(Dw+d,H\eta+h):\underbrace{\{w:Dw+d\in U\}}_{\widehat{U}}\times\underbrace{\{\eta:H\eta+h\in\Xi\}}_{\widehat{\Xi}}\to\mathbf{R}.$$

*Moreover, if the representations of $F(u,\xi)$ is $\mathcal{K}$-simple, so is the resulting representation of $\widehat{F}(w,\eta)$.*

**Theorem on superposition.** Let $F_i(u,\xi):U\times\Xi\to\mathbf{R},1\le i\le I$, be given by good representations

$$F_i(u,\xi)=\min_{v^i:(u,v^i)\in X_i}\max_{y^i\in Y_i}G^i_\xi(u,v^i;y^i)$$

and let

$$f(s)=\max_{\lambda\in\Lambda}\left[\langle R\lambda+r,s\rangle+\phi(\lambda)\right],$$

where $\Lambda\subset\mathbf{R}^\ell$ is a closed bounded and convex set and $\lambda\to R\lambda+r=[R_1\lambda+r_1;\ldots;R_I\lambda+r_I]$ is an affine mapping from $\Lambda$ to $\mathbf{R}^I_+$, and $\phi(\lambda):\Lambda\to\mathbf{R}$ is a continuous concave function. Then the superposition

$$F(u,\xi)=f(F_1(u,\xi),...,F_I(u,\xi)):U\times\Xi\to\mathbf{R}$$

can be written as:

$$F(u,\xi)=\min_{\substack{v=[v^1;\ldots;v^I]\\(u,v)\in U\times V}}\max_{\substack{z=[(w^1,\lambda^1);\ldots;(w^I,\lambda^I)]\\\in Z}}\sum_{i=1}^{I}(R_i\lambda+r_i)G^i_\xi(u,v^i;\frac{w^i}{R_i\lambda+r_i})+\phi(\lambda),\qquad(4.2.6)$$

where $V=V^1\times\ldots\times V^I$, $Z=\{z=[(w^1,\lambda^1);\ldots;(w^I,\lambda^I)]:\frac{w^i}{R_i\lambda+r_i}\in Y_i,\forall 1\le i\le I,\lambda\in\Lambda\}$. Let $T=\{t=R\lambda+r:\lambda\in\Lambda\}$; by assumption, for $t\in T$, $t_i\ge 0,\forall i$. The function $t^iG^i_\xi(u,v^i;\frac{w^i}{t^i})$ is continuous and concave in $(w^i,t^i)$, thus it is also concave in $(w^i,\lambda^i)$. Hence, we can see that the inner function of (4.2.6) is convex in $(u,v)$ and concave in $(w,\lambda)$.

Assume that $Y_i$ can be written as $Y_i = \{y^i : \psi_i(y^i) \leq 1\}$ with convex function $\psi_i$, then set $Z$ is given by a linear transformation of the convex set $\bar{Z} = \{(w,t) : t^i\psi_i(w^i/t^i) - t^i \leq 0, \forall i, t \in T\}$, which should be compact and convex. Thus, the representation (4.2.6) is a good representation.

Assume that the set $\Lambda$ has the simple form $\Lambda = \{\lambda \in \mathbf{K} : B\lambda \leq b\}$ with regular $\mathbf{K} \in \mathcal{K}$. If $Y_i$ has the simple form $Y_i = \{y^i \in \mathbf{K}_i : A_i y^i \leq a_i\}$ with regular $\mathbf{K}_i \in \mathcal{K}$, then $Z = \{z : w^i \in \mathbf{K}^i, \lambda \in \mathbf{K}, A_i w^i - a_i R_i \lambda^i \leq a_i r_i, \forall 1 \leq i \leq I, B\lambda \leq b\}$ admits the simple form. The bi-linearity of function $t^i g^i(u, v^i; \frac{w^i}{t^i})$ and $t^i\gamma^i(u, v^i; \frac{w^i}{t^i})$ can be easily derived if the functions $g^i(\cdot;\cdot)$ and $\gamma^i(\cdot;\cdot)$ are bilinear. Hence, we can conclude the following theorem.

**Proposition 4.2.4.** *Under the above assumptions, good representations of $F_i(u,\xi) : U \times \Xi \to \mathbf{R}, 1 \leq i \leq I$, induce a good representation of the superposition*

$$F(u,\xi) = f(F_1(u,\xi), ..., F_I(u,\xi)) : U \times \Xi \to \mathbf{R}$$

*Moreover, if the representations of $F_i(u,\xi)$ are $\mathcal{K}$-simple, so is the resulting representation of $F(u,\xi)$.*

**Corollary 4.2.1.** *If $F_i(u,\xi) : U \times \Xi \to \mathbf{R}, 1 \leq i \leq I$, are given by good representations, then their maximum*

$$F(u,\xi) = \max_{i=1,...,I} F_i(u,\xi) : U \times \Xi \to \mathbf{R}$$

*also admits a good representation; moreover, if the representations of $F_i(u,\xi)$ are $\mathcal{K}$-simple, so is the resulting representation of $F(u,\xi)$.*

In fact, this is a special case of the above superposition, because

$$F(u,\xi) = \max_{i=1,...,I} F_i(u,\xi) = \max_{\lambda\in\Delta} \sum_{i=1}^{I} \lambda_i F_i(u,\xi) = f(F_1(u,\xi), \ldots, F_I(u,\xi)),$$

where $\Delta = \{\lambda \geq 0 : \sum_{i=1}^{I} \lambda_i = 1\}, f(s) = \max_{\lambda\in\Delta}[\langle\lambda, s\rangle]$.

The just outlined calculus rules yield a powerful fully algorithmic calculus of saddle point representations as well as some specially good representations, which essentially suggests that the situation described in Section 4.2.1 is not all all restricted, but rather common.

### 4.2.3 The Construction and Main Results

Recall that our goal is to solve the system of convex constraints

$$\text{Find } u \in U: \quad F_i(u, \xi_*) \leq 0, 1 \leq i \leq I, \tag{4.2.7}$$

where true data $\xi_* = \mathbf{E}_{\xi \sim P}\{\xi\}$ is not available, but we can sample from $P$, and all functions $F_i(u, \xi) : U \times \Xi \to \mathbf{R}$ are convex in $u$ on $U$ and concave in $\xi$ on $\Xi$. Assume that each of the function $F_i(u, \xi)$ admits a good representation. Let

$$f(u, \xi) = \max_{i=1,\ldots,I} F_i(u, \xi).$$

Fom Corollary 4.2.1, $F(u, x)$ also admits a good representation. We say that a candidate solution $u$ is $\epsilon$-feasible to the system in (4.2.7) when $f(u, \xi_*) \leq \epsilon$.

**Course of action.** We propose to build $\epsilon$-feasible solutions to (4.2.7) by solving the following optimization problem, also referred to as *error-in-measurement optimization* problem,

$$\min_{u \in U} f(u, \xi_*), \tag{$P$}$$

where true data $\xi_* = \mathbf{E}_{\xi \sim P}\{\xi\}$ is not available, but we can sample from $P$. We assume that $U$ is a convex and compact set, and function $f(u, \xi)$ admits a good representation $(X, Y, \Phi_\xi(u, v; y))$, where $X, Y$ are compact convex sets and $\Phi_\xi(u, v; y)$ is convex in $(u, v)$, concave in $y$ and affine in $\xi$. It follows that, $(P)$ can be reformulated as a saddle-point problem:

$$\min_{x=(u,v) \in X} \max_{y \in Y} \Phi_{\xi_*}(u, v; y). \tag{$D$}$$

Assuming that both problems are solvable, we get,

$$\forall (u, v) \in X, y \in Y : f(u, \xi_*) - \mathrm{Opt}(P) \leq \epsilon_{\mathrm{sad}}(u, v; y). \tag{4.2.8}$$

This is because

$$\begin{aligned} f(u,\xi_*) - \mathrm{Opt}(P) &= \min_{v:(u,v)\in X} \max_{y\in Y} \Phi_{\xi_*}(u,v;y) - \mathrm{SadVal}(D) \\ &\le \max_{y\in Y} \Phi_{\xi_*}(u,v;y) - \mathrm{SadVal}(D) \\ &\le \max_{y\in Y} \Phi_{\xi_*}(u,v;y) - \min_{(u,v)\in X} \Phi_{\xi_*}(u,v;y) \\ &= \epsilon_{\mathrm{sad}}(u,v;y) \end{aligned}$$

That is to say, the $x$-component of any $\epsilon$-solution to $(D)$ is an $\epsilon$-solution to $(P)$.

Let $\xi \sim \mathcal{P}$ be a random variable. Note that $\Phi_{\xi_*}(u,v;y)$ is affine in $\xi_*$. Hence, a random vector from the set $\partial\Phi_{\xi}(u,v;y)$ is an unbiased estimate for the corresponding sub-differential in $\partial\Phi_{\xi_*}(u,v;y)$, meaning that we have access to stochastic oracles when solving the saddle point problem $(D)$. That being said, the reformulated saddle point problem can now be processed by a number of off-the-shelf methods, e.g. the Stochastic Approximation algorithm originated in the pioneering paper by Robbins and Monro [74] and further developed in many papers (see, e.g., [71, 72, 58, 47] and references therein). For our purposes, we will adopt the Mirror Descent Stochastic Approximation algorithm proposed in [58]. We provide below the detail of the this algorithm when applied to the problem of our interest and the corresponding well-known results from [58] for completeness. [1]

**Mirror Descent Stochastic Approximation.** We revisit here the algorithmic details of the mirror descent SA algorithm tailored to address the convex-concave saddle point $(D)$. Denote $x = (u,v)$ and $z = (x;y)$. At each iteration, we can sample $\xi_t \sim P$ and therefore have at our disposal, an unbiased stochastic sub-gradients $G_{\xi_t}(z) \in [\partial_x\Phi_{\xi_t}(x;y); -\partial_y\Phi_{\xi_T}(x;y)]$ for any input $(x;y)$ such that

$$\mathbf{E}[G_{\xi_t}(z)] \in [\partial_x\Phi_{\xi_*}(x;y); -\partial_y\Phi_{\xi_*}(x;y)].$$

Let us equip the set $Z = X \times Y$ with some distance generating function $\omega(z) : Z \to \mathbf{R}$ that is *compatible* (i.e. continuously differentiable and strongly convex with modulus 1) with

[1] Similar results can also be obtained using the Stochastic CoMP algorithm as discussed in Section 2.5.4.

respect to some norm $\|\cdot\|$. This can be obtained by aggregating the corresponding distance generating functions for the respective domains $X$ and $Y$. Let us define the prox-function, a.k.a. the Bregman distance

$$V(\widehat{z}, z) = \omega(\widehat{z}) - \omega(z) - \nabla\omega(z)^T(\widehat{z} - z)$$

and prox-mapping

$$P_z(\zeta) = \operatorname*{argmin}_{\widehat{z}\in Z}\{V(\widehat{z}, z) + \langle\zeta, \widehat{z}\rangle\}.$$

The mirror descent SA is given by the recurrence

$$z_{t+1} := [x_{t+1}; y_{t+1}] = P_{z_t}(\gamma_t G_{\xi_t}(z_t)), t = 1, \ldots, T \tag{4.2.9}$$

where the initial point $z_1 \in Z$ is chosen to be the minimizer of $\omega(z)$ on $Z$ and the step sizes $\gamma_t \geq 0, t = 1, \ldots, T$. Let us denote $\Theta[Z] = \max_{z\in Z} V(z, z_1)$ and $D_{\omega,Z} = \sqrt{2}[\sup_{z,\widehat{z}\in Z} V(z, \widehat{z})]^{1/2}$, clearly, $\Theta[Z] \leq \frac{1}{2}D_{\omega,Z}$.

**Theorem 4.2.1** ([58]). *Setting the candidate solution*

$$\bar{z}_T = \frac{\sum_{t=1}^T \gamma_t z_t}{\sum_{t=1}^T \gamma_t}.$$

***(i)*** *under the assumption that*

$$\mathbf{E}[\|G_\xi(z)\|_*^2] \leq M^2,$$

*one has*

$$\mathbf{E}[\epsilon_{sad}(\bar{z}_T)] \leq \left[\sum_{t=1}^T \gamma_t\right]^{-1}\left[2\Theta[Z] + \frac{5}{2}M^2\sum_{t=1}^T \gamma_t^2\right], \tag{4.2.10}$$

*In particular, when setting* $\gamma_t = \frac{2\theta D_{\omega,Z}}{M_*\sqrt{5T}}, t = 1, \ldots, T$, *the efficiency becomes*

$$\mathbf{E}[\epsilon_{sad}(\bar{z}_T)] \leq \frac{2\sqrt{5}\max\{\theta, \theta^{-1}\}MD_{\omega,Z}}{\sqrt{T}}.$$

***(ii)*** *under the assumption that*

$$\mathbf{E}[\exp\{\|G_\xi(z)\|_*^2/M^2\}] \leq \exp\{1\},$$

*with the above choice of stepsize, one has, for any* $\Lambda > 0$,

$$\text{Prob}\left\{\epsilon_{sad}(\bar{z}_T) > \frac{(8 + 2\Lambda)\max\{\theta, \theta^{-1}\}\sqrt{5}MD_{\omega,Z}}{\sqrt{T}}\right\} \leq 2\exp\{-\Lambda\}$$

**Our construction** contains two steps:

1. *optimization step:* Draw $N_1$ i.i.d. training samples $\{\xi_i^{train}, i = 1, \ldots, N_1\}$ and run the mirror descent SA algorithm on problem $(D)$ to obtain a candidate solution $\widehat{z} = [\widehat{u}, \widehat{v}; \widehat{y}] \in Z$; the $u$-component of $\widehat{z}$ is a feasible solution to problem $(P)$ such that with probability at least $1 - \delta_1$,

$$f(\widehat{u}, \xi_*) - \mathrm{Opt}(P) \leq O(1)\frac{MD_{\omega,Z}\log(1/\delta_1)}{\sqrt{N_1}}.$$

   This above result follows direct Theorem 4.2.1 and the relation in (4.2.8).

2. *validation step:* Draw another $N_2$ i.i.d. testing samples $\{\xi_j^{test}, j = 1, \ldots, N_2\}$ to compute a reliable upper bound $\widehat{f}_{N_2,\delta_2}(\widehat{u})$ of the true function value $f(\widehat{u}, \xi_*)$, such that with probability $1 - \delta_2$,

$$f(\widehat{u}, \xi_*) \leq \widehat{f}_{N_2,\delta_2}.$$

   We will establish such upper bounds in the next section.

### 4.2.4 Upper Bound

The true objective value of the original problem $(P)$ at a candidate solution produced by optimization step cannot be computed since $\xi_*$ is unknown. Our goal in this section is to establish some reliable upper bounds, which serve as "reasonably good" estimates of the true objective. First, we need to make some assumptions on the underlying probability density function of $\xi \sim P$. We consider a widely-used family of distributions–the subgaussian distributions, in the sequel. Here is the definition.

**Definition 4.2.3** (**Sub-Gaussianity**). *A random vector $\eta \in \mathbf{R}^n$ is said to be subgaussian with parameter $\Sigma \succeq 0$ denoted as $\eta \sim \mathrm{SG}(\Sigma)$ if $\mathbf{E}[e^{\beta^T\eta}] \leq e^{\frac{\beta^T\Sigma\beta}{2}}, \forall \beta \in \mathbf{R}^n$.*

Here are some useful properties of subgaussian random vectors (which can be found e.g. in [15]). Assume $\xi \in \mathbf{R}^n$ , $\eta \in \mathbf{R}^n$ are independent subgaussian random vectors.

1. If $\xi \sim \mathrm{SG}(\Sigma)$, then $\mathbf{E}[\xi] = 0$.

2. If $\xi \sim \mathrm{SG}(\Sigma)$, $A \in \mathbf{R}^{m\times n}$, then $A\xi \sim \mathrm{SG}(A\Sigma A^T)$.

3. If $\xi \sim \mathrm{SG}(\Sigma_1)$ and $\eta \sim \mathrm{SG}(\Sigma_2)$, then $\xi + \eta \sim \mathrm{SG}(\Sigma_1 + \Sigma_2)$.

4. If $\xi \sim \mathrm{SG}(\Sigma)$, then for any $t \geq 0$,

$$\mathrm{Prob}(\beta^T \xi \geq t) \leq \exp\left\{-\frac{t^2}{2\beta^T \Sigma \beta}\right\}, \forall \beta \in \mathbf{R}^n. \tag{4.2.11}$$

Now let us consider the following assumptions on $\xi \sim P$.

**Assumption 4.2.1.** *$\xi \sim P$ satisfies: $\xi = A\eta + \xi_*$, where random variable $\eta \in \mathbf{R}^p$ and $\eta \sim \mathrm{SG}(Q)$ for some $Q \in \mathcal{S}_+^p$, and matrix $A \in \mathbf{R}^{n \times p}$ is given.*

**Assumption 4.2.2.** *$\xi \sim P$ satisfies: $\xi = A\eta + \xi_*$, where random variable $\eta \sim SG(Q)$ with diagonal $p \times p$ matrix $Q$ has independent entries, and $A = \mathrm{Diag}[A^{(1)}, A^{(2)}, \ldots, A^{(r)}]$ with unknown blocks $A^{(j)} \in \mathbf{R}^{n_j \times p_j}, j = 1, \ldots, r$, $\sum n_j = n, \sum p_j = p$.*

Note that Assumption 4.2.1 allows $\xi$ to have dependent entries and Assumption 4.2.2 allows $\xi$ to be split into several independent blocks with unknown in advance dependency structure within the entries of a single block. This is often the case for the portfolio selection problem where we may treat $A$ as the factor loading matrix with a low rank (i.e., $p \ll n$), and $\eta$ as the factor vector.

**Upper Bound I.** Let us denote the candidate solution of $(D)$ yielded by the optimization step as $(\widehat{u}, \widehat{v}; \widehat{y})$. First of all, we compute the empirical mean $\bar{\xi}$ of the training samples, the function value $f(\widehat{u}, \bar{\xi})$ and a subgradient $g(\widehat{u}, \bar{\xi}) \in \partial_\xi f(\widehat{u}, \bar{\xi})$ at this point. Invoking concavity of $f(\widehat{u}, \cdot)$, we have

$$f(\widehat{u}, \xi_*) \leq f(\widehat{u}, \bar{\xi}) + g(\widehat{u}, \bar{\xi})^T (\xi_* - \bar{\xi}).$$

We see that in order to upper-bound $f(\widehat{u}, \xi_*)$, it suffices to upper-bound the linear form of $\xi_*$ in the right of the formula. To this end, we simply define

$$\bar{f}_N := f(\widehat{u}, \bar{\xi}) + \frac{1}{N} \sum_{j=1}^{N} g(\widehat{u}, \bar{\xi})^T (\xi_j - \bar{\xi}), \tag{4.2.12}$$

where $\xi_1, \ldots, \xi_N$ are testing samples, which are independent from $\bar{\xi}$. We immediately arrive at the following results.

**Proposition 4.2.5.** *Under Assumption 4.2.1, let* $\Omega = \sqrt{g_0^T AQA^T g_0}$, *where* $g_0 = g(\widehat{u}, \bar{\xi})$, *then*

$$\text{Prob}\left(f(\widehat{u}, \xi_*) \geq \bar{f}_N + \frac{\gamma\Omega}{\sqrt{N}}\right) \leq \exp\left\{-\frac{\gamma^2}{2}\right\}, \forall \gamma > 0. \tag{4.2.13}$$

*Proof.* First of all, by concavity of $f(u, \xi)$ in $\xi$ we have

$$f(\widehat{u}, \xi_*) - \bar{f}_N \leq \frac{1}{N}\sum_{j=1}^{N} g_0^T(\xi_j - \xi_*). \tag{4.2.14}$$

Note that by definition, for any $j$, $\xi_j - \xi_* \sim \text{SG}(AQA^T)$. From the above properties of subgaussian random vectors, we have $\frac{1}{N}\sum_{j=1}^{N}(\xi_j - \xi_*) \sim \text{SG}(AQA^T/N)$. Invoking the inequality (4.2.11) with $\beta = g_0$, we get

$$\text{Prob}\left(\frac{1}{N}\sum_{j=1}^{N} g_0^T(\xi_j - \xi_*) \geq t\right) \leq \exp\left\{-\frac{t^2 N}{2\Omega^2}\right\}.$$

Setting $t = \gamma\Omega/\sqrt{N}$ and invoking (4.2.14), we get the desired result. □

**Proposition 4.2.6.** *Under Assumption 4.2.2, assume that A belongs to the uncertainty set* $S = \{A : ||A_i||_2 \leq \rho, \forall i = 1, \ldots, p\}$, *where* $A_i$ *stands for the i-th column of A. Then* $\forall \gamma > 0$,

$$\text{Prob}\left(f(\widehat{u}, \xi_*) \geq \bar{f}_N + \frac{\gamma\rho\sqrt{\sum_{j=1}^{r} \|g_0^{(j)}\|_2^2 \text{Tr}(Q^{(j)})}}{\sqrt{N}}\right) \leq \exp\left\{-\frac{\gamma^2}{2}\right\}, \tag{4.2.15}$$

*where* $g_0 = g(\widehat{u}, \bar{\xi})$, *and* $g_0^{(j)}$ *are the consecutive blocks, of sizes* $p_1, ..., p_r$, *in* $g_0$, *and* $Q^{(j)}$ *are the consecutive* $p_j \times p_j$ *diagonal blocks in the diagonal matrix* $Q$.

*Proof.* The proof follows directly from Proposition 4.2.5 and the observation that

$$\max_{A:\|A_i\|_2 \leq \rho} g_0^T AQAg_0 \leq \max_{A:\|A_i\|_2 \leq \rho} \sum_{j=1}^{r} \text{Tr}(A^{(j)} Q^{(j)} A^{(j)^T}) \|g_0^{(j)}\|_2^2 \leq \rho \sum_{j=1}^{r} \text{Tr}(Q^{(j)}) \|g_0^{(j)}\|_2^2.$$

□

In the sequel, we provide another simple strategy to obtain upper bounds when function $f(u, \xi)$ possesses appropriate structure.

**Upper Bound II.** We assume that $f(u,\xi)$ admits a $\mathcal{K}$-simple saddle point representation, i.e.,

$$f(u,\xi) = \min_{v:(u,v)\in X} \max_{y\in Y} \Phi_\xi(u,v;y)$$

where $\Phi_\xi(u,v;y)$ is bilinear

$$\Phi_\xi(u,v;y) = \langle y, P_\xi u + Q_\xi v + R_\xi\rangle + \langle p_\xi, u\rangle + \langle q_\xi, v\rangle + c_\xi$$

with $P_\xi = \sum_i \xi_i P_i + P_0$, $Q_\xi = \sum_i \xi_i Q_i + Q_0$, $R_\xi = \sum_i r_i\xi_i + r_0$, $p_\xi = P\xi + p_0$, $q_\xi = Q\xi + q_0$, $c_\xi = c^T\xi + c_0$ that are all affine in $\xi$ with matrices $P_i, Q_i, r_i, P, Q, c$ of proper dimensions.

Let us denote the candidate solution to $(D)$ produced by the mirror descent SA algorithm by $(\widehat{u},\widehat{v};\widehat{y})$, and consider the following approximation of $f(\widehat{u},\xi_*)$:

$$F(\widehat{u},\widehat{v}) := \max_{y\in Y} \langle y, \underbrace{P_{\xi_*}\widehat{u} + Q_{\xi_*}\widehat{v} + R_{\xi_*}}_{\beta_*}\rangle + \underbrace{\langle p_{\xi_*},\widehat{u}\rangle + \langle q_{\xi_*},\widehat{v}\rangle + c_{\xi_*}}_{\alpha_*}.$$

Note that $F(\widehat{u},\widehat{v}) \geq f(\widehat{u},\xi_*)$.

Let $(E^*, \|\cdot\|_*)$ be the dual space to $(E,\|\cdot\|)$, where $\|\cdot\|$ is some norm. Assuming $Y \subseteq \{y \in E_y^* : \|y\|_* \leq R\}$ for some $R > 0$, we have $F(\widehat{u},\widehat{v}) \leq R\|\beta_*\| + \alpha_*$, where $\beta_*$ and $\alpha_*$ are defined above. Noting that both $\alpha_*$ and $\beta_*$ are affine in $\xi_*$, we can construct their unbiased estimates using the testing samples,

$$\widehat{\alpha}_N = \frac{1}{N}\sum_{j=1}^N \left(\langle p_{\xi_j},\widehat{u}\rangle + \langle q_{\xi_j},\widehat{v}\rangle + c_{\xi_j}\right)$$
$$\widehat{\beta}_N = \frac{1}{N}\sum_{j=1}^N \left(P_{\xi_j}\widehat{u} + Q_{\xi_j}\widehat{v} + r_{\xi_j}\right)$$

where $\xi_1,\ldots,\xi_N$ are i.i.d. with $\mathbf{E}[\xi_j] = \xi_*, \forall j$. Let us set

$$\bar{f}_N := \widehat{\alpha}_N + R\|\widehat{\beta}_N\|. \tag{4.2.16}$$

We clearly have

$$f(\widehat{u},\xi_*) \leq F(\widehat{u},\widehat{v}) \leq \alpha_* + R\|\beta_*\| \leq \bar{f}_N + |\alpha_* - \widehat{\alpha}_N| + R\|\beta_* - \widehat{\beta}_N\|. \tag{4.2.17}$$

Essentially, we would like to bound from above the two terms $|\alpha_* - \widehat{\alpha}_N|$ and $\|\beta_* - \widehat{\beta}_N\|$.

Denoting $\zeta_j = \xi_j - \xi_*, \forall j$, so that $\mathbf{E}[\zeta^j] = 0$, we have

$$\widehat{\alpha}_N - \alpha_* = \frac{1}{N}\sum_{j=1}^{N}\langle \zeta^j, P^T\widehat{u} + Q^T\widehat{v} + c\rangle := \frac{1}{N}\sum_{j=1}^{N}\langle \zeta_j, b(\widehat{u},\widehat{v})\rangle,$$
$$\widehat{\beta}_N - \beta_* = \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{m}(\zeta_j)_i(P_i\widehat{u} + Q_i\widehat{v} + r_i) := \frac{1}{N}\sum_{j=1}^{N} B(\widehat{u},\widehat{v})\zeta_j,$$

where the $i$-th column of matrix $B(\widehat{u},\widehat{v})$ is the vector $P_i\widehat{u} + Q_i\widehat{v} + r_i$. Let $\xi \sim P$, note that when $(\xi - \xi_*)$ follows some subgaussian distribution as in Assumption 4.2.1 and 4.2.2, the vector $\widehat{\beta}_N - \beta_*$ is also subgaussian random vector. However, it is unclear how to get a dimension-independent bound for the norm of a sum of independent subgaussian random vectors.

In [46] and [57], the authors derive exponential bounds on the probability of large deviations of random sums for some light tail distributions defined on finite-dimensional normed spaces. We hereby revisit some of the important results established in these references.

**Theorem 4.2.2** (see [46]). *Let $(E, \|\cdot\|)$ be $\kappa$-regular [2] , let $\{\zeta_j\}_{j=1}^N$ be a sequence of each other zero mean random vectors from $E$ such that*

$$\mathbf{E}_{j-1}\left\{\exp\{\|\zeta_j\|^2\sigma_j^{-2}\}\right\} \le \exp\{1\}, \forall j,$$

*then*

$$\text{Prob}\left\{\|\sum_{j=1}^{N}\zeta_j\| \ge [\sqrt{2e\kappa} + \sqrt{2}\gamma]\sqrt{\sum_{j=1}^{N}\sigma_j^2}\right\} \le 2\exp\{-\gamma^2/64\}.$$

For our purposes, we will focus on this set of "light-tail" distribution family for $P$.

**Assumption 4.2.3.** *The space $(E, \|\cdot\|)$ is a $\kappa$-regular, and $\xi \sim P$ satisfies:*

$$\mathbf{E}\left\{\exp\{\|\xi - \xi_*\|^2\sigma^{-2}\}\right\} \le \exp\{1\}$$

*for some $\sigma > 0$, where $\xi_* = \mathbf{E}[\xi]$.*

[2] *An informal definition of this regularity is that the norm on the space can be approximated, within an absolute constant factor, by a norm which is differentiable on the unit sphere with a Lipschitz continuous gradient, formal definition can be found in [46]. For instance, when $2 \le q \le \infty$, the space $(\mathbf{R}^d, \|\cdot\|_q)$ is $\kappa$-regular with $\kappa \le \min\{q-1, 2\ln(d)\}$.*

Let $\Omega_1 = \|b(\widehat{u},\widehat{v})\|_*$, $\Omega_2 = \max_{z:\|z\|=1} \|B(\widehat{u},\widehat{v})z\|$. Hence, we have

$$f(\widehat{u},\xi_*) \leq \bar{f}_N + \frac{(\Omega_1 + R\Omega_2)}{N}\|\sum_{j=1}^{N}\zeta_j\|.$$

It immediately follows from the above large deviation results that

**Corollary 4.2.2.** *Under Assumption 4.2.3, we have for all $\gamma \geq 0$,*

$$\text{Prob}\left\{\bar{f}_N < f(\widehat{u},\xi_*) - \frac{\sigma(\Omega_1 + R\Omega_2)(\sqrt{2e\kappa}+\sqrt{2}\gamma)}{\sqrt{N}}\right\} \leq 4\exp\{-\gamma^2/64\}$$

*When $\|\cdot\|$ is the Euclidean norm, we further have for all $\gamma \geq 1$,*

$$\text{Prob}\left\{\bar{f}_N < f(\widehat{u},\xi_*) - \frac{\sigma(\Omega_1 + R\Omega_2)(\sqrt{2\kappa}+\sqrt{2}\gamma)}{\sqrt{N}}\right\} \leq 2\exp\{-\gamma^2/3\}.$$

**Remark.** So far, we have presented two constructive ways to build reliable upper bounds on the function $f(\widehat{u},\xi_*)$, where $\widehat{u}$ is the candidate solution yielded by the optimization step of our procedure for the error-in-measurement optimization problem $(P)$, when the underlying sampling distribution has light tail. It also makes sense to consider heavy tail distributions (e.g. lognormal distribution), or even situations where we only have access to the bounds of certain moments. In those situations, one might need to resort to some more sophisticated resampling and estimation techniques such as jackknifing and bootstrapping, but these extensions go beyond the scope of this Thesis.

### 4.2.5 Concluding Remarks.

In this section, we have introduced the notion of of an error-in-measurement optimization, where we seek a feasible solution to a system of convex constraints $f_i(x, \mathbf{E}[\xi]) \leq 0$, $i \leq I$ with the data vector represented as the expected value $\mathbf{E}[\xi]$ of an unknown distribution from which we can draw independent samples ("measurements"). A straightforward approach to handling the situation would be to use a sample of measurements in order to build an estimate $\widehat{\xi_*}$ of $\xi_* = \mathbf{E}[\xi]$, plug this estimate into the constraints and to solve the resulting "certain" – with known vector of parameters – system of constraints. A drawback of this "plug in" approach is that it is not clear what should be the accuracy to which we need to recover $\xi_*$ in order to get a good solution to the problem of interest. We propose

an alternative approach, base on specific "saddle point" representations of the functions $f_i(x, \xi)$, and develop a fully algorithmic calculus of these representations (which, in light of this calculus, are a "common commodity"). With our approach, finding a feasible solution to the feasibility problem of interest reduces to solving a convex-concave game for which an unbiased stochastic first order oracle is available (it is readily given by measurements). We suggest to find an approximate saddle point of the game by Mirror Descent Stochastic Approximation, and develop a rigorously justified procedure allowing to validate the quality of the resulting candidate solution to the problem of interest. Note that our validation procedure is independent of how the candidate solution is obtained, and thus is applicable when the "plug in" approach is used.

## *4.3 Convex Optimization with Indirect Noisy Observations*

In the previous section , we have developed a saddle-point-based framework to solve convex feasibility problems with uncertain data represented as the expectation of a distribution from which we can draw samples, and thus – with the data allowing for direct unbiased measurements. In this section we consider "indirect stochastic programming" – the situation where direct measurement of unknown data is not allowed.

### 4.3.1 Indirect Stochastic Programming

**The situation.** Consider the situation as follows. We are given

- a *signal space* – a set $\Pi$,
- an *observation space* $\Omega$, where $\Omega$ is a complete separable metric space, and a family $\{P_\pi(\cdot)\}_{\pi\in\Pi}$ of Borel probability distributions on $\Omega$ parameterized by signals $\pi \in \Pi$,
- a *control space* – a convex set $X \subset \mathbf{R}^n$, and a real-valued *loss function* $\Phi(x, \pi) : X \times \Pi \to \mathbf{R}$ which is convex in $x \in X$.

The problem we are interested in is as follows:

*Given independent observations*

$$\omega_t \sim P_{\pi^*}(\cdot),\ t = 1, 2, ... \tag{4.3.1}$$

*coming from an unknown $\pi^*$ known to belong to $\Pi$, we want to solve the optimization problem*

$$\min_{x\in X} \Phi(x,\pi^*) \qquad (\mathcal{P}[\pi^*])$$

This setting is essentially different from the one we have considered in Section 4.2.1. A minor difference is that now we are speaking about solving a convex optimization problem rather than a convex feasibility problem. More important differences are that now we do not assume neither concavity in the unknown parameter, nor good saddle point representation, nor the fact that we are allowed for direct, albeit noisy, observations of the parameter. Let us illustrate our problem setting with two important examples first.

**Example I (Affine Signal Processing)** We want to recover the image $B(\pi_*)$ of some unknown signal $\pi_* \in \Pi \subset \mathbf{R}^q$ under a given mapping $\pi \to B(\pi) : \Pi \to \mathbf{R}^k$. Assume our observations are $\omega_t = A\pi_* + \eta_t$, where $A$ is a given matrix, $\eta_t$ are i.i.d. zero mean random noise. We want to solve the quadratic optimization problem

$$\min_{x\in X} \; x^T x - 2x^T B(\pi_*)$$

where $X$ is some convex set that contains $B(\Pi)$.

**Example II (Indirect Support Vector Machines)** We want to learn a linear classifier from observations corrupted by random noise. Specifically, we observe i.i.d. pairs $\omega_t = (s_t, \xi_t + \eta_t) \in \mathbf{R}^p$ where $(s_t, \xi_t)$ are sampled from unknown Borel probability distribution $\pi^*$ on $\{1,-1\} \times \mathbf{R}^{p-1}$, and $\eta_t$ are random noises independent of $(s_t, \xi_t)$ and sampled from a partially known distribution. We wish to minimize the expected hinge loss with respect to the uncorrupted data, namely to solve the stochastic optimization problem

$$\min_{x=[u;\gamma]\in X} \mathbf{E}_{[s;\xi]\sim\pi^*}\{\max[1 - s[u^T\xi + \gamma], 0]\}$$

where $X \subset \mathbf{R}^s \times \mathbf{R}$ is a given convex set.

Note that both examples fall into the outlined Indirect Stochastic Programming setting. In the second example, the unknown signal $\pi^*$ actually stands for a distribution. We are going to refer this type of problems as *indirect stochastic programming.* Note that

problem's "parameter" $\pi^*$ (which, as the second example shows, could even be infinite dimensional) is observed *indirectly* and in the presence of noise, which moves the problem beyond the "immediate scope" of standard techniques of Stochastic Programming, like Stochastic Approximation or Sample Average Approximation. Our goal is to develop an approach which brings the problem into the scope of these techniques.

### 4.3.2 A General Approximation Framework

Let $\mathcal{F}$ be a finite dimensional linear subspace in the space of real-valued functions on $\Omega$, and let

$$\mathcal{X} = \{(f, x) \in \mathcal{F} \times X : \int_\Omega f(\omega) P_\pi(d\omega) \geq \Phi(x, \pi) \ \forall \pi \in \Pi\}. \tag{4.3.2}$$

We clearly have

**Proposition 4.3.1.** *$\mathcal{X}$ is a convex set.*

*Proof.* Suppose $(f_1, x_1) \in \mathcal{X}$ and $(f_2, x_2) \in \mathcal{X}$, then for any $\lambda \in [0, 1]$, we have $\forall \pi \in \Pi$,

$$\begin{aligned}
\int_\Omega [\lambda f_1 + (1-\lambda) f_2](\omega) P_\pi(d\omega) &= \lambda \int_\Omega f_1(\omega) P_\pi(d\omega) + (1-\lambda) \int_\Omega f_2(\omega) P_\pi(d\omega) \\
&\geq \lambda \Phi(x_1, \pi) + (1-\lambda)\Phi(x_2, \pi) \qquad \text{[by definition of } \mathcal{X}] \\
&\geq \Phi(\lambda x_1 + (1-\lambda) x_2, \pi) \quad \text{[by convexity of } \Phi(x, \pi) \text{ in } x]
\end{aligned}$$

which implies that $\mathcal{X}$ is convex. □

As a result, the convex stochastic program

$$\min_{(f,x) \in \mathcal{X}} F(f, x) := \mathbf{E}_{\omega \sim p_{\pi^*}}[f(\omega)] \tag{$\mathcal{S}[\pi^*]$}$$

is a safe approximation of $(\mathcal{P}[\pi^*])$: the $x$-component of a feasible solution $(f, x)$ to the approximation is feasible for the problem of interest $(\mathcal{P}[\pi^*])$, and the value of the objective of the approximating problem at $(f, x)$ is an upper bound on the value of the "true" objective at $x$. On the other hand, we can sample from the distribution $P_{\pi^*}(\cdot)$, and thus in principle, we can solve the approximating problem to a desired accuracy by Stochastic Approximation [74, 72, 58] or by Sample Average Approximation (SAA), i.e. by minimizing the empirical sample-based approximation of the true expectation.

**Trade-off between approximation, estimation, and optimization error** With the SAA approach, one solves the problem

$$\min_{(f,x)\in\mathcal{X}} F^N(f,x) := \frac{1}{N}\sum_{t=1}^{N} f(\omega_t) \qquad (\mathcal{SAA}[\pi^*])$$

The excess error of this procedure can be decomposed into three terms:

$$\mathcal{E} = \mathcal{E}_{app} + \mathcal{E}_{est} + \mathcal{E}_{opt}$$

an *approximation error* term that comes from the restriction of domain using $\mathcal{F}$; an *estimation error* term that comes from Monte Carlo estimation; an *optimization error* term that comes from the inaccuracy of solutions provided by optimization solvers given fixed time budget. Observe that there is a delicate trade-off between these errors: when we enlarge $\mathcal{F}$, the approximation error decreases, while both the estimation and optimization errors increase. In order to fully characterize the error of the outline approach, we have to address the following questions

(i) (*consistency*): How to select $\mathcal{F}$ in order to recover an exact solution at least asymptotically, i.e., as $N \to \infty$ ?

(ii) (*tractability*): since the set $\mathcal{X}$ is represented by a semiinfinite system of linear constraints on $f$, a natural question is under what choices of $\mathcal{F}$ would this set be computationally tractable?

(iii) (*efficiency*): in order to ensure "good" consistency and tractability, we might need to work with very large and complex domain $\mathcal{X}$, which creates additional challenges for SA and SAA. The question is to which extent we can circumvent these difficulties.

The above questions are highly challenging, and there seems to be no universal answers. Our goal here is to investigate the outlined approach in a case-by-case manner, hoping to shed some light on its potential in several specific applications.

### 4.3.3 Application I: Affine signal processing

Assume that $\Pi$ is a compact set in some $\mathbf{R}^q$, $\Omega = \mathbf{R}^p$, and our observations are

$$\omega_t = A\pi^* + \eta_t, \qquad (4.3.3)$$

where $A$ is a given matrix, $\eta_t$ are i.i.d. zero mean observation noises with known covariance matrix $H$, and $\pi^* \in \Pi$. Let our goal be to recover the image $B(\pi^*)$ of $\pi^*$ under a given mapping $\pi \mapsto B(\pi) : \Pi \to \mathbf{R}^k$. In the sequel, we intend to use the parametric family of quadratic in $\omega$ functions $f$, that is, we set

$$\mathcal{F} = \left\{ f(\omega) = \omega^T D\omega - 2\omega^T d + \delta : D \in \mathbf{S}^p, d \in \mathbf{R}^p, \delta \in \mathbf{R} \right\}. \tag{4.3.4}$$

#### *4.3.3.1 Scalar case ($k = 1$)*

We start with the case when $B(\cdot)$ is a real-valued function given by *Fenchel-type representation*

$$B(\pi) = \min_{x \in X} \left[ \Phi(x, \pi) := \pi^T [Rx + r] + b(x) \right] \tag{4.3.5}$$

where $X$ is a convex set in some $\mathbf{R}^n$ and $b(\cdot)$ is convex on $X$. In this case computing $B(\pi^*)$ indeed reduces to solving the problem $(\mathcal{P}[\pi^*])$, so that the optimal value in $(\mathcal{S}[\pi^*])$ upper-bounds $B(\pi^*)$. With $\mathcal{F}$ given by (4.3.4), the problem $(\mathcal{S}[\pi^*])$ becomes:

$$\nu(\pi^*) := \min_{(D,d,\delta,x) \in \mathcal{Z}} \mathbf{E}_{\omega \sim p_{\pi^*}} [\omega^T D\omega - 2\omega^T d + \delta] \tag{4.3.6}$$

where

$$\mathcal{Z} := \left\{ (D, d, \delta, x) : \begin{array}{r} x \in X, \\ \pi^T A^T DA\pi + \mathrm{Tr}(DH) - 2\pi^T A^T d + \delta \\ \geq \pi^T [Rx + r] + b(x) \ \forall \pi \in \Pi \end{array} \right\}. \tag{4.3.7}$$

Note that $\mathcal{Z}$ is nothing but the set $\mathcal{X}$ from (4.3.2) described in terms of $x$ and the parameters $(D, d, \delta)$ specifying our quadratic functions $f$ rather than in terms of $x$ and $f$, as in (4.3.2). The parameterized by $\pi$ constraints in (4.3.7) are nothing but the constraints in (4.3.2), since with quadratic $f(\omega) = \omega^T D\omega - 2\omega^T d + \delta$, one has

$$\mathbf{E}_{\omega \sim p_\pi}[f(\omega)] = \int_\Omega f(\omega) P_\pi(d\omega) = \pi^T A^T DA\pi + \mathrm{Tr}(DH) - 2\pi^T A^T d + \delta.$$

Given observations $\omega_1, \omega_2, \ldots, \omega_N$, the sample average approximation (SAA) of problem (4.3.6) becomes

$$\nu^N(\pi^*) := \min_{(D,d,\delta,x) \in \mathcal{Z}} [\mathrm{Tr}(DW) - 2\bar{w}^T d + \delta] \tag{4.3.8}$$

where $W = \frac{1}{N} \sum_{i=1}^N \omega_t \omega_t^T$, $\bar{w} = \frac{1}{N} \sum_{i=1}^N \omega_t$.

***Exact Recovery and Consistency*** We first show that under mild assumptions, the optimal objective value of (4.3.6) is not just an upper bound of $B(\pi^*)$, but exactly equal to $B(\pi^*)$.

**Proposition 4.3.2.** *When the observation scheme (4.3.3) is given by an invertible $A$, $B(\pi^*)$ can be exactly recovered by solving optimization problem (4.3.6), i.e. $\nu(\pi^*) = B(\pi^*)$.*

*Proof.* Denote $x^*$ as the optimal solution to the problem $(\mathcal{P}[\pi^*])$. Hence, $x^* \in X$ and

$$B(\pi^*) = (\pi^*)^T(Rx^* + r) + b(x^*).$$

Let $D^* = 0, d^* = -\frac{1}{2}A^{-1}(Rx^* + r), \delta^* = b(x^*)$. It is easily seen that $(D^*, d^*, \delta^*, x^*) \in \mathcal{Z}$ is a feasible solution to (4.3.6). Moreover,

$$\mathbf{E}_{\omega\sim p_{\pi^*}}[\omega^T D^*\omega - 2\omega^T d^* + \delta^*] = (\pi^*)^T(Rx^* + r) + b(x^*) = B(\pi^*).$$

So $\nu(\pi^*) \leq B(\pi^*)$. Recalling that by construction $\nu(\pi^*)$ always is an upper bound on $B(\pi^*)$, the conclusion of Proposition follows. □

**Remark.** The above proposition holds true even if $\mathcal{F}$ is set to be the family of linear functions of $\omega$. The assumption that $A$ is invertible can be somehow relaxed, but we prefer to omit the related refinements.

**Tractability** The set $\mathcal{Z}$ clearly is convex, but not necessarily is computationally tractable. However, in many cases, we have at our disposal a computationally tractable convex subset $\mathcal{Z}^+$ of $\mathcal{Z}$, and we can associate problem (4.3.6) with $\mathcal{Z}^+$ in the role of $\mathcal{Z}$. Let us look at some examples.

A. $\Pi$ *is given by a single strictly feasible quadratic inequality;*

   In this case, the set $\mathcal{Z}$ is computationally tractable.

B. $\Pi$ *is a computationally tractable convex set;*

   In this case, the set $\mathcal{Z}^+ = \{(D, d, \delta, x) \in \mathcal{Z} : D \succeq 0\}$ is computationally tractable.

C. $\Pi$ *is given by a system of quadratic in* $\pi$ *inequalities* $\mathcal{S}_j(\pi) \leq 0$, $1 \leq j \leq J$.

In this case, the set

$$\mathcal{Z}^+ = \left\{ (D,d,\delta,x) : \exists t, \{\lambda_i \geq 0\} : \begin{array}{c} x \in X,\, t \geq b(x) \\ \pi^T A^T D A \pi - \pi^T [2A^T d + Rx + r] + \sum_j \lambda_j S_j(\pi) \\ + [\mathrm{Tr}(DH) + \delta - t] \geq 0,\ \forall \pi \in \mathbf{R}^q \end{array} \right\} \tag{4.3.9}$$

clearly is contained in $\mathcal{Z}$ and is computationally tractable, since the semi-infinite constraint in the description of $\mathcal{Z}^+$ reduces to a Linear Matrix Inequality in variables $D, d, \delta, x, t, \{\lambda_i\}$. Assume that the quadratic inequalities specifying $\Pi$ are :

$$\mathcal{S}_j(\pi) := \pi^T S \pi + 2 s_j^T \pi + \sigma_j \leq 0,\ 1 \leq j \leq J. \tag{4.3.10}$$

Then, the above set $\mathcal{Z}^+$ reads

$$\mathcal{Z}^+ = \left\{ (D,d,\delta,x) : \begin{array}{c} \exists t, \{\lambda_i \geq 0\} \text{ such that } x \in X,\, t \geq b(x) \\ \left[ \begin{array}{cc} A^T D A + \sum\limits_j \lambda_{j=1}^J S_j & -A^T d + \frac{Rx+r}{2} + \sum\limits_{j=1}^J \lambda_j s_j \\ -d^T A + \frac{(Rx+r)^T}{2} + \sum\limits_{j=1}^J \lambda_j s_j^T & \sum\limits_{j=1}^J \lambda_j \sigma_j + \mathrm{Tr}(DH) + \delta - t \end{array} \right] \succeq 0 \end{array} \right\} \tag{4.3.11}$$

In particular, when $X$ and epigraph of $b(x)$ are semidefinite representable, the SAA problem (4.3.8) with $\mathcal{Z}^+$ in the role of $\mathcal{Z}$ is a semidefinite program.

D. *Well-structured case where* $\Pi$, $X$, *and epigraph of* $b(x)$ *are conic representable*

In this case, the set $\mathcal{Z}^+ = \{(D,d,\delta,x) \in \mathcal{Z} : D = 0\}$ is given by a system of conic constraints.

Assume that we are in the *well-structured case*, that is,

- $\Pi$ is given by conic representation

$$\Pi = \{\pi : \exists u_\pi : P_\pi \pi + Q_\pi u_\pi - \sigma_\pi \in \mathbf{K}_\pi\} \tag{4.3.12}$$

- $\Phi(x,\pi) = \pi^T [Rx + r] + b(x)$, where the epigraph of $b(x)$ is given by conic representation

$$t \geq b(x) \quad \Leftrightarrow \quad \exists u_\sigma : tp + P_b x + Q_b u_b - \sigma_b \in \mathbf{K}_b \tag{4.3.13}$$

- $X$ is given by conic representation

$$X = \{x : \exists u_x : P_x x + Q_x u_x - \sigma_x \in \mathbf{K}_x\}. \tag{4.3.14}$$

where $\mathbf{K}_\pi, \mathbf{K}_b, \mathbf{K}_x$ are regular cones. The set $\mathcal{Z}^+ = \{(D, d, \delta, x) \in \mathcal{Z} : D = 0\}$ is given by

$$\mathcal{Z}^+ = \left\{ (D, d, \delta, x) : \exists \lambda_\pi, u_b, t, u_x : \begin{array}{l} P_x x + Q_x u_x - \sigma_x \in \mathbf{K}_x \\ tp + P_b x + Q_b u_b - \sigma_b \in \mathbf{K}_b \\ P_\pi^T \lambda_\pi - (2A^T d + Rx + r) = 0 \\ \delta - \lambda_\pi^T \sigma_\pi - t \geq 0 \\ Q_\pi^T \lambda_\pi = 0, \lambda_\pi \in \mathbf{K}_\pi^* \end{array} \right\} \tag{4.3.15}$$

We see that in the well-structured case, the SAA problem (4.3.8) in this case becomes a conic program.

*4.3.3.2 Quadratic case ($k > 1$).*

We consider the case when we want to recover the image $B(\pi^*)$, where $B(\cdot)$ is a vector of quadratic mappings. Let us set

$$\Phi(x, \pi) = x^T x - 2x^T B(\pi),$$

and let $X$ contain the image of $\Pi$ under the mapping $B(\cdot)$. In this situation, $(\mathcal{P}[\pi^*])$ is, essentially, the problem of the best, in $\|\cdot\|_2$, recovery of $B(\pi^*)$ via observations $\omega_t = A\pi^* + \eta_t$. The only optimal solution to $(\mathcal{P}[\pi^*])$ is exactly the quantity of interest $B(\pi^*)$, and $\min_{x \in X} \Phi(x, \pi^*) = 0$. We also have

$$\Phi(x, \pi^*) - \min_{x \in X} \Phi(x, \pi^*) = \|x - B(\pi^*)\|_2^2.$$

The $x$-components of the solutions obtained by processing $(\mathcal{S}[\pi^*])$ can be treated as estimates of $B(\pi^*)$. Restricting ourselves, same as above, to the family $\mathcal{F}$ of all quadratic functions on $\Omega = \mathbf{R}^p$ we end up with problem $(\mathcal{S}[\pi^*])$ and its SAA differing from (4.3.6), resp., (4.3.8) only in problems' domain $\mathcal{Z}$. Specifically, now we have

$$\nu(\pi^*) := \min_{(D,d,\delta,x)\in\mathcal{Z}} \mathbf{E}_{\omega\sim p_{\pi^*}}[\omega^T D\omega - 2\omega^T d + \delta] \tag{4.3.16}$$

$$\text{where } \mathcal{Z} := \left\{ \begin{array}{cl} & x \in X, \\ (D,d,\delta,x): & \pi^T A^T DA\pi + \mathrm{Tr}(DH) - 2\pi^T A^T d + \delta \\ & \qquad \geq x^T x - 2x^T B(\pi)\ \forall \pi \in \Pi \end{array} \right\} \tag{4.3.17}$$

***Exact Recovery and Consistency*** In our present situation, we still can show that under appropriate assumptions, solving (4.3.6) recovers the vector $B(\pi^*)$ we are looking for.

**Proposition 4.3.3.** *When the observation scheme (4.3.3) is given by an invertible $A$ and $B(\cdot)$ is a quadratic vector-valued mapping, the $x$-component of any optimal solution to (4.3.16) is equal to $B(\pi^*)$.*

*Proof.* Denote by $x^*$ an optimal solution to the problem $(\mathcal{P}[\pi^*])$. We know that $x^* = B(\pi^*)$ and $\Phi(x^*, \pi^*) = -\|B(\pi^*)\|_2^2$. W.l.o.g, let us assume $B(\cdot) = [B_1(\cdot), \ldots, B_n(\cdot)]$ where $n = dim(X)$, and

$$B_i(\pi) = \frac{1}{2}\pi^T R_i \pi + b_i^T \pi + c_i,\ 1 \leq i \leq n.$$

Set

$$\begin{array}{rcl} D^* & = & -A^{-T}(\sum_{i=1}^n x_i^* R_i)A^{-1}, \\ d^* & = & A^{-T}\sum_{i=1}^n x_i^* b_i, \\ \delta^* & = & (x^*)^T x^* - 2\sum_{i=1}^n x_i^* c_i - \mathrm{Tr}(D^* H). \end{array}$$

We can immediately see that $(D^*, d^*, \delta^*, x^*) \in \mathcal{Z}$ is a feasible solution to (4.3.16). Moreover,

$$\mathbf{E}_{\omega\sim p_{\pi^*}}[\omega^T D^*\omega - 2\omega^T d^* + \delta^*] = (\pi^*)^T A^T D^* A\pi^* + \mathrm{Tr}(D^* H) - 2(\pi^*)T A^T d^* + \delta^* = \Phi(x^*, \pi^*)$$

So $\nu(\pi^*) = \Phi(x^*, \pi^*)$ and $(D^*, d^*, \delta^*, x^*)$ is indeed an optimal solution to (4.3.16). Moreover, when $(\bar{D}, \bar{d}, \bar{\delta}, \bar{x})$ is an optimal solution to (4.3.16), we have

$$\Phi(x^*, \pi^*) = \mathbf{E}_{\omega\sim p_{\pi^*}}[\omega^T \bar{D}\omega - 2\omega^T \bar{d} + \bar{\delta}] \geq \bar{x}^T\bar{x} - 2\bar{x}^T B(\pi^*)$$

where the first equality is due to optimality and the second inequality is due to feasibility. Hence, $\bar{x} = B(\pi^*)$. □

**Remark.** When $B(\cdot)$ is affine, the above proposition still holds if $\mathcal{F}$ is set to be the family of linear functions of $\omega$.

**Tractability** The set $\mathcal{Z}$ clearly is convex, but not necessarily is computationally tractable. However, taking into account that $B(\cdot)$ is quadratic, in many cases, we indeed have at our disposal a computationally tractable convex subset $\mathcal{Z}^+$ of $\mathcal{Z}$, and we can associate problem (4.3.6) with $\mathcal{Z}^+$ in the role of $\mathcal{Z}$. Here are examples:

A. $\Pi$ *is given by a single strictly feasible quadratic inequality;*

In this case, the set $\mathcal{Z}$ is computationally tractable.

B. $\Pi$ *is a computationally tractable convex set and* $B(\cdot)$ *is affine*

In this case, the set $\mathcal{Z}^+ = \{(D, d, \delta, x) \in \mathcal{Z} : D \succeq 0\}$ is computationally tractable.

C. $\Pi$ *is given by a system of quadratic in* $\pi$ *inequalities* $\mathcal{S}_j(\pi) \leq 0$, $1 \leq j \leq J$.

In this case, the set

$$\mathcal{Z}^+ = \left\{(D, d, \delta, x) : \exists\{\lambda_i \geq 0\} : \begin{array}{l} x \in X, \\ \pi^T A^T D A \pi - 2\pi^T A^T d - 2x^T B(\pi) + \sum_j \lambda_j S_j(\pi) \\ \quad + [\mathrm{Tr}(DH) + \delta] - x^T x \geq 0, \ \forall \pi \in \mathbf{R}^q \end{array} \right\} \tag{4.3.18}$$

is computationally tractable and the semi-infinite constraint in the description of $\mathcal{Z}^+$ reduces to a Linear Matrix Inequality in variables $D, d, \delta, x, \{\lambda_i\}$.

D. *Well-structured case where* $\Pi$ *and* $X$ *are conic representable and* $B(\cdot)$ *is affine*

In this case, the set $\mathcal{Z}^+ = \{(D, d, \delta, x) \in \mathcal{Z} : D = 0\}$ is given by a system of conic constraints.

#### *4.3.3.3 Special case*

Consider the situation where we want to estimate the value $B\pi^*$ of an affine mapping $B(\cdot)$ via a *single* observation $\omega = A\pi^* + \eta$ where $\pi^* \in \Pi$, and $\Pi$ is a computationally tractable convex set. To save notation, we take $A = I$ (when $\mathrm{Ker}A = \{0\}$, the general case can be

reduced to this special one by redefining $\Pi$ and $B$); as explained above, we restrict ourselves with quadratic functions

$$f(\omega) = \omega^T D\omega - 2d^T\omega + \delta$$

Denoting by $H$ the covariance matrix of $\eta$, the semi-infinite constraints in $\mathcal{Z}$ reduces to

$$\pi^T D\pi - 2d^T\pi + \delta + \mathrm{Tr}(DH) \geq x^T x - 2x^T B\pi \geq 0 \ \forall \pi \in \Pi,$$

that is,

$$\delta \geq x^T x - \mathrm{Tr}(DH) + \max_{\pi\in\Pi} \big[2[d^T\pi - x^T B\pi] - \pi^T D\pi\big]. \tag{4.3.19}$$

The single-observation Sample Average Approximation of $(\mathcal{S}[\pi^*])$ reads

$$\min_{(D,d,\delta,x)\in Z} \big\{\omega^T D\omega - 2d^T\omega + \delta\big\} \tag{4.3.20}$$

where $Z$ is a convex subset of the domain specified by (4.3.19). Let us set $Z = Z_{\rho,r,Q}$ with

$$Z_{\rho,r,Q} = \left\{(D,d,\delta,x): \begin{array}{l} \exists (e, \|e\|_2 \leq \rho) : d = Q^T e,\ 0 \preceq D,\ \mathrm{Tr}(DH) \leq r \\ \delta \geq x^T x - \mathrm{Tr}(DH) + \max\limits_{\pi\in\Pi} \big[2[d^T\pi - x^T B\pi] - \pi^T D\pi\big] \end{array}\right\},$$

where $Q$ is some nonsingular matrix and $\rho, r \geq 0$. In this case (4.3.20) reads

$$\begin{aligned}
&\min_{D,e,x} \left\{ \max_{\pi\in\Pi} \big[-2(Q^T e)^T\omega + x^T x - \mathrm{Tr}(DH) + 2(Q^T e)^T\pi - 2x^T B\pi + \omega^T D\omega - \pi^T D\pi\big] : \right. \\
&\qquad\qquad \left. \|e\|_2 \leq \rho, 0 \preceq D, \mathrm{Tr}(DH) \leq r \right\} \\
&= \max_{\pi\in\Pi} \left\{ \min_{D,e,x} \left[ 2e^T Q[\pi - \omega] + x^T x - 2x^T B\pi + \omega^T D\omega - \pi^T D\pi - \mathrm{Tr}(DH) : \right. \right. \\
&\qquad\qquad \left. \left. \|e\|_2 \leq \rho, 0 \preceq D, \mathrm{Tr}(DH) \leq r \right] \right\} \\
&= \max_{\pi\in\Pi} \left\{ -2\rho\|Q(\pi - \omega)\|_2 - \pi^T B^T B\pi \right. \\
&\qquad\qquad \left. + \min_D \big\{\omega^T D\omega - \pi^T D\pi - \mathrm{Tr}(DH) : 0 \preceq D, \mathrm{Tr}(DH) \leq r\big\} \right\}.
\end{aligned}$$

Passing in the inner minimization problem from variable $D$ to the variable $E = H^{1/2} D H^{1/2}$, this problem becomes

$$\min_E \big\{\mathrm{Tr}\big(E \underbrace{[H^{-1/2}\omega\omega^T H^{-1/2} - H^{-1/2}\pi\pi^T H^{-1/2} - I]}_{W(\pi)}\big) : 0 \preceq E, \mathrm{Tr}(E) \leq r\big\} \tag{4.3.21}$$

Assuming $\dim\omega > 2$, the matrix $W(\pi)$ has negative minimal eigenvalue, and therefore the optimal value in (4.3.21) is $r\lambda_{\min}(W(\pi))$, where $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ are the minimal and

the maximal eigenvalues of a symmetric matrix. From our computation it follows that with $Z = Z_{\rho,r,Q}$, the $x$-component of the optimal solution to (4.3.20) (this is the only entity we are actually interested in – this is the estimate of $B\pi^*$ yielded by our construction) can be found as follows:

*Given $\omega = \pi^* + \eta$, we find*

$$\pi[\omega] = \operatorname*{argmin}_{\pi\in\Pi} \left\{ \pi^T B^T B\pi + 2\rho\|Q(\pi-\omega)\|_2 + r\left[\lambda_{\max}(H^{-1/2}[\pi\pi^T - \omega\omega^T]H^{-1/2}) + 1\right]\right\}$$

*and take*

$$x = x[\omega] = B\pi[\omega]$$

*as our estimate of $B\pi^*$.*

Note that with $Q$, $r$ fixed and as $\rho \to \infty$, $\pi[\omega]$ converges to

$$\pi_\infty(\omega) = \operatorname*{argmin}_{\pi\in\Pi} \|Q(\pi-\omega)\|_2,$$

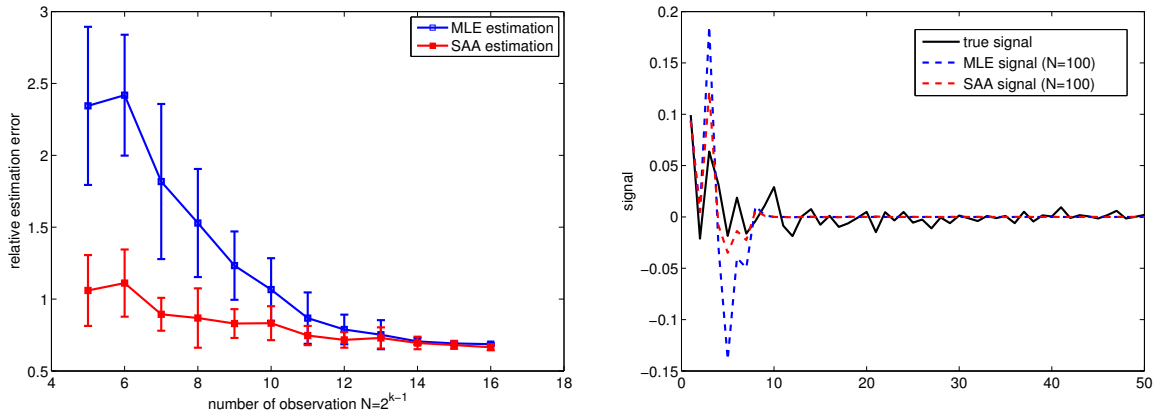
and the limiting, $\rho \to \infty$, estimate of $B\pi^*$ is as follows: we build the estimate

$$\widehat{\pi}(\omega) = \operatorname*{argmin}_{\pi\in\Pi} \|Q(\omega-\pi)\|_2$$

of $\pi^*$ and take $B\widehat{\pi}(\omega)$ as the estimate of $B\pi^*$. We see that at least the limiting, $\rho \to \infty$, case of our estimate is not completely senseless. In actual implementation, the parameters $\rho$, $r$ and $Q$ of our construction can be selected experimentally.

**Numerical illustration.** We run a simple experiment to illustrate how the approach works on the situation just described. In the experiment, we first generate a random signal $\pi^* \in \Pi = \{\pi \in \mathbf{R}^d : \pi^T\pi \le 1\}$. We then generate $N$ observations $\omega_j = A\pi^* + \eta_j, j = 1,\dots,N$, where $A = \mathrm{Diag}(1^{-\alpha}, 2^{-\alpha}, \dots, d^{-\alpha})$ with $\alpha = 5$, and $\{\eta_j\}$ are i.i.d. sampled from normal distribution $\mathcal{N}(0, \sigma^2 I)$ with $\sigma = 0.2$. Our goal is to estimate $B\pi^*$, where $B = \mathrm{Diag}(1^{-\beta}, 2^{-\beta}, \dots, d^{-\beta})$ with $\beta = 1$.

The first option to estimate $B\pi^*$ is to use the maximum likelihood estimator. Let $\bar{\omega}$ denote the sample mean $\frac{1}{N}\sum_{j=1}^N \omega_j$, so that $\bar{\omega} \sim \mathcal{N}(A\pi^*, \frac{\sigma^2}{N} I)$. Finding the maximum




(a) Relative error

(b) Recovery with 100 observations

Figure 11: Comparison between MLE and SAA methods

likelihood estimator $\widehat{\pi}_{\text{MLE}}$ of $\pi^*$ reduces to solving the optimization problem,

$$\widehat{\pi}_{\text{MLE}} := \underset{\pi \in \mathbf{R}^d : \pi^T \pi \leq 1}{\operatorname{argmin}} \|A\pi - \bar{\omega}\|_2^2. \tag{4.3.22}$$

Therefore, we can estimate $B\pi^*$ by $B\widehat{\pi}_{\text{MLE}}$.

Instead, our approach estimates $B\pi^*$ by solving Sample Average Approximation as follows:

$$\begin{aligned} \min_{x,t,D,d,\delta,\lambda} \quad & \text{Tr}(DW) - 2\bar{w}^T d + \delta \\ \text{s.t.} \quad & \begin{bmatrix} A^T D A + \lambda I & -A^T d + B^T y \\ -d^T A + y^T B & \delta + \sigma^2 \text{Tr}(D) - t - \lambda \end{bmatrix} \succeq 0 \\ & t \geq x^T x \\ & D \succeq 0, \lambda \geq 0 \\ & \text{Tr}(D) \leq M_1 \\ & \|d\|_2 \leq M_2 \end{aligned} \tag{4.3.23}$$

where $W := \frac{1}{N} \sum_{j=1}^N \omega_j \omega_j^T$, and the bounds $M_1, M_2$ are selected experimentally. The optimal solution $x$ serves as our estimate of $B\pi^*$.

In our experiments, we solve the two optimization problems (4.3.22) and (4.3.23) using CVX toolbox [35] in Matlab. We repeat the experiments for 10 instances and report in Figure 11a the averages of the relative estimation error, i.e. $\|x - B\pi^*\|_2 / \|B\pi^*\|_2$, along with the variances when the number of observation increases from $2^4$ to $2^{15}$. One can see

that the relative error of the estimator obtained by our SAA approach is much smaller than that of the MLE estimator, and the discrepancy is more significant when the number of observations is limited.

### 4.3.4 Application II: Indirect Support Vector Machines

Assume we observe i.i.d. pairs

$$\omega_t = (s_t, \xi_t + \eta_t) \in \mathbf{R}^p,$$

where $(s_t, \xi_t)$ are sampled from unknown Borel probability distribution $\pi^*$ on $\{1, -1\} \times \mathbf{R}^{p-1}$, and $\eta_t$ are independent of $(s_t, \xi_t)$ and are sampled from a partially known distribution $Q$ known to have zero mean and given covariance matrix $H$. Our goal is to solve the SVM-type Stochastic Programming problem

$$\min_{x=[u;\gamma]\in X} \mathbf{E}_{[s;\xi]\sim\pi^*}\{\max[1 - s[u^T\xi + \gamma], 0]\} \tag{SVM}$$

where $X \subset \mathbf{R}^s \times \mathbf{R}$ is a given convex set.

Assume that we know in advance that with $[s;\xi] \sim \pi^*$, the marginal distribution of $\xi$ is supported on a given compact subset $\Xi$ of $\mathbf{R}^{p-1}$. Specifying $\Pi$ as the set of all probability distributions on $\{-1, 1\} \times \Xi$ and setting

$$\Phi(x = [u;\gamma], \pi) = \mathbf{E}_{[s;\xi]\sim\pi}\{\max[1 - s[u^T\xi + \gamma], 0]\}$$

(SVM) takes the form of $(\mathcal{P}[\pi^*])$; note that in the case in question $P_\pi$ is the distribution of $[s;\xi+\eta]$ induced by $[s;\xi] \sim \pi \in \Pi$ and $\eta \sim Q$ independent of $(s,\xi)$.

By reasons to be explained below, we intend to use the setup

$$Y = \left\{ y = \left\{ \begin{array}{c} D_s \in \mathbf{S}^{p-1}, d_s \in \mathbf{R}^{p-1}, \delta_s \in \mathbf{R}, \{\mu_{si} \geq 0\}_{i\in I}, \\ \alpha_s \in \mathbf{R}^{p-1}, \beta_s \in \mathbf{R}^{p-1}, a_s \in \mathbf{R}, b_s \in \mathbf{R} \end{array} \right\}_{s=\pm 1} \right\} \tag{a}$$

$$f(y, \omega = [s;\zeta]) = \zeta^T D_s \zeta - 2d_s^T\zeta + \delta_s + \textstyle\sum_i \mu_{si} \exp\{\chi_i^T\zeta\} + \max[a_s + \alpha_s^T\zeta, b_s + \beta_s^T\zeta] \tag{b}$$

(4.3.24)

where $I$ is a given finite set and $\chi_i \in \mathbf{R}^{p-1}$, $i \in I$, are given vectors. Assume that we know a function $a(z) \geq 1$ such that

$$\mathbf{E}_{\eta\sim Q}\{\exp\{z^T\eta\}\} \geq a(z)\ \forall z$$

Then for $f$ given by (4.3.24.$b$), the conditional, $[s;\xi]$ given, expectation of $f(y,[s;\xi+\eta])$ satisfies

$$\begin{aligned}&\mathbf{E}_{\eta\sim Q}\{f(y,[s;\xi+\eta])\}\\&\geq \xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + \textstyle\sum_i \mu_{si}a(\chi_i)\exp\{\chi_i^T\xi\} + \max[a_s+\alpha_s^T\xi, b_s+\beta_s^T\xi]\end{aligned}$$

(since $\mathbf{E}_\eta\{\max[a_s+\alpha_s^T[\xi+\eta], b_s+\beta_s^T[\xi+\eta]]\} \geq \max[a_s+\alpha_s^T\xi, b_s+\beta_s^T\xi]$ by Jensen's inequality and due to the fact that $\eta$ is with zero mean). It follows that in order to ensure (4.3.2), it suffices to impose on the collection $y$ described in (4.3.24.$a$) and $x=[u;\gamma]$ the constraint $(x,y)\in\widetilde{\mathcal{Z}}$, with $\widetilde{\mathcal{Z}}$ given by

$$\begin{aligned}\widetilde{\mathcal{Z}} \;=\; \Big\{&(x=[u;\gamma], y=\{D_s,d_s,\delta_s,\{\mu_{si}\geq 0\}_{i\in I},\alpha_s,\beta_s,a_s,b_s\}_{s=\pm1}):\\&\forall(\xi\in\Xi, s=\pm1): \left\{\begin{array}{r}\xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + \sum_i \mu_{si}a(\chi_i)\exp\{\chi_i^T\xi\}\\ + \max[a_s+\alpha_s^T\xi, b_s+\beta_s^T\xi] \geq 1 - s[u^T\xi+\gamma]\\ \xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s\\ + \sum_i \mu_{si}a(\chi_i)\exp\{\chi_i^T\xi\} + \max[a_s+\alpha_s^T\xi, b_s+\beta_s^T\xi] \geq 0\end{array}\right\}\end{aligned} \tag{4.3.25}$$

Unfortunately, $\widetilde{\mathcal{Z}}$ hardly is convex (since its cross-section by a plane where all the variables except for $\alpha_s,a_s,\beta_s,b_s$ are fixed seems to be a nonconvex set in the space of $(\alpha_s,\beta_s,a_s,b_s)$). We, however, can build an inner convex approximation $\mathcal{Z}$ of $\widetilde{\mathcal{Z}}$, specifically,

$$\begin{aligned}\mathcal{Z} \;=\; \Big\{&(x=[u;\gamma], y=\{D_s,d_s,\delta_s,\{\mu_{si}\geq 0\}_{i\in I},\alpha_s,\beta_s,a_s,b_s\}_{s=\pm1}):\\&\forall(\xi\in\Xi, s=\pm1): \left\{\begin{array}{r}\xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + \sum_i \mu_{si}a(\chi_i)\exp\{\chi_i^T\xi\}\\ + [a_s+\alpha_s^T\xi] \geq 1 - s[u^T\xi+\gamma]\\ \xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + \sum_i \mu_{si}a(\chi_i)\exp\{\chi_i^T\xi\}\\ + [b_s+\beta_s^T\xi] \geq 0\end{array}\right\}\end{aligned} \tag{4.3.26}$$

Observe that $\mathcal{Z}$ is convex. Besides this,

- The set $\mathcal{Z}^+$ of all collections $(x,y)\in\mathcal{Z}$ with $D\succeq 0$ is computationally tractable, provided $\Xi$ is a computationally tractable convex compact set;
- When $I=\emptyset$, $\mathcal{Z}$ is computationally tractable, provided $\Xi$ is given by a single strictly feasible quadratic inequality, and

- $\mathcal{Z}$ admits a computationally tractable convex inner approximation $\mathcal{Z}^+$, provided $\Xi$ is a computationally tractable convex set given by a system of quadratic inequalities $S_j(\xi) \le 0$, $1 \le j \le J$. The approximation is

$$\begin{array}{rcl}
\mathcal{Z}^+ & = & \Big\{(x=[u;\gamma], y=\{D_s,d_s,\delta_s,\{\mu_{si}\ge 0\}_{i\in I}\}_{s=\pm1},\alpha_s,\beta_s,a_s,b_s\}_{s=\pm1}) : \\
& & \qquad \exists \left\{ \begin{array}{c} p_s\in\mathbf{R}^{p-1}, q_s\in\mathbf{R}^{p-1}, c_s\in\mathbf{R}, d_s\in\mathbf{R}, \\ \{\lambda_{sj}\ge 0,\nu_{sj}\ge 0\}_{j=1}^J \end{array} \right\}_{s=\pm1} : \\
& & \quad \forall (s=\pm1, \xi\in\mathbf{R}^{p-1}) : \\
& & \qquad \left.\begin{array}{l}
\xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + [a_s+\alpha_s^T\xi] + s[u^T\xi+\gamma] - 1 \\
\qquad\qquad + \sum_j \lambda_{sj} S_j(\xi) \ge p_s^T\xi + c_s, \\
\sum_i \lambda_{si} a(\chi_i)\exp\{\chi_i^T\xi\} + p_s^T\xi + c_s \ge 0, \\
\xi^T D_s\xi + \mathrm{Tr}(D_sH) - 2d_s^T\xi + \delta_s + [b_s+\beta_s^T\xi] \\
\qquad\qquad + \sum_j \nu_{sj} S_j(\xi) \ge q_s^T\xi + d_s, \\
\sum_i \lambda_{si} a(\chi_i)\exp\{\chi_i^T\xi\} + q_s^T\xi + d_s \ge 0.
\end{array} \right\}
\end{array}$$

  (note that all semi-infinite constraints here are efficiently verifiable).

**Comment.** Note that for every convex function $f([s;\zeta])$ we have

$$\mathbf{E}_\eta\{f([s;\xi+\eta])\} \ge f([s;\xi])$$

due to Jensen's inequality and the fact that $\eta$ is with zero mean. As a result, we have

$$\mathbf{E}_{([s;\xi],\eta)\sim P\times Q}\{f([s;\xi+\eta])\} \ge \mathbf{E}_{[s;\xi]\sim P}\{f([s,\xi])\},$$

so that the Stochastic Programming program

$$\min_{x=[u;\gamma]} \mathbf{E}_{([s;\xi],\eta)\sim P\times Q}\left\{\max[1-s[u^T(\xi+\eta)+\gamma],0]\right\}$$

which involves the expectation over our actual random observation is a safe approximation to the problem of interest (SVM). It is immediately seen that the safe approximation we have proposed in the main body of this section is less conservative than the one we have just outlined. The "added flexibility" stems from incorporating into the family $f(y,[s;\zeta])$, $s\in\{-1,1\}$, functions $\phi_s(\zeta)$ for which we can say something "substantial" about the relation between $\phi_s(\zeta)$ and $\mathbf{E}_{\eta\sim Q}\{\phi_s(\zeta+\eta)\}$, specifically, something more substantial than

what is said by Jensen's inequality in the case of convex $\phi_s$. The simplest examples here are quadratic functions and exponents $\exp\{\chi^T\zeta\}$, and this is what we use in (4.3.24) on the top of convex piecewise linear functions (on a closest inspection, just two pieces turn out to be enough). "Convexity considerations" do not forbid making the coefficients of the quadratic component of $f$ "variable," that is, part of the variable $y$. Unfortunately, these considerations prevent us from making the parameter(s) $\chi$ of the exponent(s) to be "variable" as well. Instead, we fix a collection of $\chi_i$'s and make variable the weights $\mu_{si}$'s of the exponents $\exp\{\chi_i^T\zeta\}$ in $f$. In actual implementation, the collection of $\chi_i$'s could be built incrementally: we start with the empty collection of exponents and solve the associated safe approximation of (SVM), thus ending up with some $x = [u; \gamma]$. We then make $\pm u$ the first pair of our $\chi$'s, get a solution $[u'; \gamma']$ to the new safe approximation of (SVM) and add the vectors $\pm u'$ to our collection of $\chi$'s, and so on.

### 4.3.5 Concluding Remarks.

In this section, we have introduced the notion of Indirect Stochastic Programming problem, i.e. convex problem in the form, $\min_{x\in X} \Phi(x, \pi^*)$, where $\pi^*$ is unknown but admits indirect noisy observations sampled from some distribution $P_{\pi^*}(\cdot)$ parametrized by $\pi^*$. In contrast to the previous section, we make no structural assumptions on the function $\Phi$ or the distribution $P_{\pi^*}(\cdot)$. We propose a general approximation scheme amenable to algorithms such as Stochastic Approximation. We demonstrate on several examples that we can build safe and computationally tractable approximations of target problems and process them with SA or SAA efficiently. We also demonstrate experimentally, albeit at this point in time very preliminary, the practical potential of our approach as applied to the affine signal processing.

## *4.4 Final Comments and Future Work*

The outlined approaches and discussions make it clear that our effort towards error-in-measurement optimization is just the beginning. Time limitations imposed on our research for this part prevent us from extensive and in-depth investigation of the numerous challenges arising here, we consider these challenges as a subject of future research where we intend

to address the issues as follows.

**Statistical behavior of estimators.** Since our approximation scheme always renders a convex stochastic programming problem, the optimization error is more or less well understood. It remains interesting to analyze the statistical error of the estimator yielded by the outlined approach. We have shown that consistency does take place in several cases in the affine signal processing. However, in general setups, especially in high-dimensional regime, it is well-known that consistent estimation when number of observations is far less than the dimensional of unknown signal, is nearly impossible unless additional structure such as sparsity of the signal is postulated. Hence, it remains interesting to incorporate sparsity into our framework into our framework and develop consistency results for more general setups, and/or to come up with reasonable non-asymptotic bounds on the statistical error. In [53], the authors show that in the context of high-dimensional sparse linear regression with corrupted data, the statistical error of the estimator obtained from some nonconvex optimization enjoys the same scaling as the minimax rates for the classical cases of perfectly observed and independently sampled observations. It would be interesting to understand whether our approach, based on convex programming and thus "computationally friendly," approach, could achieve similar results.

**Extension to variational inequalities and discrete time dynamic programming.** While the problems we focus on so far are in the form of convex minimization problems, $\min_{x\in X} \Phi(x, \pi^*)$, it makes sense to extend this to other problems with convex structure, e.g. variational inequalities,

$$\text{Find } x_* \in X : \ \langle F(x, \pi^*), x - x_* \rangle \geq 0, \forall x \in X,$$

where $\pi^*$ is unknown but admits observations sampled from some distribution $P_{\pi^*}$. Model uncertainty has been a major concern in many recent studies in stochastic modeling, e.g. portfolio selection with discrete decision epochs, and inventory control problems. It would be interesting to extend our framework further to cover dynamic settings as well. It would also be interesting to see connections with and comparisons to (distributionally) robust

optimization as discussed in a number of papers, e.g. [51, 29, 84].

**Privacy learning, and other practical applications.** The most natural application of the indirect stochastic programming, is perhaps, privacy learning, which has received enormous attention in the past decade (see, e.g., [1, 82] and references therein). Due to privacy considerations, data are artificially corrupted before becoming available for processing. As a result, inferences from the available data become in many cases an indirect stochastic programming problem. In fact, in a wide spectrum of real-world applications, such as medical tests, remote sensing, bioinformatics, chemical process engineering, data are usually subject to measurement errors due to intrinsic physical limitations, prohibitive costs or hard constraints. In our future study, we would like to investigate such real application-driven examples and investigate on these examples the applied potential of the approaches we have proposed.

# REFERENCES


[1] Agrawal, R. and Srikant, R., "Privacy-preserving data mining," in *ACM Sigmod Record*, vol. 29, pp. 439–450, ACM, 2000.

[2] Ahmadi, H. and Shanbhag, U. V., "Data-driven first-order methods for misspecified convex optimization problems: Global convergence and rate estimates," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 4228–4233, IEEE, 2014.

[3] Andersen, E. D. and Andersen, K. D.

[4] Aujol, J.-F. and Chambolle, A., "Dual norms and image decomposition models," *International Journal of Computer Vision*, vol. 63, no. 1, pp. 85–104, 2005.

[5] Bach, F., "Duality between subgradient and conditional gradient methods," *SIAM Journal on Optimization*, 2015.

[6] Beck, A. and Teboulle, M., "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[7] Becker, S., Bobin, J., and Candès, E. J., "Nesta: a fast and accurate first-order method for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.

[8] Ben-Tal, A., El Ghaoui, L., and Nemirovski, A., *Robust optimization*. Princeton University Press, 2009.

[9] Ben-Tal, A., Margalit, T., and Nemirovski, A., "The ordered subsets mirror descent optimization method with applications to tomography," *SIAM Journal on Optimization*, vol. 12, no. 1, pp. 79–108, 2001.

[10] Bertsekas, D. P., *Convex Optimization Algorithms*. Athena Scientific, 2015.

[11] Bertsimas, D., Brown, D. B., and Caramanis, C., "Theory and applications of robust optimization," *SIAM review*, vol. 53, no. 3, pp. 464–501, 2011.

[12] Bertsimas, D., Gupta, V., and Kallus, N., "Data-driven robust optimization," *arXiv preprint arXiv:1401.0212*, 2013.

[13] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 122–122, 2010.

[14] Buades, A., Coll, B., and Morel, J.-M., "A review of image denoising algorithms, with a new one," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[15] Buldygin, V., *Metric characterization of random variables and random processes*.

[16] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C., "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.

[17] Candés, E. J., Li, X., Ma, Y., and Wright, J., "Robust principal component analysis?," *Journal of the ACM (JACM)*, vol. 58, no. 3, p. 11, 2011.

[18] Cao, Y. and Xie, Y., "Low-rank matrix recovery in poisson noise," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*, pp. 384–388, IEEE, 2014.

[19] Carlavan, M. and Blanc-Féraud, L., "Sparse poisson noisy image deblurring," *Image Processing, IEEE Transactions on*, vol. 21, no. 4, pp. 1834–1846, 2012.

[20] Carroll, R. J., Ruppert, D., Stefanski, L. A., and Crainiceanu, C. M., *Measurement error in nonlinear models: a modern perspective*. CRC press, 2006.

[21] Cesa-Bianchi, N., Shwartz, S. S., and Shamir, O., "Online learning of noisy data with kernels," *COLT 2010*, p. 218, 2010.

[22] Chambolle, A. and Pock, T., "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[23] Chambolle, A. and Pock, T., "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[24] Chen, G. and Teboulle, M., "Convergence analysis of a proximal-like minimization algorithm using bregman functions," *SIAM Journal on Optimization*, vol. 3, no. 3, pp. 538–543, 1993.

[25] Chen, Y., Lan, G., and Ouyang, Y., "Optimal primal-dual methods for a class of saddle point problems," *SIAM Journal on Optimization*, vol. 24, no. 4, pp. 1779–1814, 2014.

[26] Cox, B., Juditsky, A., and Nemirovski, A., "Dual subgradient algorithms for large-scale nonsmooth learning problems," *Mathematical Programming*, pp. 1–38, 2013.

[27] Dai, B., He, N., Dai, H., and Song, L., "Scalable bayesian inference via particle mirror descent," *arXiv preprint arXiv:1506.03101*, 2015.

[28] Dai, B., Xie, B., He, N., Liang, Y., Raj, A., Balcan, M.-F. F., and Song, L., "Scalable kernel methods via doubly stochastic gradients," in *Advances in Neural Information Processing Systems*, pp. 3041–3049, 2014.

[29] Delage, E. and Ye, Y., "Distributionally robust optimization under moment uncertainty with application to data-driven problems," *Operations research*, vol. 58, no. 3, pp. 595–612, 2010.

[30] Deng, W., Lai, M.-J., Peng, Z., and Yin, W., "Parallel multi-block admm with o (1/k) convergence," 2013.

[31] Du, N., Wang, Y., He, N., and Song, L., "Time-sensitive recommendation from recurrent user activities," *Neural Information Processing Systems*, 2015.

[32] Dudik, M., Harchaoui, Z., and Malick, J., "Lifted coordinate descent for learning with trace-norm regularization," *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012.

[33] Goldfarb, D. and Ma, S., "Fast multiple-splitting algorithms for convex optimization," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 533–556, 2012.

[34] Goldfarb, D., Ma, S., and Scheinberg, K., "Fast alternating linearization methods for minimizing the sum of two convex functions," *Mathematical Programming*, vol. 141, no. 1-2, pp. 349–382, 2013.

[35] Grant, M., Boyd, S., and Ye, Y., "Cvx: Matlab software for disciplined convex programming," 2008.

[36] Harchaoui, Z., Juditsky, A., and Nemirovski, A., "Conditional gradient algorithms for norm-regularized smooth convex optimization," *Mathematical Programming*, pp. 1–38, 2013.

[37] Harmany, Z. T., Marcia, R. F., and Willett, R. M., "This is spiral-tap: sparse poisson intensity reconstruction algorithmstheory and practice," *Image Processing, IEEE Transactions on*, vol. 21, no. 3, pp. 1084–1096, 2012.

[38] He, N. and Harchaoui, Z., "Semi-proximal mirror-prox for nonsmooth composite minimization," *Neural Information Processing Systems (NIPS)*.

[39] He, N., Juditsky, A., and Nemirovski, A., "Mirror prox algorithm for multi-term composite minimization and semi-separable problems," *Computational Optimization and Applications*, vol. 61, no. 2, pp. 275–319, 2015.

[40] Hong, M., Chang, T.-H., Wang, X., Razaviyayn, M., Ma, S., and Luo, Z.-Q., "A block successive upper bound minimization method of multipliers for linearly constrained convex optimization," *arXiv preprint arXiv:1401.7079*, 2014.

[41] Jaggi, M., "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *ICML*, pp. 427–435, 2013.

[42] Jiang, H. and Shanbhag, U. V., "On the solution of stochastic optimization and variational problems in imperfect information regimes," *arXiv preprint arXiv:1402.1457*, 2014.

[43] Juditsky, A. and Nemirovski, A., "First-order methods for nonsmooth largescale convex minimization: I general purpose methods; ii utilizing problems structure," in *Optimization for Machine Learning* (Sra, S., Nowozin, S., and Wright, S., eds.), pp. 121–183, The MIT Press, 2011.

[44] Juditsky, A. and Nemirovski, A., "Solving variational inequalities with monotone operators on domains given by linear minimization oracles," *arXiv preprint arXiv:1312.107*, 2013.

[45] Juditsky, A., Nemirovski, A., Tauvel, C., and others, "Solving variational inequalities with stochastic mirror-prox algorithm," *Stochastic Systems*, vol. 1, no. 1, pp. 17–58, 2011.

[46] Juditsky, A. and Nemirovski, A. S., "Large deviations of vector-valued martingales in 2-smooth normed spaces," *arXiv preprint arXiv:0809.0813*, 2008.

[47] Lan, G., "An optimal method for stochastic composite optimization," *Mathematical Programming*, vol. 133, no. 1-2, pp. 365–397, 2012.

[48] Lan, G., "The complexity of large-scale convex programming under a linear optimization oracle," *arXiv*, 2013.

[49] Lan, G. and Zhou, Y., "Conditional gradient sliding for convex optimization," *arXiv*, 2014.

[50] Lemarchal, C., Nemirovskii, A., and Nesterov, Y., "New variants of bundle methods," *Mathematical Programming*, vol. 69, no. 1-3, pp. 111–147, 1995.

[51] Lim, A. E., Shanthikumar, J. G., and Shen, Z. M., "Model uncertainty, robust optimization, and learning," *Tutorials in Operations Research: Models, Methods, and Applications for Innovative Decision Making*, pp. 66–94, 2006.

[52] Little, R. J. and Rubin, D. B., *Statistical analysis with missing data*. John Wiley & Sons, 2014.

[53] Loh, P.-L. and Wainwright, M. J., "High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity," in *Advances in Neural Information Processing Systems*, pp. 2726–2734, 2011.

[54] Monteiro, R. D. and Svaiter, B. F., "Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers," *SIAM Journal on Optimization*, vol. 23, no. 1, pp. 475–507, 2013.

[55] Mu, C., Zhang, Y., Wright, J., and Goldfarb, D., "Scalable robust matrix recovery: Frank-wolfe meets proximal methods," *arXiv preprint arXiv:1403.7588*, 2014.

[56] Nemirovski, A., "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.

[57] Nemirovski, A., "Regular banach spaces and large deviations of random sums," *Paper in progress, E-print: http://www2. isye. gatech. edu/nemirovs*, vol. 528, 2004.

[58] Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A., "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[59] Nemirovski, A., Onn, S., and Rothblum, U. G., "Accuracy certificates for computational problems with convex structure," *Mathematics of Operations Research*, vol. 35, no. 1, pp. 52–78, 2010.

[60] Nemirovski, A. and Rubinstein, R., "An efficient stochastic approximation algorithm for stochastic saddle point problems," in *Modeling Uncertainty and examination of stochastic theory, methods, and applications* (Dror, M., L'Ecuyer, P., and Szidarovszky, F., eds.), pp. 155–184, Kluwer Academic Publishers, 2002.

[61] Nemirovsky, A. S. and Yudin, D. B., "Problem complexity and method efficiency in optimization," *Wiley-Interscience Series in Discrete Mathematics*, 1983.

[62] Nesterov, Y., "Smooth minimization of non-smooth functions," *Mathematical programming*, vol. 103, no. 1, pp. 127–152, 2005.

[63] Nesterov, Y., "Gradient methods for minimizing composite functions," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, 2013.

[64] Nesterov, Y., "Gradient methods for minimizing composite objective function," *CORE*, 2007.

[65] Nesterov, Y., "Complexity bounds for primal-dual methods minimizing the model of objective function," tech. rep., Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2015.

[66] Orabona, F., Argyriou, A., and Srebro, N., "Prisma: Proximal iterative smoothing algorithm," *arXiv preprint arXiv:1206.2372*, 2012.

[67] Ouyang, H., He, N., Tran, L., and Gray, A., "Stochastic alternating direction method of multipliers," in *Proceedings of the 30th International Conference on Machine Learning*, pp. 80–88, 2013.

[68] Ouyang, Y., Chen, Y., Lan, G., and Pasiliao Jr, E., "An accelerated linearized alternating direction method of multipliers," *SIAM Journal on Imaging Sciences*, vol. 8, no. 1, pp. 644–681, 2015.

[69] Parikh, N. and Boyd, S., "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 123–231, 2013.

[70] Pierucci, F., Harchaoui, Z., and Malick, J., "A smoothing approach for composite conditional gradient with nonsmooth loss," in *Conférence dApprentissage Automatique–Actes CAP14*, 2014.

[71] Polyak, B. T., "New stochastic approximation type procedures," *Automat. i Telemekh*, vol. 7, no. 98-107, p. 2, 1990.

[72] Polyak, B. T. and Juditsky, A. B., "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.

[73] Qin, Z. and Goldfarb, D., "Structured sparsity via alternating direction methods," *The Journal of Machine Learning Research*, vol. 13, pp. 1373–1406, 2012.

[74] Robbins, H. and Monro, S., "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.

[75] Rockafellar, R., "Minimax theorems and conjugate saddle functions," tech. rep., DTIC Document, 1964.

[76] Scheinberg, K., Goldfarb, D., and Bai, X., "Fast first-order methods for composite convex optimization with backtracking," *Foundations of Computational Mathematics*, vol. 14, no. 3, pp. 389–417, 2014.

[77] Sra, S., Kim, D., and Schölkopf, B., "Non-monotonic poisson likelihood maximization," tech. rep., Tech. Rep. 170, Max Planck Institute for Biological Cybernetics, 2008.

[78] Tran-Dinh, Q., Kyrillidis, A., and Cevher, V., "Composite self-concordant minimization," *arXiv preprint arXiv:1308.2867*, 2013.

[79] Tseng, P., "Alternating projection-proximal methods for convex programming and variational inequalities," *SIAM Journal on Optimization*, vol. 7, no. 4, pp. 951–965, 1997.

[80] Tseng, P., "On accelerated proximal gradient methods for convex-concave optimization," *submitted to SIAM Journal on Optimization*, 2008.

[81] Wainwright, M. J., "Structured regularizers for high-dimensional problems: Statistical and computational issues," *Annual Review of Statistics and Its Application*, vol. 1, pp. 233–253, 2014.

[82] Wainwright, M. J., Jordan, M. I., and Duchi, J. C., "Privacy aware learning," in *Advances in Neural Information Processing Systems*, pp. 1430–1438, 2012.

[83] Wen, Z., Goldfarb, D., and Yin, W., "Alternating direction augmented lagrangian methods for semidefinite programming," *Mathematical Programming Computation*, vol. 2, no. 3-4, pp. 203–230, 2010.

[84] Wiesemann, W., Kuhn, D., and Sim, M., "Distributionally robust convex optimization," *Operations Research*, vol. 62, no. 6, pp. 1358–1376, 2014.

[85] Zhang, X., Yu, Y., and Schuurmans, D., "Accelerated training for matrix-norm regularization: A boosting approach," in *NIPS*, 2012.

# VITA

Niao He was born in Macheng, Hubei Province, China in 1990. She enrolled in the Special Class for Gifted Young from University of Science and Technology of China in 2006 and obtained her B.S. degree in Mathematics in 2010. Afterwards, she attended Georgia Institute of Technology from 2010 to 2015 graduating with a M.S. degree in Computational Science and Engineering and a Ph.D. degree in Operations Research. Starting from January 2016, she will join the Department of Industrial and Enterprise Systems Engineering at University of Illinois Urbana-Champaign as an Assistant Professor.