TECHNISCHE
UNIVERSITÄT
DARMSTADT

A Parallel Successive Convex Approximation Framework with Smoothing Majorization for Phase Retrieval

vom Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

Tianyi Liu (刘添翼), M.Sc.

Referent:	Prof. Dr.-Ing. Marius Pesavento
Korreferent:	Prof. Dr. Stefan Ulbrich
Tag der Einreichung:	03.07.2024

Darmstadt, 2024

Liu, Tianyi: *A Parallel Successive Convex Approximation Framework with Smoothing Majorization for Phase Retrieval*

Darmstadt, Technische Universität Darmstadt

Jahr der Veröffentlichung der Dissertation auf TUprints: 2024

URN: urn:nbn:de:tuda-tuprints-282016

Tag der mündlichen Prüfung: 26.09.2024

Erklärungen laut Promotionsordnung

§ 8 Abs. 1 lit. d PromO

Ich versichere hiermit, dass von mir zu keinem vorherigen Zeitpunkt bereits ein Promotionsversuch unternommen wurde. Andernfalls versichere ich, dass der promotionsführende Fachbereich über Zeitpunkt, Hochschule, Dissertationsthema und Ergebnis dieses Versuchs informiert ist.

§ 9 Abs. 1 PromO

Ich versichere hiermit, dass die vorliegende Dissertation, abgesehen von den in ihr ausdrücklich genannten Hilfsmitteln, selbstständig und nur unter Verwendung der angegebenen Quellen verfasst wurde. Weiterhin versichere ich, dass die "Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Technischen Universität Darmstadt" sowie die "Leitlinien zum Umgang mit digitalen Forschungsdaten an der TU Darmstadt" in den jeweils aktuellen Versionen bei der Verfassung der Dissertation beachtet wurden.

§ 9 Abs. 2 PromO

Ich versichere hiermit, dass die vorliegende Dissertation bisher noch nicht zu Prüfungszwecken gedient hat.

Darmstadt, 03.07.2024

Tianyi Liu, M.Sc.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Dr.-Ing. Marius Pesavento, for his continuous support, guidance, and encouragement throughout my doctoral journey. His invaluable feedback and insightful discussions have greatly contributed to the success of this research.

I am also immensely grateful to my collaborators Dr. Andreas M. Tillmann, Dr. Yang Yang, and Prof. Dr. Yonina C. Eldar for their valuable advice, constructive comments, and suggestions, which have significantly enhanced the quality of this work.

I would like to thank Prof. Dr. mont. Mario Kupnik, Prof. Dr. Stefan Ulbrich, Prof. Dr. rer. nat. Sebastian Schöps, and Prof. Dr.-Ing. Abdelhak M. Zoubir for their work as committee members of my doctoral defense.

For valuable discussions and support, I would like to thank my current and former colleagues in the Communication Systems Group: Florian Bahlke, Kaleb Debre, Yufan Fan, Ganapati Hegde, Gerta Kushe, Raphael Müller, Fabio Nikolay, David Schenck, Lukas Schynol, Christian Steffens, and Minh Trinh-Hoang. Their useful discussions and advice greatly improved my technical skills as a researcher. Special thanks go to Marlis Gorecki, Rojin Ulusoy, Qiulan Weng for their help in administrative and teaching-related issues.

My heartfelt thanks to my family for their unwavering love, patience, and encouragement. I owe my deepest gratitude to my parents and my girlfriend Tina for their constant support and understanding.

Lastly, I would like to acknowledge the contribution of all those who have directly or indirectly supported me in this endeavor. Your encouragement has been invaluable.

Kurzfassung

Diese Dissertation befasst sich mit dem Entwurf und der Analyse von approximationsbasierten Methoden für nichtkonvexe und nichtglatte Optimierungsprobleme. Die Hauptidee dieser Methoden besteht darin, ein schwieriges Optimierungsproblem zu lösen, indem es in eine Reihe von einfacheren Ersatz- bzw. approximativen Problemen umgewandelt wird. In den beiden weit verbreiteten Optimierungsrahmenwerken, nämlich dem Majorization-Minimization (MM) Rahmenwerk und dem sukzessiven konvexen Approximation (SCA) Rahmenwerk, wird die approximative Funktion so gestaltet, dass sie eine globale obere Schranke, genannt Majorisierer, der ursprünglichen Zielfunktion bzw. konvex ist. Allgemein gesprochen gibt es zwei Anforderungen an die approximative Funktion, nämlich die Nähe zur ursprünglichen Zielfunktion und die geringe Rechenkomplexität der Minimierung der approximativen Funktion. Insbesondere konzentrieren wir uns auf Techniken, die verwendet werden können, um ein parallelisierbares approximatives Problem zu konstruieren, um moderne Multicore-Computing-Plattformen zu nutzen.

Der erste Teil dieser Arbeit zielt darauf ab, einen effizienten parallelisierbaren algorithmischen Rahmen basierend auf Approximationstechniken für eine breite Klasse von nichtkonvexen und nichtglatten Optimierungsproblemen zu entwickeln. Das klassische Konvergenzergebnis von MM basiert auf der Konsistenz der Richtungsableitungen in allen Richtungen zwischen der ursprünglichen Zielfunktion und ihrem Majorisierer an dem Punkt, an dem der Majorisierer konstruiert wird. Diese Anforderung beschränkt den am nichtdifferenzierbaren Punkt der ursprünglichen Funktion konstruierten Majorisierer darauf, ebenfalls nicht glatt zu sein, was seine Fähigkeit zur Vereinfachung nichtglatter Probleme einschränkt, da die Minimierung der majorisierenden Funktion, wenn sie auf nicht glatt beschränkt ist, weiterhin schwierig sein kann. Daher lockern wir in dieser Arbeit die Konsistenz der Ableitung im Majorisierungsschritt, sodass ein glatter Majorisierer, der leicht minimiert werden kann, für eine breite Klasse von nichtglatten Problemen zugelassen wird. Als Ergebnis dieser Lockerung der Ableitungskonsistenz führt die glättende Majorisierung zur Konvergenz zu einer stationären Lösung in einem entspannteren Sinne als das klassische MM. Darüber hinaus ermöglicht uns die Glätte der majorisierenden Funktion im Gegensatz zur exakten Minimierung der möglicherweise nichtkonvexen majorisierenden Funktion die Anwendung der Idee von SCA sowie verfügbarer separierbarer Approximationstechniken, um einen approximativen Minimierer der majorisierenden Funktion effizient zu erhalten. Daher entwickeln wir einen parallelisierbaren inexakten MM-Rahmen, genannt *Smoothing SCA*, indem wir die glättende Majorisierungstechnik und die Idee der sukzessiven konvexen Approximation kombinieren. In diesem Rahmen wird die Konstruktion des approximativen

Problems bei jeder Iteration in zwei Schritte unterteilt, nämlich die glättende Majorisierung und die konvexe Approximation, sodass die beiden Anforderungen an die approximative Funktion getrennt behandelt werden können. Darüber hinaus können sowohl das exakte als auch das inexakte MM mit glättender Majorisierung blockkoordinatenweise implementiert werden, um potenzielle separierbare Strukturen der Einschränkungen im Optimierungsproblem auszunutzen. Das Konvergenzverhalten der vorgeschlagenen Rahmenwerke wird entsprechend analysiert.

Im zweiten Teil dieser Arbeit wird das von uns hauptsächlich propagierte Rahmenwerk, das glättende SCA-Rahmenwerk, verwendet, um das Phasenrückgewinnungsproblem mit Wörterbuchlernen anzugehen. Zwei effiziente Parallelalgorithmen werden entwickelt, indem das glättende SCA auf zwei komplementäre nichtkonvexe und nichtglatte Formulierungen angewendet wird, die beide auf einem Kriterium der kleinsten Quadrate basieren. Die Rechenkomplexitäten der vorgeschlagenen Algorithmen werden theoretisch analysiert und sowohl ihre Fehlerleistung als auch ihre Rechenzeit werden durch umfangreiche Simulationen im Kontext der blinden Kanalschätzung aus Subband-Amplitudenmessungen in einem Mehrantennen-Zugriffsnetzwerk im Vergleich zu den modernsten Methoden bewertet.

Abstract

This dissertation is concerned with the design and analysis of approximation-based methods for nonconvex nonsmooth optimization problems. The main idea behind those methods is to solve a difficult optimization problem by converting it into a sequence of simpler surrogate/approximate problems. In the two widely-used optimization frameworks, namely, the majorization-minimization (MM) framework and the successive convex approximation (SCA) framework, the approximate function is designed to be a global upper bound, called majorizer, of the original objective function and convex, respectively. Generally speaking, there are two desiderata of the approximate function, i.e., the tightness to the original objective function and the low computational complexity of minimizing the approximate function. In particular, we focus on techniques that can be used to construct a parallelizable approximate problem so as to take advantage of modern multicore computing platforms.

The first part of this thesis aims to develop an efficient parallelizable algorithmic framework based on approximation techniques for a broad class of nonconvex nonsmooth optimization problems. The classic convergence result of MM is established on the consistency of directional derivatives in all directions between the original objective function and its majorizer at the point where the majorizer is constructed. This requirement restricts the majorizer constructed at a nondifferentiable point of the original function to be also nonsmooth, which hinders its capability of simplifying nonsmooth problems since the minimization of the majorizing function, if restricted to be nonsmooth, may still be difficult. Therefore, in this thesis, we relax the derivative consistency in the majorization step so that a smooth majorizer that can be easily minimized is permitted for a wide class of nonsmooth problems. As a result of this relaxation of derivative consistency, the smoothing majorization leads to the convergence to a stationary solution in a more relaxed sense than the classic MM. Furthermore, in contrast to minimizing the possibly nonconvex majorizing function exactly, the smoothness of the majorizing function allows us to employ the idea of SCA, along with available separable approximation techniques, to obtain an approximate minimizer of the majorizing function efficiently. Thus, we develop a parallelizable inexact MM framework, termed *smoothing SCA*, by combining the smoothing majorization technique and the idea of successive convex approximation. In this framework, the construction of the approximate problem at each iteration is divided into two steps, namely, the smoothing majorization and the convex approximation, so that the two desiderata of the approximate function can be treated separately. In addition, both the exact and inexact MM with smoothing majorization can be implemented block-coordinatewise to exploit potential separable

structures of the constraints in the optimization problem. The convergence behaviors of the proposed frameworks are analyzed accordingly.

In the second part of this thesis, as our mainly promoted framework, the smoothing SCA framework is employed to address the phase retrieval with dictionary learning problem. Two efficient parallel algorithms are developed by applying the smoothing SCA to two complementary nonconvex nonsmooth formulations, respectively, which are both based on a least-squares criterion. The computational complexities of the proposed algorithms are theoretically analyzed and both their error performance and computational time are evaluated by extensive simulations in the context of blind channel estimation from subband magnitude measurements in multi-antenna random access network, in comparison to the state-of-the-art methods.

Contents

Chapter 1
Introduction

Consider a general constrained minimization problem:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \quad f(\boldsymbol{x}) \tag{1.1}$$

with the feasible set $\mathcal{X} \in \mathbb{R}^n$ and the objective function $f : \mathbb{R}^n \to \mathbb{R}$.

Assumption 1.1. *We make the following blanket assumptions throughout the thesis unless otherwise noted:*

1) $\mathcal{X} \subseteq \mathbb{R}^n$ *is closed, convex, and nonempty;*

2) f *is continuous and coercive on* $\mathcal{X}$*.*

The above assumptions are quite standard and are fulfilled by a wide range of practical problems. In particular, the continuity and coercivity of f ensures the existence of optimal solutions even if the feasible set $\mathcal{X}$ is not bounded [Ber16, Prop. A.8].

In this dissertation, we aim to develop an efficient parallelizable algorithmic framework for a broad class of nonconvex and nonsmooth optimization problems. Many practical applications in signal processing, communications, and machine learning result in nonconvex and nonsmooth optimization problems that can not be solved analytically but by an iterative procedure. Most of the effective iterative methods are developed based on approximation techniques, where the basic idea is to solve a difficult problem by converting it into a sequence of simpler surrogate/approximate problems. The existing approximation-based methods differ from each other mainly on the construction of the approximate problems. Generally speaking, there are two desiderata of the approximate function, namely, the tightness to the original objective function and the low computational complexity of minimizing the approximate function. In particular, we focus on the techniques that can be used to construct a parallelizable approximate problem so as to take advantage of modern multicore computing platforms.

One popular approximation-based algorithmic framework is the majorization-minimization (MM) framework [Lan16, SBP17, Ngu17, LWLZ21], where the approximate function is designed to be a global upper bound of the objective function and tangent to it at the current iterate, referred to as a majorizer of the original objective

function, so that the minimizer of the approximate function also provides a decrease of the original objective function. This procedure is simple but attractive since it generates a solution sequence that monotonically decreases the original objective function and, thus, provides the convergence to a stationary solution of the original problem under some derivative consistency assumption [RHL13, LWLZ21]. A comprehensive list of available majorization techniques can be found in the survey paper [SBP17] and the book [Lan16], which are mostly derived from the properties of convex functions. Many other algorithms can be viewed as special cases of MM generated with different majorization techniques, such as the well-known expectation-minimization (EM), proximal algorithms, concave-convex procedures (CCCP), and so on [SBP17].

Despite the attractive properties of MM, in practice, it may not be easy to construct a global upper bound for the objective function unless it possesses a certain convexity/concavity structure. An alternative approach that has also been widely used is the successive convex approximation (SCA) framework [YP17, SFL17a, SS18], where the approximate function is not required to be a global upper bound of the original objective function but a convex function. By relaxing the global bounding requirement, we can easily construct an approximate problem that can be solved at a lower computational cost, even in a parallel or distributed manner. In fact, this framework has been used to develop parallel and distributed algorithms for a wide range of practical applications that involve large-scale networked systems, e.g., information processing over networks, communication networks, sensor networks, data-based networks, and machine learning [SFL$^+$17b, SS18] However, the general convergence of SCA is only established for smooth functions or the class of structured nonsmooth functions where the nonsmooth component is convex.

The classic convergence result of MM is established on the directional differentiability of the objective function and the consistency of directional derivatives in all directions between the original objective function and its majorizer at the point where the majorizer is constructed. This requirement restricts the majorizer constructed at a nondifferentiable point of the original function to be also nonsmooth, which hinders its capability of simplifying nonsmooth problems since the minimization of the majorizing function, if restricted to be nonsmooth, may still be difficult. Therefore, in this thesis, we relax the derivative consistency in the majorization step so that a smooth majorizer that can be easily minimized is permitted for a wide class of nonsmooth problems. As a result of this relaxation of derivative consistency, the smoothing majorization leads to the convergence to a stationary solution in a more relaxed sense than the classic MM.

Furthermore, in contrast to minimizing the possibly nonconvex majorizing function

exactly, the smoothness of the majorizing function allows us to employ the idea of SCA, along with available separable approximation techniques, to efficiently obtain an approximate minimizer of the majorizing function that still provides a decrease of the original objective function. Thus, we develop a parallelizable inexact MM framework by combining the smoothing majorization technique and the idea of convex approximation. In this framework, the construction of the approximate problem at each iteration is divided into two steps, namely, the smoothing majorization and the convex approximation, so that the two desiderata of the approximate function can be treated separately. Specifically, the smoothing majorization step aims to design a smooth majorizer that is as tight to the original function as possible, regardless of the complexity. Then, in the convex approximation step, we further develop a convex approximate function that can be easily minimized, for the smooth majorizer. In particular, the separable approximation techniques can be employed to develop an approximate problem that can be decomposed and solved in parallel.

1.1 Original Contributions

This dissertation concentrates on the design and analysis of parallelizable approximation-based methods for nonconvex nonsmooth optimization problems. The original contributions in this thesis are extensions of the work

[LTY+22] T. Liu, A. M. Tillmann, Y. Yang, Y. C. Eldar, and M. Pesavento, "Extended successive convex approximation for phase retrieval with dictionary learning," *IEEE Transactions on Signal Processing*, vol. 70, pp. 6300–6315, 2022.

and consist of the following two parts.

In the first part of this thesis, i.e., Chapter 4, as a generalization of the algorithms that we developed in [LTY+22], we propose an efficient parallelizable algorithmic framework for a broad class of nonconvex nonsmooth optimization problems, by combining the smoothing majorization technique and the idea of SCA. As aforementioned, we study the smoothing majorization technique, which relaxes the derivative consistency in the majorization step so that a smooth majorizer that can be easily minimized is permitted for a wide class of nonsmooth problems. Specifically, as a generalization of the majorization technique that we employed in [LTY+22], we consider the situation that the majorizing function preserves only a subgradient of the original objective function.

We first describe the exact MM framework with such smoothing majorization, abbreviated as *smoothing MM*, and analyze its convergence behavior. In general, compared to the classic MM, the smoothing MM sacrifices the tightness of the convergence set with respect to the local minima in order to construct an approximate problem that can be easily addressed. In contrast to minimizing the possibly nonconvex majorizing function exactly, the smoothness of the majorizing function allows us to employ the idea of SCA, along with the available separable convex approximation techniques, to obtain an approximate minimizer of the majorizing function efficiently. This motivated our idea in [LTY+22] of combining the smoothing majorization and the separable convex approximation techniques to address the phase retrieval with dictionary learning problem. We generalize the algorithms in [LTY+22] to the aforementioned class of smoothing majorization techniques to develop a parallelizable inexact MM framework, named *smoothing SCA*, and provide a unified convergence analysis. Finally, similar to the classic MM and SCA frameworks, the smoothing MM and SCA can also be implemented block-coordinatewise to exploit potential separable structures of the constraints in the optimization problem, and the convergence behavior is studied accordingly.

The main content of the second part, comprising Chapter 5, has been published in [LTY+22]. As our mainly promoted framework, the smoothing SCA framework is employed to address the phase retrieval with dictionary learning problem. Phase retrieval focuses on reconstructing unknown signals from the magnitude measurements of linear combinations. When combined with dictionary learning, phase retrieval also incorporates the prior knowledge that the signal can be sparsely represented using an unknown dictionary. Two efficient parallel algorithms are developed by applying the smoothing SCA to two complementary nonconvex nonsmooth formulations, respectively, which are both based on a least-squares (LS) criterion. The first algorithm is termed *compact-SCAphase* and is preferable in the case of moderately diverse mixture models with a low number of mixing components. It adopts a compact formulation that avoids auxiliary variables. The proposed algorithm is highly scalable and has reduced parameter tuning costs. The second algorithm, referred to as *SCAphase*, uses auxiliary variables and is favorable in the case of highly diverse mixture models. It also renders simple incorporation of additional side constraints. The performance of both methods is evaluated when applied to blind channel estimation from subband magnitude measurements in a multi-antenna random access network. Simulation results show the efficiency of the proposed techniques compared to state-of-the-art methods.

To summarize, the main contributions of this thesis are as follows:

On the optimization theoretical aspect:

- Analyzing the convergence behavior of the MM framework with smoothing majorization where the derivative consistency is relaxed so that a smooth majorizer that is designed to have a simple minimization is permitted for a wide class of nonconvex nonsmooth problems.
- Proposing a parallelizable inexact MM framework by combining the smooth majorization technique and the idea of successive convex approximation. Its convergence behavior is also analyzed.
- Presenting the block-coordinatewise extensions of both the exact and inexact MM with smoothing majorization and analyzing their convergence behaviors accordingly.

On the application aspect:

- Proposing two efficient parallel algorithms for the phase retrieval with dictionary learning problem by applying the smoothing SCA framework to two complementary formulations, respectively.
- Refining the search range for suitable values of the sparsity parameter for both algorithms
- Proposing an efficient procedure based on rational approximation for solving the ℓ_2-norm constrained LS subproblems to reduce the overall computational complexity of compact-SCAphase.
- Analyzing theoretically the computational complexities of the proposed algorithms in comparison to the state-of-the-art method.
- Conducting extensive simulations in the context of blind channel estimation in a multi-antenna random access network in view of parameter selection, estimation quality, convergence speed, computational time, and robustness to initialization.

Lastly, we clarify the contribution of each author in the paper [LTY+22] as follows. A. Tillmann initiated the idea of applying the SCA framework to the phase retrieval with dictionary learning problem. T. Liu developed and implemented the algorithms, established their convergence, performed the numerical simulations, and wrote the manuscript. All co-authors provided useful comments on the development of the algorithms, the revision of the manuscript, and the responses to the reviewers. The interesting discussions with the reviewers of [LTY+22] guided us to the theory of subdifferentiation, which helped us with the convergence analysis of the algorithms in [LTY+22] and that of the general framework that we present in this thesis.

1.2 Thesis Structure

In **Chapter 2**, some basic concepts and facts in the theory of subdifferentiation and convex analysis that are relevant to the discussions in the following chapters are introduced.

Chapter 3 provides a short review of the related existing optimization frameworks, including the majorization-minimization (MM) framework, the successive convex approximation (SCA) framework, and their block-coordinatewise extensions.

Chapter 4 contains the main theoretical contributions of this thesis. In Section 4.1, the MM framework with smoothing majorization that relaxes the derivative consistency is described and its convergence behavior is studied. Then, in Section 4.2, a parallelizable inexact MM framework is developed by combining the smoothing majorization technique and the idea of successive convex approximation. Finally, the block-coordinatewise extensions of both the exact and inexact MM with smoothing majorization, as well as their convergence analyses, are presented in Section 4.3.

In **Chapter 5**, the smoothing SCA framework developed in the previous chapter is employed to derive two efficient parallel algorithms for the phase retrieval with dictionary learning problem. The computational complexities of the proposed algorithms are theoretically analyzed and both their error performance and computational time are evaluated by extensive simulations in the context of blind channel estimation from subband magnitude measurements in multi-antenna random access network, in comparison to the state-of-the-art methods.

The conclusions, together with some open problems and future research directions, are detailed in **Chapter 6**.

1.3 Publications

This dissertation is based on the following publications.

Internationally Refereed Journal Article

- T. Liu, A. M. Tillmann, Y. Yang, Y. C. Eldar, and M. Pesavento, "Extended successive convex approximation for phase retrieval with dictionary learning," *IEEE Transactions on Signal Processing*, vol. 70, pp. 6300–6315, 2022.

Book Section

- K. Ardah, M. Haardt, T. Liu, F. Matter, M. Pesavento, and M. E. Pfetsch, "Recovery under side constraints," in *Compressed sensing in information processing*, G. Kutyniok, H. Rauhut, and R. J. Kunsch, Eds., Cham: Springer International Publishing, 2022, pp. 213–246.

Internationally Refereed Conference Paper

- T. Liu, A. M. Tillmann, Y. Yang, Y. C. Eldar, and M. Pesavento, "A parallel algorithm for phase retrieval with dictionary learning," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2021.

The other publications produced during the period of doctoral candidacy are listed as follows.

Preprints

- T. Liu, S. P. Deram, K. Ardah, M. Haardt, M. E. Pfetsch, and M. Pesavento, "Gridless Parameter Estimation in Partly Calibrated Rectangular Arrays." submitted to the *IEEE Transactions on Signal Processing*, 2024. URL: `http://arxiv.org/abs/2406.16041`.

- T. Liu, F. Matter, A. Sorg, M. E. Pfetsch, M. Haardt, and M. Pesavento, "Joint sparse estimation with cardinality constraint via mixed-integer semidefinite programming." submitted to the *IEEE Transactions on Signal Processing*, 2023. URL: `http://arxiv.org/abs/2311.03501`.

Internationally Refereed Conference Papers

- T. Liu and M. Pesavento, "Blind Phase-Offset Estimation in Sparse Partly Calibrated Arrays," in *Proceedings of IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, Jul. 2024. **selected for the best student paper contest**.

- T. Liu, S. P. Deram, K. Ardah, M. Haardt, M. E. Pfetsch, and M. Pesavento, "Gridless parameter estimation in partly calibrated rectangular arrays," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2024.

- T. Liu, F. Matter, A. Sorg, M. E. Pfetsch, M. Haardt, and M. Pesavento, "Joint sparse estimation with cardinality constraint via mixed-integer semidefinite programming," in *Proceedings of IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec. 2023.

- Y. Zhang, T. Liu, and M. Pesavento, "Direction-of-arrival estimation for correlated sources and low sample size," in *Proceedings of European Signal Processing Conference (EUSIPCO)*, Sep. 2023.

- X. Wang, T. Liu, M. Trinh-Hoang, and M. Pesavento, "GPU-accelerated parallel optimization for sparse regularization," in *Proceedings of Sensor Array and Multichannel Signal Processing Workshop (SAM)*, Jun. 2020.

- T. Liu, M. T. Hoang, Y. Yang, and M. Pesavento, "A block coordinate descent algorithm for sparse Gaussian graphical model inference with laplacian constraints," in *Proceedings of IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec. 2019.

- T. Liu, M. T. Hoang, Y. Yang, and M. Pesavento, "A parallel optimization approach on the infinity norm minimization problem," in *Proceedings of European Signal Processing Conference (EUSIPCO)*, Sep. 2019. **selected for the best student paper contest**.

Chapter 2
Preliminaries

This chapter provides an introduction to some basic concepts and facts relevant to the following discussions in the thesis. We begin with several notions of stationarity, including, most importantly, the generalized notions of stationarity for nonconvex nonsmooth optimization problems, in Section 2.1. For simplicity of presentation, the stationarity is first discussed for functions with real arguments. In the end, some concepts in complex differentiation are presented in Section 2.3, with which the introduced notions of stationarity can be readily extended to problems with complex variables as those studied in Chapter 5. Additionally, Section 2.2 includes several generalized concepts of convexity that are involved in the analyses in the rest of the thesis.

2.1 Notions of Stationarity

This section begins with a short review of the classic concept of stationarity for smooth optimization problems, which is established on the derivatives by Fermat's rule as a necessary condition for local optimality. We first present some useful concepts.

Definition 2.1 (Directional derivative)**.** *The one-sided directional derivative of a function $f : \mathbb{R}^n \to \mathbb{R}$ at a point $\boldsymbol{x} \in \mathbb{R}^n$ in a direction $\boldsymbol{d} \in \mathbb{R}^n$ is defined as*

$$f'_{\boldsymbol{d}}(\boldsymbol{x}) = \lim_{t \downarrow 0} \frac{f(\boldsymbol{x} + t\boldsymbol{d}) - f(\boldsymbol{x})}{t}, \tag{2.1}$$

provided that the limit exists, where $t \downarrow 0$ means that t approaches 0 from above.

Definition 2.2 (Directional differentiability)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be directionally differentiable at $\boldsymbol{x} \in \mathbb{R}^n$ if the directional derivative $f'_{\boldsymbol{d}}(\boldsymbol{x})$ of f at $\boldsymbol{x}$ exists in any direction $\boldsymbol{d} \in \mathbb{R}^n$. The function f is said to be directionally differentiable if it is directionally differentiable at every $\boldsymbol{x} \in \mathbb{R}^n$.*

Definition 2.3 (Gradient)**.** *The directional derivative of $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x} \in \mathbb{R}^n$ along the ith coordinate, i.e., $f'_{\boldsymbol{e}_i}(\boldsymbol{x})$ with $\boldsymbol{e}_i$ being the ith standard basis vector (all component are 0 except for the ith component which is 1), is called the ith partial derivative of f at $\boldsymbol{x}$ and it is denoted by $\frac{\partial f}{\partial x_i}$, where x_i denote the ith component of the vector $\boldsymbol{x}$. Provided that all these partial derivatives exist, the gradient of f at $\boldsymbol{x}$ is defined as the vector*

$$\nabla f(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \in \mathbb{R}^n. \tag{2.2}$$

Definition 2.4 (Hessian)**.** *The Hessian of $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x} \in \mathbb{R}^n$, denoted by $\nabla^2 f(\boldsymbol{x})$, is an $n \times n$ matrix whose (i,j)th entry being*

$$\left[\nabla^2 f(\boldsymbol{x})\right]_{i,j} = \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j}. \tag{2.3}$$

In words, the Hessian contains all the second-order partial derivatives of the function. Note that the Hessian matrix is symmetric since $\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\boldsymbol{x})}{\partial x_j \partial x_i}$.

For analyzing the differential behavior of f only with respect to part of the coordinates, say $\boldsymbol{x}' = [x_{i_1}, \ldots, x_{i_k}]^\mathsf{T} \in \mathbb{R}^k$ with $k \leq n$, we define $\nabla_{\boldsymbol{x}'} f(\boldsymbol{x}) = \left[\frac{\partial f}{\partial x_{i_1}}, \ldots, \frac{\partial f}{\partial x_{i_k}}\right]^\mathsf{T} \in \mathbb{R}^k$ as the partial gradient of f in the coordinates $\boldsymbol{x}'$ at the point $\boldsymbol{x} \in \mathbb{R}^n$, and the partial Hessian $\nabla^2_{\boldsymbol{x}'} f(\boldsymbol{x}) \in \mathbb{R}^{k \times k}$ as the principal submatrix of $\nabla^2 f(\boldsymbol{x})$ involving the coordinates $\boldsymbol{x}'$.

Definition 2.5 (Gateaux differentiability)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is called (Gateaux) differentiable at $\boldsymbol{x} \in \mathbb{R}^n$ if and only if the gradient $\nabla f(\boldsymbol{x})$ exists and satisfies*

$$\left(\nabla f(\boldsymbol{x})\right)^\mathsf{T} \boldsymbol{d} = f'_{\boldsymbol{d}}(\boldsymbol{x}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n. \tag{2.4}$$

Definition 2.6 (Convex function)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be convex on the set $\mathcal{X} \subseteq \mathbb{R}^n$ if $\mathcal{X}$ is convex and*

$$f\left(\alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{y}\right) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{y}) \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, \forall\, \alpha \in [0,1]. \tag{2.5}$$

The function f is said to be strictly convex on $\mathcal{X}$ if the above inequality is satisfied with strict inequality for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ with $\boldsymbol{x} \neq \boldsymbol{y}$, and all $\alpha \in (0,1)$. We say that f is locally (strictly) convex at $\boldsymbol{x} \in \mathbb{R}^n$ if there exists some $\varepsilon > 0$ such that f is (strictly) convex on the ε-neighborhood $\mathcal{B}(\boldsymbol{x}, \varepsilon) \cap \mathcal{X}$ of $\boldsymbol{x}$.

A function f is said to be concave if $-f$ is convex. Here are some facts about convex functions:

- The convex functions are always directionally differentiable [Roc70, Thm. 23.1].
- Provided that f is differentiable, f is convex on a convex set $\mathcal{X}$ if and only if

  $$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \left(\nabla f(\boldsymbol{x})\right)^\mathsf{T} (\boldsymbol{y} - \boldsymbol{x}) \quad \forall\, \boldsymbol{y} \in \mathcal{X}. \tag{2.6}$$

  This means that the first-order Taylor approximation is a global lower bound for a differentiable convex function.

- Provided that f is twice differentiable, i.e., its Hessian exits everywhere, f is convex on a convex set $\mathcal{X}$ if and only if its Hessian is positive semidefinite:

$$\nabla^2 f(\boldsymbol{x}) \succeq 0 \quad \forall \boldsymbol{x} \in \mathcal{X}. \tag{2.7}$$

For differentiable functions, the concept of stationarity is introduced by Fermat's rule as a necessary condition for local minima.

Definition 2.7 (Stationarity for unconstrained minimization)**.** *Assume that the function $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable. A point $\boldsymbol{x} \in \mathbb{R}^n$ is said to be a stationary point of f if it satisfies*

$$\nabla f(\boldsymbol{x}) = \mathbf{0}. \tag{2.8}$$

By Fermat's rule, the equation (2.8) is a necessary condition for $\boldsymbol{x}$ being a local minimum point of f. In fact, a stationary point $\boldsymbol{x}$ is either a local minimum, a local maximum, or a saddle point. Now consider an unconstrained minimization problem where the function f is to be minimized on a subset $\mathcal{X} \subseteq \mathbb{R}^n$. The condition (2.8) is not necessarily satisfied when f attains a local minimum at the boundary of $\mathcal{X}$. Hence, the concept of stationarity is generalized as follows to further include potential local minimum points at the boundary of the feasible set [HL01].

Definition 2.8 (Stationarity for constrained minimization)**.** *Assume that the function $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable. A point $\boldsymbol{x} \in \mathbb{R}^n$ is said to be a stationary point of the problem of minimizing f over the set $\mathcal{X} \subseteq \mathbb{R}^n$ if it satisfies*

$$f'_{\boldsymbol{d}}(\boldsymbol{x}) \geq 0 \quad \forall\, \boldsymbol{d} \text{ with } \boldsymbol{x} + \boldsymbol{d} \in \mathcal{X}. \tag{2.9}$$

It is easy to show that a local minimum point $\boldsymbol{x}$ fulfills the stationarity condition (2.9) by the fact that any difference quotient at a local minimum point is nonnegative, i.e.,

$$\frac{f(\boldsymbol{x} + \varepsilon \boldsymbol{d}) - f(\boldsymbol{x})}{\varepsilon} \geq 0 \quad \text{for any } \varepsilon > 0 \text{ and } \boldsymbol{d} \in \mathbb{R}^n \text{ such that } \boldsymbol{x} + \varepsilon \boldsymbol{d} \in \mathcal{X}. \tag{2.10}$$

For an interior point $\boldsymbol{x} \in \operatorname{int}(\mathcal{X})$, the inequality in (2.9) needs to be satisfied in all directions $\boldsymbol{d} \in \mathbb{R}^n$ and, hence, the condition (2.9) becomes equivalent to (2.8). Moreover, the concept of stationarity in Definition 2.8 can directly be extended to the functions that are nondifferentiable but only directionally differentiable.

When f is not stationary at $\boldsymbol{x}$, then there exists a descent direction of f at $\boldsymbol{x}$ defined as follows.

Definition 2.9 (Descent direction)**.** *The vector $\boldsymbol{d} \in \mathbb{R}^n$ is said to be a descent direction of the function $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x}$ if*

$$f'_{\boldsymbol{d}}(\boldsymbol{x}) < 0. \tag{2.11}$$

When (2.11) is satisfied at $\boldsymbol{x}$, a decrease of f can be obtained by updating the variable $\boldsymbol{x}$ along the descent direction $\boldsymbol{d}$. The reason is that, by the definition of the limit, there exists a small $\delta > 0$ such that

$$\frac{f(\boldsymbol{x} + \varepsilon \boldsymbol{d}) - f(\boldsymbol{x})}{\varepsilon} < 0 \quad \forall\, \varepsilon \in (0, \delta), \tag{2.12}$$

since the limit of the difference quotient, i.e., $f'_{\boldsymbol{d}}(\boldsymbol{x})$, is negative. However, the converse is not always true. That is, $f(\boldsymbol{y}) < f(\boldsymbol{x})$ for an arbitrary function f does not necessarily imply that $\boldsymbol{y} - \boldsymbol{x}$ is a descent direction of f at $\boldsymbol{x}$.

Finally, we remark that, in the general case, the stationarity is only a necessary condition for the local optimality. In the special case where the optimization problem is convex, the stationarity also implies the global optimality [BV04].

In the following, an introduction to the theory of subdifferentiation for convex and, more generally, nonconvex nonsmooth functions is provided. In particular, the so-called subdifferential is proposed as a generalization of the gradient for extending the stationarity concept to nonsmooth functions. In contrast to smooth optimization, where the definition of stationarity is standard, there are numerous definitions of subdifferential proposed for establishing the stationarity concept in nonsmooth optimization, and each of them possesses attractive properties for a different class of nonconvex nonsmooth functions. Then, only a few of the existing definitions of subdifferential and their corresponding stationarity concepts that are relevant to the later discussions in this thesis will be presented. We refer interested readers to the survey paper [LSM20] (and the references therein) for a comprehensive comparison of different definitions of subdifferential and the corresponding generalized stationarity concepts.

2.1.1 Convex Nonsmooth Functions

We begin with a review of the theory of subdifferentiation for convex nonsmooth functions. Inspired by the fact in (2.6) that, if the function f is convex and differentiable at $\boldsymbol{x}$, the gradient $\nabla f(\boldsymbol{x})$ provides an affine minorant of f at $\boldsymbol{x}$, the notion of subdifferential is introduced in the nonsmooth case as a reasonable generalization of the gradient.

Definition 2.10 (Convex subdifferential)**.** *For an extended-real-valued convex function* $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$*, the set*

$$\partial f(\boldsymbol{x}) = \left\{ \boldsymbol{s} \in \mathbb{R}^n \mid f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{s}^{\mathsf{T}}(\boldsymbol{y} - \boldsymbol{x}) \text{ for all } \boldsymbol{y} \right\} \tag{2.13}$$

is called the subdifferential of f *at* $\boldsymbol{x} \in \mathbb{R}^n$*. A vector* $\boldsymbol{s} \in \partial f(\boldsymbol{x})$ *is called a subgradient of* f *at* $\boldsymbol{x}$*.*

For studying the variation of f only with respect to part of the coordinates, say $\boldsymbol{x}' = [x_{i_1}, \ldots, x_{i_k}]^{\mathsf{T}} \in \mathbb{R}^k$ with $k \leq n$, let $\partial_{\boldsymbol{x}'} f(\boldsymbol{x}) \subseteq \mathbb{R}^k$ denote the subdifferential of f with respect to the coordinates $\boldsymbol{x}'$ at the point $\boldsymbol{x} \in \mathbb{R}^n$. The above definition is given for extended-valued functions as later we will consider the functions that take infinite values such as the indicator function. The subdifferential is a closed convex set. The following properties of the convex subdifferential are useful.

- **Differentiable function:** If f is differentiable at $\boldsymbol{x}$, then its gradient at $\boldsymbol{x}$ is the unique subgradient at $\boldsymbol{x}$, i.e., $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$ [HL01, Ch. D, Corollary 2.1.4].

- **Sum rule:** Note that the nonnegative combination preserves the convexity of functions. For two convex functions $f_1, f_2 : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ and $\alpha_1, \alpha_2 \geq 0$, we have [HL01, Ch. D, Thm. 4.1.1]

  $$\partial(\alpha_1 f_1 + \alpha_2 f_2)(\boldsymbol{x}) = \alpha_1 \partial f_1(\boldsymbol{x}) + \alpha_2 \partial f_2(\boldsymbol{x}). \tag{2.14}$$

  In particular, later we will consider the sum of two functions.

- **Indicator:** Define the indicator function $\mathbb{I}_{\mathcal{X}} : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ of a set $\mathcal{X} \subseteq \mathbb{R}^n$ as

  $$\mathbb{I}_{\mathcal{X}}(\boldsymbol{x}) = \begin{cases} 0, & \boldsymbol{x} \in \mathcal{X}, \\ +\infty, & \text{elsewhere.} \end{cases} \tag{2.15}$$

  The indicator function $\mathbb{I}_{\mathcal{X}}$ is convex if and only if the set $\mathcal{X}$ is convex. When $\mathcal{X}$ is convex, the subdifferential of the indicator function can be constructed by (2.13) as

  $$\partial \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}) = \left\{ \boldsymbol{s} \in \mathbb{R}^n \mid \boldsymbol{s}^{\mathsf{T}}(\boldsymbol{y} - \boldsymbol{x}) \leq 0 \text{ for all } \boldsymbol{y} \in \mathcal{X} \right\} \tag{2.16}$$

  if $\boldsymbol{x} \in \mathcal{X}$ and $\partial \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}) = \emptyset$ otherwise. The set on the right-hand side in (2.16) is known as the normal cone to $\mathcal{X}$ at $\boldsymbol{x}$, which we denote by $\mathcal{N}_{\mathcal{X}}(\boldsymbol{x})$. The normal cone $\mathcal{N}_{\mathcal{X}}(\boldsymbol{x})$ contains all points that do not make an acute angle with any line segment in $\mathcal{X}$ with $\boldsymbol{x}$ as an endpoint. At an interior point $\boldsymbol{x} \in \operatorname{int}(\mathcal{X})$, the normal cone reduces to $\mathcal{N}_{\mathcal{X}}(\boldsymbol{x}) = \{\boldsymbol{0}\}$.

- **Fermat's rule:** With the concept of subdifferential, Fermat's rule can be generalized to nondifferentiable functions as

$$\mathbf{0} \in \partial f(\boldsymbol{x}) \tag{2.17}$$

holds at a local minimum point $\boldsymbol{x}$ of f. This can be used to define the notion of stationarity for convex nonsmooth functions.

To present a unified notion of stationarity for general convex minimization problems, including constrained and unconstrained problems, we introduce the following reformulation of constrained problems. The constrained minimization problem (1.1) can be rewritten as the following unconstrained minimization of an extended-valued function:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \quad f(\boldsymbol{x}) + \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}). \tag{2.18}$$

In other words, we enforce the objective function value to be $+\infty$ outside the feasible set $\mathcal{X}$. In the case where the constrained problem (1.1) is convex, i.e., where f is a convex function and $\mathcal{X}$ is a convex set, the problem (2.18) is an unconstrained minimization of a convex function, whose stationary points are determined by the condition

$$\mathbf{0} \in \partial(f + \mathbb{I}_{\mathcal{X}})(\boldsymbol{x}) = \partial f(\boldsymbol{x}) + \partial \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}) = \partial f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{x}). \tag{2.19}$$

The first equality in (2.19) comes from the sum rule of subdifferential. In summary, the concept of stationarity for convex nonsmooth optimization problems is given as follows.

Definition 2.11 (Stationarity for convex nonsmooth problems)**.** *Consider the constrained minimization problem* (1.1) *where f is a convex function and $\mathcal{X}$ is a convex set. A point $\boldsymbol{x} \in \mathcal{X}$ is said to be a stationary point of the problem* (1.1) *if it satisfies*

$$\mathbf{0} \in \partial f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{x}). \tag{2.20}$$

Similar to the smooth case, at an interior point of $\mathcal{X}$, the condition (2.20) reduces to Fermat's rule (2.17) for unconstrained minimization. Since the problem is convex, the notion of stationarity in Definition 2.11 also implies global optimality.

It is not difficult to see that the construction (2.13) of the subdifferential does not provide much useful information when the function is no longer convex. For instance, the subdifferential does not always coincide with the gradient for nonconvex differentiable functions. As a consequence, a local maximum of a nonconvex differentiable function will be identified as a stationary point according to the standard stationarity (2.8) in

the smooth case, whereas the construction (2.13) of the subdifferential at this point is empty. Therefore, in the following, we further present two generalized definitions of subdifferential that extend some desired properties and calculus rules of the gradient and the convex subdifferential to some important classes of nonconvex nonsmooth functions.

2.1.2 Nonconvex Nonsmooth Functions

As mentioned earlier, convex functions are known to be directionally differentiable. Hence, when we move beyond convex functions, one direction to explore is the class of directionally differentiable functions. In particular, in the convex case, it can be shown that the subdifferential $\partial f(\boldsymbol{x})$ defined in (2.13) can be equivalently constructed as follows by using the directional derivatives.

Definition 2.12 (Generalized subdifferential based on directional derivatives)**.** *Assume that the function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ admits directional derivatives, which may be infinite, at $\boldsymbol{x} \in \mathbb{R}^n$. Then the set*

$$\underline{\partial} f(\boldsymbol{x}) = \left\{ \boldsymbol{s} \in \mathbb{R}^n \mid \boldsymbol{s}^\mathsf{T} \boldsymbol{d} \leq f'_{\boldsymbol{d}}(\boldsymbol{x}) \text{ for all } \boldsymbol{d} \in \mathbb{R}^n \right\} \tag{2.21}$$

is considered as a generalized subdifferential of f at $\boldsymbol{x}$. Its elements are referred to as generalized subgradients.

For convex functions, the generalized concept of subdifferential (2.21) coincides with the convex subdifferential (2.13), i.e., $\underline{\partial} f(\boldsymbol{x}) = \partial f(\boldsymbol{x})$ [HL01, Thm. 1.2.2]. Moreover, unlike the convex subdifferential (2.13), the generalized concept of subdifferential (2.21) is always consistent with the gradient if it exists, i.e., $\underline{\partial} f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$, regardless of whether f is convex or not.

The directional derivative $f'_{\boldsymbol{d}}(\boldsymbol{x})$ is positively homogeneous in $\boldsymbol{d}$, i.e., $f'_{\gamma \boldsymbol{d}}(\boldsymbol{x}) = \gamma f'_{\boldsymbol{d}}(\boldsymbol{x})$ for any $\boldsymbol{d} \in \mathbb{R}^n$ and $\gamma > 0$. When $f'_{\boldsymbol{d}}(\boldsymbol{x})$ is convex in $\boldsymbol{d}$, which does not require the convexity of f, some attractive calculus of the convex subdifferential, such as the sum rule in (2.14) with equality, still holds for the generalized subdifferential $\underline{\partial} f(\boldsymbol{x})$. If $f'_{(\cdot)}(\boldsymbol{x})$ is also proper and closed, then it corresponds to the support function of the generalized subdifferential, i.e.,

$$f'_{\boldsymbol{d}}(\boldsymbol{x}) = \sup_{\boldsymbol{s} \in \underline{\partial} f(\boldsymbol{x})} \boldsymbol{s}^\mathsf{T} \boldsymbol{d}. \tag{2.22}$$

Another motivation of the generalized concept of subdifferential in (2.21) is that, with this subdifferential, the concept of stationarity can be generalized as follows to the minimization of nonconvex nonsmooth but directionally differentiable functions over a convex set.

Definition 2.13 (Directional stationarity)**.** *Consider the constrained minimization problem* (1.1) *where f is directionally differentiable and $\mathcal{X}$ is a convex set. A point $\boldsymbol{x} \in \mathcal{X}$ is said to be a stationary point of the problem* (1.1) *if it satisfies*

$$\mathbf{0} \in \underline{\partial} f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{x}). \tag{2.23}$$

It is easy to see that the condition (2.23) is equivalent to the condition (2.9), which has been proven to be a necessary condition for f attaining a local minimum over $\mathcal{X}$ at $\boldsymbol{x}$ even without the convexity of f. The stationarity defined in Definition 2.13 is then referred to as directional stationarity (d-stationarity) to be distinguished from the other generalized notions of stationarity that will be introduced later.

Next, we move beyond directionally differentiable functions and introduce another generalization of subdifferential, known as the Clarke subdifferential, that can be used to study the variation of a very general class of nonconvex and nonsmooth functions. We start to present the definition of Clarke subdifferential with the class of locally Lipschitz functions, which later will be extended to any functions. The class of locally Lipschitz functions, defined as follows, captures a broad diversity of nonconvex functions [Cla75] and includes the class of convex functions as a special case [Roc70, Thm. 10.4].

Definition 2.14 (Global and local Lipschitz continuity)**.** *A function $f : \mathcal{X} \to \mathbb{R}$ is said to satisfy a Lipschitz condition on a subset $\mathcal{X}' \subseteq \mathcal{X}$ if there exists a constant $L > 0$ such that*

$$|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \leq L\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2 \quad \forall\, \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}'. \tag{2.24}$$

1) *f is said to be Lipschitz continuous (or simply Lipschitz) if it satisfies a Lipschitz condition with some $L > 0$ on $\mathcal{X}$.*

2) *f is said to be Lipschitz near a point $\boldsymbol{x} \in \mathcal{X}$ if there exists some $\varepsilon > 0$ such that f restricted to the ε-neighborhood $\mathcal{B}(\boldsymbol{x}, \varepsilon) \cap \mathcal{X}$ of $\boldsymbol{x}$ is Lipschitz.*

3) *f is said to be locally Lipschitz if f is Lipschitz near every $\boldsymbol{x} \in \mathcal{X}$.*

A locally Lipschtiz function may be neither differentiable nor directionally differentiable even though its difference quotient is bounded. However, the boundedness of difference quotients ensures that the following generalized directional derivative, which we refer to as Clarke directional derivative, exists everywhere.

Definition 2.15 (Clarke directional derivative)**.** *[Cla90] The Clarke directional derivative of a function $f : \mathbb{R}^n \to \mathbb{R}$ at a point $\boldsymbol{x} \in \mathbb{R}^n$ in a direction $\boldsymbol{d} \in \mathbb{R}^n$ is defined as*

$$f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x}) = \limsup_{\boldsymbol{x}' \to \boldsymbol{x},\, t \downarrow 0} \frac{f(\boldsymbol{x}' + t\boldsymbol{d}) - f(\boldsymbol{x}')}{t}, \tag{2.25}$$

where the upper limit exists if f is Lipschitz near $\boldsymbol{x}$.

Unlike the classic directional derivative (2.1), the Clarke directional derivative (2.25) captures the variation of the function f in the neighborhood of $\boldsymbol{x}$, not just along a ray emanating from $\boldsymbol{x}$. For locally Lipschitz functions, the Clarke directional derivative $f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x})$ is a finite convex and positively homogeneous function of the direction $\boldsymbol{d}$ for any $\boldsymbol{x}$. Thus, inspired by the generalization of subdifferential in (2.21), the following generalized concept of subdifferential is introduced in [Cla90] for the analysis of nonconvex and nonsmooth functions.

Definition 2.16 (Clarke subdifferential)**.** *The Clarke subdifferential of a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x} \in \mathbb{R}^n$ is defined as the nonempty convex compact set whose support function is $f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x})$, i.e.,*

$$\partial^C f(\boldsymbol{x}) = \left\{ \boldsymbol{s} \in \mathbb{R}^n \mid \boldsymbol{s}^{\mathsf{T}} \boldsymbol{d} \leq f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x}) \text{ for all } \boldsymbol{d} \in \mathbb{R}^n \right\}. \tag{2.26}$$

Its elements are referred to as Clarke subgradients.

Also, the Clarke subdifferential $\partial^C f(\boldsymbol{x})$ defined above is equivalent to the convex hull of the sets of limits of the gradient of any sequence that converges to $\boldsymbol{x}$ [Cla90, Sec. 2.5], i.e.,

$$\partial^C f(\boldsymbol{x}) = \operatorname{co} \left\{ \boldsymbol{s} \in \mathbb{R}^n \mid \exists \boldsymbol{x}_i \to \boldsymbol{x}, \nabla f(\boldsymbol{x}_i) \to \boldsymbol{s} \right\}. \tag{2.27}$$

The above two constructions of Clarke subdifferential cannot be applied to extended-valued functions, such as the indicator function, which are apparently not locally Lipschitz. To resolve this issue, an extended definition of Clarke subdifferential is presented in [Cla90] for any function taking values in $\mathbb{R} \cup \{+\infty\}$, locally Lipschitz or not, which is consistent with Definition 2.16 for locally Lipschitz case. As the discussions in the following chapters are limited to constrained optimization problems with locally Lipschitz objective functions, the extended definition for non-Lipschitz functions is omitted and the reader is referred to [Cla90, Def. 2.4.10] for the precise definition. In fact, construction (2.26) of the Clark subdifferential is also valid in the case where the Clarke directional derivatives exist but take values in $\mathbb{R} \cup \{+\infty\}$.

By definition, the Clarke directional derivative $f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x})$ is generally an upper bound on the difference quotient in the neighborhood of $\boldsymbol{x}$. In particular, we have

$$f_{\boldsymbol{d}}'(\boldsymbol{x}) \leq f_{\boldsymbol{d}}^{\circ}(\boldsymbol{x}), \tag{2.28}$$

when the directional derivative $f'_{\boldsymbol{d}}(\boldsymbol{x})$ exists. This leads to the following relation between the two generalized subdifferentials $\underline{\partial} f$ in (2.21) and $\partial^C f$ in (2.26):

$$\underline{\partial} f(\boldsymbol{x}) \subseteq \partial^C f(\boldsymbol{x}), \tag{2.29}$$

and the two generalized subdifferentials become equivalent when the equality holds in (2.28). As we will see later, the Clarke subdifferential retains many of the useful properties of the classic convex subdifferential in (2.13) when the equality holds in (2.28). We first introduce the following concept.

Definition 2.17 (Subdifferentially regular function)**.** *[Cla90, Definition 2.3.4] A locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$ is subdifferentially regular (or simply regular) at $\boldsymbol{x} \in \mathbb{R}^n$ if, for every direction $\boldsymbol{d} \in \mathbb{R}^n$, the ordinary directional derivative exists and coincides with the Clarke directional derivative, i.e.,*

$$f'_{\boldsymbol{d}}(\boldsymbol{x}) = f^{\circ}_{\boldsymbol{d}}(\boldsymbol{x}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n. \tag{2.30}$$

The function f is said to be regular if it is regular at every $\boldsymbol{x} \in \mathbb{R}^n$.

Although the above concept of regularity is described for locally Lipschitz functions, it can be extended to non-Lipschitz functions in a similar fashion as the Clarke subdifferential (see [Cla90, Def. 2.4.10]). Some classes of regular functions are listed in the following.

Proposition 2.1 (Some classes of regular functions)**.** *[Cla90, Proposition 2.3.6] Let the function f be locally Lipschitz near $\boldsymbol{x}$.*

1) If f is strictly differentiable at $\boldsymbol{x}$, then f is regular at $\boldsymbol{x}$.

2) If f is locally convex at $\boldsymbol{x}$, then f is regular at $\boldsymbol{x}$.

Now we are ready to introduce the following properties of the Clarke subdifferential.

- **Smooth function:** If f is smooth, i.e., continuously differentiable, at $\boldsymbol{x}$, then

$$\partial^C f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\} \tag{2.31}$$

  In fact, the equality (2.31) holds, i.e., $\partial^C f(\boldsymbol{x})$ is a singleton, if and only if f is strictly differentiable at $\boldsymbol{x}$ [Cla90, Prop. 2.2.4]. If f is differentiable, but not necessarily strictly differentiable, at $\boldsymbol{x}$, then $\nabla f(\boldsymbol{x}) \in \partial^C f(\boldsymbol{x})$ [Cla90, Prop. 2.2.2].

- **Convex function:** It is easy to verify that convex functions are locally Lipschitz and regular. In this case, the Clarke subdifferential coincides with the classic convex subdifferential as $\partial^C f(\boldsymbol{x}) = \underline{\partial} f(\boldsymbol{x}) = \partial f(\boldsymbol{x})$.

- **Scaling rule:** For any scalar $\alpha \in \mathbb{R}$, we have

$$\partial^C(\alpha f)(\boldsymbol{x}) = \alpha \partial^C f(\boldsymbol{x}). \tag{2.32}$$

  In particular, when f is concave, i.e., $-f$ is convex, we have

$$\partial^C f(\boldsymbol{x}) = -\partial^C(-f)(\boldsymbol{x}) = -\partial(-f)(\boldsymbol{x}). \tag{2.33}$$

- **Sum rule:** In the general case, we only have the following weaker version of sum rule:

$$\partial^C(f_1 + f_2)(\boldsymbol{x}) \subseteq \partial^C f_1(\boldsymbol{x}) + \partial^C f_2(\boldsymbol{x}). \tag{2.34}$$

  Nevertheless, the sum rule holds with equality if f_1 and f_2 are regular at $\boldsymbol{x}$.

- **Indicator:** Similar to the convex subdifferential, the Clarke subdifferential $\partial^C \mathbb{I}_{\mathcal{X}}(\boldsymbol{x})$ of the indicator function corresponds to a generalized normal cone, which we refer to as Clarke normal cone and denote by $\mathcal{N}_{\mathcal{X}}^C(\boldsymbol{x})$, at $\boldsymbol{x} \in \mathcal{X}$ for any set $\mathcal{X}$, not necessarily convex. Similarly, we have $\mathcal{N}_{\mathcal{X}}^C(\boldsymbol{x}) = \{\mathbf{0}\}$ at an interior point $\boldsymbol{x} \in \operatorname{int}(\mathcal{X})$. When $\mathcal{X}$ is convex, the Clarke normal cone coincides with the convex normal cone, i.e., $\mathcal{N}_{\mathcal{X}}^C(\boldsymbol{x}) = \mathcal{N}_{\mathcal{X}}(\boldsymbol{x})$. The reader is referred to [Cla90] for the details of the generalized concept of the normal cone.

- **Fermat's rule:** With this generalized concept of subdifferential, Fermat's rule can be extended to general nonconvex and nonsmooth functions as follows [Cla90, Prop. 2.3.2]. If f attains a local minimum at $\boldsymbol{x}$, then

$$\mathbf{0} \in \partial^C f(\boldsymbol{x}). \tag{2.35}$$

  Like the usual concept of stationarity in the smooth case, the condition (2.35) is a necessary condition for the local optimality but not sufficient unless f is convex.

Applying Fermat's rule to the reformulation in (2.18), we obtain the following necessary condition for $\boldsymbol{x}$ being a local minimizer of a nonconvex and nonsmooth function f over the set $\mathcal{X}$:

$$\mathbf{0} \in \partial^C(f + \mathbb{I}_{\mathcal{X}})(\boldsymbol{x}). \tag{2.36}$$

Moreover, the following is also satisfied at $\boldsymbol{x}$:

$$\mathbf{0} \in \partial^C f(\boldsymbol{x}) + \partial^C \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}) = \partial^C f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}}^C(\boldsymbol{x}), \tag{2.37}$$

since, by the sum rule of the Clarke subdifferential, we have

$$\partial^C(f + \mathbb{I}_{\mathcal{X}})(\boldsymbol{x}) \subseteq \partial^C f(\boldsymbol{x}) + \partial^C \mathbb{I}_{\mathcal{X}}(\boldsymbol{x}). \tag{2.38}$$

The two conditions (2.36) and (2.37) are generally not equivalent since the sum rule does not hold with equality. Both conditions are often used to define a generalized notion of stationarity for nonconvex nonsmooth problems [LSM20, PRA17]. Although the condition (2.36) is tighter than (2.37), it is usually difficult to calculate the Clarke subdifferential of the non-Lipschitz function $f + \mathbb{I}_{\mathcal{X}}$ in practical applications even if the feasible set $\mathcal{X}$ is restricted to be convex. Therefore, in this thesis, we adopt the following generalized stationarity based on the condition (2.37).

Definition 2.18 (Clarke stationarity)**.** *A point $\boldsymbol{x} \in \mathcal{X}$ is said to be a Clarke stationary (C-stationary) point of the constrained minimization problem* (1.1) *if it satisfies*

$$\mathbf{0} \in \partial^C f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}}^C(\boldsymbol{x}). \tag{2.39}$$

As the discussions in the following chapters are limited to the constrained minimization over a convex set, more specifically, a closed and convex set, we will simplify the Clarke normal cone $\mathcal{N}_{\mathcal{X}}^C$ to the convex normal cone $\mathcal{N}_{\mathcal{X}}$.

Consider the minimization over a convex feasible set. Due to the inclusion property in (2.29), if $\boldsymbol{x}$ is a d-stationary point of the problem (1.1), then it is also a C-stationary point of (1.1). That is, we have the following implication:

$$\text{d-stationarity} \quad \Longrightarrow \quad \text{C-stationarity}. \tag{2.40}$$

This means that, compared to the d-stationarity, the C-stationarity may include more stationary points that are not local minima. The Clarke subdifferential sacrifices the tightness of stationarity in order to obtain useful properties and calculus rules, similar to that of the gradient and the convex subdifferential, for a wide class of nonconvex and nonsmooth functions. In Figure 2.1, we give two examples to show the implication in (2.40) at nondifferentiable points as well as the fact that the reverse implication does not always hold.

2.1.3 Coordinatewise Stationarity

Consider a subclass of the constrained optimization problem (1.1) where the feasible set $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_K$ is a Cartesian product of lower dimensional closed convex sets $\mathcal{X}_k \subseteq \mathbb{R}^{n_k}$. In this case, the problem can be rewritten as

$$\begin{aligned} \min_{\{\boldsymbol{x}_k\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k,\ k = 1, \ldots, K, \end{aligned} \tag{2.41}$$

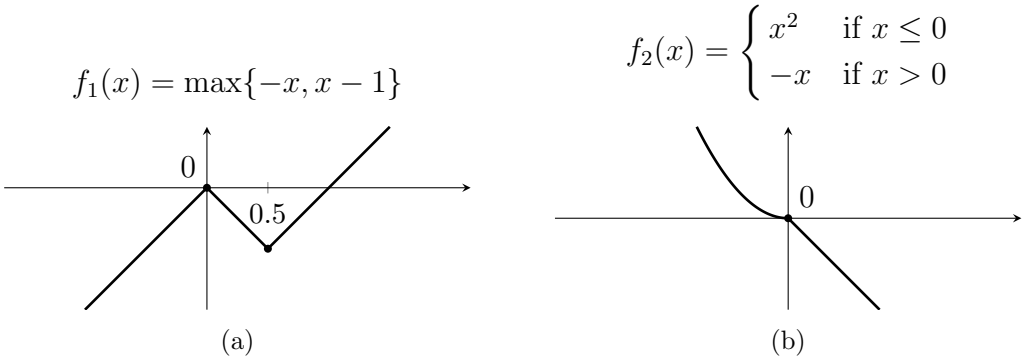


Figure 2.1. Examples univariate functions for showing the relation (2.40) between the two generalized subdifferentials. (a) For $f_1 : \mathbb{R} \to \mathbb{R}$, we have $\underline{\partial} f_1(0) = \emptyset$ and $\partial^C f_1(0) = [-1, 1]$. It implies that $x = 0$, which is a local maximum, is C-stationary but not d-stationary. The local minimum $x = 0.5$ is the unique d-stationary point, which is also C-stationary since $\partial^C f(0.5) = \underline{\partial} f(0.5) = [-1, 1]$. (b) For $f_2 : \mathbb{R} \to \mathbb{R}$, we have $\underline{\partial} f_2(0) = \emptyset$ and $\partial^C f_2(0) = [-1, 0]$. It follows that $x = 0$, which is neither a local minimum nor a local maximum, is C-stationary but not d-stationary.

where the optimization variable $\boldsymbol{x}$ in (1.1) is partitioned accordingly as $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K)$ with each $\boldsymbol{x}_k \in \mathbb{R}^{n_k}$ and $n_1 + \cdots + n_K = n$. A simple idea to break the problem (2.41) into a sequence of simpler problems by exploiting this separable structure of the constraints is alternately minimizing the objective function f with respect to each block variable $\boldsymbol{x}_k$ while the other blocks are fixed, which is known as the block coordinate descent (BCD) method and will be introduced in Section 3.3. However, since each block $\boldsymbol{x}_k$ is updated separately, it in general only achieves a coordinatewise minimum or a coordinatewise stationary point defined as follows.

Definition 2.19 (Coordinatewise minimum)**.** *A point $\boldsymbol{x} \in \mathcal{X}$ is a coordinatewise minimum solution of the problem* (2.41) *if*

$$f(\boldsymbol{x} + \boldsymbol{d}) \ge f(\boldsymbol{x}) \quad \forall\, \boldsymbol{d} = (\boldsymbol{0}, \ldots, \boldsymbol{0}, \boldsymbol{d}_k, \boldsymbol{0}, \ldots, \boldsymbol{0}) \text{ with } \boldsymbol{x}_k + \boldsymbol{d}_k \in \mathcal{X}_k, \quad \forall\, k = 1, \ldots, K. \tag{2.42}$$

Moreover, $\boldsymbol{x}$ is said to be a coordinatewise local minimum point of the function f if there exists some $\varepsilon > 0$ such that $\boldsymbol{x}$ is a coordinatewise minimum point of f on the ε-neighborhood $\mathcal{B}(\boldsymbol{x}, \varepsilon) \cap \mathcal{X}$ of $\boldsymbol{x}$.

Similar to the concept of stationarity, the coordinatewise stationarity is introduced below as a necessary condition for the coordinatewise local minimum.

Definition 2.20 (Coordinatewise stationarity)**.** *A point $\boldsymbol{x} \in \mathcal{X}$ is said to be a coordinatewise stationary point of the problem* (2.41) *if it satisfies*

$$\boldsymbol{0} \in \partial_{\boldsymbol{x}_k} f(\boldsymbol{x}) + \mathcal{N}_{\mathcal{X}_k}(\boldsymbol{x}_k) \quad \forall\, k = 1, \ldots, K. \tag{2.43}$$

Similarly, $\boldsymbol{x}$ is said to be a coordinatewise d-stationary point and a coordinatewise C-stationary point if (2.43) *is satisfied with the generalized subdifferentials $\underline{\partial}_{\boldsymbol{x}_k} f$ in* (2.21) *and $\partial^C_{\boldsymbol{x}_k} f$ in* (2.26)*, respectively.*

In other words, a coordinatewise stationary point is a point where f is stationary with respect to each block of variables, respectively, which is a weaker form of stationarity. A stationary point is apparently always a coordinatewise stationary point, but the converse does not necessarily hold. The following concept is introduced to describe the class of functions for which coordinatewise stationarity also implies joint stationarity.

Definition 2.21 (Coordinatewise regularity)**.** *The function f is said to be coordinatewise regular at a coordinatewise stationary (resp., in a generalized sense) point $\boldsymbol{x}$ if $\boldsymbol{x}$ is also a stationary (resp., in a generalized sense) point of f.*

It is easy to justify that the following class of functions is coordinatewise regular everywhere [DFKS15, SS18]:

$$f(\boldsymbol{x}) = u(\boldsymbol{x}) + \sum_{k=1}^{K} v_k(\boldsymbol{x}_k), \tag{2.44}$$

where u is continuously differentiable but each v_k is not necessarily smooth. That is, the nonsmooth component of f is also separable across the block variables. Note that this case certainly does not include all situations where the coordinatewise regularity holds.

2.2 Notions of Convexity

In this section, we briefly introduce several generalized notions of convexity that are involved in the analyses in the following chapters. For simplicity, some of the concepts below are only described for differentiable functions, which can certainly be extended to some classes of nondifferentiable functions.

Definition 2.22 (Quasiconvex function)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be quasiconvex on the set $\mathcal{X} \subseteq \mathbb{R}^n$ if $\mathcal{X}$ is convex and*

$$f\left(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}\right) \leq \max\left(f(\boldsymbol{x}), f(\boldsymbol{y})\right) \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X},\, \forall \alpha \in [0,1]. \tag{2.45}$$

The function f is said to be strictly quasiconvex on $\mathcal{X}$ if the above inequality is satisfied with strict inequality for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ with $\boldsymbol{x} \neq \boldsymbol{y}$, and all $\alpha \in (0,1)$.

This implies that, if $f(\boldsymbol{y}) \leq f(\boldsymbol{x})$, then the function value on the line segment from $\boldsymbol{x}$ to $\boldsymbol{y}$ is bounded from above by $f(\boldsymbol{x})$, i.e.,

$$f(\boldsymbol{y}) \leq f(\boldsymbol{x}) \quad \Longrightarrow \quad f\left(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}\right) \leq f(\boldsymbol{x}) \quad \forall\, \alpha \in [0,1], \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}. \tag{2.46}$$

Definition 2.23 (Pseudoconvex function)**.** *[Man94, Sec. 9.3] A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be pseudoconvex on the set $\mathcal{X} \subseteq \mathbb{R}^n$ if*

$$\left(\nabla f(\boldsymbol{x})\right)^{\mathsf{T}} (\boldsymbol{y} - \boldsymbol{x}) \geq 0 \quad \Longrightarrow \quad f(\boldsymbol{y}) \geq f(\boldsymbol{x}) \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}. \tag{2.47}$$

This implies that any stationary point, defined according to (2.9), of a pseudoconvex function f on the set $\mathcal{X}$ is a global minimizer of f over $\mathcal{X}$ [Man94, Thm. 9.3.3]. Moreover, the pseudoconvex function can be equivalently defined by the implication:

$$f(\boldsymbol{y}) < f(\boldsymbol{x}) \quad \Longrightarrow \quad \left(\nabla f(\boldsymbol{x})\right)^{\mathsf{T}} (\boldsymbol{y} - \boldsymbol{x}) < 0 \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}. \tag{2.48}$$

That is, $f(\boldsymbol{y}) < f(\boldsymbol{x})$ implies that $\boldsymbol{y} - \boldsymbol{x}$ is a descent direction of f [Man94, Thm. 9.3.5].

Definition 2.24 (Strongly convex function)**.** *[BV04, Sec. 9.1.2] A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be strongly convex on the set $\mathcal{X} \subseteq \mathbb{R}^n$ with parameter $m > 0$ if $\mathcal{X}$ is convex and*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \left(\nabla f(\boldsymbol{x})\right)^{\mathsf{T}} (\boldsymbol{y} - \boldsymbol{x}) + \frac{m}{2}\|\boldsymbol{y} - \boldsymbol{x}\|_2^2 \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X},\, \forall \alpha \in [0,1]. \tag{2.49}$$

Similar to the property (2.7) of convex functions, if f is twice differentiable, the strong convexity condition (2.49) implies that

$$\nabla^2 f(\boldsymbol{x}) - m\boldsymbol{I} \succeq 0. \tag{2.50}$$

The relationship among different degrees of the generalized concepts of convexity is summarized in Fig. 2.2 where the arrow indicates the direction of the implication.

2.3 Complex Differentiation

In this section, we present some concepts and facts in complex differentiation, including the gradient, since we will deal with complex variables in the problem studied in Chapter 5. In particular, we focus on the real-valued functions with complex arguments.

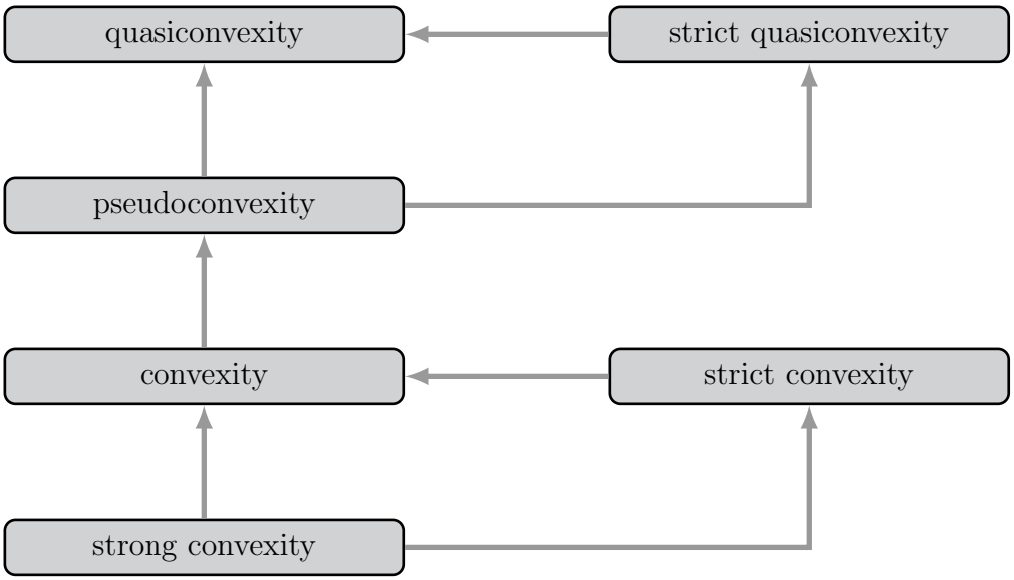


Figure 2.2. Relationship among different degrees of convexity [YP17].

First, recall that the first-order Taylor expansion of a function $f(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x}^o \in \mathbb{R}^n$ is given with the gradient as

$$f(\boldsymbol{x}^o) + (\nabla f(\boldsymbol{x}))^{\mathsf{T}} (\boldsymbol{x} - \boldsymbol{x}^o). \tag{2.51}$$

Now consider a real-valued function $f(z)$ defined on the complex domain $\mathbb{C}$. By expressing the complex argument z in terms of its real and imaginary parts as

$$z = x + \mathrm{j}y \quad \text{with } x, y \in \mathbb{R}, \tag{2.52}$$

the function f can be regarded as a real-valued function defined on $\mathbb{R}^2$, i.e., $f(x, y)$, which has the following first-order Taylor expansion at the point $z^o = x^o + \mathrm{j}y^o$:

$$f(x^o, y^o) + \frac{\partial f(x^o, y^o)}{\partial x}(x - x^o) + \frac{\partial f(x^o, y^o)}{\partial y}(y - y^o). \tag{2.53}$$

In other words, a real-valued function with complex arguments can always be rewritten as a function with real arguments of a double size, and then, its variation and stationarity can be studied in the expanded real variable space. However, in the following, we introduce a more compact definition of the gradient directly in the complex variable space based on the Wirtinger derivatives. The first-order Taylor expansion in (2.53) can be equivalently expressed with the complex argument z and its conjugate $\overline{z}$ as

$$f(z^o) + \frac{\partial f(z^o)}{\partial z}(z - z^o) + \frac{\partial f(z^o)}{\partial \overline{z}} \left(\overline{z} - \overline{z}^o\right), \tag{2.54}$$

where $\frac{\partial}{\partial z}$ and $\frac{\partial}{\partial \overline{z}}$ are the Wirtinger partial derivative operators [Wir27, Say14] for a

complex variable defined as

$$\begin{cases} \frac{\partial}{\partial z} = \frac{1}{2}\left(\frac{\partial}{\partial x} - \mathrm{j}\frac{\partial}{\partial y}\right), \\ \frac{\partial}{\partial \overline{z}} = \frac{1}{2}\left(\frac{\partial}{\partial x} + \mathrm{j}\frac{\partial}{\partial y}\right). \end{cases} \tag{2.55}$$

Observe that, as f takes real values, we have

$$\frac{\partial f(z)}{\partial \overline{z}} = \overline{\left(\frac{\partial f(z)}{\partial z}\right)}.$$

Thus, the first-order Taylor expansion in (2.54) can be simplified to

$$f(z^o) + \Re\left(2\overline{\left(\frac{\partial f}{\partial \overline{z}}\right)}(z - z^o)\right).$$

Similarly, for a real-valued function $f(\boldsymbol{z})$ defined on a complex vector space $\mathbb{C}^n$, the first-order Taylor expansion at $\boldsymbol{z}^o \in \mathbb{C}^n$ can be written in a compact form as

$$f(\boldsymbol{z}^o) + \sum_{i=1}^{n}\left\{\frac{\partial f(\boldsymbol{z}^o)}{\partial z_i}(z_i - z_i^o) + \frac{\partial f(\boldsymbol{z}^o)}{\partial \overline{z}_i}(\overline{z}_i - \overline{z}_i^o)\right\} = f(\boldsymbol{z}^o) + \Re\left(2\left(\frac{\partial f(\boldsymbol{z}^o)}{\partial \overline{\boldsymbol{z}}}\right)^{\mathsf{H}}(\boldsymbol{z} - \boldsymbol{z}^o)\right), \tag{2.56}$$

where

$$\begin{cases} \frac{\partial}{\partial \boldsymbol{z}} = \left[\frac{\partial}{\partial z_1}, \ldots, \frac{\partial}{\partial z_n}\right]^{\mathsf{T}}, \\ \frac{\partial}{\partial \overline{\boldsymbol{z}}} = \left[\frac{\partial}{\partial \overline{z}_1}, \ldots, \frac{\partial}{\partial \overline{z}_n}\right]^{\mathsf{T}}. \end{cases} \tag{2.57}$$

Therefore, instead of taking all partial derivatives $\left(\frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}}, \frac{\partial f(\boldsymbol{z})}{\partial \overline{\boldsymbol{z}}}\right) \in \mathbb{C}^{2n}$, we define the gradient of f at $\boldsymbol{z}$ as

$$\nabla f(\boldsymbol{z}) = 2\frac{\partial f(\boldsymbol{z})}{\partial \overline{\boldsymbol{z}}} \in \mathbb{C}^n, \tag{2.58}$$

based on the compact expression (2.56) of the first-order Taylor expansion. Moreover, with the above definition of the gradient, the first-order Taylor expansion at $\boldsymbol{z}^o$ can be expressed in a unified compact form

$$f(\boldsymbol{z}^o) + \Re\left((\nabla f(\boldsymbol{z}^o))^{\mathsf{H}}(\boldsymbol{z} - \boldsymbol{z}^o)\right), \tag{2.59}$$

irrespectively the variable $\boldsymbol{z}$ taking values in $\mathbb{R}^n$ or $\mathbb{C}^n$. In consequence, the theory of subdifferentiation, as well as the concepts of stationarity established on it, presented in Section 2.1 can be readily extended to real-valued functions with complex arguments with the definition of gradient in (2.58) and the expression (2.59) of the first-order Taylor expansion.

Chapter 3
Prior Work on Optimization Frameworks

In this chapter, we provide a short review of several related existing algorithmic frameworks for optimization, including the majorization-minimization (MM) framework in Section 3.1, the successive convex approximation (SCA) framework in Section 3.2, and their block-coordinatewise extensions in Section 3.3. Those algorithms, as well as the algorithmic frameworks proposed later in Chapter 4, can be classified as approximation-based methods in the sense that they attempt to solve a difficult optimization problem by converting it into a sequence of simpler surrogate/approximate problems. The main difference among them lies in the construction of the approximate problems. Generally speaking, there are two main desiderata of the approximate function, namely, the tightness to the original objective function and the low computational complexity of minimizing the approximate function.

3.1 Majorization-Minimization

In the majorization-minimization (MM) algorithmic framework, also known as the successive upper-bound minimization (SUM) algorithm [RHL13], the approximate function is designed to be a global upper bound of the objective function and tangent to it at the current point so that the minimizer of the approximate function also provides a decrease of the original objective function.

Consider the general constrained optimization problem in (1.1). Let $\boldsymbol{x}^{(t)}$ be the approximate solution at the tth iteration. In each iteration we first construct a surrogate function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right) : \mathcal{X} \to \mathbb{R}$ that majorizes the original objective function f at the current point $\boldsymbol{x}^{(t)}$ according to the following definition.

Definition 3.1 (Majorization)**.** *The function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ is said to be a majorizer of the function $f(\boldsymbol{x})$ at the point $\boldsymbol{x}^{(t)}$ if it satisfies the following two conditions:*

$$\text{(Tangent condition)} \qquad \widehat{f}\left(\boldsymbol{x}^{(t)};\boldsymbol{x}^{(t)}\right) = f\left(\boldsymbol{x}^{(t)}\right), \tag{3.1}$$

$$\text{(Domination condition)} \qquad \widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right) \geq f(\boldsymbol{x}) \text{ for all } \boldsymbol{x} \in \mathcal{X}. \tag{3.2}$$

In other words, the graph of the majorizing function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ lies above the graph of the original function f and is tangent to it at the point $\boldsymbol{x} = \boldsymbol{x}^{(t)}$. Then the variable $\boldsymbol{x}$

is updated by minimizing the majorizing function

$$\boldsymbol{x}^{(t+1)} \in \operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}} \quad \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right). \tag{3.3}$$

The procedure of MM is visualized in Figure 3.1.

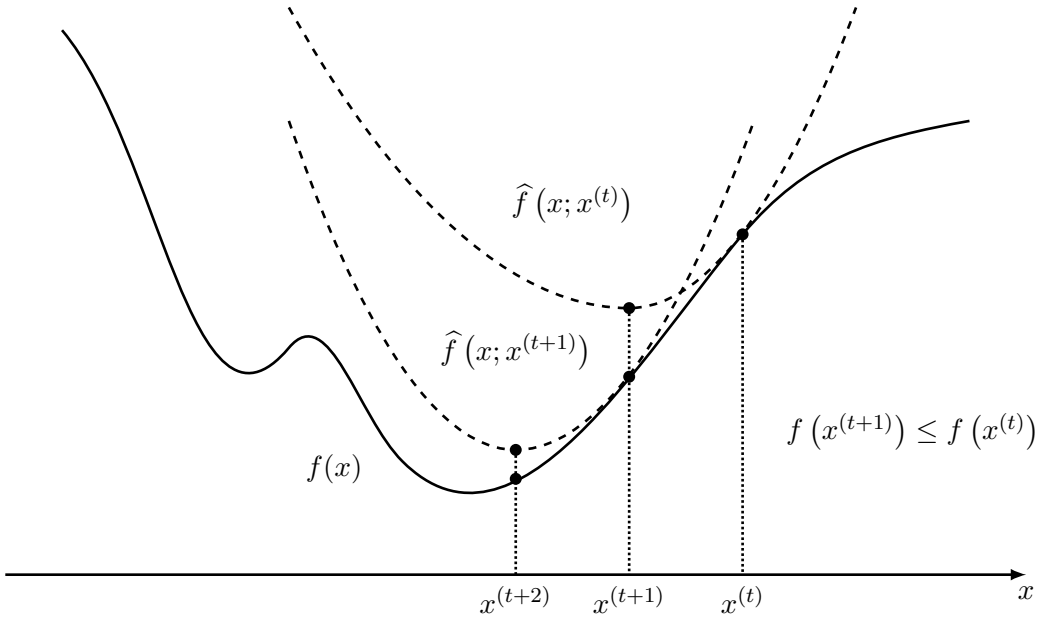


Figure 3.1. General principle of the MM algorithm [SBP17].

Due to the tangent and domination conditions, a decrease of the original function f is obtained through the minimization of the majorizing function.

Proposition 3.1 (Nonincreasing monotone sequence of objective values by the MM algorithm)**.** *If $\widehat{f}(\cdot;\cdot)$ is a majorization of the objective function f by Definition 3.1 and $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ is a solution sequence generated by the MM algorithm according to* (3.3)*, then the sequence of objective function values is monotonically nonincreasing, i.e.,*

$$f\left(\boldsymbol{x}^{(t+1)}\right) \leq \widehat{f}\left(\boldsymbol{x}^{(t+1)}; \boldsymbol{x}^{(t)}\right) \leq \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) = f\left(\boldsymbol{x}^{(t)}\right). \tag{3.4}$$

Thus, the sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ monotonically decreases the original function f and iteratively approaches a minimum point of f.

Consider the general case where f is not necessarily smooth but directionally differentiable everywhere in $\mathcal{X}$. It is proven in [RHL13] that the MM algorithm converges to a stationary point of the original problem (1.1) under the following assumptions, including the majorization conditions in Definition 3.1.

Assumption 3.1. *Let the approximate function $\widehat{f}(\cdot;\cdot)$ satisfy the following assumptions:*

1) $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ is continuous in $(\boldsymbol{x},\boldsymbol{y})$ for all $\boldsymbol{x},\boldsymbol{y}\in\mathcal{X}$;

2) Tangency: $\widehat{f}(\boldsymbol{y};\boldsymbol{y}) = f(\boldsymbol{y})$ for all $\boldsymbol{y}\in\mathcal{X}$;

3) Upper bound: $\widehat{f}(\boldsymbol{x};\boldsymbol{y}) \geq f(\boldsymbol{y})$ for all $\boldsymbol{x},\boldsymbol{y}\in\mathcal{X}$;

4) Derivative consistency: $\widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} = f'_{\boldsymbol{d}}(\boldsymbol{y})$ for all $\boldsymbol{d}$ with $\boldsymbol{y}+\boldsymbol{d}\in\mathcal{X}$.

Note that the derivative consistency condition in Assumption 3.1 equivalently requires that the majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ preserves the whole generalized subdifferential defined in (2.21) of the original function f at the tangent point $\boldsymbol{y}$, i.e., $\underline{\partial} f(\boldsymbol{y})$, when $\underline{\partial} f(\boldsymbol{y})$ is not empty. Then the following convergence statement is established in [RHL13] for the MM algorithm.

Theorem 3.1 (Convergence of the MM algorithm)**.** *Provided that Assumption 3.1 is satisfied, every limit point of the solution sequence generated by the MM algorithm is a d-stationary point, defined in* (2.23)*, of the problem* (1.1)*.*

Remark 3.1. *The property stated in Theorem 3.1 is usually considered in the literature as the asymptotic convergence of an iterative algorithm. However, Theorem 3.1 only implies that, if the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ is convergent, it must converge to a stationary point of the problem* (1.1)*. Theorem 3.1 does not say anything about the convergence of the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$, which requires that all its subsequences converge to the same unique limit point. Similarly, for the other optimization frameworks presented in this thesis, we analyze their asymptotic convergence only in terms of the stationarity of limit points of the generated solution sequence. Nevertheless, as we consider only the case where the problem* (1.1) *attains the minimum, which implies that f is bounded below, we have the following trivial convergence of the objective function in the strict sense. If an algorithm generates a monotonically nonincreasing sequence of objective values, like Proposition 3.1, then, by the monotone convergence theorem [Ber16, Prop. A.3], this sequence of objective values converges to its minimum. Moreover, the continuity and coercivity of f in Assumption 1.1 ensure that any sublevel set of f is compact, which further implies that the whole solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ is in a compact set since it monotonically decreases f. Then, according to the Bolzano–Weierstrass theorem [MN07, Ch.4, Thm. 7], the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ always admits a limit point. The Corollary 1 in [RHL13] claims a convergence of the MM framework stronger than Theorem 3.1, under an additional assumption that $\boldsymbol{x}$*

is initialized in a compact sublevel set of f, which also implies that the whole solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ is in a compact set. Specifically, it claims that the distance from the generated solution sequence to the set of the d-stationary points of the problem (1.1) *asymptotically converges to zero.*[1] *However, the above corollary in [RHL13] is not correct because the proof does not account for the fact that, according to the Bolzano–Weierstrass theorem, the compactness of a sequence only ensures the existence of a convergent subsequence, i.e., the existence of a limit point, not the convergence of the whole sequence.*

In the special case where both the original function $f(\boldsymbol{x})$ and the majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ are differentiable in $\boldsymbol{x}$ and the feasible set $\mathcal{X}=\mathbb{R}^n$, the derivative consistency condition 4) in Assumption 3.1 is enforced by the majorization conditions, which can be shown as follows. The majorization conditions 2) and 3) imply that

$$\frac{\widehat{f}(\boldsymbol{y}+\varepsilon\boldsymbol{d};\boldsymbol{y})-\widehat{f}(\boldsymbol{y};\boldsymbol{y})}{\varepsilon}\geq\frac{f(\boldsymbol{y}+\varepsilon\boldsymbol{d})-f(\boldsymbol{y})}{\varepsilon}$$

for any direction $\boldsymbol{d}$ and $\varepsilon>0$. Taking limits for $\varepsilon\to 0$ produces

$$\widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}}\geq f'_{\boldsymbol{d}}(\boldsymbol{y})\quad\forall\,\boldsymbol{d}. \tag{3.5}$$

Since both $f(\boldsymbol{x})$ and $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ are differentiable in $\boldsymbol{x}$, we have

$$\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})\geq\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}f(\boldsymbol{y}). \tag{3.6}$$

The inequality (3.5) naturally also holds for the opposite direction $-\boldsymbol{d}$, which leads to the result

$$-\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})\geq-\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}f(\boldsymbol{y}). \tag{3.7}$$

Then, combining (3.6) and (3.7), we can conclude that the derivative consistency condition 4) in Assumption 3.1 is satisfied as

$$\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})=\boldsymbol{d}^{\mathsf{T}}\nabla_{\boldsymbol{x}}f(\boldsymbol{y})\quad\forall\,\boldsymbol{d}. \tag{3.8}$$

Moreover, the result in (3.8) also implies the consistency of gradient between the original function $f(\boldsymbol{x})$ and the majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ at the point $\boldsymbol{y}$, i.e., $\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})=\nabla_{\boldsymbol{x}}f(\boldsymbol{y})$, for all $\boldsymbol{y}$.

Remark 3.2 (Inexact MM)**.** *Strictly speaking, the exact minimization of the majorizing function is not required in order to obtain a nonincreasing sequence of objective function values since* (3.4) *depends only on decreasing the majorizing function, not minimizing it. Instead of minimizing exactly $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$, one can find a point $\boldsymbol{x}^{(t+1)}$ that satisfies $\widehat{f}\left(\boldsymbol{x}^{(t+1)};\boldsymbol{x}^{(t)}\right)\leq\widehat{f}\left(\boldsymbol{x}^{(t)};\boldsymbol{x}^{(t)}\right)$. This extension is referred to as the generalized MM algorithm [DLR77, MK07] or the inexact MM algorithm [BP18].*

[1] This property still does not imply the convergence of the solution sequence unless the problem (1.1) has a unique d-stationary point.

In fact, the monotonic descent property (3.4) requires neither the exact minimization of the majorization nor the derivative consistency between the majorization and the original objective function. Those conditions are required to ensure that the solution sequence stops at, or asymptotically approaches, a stationary point of the original problem. However, as we will see in the next chapter, those requirements can be relaxed in order to construct an approximate problem that can be more efficiently solved.

The survey paper [LWLZ21] provides a unified treatment of the convergence of MM algorithms in different scenarios, including results for diverse classes of nonconvex nonsmooth problems and nonasymptotic analysis. A comprehensive overview of MM algorithms can be found in the book [Lan16] and the survey papers [SBP17, Ngu17], together with various techniques for constructing a majorizing function that have been considered in the literature. In fact, many other algorithms can be viewed as special cases of MM generated with different majorization techniques, such as the well-known expectation-minimization (EM), proximal algorithms, concave-convex procedures (CCCP), and so on [SBP17]. Most of the majorization techniques are derived from the properties of convex functions and we introduce below one of them that is involved in the discussions in the rest of this thesis.

Consider the class of functions f that can be decomposed as

$$f(\boldsymbol{x}) = f_0(\boldsymbol{x}) + f_{\text{ccv}}(\boldsymbol{x}), \tag{3.9}$$

where f_{ccv} is a differentiable concave function. Recall the inequality in (2.6) that a convex function is minorized by its supporting hyperplane. Thus, a majorization of f in (3.9) at a point $\boldsymbol{y}$ can be obtained by linearizing the concave component f_{ccv} as follows:

$$f(\boldsymbol{x}) \leq f_0(\boldsymbol{x}) + f_{\text{ccv}}(\boldsymbol{y}) + \left(\nabla f_{\text{ccv}}(\boldsymbol{y})\right)^{\mathsf{T}} (\boldsymbol{x} - \boldsymbol{y}). \tag{3.10}$$

As a special case, if f_0 is convex, the minimization of f in (3.9) becomes an instance of DC (Difference of Convex functions) program [LTPD18, LTPD23].In this case, the resulting majorizing function in (3.10) is also convex and the MM algorithms generated by this majorization technique are called concave-convex procedures (CCCP) [YR03, QD11, LB16]. Furthermore, the supporting hyperplane inequality also implies that, if there exists a differentiable convex function $f_{\text{cvx}}(\boldsymbol{x})$ that preserves the function value and the gradient of the concave function $f_{\text{ccv}}(\boldsymbol{x})$ at $\boldsymbol{y}$, then f_{cvx} is also a majorization of f_{ccv} at $\boldsymbol{y}$ as

$$f_{\text{ccv}}(\boldsymbol{x}) \leq f_{\text{ccv}}(\boldsymbol{y}) + \left(\nabla f_{\text{ccv}}(\boldsymbol{y})\right)^{\mathsf{T}} (\boldsymbol{x} - \boldsymbol{y}) \leq f_{\text{cvx}}(\boldsymbol{x}). \tag{3.11}$$

Thus, the function f in (3.9) can be majorized by

$$f(\boldsymbol{x}) \leq f_0(\boldsymbol{x}) + f_{\text{cvx}}(\boldsymbol{x}). \tag{3.12}$$

In some scenarios, especially when f_0 is nonconvex, a convex upper bound for the concave component f_{ccv} is preferred to a tighter linear one, since it can be used to generate a convex overall majorizing function, whose minimizer can be obtained at a lower computational cost. Of course, depending on the specific structure of the concave function, there may also exists a concave majorization, which is tighter than the linear one. Note that both the majorization techniques in (3.10) and (3.12) satisfy the derivative consistency condition in Assumption 3.1 because of the differentiability of f_{ccv}.

3.2 Successive Convex Approximation

In practice, it may not be easy to construct a global upper bound for the objective function unless it possesses certain convexity/concavity structure. An alternative approach is the successive convex approximation (SCA) framework [YP17, SFL17a, SS18], where the approximate function is not required to be a global upper bound of the original objective function but a convex function. By relaxing the global bounding requirement, we can construct an approximate problem that can be solved at a lower computational cost, even in a parallel or distributed manner.

We first consider the case where the objective function f is smooth. At the tth iteration, we first solve the following approximate problem:

$$\widehat{\boldsymbol{x}}^{(t)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \quad \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right), \tag{3.13}$$

where $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ is a convex approximation of f at the point $\boldsymbol{x}^{(t)}$. More generally, only a weaker form of convexity, namely, pseudoconvexity defined in Section 2.2, is required on the approximate function $\widehat{f}$ for the convergence of the algorithm. The solution $\widehat{\boldsymbol{x}}^{(t)}$ does not always decrease the original function f. However, under the conditions in Assumption 3.2, the vector $\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}$ indicates a descent direction of f, i.e.,

$$\left(\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}} \left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right) < 0. \tag{3.14}$$

Then the variable $\boldsymbol{x}$ is updated along the descent direction as

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)} \left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right) \tag{3.15}$$

with a suitable step size $\gamma^{(t)} \in (0, 1]$. The next iterate $\boldsymbol{x}^{(t+1)}$ obtained by (3.15) is located on the line segment between $\boldsymbol{x}^{(t)}$ and $\widehat{\boldsymbol{x}}^{(t)}$ and, hence, is ensured to belong to the feasible set $\mathcal{X}$ due to the convexity of $\mathcal{X}$. The following approaches are often employed to determine the step size $\gamma^{(t)}$ [SFL17a, RHL13, YP17].

1. **Constant step size:** The step size $\gamma^{(t)}$ is fixed at all iterations.

2. **Diminishing step size:** The step size $\gamma^{(t)}$ is chosen so that

$$\lim_{t\to\infty} \gamma^{(t)} = 0 \quad \text{and} \quad \sum_{t=0}^{\infty} \gamma^{(t)} = \infty.$$

3. **Line search:** The line search approach finds a step size $\gamma^{(t)}$ that ensures a sufficient decrease of the original function f. Hence, similar to the MM algorithm, a nonincreasing sequence of objective function values is obtained in this approach, i.e.,

$$f\left(\boldsymbol{x}^{(t+1)}\right) \leq f\left(\boldsymbol{x}^{(t)}\right). \tag{3.16}$$

In the following, we briefly introduce two widely used line search methods, namely the **exact line search** and the **successive line search** (also known as **backtracking line search**). In the exact line search, the step size $\gamma^{(t)}$ is chosen such that the original function f is minimized along the descent direction $\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}$. That is, $\gamma^{(t)}$ is the optimal solution of the following optimization problem:

$$\gamma^{(t)} \in \operatorname*{argmin}_{\gamma\in[0,1]} \quad f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right). \tag{3.17}$$

If the optimal solution of problem (3.17) can not be computed efficiently, the successive line search can alternatively be employed, which successively decreases the step size until a required amount of decrease on the original function f is achieved. A commonly used stopping criterion for the successive line search is the **Armijo rule**. In the Armijo rule, we successively try step sizes $\gamma \in \{\beta^0, \beta^1, \ldots\}$, i.e., a geometric sequence with a constant decrease rate $0 < \beta < 1$, until we find the smallest $k \in \mathbb{N}$ such that $\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})$ with $\gamma = \beta^k$ satisfies the following sufficient decrease condition:

$$f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) - f\left(\boldsymbol{x}^{(t)}\right) \leq \gamma\sigma\delta$$
$$\text{with } \delta = \left(\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}} \left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right), \tag{3.18}$$

where $0 < \sigma < 1$ and δ represents the directional derivative of f in the descent direction $\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}$. In other words, the step size is obtained by solving the following optimization problem:

$$\gamma^{(t)} = \operatorname*{argmax}_{\gamma} \gamma \quad \text{s.t. } \gamma \in \{\beta^k \mid k \in \mathbb{N}\} \text{ and (3.18)}. \tag{3.19}$$

Common choices for the parameters are $\beta = 0.5$ and $\sigma = 0.001$.

In this thesis we focus on the line search approach. In this case, i.e., when the descent property (3.16) holds, the convergence of the SCA algorithm is demonstrated in [YP17] under the following regularity conditions. The solution set of the approximate problem (3.13), which is a set-valued map of the point $\boldsymbol{x}^{(t)}$, is denoted by

$$M\left(\boldsymbol{x}^{(t)}\right) = \left\{ \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}}\, \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) \right\}. \tag{3.20}$$

Assumption 3.2. *We make the following assumptions:*

1) *f is continuously differentiable on $\mathcal{X}$;*
2) *$\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ is continuously differentiable in $\boldsymbol{x} \in \mathcal{X}$ for any given $\boldsymbol{y} \in \mathcal{X}$ and continuous in $\boldsymbol{y} \in \mathcal{X}$ for any given $\boldsymbol{x} \in \mathcal{X}$;*
3) *Gradient consistency: $\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y}; \boldsymbol{y}) = \nabla_{\boldsymbol{x}} f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;*
4) *$\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ is pseudoconvex in $\boldsymbol{x} \in \mathcal{X}$ for any given $\boldsymbol{y} \in \mathcal{X}$;*
5) *The map $M\left(\boldsymbol{x}^{(t)}\right)$ is nonempty for $t \in \mathbb{N}$;*
6) *Given any convergent subsequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathcal{T}}$ with $\mathcal{T} \subseteq \mathbb{N}$, the sequence $\left(\widehat{\boldsymbol{x}}^{(t)}\right)_{t\in\mathcal{T}}$ is bounded.*

In particular, the gradient consistency ensures that any fixed point $\boldsymbol{y} \in M(\boldsymbol{y})$ is a stationary point of the original problem. If a fixed point is not achieved in a finite number of iterations, with the conditions 5 and 6, the asymptotic convergence of the SCA framework is established in the following theorem [YP17].

Theorem 3.2 (Convergence of the SCA algorithm)**.** *Suppose that Assumption 3.2 is satisfied and that the line search approach is employed. Then every limit point of the solution sequence generated by the SCA algorithm is a stationary point of the problem* (1.1)*.*

As for a fixed or diminishing step size, the convergence of the SCA algorithm is established with additional assumptions including that the original function f is Lipschitz differentiable and that the approximate function $\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ is strongly convex in $\boldsymbol{x}$. See [SFL17a] for the details of the additional regularity conditions and the proof of convergence in this case. Furthermore, if the convex approximation $\widehat{f}(\boldsymbol{x}; \boldsymbol{x}^{(t)})$ is also designed to be a majorization of f at $\boldsymbol{x}^{(t)}$, then the algorithm can also be viewed as an instance of the MM algorithm. In this case, the constant step size $\gamma^{(t)} = 1$ can be employed as it already ensures the descent property (3.16), and the convergence of the algorithm is justified under the MM framework.

3.2.1 Extension to Structured Nonsmooth Problems

The procedure and convergence result of the SCA algorithm described above are established under the smoothness condition on the objective function. However, all the preceding results can be easily generalized to the following class of structured nonsmooth problems [YP17, SFL17a]. Consider the following modification of the smooth problem in (1.1):

$$\min_{\boldsymbol{x}\in\mathcal{X}} \quad f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{3.21}$$

where the additional regularization term g is convex but possibly nondifferentiable. Typical examples for g are $\|\cdot\|_1$ and $\|\cdot\|_2$. In particular, the problem (3.21) can be rewritten as an instance of problem (1.1) with the help of an auxiliary variable as follows:

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^n,\, y\in\mathbb{R}} \quad & f(\boldsymbol{x}) + y \\ \text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y. \end{aligned} \tag{3.22}$$

The objective function in (3.22) is smooth and the feasible set $\mathcal{C} = \{(\boldsymbol{x}, y) \in \mathbb{R}^{n+1} \mid \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y\}$ is closed and convex due to the convexity of g. Thus, the preceding algorithmic procedure and convergence result can be readily applied to the reformulated problem (3.22). In particular, the approximate problem around $(\boldsymbol{x}^{(t)}, y^{(t)})$ at the tth iteration is

$$\left(\widehat{\boldsymbol{x}}^{(t)}, \hat{y}^{(t)}\right) = \underset{\boldsymbol{x}\in\mathbb{R}^n, y\in\mathbb{R}}{\operatorname{argmin}} \ \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + y \quad \text{s.t. } \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y, \tag{3.23}$$

where only the smooth function $f(\boldsymbol{x})$ needs to be replaced by its approximation $\widehat{f}(\boldsymbol{x}; \boldsymbol{x}^{(t)})$ constructed to satisfy Assumption 3.2 except for the pseudoconvexity condition 4). Instead, to ensure that Assumption 3.2 is also satisfied for the overall objective function of the approximate problem (3.23), most importantly, to ensure the pseudoconvexity of the overall objective function, we make the following stricter assumption on the approximate function $\widehat{f}$:

Assumption 3.3. *$\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ is convex in $\boldsymbol{x} \in \mathcal{X}$ for any given $\boldsymbol{y} \in \mathcal{X}$.*

The convexity of the approximate function $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ ensures that the whole objective function $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + y$ in (3.23) is convex and, hence, pseudoconvex in $(\boldsymbol{x}, y)$. Moreover, the gradient consistency between the original and the approximate function at $\left(\boldsymbol{x}^{(t)}, y^{(t)}\right)$ can be easily verified as follows

$$\begin{aligned} \nabla_{\boldsymbol{x}} \left(\widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) + y^{(t)}\right) &= \nabla_{\boldsymbol{x}} \left(f\left(\boldsymbol{x}^{(t)}\right) + y^{(t)}\right), \\ \nabla_{y} \left(\widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) + y^{(t)}\right) &= \nabla_{y} \left(f\left(\boldsymbol{x}^{(t)}\right) + y^{(t)}\right) = 1. \end{aligned}$$

Therefore, the overall approximate function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)+y$ also satisfies Assumption 3.2. The approximate problem (3.23) can be rewritten as

$$\widehat{\boldsymbol{x}}^{(t)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \quad \widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right) + g(\boldsymbol{x}), \tag{3.24}$$

and the solution for the auxiliary variable $\hat{y}^{(t)} = g\left(\widehat{\boldsymbol{x}}^{(t)}\right)$ as y needs to be minimized. Again, only the smooth component f is replaced by a convex approximation, whereas the convex regularization term g remains unaltered. Then the variables $\boldsymbol{x}$ and y are updated along the descent direction by

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)}\left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right), \tag{3.25a}$$

$$y^{(t+1)} = y^{(t)} + \gamma^{(t)}\left(\hat{y}^{(t)} - y^{(t)}\right). \tag{3.25b}$$

Note that the fact that $y^{(t)} \geq g\left(\boldsymbol{x}^{(t)}\right)$ and $\hat{y}^{(t)} = g\left(\widehat{\boldsymbol{x}}^{(t)}\right)$ ensures that $y^{(t+1)} \geq g\left(\boldsymbol{x}^{(t+1)}\right)$, due to the Jensen's inequality of the convex function g [BV04]. On the other hand, a lower value of $f(\boldsymbol{x})+y$ can always be achieved by further reducing $y^{(t+1)}$ to $g\left(\boldsymbol{x}^{(t+1)}\right)$. That is, we can replace the update rule (3.25b) by the enhanced rule

$$y^{(t+1)} = g\left(\boldsymbol{x}^{(t+1)}\right) \quad \forall\, t \in \mathbb{N} \tag{3.26}$$

without destroying the monotonic decrease of the objective function value provided by the line search method. Consequently, as shown in the above procedure, the auxiliary variable y does not need to be calculated explicitly in the SCA algorithm for the class of nonsmooth problem in (3.21).

Furthermore, a modified line search method is developed in [YP17] for the nonsmooth problem (3.21) based on the equivalent reformulation in (3.22) to find a suitable step size $\gamma^{(t)}$ for the variable updating at a lower computational cost. The modifications on the exact line search and the successive line search, respectively, are presented as follows.

1. **Exact line search:** The exact line search carried out over the original nondifferentiable objective function in (3.21) is given by

$$\gamma^{(t)} \in \underset{\gamma\in[0,1]}{\operatorname{argmin}} \quad f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) + g\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right), \tag{3.27}$$

   which is computationally demanding since g is nondifferentiable. To reduce the computational complexity, we can instead apply the exact line search to the reformulated smooth objective function in (3.22), which leads to the following optimization problem:

$$\gamma^{(t)} \in \underset{\gamma\in[0,1]}{\operatorname{argmin}} \quad f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) + \left(y^{(t)} + \gamma(\hat{y}^{(t)} - y^{(t)})\right). \tag{3.28}$$

Removing the auxiliary variable y according to the update rule in (3.25b), we obtain

$$\gamma^{(t)} \in \underset{\gamma\in[0,1]}{\operatorname{argmin}} \quad f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) + \gamma\left(g\left(\widehat{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)\right), \tag{3.29}$$

where the constant $g\left(\boldsymbol{x}^{(t)}\right)$ is omitted. This is equivalent to replacing g, as a function of γ, in (3.27) by the following linear majorization on the interval $[0, 1]$:

$$g\left(\boldsymbol{x}^{(t)}\right) + \gamma\left(g\left(\widehat{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)\right), \tag{3.30}$$

which is constructed based on the Jensen's inequality of convex functions [BV04]. Hence, the step size $\gamma^{(t)}$ obtained by (3.29) also ensures a decrease of the original objective function in (3.21) according to the principle of the MM algorithm.

2. **Successive line search (Armijo rule):** Similarly, we can apply the successive line search to the smoothed objective function in (3.29) to achieve a decrease of the original objective function at a lower computational cost. Specifically, the step size γ is successively reduced until the following stopping criterion is satisfied:

$$\begin{aligned} &f\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) - f\left(\boldsymbol{x}^{(t)}\right) + \gamma\left(g\left(\widehat{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)\right) \leq \gamma\sigma\delta \\ &\qquad \text{with } \delta = \left(\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}}\left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right) + g\left(\widehat{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right). \end{aligned} \tag{3.31}$$

Another variant of the Armijo rule for the class of nondifferentiable problem in (3.21) is also often used, which directly evaluates the decrease of the original objective function, instead of the smoothed majorization, to avoid the poor approximation of the linear majorization in (3.30) in some scenarios [HSDR14, YT11, TY09]. Define $h(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x})$. The step size γ is successively reduced until the following stopping criterion is satisfied:

$$\begin{aligned} &h\left(\boldsymbol{x}^{(t)} + \gamma(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)})\right) - h\left(\boldsymbol{x}^{(t)}\right) \leq \gamma\sigma\delta \\ &\qquad \text{with } \delta = \left(\nabla_{\boldsymbol{x}} f(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}}\left(\widehat{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right) + g\left(\widehat{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right), \end{aligned} \tag{3.32}$$

where the original objective function h is evaluated and only the required sufficient decrement is calculated according to the smoothed majorization in (3.29)

3.2.2 Separable Convex Approximation

The SCA framework includes many other algorithms as special cases when different approximate functions are employed. In this thesis, we are interested in separable approximation techniques since, in many applications, the constraints are separable such

that the optimization variables can be decomposed into several independent blocks, as shown in (2.41). This block structure can often be exploited for designing low-complexity algorithms that can be implemented in a parallel or distributed manner, e.g., on modern multi-core DSPs, GPUs and FPGAs as well as in cloud computing networks.

Let us consider a modified version of the minimization problem in (2.41) with an additional convex nonsmooth regularization term $g(\boldsymbol{x})$ that is also separable across the blocks of variables. Consider the following class of problems:

$$\begin{aligned} \min_{\{\boldsymbol{x}_k\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) + \underbrace{\sum_{k=1}^K g_k(\boldsymbol{x}_k)}_{g(\boldsymbol{x})} \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k, k = 1, \ldots, K, \end{aligned} \tag{3.33}$$

where f is smooth but possibly nonconvex, each $g_k(\boldsymbol{x}_k)$ is convex in the corresponding block variable $\boldsymbol{x}_k$ but not necessarily smooth, and the feasible set has a Cartesian product structure with $\mathcal{X}_k$ being closed and convex for all $k = 1, \ldots, K$. As discussed in the previous subsection, the regularization term g will remain unchanged due to its convexity, and only the smooth component f is approximated by a convex function. To take advantage of the separable structure of the constraints, we can employ the following type of approximation:

$$\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) = \sum_{k=1}^K \widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right), \tag{3.34}$$

where each component $\widehat{f}_k$ is a coordinatewise convex approximation of f that satisfies the gradient consistency at $\boldsymbol{x}^{(t)}$ along the coordinates of $\boldsymbol{x}_k$, i.e., $\nabla_{\boldsymbol{x}_k} \widehat{f}_k\left(\boldsymbol{x}_k^{(t)}; \boldsymbol{x}^{(t)}\right) = \nabla_{\boldsymbol{x}_k} f\left(\boldsymbol{x}^{(t)}\right)$. It is not difficult to verify that the overall approximation $\widehat{f}(\boldsymbol{x}; \boldsymbol{x}^{(t)})$ is convex in $\boldsymbol{x}$ and preserves the gradient of f at $\boldsymbol{x}^{(t)}$ due to the separability of the gradient. The advantage of this type of approximation is that the resulting approximate problem (3.24) can be decomposed into the following K subproblems:

$$\widehat{\boldsymbol{x}}_k \in \left\{ \operatorname*{argmin}_{\boldsymbol{x}_k \in \mathcal{X}_k} \widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right) + g_k(\boldsymbol{x}_k) \right\} \quad \forall\, k = 1, \ldots, K. \tag{3.35}$$

Each subproblem in (3.35) exclusively depends on a single block variable and, hence, can be solved independently. The separability of the approximate problem makes this algorithm suitable for implementation on modern parallel or distributed architectures.

In the following, we briefly introduce several techniques for constructing a separable convex approximation in the form (3.34).

- **Nonlinear Jacobi algorithm:** Consider that the function f has a blockwise convexity structure. That is, f is convex in each block variable $\boldsymbol{x}_k$ separately, but not jointly. In this case, the nonlinear Jacobi algorithm [YP17, SFS+14, BT15] can be used to construct a separable convex approximation. The convex approximation of f with respect to each block variable $\boldsymbol{x}_k$ is simply chosen to be

$$\widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right) = f\left(\boldsymbol{x}_k, \boldsymbol{x}_{-k}^{(t)}\right), \tag{3.36}$$

  where all the other block variables except $\boldsymbol{x}_k$, contained by $\boldsymbol{x}_{-k} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{k-1}, \boldsymbol{x}_{k+1}, \ldots, \boldsymbol{x}_K)$, are fixed at their value at the current iterate. In other words, the overall approximation $\widehat{f}$ in (3.34) captures the variation of f along the coordinates of each block variable $\boldsymbol{x}_k$, respectively, when the other blocks are fixed.

- **Quadratic approximation:** For a general function f, the quadratic approximation can be employed, which corresponds to the Newton-type methods [Ber16]. The general expression for a quadratic approximation of f along the coordinates of the block $\boldsymbol{x}_k$ is given by

$$\begin{aligned}\widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right) = f\left(\boldsymbol{x}^{(t)}\right) &+ \left(\nabla_{\boldsymbol{x}_k} f\left(\boldsymbol{x}^{(t)}\right)\right)^{\mathsf{T}} \left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\right) \\ &+ \frac{1}{2}\left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\right)^{\mathsf{T}} \boldsymbol{H}_k\left(\boldsymbol{x}_k^{(t)}\right)\left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\right).\end{aligned} \tag{3.37}$$

  The Hessian $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right) \in \mathbb{R}^{n_k \times n_k}$ of the quadratic approximation $\widehat{f}_k$, which possibly depends on $\boldsymbol{x}^{(t)}$, is designed to be an approximation of the partial Hessian $\nabla^2_{\boldsymbol{x}_k} f(\boldsymbol{x}^{(t)})$ of the original function f at $\boldsymbol{x}^{(t)}$ restricted to the coordinates of $\boldsymbol{x}_k$, if existent. Note that, although the convexity of $\widehat{f}_k$ only requires the positive semidefiniteness of $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right)$, $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right)$ is usually designed to be positive definite to avoid that $\widehat{f}_k$ is unbounded from below. The reason is that the function $\widehat{f}_k$ restricted to the line defined by an eigenvector of $\boldsymbol{H}_k(\boldsymbol{x}^{(t)})$ associated with a zero-eigenvalue becomes linear. If f is partially strictly convex in the block $\boldsymbol{x}_k$, i.e., $\nabla^2_{\boldsymbol{x}_k} f(\boldsymbol{x}^{(t)}) \succ 0$, one may simply choose $\boldsymbol{H}_k(\boldsymbol{x}^{(t)}) = \nabla^2_{\boldsymbol{x}_k} f(\boldsymbol{x}^{(t)})$, and (3.37) becomes the second-order Taylor expansion of f at $\boldsymbol{x}^{(t)}$ restricted to $\boldsymbol{x}_k$.

  Moreover, the quadratic approximation can be used to construct a convex approximation that also majorizes the original function by choosing $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right) = \boldsymbol{M}$, if exists, such that $\boldsymbol{M} - \nabla^2_{\boldsymbol{x}_k} f(\boldsymbol{x}^{(t)}) \succeq 0$ for all $\boldsymbol{x}_k \in \mathcal{X}_k$. In particular, when the partial gradient $\nabla_{\boldsymbol{x}_k} f(\boldsymbol{x})$ is not necessarily differentiable but Lipschitz continuous with a constant $L_k > 0$, a convex quadratic majorizer can be obtained by choosing $\boldsymbol{H}_k(\boldsymbol{x}^{(t)}) = L_k \boldsymbol{I}_k$.

  Another popular subclass of quadratic approximation is

$$\widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right) = f\left(\boldsymbol{x}^{(t)}\right) + \left(\nabla_{\boldsymbol{x}_k} f\left(\boldsymbol{x}^{(t)}\right)\right)^{\mathsf{T}} \left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\right) + \frac{1}{2\tau_k}\left\|\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\right\|_2^2, \tag{3.38}$$

i.e., with $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right) = \frac{1}{\tau_k}\boldsymbol{I}_k$ and $\tau_k > 0$, which is the idea of the proximal gradient method [PB14, BT15]. The proximal-type approximation economizes on the computational complexity compared to the Taylor expansion, due to the diagonal structure of the Hessian. The parameter τ_k is often interpreted as a step size in the proximal gradient method. The reason is that, the minimization of only $\widehat{f}_k$ admits the solution $\boldsymbol{x}_k^{(t)} - \tau_k \nabla_{\boldsymbol{x}_k} f\left(\boldsymbol{x}^{(t)}\right)$, which is equivalent to an gradient descent update with a step size τ_k.

Without increasing the computational complexity of the approximate problem, one may generalize the Hessian in (3.38) to an arbitrary positive diagonal matrix so that part of the information of the second-order variations of the original function f can be preserved in the quadratic approximation. As an example, we retain the diagonal structure of $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right)$ but choose the diagonal entries to be

$$\left[\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right)\right]_i = \max\left\{\frac{\partial^2 f(\boldsymbol{x}^{(t)})}{\partial x_{k,i}^2}, \varepsilon\right\} \quad \text{for } i = 1, \ldots, n_k, \tag{3.39}$$

where $x_{k,i}$ is the ith element of the block $\boldsymbol{x}_k$ and ε is some positive value to ensure the positive definiteness of $\boldsymbol{H}_k\left(\boldsymbol{x}^{(t)}\right)$.

The above approximation techniques can certainly be used in a hybrid manner. For example, one may choose the Jacobi-type approximation for the blocks in which the original function f is partially convex and use the quadratic approximation for the other blocks. Also, a simple proximal term $\frac{1}{2\tau_k}\|\boldsymbol{x}_k - \boldsymbol{x}_k^{(t)}\|_2^2$ may be added to the Jacobi-type approximation (3.36) so that the approximate problem admits a unique solution, if required. More convex approximation techniques can be found in [YP17, YPLO20, SFL17a, SS18].

3.3 Block Coordinate Descent and Inexact Block Coordinate Descent

3.3.1 Block Coordinate Descent

Consider the previously introduced class of optimization problems in (2.41) with separable constraints. Another class of method that exploits the separable structure of the constraints is the block coordinate descent (BCD) method [Ber16, BT15, OR70], which appears in the literature under other different names such as the nonlinear Gauss-Seidel

method [BT15] and the alternating minimization method. The BCD method is one of the first variable decomposition methods for solving the above problem. In contrast to the nonlinear Jacobi algorithm in (3.36), where the partial minimizations with respect to the different block variables $\boldsymbol{x}_k$ are performed simultaneously, the nonlinear Gauss-Seidel algorithm minimizes the objective function with respect to each block variable successively in a predetermined or adaptive order. Specifically, at the tth iteration of the algorithm, the selected block variable, say $\boldsymbol{x}_k$, is updated by its nonlinear best-response

$$\boldsymbol{x}_k^{(t+1)} = \underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} \; f\left(\boldsymbol{x}_k, \boldsymbol{x}_{-k}^{(t)}\right), \tag{3.40}$$

whereas the rest of the variables are fixed to their values of the preceding iteration, i.e.,

$$\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)} \quad \forall\, i \neq k. \tag{3.41}$$

Since the block variables are updated sequentially, the newest value of the other block variables is always used in the update (3.40) of each block variable, which sometimes leads to faster convergence than the parallel-update counterpart – the Jacobi algorithm. Also because of its simple implementation, the BCD method has been employed in various applications such as image denoising and reconstruction [CZ97], dynamic programming [HS75, LK70, ZW89], and power allocation in wireless communication systems [SRLH11]. Especially for convex problems, the partial minimization in (3.40) can usually be solved at a much lower computational cost than the original problem (2.41), and sometimes even admits a closed-form solution. However, when f is nonconvex, it often becomes difficult to solve the per block subproblem (3.40) exactly. Moreover, convergence of the BCD method typically requires that f is (pseudo)convex or that the partial minimization in (3.40) has a unique solution. A summary of the convergence results of the BCD methods in various scenarios is provided in [Tse01]. Otherwise, a classic example of Powell [Pow73] shows that, due to the lack of convexity, the solution sequence generated by the BCD method does not approach any of the stationary points of the problem.

Similar to the MM and SCA frameworks described in the previous sections, the convergence of BCD also relies on the monotonic decrement of the objective function. This, as mentioned in Remark 3.2, does not strictly require the exact solution of the subproblem (3.40). Therefore, to overcome the aforementioned drawbacks of the BCD method, especially for nonconvex problems, one may instead minimize a well-structured approximation of the objective function with respect to a block variable at each iteration, which is referred to as the inexact BCD method. In the following subsections, we briefly introduce two inexact BCD methods, i.e., the extensions of the MM and SCA frameworks, respectively, in a BCD manner, where the approximate functions

are designed to be global upper bounds of the original objective function and convex, respectively.

As a trivial generalization of the preceding BCD procedure, one may update more than one block variable by minimizing the objective function or its approximation jointly with respect to multiple selected block variables. Moreover, the blocks can be chosen according to a fixed, random, or adaptive order. Let $\mathcal{I}^{(t)} \subseteq \{1, \ldots, K\}$ denote the set of indices of the block variables selected to be updated at iteration t. We list below some of the block selection rules are commonly used in the literature [HRLP16, RHL13, YPLO20]:

1. **Deterministic rules:**

   - **Cyclic rule:** The block variables are updated in a fixed cyclic order, e.g., $\mathcal{I}^{(t)} = \{ \mod (t, K) + 1\}$.
   - **Essentially cyclic rule:** There exists a given period $T \geq 1$ during which each block is updated at least once, i.e.,

   $$\bigcup_{i=1}^{T} \mathcal{I}^{(t+i)} = \{1, \ldots, K\} \quad \forall\, t \in \mathbb{N}. \tag{3.42}$$

   The cyclic rule is a special case of the essentially cyclic rule with $T = K$.

2. **Randomized rule:** At each iteration t, each block is independently and randomly chosen according to some nonzero probability, i.e.,

   $$\Pr\left(k \in \mathcal{I}^{(t)}\right) = p_k^{(t)} \quad \forall\, k = 1, \ldots, K \tag{3.43}$$

   with $\sum_{k=1}^{K} p_k^{(t)} = 1$ and $p_k^{(t)} \geq p_{\min} > 0$. The selections are independent from that made at the previous iterations. Some examples of selection distributions can be found in [DFKS15].

3. **Greedy rules:** Choose the block that may lead to a fast convergence, e.g., the block with the largest gradient (Gauss-Southwell rule) or the block that gives the largest improvement on the objective function value (maximum block improvement rule). More recent greedy rules are proposed in [NSL$^+$15, NLS22].

For the simplicity of presentation, we will describe the following inexact BCD methods with a single block variable updated at each iteration unless otherwise noted.

Finally, we recall the standard property of a limit point of any convergent coordinate-wise scheme, including both exact and inexact BCD methods, that has been mentioned

in Section 2.1. Since the optimization variables are coordinatewise updated, only a coordinatewise stationary point is asymptotically achieved, unless the objective function is coordinatewise regular.

3.3.2 Block Successive Upper-Bound Minimization

The block successive upper-bound minimization (BSUM) framework is an extension of the MM framework in a BCD manner, which successively updates each block variable by minimizing a coordinatewise global upper bound of the original objective function. Consider the optimization problem (2.41) with the separable constraints. At the tth iteration, let $\boldsymbol{x}_k$ be the block variable selected to be updated at the tth iteration, whereas the other block variables are fixed to their values at the current iterate $\boldsymbol{x}^{(t)}$. We first construct an approximate function $\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{x}^{(t)})$ that is a majorization of f at $\boldsymbol{x}^{(t)}$ in the coordinates of $\boldsymbol{x}_k$. Then the selected block variable $\boldsymbol{x}_k$ is updated by minimizing the majorizing function

$$\boldsymbol{x}_k^{(t+1)} \in \operatorname*{argmin}_{\boldsymbol{x}_k \in \mathcal{X}_k} \quad \widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right), \tag{3.44}$$

which also ensures a decrease of the original objective function. Specifically, we make the following assumptions on the coordinatewise approximate functions.

Assumption 3.4. *We make the following assumptions for all $k = 1, \ldots, K$:*

1) *$\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is continuous in $(\boldsymbol{x}_k, \boldsymbol{y})$ for all $\boldsymbol{x}_k \in \mathcal{X}_k$ and $\boldsymbol{y} \in \mathcal{X}$;*
2) *Tangency: $\widehat{f}_k(\boldsymbol{y}_k; \boldsymbol{y}) = f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;*
3) *Upper bound: $\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y}) \geq f(\boldsymbol{x}_k, \boldsymbol{y}_{-k})$ for all $\boldsymbol{x}_k \in \mathcal{X}_k$ and $\boldsymbol{y} \in \mathcal{X}$;*
4) *Derivative consistency: $\left(\widehat{f}_k\right)'_{\boldsymbol{d}}(\boldsymbol{x}_k; \boldsymbol{y})\big|_{\boldsymbol{x}_k = \boldsymbol{y}_k} = f'_{\boldsymbol{d}}(\boldsymbol{y})$ for all $\boldsymbol{d} = (\boldsymbol{0}, \ldots, \boldsymbol{0}, \boldsymbol{d}_k, \boldsymbol{0}, \ldots, \boldsymbol{0})$ with $\boldsymbol{y}_k + \boldsymbol{d}_k \in \mathcal{X}_k$;*
5) *$\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is quasiconvex in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$;*
6) *The subproblem (3.44) has a unique solution for any $\boldsymbol{x}^{(t)} \in \mathcal{X}$.*

The MM algorithm can be viewed as a single-block version of BSUM. Nevertheless, compared to Assumption 3.1 for the MM algorithm, the convergence of the BSUM algorithm additionally requires the quasiconvexity of the coordinatewise approximate functions and the uniqueness of the solution of each subproblem, which is similar to the

convergence conditions on the subproblem (3.40) of the exact BCD method [Tse01]. As an advantage of the inexact BCD method over the exact BCD method, we can properly construct an approximate function to satisfy the aforementioned conditions when the original objective function does not. Then, the convergence of the BSUM algorithm under Assumption 3.4 is proven in [RHL13] for deterministic update rules, which can be stated as follows.

Theorem 3.3 (Convergence of the BSUM algorithm)**.** *[RHL13, Thm. 2] Provided that Assumption 3.4 is satisfied, then every limit point $\boldsymbol{z}$ of the solution sequence generated by the BSUM algorithm is a coordinate d-stationary point of the problem* (2.41)*. In addition, if f is coordinatewise regular at $\boldsymbol{z}$, then $\boldsymbol{z}$ is a d-stationary point of* (2.41)*.*

3.3.3 Block Successive Convex Approximation

As a coordinatewise extension of the SCA framework, the block successive convex approximation (BSCA) framework successively updates each block variable by minimizing a convex approximation of the original objective function in the corresponding coordinates.

We first consider the case where the objective function f in (2.41) is smooth. At the tth iteration, provided that the block variable $\boldsymbol{x}_k$ is chosen to be updated, we first solve the following approximate problem:

$$\widehat{\boldsymbol{x}}_k^{(t)} \in \underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} \quad \widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right), \tag{3.45}$$

where $\widehat{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right)$ is a convex approximation of f at the current iterate $\boldsymbol{x}^{(t)}$ in the coordinates of $\boldsymbol{x}_k$. Under the conditions in Assumption 3.5, the vector $\widehat{\boldsymbol{x}}_k^{(t)} - \boldsymbol{x}_k^{(t)}$ indicates a descent direction of f in the chosen coordinates. Then the blocks of $\boldsymbol{x}$ are updated according to the following rule:

$$\boldsymbol{x}_i^{(t+1)} = \begin{cases} \boldsymbol{x}_k^{(t)} + \gamma^{(t)}\left(\widehat{\boldsymbol{x}}_k^{(t)} - \boldsymbol{x}_k^{(t)}\right) & \text{if } i = k, \\ \boldsymbol{x}_i^{(t)} & \text{otherwise,} \end{cases} \tag{3.46}$$

where $\gamma^{(t)} \in (0, 1]$ is the step size for the updating along the chosen coordinates. Similar to the SCA algorithm, a suitable step size $\gamma^{(t)}$ that ensures a sufficient decrease of the original objective function can be obtained by exact or successive line search described in Section 3.2, along the descent direction in the chosen coordinates.

We denote the solution set of the approximate problem (3.45), which is a set-valued map of the point $\boldsymbol{x}^{(t)}$, by

$$M_k(\boldsymbol{x}^{(t)}) = \left\{ \underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} \, \widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{x}) \right\}. \tag{3.47}$$

The convergence of the BSCA algorithm is proven in [YPLO20] under the following assumptions when the step size is obtained by a line search method.

Assumption 3.5. *We make the following assumptions for all $k = 1, \ldots, K$:*

1) *f is continuously differentiable on $\mathcal{X}$;*
2) *$\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is continuously differentiable in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$ and continuous in $\boldsymbol{y} \in \mathcal{X}$ for any given $\boldsymbol{x}_k \in \mathcal{X}_k$;*
3) *Gradient consistency: $\nabla_{\boldsymbol{x}_k} \widehat{f}_k(\boldsymbol{y}_k; \boldsymbol{y}) = \nabla_{\boldsymbol{x}_k} f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;*
4) *$\widehat{f}(\boldsymbol{x}_k; \boldsymbol{y})$ is strictly convex in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$;*
5) *The map $M_k\left(\boldsymbol{x}^{(t)}\right)$ is nonempty for $t \in \mathbb{N}$.*

Note that, compared to the single-block version – the SCA framework, the convergence of BSCA requires a stricter degree of convexity on the approximate functions. The differentiable functions are shown to be coordinatewise regular at all feasible points $\boldsymbol{x} \in \mathcal{X}$ [Tse01]. Therefore, under Assumption 3.5, the convergence to stationary points can be established for the BSCA algorithm. Moreover, the convergence of the BSCA algorithm is investigated in [YPLO20] for both the deterministic and the randomized update and the result is stated in the following theorem.

Theorem 3.4 (Convergence of the BSCA algorithm)**.** *[YPLO20, Thm. 1] Suppose that Assumption 3.5 is satisfied and that the line search approach is employed. Then every limit point of the solution sequence generated by the BSCA algorithm is a stationary point of the problem* (2.41) *(with probability* 1 *for the randomized update).*

Similarly to the SCA framework, the above procedure and convergence result of the BSCA algorithm can readily be applied to the following class of separable nonsmooth problems:

$$\begin{aligned} \min_{\{\boldsymbol{x}_k\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) + \sum_{k=1}^K g_k(\boldsymbol{x}_k) \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k,\ k = 1, \ldots, K, \end{aligned} \tag{3.48}$$

where f is smooth and the additional regularization terms g_k are convex but possibly nondifferentiable, since the problem (3.48) can be rewritten in the form of (2.41) as follows:

$$\begin{aligned} \min_{\{\boldsymbol{x}_k \in \mathbb{R}^{n_k},\, y_k \in \mathbb{R}\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) + \sum_{k=1}^K y_k \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k,\ g_k(\boldsymbol{x}_k) \leq y_k,\ k = 1, \ldots, K, \end{aligned} \tag{3.49}$$

with a smooth objective function. Then, an approximate function of the nonsmooth objective function in (3.48) in the coordinates of a block variable $\boldsymbol{x}_k$ is obtained by constructing a coordinatewise approximation for the smooth component f according to Assumption 3.5 and retaining the convex regularization term g_k unchanged. We refer the reader to [YPLO20] for the details of the customization of the BSCA algorithm for the class of nonsmooth problems in (3.48). Additionally, since the nonsmooth component of the objective function in (3.48) is separable across the blocks, the objective function is coordinatewise regular at all feasible points $\boldsymbol{x} \in \mathcal{X}$ [Tse01]. Hence, the convergence to a stationary point of the problem (3.48) is provided by the BSCA algorithm.

The updating rule (3.46) in the procedure of the BSCA algorithm described above requires the exact solution $\widehat{\boldsymbol{x}}_k^{(t)}$ of the coordinatewise approximate problem (3.45), which is available when $\widehat{\boldsymbol{x}}_k^{(t)}$ admits a closed-form expression. However, in practice, it is not always possible to construct an approximate problem that admits a closed-form solution. In this case, an iterative algorithm is required for solving the approximate problem (3.45), and this provides only an approximate solution of (3.45) as most iterative algorithms only show asymptotic convergence. Moreover, it is generally difficult to evaluate the accuracy of an approximate solution obtained by an iterative algorithm with respect to the exact solution $\widehat{\boldsymbol{x}}_k^{(t)}$, especially if the approximate function $\widehat{f}_k$ is nonconvex. This hinders the convergence analysis of the BSCA algorithm. To overcome this issue, the authors in [YPLO20] further develop a so-called inexact BSCA algorithm, which finds an approximate solution of the coordinatewise approximate problem (3.45) by running another standard SCA algorithm for a finite number of iterations. We refer the reader to [YPLO20] for the details of the inexact BSCA algorithm and the additional convergence analysis.

Chapter 4
Proposed Frameworks based on Smoothing Majorization

The derivative consistency condition in Assumption 3.1 of the classic MM framework in Section 3.1 requires the directional differentiability of the objective function and the consistency of directional derivatives in all directions between the original objective function and its majorizer at the point where the majorizer is constructed. This condition restricts the majorizer constructed at a nondifferentiable point of the original function to be also nonsmooth. Recall that the principle of the MM framework is to convert a difficult optimization problem into a sequence of simpler approximate problems via majorization. Hence, this restriction hinders its capability of simplifying nonsmooth problems since the minimization of the majorizing function, if restricted to be nonsmooth, may still be difficult.

Therefore, in this chapter, we relax the derivative consistency in the majorization step so that a smooth majorizer that can be easily minimized is permitted for a wide class of nonsmooth problems. Specifically, as a generalization of the majorization technique that we employed in [LTY+22], we consider the situation that the majorizing function preserves only a subgradient of the original objective function. The MM framework with such smoothing majorization, abbreviated as *smoothing MM*, is detailed in Section 4.1, together with its convergence analysis. As a result of this relaxation of derivative consistency, the smoothing MM converges to a stationary point in a more relaxed sense than the classic MM. In other words, compared to the classic MM, the smoothing MM sacrifices the tightness of the convergence set with respect to the local minima in order to construct an approximate problem that can be easily addressed. In some scenarios, the exact minimization of the smooth majorizing function may still be difficult, especially if it is nonconvex. Meanwhile, the smoothness of the majorizing function allows us to employ the idea of SCA in Section 3.2 to obtain an approximate minimizer of the majorizing function efficiently. This motivated our idea in [LTY+22] of combining the smoothing majorization and the separable convex approximation techniques to address the phase retrieval with dictionary learning problem. In Section 4.2, we generalize the algorithms in [LTY+22] to develop an inexact MM framework, named *smoothing SCA*, and provide a unified convergence analysis. Finally, similar to the classic MM and SCA frameworks, the smoothing MM and SCA can also be implemented in a BCD manner to exploit potential separable structures of the constraints in the optimization problem. The block-coordinatewise versions of the smoothing MM and SCA, as well as their convergence analyses, are presented in Section 4.3.

4.1 Smoothing Majorization-Minimization

Consider the general constrained optimization problem (1.1) where the objective function f is nonconvex and nonsmooth, and the feasible set $\mathcal{X}$ is convex. In this section, we relax the requirement on the majorizing function in the classic MM framework so that a smooth approximate problem that can be easily addressed is permitted for nonsmooth problems. Specifically, we consider the following assumptions on the majorizing function $\widehat{f}$ at each iteration.

Assumption 4.1 (Smoothing majorization)**.** *Let the approximate function $\widehat{f}(\cdot;\cdot)$ satisfy the following:*

1) *$\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ is continuously differentiable in $\boldsymbol{x} \in \mathcal{X}$ for any given $\boldsymbol{y} \in \mathcal{X}$ and continuous in $\boldsymbol{y} \in \mathcal{X}$ for any given $\boldsymbol{x} \in \mathcal{X}$;*
2) *Tangency: $\widehat{f}(\boldsymbol{y};\boldsymbol{y}) = f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;*
3) *Upper bound: $\widehat{f}(\boldsymbol{x};\boldsymbol{y}) \geq f(\boldsymbol{y})$ for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$;*
4) *Subgradient consistency: $\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y}) \in \partial^C f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$.*

Compared to Assumption 3.1 of the classic MM algorithm, the above assumption enforces the majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ to be smooth even when f is nondifferentiable at $\boldsymbol{y}$, which is impossible if the derivative consistency condition 4) in Assumption 3.1 needs to be satisfied. To this end, a relaxed subgradient consistency condition is incorporated in Assumption 4.1, which only requires that the gradient $\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})$ of the smooth majorizing function is a C-subgradient of the original function f at $\boldsymbol{y}$. Note that, unlike the derivative consistency condition in Assumption 3.1, this subgradient consistency does not require the directional differentiability of the original function. However, the following property of the directional derivatives can be observed when both the original and majorizing functions are directionally differentiable.

Proposition 4.1. *Let the function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ be a majorization of the function $f(\boldsymbol{x})$ at the point $\boldsymbol{y} \in \mathcal{X}$ according to Definition 3.1. If both $f(\boldsymbol{x})$ and $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ are directionally differentiable at $\boldsymbol{y}$, then we have*

$$\widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} \geq f'_{\boldsymbol{d}}(\boldsymbol{y}) \quad \forall \boldsymbol{d} \text{ with } \boldsymbol{y}+\boldsymbol{d} \in \mathcal{X}. \tag{4.1}$$

Proof. The majorization conditions in Definition 3.1 imply that

$$\frac{\widehat{f}(\boldsymbol{y}+\varepsilon\boldsymbol{d};\boldsymbol{y}) - \widehat{f}(\boldsymbol{y};\boldsymbol{y})}{\varepsilon} \geq \frac{f(\boldsymbol{y}+\varepsilon\boldsymbol{d}) - f(\boldsymbol{y})}{\varepsilon} \tag{4.2}$$

for any direction $\boldsymbol{d}$ and $\varepsilon > 0$ with $\boldsymbol{y} + \varepsilon \boldsymbol{d} \in \mathcal{X}$. Then the property in (4.1) is obtained by taking limits for $\varepsilon \to 0$. □

Hence, compared to Assumption 3.1, which enforces the directional derivatives of the majorizing function to be consistent with that of the original function, Assumption 4.1 relaxes the directional derivatives of the smooth majorizing function to be in the following interval:

$$f'_{\boldsymbol{d}}(\boldsymbol{y}) \leq \widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} = \Big(\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\Big)^{\mathsf{T}} \boldsymbol{d} \leq f^{\circ}_{\boldsymbol{d}}(\boldsymbol{y}) \quad \forall\, \boldsymbol{y} \in \mathcal{X}, \boldsymbol{d} \text{ with } \boldsymbol{y} + \boldsymbol{d} \in \mathcal{X}, \tag{4.3}$$

which comes from Proposition 4.1 and the Definition 2.16 of the Clarke subdifferential.

Remark 4.1. *Suppose that the original function f is directionally differentiable. Then, the subgradient consistency condition 4) in Assumption 4.1 is automatically satisfied for all interior points $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$ if all the other conditions in Assumption 4.1 are satisfied. The proof is as follows. For an interior point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$, the bounding property in* (4.1) *holds for all directions $\boldsymbol{d} \in \mathbb{R}^n$. Due to the smoothness of the majorization, this implies that*

$$-\Big(\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\Big)^{\mathsf{T}} \boldsymbol{d} = -\widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} \leq -f'_{\boldsymbol{d}}(\boldsymbol{y}) \leq f^{\circ}_{\boldsymbol{d}}(\boldsymbol{y}) \quad \forall \boldsymbol{d} \in \mathbb{R}^n, \tag{4.4}$$

where the last inequality comes from the definition of Clarke directional derivative in (2.25)*. By the definition of Clarke subdifferential in* (2.26)*, we conclude from* (4.4) *that the subgradient consistency is satisfied for any interior point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$, i.e., $\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y}) \in \partial^C f(\boldsymbol{y})$.*

To ensure the existence of a smooth majorizer that satisfies Assumption 4.1, some additional assumptions on the original objective function are required. As an example, for an interior point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$, the bounding property in (4.3) holds for all directions $\boldsymbol{d} \in \mathbb{R}^n$, which implies that the original function f must satisfy

$$f'_{-\boldsymbol{d}}(\boldsymbol{y}) \leq -\Big(\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\Big)^{\mathsf{T}} \boldsymbol{d} \leq -f'_{\boldsymbol{d}}(\boldsymbol{y}) \tag{4.5}$$

by considering both directions $\boldsymbol{d}$ and $-\boldsymbol{d}$. More generally, the following property is observed when a smooth majorizer satisfying Assumption 4.1 exists.

Proposition 4.2 (Existence of smooth majorizer)**.** *Let the function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ be a majorizer of a locally Lipschitz function $f(\boldsymbol{x})$ at a point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$ that satisfies Assumption 4.1. If f is nondifferentiable at $\boldsymbol{y}$ then f is not regular at $\boldsymbol{y}$.*

Proof. Assume that the gradient $\nabla_{\boldsymbol{x}} f(\boldsymbol{y})$ does not exist. Provided that f is regular at $\boldsymbol{y}$ according to Definition 2.17, then the Clarke subdifferential $\partial^C f(\boldsymbol{y})$ coincides with

the classic convex subdifferential at $\boldsymbol{y}$ and the C-directional derivative $f^{\circ}_{\boldsymbol{d}}(\boldsymbol{y})$ coincides with the ordinary directional derivative $f'_{\boldsymbol{d}}(\boldsymbol{y})$ for any direction $\boldsymbol{d}$. In this case, the bounding property in (4.3) reduces to the consistency of the directional derivatives between the original and majorizing functions as

$$\left(\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})\right)^{\mathsf{T}}\boldsymbol{d} = f'_{\boldsymbol{d}}(\boldsymbol{y}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n, \tag{4.6}$$

which implies that the gradient of f exists at $\boldsymbol{y}$ and $\nabla_{\boldsymbol{x}} f(\boldsymbol{y}) = \nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})$. This contradicts the assumption that the gradient $\nabla_{\boldsymbol{x}} f(\boldsymbol{y})$ does not exist. Therefore, f is not regular at $\boldsymbol{y}$. □

Proposition 4.2 suggests that it is impossible to construct a smooth majorizing function for f at an interior point of $\mathcal{X}$ where f is nondifferentiable and regular. Hence, to ensure the existence of a smooth majorizer at any point on $\mathcal{X}$, we further make the following assumption on the objective function f.

Assumption 4.2. *Let the objective function f satisfy the following:*

1) f is locally Lipschitz on $\mathcal{X}$;

2) f is not regular at any point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$ where f is nondifferentiable.

In Figure 4.1, we present an example function to illustrate the differences between the majorizing function constructed in the classic MM and that in the proposed smoothing MM, as well as the conditions for the existence of a smooth majorizer at a nondifferentiable point derived above. The univariate function $f(x) = \min\{\frac{-x+1}{2}, \max\{\frac{-x-1}{2}, -x\}\}$ in Figure 4.1 is smooth everywhere else except for the two points $x = -1$ and $x = 1$. In particular, f is locally concave at $x = -1$ and locally convex at $x = 1$, and the Clarke subdifferentials at both points are equal and given by $\partial^C f(-1) = \partial^C f(1) = \left[-1, -\frac{1}{2}\right]$. For instance, we construct quadratic upper bounds for f since f is piecewise linear. As f is not regular at $x = -1$, we can construct a smooth majorizer $\widehat{f}(x;-1) = \frac{1}{2}x^2 + \frac{1}{2}$ for f at $x = -1$ that satisfies the relaxed subgradient consistency, which preserves only the directional derivative of f in the positive direction but not that in the negative direction. On the contrary, f is regular at $x = 1$ and, hence, there exists no smooth majorizer. At this point, we give an example majorizer $\widehat{f}(x;1) = \max\left\{\frac{1}{2}(x-2)^2 - \frac{3}{2}, \frac{1}{2}(x-\frac{3}{2})^2 - \frac{9}{8}\right\}$ that satisfies the strict derivative consistency in the classic MM. It demonstrates that the majorizer is also nondifferentiable at this nondifferentiable point of the original function when it is required to preserve the directional derivatives of the original function in all directions.

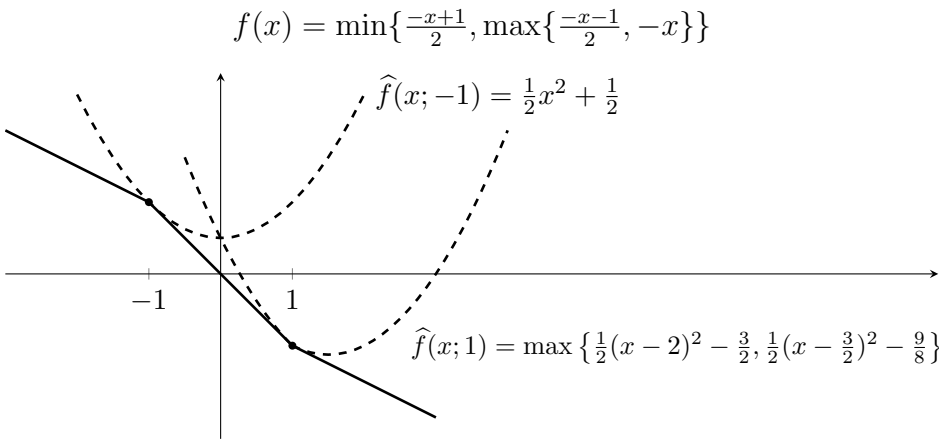


Figure 4.1. Comparison between the strict derivative consistency in the classic MM and the relaxed subgradient consistency in the smoothing MM. A smooth majorizer for f exists at $x = -1$ but not at $x = 1$.

4.1.1 Convergence Analysis

Now we can establish the convergence of the MM algorithm with smoothing majorization as follows based on the generalized concepts of stationarity introduced in Section 2.1.

Theorem 4.1. *Provided that Assumptions 4.1 and 4.2 are satisfied, every limit point of the solution sequence generated by the MM algorithm is a C-stationary point of the problem* (1.1).

Proof. Recall Proposition 3.1 that a nonincreasing sequence of objective function values is obtained through the minimization of the majorizing function, i.e.,

$$f\left(\boldsymbol{x}^{(0)}\right) \geq f\left(\boldsymbol{x}^{(1)}\right) \geq f\left(\boldsymbol{x}^{(2)}\right) \geq \cdots . \tag{4.7}$$

Assume that there exists a subsequence $\left(\boldsymbol{x}^{(t_j)}\right)_{j\in\mathbb{N}}$ converging to a limit point $\boldsymbol{z} \in \mathcal{X}$, i.e., $\lim_{j\to\infty} \boldsymbol{x}^{(t_j)} = \boldsymbol{z}$. Then,

$$\begin{aligned} \widehat{f}\left(\boldsymbol{x}^{(t_{j+1})}; \boldsymbol{x}^{(t_{j+1})}\right) &= f\left(\boldsymbol{x}^{(t_{j+1})}\right) \leq f\left(\boldsymbol{x}^{(t_j+1)}\right) && (4.8)\\ &\leq \widehat{f}\left(\boldsymbol{x}^{(t_j+1)}; \boldsymbol{x}^{(t_j)}\right) \leq \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t_j)}\right) \quad \forall\, \boldsymbol{x} \in \mathcal{X}. && (4.9)\end{aligned}$$

Taking limits for $j \to \infty$ produces

$$\widehat{f}\left(\boldsymbol{z}; \boldsymbol{z}\right) \leq \widehat{f}\left(\boldsymbol{x}; \boldsymbol{z}\right) \quad \forall\, \boldsymbol{x} \in \mathcal{X}. \tag{4.10}$$

The limit point $\boldsymbol{z}$ is a global minimizer of the approximate function $\widehat{f}(\boldsymbol{x};\boldsymbol{z})$ and, hence, satisfies the C-stationarity condition in Definition 2.18, i.e.,

$$\mathbf{0} \in \partial^C \widehat{f}(\boldsymbol{z};\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{z}), \tag{4.11}$$

where $\partial^C \widehat{f}(\boldsymbol{z};\boldsymbol{z}) = \left\{\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{z};\boldsymbol{z})\right\}$ by the smoothness assumption 1) in Assumption 4.1. The subgradient consistency condition in Assumption 4.1 implies that

$$\partial^C \widehat{f}(\boldsymbol{z};\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{z}) \subseteq \partial^C f(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{z}). \tag{4.12}$$

Combining (4.11) and (4.12), we obtain

$$\mathbf{0} \in \partial^C f(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{z}), \tag{4.13}$$

which implies that the limit point $\boldsymbol{z}$ is a C-stationary point of the original problem (1.1). □

We remark that, as shown by (4.12), if $\boldsymbol{z}$ is a C-stationary point of the smooth majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{z})$ on $\mathcal{X}$, then it is also a C-stationary point of the original function f on $\mathcal{X}$. However, the converse is not always true unless the Clarke subdifferential $\partial^C f(\boldsymbol{z})$ is a singleton, i.e., $\partial^C f(\boldsymbol{z})$ contains only one element. In the general case, assume that $\boldsymbol{z}$ is a C-stationary point of f and that $\partial^C f(\boldsymbol{z})$ is not a singleton. Then it is only guaranteed that there exists a subgradient $\boldsymbol{s} \in \partial^C f(\boldsymbol{z})$ such that the majorization $\widehat{f}(\boldsymbol{x};\boldsymbol{z})$ is also C-stationary at $\boldsymbol{z}$ if $\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{z};\boldsymbol{z}) = \boldsymbol{s}$. In other words, whether a C-stationary point $\boldsymbol{z}$ of the original function f is also a C-stationary point of the majorization $\widehat{f}(\boldsymbol{x};\boldsymbol{z})$ depends on the choice of the subgradient in $\partial^C f(\boldsymbol{z})$ for constructing $\widehat{f}(\boldsymbol{x};\boldsymbol{z})$. In particular, the following property is observed.

Proposition 4.3 (Stationary points that are not local minima)**.** *Let the function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ be a majorizer of a locally Lipschitz function $f(\boldsymbol{x})$ at an interior point $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$ that satisfies Assumption 4.1. Suppose that $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ is stationary at $\boldsymbol{y}$ and that f is nondifferentiable at $\boldsymbol{y}$. Then there exists a descent direction $\boldsymbol{d}$ of f at $\boldsymbol{y}$, i.e., $f'_{\boldsymbol{d}}(\boldsymbol{y}) < 0$. In other words, $\boldsymbol{y}$ is a C-stationary point, but not a d-stationary point and, hence, not a local minimum point, of f.*

Proof. Since the differentiable function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ is stationary at $\boldsymbol{y} \in \operatorname{int}(\mathcal{X})$, we have

$$\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y}) = \mathbf{0}. \tag{4.14}$$

Then Proposition 4.1 suggests that

$$f'_{\boldsymbol{d}}(\boldsymbol{y}) \leq \widehat{f}'_{\boldsymbol{d}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} = \left(\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\right)^{\mathsf{T}} \boldsymbol{d} = 0 \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n. \tag{4.15}$$

Assume that f has no descent direction at $\boldsymbol{y}$, i.e., $f'_{\boldsymbol{d}}(\boldsymbol{y}) = 0$ for all $\boldsymbol{d} \in \mathbb{R}^n$. This implies that f is differentiable at $\boldsymbol{y}$ and that $\nabla_{\boldsymbol{x}} f(\boldsymbol{y}) = \boldsymbol{0}$, which contradicts with the assumption that f is nondifferentiable at $\boldsymbol{y}$. Therefore, there exists at least one direction $\boldsymbol{d}$ at $\boldsymbol{y}$ such that $f'_{\boldsymbol{d}}(\boldsymbol{y}) < 0$. □

As mentioned in Section 2.1, the stationarity based on the Clarke subdifferential may contain stationary points that are not local minima. Proposition 4.3 reveals a subset of stationary points that are not local minima and also provides a strategy for avoiding them. Recall that we define the set-valued map

$$M\left(\boldsymbol{x}^{(t)}\right) = \left\{ \operatorname*{argmin}_{\boldsymbol{x} \in \mathcal{X}} \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) \right\}. \tag{4.16}$$

Assume that the MM algorithm has achieved a fixed point $\boldsymbol{z}$ in the interior of the feasible set $\mathcal{X}$, i.e., $\boldsymbol{z} \in \operatorname{int}(\mathcal{X})$ and $\boldsymbol{z} \in M(\boldsymbol{z})$. As a global minimizer of the majorizing function $\widehat{f}(\boldsymbol{x}; \boldsymbol{z})$, $\boldsymbol{z}$ is a stationary point of $\widehat{f}(\boldsymbol{x}; \boldsymbol{z})$, and hence, a C-stationary point of the original function f. By Proposition 4.3, if f is nondifferentiable at $\boldsymbol{z}$, then f does not attain a local minimum at $\boldsymbol{z}$ and, instead, it admits at least one descent direction. Thus, to avoid terminating the algorithm at $\boldsymbol{z}$, a descent update for the variable $\boldsymbol{x}$ can be obtained by the following procedure. Note that, since f is nondifferentiable at $\boldsymbol{z}$, the Clarke subdifferential $\partial^C f(\boldsymbol{z})$ is not a singleton [Cla90, Prop. 2.2.4]. If the zero vector is not the only subgradient in $\partial^C f(\boldsymbol{z})$ that produces a smooth majorizer of f at $\boldsymbol{z}$, then, a descent update from $\boldsymbol{z}$ can be easily obtained by constructing another majorizer at $\boldsymbol{z}$ with a nonzero subgradient in $\partial^C f(\boldsymbol{z})$. However, according to (4.3), all subgradients $\boldsymbol{s}$ in $\partial^C f(\boldsymbol{z})$ that can generate a smooth majorizing function satisfying Assumption 4.1 are given by the solution set of the following system of linear inequalities

$$f'_{\boldsymbol{d}}(\boldsymbol{z}) \leq \boldsymbol{s}^{\mathsf{T}} \boldsymbol{d} \leq f^{\circ}_{\boldsymbol{d}}(\boldsymbol{z}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n, \tag{4.17}$$

which may be unique. If the zero vector is the unique subgradient in $\partial^C f(\boldsymbol{z})$ that produces a smooth majorizer of f at $\boldsymbol{z}$, one can still find a descent direct of f at $\boldsymbol{z}$ by random search and then a decrease of f can be obtained by performing a line search method described in Section 3.2 on f along the obtained descent direction. As a special case, $\boldsymbol{s} = \boldsymbol{0}$ is the unique solution of (4.17) if there exists a set of n linear independent directions $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_n \in \mathbb{R}^n$ with

$$f'_{\boldsymbol{d}_i}(\boldsymbol{z}) = f'_{-\boldsymbol{d}_i}(\boldsymbol{z}) = 0 \quad \forall\, i = 1, \ldots, n. \tag{4.18}$$

Define the matrix $\boldsymbol{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_n]$, which has full rank. By considering both directions $\boldsymbol{d}_i$ and $-\boldsymbol{d}_i$ for $i = 1, \ldots, n$, the condition (4.18) leads to the system of linear equations $\boldsymbol{D}^{\mathsf{T}} \boldsymbol{s} = \boldsymbol{0}$ with the unique solution $\boldsymbol{s} = \boldsymbol{0}$, as part of the system of linear inequalities in (4.17). Nevertheless, under Assumption 4.2, it is difficult to find an example function

that satisfies (4.18) at a nondifferentiable interior point. For the future work, it is of interest to investigate whether this case is excluded by Assumption 4.2. Moreover, we remark that, in a practical application, it can often be directly observed whether there exist multiple subgradients that can produce a smooth majorizing function at a nondifferentiable point without the verification of additional conditions such as (4.18).

In Figure 4.2, we illustrate the type of nondifferentiable stationary points in Proposition 4.3 and the procedure described above with the univariate function f_2 in Figure 2.1(b), which satisfies Assumption 4.2. At the nondifferentiable point $x = 0$, the Clarke subdifferential of f_2 is given by $\partial^C f_2(0) = [-1, 0]$, which is not a singleton. Assume that $x^{(t)} = 0$. Since $0 \in \partial^C f_2(0)$, f_2 is C-stationary at $x^{(t)}$ and we may construct a smooth majorizer of f_2 at $x^{(t)}$ that is also stationary at $x^{(t)}$, e.g., $\widehat{f}_2(x; x^{(t)}) = 2x^2$. However, $x = 0$ is not a local minimum of f_2. For the simple example function f_2, the Clarke subdifferential can be evaluated analytically and, therefore, we can directly avoid choosing a smooth majorizer at $x = 0$ that preserves the subgradient 0, which is usually impossible in practical applications. In practical applications, Proposition 4.3 is used to identify this type of C-stationary points that are not local minima. Then, since $\partial^C f(0)$ is not a singleton, instead of terminating the algorithm at $x = 0$, at the next iteration, we can construct another smooth majorizer of f_2 at $x = 0$ that preserves a nonzero subgradient of f_2, e.g., $\widehat{f}_2(x; x^{(t+1)}) = 2(x - \frac{1}{4})^2 - \frac{1}{8}$, which can provide a decrease of f_2.

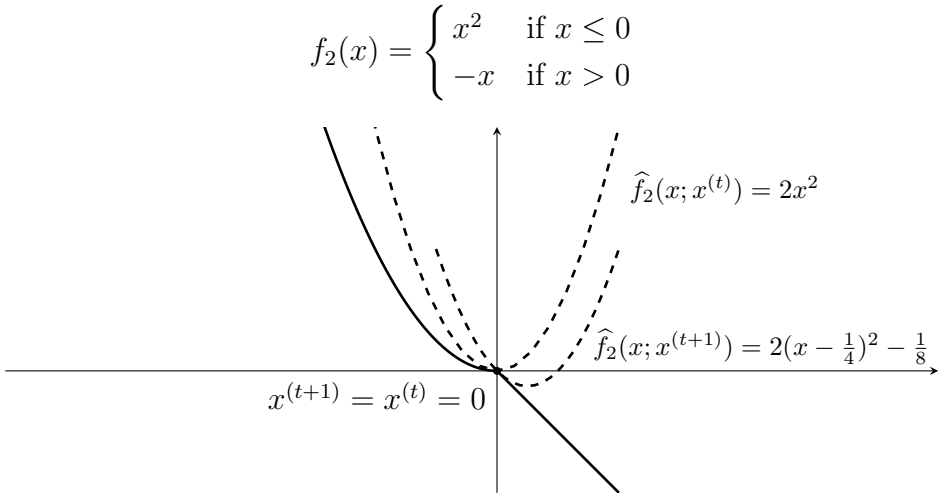


Figure 4.2. Example function for illustrating Proposition 4.3 and the motivated termination condition in Algorithm 1. For $f_2 : \mathbb{R} \to \mathbb{R}$, we have $\partial^C f_2(0) = [-1, 0]$ and any subgradient in $\partial^C f_2(0)$ can be used to construct a smooth majorizer of f_2 at $x = 0$.

The proposed MM framework with smoothing majorization for the problem (1.1) is outlined in Algorithm 1, including the termination condition described above.

Algorithm 1: The MM Algorithm with Smoothing Majorization for Solving Problem (1.1)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Construct a smooth majorizer $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$;
  Let $\boldsymbol{x}^{(t+1)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$;
  $t \leftarrow t + 1$;
until *converged* **and** *($\boldsymbol{x}^{(t)} \notin \operatorname{int}(\mathcal{X})$* **or** *$\nabla_{\boldsymbol{x}} f\left(\boldsymbol{x}^{(t)}\right)$ exists)*;
return $\boldsymbol{x}^{(t)}$

A weaker result similar to Proposition 4.3 is presented in [LWLZ21, Prop. 11]. It is stated that, for an unconstrained minimization problem, if the majorization is restricted to be differentiable, then every limit point $\boldsymbol{z}$ of the MM algorithm is critical for the original objective function f in the sense that

$$\mathbf{0} \in \partial^F(-f)(\boldsymbol{z}), \tag{4.19}$$

where $\partial^F(-f)(\boldsymbol{z})$ is the Fréchet subdifferential [LSM20,Kru03,MNY06,Mor06] of $-f$ at $\boldsymbol{z}$ and coincides with the subdifferential (2.21) constructed with the directional derivatives for locally Lipschitz functions [Kru03, Prop. 1.38]. Similarly to the stationarity based on the other generalized definitions of subdifferential introduced in Section 2.1, (4.19) is only a necessary condition for f attaining a local maximum at $\boldsymbol{z}$. In contrast, in Proposition 4.3, we further conclude the existence of a descent direction at $\boldsymbol{z}$ in the case where f is nondifferentiable at $\boldsymbol{z}$.

Remark 4.2 (Comparison to the classic MM framework)**.** *The classic MM framework introduced in Section 3.1 requires the consistency of the directional derivatives between the original objective function (possibly nonsmooth) and its majorizer in all directions at the current iterate, and provides the convergence to a d-stationary point of the problem. In contrast, the proposed MM algorithm with smoothing majorization sacrifices the tightness of the convergence set with respect to the local minima in order to construct a surrogate problem that can be easily addressed. In practical applications, the differentiable points of the objective function are usually also strictly differentiable, where the C-stationarity is equivalent to the d-stationarity. On the other hand, the nondifferentiable points of the objective function in the interior of the feasible set are excluded from the local minima according to Proposition 4.3. Therefore, this relaxation of the convergence set caused by the smoothing majorization is rather harmless. Whereas the traditional MM framework is broadly applicable to problems with directionally differentiable objective functions, the smoothing majorization technique is limited to functions meeting the criteria outlined in Assumption 4.1.*

4.1.2 Smoothing Majorization Techniques for Nonsmooth Functions

In the following, we present two majorization techniques that can be used to construct a smooth majorizing function satisfying the subgradient consistency condition in Assumption 4.1 for nonsmooth functions.

Let us consider generally the class of functions f that can be decomposed as

$$f(\boldsymbol{x}) = u(\boldsymbol{x}) + v(\boldsymbol{x}) \tag{4.20}$$

where both u and v are locally Lipschitz, u is smooth but not necessarily convex, and v is nonsmooth. Due to the smoothness of u, we only need to find a smooth majorizer for the nonsmooth component v. More specifically, the following property holds.

Proposition 4.4. *Let $\widehat{v}(\boldsymbol{x};\boldsymbol{y})$ be a smooth majorizing function of v in (4.20) at the point y that satisfies Assumption 4.1. Then the function $\widehat{f}(\boldsymbol{x};\boldsymbol{y}) = u(\boldsymbol{x}) + \widehat{v}(\boldsymbol{x};\boldsymbol{y})$ is a smooth majorizing function of the function f in (4.20) at $\boldsymbol{y}$ that satisfies Assumption 4.1.*

Proof. As the other conditions in Assumption 4.1 can be easily verified for the function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$, in the following, we only demonstrate the subgradient consistency between f and $\widehat{f}$ at $\boldsymbol{y}$. By the definition in (2.25), the Clarke subdifferential of f at the point $\boldsymbol{x}$ in the direction $\boldsymbol{d}$ is given by

$$\begin{aligned} f^{\circ}_{\boldsymbol{d}}(\boldsymbol{x}) &= \limsup_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \left\{ \frac{u(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - u(\boldsymbol{x}')}{\varepsilon} + \frac{v(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - v(\boldsymbol{x}')}{\varepsilon} \right\} \\ &\leq \underbrace{\limsup_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \frac{u(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - u(\boldsymbol{x}')}{\varepsilon}}_{u^{\circ}_{\boldsymbol{d}}(\boldsymbol{x})} + \underbrace{\limsup_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \frac{v(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - v(\boldsymbol{x}')}{\varepsilon}}_{v^{\circ}_{\boldsymbol{d}}(\boldsymbol{x})}, \end{aligned} \tag{4.21}$$

where the last inequality, i.e., the sum rule (2.34) of the Clarke subdifferential, comes from the subadditivity of the limit superior and it holds with equality if one of the difference quotients converges [LLN22, Ex. 2.5.3 and 2.5.4]. The function u is continuously differentiable and, hence, strictly differentiable, which implies that the following limit exists

$$\lim_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \frac{u(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - u(\boldsymbol{x}')}{\varepsilon} = u'_{\boldsymbol{d}}(\boldsymbol{x}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n.$$

Consequently, we have

$$u^{\circ}_{\boldsymbol{d}} = \limsup_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \frac{u(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - u(\boldsymbol{x}')}{\varepsilon} = \lim_{\boldsymbol{x}'\to\boldsymbol{x},\,\varepsilon\downarrow 0} \frac{u(\boldsymbol{x}'+\varepsilon\boldsymbol{d}) - u(\boldsymbol{x}')}{\varepsilon} = u'_{\boldsymbol{d}}(\boldsymbol{x}),$$

and the equality holds in (4.21), i.e.,

$$f'^{\circ}_{\boldsymbol{d}}(\boldsymbol{x}) = u'_{\boldsymbol{d}}(\boldsymbol{x}) + v'^{\circ}_{\boldsymbol{d}}(\boldsymbol{x}) \quad \forall\, \boldsymbol{d} \in \mathbb{R}^n.$$

By the definition of the Clarke subdifferential in (2.26), this further implies that

$$\partial^C f(\boldsymbol{x}) = \partial^C u(\boldsymbol{x}) + \partial^C v(\boldsymbol{x}) = \{\nabla_{\boldsymbol{x}} u(\boldsymbol{x})\} + \partial^C v(\boldsymbol{x}),$$

where the last inequality comes from the strict differentiability of u. Since $\widehat{v}(\boldsymbol{x};\boldsymbol{y})$ preserves a subgradient of v at $\boldsymbol{y}$, i.e., $\nabla_{\boldsymbol{x}}\widehat{v}(\boldsymbol{y};\boldsymbol{y}) \in \partial^C v(\boldsymbol{y})$, we have

$$\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y}) = \nabla_{\boldsymbol{x}} u(\boldsymbol{y}) + \nabla_{\boldsymbol{x}}\widehat{v}(\boldsymbol{y};\boldsymbol{y}) \in \partial^C u(\boldsymbol{y}) + \partial^C v(\boldsymbol{y}) = \partial^C f(\boldsymbol{y}).$$

Thus, we can conclude that $\widehat{f}(\boldsymbol{x};\boldsymbol{y}) = u(\boldsymbol{x}) + \widehat{v}(\boldsymbol{x};\boldsymbol{y})$ is a smooth majorizing function of f at $\boldsymbol{y}$ satisfying Assumption 4.1. □

We show below some smooth majorizing functions for the nonsmooth function v derived based on different structures assumed on v, respectively.

1. **Linear Majorization with Subgradient:** As a generalization of the gradient, a subgradient defined in (2.13) of a convex function generates a supporting hyperplane that minorizes the function. Therefore, when the nonsmooth component v in (4.20) is concave, it possesses the following linear majorization at a point $\boldsymbol{y}$:

$$\widehat{v}(\boldsymbol{x};\boldsymbol{y}) = v(\boldsymbol{y}) + \boldsymbol{s}^{\mathsf{T}}(\boldsymbol{x}-\boldsymbol{y}) \geq v(\boldsymbol{x}) \tag{4.22}$$

with

$$\boldsymbol{s} \in -\partial(-v)(\boldsymbol{y}) = \partial^C v(\boldsymbol{y}), \tag{4.23}$$

where the last inequality comes from (2.33).

Recall that the sum rule (2.34) of the Clarke subdifferential holds with equality for concave functions as they are subdifferentially regular. Thus, if the nonsmooth function v can be written as a sum of multiple concave functions, i.e.,

$$v(\boldsymbol{x}) = \sum_{i=1}^{I} v_i(\boldsymbol{x}) \tag{4.24}$$

with each v_i being concave but not necessarily smooth, then a subgradient of v can be obtained easily from the subgradients of each v_i. Specifically, since $\partial^C v(\boldsymbol{x}) = \sum_{i=1}^{I} \partial^C v_i(\boldsymbol{x})$, we have

$$\boldsymbol{s}_i \in \partial^C v_i(\boldsymbol{x}) \quad \forall\, i = 1,\ldots,I \quad \Longrightarrow \quad \sum_{i=1}^{I} \boldsymbol{s}_i \in \partial^C v(\boldsymbol{x}).$$

In other words, each v_i can be linearized independently for the construction of the majorization in (4.22).

Furthermore, similar to the differentiable concave case discussed in Section 3.1, any smooth convex function that preserves the function value and a subgradient of v at $\boldsymbol{y}$ also majorizes the nonsmooth concave function v, which is preferred if a low-complexity approximate problem is demanded. Depending on the specific structure of v, it may be possible to construct even a smooth concave majorization, which is certainly tighter than the linearization with the same subgradient.

2. **Pointwise Minimum:** Suppose that the component v in (4.20) can be expressed as the pointwise minimum of a finite collection of smooth functions $\{v_i\}_{i=1}^{I}$, i.e.,

$$v(\boldsymbol{x}) = \min_{i=1,\dots,I} v_i(\boldsymbol{x}), \tag{4.25}$$

with $I < \infty$. Note that, the above structure alone does not ensure the smoothness of v. For any point $\boldsymbol{y}$, let $\mathcal{I}(\boldsymbol{y})$ denote the set of indices i for which $v_i(\boldsymbol{y}) = v(\boldsymbol{y})$, i.e., the indices at which the minimum defining v is attained. Based on this pointwise minimum structure, a natural smooth majorizer for v at $\boldsymbol{y}$ is

$$\widehat{v}(\boldsymbol{x};\boldsymbol{y}) = v_k(\boldsymbol{x}) \geq v(\boldsymbol{x}) \quad \text{with } k \in \mathcal{I}(\boldsymbol{y}). \tag{4.26}$$

In other words, a smooth majorizer is constructed by replacing v by one component v_k in (4.25) that achieves the pointwise minimum at the current point $\boldsymbol{y}$. The relation in (4.25) is equivalent to $-v(\boldsymbol{x}) = \max_{i=1,\dots,I} -v_i(\boldsymbol{x})$ and the following property holds for the Clarke subdifferential of a pointwise maximum [Cla90, Prop. 2.3.12]:

$$\partial^C(-v)(\boldsymbol{x}) = \operatorname{co}\left\{\partial^C(-v_i)(\boldsymbol{x}) \mid i \in \mathcal{I}(\boldsymbol{x})\right\}, \tag{4.27}$$

where $\operatorname{co}\{\cdot\}$ denotes the convex hull. Recall that $\partial^C(-v)(\boldsymbol{x}) = -\partial^C v(\boldsymbol{x})$ and we have $\partial^C v_i(\boldsymbol{x}) = \{\nabla_{\boldsymbol{x}} v_i(\boldsymbol{x})\}$ for $i = 1, \dots, I$ due to their smoothness. It follows that any majorizing function $\widehat{v}(\boldsymbol{x};\boldsymbol{y})$ constructed by (4.26) retains a subgradient of v at $\boldsymbol{y}$, i.e.,

$$\nabla_{\boldsymbol{x}} v_k(\boldsymbol{y}) \in \partial^C v(\boldsymbol{y}) \quad \forall\, k \in \mathcal{I}(\boldsymbol{y}). \tag{4.28}$$

The majorization in (4.26) can be readily extended to the functions that can be expressed as pointwise minima of an infinite collection of smooth functions, i.e., $I = \infty$. More generally, we assume that the nonsmooth function v can be written as the following pointwise minimum:

$$v(\boldsymbol{x}) = \min_{\boldsymbol{\theta} \in \Theta} v_{\boldsymbol{\theta}}(\boldsymbol{x}), \tag{4.29}$$

where $v_{\boldsymbol{\theta}}$ is a family of functions on $\mathbb{R}^n$ parameterized by $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^m$ and $v_{\boldsymbol{\theta}}$ is smooth in $\boldsymbol{x}$ for any given $\boldsymbol{\theta} \in \Theta$. Define the set $\Theta^\star(\boldsymbol{y}) = \left\{ \operatorname*{argmin}_{\boldsymbol{\theta}\in\Theta} v_{\boldsymbol{\theta}}(\boldsymbol{y}) \right\}$. Similar to (4.26), v possess the following smooth majorizer at $\boldsymbol{y}$:

$$\widehat{v}(\boldsymbol{x};\boldsymbol{y}) = v_{\boldsymbol{\theta}^\star}(\boldsymbol{x}) \geq v(\boldsymbol{x}) \quad \text{with } \boldsymbol{\theta}^\star \in \Theta^\star(\boldsymbol{y}). \tag{4.30}$$

Some properties similar to (4.27) can be derived under some mild assumptions on the functions $v_{\boldsymbol{\theta}}$ and the parameter space Θ (see [Cla90, Sec. 2.8]). Thus, following the same line of analysis as in the previous finite case, we can justify the subgradient consistency between v and the above majorizing function, which is, therefore, omitted.

In the following, we demonstrate a simple application where the preceding smoothing majorization techniques can be applied. The phase retrieval problem, which will be studied deeply in Chapter 5, aims at recovering unknown signal $\boldsymbol{x} \in \mathbb{C}^N$ from phaseless, possibly noise-corrupted, measurements of linear mixtures $y_i \approx |\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x}|, i = 1, \ldots, M$. Let us consider the oversampled case, i.e., $M \geq N$. Define $\boldsymbol{y} = [y_1, \ldots, y_M]^{\mathsf{T}} \in \mathbb{R}_+^M$ and $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_M]^{\mathsf{H}} \in \mathbb{C}^{M\times N}$. The recovery problem can be formulated as the following nonlinear least-squares (LS) problem:

$$\min_{\boldsymbol{x}\in\mathbb{C}^n} \quad \underbrace{\frac{1}{2}\|\boldsymbol{y} - |\boldsymbol{A}\boldsymbol{x}|\|_2^2}_{f(\boldsymbol{x})}, \tag{4.31}$$

where the operation $|\cdot|$ is applied elementwise. The objective function f can be expanded as

$$f(\boldsymbol{x}) = \underbrace{\frac{1}{2}\|\boldsymbol{y}\|_2^2 + \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x}\|_2^2}_{u(\boldsymbol{x})} \underbrace{-\boldsymbol{y}^{\mathsf{T}}|\boldsymbol{A}\boldsymbol{x}|}_{v(\boldsymbol{x})}, \tag{4.32}$$

where u is smooth and v is almost smooth everywhere except for the points $\boldsymbol{x}$ where $\boldsymbol{A}\boldsymbol{x}$ contains zero entries. For majorizing v, we note that

$$|x| = |x \cdot \mathrm{e}^{\mathrm{j}\theta}| \geq \Re(x \cdot \mathrm{e}^{\mathrm{j}\theta}) \quad \text{for any } x \in \mathbb{C} \text{ and } \theta \in [0, 2\pi), \tag{4.33}$$

and that equality holds for $\theta = -\arg(x)$. Thus, due to the nonnegativity of $\boldsymbol{y}$, v is majorized at $\boldsymbol{x}^{(t)}$ by the following smooth function

$$\widehat{v}(\boldsymbol{x};\boldsymbol{x}^{(t)}) = -\boldsymbol{y}^{\mathsf{T}}\Re\left(\boldsymbol{A}\boldsymbol{x} \odot \mathrm{e}^{-\mathrm{j}\arg\left(\boldsymbol{A}\boldsymbol{x}^{(t)}\right)}\right) = -\Re\left(\left(\boldsymbol{y}^{(t)}\right)^{\mathsf{H}}\boldsymbol{A}\boldsymbol{x}\right), \tag{4.34}$$

with

$$\boldsymbol{y}^{(t)} = \boldsymbol{y} \odot \mathrm{e}^{\mathrm{j}\arg\left(\boldsymbol{A}\boldsymbol{x}^{(t)}\right)},$$

where the operations $\mathrm{e}^{(\cdot)}$ and $\arg(\cdot)$ are applied elementwise, and one may set $\arg\left(\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x}\right)$ to be an arbitrary value in $[0, 2\pi)$ if $\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x} = 0$. It follows that f is majorized at $\boldsymbol{x}^{(t)}$ by a linear LS objective

$$\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) = u(\boldsymbol{x}) + \widehat{v}(\boldsymbol{x}; \boldsymbol{x}^{(t)}) = \frac{1}{2}\left\|\boldsymbol{y}^{(t)} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2. \tag{4.35}$$

The above majorizing function has a minimizer

$$\boldsymbol{x}^{(t+1)} = \left(\boldsymbol{A}^{\mathsf{H}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathsf{H}}\boldsymbol{y}^{(t)},$$

which corresponds to the well-known Gerchberg-Saxton algorithm [GS72].

The majorization in (4.34) can in fact be regarded as an instance of both the majorization techniques in (4.22) and (4.30). On the one hand, the cross term v in (4.32) can be written as

$$v(\boldsymbol{x}) = -\boldsymbol{y}^{\mathsf{T}}|\boldsymbol{A}\boldsymbol{x}| = \sum_{i=1}^{M} \underbrace{-y_i|\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x}|}_{v_i(\boldsymbol{x})}, \tag{4.36}$$

where each v_i is concave and is smooth everywhere else except for the points $\boldsymbol{x}$ with $\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x} = 0$. According to the construction (2.27) of the Clarke subdifferential, the gradient or subdifferential of v_i is given by

$$\begin{cases} \nabla_{\boldsymbol{x}} v_i(\boldsymbol{x}) = -y_i \mathrm{e}^{\mathsf{j}\arg\left(\boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x}\right)}\boldsymbol{a}_i & \text{if } \boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x} \neq 0, \\ \partial^C v_i(\boldsymbol{x}) = \operatorname{co}\left\{y_i \mathrm{e}^{\mathsf{j}\theta}\boldsymbol{a}_i \mid \theta \in [0, 2\pi)\right\} & \text{if } \boldsymbol{a}_i^{\mathsf{H}}\boldsymbol{x} = 0. \end{cases} \tag{4.37}$$

Thus, the majorization in (4.34) is a linearization of the concave nonsmooth function v with a subgradient, which corresponds to the majorization technique in (4.22).

On the other hand, the nonsmooth function v in (4.32) can be viewed as the following pointwise minimum:

$$v(\boldsymbol{x}) = \min_{\boldsymbol{\theta}\in[0,2\pi)^M} \underbrace{-\Re\left(\left(\boldsymbol{y} \odot \mathrm{e}^{\mathsf{j}\boldsymbol{\theta}}\right)^{\mathsf{H}}\boldsymbol{A}\boldsymbol{x}\right)}_{v_{\boldsymbol{\theta}}(\boldsymbol{x})}, \tag{4.38}$$

and the minimum solution at $\boldsymbol{x} = \boldsymbol{x}^{(t)}$ is $\boldsymbol{\theta}^\star = \arg\left(\boldsymbol{A}\boldsymbol{x}^{(t)}\right)$. Therefore, the majorization in (4.34) is also an instance of the majorization technique in (4.30) for pointwise minima.

4.1.3 Nonsmooth Regularization

In this subsection, we extend the proposed MM framework with smoothing majorization to address the following composite optimization problem:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \quad f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{4.39}$$

Algorithm 2: The MM Algorithm with Smoothing Majorization for Solving Problem (4.39)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Construct a smooth majorizer $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$ that satisfies Assumption 4.1 ;
  Let $\boldsymbol{x}^{(t+1)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + g(\boldsymbol{x})$;
  $t \leftarrow t + 1$;
until *converged* **and** *($\boldsymbol{x}^{(t)} \notin \operatorname{int}(\mathcal{X})$* **or** *$\nabla_{\boldsymbol{x}} f\left(\boldsymbol{x}^{(t)}\right)$ exists)*;
return $\boldsymbol{x}^{(t)}$

where the nondifferentiable nonconvex function f satisfies Assumption 4.2 and g is a convex but not necessarily differentiable (possibly separable) regularization term such as $\|\cdot\|_1$. Due to the convexity of g, there may exist interior points in $\mathcal{X}$ where the overall objective function $f + g$ is locally convex, hence, regular, and nondifferentiable. That is, the overall objective function $f + g$ is not guaranteed to satisfy Assumption 4.2. Therefore, instead of constructing a smooth majorizing function for the overall objective function $f + g$, we only replace f with a smooth majorizer and leave g unaltered. Specifically, the approximation of the problem (4.39) at the tth iteration is

$$\boldsymbol{x}^{(t+1)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \quad \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + g(\boldsymbol{x}), \tag{4.40}$$

where $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ is a smooth majorizer of f at the current iterate $\boldsymbol{x}^{(t)}$ that satisfies Assumption 4.1. With this modification, the proposed MM algorithm with smoothing majorization for the composite problem (4.39) is then outlined in Algorithm 2.

Next, to analyze the convergence behavior of the modified MM algorithm, we employ the same reformulation technique as in (3.22) to equivalently reformulate the problem (4.39) as an instance of problem (1.1) as follows

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^n,\, y\in\mathbb{R}} \quad & f(\boldsymbol{x}) + y \\ \text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y, \end{aligned} \tag{4.41}$$

with the help of an auxiliary variable y. It is easy to verify that the overall objective function $f(\boldsymbol{x}) + y$ satisfies Assumption 4.2 if f satisfies Assumption 4.2, and that the feasible set

$$\mathcal{C} = \{(\boldsymbol{x}, y) \in \mathbb{R}^{n+1} \mid \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y\} \tag{4.42}$$

is closed and convex. Similarly, the approximate problem (4.40) can be rewritten as

$$\begin{aligned} \left(\boldsymbol{x}^{(t+1)}, y^{(t+1)}\right) = \underset{\boldsymbol{x}\in\mathbb{R}^n,\, y\in\mathbb{R}}{\operatorname{argmin}} \quad & \widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + y \\ \text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \le y, \end{aligned} \tag{4.43}$$

and the solution of the auxiliary variable is $y^{(t+1)} = g\left(\boldsymbol{x}^{(t+1)}\right)$. As can be easily verified, the overall approximate function $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)+y$ satisfies Assumption 4.1 when $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ is a smooth majorizing function of f satisfying Assumption 4.1. In particular, the subgradient consistency of $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + y$ at $\left(\boldsymbol{x}^{(t)}, y^{(t)}\right)$ can be illustrated as follows

$$\nabla_{(\boldsymbol{x},y)} \left(\widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) + y^{(t)}\right) = \left(\nabla_{\boldsymbol{x}} \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right), 1\right) \in \partial^C_{(\boldsymbol{x},y)} \left(f\left(\boldsymbol{x}^{(t)}\right) + y^{(t)}\right) = \partial^C f\left(\boldsymbol{x}^{(t)}\right) \times \{1\}, \quad (4.44)$$

when $\nabla_{\boldsymbol{x}} \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) \in \partial^C f\left(\boldsymbol{x}^{(t)}\right)$. Consequently, Algorithm 2 can be viewed as applying Algorithm 1 on the reformulated problem (4.41), whose convergence is then justified by Theorem 4.1. That is, every limit point $(\boldsymbol{x}^\star, y^\star)$, where $y^\star = g(\boldsymbol{x}^\star)$, of the solution sequence generated by Algorithm 2 is a C-stationary point of the reformulated problem (4.41) in the sense that

$$\mathbf{0} \in \partial^C_{(\boldsymbol{x},y)} \left(f(\boldsymbol{x}^\star) + y^\star\right) + \mathcal{N}_{\mathcal{C}}(\boldsymbol{x}^\star, y^\star). \quad (4.45)$$

The set $\mathcal{C}$ in (4.42) corresponds to the epigraph of the function $g + \mathbb{I}_{\mathcal{X}}$, whose normal cone is given by [Cla90, Corollary of Thm. 2.4.9]

$$\mathcal{N}_{\mathcal{C}}(\boldsymbol{x}^\star, y^\star) = \partial^C \left(g + \mathbb{I}_{\mathcal{X}}\right)(\boldsymbol{x}^\star) \times \{-1\}.$$

Recall that the sum rule (2.34) of the Clarke subdifferential holds with equality for convex functions as they are subdifferentially regular. It follows from the convexity of the function g and the set $\mathcal{X}$ that

$$\mathcal{N}_{\mathcal{C}}(\boldsymbol{x}^\star, y^\star) = \left(\partial^C g\left(\boldsymbol{x}^\star\right) + \partial \mathbb{I}_{\mathcal{X}}\left(\boldsymbol{x}^\star\right)\right) \times \{-1\} = \left(\partial^C g\left(\boldsymbol{x}^\star\right) + \mathcal{N}_{\mathcal{X}}\left(\boldsymbol{x}^\star\right)\right) \times \{-1\}. \quad (4.46)$$

Combining (4.44) and (4.46), we can equivalently express the C-stationarity condition (4.45) of the reformulated problem (4.41) as

$$\mathbf{0} \in \partial^C f\left(\boldsymbol{x}^\star\right) + \partial^C g\left(\boldsymbol{x}^\star\right) + \mathcal{N}_{\mathcal{X}}\left(\boldsymbol{x}^\star\right). \quad (4.47)$$

In the general case, due to the weak sum rule (2.34) of the Clarke subdifferential, the limit point $\boldsymbol{x}^\star$ that satisfies (4.47) as a C-stationary point of the reformulated problem (4.41) may not be a C-stationary point of the original problem (4.39), which needs to satisfy

$$\mathbf{0} \in \partial^C \left(f + g\right)\left(\boldsymbol{x}^\star\right) + \mathcal{N}_{\mathcal{X}}\left(\boldsymbol{x}^\star\right). \quad (4.48)$$

Furthermore, Proposition 4.3 can also be applied to the reformulated problem (4.39) to identify a subset of stationary points that are not local minima. Similar to Algorithm 1, an additional termination condition based on Proposition 4.3 is then incorporated in Algorithm 2 to exclude those stationary points.

4.2 Smoothing Successive Convex Approximation

In the previous section, to reduce the computational complexity of the approximate problem, we employ a relaxed subgradient consistency condition so that a smooth majorizer can be constructed for nonsmooth problems. However, in some scenarios, it is still difficult to minimize the smooth majorizing function exactly, e.g., due to its nonconvexity. On the other hand, as mentioned in Remark 3.2, the monotonic descent property of the objective function value, which is the key to convergence, depends only on decreasing the majorizing function, not minimizing it. The smoothness of the majorizing function allows us to employ the idea of SCA in Section 3.2 to obtain an approximate minimizer of the majorizing function efficiently. Therefore, in this section, to further reduce the computational complexity, we develop an inexact MM framework, termed the *smoothing SCA* framework, by combining the smoothing majorization technique and the idea of convex approximation.

Specifically, at the tth iteration, we further construct a convex approximation $\widetilde{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ of the smooth majorizing function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ at the current iterate $\boldsymbol{x}^{(t)}$ that satisfies the following assumptions.

Assumption 4.3 (Convex approximation)**.** *Let the approximate function $\widetilde{f}(\cdot;\cdot)$ satisfy the following:*

1) *$\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ are continuously differentiable in $\boldsymbol{x}\in\mathcal{X}$ for any given $\boldsymbol{y}\in\mathcal{X}$ and continuous in $\boldsymbol{y}\in\mathcal{X}$ for any given $\boldsymbol{x}\in\mathcal{X}$;*
2) *Gradient consistency: $\nabla_{\boldsymbol{x}}\widetilde{f}(\boldsymbol{y};\boldsymbol{y})=\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{y};\boldsymbol{y})$ for all $\boldsymbol{y}\in\mathcal{X}$;*
3) *$\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ is pseudoconvex in $\boldsymbol{x}\in\mathcal{X}$ for any given $\boldsymbol{y}\in\mathcal{X}$.*

Then the following approximate problem is solved

$$\widetilde{\boldsymbol{x}}^{(t)}\in\operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}}\quad\widetilde{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right). \tag{4.49}$$

The solution $\widetilde{\boldsymbol{x}}^{(t)}$ may reduce neither the majorizing function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ nor the original function f since $\widetilde{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ is not necessarily a global upper bound of $\widehat{f}$ or f. However, as discussed for the SCA framework, Assumption 4.3 ensures that the difference $\widetilde{\boldsymbol{x}}^{(t)}-\boldsymbol{x}^{(t)}$ indicates a descent direction of the majorizing function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$. Hence, the variable $\boldsymbol{x}$ can be updated along the descent direction as

$$\boldsymbol{x}^{(t+1)}=\boldsymbol{x}^{(t)}+\gamma^{(t)}\left(\widetilde{\boldsymbol{x}}^{(t)}-\boldsymbol{x}^{(t)}\right) \tag{4.50}$$

Algorithm 3: The SCA Algorithm Extended by Smoothing Majorization for Solving Problem (1.1)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Construct a smooth majorizer $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$;
  Construct a pseudoconvex approximation $\widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ at $\boldsymbol{x}^{(t)}$;
  Let $\widetilde{\boldsymbol{x}}^{(t)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$;
  Compute the step size $\gamma^{(t)}$ by the exact line search (4.51) or the successive line search (4.52) ;
  Let $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)}\left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right)$;
  $t \leftarrow t + 1$;
until *converged* **and** *($\boldsymbol{x}^{(t)} \notin \operatorname{int}(\mathcal{X})$* **or** *$\nabla_{\boldsymbol{x}} f\left(\boldsymbol{x}^{(t)}\right)$ exists)*;
return $\boldsymbol{x}^{(t)}$

with a suitable step size $\gamma^{(t)} \in (0, 1]$ since the feasible set $\mathcal{X}$ is convex.

In this thesis, we focus on the line search approach introduced in Section 3.2 for choosing a step size $\gamma^{(t)}$. Specifically, we perform a line search method on the majorizing function $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$, due to its smoothness, to find a step size $\gamma^{(t)}$ that provides a sufficient decrease of the majorizing function $\widehat{f}$ as well as the original function f. The exact line search in (3.17) and the successive line search with Armijo rule in (3.19) are customized for the majorizing function $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ as follows. In the exact line search, the step size $\gamma^{(t)}$ is chosen to minimize the majorizing function $\widehat{f}$ along the descent direction, i.e.,

$$\gamma^{(t)} \in \underset{\gamma\in[0,1]}{\operatorname{argmin}} \quad \widehat{f}\left(\boldsymbol{x}^{(t)} + \gamma(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}); \boldsymbol{x}^{(t)}\right). \tag{4.51}$$

On the other hand, the Armijo rule can be used when the optimization problem (4.51) can not be solved efficiently. In the Armijo rule, the step size is successively decreased at a geometric rate until a condition on the sufficient decrease of $\widehat{f}$ is fulfilled, which is expressed as the following optimization problem:

$$\begin{aligned} \gamma^{(t)} = \underset{\gamma}{\operatorname{argmax}} \quad & \gamma \\ \text{s.t.} \quad & \gamma \in \{\beta^k \mid k \in \mathbb{N}\}, \\ & \widehat{f}\left(\boldsymbol{x}^{(t)} + \gamma(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}); \boldsymbol{x}^{(t)}\right) - \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) \leq \gamma\sigma\delta \end{aligned} \tag{4.52}$$

with the directional derivative $\delta = \left(\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}}\left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right)$ and $0 < \beta, \sigma < 1$.

The proposed smoothing SCA framework for solving the problem (1.1) is then outlined in Algorithm 3, which can be viewed as an inexact MM. In summary, compared to the exact MM framework with smoothing majorization discussed in Section 4.1, in the

framework introduced in this section, the construction of the approximate problem is divided into two steps, namely, the smoothing majorization and the convex approximation, so that the two desiderata of the approximate function, i.e., the tightness to the original function and the low complexity of minimizing the approximate function, can be treated separately. Specifically, the smoothing majorization step aims to design a smooth majorizer that is as tight to the original function as possible, regardless of the complexity. For instance, one may choose a tighter concave majorizer for a concave function, rather than its tightest convex majorizer, i.e., a supporting hyperplane, as presented in (4.22). Then, in the convex approximation step, we further develop a convex approximate function that can be easily minimized, for the smooth majorizer. In particular, the separable approximation techniques in Section 3.2.2 can be employed to exploit potential separable structures of the constraints, so as to develop an approximate problem that can be decomposed and solved in parallel. Moreover, as demonstrated in the quadratic approximation technique in Section 3.2.2, without the global upper bound constraint, it provides the flexibility in designing an approximate function that preserves more local behavior, e.g., second-order partial derivatives, of the majorizing function and, hence, also the original function.

4.2.1 Convergence Analysis

Next, we study the convergence behavior of the proposed smoothing SCA algorithm. To this end, we first present the following property. The solution set of the approximate problem (4.49), which is a set-valued map of the point $\boldsymbol{x}^{(t)}$, is denoted by

$$\widetilde{M}\left(\boldsymbol{x}^{(t)}\right) = \left\{ \operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}} \widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) \right\}. \tag{4.53}$$

Proposition 4.5 (Stationary point and descent direction)**.** *Suppose that Assumptions 4.1, 4.2, and 4.3 are satisfied. Then the following two facts hold.*

1) *Every fixed point $\boldsymbol{y}$ of the smoothing SCA algorithm, i.e., $\boldsymbol{y} \in \widetilde{M}(\boldsymbol{y})$, is a C-stationary point of the problem* (1.1)*;*

2) *If $\boldsymbol{y}$ is not a fixed point of the smoothing SCA algorithm, then the vector $\widetilde{\boldsymbol{y}} - \boldsymbol{y}$ for any $\widetilde{\boldsymbol{y}} \in \widetilde{M}(\boldsymbol{y})$ is a descent direction of $f(\boldsymbol{x})$ in the problem* (1.1)*, i.e.,*

$$f'_{\widetilde{\boldsymbol{y}}-\boldsymbol{y}}(\boldsymbol{y}) < 0 \quad \forall\, \widetilde{\boldsymbol{y}} \in \widetilde{M}(\boldsymbol{y}). \tag{4.54}$$

Proof. 1) Suppose that $\boldsymbol{y}$ is a fixed point of the smoothing SCA algorithm, i.e., $\boldsymbol{y} \in \widetilde{M}(\boldsymbol{y})$. Then, by the definition in (4.53), $\boldsymbol{y}$ is a global optimal solution of the

approximate problem (4)) and, hence, satisfies the C-stationarity condition in Definition 2.18, i.e.,

$$\mathbf{0} \in \partial^C \widetilde{f}(\boldsymbol{y};\boldsymbol{y}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{y}). \tag{4.55}$$

Due to the smoothness of both the majorization $\widehat{f}$ and the approximation $\widetilde{f}$, and the subgradient consistency in Assumptions4.1 and 4.3, we have

$$\partial^C \widetilde{f}(\boldsymbol{y};\boldsymbol{y}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{y}) = \partial^C \widehat{f}(\boldsymbol{y};\boldsymbol{y}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{y}) \subseteq \partial^C f(\boldsymbol{y}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{y}) \tag{4.56}$$

with $\partial^C \widetilde{f}(\boldsymbol{y};\boldsymbol{y}) = \left\{\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\right\}$ and $\partial^C \widehat{f}(\boldsymbol{y};\boldsymbol{y}) = \left\{\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{y};\boldsymbol{y})\right\}$ for any $\boldsymbol{y} \in \mathcal{X}$. Combining (4.55) and (4.56), we obtain

$$\mathbf{0} \in \partial^C f(\boldsymbol{y}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{y}), \tag{4.57}$$

which implies that the fixed point $\boldsymbol{y}$ of the smoothing SCA algorithm is a C-stationary point of the original problem (1.1).

2) Consider a point $\boldsymbol{y}$ that is not a fixed point of the smoothing SCA algorithm, i.e., $\boldsymbol{y} \notin \widetilde{M}(\boldsymbol{y})$. Define a point $\widetilde{\boldsymbol{y}} \in \widetilde{M}(\boldsymbol{y})$, which, by the definition in (4.53), is a global minimizer of the approximate function $\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ on $\mathcal{X}$. Since $\boldsymbol{y} \notin \widetilde{M}(\boldsymbol{y})$, we have

$$\widetilde{f}(\widetilde{\boldsymbol{y}};\boldsymbol{y}) < \widetilde{f}(\boldsymbol{y};\boldsymbol{y}). \tag{4.58}$$

Then the pseudoconvexity of $\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ in $\boldsymbol{x} \in \mathcal{X}$ implies that the vector $\widetilde{\boldsymbol{y}} - \boldsymbol{y} \neq \mathbf{0}$ indicates a descent direction of $\widetilde{f}$ in $\boldsymbol{x}$, i.e.,

$$\widetilde{f}'_{\widetilde{\boldsymbol{y}}-\boldsymbol{y}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} \leq 0. \tag{4.59}$$

Combining this with the subgradient consistency in Assumptions 4.1 and 4.3 and Proposition 4.1, we obtain

$$f'_{\widetilde{\boldsymbol{y}}-\boldsymbol{y}}(\boldsymbol{y}) \leq \widehat{f}'_{\widetilde{\boldsymbol{y}}-\boldsymbol{y}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} = \widetilde{f}'_{\widetilde{\boldsymbol{y}}-\boldsymbol{y}}(\boldsymbol{x};\boldsymbol{y})\big|_{\boldsymbol{x}=\boldsymbol{y}} < 0. \tag{4.60}$$

□

Proposition 4.5 implies that the smoothing SCA algorithm can successively decrease the objective function value by line search until a fixed point is achieved, which is ensured to be a C-stationary point of the original problem (1.1).

Similar to the SCA framework introduced in Section 3.2, the following mild assumptions are also required to establish the convergence of the smoothing SCA algorithm.

Assumption 4.4. *We make the following assumptions:*

1) The map $\widetilde{M}\left(\boldsymbol{x}^{(t)}\right)$ is nonempty for $t \in \mathbb{N}$;

2) Given any convergent subsequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathcal{T}}$ with $\mathcal{T} \subseteq \mathbb{N}$, the sequence $\left(\widetilde{\boldsymbol{x}}^{(t)}\right)_{t\in\mathcal{T}}$ is bounded.

Then the convergence of the smoothing SCA algorithm is stated in the following theorem.

Theorem 4.2 (Convergence of the smoothing SCA algorithm)**.** *Suppose that Assumptions 4.1, 4.2, 4.3, and 4.4 are satisfied and that the line search approach is employed. Then every limit point of the solution sequence generated by Algorithm 3 is a C-stationary point of the problem* (1.1)*.*

Proof. If a fixed point of the smoothing SCA algorithm is achieved in a finite number of iterations, then the fact 1) in Proposition 4.5 implies that the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_t$ has converged to a C-stationary point of the original problem (1.1). Otherwise, by following the same procedures as in [YP17], we show that the limit of any convergent subsequence of the iterates $\boldsymbol{x}^{(t)}$ generated by the smoothing SCA algorithm is a C-stationary point of the original problem based on the fact 2) in Proposition 4.5.

The fact 2) in Proposition 4.5 implies that the sequence of objective function values $\left(f(\boldsymbol{x}^{(t)})\right)_{t\in\mathbb{N}}$ is monotonically nonincreasing. Assumption 1.1 ensures that the function f is bounded below on $\mathcal{X}$. Then, by the monotone convergence theorem [Ber16, Prop. A.3], the sequence $\left(f(\boldsymbol{x}^{(t)})\right)_{t\in\mathbb{N}}$ converges to a local minimum of f in $\mathcal{X}$. Thus, for any two convergent subsequences $(\boldsymbol{x}^{(t)})_{t\in\mathcal{T}_1}$ and $(\boldsymbol{x}^{(t)})_{t\in\mathcal{T}_2}$ with $\mathcal{T}_1, \mathcal{T}_2 \subseteq \mathbb{N}$, it holds that

$$\lim_{t\to\infty} f(\boldsymbol{x}^{(t)}) = \lim_{t\in\mathcal{T}_1, t\to\infty} f(\boldsymbol{x}^{(t)}) = \lim_{t\in\mathcal{T}_2, t\to\infty} f(\boldsymbol{x}^{(t)}). \tag{4.61}$$

Since $f(\boldsymbol{x})$ is a continuous function, it follows that

$$f\left(\lim_{t\in\mathcal{T}_1,\, t\to\infty} \boldsymbol{x}^{(t)}\right) = f\left(\lim_{t\in\mathcal{T}_2,\, t\to\infty} \boldsymbol{x}^{(t)}\right). \tag{4.62}$$

Now consider a convergent subsequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathcal{T}\subseteq\mathbb{N}}$ with limit point $\boldsymbol{z} \in \mathcal{X}$, i.e., $\lim_{t\in\mathcal{T}, t\to\infty} \boldsymbol{x}^{(t)} = \boldsymbol{z}$. Under the assumptions that $\widetilde{f}(\boldsymbol{x}; \boldsymbol{y})$ is continuous in both $\boldsymbol{x}$ and $\boldsymbol{y}$, and that $\left(\widetilde{\boldsymbol{x}}^{(t)}\right)_{t\in\mathcal{T}}$ is bounded, it follows from the maximum theorem [Ber97, Sec. VI.3] that there exists a convergent subsequence $\left(\widetilde{\boldsymbol{x}}^{(t)}\right)_{t\in\mathcal{T}_s\subseteq\mathcal{T}}$ with limit point $\widetilde{\boldsymbol{z}} \in \widetilde{M}(\boldsymbol{z})$. Moreover, since both $\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ and $\nabla_{\boldsymbol{x}}\widehat{f}(\boldsymbol{x}; \boldsymbol{y})$ are continuous in $\boldsymbol{x} \in \mathcal{X}$ for any given $\boldsymbol{y} \in \mathcal{X}$, applying the maximum theorem on the line search problem implies that there exists a subsequence $\left(\boldsymbol{x}^{(t+1)}\right)_{t\in\mathcal{T}_{s'}\subseteq\mathcal{T}_s}$ that converges to $\boldsymbol{z}'$ defined as $\boldsymbol{z}' = \boldsymbol{z} + \gamma(\widetilde{\boldsymbol{z}} - \boldsymbol{z})$,

where γ is the step size obtained by either the exact or successive line search on the majorizing function $\widehat{f}(\boldsymbol{s};\boldsymbol{z})$ at $\boldsymbol{z}$ in the direction $\widetilde{\boldsymbol{z}}-\boldsymbol{z}$. If $\boldsymbol{z}$ is not a C-stationary point of f, which, by (4.56), is neither a stationary point of the approximate function $\widetilde{f}(\boldsymbol{x};\boldsymbol{z})$, i.e., $\boldsymbol{z}\notin\widetilde{M}(\boldsymbol{z})$, then the fact 2) in Proposition 4.5 implies that $f(\boldsymbol{z}')<f(\boldsymbol{z})$, which contradicts (4.62). Therefore, any limit point of the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$ generated by the smoothing SCA algorithm is a C-stationary point of the original problem (1.1). □

As justified by (4.56), the point $\boldsymbol{y}$ is a stationary point of the smooth majorizing function $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ if and only if it is a stationary point of the convex approximation $\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ because of the gradient consistency between $\widehat{f}(\boldsymbol{x};\boldsymbol{y})$ and $\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ at the point $\boldsymbol{y}$ where the approximations are performed. Therefore, Proposition 4.3 can be readily extended to the convex approximation $\widetilde{f}$. That is, if a point $\boldsymbol{y}\in\operatorname{int}(\mathcal{X})$ is a stationary point of the convex approximation $\widetilde{f}(\boldsymbol{x};\boldsymbol{y})$ and the original function f is nondifferentiable at $\boldsymbol{y}$, then $\boldsymbol{y}$ is a C-stationary point of f, but not a local minimum point, since a descent direction of f exists at $\boldsymbol{y}$. This property leads to an additional termination condition that is included in the smoothing SCA Algorithm 3. When a fixed point $\boldsymbol{z}$ is achieved on the interior of the feasible set $\operatorname{int}(\mathcal{X})$ and f is nondifferentiable at $\boldsymbol{z}$, then, instead of terminating, the algorithm proceeds by constructing another majorizing function with a nonzero subgradient in $\partial^C f(\boldsymbol{z})$.

4.2.2 Nonsmooth Regularization

Similar to the proposed MM algorithm with smoothing majorization, the smoothing SCA Algorithm 3 can be easily extended to address the problem (4.39), where the objective function contains an additional convex nonsmooth regularization term and, hence, may not be majorized by a smooth function.

At the tth iteration, the function f is majorized at the iterate $\boldsymbol{x}^{(t)}$ by a smooth function $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$ and then approximated by a pseudoconvex function $\widetilde{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)$, where both functions satisfy Assumptions 4.1 and 4.3, respectively, whereas the regularization g remains unchanged. This leads to the following approximate problem:

$$\widetilde{\boldsymbol{x}}^{(t)}\in\operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}}\quad\widetilde{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)+g(\boldsymbol{x}). \tag{4.63}$$

The difference $\widetilde{\boldsymbol{x}}^{(t)}-\boldsymbol{x}^{(t)}$ provides a descent direction of the majorization $\widehat{f}\left(\boldsymbol{x};\boldsymbol{x}^{(t)}\right)+g(\boldsymbol{x})$, as well as the original objective function $f+g$. Then a decrease of the original

objective function $f + g$ in (4.39) is achieved by updating the optimization variable along the descent direction

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)} \left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)} \right) \tag{4.64}$$

with a step size $\gamma^{(t)} \in (0, 1]$ obtained by performing line search on the majorizing function. In particular, to further reduce the computational cost of line search, we employ the modified exact and successive line search methods introduced in Section 3.2, where the convex regularization g restricted to the descent direction, i.e., $g\left(\boldsymbol{x}^{(t)} + \gamma \left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)} \right)\right)$, is replaced by its linear majorization at $\boldsymbol{x}^{(t)}$: $g\left(\boldsymbol{x}^{(t)}\right) + \gamma \left(g\left(\widetilde{\boldsymbol{x}}^{(t)} - g\left(\boldsymbol{x}^{(t)}\right)\right)\right)$. The exact line search in (3.27) customized for the majorizing function is expressed as the following optimization problem:

$$\gamma^{(t)} \in \underset{\gamma \in [0,1]}{\operatorname{argmin}} \quad \widehat{f}\left(\boldsymbol{x}^{(t)} + \gamma(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}); \boldsymbol{x}^{(t)}\right) + \gamma \left(g\left(\widetilde{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)\right). \tag{4.65}$$

On the other hand, in the Armijo rule, the step size is successively decreased at a geometric rate until the required sufficient decrease of function value is achieved, which is expressed as the following optimization problem:

$$\begin{aligned} \gamma^{(t)} = \underset{\gamma}{\operatorname{argmax}} \quad & \gamma \\ \text{s.t.} \quad & \gamma \in \{\beta^k \mid k \in \mathbb{N}\} \text{ and sufficient decrease constraint.} \end{aligned} \tag{4.66}$$

Either of the two sufficient decrease constraints given in (3.31) and (3.32), respectively, can be used, which are given as follows for the majorizing function $\widehat{f}(\boldsymbol{x}; \boldsymbol{x}^{(t)}) + g(\boldsymbol{x})$:

$$\widehat{f}\left(\boldsymbol{x}^{(t)} + \gamma(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}); \boldsymbol{x}^{(t)}\right) - \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) + \gamma \left(g\left(\widetilde{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)\right) \leq \gamma\sigma\delta \tag{4.67}$$

and

$$\begin{aligned} \widehat{f}\left(\boldsymbol{x}^{(t)} + \gamma(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}); \boldsymbol{x}^{(t)}\right) - \widehat{f}\left(\boldsymbol{x}^{(t)}; \boldsymbol{x}^{(t)}\right) & \\ + g\left(\boldsymbol{x}^{(t)} + \gamma \left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right)\right) - g\left(\boldsymbol{x}^{(t)}\right) \leq \gamma\sigma\delta & \end{aligned} \tag{4.68}$$

with $\delta = \left(\nabla_{\boldsymbol{x}} \widehat{f}(\boldsymbol{x}^{(t)})\right)^{\mathsf{T}} \left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right) + g\left(\widetilde{\boldsymbol{x}}^{(t)}\right) - g\left(\boldsymbol{x}^{(t)}\right)$ and $0 < \beta, \sigma < 1$. With the above modifications, the proposed smoothing SCA algorithm for the regularized problem (4.39) is then outlined in Algorithm 4.

Similar to the proposed exact MM algorithm with smoothing majorization, the convergence of the proposed smoothing SCA Algorithm 4 for the regularized problem (4.39) can be established based on the equivalent reformulation in (4.41) with an auxiliary variable. Moreover, by the same reformulation technique, the approximate problem (4.63) is rewritten as

$$\begin{aligned} \left(\widetilde{\boldsymbol{x}}^{(t)}, \tilde{y}^{(t)}\right) = \underset{\boldsymbol{x} \in \mathbb{R}^n,\, y \in \mathbb{R}}{\operatorname{argmin}} \quad & \widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + y \\ \text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X},\, g(\boldsymbol{x}) \leq y, \end{aligned} \tag{4.69}$$

Algorithm 4: The SCA Algorithm Extended by Smoothing Majorization for Solving Problem (4.39)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Construct a smooth majorizer $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$;
  Construct a pseudoconvex approximation $\widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ of $\widehat{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right)$ at $\boldsymbol{x}^{(t)}$;
  Let $\widetilde{\boldsymbol{x}}^{(t)} \in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} \widetilde{f}\left(\boldsymbol{x}; \boldsymbol{x}^{(t)}\right) + g(\boldsymbol{x})$;
  Compute the step size $\gamma^{(t)}$ by the exact line search (4.65) or the successive line search (4.66) ;
  Let $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)}\left(\widetilde{\boldsymbol{x}}^{(t)} - \boldsymbol{x}^{(t)}\right)$;
  $t \leftarrow t + 1$;
until *converged* **and** *($\boldsymbol{x}^{(t)} \notin \operatorname{int}(\mathcal{X})$* **or** *$\nabla_{\boldsymbol{x}} f\left(\boldsymbol{x}^{(t)}\right)$ exists)*;
return $\boldsymbol{x}^{(t)}$

and, as can be easily verified, the solution for the auxiliary variable is $\tilde{y}^{(t)} = g\left(\widetilde{\boldsymbol{x}}^{(t)}\right)$. This is equivalent to performing a smoothing majorization and then a convex approximation that satisfies Assumptions 4.1 and 4.3, respectively, on the reformulated problem (4.39). However, different from directly performing Algorithm 3 on the reformulated problem (4.41), which only updates the auxiliary variable along the descent direction as $y^{(t+1)} = y^{(t)} + \gamma^{(t)}\left(\tilde{y}^{(t)} - y^{(t)}\right)$, Algorithm 4 implicitly minimizes the problem (4.41) with respect to y with $\boldsymbol{x}$ fixed at $\boldsymbol{x}^{(t+1)}$ after line search by incorporating the update $y^{(t+1)} = g\left(\boldsymbol{x}^{(t+1)}\right)$. This modification, as discussed in Section 3.2, does not destroy the property that the generated solution sequence monotonically decreases the original objective function. Therefore, the convergence of Algorithm 4 can still be justified by Proposition 4.5 and Theorem 4.2 based on the equivalent reformulation in (4.41). That is, every limit point $\boldsymbol{x}^\star$ of the solution sequence generated by the proposed smoothing SCA Algorithm 4 for the regularized problem (4.39) is a C-stationary point of the equivalently reformulated problem (4.41), which satisfies (4.47). Likewise, Proposition 4.3 can be readily extended to Algorithm 4 and used to exclude a subset of stationary points that are not local minima of the original problem (4.39), which results in the additional termination condition incorporated in Algorithm 4.

4.2.3 Connection to Other SCA Algorithms

A similar extension of the SCA framework is introduced in [YPCO18] to address the following composite problem:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \quad u(\boldsymbol{x}) + \underbrace{g^+(\boldsymbol{x}) - g^-(\boldsymbol{x})}_{g(\boldsymbol{x})}, \tag{4.70}$$

where u is smooth but not necessarily convex, and both g^+ and g^- are convex but possibly nonsmooth. That is, the nonsmooth regularization g is nonconvex but has a DC structure. Similar to our smoothing SCA framework, the SCA algorithm proposed in [YPCO18] consists of the three steps, i.e., the smoothing majorization, the convex approximation, and the line search performed on the majorizing function. Specifically, the nonsmooth concave component $-g^-$ is majorized, according to the linearization majorization technique presented in (4.22), by a supporting hyperplane that preserves a subgradient at the current iterate, whereas the nonsmooth convex component g^+, as well as the smooth component u, remain unchanged. It can be immediately concluded that the algorithm in [YPCO18] is a special case of the smoothing SCA framework developed in this section. In other words, in this section, we generalize the idea of [YPCO18] to other types of smoothing majorization techniques, including the examples presented in Section 4.1.2, so as to establish an algorithmic framework for a wide class of nonconvex nonsmooth optimization problems. Moreover, we provide a unified convergence analysis for this generalized algorithmic framework.

4.3 Block-Coordinatewise Versions

Similar to the classic MM and SCA frameworks, the smoothing MM and smoothing SCA algorithms proposed in the previous sections can be implemented in a BCD manner to exploit the potential separable structure of the constraints in the optimization problem, which, together with their convergence analyses in the case where a deterministic block selection rule is used, are presented in this section.

4.3.1 BSUM Framework with Smoothing Majorization

In this subsection, we derive a block-coordinatewise extension of the MM framework with smoothing majorization proposed in Section 4.1 or, equivalently, a smoothing version of the BSUM framework in Section (3.3.2) that successively updates part of the variables by minimizing a global upper bound along the chosen coordinates. Similar to the joint-update case, the smoothing BSUM sacrifices the tightness of the derivative consistency between the coordinatewise upper bound and the original objective function with the benefit that the upper bound is ensured to be smooth and, hence, can be easily minimized.

Consider the previously introduced class of optimization problems in (2.41) with separable constraints. Let us first assume that the objective function f is nonconvex and

nonsmooth but fulfills Assumption 4.2 so that there always exists a smooth majorizing function of f at any point in the feasible set. Similar to the joint-update case, later we will extend the algorithm to address the problem containing an additional convex nonsmooth regularization where the whole objective function can not be directly majorized by a smooth function. We present the smoothing BSUM in the general case where multiple block variables can be selected at each iteration. Specifically, at the tth iteration, let $\mathcal{I}^{(t)} \subseteq \{1, \ldots, K\}$ be the set of indices of the block variables selected to be updated at iteration t and $\boldsymbol{x}_{\mathcal{I}^{(t)}} = (\boldsymbol{x}_k)_{k\in\mathcal{I}^{(t)}}$ be the collection of the selected block variables. We first construct a smooth approximate function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ that majorizes the original objective function f at the current $\boldsymbol{x}^{(t)}$ in the selected coordinates $\mathcal{I}^{(t)}$. Then the selected block variables $\boldsymbol{x}_{\mathcal{I}^{(t)}}$ is updated by minimizing the coordinatewise majorizing function, i.e.,

$$\boldsymbol{x}_{\mathcal{I}^{(t)}} \in \left\{ \underset{\boldsymbol{x}_{\mathcal{I}^{(t)}}}{\operatorname{argmin}}\, \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right) \quad \text{s.t. } \boldsymbol{x}_i \in \mathcal{X}_i, \forall\, i \in \mathcal{I}^{(t)} \right\}, \tag{4.71}$$

which ensures a decrease of the original objective function along the chosen coordinates. The rest of the block variables then retain their values at the current iterate, i.e., $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)}$ for all $i \notin \mathcal{I}^{(t)}$. The complete description of the proposed BSUM algorithm with smoothing majorization for solving problem (2.41) is given in Algorithm 5. In particular, the MM Algorithm 1 with smoothing majorization proposed in Section 4.1 can be regarded as a special instance of Algorithm 5 where all blocks are jointly updated at each iteration, i.e., with $\mathcal{I}^{(t)} = \{1, \ldots, K\}$, or, equivalently, a single-block version of Algorithm 5, i.e., with $K = 1$.

Remark 4.3. *In the case where more than one block variable is to be updated, i.e., $\left|\mathcal{I}^{(t)}\right| > 1$, the approximate function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ is required to be a global upper bound of f when all the chosen block variables $\boldsymbol{x}_{\mathcal{I}^{(t)}}$ can be jointly varied. As a counter example, the following separable approximation*

$$\sum_{i\in\mathcal{I}^{(t)}} \widehat{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right), \tag{4.72}$$

where each component $\widehat{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right)$ is a majorization of f restricted to one selected block variable $\boldsymbol{x}_i$, usually has an advantage of reduced computational complexity as it can be minimized independently and in parallel with respect to each block variable. Nonetheless, the approximation (4.72) *is not guaranteed to be a joint majorization of f along all the chosen coordinates and, hence, it cannot be used.*

As presented in Section 3.3, different rules can be used for the block selection in a coordinatewise scheme. Nevertheless, we establish in the following the convergence of

Algorithm 5: The BSUM Algorithm with Smoothing Majorization for Solving Problem (2.41)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Pick index set $\mathcal{I}^{(t)}$;
  Construct a smooth majorizer $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$ in the selected coordinates $\mathcal{I}^{(t)}$;
  Let

$$\boldsymbol{x}_{\mathcal{I}^{(t)}} \in \left\{ \underset{\boldsymbol{x}_{\mathcal{I}^{(t)}}}{\operatorname{argmin}} \, \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right) \quad \text{s.t.}\, \boldsymbol{x}_i \in \mathcal{X}_i, \forall\, i \in \mathcal{I}^{(t)} \right\};$$

  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)}$ for all $i \notin \mathcal{I}^{(t)}$;
  $t \leftarrow t + 1$;
until *converged* **and** $\bigwedge_{i=1}^{K} \left(\boldsymbol{x}_i^{(t)} \notin \operatorname{int}\left(\mathcal{X}_i\right)$ **or** $\nabla_{\boldsymbol{x}_i} f\left(\boldsymbol{x}^{(t)}\right)$ *exits*$\Big)$;
return $\boldsymbol{x}^{(t)}$

the modified BSUM framework with smoothing majorization only in the case where the blocks are updated in a fixed order. For simplicity of presentation, we consider that a single block variable, say $\boldsymbol{x}_k$, is updated at each iteration. Similar to the joint-update counterpart, the following assumptions are required on each coordinatewise majorizing function $\widehat{f}_k$.

Assumption 4.5 (Smoothing majorization)**.** *Let the coordinatewise approximate functions $\widehat{f}_k(\cdot;\cdot)$ satisfy the following assumptions for $k = 1, \ldots, K$:*

1) *$\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is continuously differentiable in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$ and continuous in $\boldsymbol{y} \in \mathcal{X}$ for any given $\boldsymbol{x}_k \in \mathcal{X}_k$;*
2) *Tangency: $\widehat{f}_k(\boldsymbol{y}_k; \boldsymbol{y}) = f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;*
3) *Upper bound: $\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y}) \geq f(\boldsymbol{y})$ for all $\boldsymbol{x}_k \in \mathcal{X}_k$ and $\boldsymbol{y} \in \mathcal{X}$;*
4) *Subgradient consistency: $\nabla_{\boldsymbol{x}_k} \widehat{f}_k(\boldsymbol{y}_k; \boldsymbol{y}) \in \partial^C_{\boldsymbol{x}_k} f(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$.*

The following additional assumptions that have been included in Assumption 3.4 for the classic BSUM framework are also required for establishing the convergence of the coordinatewise variant of MM.

Assumption 4.6. *We make the following assumptions for all $k = 1, \ldots, K$:*

1) *$\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is quasiconvex in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$;*

2) The subproblem (4.71) *has a unique solution for any* $\boldsymbol{x}^{(t)} \in \mathcal{X}$.

Then the following theorem establishes the convergence of the modified BSUM algorithm with smoothing majorization.

Theorem 4.3 (Convergence of the BSUM algorithm with smoothing majorization)**.** *Suppose that a deterministic block selection rule is used. Provided that Assumptions 4.2, 4.5 and 4.6 are satisfied, then every limit point* $\boldsymbol{z}$ *of the solution sequence generated by Algorithm 5 is a coordinate C-stationary point of the problem* (2.41)*. In addition, if* f *is coordinatewise regular at* $\boldsymbol{z}$*, then* $\boldsymbol{z}$ *is a C-stationary point of* (2.41)*.*

Proof. Theorem 4.3 can be proved by following a similar line of argument as that in [Ber16, Sec. 3.7] for the exact BCD method. Without loss of generality, we prove the convergence of Algorithm 5 in a case where the block variables are updated in the cyclic order from the first to the Kth block since the following argument can be readily extended to an arbitrary deterministic block selection rule. First, the sequence of objective function values $\left(f(\boldsymbol{x}^{(t)})\right)_{t\in\mathbb{N}}$ is monotonically nonincreasing due to the minimization of the majorizing function, i.e.,

$$f\left(\boldsymbol{x}^{(0)}\right) \geq f\left(\boldsymbol{x}^{(1)}\right) \geq f\left(\boldsymbol{x}^{(2)}\right) \geq \cdots .$$

Let $\boldsymbol{z}$ be a limit point of the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$. By the monotone convergence theorem, we have

$$\lim_{t\to\infty} f\left(\boldsymbol{x}^{(t)}\right) = f(\boldsymbol{z}). \tag{4.73}$$

As the number of blocks is finite, each block variable is updated infinitely often in the solution sequence. Hence, without loss of generality, we can assume that there exists a convergent subsequence $\left(\boldsymbol{x}^{(t_j)}\right)_{j\in\mathbb{N}}$ with the limit point $\boldsymbol{z}$ where the Kth block is updated, i.e., for all $j \in \mathbb{N}$,

$$\begin{cases} \boldsymbol{x}_K^{(t_j)} = \underset{\boldsymbol{x}_K\in\mathcal{X}_K}{\operatorname{argmin}} \quad \widehat{f}_K\left(\boldsymbol{x}_K; \boldsymbol{x}^{(t_j-1)}\right) \\ \boldsymbol{x}_i^{(t_j)} = \boldsymbol{x}_i^{(t_j-1)} \quad \forall\, i = 1, \ldots, K-1. \end{cases}$$

Then, the first block is updated in the next iteration, i.e.,

$$\boldsymbol{x}_1^{(t_j+1)} = \underset{\boldsymbol{x}_1\in\mathcal{X}_1}{\operatorname{argmin}} \quad \widehat{f}_1\left(\boldsymbol{x}_1; \boldsymbol{x}^{(t_j)}\right).$$

Consequently, we have

$$\begin{aligned} \widehat{f}_1\left(\boldsymbol{x}_1^{(t_{j+1})}; \boldsymbol{x}^{(t_{j+1})}\right) &= f\left(\boldsymbol{x}^{(t_{j+1})}\right) \leq f\left(\boldsymbol{x}^{(t_j+1)}\right) \\ &\leq \widehat{f}_1\left(\boldsymbol{x}^{(t_j+1)}; \boldsymbol{x}^{(t_j)}\right) \leq \widehat{f}_1\left(\boldsymbol{x}_1; \boldsymbol{x}^{(t_j)}\right) \quad \forall\, \boldsymbol{x}_1 \in \mathcal{X}_1. \end{aligned}$$

Taking limits for $j \to \infty$ leads to

$$\widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z}) \leq \widehat{f}_1(\boldsymbol{x}_1; \boldsymbol{z}) \quad \forall\, \boldsymbol{x}_1 \in \mathcal{X}_1, \tag{4.74}$$

which suggests that $\boldsymbol{z}_1$ is the global minimizer of the majorizing function $\widehat{f}_1(\boldsymbol{x}_1; \boldsymbol{z})$ on $\mathcal{X}_1$. Hence, $\boldsymbol{z}_1$ satisfies the C-stationarity condition (2.39), i.e.,

$$\mathbf{0} \in \partial^C_{\boldsymbol{x}_1} \widehat{f}_1\left(\boldsymbol{z}_1; \boldsymbol{z}\right) + \mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1), \tag{4.75}$$

where $\partial^C_{\boldsymbol{x}_1} \widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z}) = \left\{\nabla_{\boldsymbol{x}_1} \widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z})\right\}$ due to the smoothness of $\widehat{f}_1$. The subgradient consistency condition in Assumption 4.5 implies that

$$\partial^C_{\boldsymbol{x}_1} \widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1) \subseteq \partial^C_{\boldsymbol{x}_1} f(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1). \tag{4.76}$$

Combining (4.75) and (4.76), we obtain

$$\mathbf{0} \in \partial^C_{\boldsymbol{x}_1} f(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1), \tag{4.77}$$

which implies that f is C-stationary at $\boldsymbol{z}$ along the coordinates of the first block $\boldsymbol{x}_1$.

The next objective is to show that the subsequence $\left(\boldsymbol{x}^{(t_j+1)}\right)_{j\in\mathbb{N}}$ also converges to the same limit point $\boldsymbol{z}$, so that, by repeating the preceding argument on the subproblem

$$\boldsymbol{x}_2^{(t_j+2)} = \underset{\boldsymbol{x}_2 \in \mathcal{X}_2}{\operatorname{argmin}} \quad \widehat{f}_2\left(\boldsymbol{x}_2; \boldsymbol{x}^{(t_j+1)}\right),$$

we will have

$$\mathbf{0} \in \partial^C_{\boldsymbol{x}_2} f(\boldsymbol{z}_2; \boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_2}(\boldsymbol{z}_2). \tag{4.78}$$

Moreover, by repeating the above procedure, we can successively obtain

$$\mathbf{0} \in \partial^C_{\boldsymbol{x}_k} f(\boldsymbol{z}_k; \boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_k}(\boldsymbol{z}_k) \quad \forall\, k = 1, \ldots, K, \tag{4.79}$$

which justifies that the limit point $\boldsymbol{z}$ is a coordinatewise C-stationary point of the original problem (2.41).

Now we show that the sequence $\left(\boldsymbol{x}^{(t_j+1)}\right)_{t\in\mathbb{N}}$ also converges to $\boldsymbol{z}$ or, equivalently, $\boldsymbol{x}_1^{(t_j+1)} \to \boldsymbol{z}_1$ since $\boldsymbol{x}^{(t_j+1)}$ differs from $\boldsymbol{x}^{(t_j)}$ only on the first block. To this end, we assume the contrary or, equivalently, the difference $\boldsymbol{x}_1^{(t_j+1)} - \boldsymbol{x}_1^{(t_j)}$ does not converge to zero. Define

$$\alpha^{(t_j)} = \left\| \boldsymbol{x}_1^{(t_j+1)} - \boldsymbol{x}_1^{(t_j)} \right\|_2,$$

and the normalized difference

$$\boldsymbol{s}_1^{(t_j)} = \frac{\boldsymbol{x}_1^{(t_j+1)} - \boldsymbol{x}_1^{(t_j)}}{\alpha^{(t_j)}}.$$

By possibly restricting to a subsequence of (t_j), there exists an $\breve{\alpha} > 0$ and a $\breve{j} \in \mathbb{N}$ such that

$$\alpha^{(t_j)} \geq \breve{\alpha} \quad \forall\, j \geq \breve{j}.$$

Thus, for any given $\epsilon \in [0,1]$, we have $0 \leq \epsilon\breve{\alpha} \leq \alpha^{(t_j)}$, this implies that the vector $\boldsymbol{x}_1^{(t_j)} + \epsilon\breve{\alpha}\boldsymbol{s}_1^{(t_j)}$ lies on the line segment from $\boldsymbol{x}_1^{(t_j)}$ to $\boldsymbol{x}_1^{(t_j)} + \alpha^{(t_j)}\boldsymbol{s}^{(t_j)} = \boldsymbol{x}^{(t_j+1)}$. Due to the majorization Assumption 4.5, we have

$$f\left(\boldsymbol{x}^{(t_j+1)}\right) = \widehat{f}_1\left(\boldsymbol{x}_1^{(t_j+1)}; \boldsymbol{x}^{(t_j)}\right) \leq \widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)} + \epsilon\breve{\alpha}\boldsymbol{s}_1^{(t_j)}; \boldsymbol{x}^{(t_j)}\right) \tag{4.80a}$$

$$\leq \widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)}; \boldsymbol{x}^{(t_j)}\right) = f\left(\boldsymbol{x}^{(t_j)}\right) \quad \forall\, \epsilon \in [0,1], \tag{4.80b}$$

where (4.80a) comes from the minimization of the majorizing function and (4.80b) from the quasiconvexity of the majorizing function. As $\|\boldsymbol{s}_1^{(t_j)}\| = 1$, the sequence $\left(\boldsymbol{s}_1^{(t_j)}\right)$ belongs to a compact set and, hence, has a limit point $\boldsymbol{s}^\star$. The convergence of the objective function value shown in (4.73) ensure that both subsequences $\left(f(\boldsymbol{x}^{(t_j)})\right)$ and $\left(f(\boldsymbol{x}^{(t_j+1)})\right)$ converge to $f(\boldsymbol{z})$ as $j \to \infty$. We further restrict to a subsequence of $\left(\boldsymbol{s}_1^{(t_j)}\right)$ that converges to $\boldsymbol{s}_1^\star$ and take the limits of (4.80) for $j \to \infty$, which leads to the following bounding property:

$$f(\boldsymbol{z}) \leq \widehat{f}_1\left(\boldsymbol{z}_1 + \epsilon\breve{\alpha}\boldsymbol{s}_1^\star; \boldsymbol{z}\right) \leq \widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z}) \leq f(\boldsymbol{z}) \quad \forall\, \epsilon \in [0,1], \tag{4.81}$$

or, equivalently,

$$f(\boldsymbol{z}) = \widehat{f}_1(\boldsymbol{z}_1; \boldsymbol{z}) = \widehat{f}_1\left(\boldsymbol{z}_1 + \epsilon\breve{\alpha}\boldsymbol{s}_1^\star; \boldsymbol{z}\right) \quad \forall\, \epsilon \in [0,1]. \tag{4.82}$$

This contradicts the assumption that the majorizing function is uniquely minimized with respect to the selected coordinates at each iteration if $\breve{\alpha}\boldsymbol{s}_1^\star \neq \boldsymbol{0}$. Hence, the subsequence $\left(\boldsymbol{x}^{(t_j+1)}\right)_{t\in\mathbb{N}}$ must converge to $\boldsymbol{z}$, which, as aforementioned, can be used to justify that f is C-stationary at $\boldsymbol{z}$ along the coordinates of the second block $\boldsymbol{x}_2$.

By repeating the preceding procedure, we can easily prove that the original objective function f is C-stationary at $\boldsymbol{z}$ along each block of coordinates, respectively, and conclude that $\boldsymbol{z}$ is a coordinatewise C-stationary point of the original problem (2.41). When the original objective function f is coordinatewise regular at $\boldsymbol{z}$, the coordinatewise C-stationarity of f at $\boldsymbol{z}$ also implies the joint C-stationarity of f at $\boldsymbol{z}$. □

In addition, Proposition 4.3 can be coordinatewise applied to the problem (2.41) to identify a subset of coordinatewise C-stationary points of (2.41) that are not coordinatewise local minima. Suppose that the coordinatewise smooth majorizer $\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is stationary at $\boldsymbol{y} \in \mathcal{X}$ with $\boldsymbol{y}_k \in \operatorname{int}(\mathcal{X}_k)$, which implies that the original objective function f is C-stationary along the coordinates of $\boldsymbol{x}_k$. By Proposition 4.3, if the partial

gradient $\nabla_{\boldsymbol{x}_k} f(\boldsymbol{y})$ does not exist, then f does not achieve a local minimum at $\boldsymbol{y}$ along the coordinates of $\boldsymbol{x}_k$ as a strict descent direction of f along the coordinates of $\boldsymbol{x}_k$ exists at $\boldsymbol{y}$. This property motivates the additional termination condition in Algorithm 5 to exclude those coordinatewise C-stationary points.

Similar to the joint-update counterpart – the smooth MM Algorithm 1, the above procedure and convergence result can be readily extended to the following subclass of the composite problem in (4.39):

$$\begin{aligned} \min_{\{\boldsymbol{x}_k\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) + \underbrace{\sum_{k=1}^K g_k(\boldsymbol{x}_k)}_{g(\boldsymbol{x})} \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k, k = 1, \ldots, K, \end{aligned} \tag{4.83}$$

where both the convex nonsmooth regularization g and the constraints are separable across the blocks of variables. Each component $g_k(\boldsymbol{x}_k)$ of the regularization term is convex in the corresponding coordinates, but not necessarily smooth. The nonconvex nonsmooth function f satisfies Assumption 4.2. However, as explained in Section 4.1, due to the additional convex nonsmooth regularization g, Assumption 4.2 may not hold for the overall objective function of (4.83), which hinders the application of Algorithm 5. Moreover, as the regularization g is also separable across the blocks of variables, we can leave g unchanged and construct a smooth majorizer only for f along the chosen coordinates. That is, the following modified approximate problem is solved at the tth iteration:

$$\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t+1)} \in \left\{ \operatorname*{argmin}_{\boldsymbol{x}_{\mathcal{I}^{(t)}}} \widehat{f}_{\mathcal{I}^{(t)}} \left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)} \right) + \sum_{i \in \mathcal{I}^{(t)}} g_i(\boldsymbol{x}_i) \quad \text{s.t. } \boldsymbol{x}_i \in \mathcal{X}_i, \forall\, i \in \mathcal{I}^{(t)} \right\}, \tag{4.84}$$

where $\widehat{f}_{\mathcal{I}^{(t)}} \left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)} \right)$ is designed to be a smooth majorizer of f at $\boldsymbol{x}^{(t)}$ along the chosen coordinates of $\boldsymbol{x}_{\mathcal{I}^{(t)}}$. The complete description of this extension of the smooth BSUM algorithm to address the regularized problem (4.83) is given in Algorithm 6

The convergence of the modified smooth BSUM Algorithm 6 can be justified by following the same line of analysis as in Section 4.1. Specifically, Algorithm 6 can be demonstrated to be equivalent to performing the smooth BSUM Algorithm 5 on the following equivalent reformulation of (4.83):

$$\begin{aligned} \min_{\{\boldsymbol{x}_k \in \mathbb{R}^{n_k},\, y_k \in \mathbb{R}\}_{k=1}^K} \quad & f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) + \sum_{k=1}^K y_k \\ \text{s.t.} \quad & \boldsymbol{x}_k \in \mathcal{X}_k,\, g_k(\boldsymbol{x}_k) \le y_k,\, k = 1, \ldots, K. \end{aligned} \tag{4.85}$$

Algorithm 6: The BSUM Algorithm with Smoothing Majorization for Solving Problem (4.83)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Pick index set $\mathcal{I}^{(t)}$;
  Construct a smooth majorizer $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$ in the selected coordinates $\mathcal{I}^{(t)}$;
  Let

$$\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t+1)} \in \left\{ \underset{\boldsymbol{x}_{\mathcal{I}^{(t)}}}{\operatorname{argmin}} \, \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right) + \sum_{i \in \mathcal{I}^{(t)}} g_i(\boldsymbol{x}_i) \quad \text{s.t.}\, \boldsymbol{x}_i \in \mathcal{X}_i, \forall\, i \in \mathcal{I}^{(t)} \right\};$$

  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)}$ for all $i \notin \mathcal{I}^{(t)}$;
  $t \leftarrow t + 1$;
until *converged* **and** $\bigwedge_{i=1}^{K} \left(\boldsymbol{x}_i^{(t)} \notin \operatorname{int}\left(\mathcal{X}_i\right) \textbf{ or } \nabla_{\boldsymbol{x}_i} f\left(\boldsymbol{x}^{(t)}\right) \textit{ exits}\right)$;
return $\boldsymbol{x}^{(t)}$

Thus, by Theorem 4.3, every limit point $\boldsymbol{z}$ of the solution sequence generated by Algorithm 6 is a coordinatewise C-stationary point of the reformulation (4.85) that satisfies

$$\mathbf{0} \in \partial^C_{\boldsymbol{x}_k} f(\boldsymbol{z}) + \partial^C_{\boldsymbol{x}_k} g(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}_k}(\boldsymbol{z}_k) \quad \forall\, k = 1, \ldots, K. \tag{4.86}$$

Moreover, the problem (4.85) is coordinatewise regular if f is coordinatewise regular. If this is true, then $\boldsymbol{z}$ becomes a joint C-stationary point of (4.85) that satisfies

$$\mathbf{0} \in \partial^C f(\boldsymbol{z}) + \partial^C g(\boldsymbol{z}) + \mathcal{N}_{\mathcal{X}}(\boldsymbol{z}). \tag{4.87}$$

4.3.2 BSCA Framework with Smoothing Majorization

Similar to the single-block version – the smoothing SCA algorithm in Section 4.2, the idea of convex approximation can be employed to efficiently obtain an approximate minimizer of the coordinatewise smooth majorizing function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}\right)$ when the exact minimization of $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}\right)$ is still difficult. Thus, in the following, we present a block-coordinatewise version of the smoothing SCA framework, referred to as the *smoothing BSCA* framework.

At the tth iteration, instead of minimizing the smooth majorizing function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}\right)$ exactly, we further construct a convex approximation $\widetilde{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}\right)$ of $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}\right)$ at the current iterate $\boldsymbol{x}^{(t)}$ that can be easily minimized. Moreover, when

multiple block variables are selected, i.e., $\left|\mathcal{I}^{(t)}\right| > 1$, unlike the majorization, the convex approximation can be designed to be separable across the blocks to better take advantage of the separable structure of the constraints. Specifically, we employ the following type of approximation:

$$\widetilde{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}^{(t)}\right)=\sum_{i\in\mathcal{I}^{(t)}}\widetilde{f}_i\left(\boldsymbol{x}_i;\boldsymbol{x}^{(t)}\right), \tag{4.88}$$

where $\widetilde{f}_i\left(\boldsymbol{x}_i;\boldsymbol{x}^{(t)}\right)$ is a convex approximation of the smooth majorizing function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}\right)$ along the coordinates of each chosen block $\boldsymbol{x}_i$. Then the approximate problem at iteration t is given by

$$\widetilde{\boldsymbol{x}}_{\mathcal{I}^{(t)}}\in\left\{\underset{\boldsymbol{x}_{\mathcal{I}^{(t)}}}{\operatorname{argmin}}\,\widetilde{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}^{(t)}\right)\quad\text{s.t.}\,\boldsymbol{x}_i\in\mathcal{X}_i,\forall\, i\in\mathcal{I}^{(t)}\right\}, \tag{4.89}$$

which apparently can be decomposed into the following subproblems:

$$\widetilde{\boldsymbol{x}}_i\in\left\{\underset{\boldsymbol{x}_i\in\mathcal{X}_i}{\operatorname{argmin}}\,\widetilde{f}_i\left(\boldsymbol{x}_i;\boldsymbol{x}^{(t)}\right)\right\}\quad\forall\, i\in\mathcal{I}^{(t)}. \tag{4.90}$$

Each subproblem in (4.90) exclusively depends on a single block variable and, hence, can be solved independently and in parallel. The difference $\widetilde{\boldsymbol{x}}^{(t)}_{\mathcal{I}^{(t)}}-\boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}}$ indicates a descent direction of the smooth majorizing function due to the convexity of the approximate function. The variable $\boldsymbol{x}$ is then updated according to the following rule:

$$\boldsymbol{x}_i^{(t+1)}=\begin{cases}\boldsymbol{x}_i^{(t)}+\gamma^{(t)}\left(\widetilde{\boldsymbol{x}}_i^{(t)}-\boldsymbol{x}_i^{(t)}\right) & \forall\, i\in\mathcal{I}^{(t)},\\ \boldsymbol{x}_i^{(t)} & \text{otherwise,}\end{cases} \tag{4.91}$$

That is, only the selected block variables are updated along the descent direction given by the solutions of the convex approximations with a suitable step size $\gamma^{(t)}\in(0,1]$, whereas the other block variables retain their values at the current iterate.

As aforementioned, we focus on the line search approach introduced in Section 3.2 for choosing a step size $\gamma^{(t)}$, including the exact and the successive line search. Similar to the joint-update counterpart in Section 4.2, we perform a line search method on the coordinatewise majorizing function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}^{(t)}\right)$ due to its smoothness and majorization property, which ensures that the step size $\gamma^{(t)}$ also provides a sufficient decrease of the original function f along the considered coordinates of $\boldsymbol{x}_{\mathcal{I}^{(t)}}$. Specifically, the exact line search in (3.17) and the successive line search with Armijo rule in (3.19) are customized for the coordinatewise majorizing function $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}^{(t)}\right)$ as follows. In the exact line search, the step size $\gamma^{(t)}$ is chosen to minimize $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}};\boldsymbol{x}^{(t)}\right)$ along the descent direction $\widetilde{\boldsymbol{x}}^{(t)}_{\mathcal{I}^{(t)}}-\boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}}$, i.e.,

$$\gamma^{(t)}\in\underset{\gamma\in[0,1]}{\operatorname{argmin}}\quad\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}}+\gamma\left(\widetilde{\boldsymbol{x}}^{(t)}_{\mathcal{I}^{(t)}}-\boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}}\right);\boldsymbol{x}^{(t)}\right). \tag{4.92}$$

Algorithm 7: The BSCA Algorithm Extended by Smoothing Majorization for Solving Problem (2.41)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;
repeat
  Pick index set $\mathcal{I}^{(t)}$;
  Construct a smooth majorizer $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$ in the selected coordinates $\mathcal{I}^{(t)}$;
  Construct coordinatewise convex approximations $\widetilde{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right)$ of $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ at $\boldsymbol{x}^{(t)}$ along the coordinates of each selected block $\boldsymbol{x}_i$, respectively, for all $i \in \mathcal{I}^{(t)}$;
  Let
$$\widetilde{\boldsymbol{x}}_i \in \left\{ \underset{\boldsymbol{x}_i \in \mathcal{X}_i}{\operatorname{argmin}} \widetilde{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right) \right\} \quad \forall\, i \in \mathcal{I}^{(t)};$$
  Compute the step size $\gamma^{(t)}$ by the exact line search (4.92) or the successive line search (4.93);
  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)} + \gamma^{(t)}\left(\widetilde{\boldsymbol{x}}_i^{(t)} - \boldsymbol{x}_i^{(t)}\right)$ for all $i \in \mathcal{I}^{(t)}$;
  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)}$ for all $i \notin \mathcal{I}^{(t)}$;
  $t \leftarrow t + 1$;
until *converged* **and** $\bigwedge_{i=1}^{K}\left(\boldsymbol{x}_i^{(t)} \notin \operatorname{int}\left(\mathcal{X}_i\right)\right.$ **or** $\left.\nabla_{\boldsymbol{x}_i} f\left(\boldsymbol{x}^{(t)}\right)\ \textit{exits}\right)$;
return $\boldsymbol{x}^{(t)}$

As an alternative, the Armjio rule can be used when the optimization problem (4.92) cannot be solved efficiently. In the Armijo rule, the step size is successively decreased at a geometric rate until a condition on the sufficient decrease of $\widehat{f}_{\mathcal{I}^{(t)}}$ is fulfilled, which is expressed as the following optimization problem:

$$\begin{aligned} \gamma^{(t)} = \underset{\gamma}{\operatorname{argmax}} \quad & \gamma \\ \text{s.t.} \quad & \gamma \in \left\{\beta^k \mid k \in \mathbb{N}\right\}, \\ & \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)} + \gamma\left(\widetilde{\boldsymbol{x}}_{\mathcal{I}^{(t)}}^{(t)} - \boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}\right); \boldsymbol{x}^{(t)}\right) - \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}; \boldsymbol{x}^{(t)}\right) \leq \gamma\sigma\delta \end{aligned} \tag{4.93}$$

with the directional derivative $\delta = \sum_{i \in \mathcal{I}^{(t)}}\left(\nabla_{\boldsymbol{x}_i}\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}\right)\right)^{\mathsf{T}}\left(\widetilde{\boldsymbol{x}}_i^{(t)} - \boldsymbol{x}_i^{(t)}\right)$ and $0 < \beta, \sigma < 1$. The proposed smoothing BSCA framework for solving the problem (2.41) with separable constraints is then outlined in Algorithm 7.

In the following, we demonstrate the convergence of the proposed BSCA framework extended by smoothing majorization in the case where the blocks are updated in a fixed order. For simplicity of presentation, we consider that only a single block variable, say $\boldsymbol{x}_k$, is updated at each iteration. First, the following assumptions are made on each

approximate function $\widetilde{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right)$ so that it is a convex approximation of the corresponding coordinatewise smooth majorizer $\widehat{f}_k(\boldsymbol{x}_k; \boldsymbol{x}^{(t)})$ along the chosen coordinates.

Assumption 4.7 (Convex approximation)**.** *Let the coordinatewise approximate functions $\widetilde{f}_k(\cdot;\cdot)$ satisfy the following assumptions for $k = 1, \ldots, K$:*

1) $\widetilde{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ are continuously differentiable in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$ and continuous in $\boldsymbol{y} \in \mathcal{X}$ for any given $\boldsymbol{x}_k \in \mathcal{X}_k$;

2) Gradient consistency: $\nabla_{\boldsymbol{x}_k} \widetilde{f}(\boldsymbol{y}_k; \boldsymbol{y}) = \nabla_{\boldsymbol{x}_k} \widehat{f}(\boldsymbol{y}_k; \boldsymbol{y})$ for all $\boldsymbol{y} \in \mathcal{X}$;

3) $\widetilde{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is strictly convex in $\boldsymbol{x}_k \in \mathcal{X}_k$ for any given $\boldsymbol{y} \in \mathcal{X}$.

Note that, different from the joint-update counterpart, each coordinatewise approximation $\widetilde{f}_k(\boldsymbol{x}_k; \boldsymbol{y})$ is required to be strictly convex, instead of pseudoconvex. Moreover, as in the classic BSCA framework, the following mild assumption is also required.

Assumption 4.8. *The map*

$$\widetilde{M}_k\left(\boldsymbol{x}^{(t)}\right) = \left\{ \operatorname*{argmin}_{\boldsymbol{x}_k \in \mathcal{X}_k} \widetilde{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right) \right\}$$

is nonempty for $t \in \mathbb{N}$.

Then the following theorem establishes the convergence of the proposed smoothing BSCA algorithm.

Theorem 4.4 (Convergence of the smoothing BSCA algorithm)**.** *Suppose that the line search approach is employed and that the block variables are updated in a fixed order. Provided that Assumptions 4.2, 4.5, 4.7, and 4.8 are satisfied, then every limit point $\boldsymbol{z}$ of the solution sequence generated by Algorithm 7 is a stationary point of the problem* (2.41)*. In addition, if f is coordinatewise regular at $\boldsymbol{z}$, then $\boldsymbol{z}$ is a C-stationary point of* (2.41)*.*

Proof. The proof is similar to that in [RHL13, YPLO20] for the classic BSCA framework. Without loss of generality, we prove the convergence of Algorithm 5 in a case where the block variables are updated in the cyclic order from the first to the Kth block since the following argument can be readily extended to an arbitrary deterministic block selection rule.

Let $\boldsymbol{z}$ be a limit point of the solution sequence $\left(\boldsymbol{x}^{(t)}\right)_{t\in\mathbb{N}}$. The monotonically nonincreasing property of the sequence of objective function values $\left(f(\boldsymbol{x}^{(t)})\right)_{t\in\mathbb{N}}$ ensures its convergence, i.e.,

$$\lim_{t\to\infty} f\left(\boldsymbol{x}^{(t)}\right) = f(\boldsymbol{z}).$$

Assume that the block k_t is selected to be updated at the tth iteration. Since the step size is obtained by performing the Armijo rule on the smooth majorizing function $\widehat{f}_{k_t}\left(\boldsymbol{x}_{k_t};\boldsymbol{x}^{(t)}\right)$, we have

$$\begin{aligned} f\left(\boldsymbol{x}^{(t+1)}\right) - f\left(\boldsymbol{x}^{(t)}\right) &\leq \widehat{f}_{k_t}\left(\boldsymbol{x}_{k_t}^{(t+1)};\boldsymbol{x}^{(t)}\right) - \widehat{f}_{k_t}\left(\boldsymbol{x}_{k_t}^{(t)};\boldsymbol{x}^{(t)}\right) \\ &\leq \sigma\gamma^{(t)}\left(\nabla_{\boldsymbol{x}_{k_t}}\widehat{f}_{k_t}\left(\boldsymbol{x}_{k_t}^{(t)};\boldsymbol{x}^{(t)}\right)\right)^{\mathsf{T}}\boldsymbol{d}_{k_t}^{(t)} \leq 0, \end{aligned}$$

where $\boldsymbol{d}_{k_t}^{(t)} = \widetilde{\boldsymbol{x}}_{k_t}^{(t)} - \boldsymbol{x}_{k_t}^{(t)}$ denotes the descent direction at the tth iteration. The convergence of the objective function value implies that

$$\lim_{t\to\infty}\gamma^{(t)}\left(\nabla_{\boldsymbol{x}_{k_t}}\widehat{f}_{k_t}\left(\boldsymbol{x}_{k_t}^{(t)};\boldsymbol{x}^{(t)}\right)\right)^{\mathsf{T}}\boldsymbol{d}_{k_t}^{(t)} = 0. \tag{4.94}$$

Without loss of generality, we can assume that there exists a convergent subsequence $\left(\boldsymbol{x}^{(t_j)}\right)_{j\in\mathbb{N}}$ with the limit point $\boldsymbol{z}$ where the Kth block is updated. Then the first block is to be updated in the next iteration, i.e.,

$$\widetilde{\boldsymbol{x}}_1^{(t_j+1)} \in \underset{\boldsymbol{x}_1\in\mathcal{X}_1}{\operatorname{argmin}} \quad \widetilde{f}_1\left(\boldsymbol{x}_1;\boldsymbol{x}^{(t_j)}\right).$$

We claim that, by further restricting to a subsequence if necessary, the descent direction $\boldsymbol{d}_1^{(t_j)}$ converges to zero as $j\to\infty$, i.e.,

$$\lim_{j\to\infty}\boldsymbol{d}_1^{(t_j)} = \boldsymbol{0}. \tag{4.95}$$

We show this by contradiction. We assume the contrary that there exists an $\alpha\in(0,1)$ and a $\breve{j}\in\mathbb{N}$ such that

$$\left\|\boldsymbol{d}_1^{(t_j)}\right\|_2 \geq \alpha \quad \forall\, j\geq\breve{j}. \tag{4.96}$$

Define the normalized descent direction

$$\boldsymbol{p}_1^{(t_j)} = \frac{\boldsymbol{d}_1^{(t_j)}}{\left\|\boldsymbol{d}_1^{(t_j)}\right\|_2}.$$

It follows from (4.94) that

$$\lim_{j\to\infty}\gamma^{(t_j)}\left\|\boldsymbol{d}_1^{(t_j)}\right\|_2\left(\nabla_{\boldsymbol{x}_1}\widehat{f}\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{(t_j)} = 0, \tag{4.97}$$

This implies that either of the following two cases is true and we demonstrate the contradiction that none of the two cases can be true.

Case A: The first case implied by (4.97) is that $\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{(t_j)}\to 0$ along a subsequence of (t_j). As $\|\boldsymbol{p}_1^{(t_j)}\|_2=1$, the sequence $\left(\boldsymbol{p}_1^{(t_j)}\right)$ belongs to a compact set and, hence, has a limit point $\boldsymbol{p}_1^{\star}$. By taking the limit of the directional derivative $\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{(t_j)}$ along a convergent subsequence, we obtain

$$\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1(\boldsymbol{z}_1;\boldsymbol{z})\right)^{\mathsf{T}}\boldsymbol{p}_1^{\star}=0.$$

the strict convexity of the approximate function $\widetilde{f}_1(\boldsymbol{x}_1;\boldsymbol{z})$ in $\boldsymbol{x}_1$ implies that

$$\widetilde{f}_1(\boldsymbol{z}_1+\alpha\boldsymbol{p}_1^{\star};\boldsymbol{z})>\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})+\alpha\left(\nabla_{\boldsymbol{x}_1}\widetilde{f}_1\left(\boldsymbol{z}_1;\boldsymbol{z}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{\star}\geq\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z}),\tag{4.98}$$

where the last inequality comes from the subgradient consistency condition in Assumption 4.7. On the other hand, since $\boldsymbol{x}_1^{(t_j)}+\alpha\boldsymbol{p}_1^{(t_j)}$ lies on the line segment from $\boldsymbol{x}_1^{(t_j)}$ to the minimizer $\widetilde{\boldsymbol{x}}_1^{(t_j)}$ of the approximate function $\widetilde{f}_1$, the convexity of $\widetilde{f}_1$ also implies that

$$\widetilde{f}_1\left(\boldsymbol{x}_1^{(t_j)}+\alpha\boldsymbol{p}_1^{(t_j)}\right)\leq\widetilde{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right).\tag{4.99}$$

Taking limits for $j\to\infty$ leads to

$$\widetilde{f}_1\left(\boldsymbol{z}_1+\alpha\boldsymbol{p}_1^{\star};\boldsymbol{z}\right)\leq\widetilde{f}_1\left(\boldsymbol{z}_1;\boldsymbol{z}\right).\tag{4.100}$$

This contradicts (4.98) and, hence, Case A cannot be true.

Case B: The second case implied by (4.97) is that $\gamma^{(t_j)}\|\boldsymbol{d}_1^{(t_j)}\|_2\to 0$ along a subsequence of (t_j). Let us restrict to that subsequence. Due to the hypothesis (4.96), we have

$$\lim_{j\to\infty}\gamma^{(t_j)}=0,$$

which further implies that there exists a $\breve{j}'\in\mathbb{N}$ such that, for all $j\geq\breve{j}'$,

$$\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)}+\frac{\gamma^{(t_j)}}{\beta}\boldsymbol{d}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)-\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)>\sigma\frac{\gamma^{(t_j)}}{\beta}\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\right)^{\mathsf{T}}\boldsymbol{d}_1^{(t_j)}.$$

Rearranging the terms, we obtain

$$\frac{\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)}+\frac{\gamma^{(t_j)}}{\beta}\|\boldsymbol{d}_1^{(t_j)}\|_2\boldsymbol{p}_1^{(t-j)};\boldsymbol{x}^{(t_j)}\right)-\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)}{\frac{\gamma^{(t_j)}}{\beta}\|\boldsymbol{d}_1^{(t_j)}\|_2}>$$
$$\sigma\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{x}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{(t_j)}\quad\forall\, j\geq\breve{j}'.$$

Letting $j\to\infty$ along the subsequence such that $\boldsymbol{p}_1^{(t_j)}\to\boldsymbol{p}_1^{\star}$, we have

$$\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{z}_1;\boldsymbol{z}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{\star}\geq\sigma\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{z}_1;\boldsymbol{z}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{\star},$$

which implies that $\left(\nabla_{\boldsymbol{x}_1}\widehat{f}_1\left(\boldsymbol{z}_1;\boldsymbol{z}\right)\right)^{\mathsf{T}}\boldsymbol{p}_1^{\star}\geq 0$ since $0<\sigma<1$. Consequently, following the same line of argument as in the previous case, we can justify that both (4.98) and (4.99) hold, which is a contradiction. Thus, we can conclude that the hypothesis (4.96) cannot be true and, on the contrary, the condition (4.95) must hold. In other words, the approximate solution $\widetilde{\boldsymbol{x}}_1^{(t_j)}$ converges to the same limit point $\boldsymbol{z}_1$ as $\boldsymbol{x}_1^{(t_j)}$ in some subsequence.

On the other hand, $\widetilde{\boldsymbol{x}}_1^{(t_j)}$ is the minimizer of the approximate function $\widetilde{f}_1\left(\boldsymbol{x}_1;\boldsymbol{x}^{(t_j)}\right)$, i.e.,

$$\widetilde{f}_1\left(\widetilde{\boldsymbol{x}}_1^{(t_j)};\boldsymbol{x}^{(t_j)}\right)\leq\widetilde{f}_1\left(\boldsymbol{x}_1;\boldsymbol{x}^{(t_j)}\right)\quad\forall\,\boldsymbol{x}_1\in\mathcal{X}_1.$$

Taking limits in the aforementioned convergent subsequence leads to

$$\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})\leq\widetilde{f}_1(\boldsymbol{x}_1;\boldsymbol{z})\quad\forall\,\boldsymbol{x}_1\in\mathcal{X}_1.$$

This implies that $\boldsymbol{z}_1$ is a global minimizer of the approximate function $\widetilde{f}_1(\boldsymbol{x}_1;\boldsymbol{z})$ on $\mathcal{X}_1$ and, hence, satisfies the C-stationarity condition (2.39), i.e.,

$$\boldsymbol{0}\in\partial_{\boldsymbol{x}_1}^{C}\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1).\tag{4.101}$$

From the subgradient consistency conditions in Assumption 4.5 and 4.7, we have

$$\partial_{\boldsymbol{x}_1}^{C}\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1)=\partial_{\boldsymbol{x}_1}^{C}\widehat{f}(\boldsymbol{z}_1;\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1)\subseteq\partial_{\boldsymbol{x}_1}^{C}f(\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1),\tag{4.102}$$

where $\partial_{\boldsymbol{x}_1}^{C}\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})=\left\{\nabla_{\boldsymbol{x}_1}\widetilde{f}_1(\boldsymbol{z}_1;\boldsymbol{z})\right\}$ and $\partial_{\boldsymbol{x}_1}^{C}\widehat{f}_1(\boldsymbol{z}_1;\boldsymbol{z})=\left\{\nabla_{\boldsymbol{x}_1}\widehat{f}_1(\boldsymbol{z}_1;\boldsymbol{z})\right\}$ due to the smoothness. Combining (4.101) and (4.102), we obtain

$$\boldsymbol{0}\in\partial_{\boldsymbol{x}_1}^{C}f(\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_1}(\boldsymbol{z}_1),\tag{4.103}$$

which implies that f is C-stationary at $\boldsymbol{z}$ along the coordinates of the first block $\boldsymbol{x}_1$.

Furthermore, since $\widetilde{\boldsymbol{x}}_1^{(t_j)}$ converges to $\boldsymbol{z}_1$ as $j\to\infty$, the next iterate $\boldsymbol{x}^{(t_j+1)}$, where the first block is updated toward $\widetilde{\boldsymbol{x}}_1^{(t_j)}$, also converges to $\boldsymbol{z}$, i.e., $\lim_{j\to\infty}\boldsymbol{x}^{(t_j+1)}=\boldsymbol{z}$. Therefore, by repeating the preceding argument on the subproblem at the next iteration

$$\widetilde{\boldsymbol{x}}_2^{(t_j+1)}=\underset{\boldsymbol{x}_2\in\mathcal{X}_2}{\operatorname{argmin}}\quad\widetilde{f}_2\left(\boldsymbol{x}_2;\boldsymbol{x}^{(t_j+1)}\right),$$

we will have

$$\boldsymbol{0}\in\partial_{\boldsymbol{x}_2}^{C}f(\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_2}(\boldsymbol{z}_2),\tag{4.104}$$

and $\lim_{j\to\infty}\boldsymbol{x}^{(t_j+2)}=\boldsymbol{z}$. Moreover, we can continue to obtain

$$\boldsymbol{0}\in\partial_{\boldsymbol{x}_k}^{C}f(\boldsymbol{z})+\mathcal{N}_{\mathcal{X}_k}(\boldsymbol{z}_k)\quad\forall\,k=1,\ldots,K,\tag{4.105}$$

which justifies that the limit point $\boldsymbol{z}$ is a coordinatewise C-stationary point of the original problem (2.41). When the original objective function f is coordinatewise regular at $\boldsymbol{z}$, the coordinatewise C-stationarity of f at $\boldsymbol{z}$ also implies the joint C-stationarity.

□

Furthermore, Proposition 4.3 can be coordinatewise applied to each convex approximation $\widetilde{f}_k\left(\boldsymbol{x}_k; \boldsymbol{x}^{(t)}\right)$ to identify a subset of coordinatewise C-stationary point of the original problem (2.41) that are not local minima. This leads to the additional termination condition incorporated in Algorithm 7.

Similar to the proposed smooth BSUM framework in Section 4.3.1, Algorithm 7 can be further extended to address the composite problem in (4.83) where the objective function contains an additional nonconvex nonsmooth regularization term that is separable across the blocks of variables. That is done by performing Algorithm 7 on the equivalent reformulation (4.85) of the problem (4.83). It leads to the following approximate problem at the tth iteration:

$$\widetilde{\boldsymbol{x}}_{\mathcal{I}^{(t)}} \in \left\{ \underset{\boldsymbol{x}_{\mathcal{I}^{(t)}}}{\operatorname{argmin}} \, \widetilde{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right) + \sum_{i \in \mathcal{I}^{(t)}} g_i(\boldsymbol{x}_i) \quad \text{s.t. } \boldsymbol{x}_i \in \mathcal{X}_i, \forall\, i \in \mathcal{I}^{(t)} \right\}. \tag{4.106}$$

That is, only a coordinatewise convex approximation for the component f is constructed and the separable regularization g remains unchanged. The approximate problem (4.106) can be decomposed into the subproblems

$$\widetilde{\boldsymbol{x}}_i \in \left\{ \underset{\boldsymbol{x}_i \in \mathcal{X}_i}{\operatorname{argmin}} \, \widetilde{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right) + g_i(\boldsymbol{x}_i) \right\} \quad \forall\, i \in \mathcal{I}^{(t)}, \tag{4.107}$$

and solved in parallel. Then the chosen block variables are updated along the descent direction according to the rule in (4.91) with a suitable step size $\gamma^{(t)}$ obtained by performing line search on the corresponding majorizing function in (4.84). Recall that the modified exact line search (3.27) and the successive line search with Armijo rule (3.31) and (3.32) should be used due to the presence of regularization. The exact line search in (3.27) customized for the majorizing function in (4.84) is expressed as the following optimization problem:

$$\gamma^{(t)} \in \underset{\gamma \in [0,1]}{\operatorname{argmin}} \, \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}} + \gamma\left(\widetilde{\boldsymbol{x}}^{(t)}_{\mathcal{I}^{(t)}} - \boldsymbol{x}^{(t)}_{\mathcal{I}^{(t)}}\right); \boldsymbol{x}^{(t)}\right) + \gamma \sum_{i \in \mathcal{I}^{(t)}} \left(g_i\left(\widetilde{\boldsymbol{x}}^{(t)}_i\right) - g_i\left(\boldsymbol{x}^{(t)}_i\right)\right). \tag{4.108}$$

Alternatively, in the Armijo rule, the step size is successively decreased at a geometric rate until the required sufficient decrease of function value is achieved, which is expressed as the following optimization problem:

$$\begin{aligned} \gamma^{(t)} = \underset{\gamma}{\operatorname{argmax}} \quad & \gamma \\ \text{s.t.} \quad & \gamma \in \{\beta^k \mid k \in \mathbb{N}\} \text{ and sufficient decrease constraint.} \end{aligned} \tag{4.109}$$

Either of the two sufficient decrease constraints given in (3.31) and (3.32), respectively, can be used, which are written as follows for the majorizing function in (4.84):

$$\begin{aligned} \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)} + \gamma\left(\widetilde{\boldsymbol{x}}_{\mathcal{I}^{(t)}}^{(t)} - \boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}\right); \boldsymbol{x}^{(t)}\right) - \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}; \boldsymbol{x}^{(t)}\right) \\ + \gamma \sum_{i \in \mathcal{I}^{(t)}} \left(g_i\left(\widetilde{\boldsymbol{x}}_i^{(t)}\right) - g_i\left(\boldsymbol{x}_i^{(t)}\right)\right) \leq \gamma\sigma\delta \end{aligned} \tag{4.110}$$

and

$$\begin{aligned} \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)} + \gamma\left(\widetilde{\boldsymbol{x}}_{\mathcal{I}^{(t)}}^{(t)} - \boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}\right); \boldsymbol{x}^{(t)}\right) - \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}; \boldsymbol{x}^{(t)}\right) \\ + \sum_{i \in \mathcal{I}^{(t)}} g_i\left(\boldsymbol{x}_i^{(t)} + \gamma\left(\widetilde{\boldsymbol{x}}_i^{(t)} - \boldsymbol{x}_i^{(t)}\right)\right) - g_i\left(\boldsymbol{x}_i^{(t)}\right) \leq \gamma\sigma\delta \end{aligned} \tag{4.111}$$

with $\delta = \sum_{i \in \mathcal{I}^{(t)}} \left(\nabla_{\boldsymbol{x}_i} \widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}^{(t)}\right)\right)^{\mathsf{T}} \left(\widetilde{\boldsymbol{x}}_i^{(t)} - \boldsymbol{x}_i^{(t)}\right) + g_i\left(\widetilde{\boldsymbol{x}}_i^{(t)}\right) - g_i\left(\boldsymbol{x}_i^{(t)}\right)$ and $0 < \beta, \sigma < 1$. With the above modifications, the proposed smoothing BSCA algorithm for the regularized problem (4.83) is then outlined in Algorithm 8. The convergence results of the proposed smoothing BSCA Algorithm 7 can be readily extended to Algorithm 8 by following the same line of analysis as in Section 4.3.1 based on the reformulation (4.85). The derivation is, therefore, not repeated. Every limit point of the solution sequence generated by Algorithm 8 is a coordinatewise C-stationary point of the reformulation (4.85) that satisfies the condition (4.86) and it is a C-stationary point satisfying the condition (4.87) if, additionally, f is coordinatewise regular.

Algorithm 8: The BSCA Algorithm Extended by Smoothing Majorization for Solving Problem (4.83)

Initialize $\boldsymbol{x}^{(0)} \in \mathcal{X}$ and $t \leftarrow 0$;

repeat

  Pick index set $\mathcal{I}^{(t)}$;

  Construct a smooth majorizer $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ of f at $\boldsymbol{x}^{(t)}$ in the selected coordinates $\mathcal{I}^{(t)}$;

  Construct coordinatewise convex approximations $\widetilde{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right)$ of $\widehat{f}_{\mathcal{I}^{(t)}}\left(\boldsymbol{x}_{\mathcal{I}^{(t)}}; \boldsymbol{x}^{(t)}\right)$ at $\boldsymbol{x}^{(t)}$ along the coordinates of each selected block $\boldsymbol{x}_i$, respectively, for all $i \in \mathcal{I}^{(t)}$;

  Let

$$\widetilde{\boldsymbol{x}}_i \in \left\{ \operatorname*{argmin}_{\boldsymbol{x}_i \in \mathcal{X}_i} \widetilde{f}_i\left(\boldsymbol{x}_i; \boldsymbol{x}^{(t)}\right) + g_i(\boldsymbol{x}_i) \right\} \quad \forall\, i \in \mathcal{I}^{(t)};$$

  Compute the step size $\gamma^{(t)}$ by the exact line search (4.108) or the successive line search (4.109);

  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)} + \gamma^{(t)}\left(\widetilde{\boldsymbol{x}}_i^{(t)} - \boldsymbol{x}_i^{(t)}\right)$ for all $i \in \mathcal{I}^{(t)}$;

  Let $\boldsymbol{x}_i^{(t+1)} = \boldsymbol{x}_i^{(t)}$ for all $i \notin \mathcal{I}^{(t)}$;

  $t \leftarrow t + 1$;

until *converged* **and** $\bigwedge_{i=1}^{K}\left(\boldsymbol{x}_i^{(t)} \notin \operatorname{int}\left(\mathcal{X}_i\right)\right.$ **or** $\left.\nabla_{\boldsymbol{x}_i} f\left(\boldsymbol{x}^{(t)}\right)\ \textit{exits}\right)$;

return $\boldsymbol{x}^{(t)}$

Chapter 5
Phase Retrieval with Dictionary Learning

Phase retrieval aims at recovering unknown signals from magnitude measurements of linear mixtures. In this chapter, we consider the phase retrieval with dictionary learning problem, which, apart from the measurements, includes other prior information that the signal admits a sparse representation over an unknown dictionary. The task is to jointly estimate the dictionary and the sparse representation from magnitude-only measurements. To this end, we study two complementary formulations and develop efficient parallel algorithms based on the SCA framework extended by smoothing majorization that is described in Section 4.2. The first algorithm is termed *compact-SCAphase* and is preferable in the case of moderately diverse mixture models with a low number of mixing components. It adopts a compact formulation that avoids auxiliary variables. The proposed algorithm is highly scalable and has reduced parameter tuning costs. The second algorithm, referred to as *SCAphase*, uses auxiliary variables and is favorable in the case of highly diverse mixture models. It also renders simple incorporation of additional side constraints. The performance of both methods is evaluated when applied to blind channel estimation from subband magnitude measurements in a multi-antenna random access network. Simulation results show the efficiency of the proposed techniques compared to state-of-the-art methods.

The key contributions presented in this chapter originate from [LTY+22]. This chapter is organized as follows. The phase retrieval and phase retrieval with dictionary learning are revised in Section 5.1. In Section 5.2, we introduce the signal model and provide two different mathematical formulations with and without auxiliary variables, respectively, for the phase retrieval with dictionary learning problem. The proposed algorithms for both formulations are described in Section 5.3 and 5.4, respectively. In Section 5.5, we analyze the computational complexity of the proposed algorithms in comparison to the state-of-the-art method SC-PRIME [QP17]. Simulation results on synthetic data are presented and discussed in Section 5.6, while Section 5.7 summarizes this chapter.

5.1 Motivation

Phase retrieval refers to the problem of recovering unknown signals from the (squared) magnitude of linear measurements corrupted by additive noise. It has received considerable attention in various applications such as diffraction imaging [CLS15a, SEC+15],

astronomy [Fie82,FD87], and X-ray crystallography [Mil90,Har93], where the measurements of intensities are much easier to obtain than that of the complex magnitudes and phases. In some other applications, including non-coherent direction-of-arrival estimation [KHE14], phase information may be available but imprecise, e.g., due imperfect phase synchronization and phase noise.

In recent years, numerous phase retrieval approaches have been developed, which can be principally classified as nonconvex and convex ones. In the nonconvex optimization methods, the recovery problem is formulated as a nonconvex least-squares (LS) problem. Stationary points of the nonconvex formulation can then be obtained by classic continuous optimization algorithms such as alternating projections [GS72,Fie82], gradient descent [CLS15b, WGE18, CCFM19], and alternating direction method of multipliers (ADMM) [WYLM12, LSJL17]. A popular class of convex optimization approaches employs semidefinite relaxation [CSV13, CESV15, WM15, JEH16], which lifts the problem to a higher dimension and is, hence, computationally prohibitive for large-scale problems. Recently, some non-lifting convex optimization approaches have been developed based on solving a basis pursuit problem in the dual domain, including PhaseMax [GS18] and PhaseEqual [WFDL20]. A comprehensive review of recent advances in phase retrieval from a numerical perspective is presented in [FS20].

On the other hand, additional prior information on the unknown signal, such as sparsity, can be used to improve the uniqueness and stability of the reconstruction [EHM16]. Most of the aforementioned phase retrieval approaches have been adapted to recovering signals that are sparse either in the standard basis or in a known dictionary [SBE14, ESM$^+$14, QP17, WZG$^+$18, PBES18, SAH18, YPEO19, WFDL20]. The GESPAR algorithm is based on the damped Gauss-Newton method [SBE14]. MM algorithms are devised in [QP17]. In [WZG$^+$18], the Truncated Amplitude Flow (TAF) method is extended to recovering sparse signals. The STELA algorithm proposed in [YPEO19] is based on SCA and can be parallelized.

Phase retrieval was generalized in [TEM16] to jointly learning an unknown dictionary and a sparse representation. To tackle the joint estimation problem, the authors propose a regularized nonconvex LS formulation with squared magnitude measurements and develop an alternating minimization algorithm termed DOLPHIn. In [QP17], the authors apply a similar regularized LS formulation to magnitude measurements and solve it by an algorithm that is based on BSUM, named SC-PRIME. There, it is shown by both theoretical justification and numerical results that the reconstruction from magnitude measurement outperforms that from intensity measurements. However, the use of auxiliary variables in both aforementioned methods depresses the scalability and, more notably, increases the number of hyperparameters that require tuning. Moreover,

neither of the two methods can take full benefit of modern parallel hardware architectures. In addition, SC-PRIME often suffers from slow convergence due to the loose approximation associated with the BSUM algorithm.

As shown in Section 3.2.2, the idea of SCA enables the construction of approximate problems that can be parallelized. Therefore, in this chapter, we employ the smoothing SCA framework in Section 4.2 to address the phase retrieval with dictionary learning problem given the magnitude measurements, which is formulated as a nonsmooth and nonconvex LS problem. Two efficient parallel algorithms are proposed by applying the smoothing SCA framework to two complementary formulations, respectively. Specifically, we first study a compact formulation that avoids the auxiliary variables, and the proposed algorithm based on the smoothing SCA is termed *compact-SCAphase*. Then another algorithm based on the smoothing SCA framework is developed for the conventional formulation with auxiliary variables, which is referred to as *SCAphase* (smoothing Successive Convex Approximation for phase retrieval with dictionary learning). The performance of the proposed algorithms is evaluated when applied to blind sparse channel estimation from subband magnitude measurements in a multi-antenna random access network. Simulation results on synthetic data show the fast convergence of the proposed algorithms compared to the state-of-the-art method SC-PRIME [QP17]. In the case with less diverse linear mixing models, compact-SCAphase is more competitive than SCAphase in terms of both computational complexity and parameter tuning cost. However, for highly diverse linear measurement operators, the computational complexity of compact-SCAphase dramatically grows, compared to SCAphase.

To summarize, the main contributions included in this chapter are:

- Proposing two efficient parallel algorithms for the phase retrieval with dictionary learning problem by applying the smoothing SCA framework to two complementary formulations, respectively.
- Refining the search range for suitable values of the sparsity parameter for both algorithms
- Proposing an efficient procedure based on rational approximation for solving the ℓ_2-norm constrained LS subproblems to reduce the overall computational complexity of compact-SCAphase.
- Analyzing theoretically the computational complexities of the proposed algorithms in comparison to the state-of-the-art method.

- Conducting extensive simulations in the context of blind channel estimation in a multi-antenna random access network in view of parameter selection, estimation quality, convergence speed, computational time, and robustness to initialization.

Notations: In this chapter, the soft-thresholding operator is denoted by

$$\mathcal{S}_\lambda(x) = \max\{0, |x| - \lambda\} \cdot \mathrm{e}^{\mathrm{j}\arg(x)}. \tag{5.1}$$

5.2 Problem Formulation

We consider the following nonlinear system. For an input signal $\boldsymbol{X} \in \mathbb{K}^{N\times I}$, $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$, the following noise-corrupted magnitude-only measurements are observed:

$$\boldsymbol{Y} = |\mathcal{F}(\boldsymbol{X})| + \boldsymbol{N}, \tag{5.2}$$

where $\mathcal{F} : \mathbb{C}^{N\times I} \to \mathbb{C}^{M_1\times M_2}$ is a linear operator, $\boldsymbol{N}$ is a noise matrix, and the absolute value operation $|\cdot|$ is applied elementwise. The negative entries of $\boldsymbol{Y}$ caused by noise will be set to 0. A general linear mixing operator $\mathcal{F}(\boldsymbol{X})$ can be written as

$$\mathcal{F}(\boldsymbol{X}) = \sum_{k=1}^{K} \boldsymbol{A}_k \boldsymbol{X} \boldsymbol{B}_k, \tag{5.3}$$

where $\boldsymbol{A}_k \in \mathbb{C}^{M_1\times N}$ and $\boldsymbol{B}_k \in \mathbb{C}^{I\times M_2}$, $k = 1, \ldots, K$, perform the row and column mixing, respectively, and the number of distinct mixing components K is termed as the diversity of the mixing operator $\mathcal{F}$ in this thesis. Note that the linear operator $\mathcal{F}$ in (5.3) can be written equivalently in a vectorized form

$$\mathrm{vec}\left(\mathcal{F}(\boldsymbol{X})\right) = \boldsymbol{F} \cdot \mathrm{vec}(\boldsymbol{X}) \quad \text{with } \boldsymbol{F} = \sum_{k=1}^{K} \boldsymbol{B}_k^{\mathsf{T}} \otimes \boldsymbol{A}_k, \tag{5.4}$$

which we will also use in this thesis. Moreover, each column $\boldsymbol{x}_i$ of $\boldsymbol{X}$ is assumed to admit a sparse representation over an unknown dictionary $\boldsymbol{D} \in \mathbb{K}^{N\times P}$, i.e., $\boldsymbol{x}_i = \boldsymbol{D}\boldsymbol{z}_i$ with a sparse code vector $\boldsymbol{z}_i \in \mathbb{K}^P$. Let $\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_I]$ summarize the code vectors. Our objective is to jointly learn the dictionary $\boldsymbol{D}$ and the sparse codes $\boldsymbol{Z}$ so as to minimize the (LS) reconstruction error.

To this end, we solve the following compact formulation for the phase retrieval with dictionary learning (cPRDL) problem:

$$\text{cPRDL:} \quad \min_{\boldsymbol{D}\in\mathcal{D},\,\boldsymbol{Z}} \quad \frac{1}{2}\|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{D}\boldsymbol{Z})|\|_{\mathsf{F}}^2 + \lambda\|\boldsymbol{Z}\|_{1,1}. \tag{5.5}$$

The first term evaluates the data fidelity by the LS criterion, which is nonsmooth and nonconvex due to the absolute value operation and the bilinear term $\boldsymbol{DZ}$. The second term promotes sparsity in $\boldsymbol{Z}$ with a regularization parameter $\lambda \geq 0$. To avoid scaling ambiguities in the bilinear term $\boldsymbol{DZ}$, we restrict $\boldsymbol{D}$ to be in the convex set

$$\mathcal{D} = \{\boldsymbol{D} \in \mathbb{K}^{N \times P} \mid \|\boldsymbol{d}_p\|_2 \leq 1 \ \forall p = 1, \ldots, P\}.$$

Each column $\boldsymbol{d}_p$ is called an atom and the dictionary size must be below the number of columns in $\boldsymbol{X}$, i.e., $P < I$. Otherwise, each column $\boldsymbol{x}_i$ can be trivially represented by a 1-sparse vector $\boldsymbol{z}_i$ with an atom $\boldsymbol{x}_i / \|\boldsymbol{x}_i\|_2$.

An alternative formulation for phase retrieval with dictionary learning (PRDL), which we will also consider, is constructed as follows with an auxiliary variable $\boldsymbol{X}$:

$$\text{PRDL:} \quad \min_{\boldsymbol{X},\, \boldsymbol{D} \in \mathcal{D},\, \boldsymbol{Z}} \quad \frac{1}{2}\|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{X})|\|_{\mathsf{F}}^2 + \frac{\mu}{2}\|\boldsymbol{X} - \boldsymbol{DZ}\|_{\mathsf{F}}^2 + \rho\|\boldsymbol{Z}\|_{1,1}. \tag{5.6}$$

The additional second term measures how well the signal $\boldsymbol{X}$ can be approximated by the sparse representation $\boldsymbol{DZ}$. Two regularization parameters $\mu \geq 0$ and $\rho \geq 0$ are used to balance the data fidelity, the approximation quality, and the code sparsity.

The formulation (5.6) was first proposed in [TEM16], however, with the intensity measurements

$$\widetilde{\boldsymbol{Y}} = |\mathcal{F}(\boldsymbol{X})|^2 + \boldsymbol{N},$$

which results in another smooth data fidelity term $\frac{1}{4}\|\widetilde{\boldsymbol{Y}} - |\mathcal{F}(\boldsymbol{X})|^2\|_{\mathsf{F}}^2$. In [QP17], the authors have shown that, for the intensity measurements $\widetilde{\boldsymbol{Y}}$, it is also beneficial, in the high SNR regime, to use formulation (5.6) with the modulus information $\sqrt{\widetilde{\boldsymbol{Y}}}$, where $\sqrt{\cdot}$ is applied elementwise, due to the reduced noise level in $\sqrt{\widetilde{\boldsymbol{Y}}}$. Thus, we consider the magnitude measurement model (5.2).

In [QP17], the state-of-the-art SC-PRIME algorithm is devised for the conventional formulation (5.6) based on BSUM. (In fact, SC-PRIME belongs to the smoothing BSUM framework that we established in Section 4.3.1 since it fulfills only the relaxed subgradient consistency, not the stricter derivative consistency required by the classic BSUM framework.) However, as BSUM strictly requires the approximate function to be a global upper bound, SC-PRIME lacks the flexibility to take full advantage of modern parallel hardware architectures. Also, the conservative majorization in SC-PRIME often results in slow convergence. Therefore, we develop the *compact-SCAphase* and *SCAphase* algorithms for the compact formulation (5.5) and conventional formulation (5.6), respectively, based on the smoothing SCA framework developed in Section 4.2. Both proposed algorithms can be easily parallelized, e.g., on modern multicore DSPs, GPUs and FPGAs as well as in cloud computing networks.

The two proposed algorithms are advantageous in different scenarios. The conventional formulation (5.6) is not suitable for large-scale problems due to the introduction of auxiliary variables. Also, the complexity of tuning two regularization parameters μ and ρ in (5.6) is significantly higher than that of tuning one parameter. However, compared to SCAphase, the computational complexity of compact-SCAphase grows dramatically with the increase of diversity of the designed linear measurement operator $\mathcal{F}$. Moreover, the conventional formulation (5.6) admits simple incorporation of additional prior information on $\boldsymbol{X}$ such as nonnegativity in radio astronomic imaging [Fie82, FD87] and X-ray crystallography [Mil90, Har93].

As demonstrated by the smoothing BSCA framework in Section 4.3.2, the proposed compact-SCAphase and SCAphase algorithms can certainly be implemented in a BCD manner so that the per-iteration complexity can be customized according to the computational and memory capacity of the hardware. Nevertheless, since we investigate by the simulations only the performance of the algorithms in the case where all variables are jointly updated in each iteration, in the following sections, we describe the proposed compact-SCAphase and SCAphase algorithms in a joint-update version. The derivations are based on the model with complex-valued variables, which can be readily extended to the real-valued case. The cPRDL and PRDL problems are not guaranteed to be coordinatewise regular everywhere due to the nonsmoothness. Hence, a block-coordinatewise implementation of the proposed algorithms may only converge to a coordinatwise stationary point as analyzed in Section 4.3.2.

5.3 Proposed Algorithm for Formulation cPRDL

In this section, we apply the smoothing SCA framework proposed in Section 4.2 on the compact formulation (5.5) to derive an efficient iterative algorithm that finds a stationary point of (5.5) via a sequence of separable approximate problems. We denote the objective function in (5.5) by $h(\boldsymbol{D}, \boldsymbol{Z}) = f(\boldsymbol{D}, \boldsymbol{Z}) + g(\boldsymbol{Z})$ with

$$f(\boldsymbol{D}, \boldsymbol{Z}) = \frac{1}{2}\|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{D}\boldsymbol{Z})|\|_{\mathsf{F}}^2 \quad \text{and} \quad g(\boldsymbol{Z}) = \lambda\|\boldsymbol{Z}\|_{1,1}. \tag{5.7}$$

The problem is challenging since the regularization g is nonsmooth and, more notably, f is nonsmooth and nonconvex. To overcome this difficulty, in each iteration, we first majorize f by a smooth function, which is then minimized approximately by solving a separable convex approximate problem. In particular, we obtain a descent direction of the majorizing function by minimizing exactly its convex approximation. The variable can then be updated along this descent direction with a suitable step size, which can

be efficiently obtained by exact line search. Consequently, a decrease of the original objective function h is also ensured.

Once a stationary point $(\boldsymbol{D}^\star, \boldsymbol{Z}^\star)$ of the cPRDL problem in (5.5) has been obtained by the compact-SCAphase algorithm, we optionally perform a debiasing step similar to that in [FNW07] to further improve the estimation quality, which solves an instance of the cPRDL problem with $\lambda = 0$ and a restriction that the entries $z_{p,i}$ having zero values in $\boldsymbol{Z}^\star$ are fixed at zero.

5.3.1 Smooth Majorization

We first derive a smooth majorizing function for f in (5.7) by the same majorization technique performed on the simple phase retrieval problem in Section 4.1.2. Let $\boldsymbol{S} = (\boldsymbol{D}, \boldsymbol{Z})$ denote the collection of all variables, and let $\boldsymbol{S}^{(t)} = (\boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$ be the current point at iteration t. The function f can be expanded as

$$f(\boldsymbol{S}) = \frac{1}{2}(\|\boldsymbol{Y}\|_{\mathsf{F}}^2 + \|\mathcal{F}(\boldsymbol{D}\boldsymbol{Z})\|_{\mathsf{F}}^2) \underbrace{- \operatorname{tr}(\boldsymbol{Y}^{\mathsf{T}} |\mathcal{F}(\boldsymbol{D}\boldsymbol{Z})|)}_{v(\boldsymbol{S})},$$

where the cross term $v(\boldsymbol{S})$ is nonsmooth due to the absolute value operation. As $\boldsymbol{Y}$ contains nonnegative entries, the nonsmooth cross term $v(\boldsymbol{S})$ can be written as the following pointwise minimum:

$$v(\boldsymbol{S}) = \min_{\boldsymbol{\Theta} \in [0, 2\pi)^{M_1 \times M_2}} -\Re\left(\operatorname{tr}\left(\left(\boldsymbol{Y} \odot \mathrm{e}^{\mathsf{j}\boldsymbol{\Theta}}\right)^{\mathsf{H}} \mathcal{F}(\boldsymbol{D}\boldsymbol{Z})\right)\right), \tag{5.8}$$

and the minimum solution at $\boldsymbol{S} = \boldsymbol{S}^{(t)}$ is $\breve{\boldsymbol{\Theta}} = \arg\left(\mathcal{F}\left(\boldsymbol{D}^{(t)}\boldsymbol{Z}^{(t)}\right)\right)$, where $\mathrm{e}^{(\cdot)}$ and $\arg(\cdot)$ are applied elementwise. Thus, defining

$$\boldsymbol{Y}^{(t)} = \boldsymbol{Y} \odot \mathrm{e}^{\mathsf{j}\arg(\mathcal{F}(\boldsymbol{D}^{(t)}\boldsymbol{Z}^{(t)}))},$$

we can construct the following smooth majorizing function for f at $\boldsymbol{S}^{(t)}$:

$$\begin{aligned}\widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) &= \frac{1}{2}\left(\|\boldsymbol{Y}\|_{\mathsf{F}}^2 + \|\mathcal{F}(\boldsymbol{D}\boldsymbol{Z})\|_{\mathsf{F}}^2\right) - \Re\left(\operatorname{tr}\left(\left(\boldsymbol{Y}^{(t)}\right)^{\mathsf{H}} \mathcal{F}(\boldsymbol{D}\boldsymbol{Z})\right)\right) \\ &= \frac{1}{2}\left\|\boldsymbol{Y}^{(t)} - \mathcal{F}(\boldsymbol{D}\boldsymbol{Z})\right\|_{\mathsf{F}}^2,\end{aligned} \tag{5.9}$$

which, as proved in Section 4.1.2, satisfies Assumption 4.1, in particular, the following subgradient consistency at $\boldsymbol{S}^{(t)}$:

$$\nabla \widehat{f}\left(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)}\right) \in \partial^C f\left(\boldsymbol{S}^{(t)}\right). \tag{5.10}$$

The smooth majorizing function $\widehat{f}$ has the partial gradients

$$\begin{cases} \nabla_{\boldsymbol{D}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \mathcal{F}^*\big(\mathcal{F}(\boldsymbol{DZ}) - \boldsymbol{Y}^{(t)}\big) \cdot \boldsymbol{Z}^{\mathsf{H}}, \\ \nabla_{\boldsymbol{Z}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \boldsymbol{D}^{\mathsf{H}} \cdot \mathcal{F}^*\big(\mathcal{F}(\boldsymbol{DZ}) - \boldsymbol{Y}^{(t)}\big), \end{cases} \tag{5.11}$$

where $\mathcal{F}^*(\cdot)$ is the adjoint of the linear operator $\mathcal{F}$. However, $\widehat{f}$ is nonconvex due to the bilinear map $\boldsymbol{DZ}$. Then function $\widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) + g(\boldsymbol{Z})$ is a majorizing function of the objective function h at $\boldsymbol{S}^{(t)}$.

5.3.2 Separable Convex Approximation

As described in Section 4.2, instead of minimizing the majorizing function $\widehat{f}$ exactly, we can employ the idea of SCA to obtain an approximate minimizer at a significantly lower computational cost, due to the smoothness of $\widehat{f}$. In particular, the approximation techniques introduced in Section 3.2.2 are used to construct a separable convex approximation of $\widehat{f}$ that can be minimized in parallel.

As the regularization g is convex and separable, we leave g unaltered and only design a separable convex approximation for $\widehat{f}$ at the current point $\boldsymbol{S}^{(t)}$. As $\widehat{f}$ is partially convex in $\boldsymbol{D}$ and $\boldsymbol{Z}$, respectively, we adopt the Jacobi-type approximation (3.36), where the approximate function is the sum of several components. In each component, only part of the variables are varied while the rest are fixed to their current values. Let $\widetilde{f}_D(\boldsymbol{D}; \boldsymbol{S}^{(t)})$ and $\widetilde{f}_Z(\boldsymbol{Z}; \boldsymbol{S}^{(t)})$ be the approximate functions of $\widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})$ over $\boldsymbol{D}$ and $\boldsymbol{Z}$, respectively. They are devised as

$$\begin{aligned} \widetilde{f}_D(\boldsymbol{D}; \boldsymbol{S}^{(t)}) &= \sum_{p=1}^{P} \widehat{f}\left(\boldsymbol{d}_p, \boldsymbol{D}_{-p}^{(t)}, \boldsymbol{Z}^{(t)}; \boldsymbol{S}^{(t)}\right), \\ \widetilde{f}_Z(\boldsymbol{Z}; \boldsymbol{S}^{(t)}) &= \sum_{i=1}^{I} \sum_{p=1}^{P} \widehat{f}\left(z_{p,i}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}_{-(p,i)}^{(t)}; \boldsymbol{S}^{(t)}\right), \end{aligned} \tag{5.12}$$

where $\boldsymbol{D}_{-p} \in \mathbb{C}^{N \times (P-1)}$ is obtained by removing $\boldsymbol{d}_p$ from $\boldsymbol{D}$ and $\boldsymbol{Z}_{-(p,i)}$ the collection of all entries of $\boldsymbol{Z}$ except $z_{p,i}$. Then the convex approximation of $\widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)})$ is

$$\widetilde{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \widetilde{f}_D(\boldsymbol{D}; \boldsymbol{S}^{(t)}) + \widetilde{f}_Z(\boldsymbol{Z}; \boldsymbol{S}^{(t)}) + \lambda \|\boldsymbol{Z}\|_{1,1}$$

and the approximate problem reads

$$(\widetilde{\boldsymbol{D}}^{(t)}, \widetilde{\boldsymbol{Z}}^{(t)}) = \underset{\boldsymbol{D} \in \mathcal{D},\, \boldsymbol{Z}}{\operatorname{argmin}} \quad \widetilde{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}). \tag{5.13}$$

The columns of $\boldsymbol{D}$ and all the entries of $\boldsymbol{Z}$ are separable in the objective function of (5.13) and the constraint set $\mathcal{D}$ is a Cartesian product of compact convex sets, each

of which involves one column $\boldsymbol{d}_p$. Consequently, problem (5.13) can be decomposed into $P + (P \times I)$ subproblems. Each subproblem exclusively depends on a column $\boldsymbol{d}_p$ or a single variable $z_{p,i}$ and, hence, can be solved in parallel.

Define $\Delta\boldsymbol{D} = \widetilde{\boldsymbol{D}}^{(t)} - \boldsymbol{D}^{(t)}$ and $\Delta\boldsymbol{Z} = \widetilde{\boldsymbol{Z}}^{(t)} - \boldsymbol{Z}^{(t)}$. According to Proposition 4.5, the difference $(\Delta\boldsymbol{D}, \Delta\boldsymbol{Z})$ is a descent direction of the majorizing function $\widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)})$ in the domain of (5.5). Thus, the following simultaneous update rule can be applied:

$$\boldsymbol{D}^{(t+1)} = \boldsymbol{D}^{(t)} + \gamma^{(t)}\Delta\boldsymbol{D} \quad \text{and} \quad \boldsymbol{Z}^{(t+1)} = \boldsymbol{Z}^{(t)} + \gamma^{(t)}\Delta\boldsymbol{Z}, \tag{5.14}$$

where $\gamma^{(t)} \in (0, 1]$ is the step size. When $(\widetilde{\boldsymbol{D}}^{(t)}, \widetilde{\boldsymbol{Z}}^{(t)}) = (\boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$, a stationary point, in fact, a global minimizer, of $\widetilde{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)})$ is achieved, which is also stationary for the majorizing problem and the original problem (5.5) in the Clarke sense according to Theorem 4.2.

In the following, we describe the efficient solution approaches for the subproblems decomposed from (5.13).

Descent direction for $\boldsymbol{D}$

The P independent subproblems decomposed from problem (5.13) involving $\boldsymbol{D}$ can be written as

$$\min_{\boldsymbol{d}_p} \ \widehat{f}\left(\boldsymbol{d}_p, \boldsymbol{D}_{-p}^{(t)}, \boldsymbol{Z}^{(t)}; \boldsymbol{S}^{(t)}\right) \quad \text{s.t.} \ \frac{1}{2}\left(\|\boldsymbol{d}_p\|_2^2 - 1\right) \leq 0. \tag{5.15}$$

Each subproblem in (5.15) is an ℓ_2-norm constrained LS, which has no closed-form solution. However, as Slater's condition is satisfied for (5.15), strong duality holds and, hence, the primal and dual optimal solutions can be obtained by solving the Karush-Kuhn-Tucker (KKT) optimality system [BV04, Sec. 5.5.3]. By vectorization, we express $\widehat{f}(\boldsymbol{d}_p, \boldsymbol{D}_{-p}^{(t)}, \boldsymbol{Z}^{(t)}; \boldsymbol{S}^{(t)})$ as

$$\widehat{f}\left(\boldsymbol{d}_p, \boldsymbol{D}_{-p}^{(t)}, \boldsymbol{Z}^{(t)}; \boldsymbol{S}^{(t)}\right) = \frac{1}{2}\left\|\mathrm{vec}\left(\boldsymbol{Y}_p^{(t)}\right) - \boldsymbol{H}_p\boldsymbol{d}_p\right\|_2^2, \tag{5.16}$$

where $\boldsymbol{Y}_p^{(t)} = \boldsymbol{Y}^{(t)} - \mathcal{F}\left(\boldsymbol{D}_{-p}^{(t)}\boldsymbol{Z}_{-p}^{(t)}\right)$ with $\boldsymbol{Z}_{-p} \in \mathbb{C}^{(P-1)\times I}$ obtained by removing the pth row of $\boldsymbol{Z}$, and $\boldsymbol{H}_p = \boldsymbol{F} \cdot \left(\boldsymbol{z}_{p:}^{(t)} \otimes \boldsymbol{I}_N\right)$ with $\boldsymbol{F}$ in (5.4). Then the Lagrangian associated with (5.15) is

$$L(\boldsymbol{d}_p, \nu_p) = \frac{1}{2}\|\mathrm{vec}(\boldsymbol{Y}_p^{(t)}) - \boldsymbol{H}_p\boldsymbol{d}_p\|_{\mathsf{F}}^2 + \frac{\nu_p}{2}(\|\boldsymbol{d}_p\|_2^2 - 1), \tag{5.17}$$

where $\nu_p \geq 0$ is a Lagrangian multiplier. Let $\widetilde{\boldsymbol{d}}_p^{(t)}$ and $\widetilde{\nu}_p^{(t)}$ be a pair of primal and dual optimal solutions, and let $\boldsymbol{H}_p = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$ be the compact singular value decomposition (SVD) of $\boldsymbol{H}_p$ and $\sigma_1 \geq \cdots \geq \sigma_r > 0$ the nonzero singular values with

$r = \operatorname{rank}(\boldsymbol{H}_p)$, $\boldsymbol{U} \in \mathbb{C}^{M_1 M_2 \times r}$, $\boldsymbol{\Sigma} \in \mathbb{C}^{r \times r}$, and $\boldsymbol{V} \in \mathbb{C}^{N \times r}$. The solution $\widetilde{\boldsymbol{d}}_p^{(t)}$ of problem (5.15) holds

$$\widetilde{\boldsymbol{d}}_p^{(t)} = \boldsymbol{V}\big(\boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{\Sigma} + \widetilde{\nu}_p^{(t)}\boldsymbol{I}_r\big)^{\dagger}\boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{U}^{\mathsf{H}}\operatorname{vec}(\boldsymbol{Y}_p^{(t)}) \tag{5.18}$$

by solving the KKT system. Define the rational function

$$\psi_p(\nu_p) = \sum_{i=1}^{r} \frac{|c_{i,p}|^2}{(\sigma_i^2 + \nu_p)^2} \quad \text{with } \boldsymbol{c}_p = \boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{U}^{\mathsf{H}}\operatorname{vec}(\boldsymbol{Y}_p^{(t)}). \tag{5.19}$$

The dual optimal point $\widetilde{\nu}_p^{(t)}$ required in (5.18) is determined by

$$\begin{cases} \widetilde{\nu}_p^{(t)} = 0, & \text{if } \psi_p(0) \le 1, \\ \widetilde{\nu}_p^{(t)} \in \{\nu_p > 0 \mid \psi_p(\nu_p) = 1\}, & \text{otherwise.} \end{cases} \tag{5.20}$$

In the case where $\psi_p(0) > 1$, $\widetilde{\nu}_p^{(t)}$ is the unique solution of

$$\psi_p(\nu_p) = 1 \quad \text{for} \quad \nu_p \in (0, +\infty), \tag{5.21}$$

which has no closed-form expression, except for the case where all singular values σ_i are identical. In the general case, to solve (5.21), we develop an efficient iterative algorithm based on successive rational approximation (cf. [BNS78, Li93]), which is outlined in Algorithm 10 and will be described in Section 5.3.4. The intermediate derivations of the primal and dual solutions in (5.18)-(5.20) from the KKT system are included in Appendix A.1.1.

For the particular case with the linear operator $\mathcal{F}$ in (5.47) that is investigated in the simulations, the SVD of $\boldsymbol{H}_p$ can be calculated analytically given the SVD of $\boldsymbol{A}$. Hence, the complexity is significantly reduced compared to the general case where an iterative algorithm, e.g., QR algorithm [GVL13], is needed to obtain the SVD of $\boldsymbol{H}_p$ for every column $\boldsymbol{d}_p$ in each iteration. Then the proposed SCA algorithm for the cPRDL problem in (5.5) is competitive with that for the PRDL problem in (5.6) in terms of complexity. Details on the simplified solution approach for $\mathcal{F}$ in (5.47) can be found in Appendix A.1.2.

Descent direction for $\boldsymbol{Z}$

The subproblem decomposed from (5.13) involving each entry $z_{p,i}$ is a univariate LASSO [Tib96] in Lagrangian form, which admits a closed-form solution

$$\widetilde{z}_{p,i}^{(t)} = \frac{1}{\left\|\boldsymbol{F}_i\boldsymbol{d}_p^{(t)}\right\|_2^2}\mathcal{S}_{\lambda}\left(\left\|\boldsymbol{F}_i\boldsymbol{d}_p^{(t)}\right\|_2^2 z_{p,i}^{(t)} - \nabla_{z_{p,i}}\widehat{f}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)})\right) \tag{5.22}$$

Algorithm 9: compact-SCAphase

Input: $\boldsymbol{Y} \in \mathbb{R}_+^{M_1 \times M_2}$, $\lambda \geq 0$, tolerance $\varepsilon > 0$

Initialize $\boldsymbol{D}^{(0)} \in \mathcal{D}$ and $\boldsymbol{Z}^{(0)}$ randomly, $t \leftarrow 0$;
repeat
  for $p = 1, \ldots, P$ **do** in parallel
    $\boldsymbol{H}_p \leftarrow \boldsymbol{F} \cdot \left(\boldsymbol{z}_{p:}^{(t)} \otimes \boldsymbol{I}_N\right)$;
    Compute the compact SVD of $\boldsymbol{H}_p$;
    Compute dual optimal value $\tilde{\nu}_p^{(t)}$ using (5.20);
    Compute $\widetilde{\boldsymbol{d}}_p^{(t)}$ according to (5.18);
  end
  for $p = 1, \ldots, P,\ i = 1, \ldots, I$ **do** in parallel
    Compute $\widetilde{z}_{p,i}^{(t)}$ according to (5.22);
  end
  Compute step size $\gamma^{(t)}$ by exact line search (5.24);
  Update the variables using (5.14) and $t \leftarrow t + 1$;
until *stopping criterion* (5.32) *achieved* **and** *(* $\boldsymbol{D}^{(t)} \notin \operatorname{int}(\mathcal{D})$ **or** $\nabla f(\boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$ *exists)*;
return $\boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)}$

with the soft thresholding operator $\mathcal{S}_\lambda$ defined in (5.1). Matrix $\boldsymbol{F}_i$ in (5.22) is the ith block of $\boldsymbol{F}$ in the partition

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1 & \cdots & \boldsymbol{F}_I \end{bmatrix} \quad \text{with} \quad \boldsymbol{F}_i \in \mathbb{C}^{M_1 M_2 \times N} \text{ for } i = 1, \ldots, I. \tag{5.23}$$

5.3.3 Step Size Computation

As introduced in Section 4.2, a suitable step size $\gamma^{(t)}$ for the update in (5.14) that ensures a sufficient decrease of the original objective function can be efficiently obtained by performing the exact or successive line search methods on the the majorizing function. In particular, considering that the problem (5.5) contains a convex nonsmooth regularization g, the modified exact line search in (4.65), where the convex nonsmooth regularization g restricted to the descent direction is replaced by its linear majorization at the current iterate, is employed. The exact line search (4.65) customized for the majorizing function $\widehat{f}$ in (5.9) can be expressed as

$$\begin{aligned} \gamma^{(t)} &= \operatorname*{argmin}_{\gamma \in [0,1]} \left\{ \widehat{f}\big(\boldsymbol{D}^{(t)} + \gamma \Delta \boldsymbol{D}, \boldsymbol{Z}^{(t)} + \gamma \Delta \boldsymbol{Z}; \boldsymbol{S}^{(t)}\big) + \gamma\big(g(\widetilde{\boldsymbol{Z}}^{(t)}) - g(\boldsymbol{Z}^{(t)})\big) \right\}, \\ &= \operatorname*{argmin}_{\gamma \in [0,1]} \left\{ \frac{1}{4} w_4 \gamma^4 + \frac{1}{3} w_3 \gamma^3 + \frac{1}{2} w_2 \gamma^2 + w_1 \gamma \right\}, \end{aligned} \tag{5.24}$$

where

$$
\begin{aligned}
w_4 &= 2\|\boldsymbol{M}_2\|_{\mathsf{F}}^2, \\
w_3 &= 3\Re\left(\mathrm{tr}\left(\boldsymbol{M}_2^{\mathsf{H}}\boldsymbol{M}_1\right)\right), \\
w_2 &= 2\Re\left(\mathrm{tr}\left(\boldsymbol{M}_2^{\mathsf{H}}\boldsymbol{M}_0\right)\right) + \|\boldsymbol{M}_1\|_{\mathsf{F}}^2, \\
w_1 &= \Re\left(\mathrm{tr}\left(\boldsymbol{M}_1^{\mathsf{H}}\boldsymbol{M}_0\right)\right) + \lambda\left(\|\widetilde{\boldsymbol{Z}}^{(t)}\|_1 - \|\boldsymbol{Z}^{(t)}\|_1\right),
\end{aligned}
$$

and

$$
\begin{aligned}
\boldsymbol{M}_2 &= \mathcal{F}(\Delta\boldsymbol{D}\Delta\boldsymbol{Z}), \\
\boldsymbol{M}_1 &= \mathcal{F}(\Delta\boldsymbol{D}\boldsymbol{Z}^{(t)} + \boldsymbol{D}^{(t)}\Delta\boldsymbol{Z}), \\
\boldsymbol{M}_0 &= \mathcal{F}(\boldsymbol{D}^{(t)}\boldsymbol{Z}^{(t)}) - \boldsymbol{Y}^{(t)}.
\end{aligned}
$$

Problem (5.24) corresponds to minimizing a fourth-order polynomial on the interval $[0,1]$ and can be solved by computing its stationary points, i.e., the real roots of its derivative, a cubic polynomial, in $[0,1]$. The analytical expressions of all three roots of a third-order polynomial in the complex domain are given by the following well-known cubic formula:

$$
\hat{\gamma}_k = -\frac{1}{3w_4}\left(w_3 + \xi^k C + \frac{\Sigma_0}{\xi^k C}\right) \quad \text{for } k = 0,1,2, \tag{5.25}
$$

where

$$
\begin{aligned}
\Sigma_0 &= w_3^2 - 3w_4w_2, \\
\Sigma_1 &= 2w_3^3 - 9w_4w_3w_2 + 27w_4^2w_1, \\
C &= \sqrt[3]{\frac{\Sigma_1 + \sqrt{\Sigma_1^2 - 4\Sigma_0^3}}{2}}, \\
\xi &= \frac{-1 + j\sqrt{3}}{2}.
\end{aligned}
$$

Among them, only an even number of roots can be non-real. Specifically, if $\Sigma_1^2 - 4\Sigma_0^3 \geq 0$, there is only one real root or three identical real roots, which corresponds to $\hat{\gamma}_0$. Then the step size $\gamma^{(t)}$ is the projection of $\hat{\gamma}_0$ onto $[0,1]$, i.e., $\gamma^{(t)} = \mathcal{P}_{[0,1]}(\hat{\gamma}_0) = \min\left\{\max\left\{\hat{\gamma}, 0\right\}, 1\right\}$. Otherwise, if $\Sigma_1^2 - 4\Sigma_0^3 < 0$, all three roots are real. Let $\mathcal{C}$ denote the set of real roots in the interval $[0,1]$, i.e., $\mathcal{C} = \{\hat{\gamma}_k \mid 0 \leq \hat{\gamma}_k \leq 1, k = 0,1,2\}$. Then the solution $\gamma^{(t)}$ of the line search problem (5.24) lies in the set $\mathcal{C} \cup \{1\}$. If set $\mathcal{C}$ contains more than one element, which implies that the line search function is not unimodal in $[0,1]$, then the evaluation of the objective function of problem (5.24) at the points in $\mathcal{C} \cup \{1\}$ is required to obtain the step size $\gamma^{(t)}$.

As demonstrated by the smoothing BSCA framework in Section 4.3.2, this proposed algorithm can alternatively be implemented in a BCD manner. Compared to the

joint-update case, when only one block variable, $\boldsymbol{D}$ or $\boldsymbol{Z}$, is selected to be updated at iteration t, the line search problem (5.24) reduces to a convex quadratic program, which has a closed-form solution

$$\gamma^{(t)} = \mathcal{P}_{[0,1]}\left(-w_1/w_2\right).$$

The line search (5.24) always finds a nonzero step size $\gamma^{(t)}$ since $(\Delta\boldsymbol{D}, \Delta\boldsymbol{Z})$ is a descent direction of $\widehat{h}$, until a stationary point of h is attained. With the step size $\gamma^{(t)}$ obtained by the line search (5.24), the update (5.14) then ensures a monotonic decrease of the original objective function h in (5.5).

Finally, the proposed compact-SCAphase algorithm for solving the cPRDL problem in (5.5) is outlined in Algorithm 9.

5.3.4 Rational Approximation

Borrowing the idea in [BNS78, Li93], we develop a successive rational approximation algorithm, outlined in Algorithm 10, for efficiently solving the rational equation (5.21), which yields the dual optimal solution of (5.15). We omit the column index p in the derivations below as we discuss only one column.

Let $\nu^{(l)}$ be the approximate solution at iteration l. As $\psi(\nu)$ has all negative poles, it decreases monotonically in $[0, +\infty)$. Hence, we interpolate $\psi(\nu)$ at $\nu^{(l)}$ by a simple rational function

$$F(\nu; \alpha, \beta) = \frac{\alpha}{(\beta - \nu)^2}, \tag{5.26}$$

where parameters α and β are chosen such that $F(\nu^{(l)}; \alpha, \beta) = \psi(\nu^{(l)})$ and $F'(\nu^{(l)}; \alpha, \beta) = \psi'(\nu^{(l)})$. It is easily verified that

$$\alpha = \frac{4\big(\psi(\nu^{(l)})\big)^3}{\big(\psi'(\nu^{(l)})\big)^2} \quad \text{and} \quad \beta = \nu^{(l)} + \frac{2\psi(\nu^{(l)})}{\psi'(\nu^{(l)})}. \tag{5.27}$$

Algorithm 10: Rational Approximation for Solving (5.21).

Input: Rational function $\psi(\nu)$, tolerance $\eta > 0$

Initialize $\nu^{(0)} \leftarrow 0,\ l \leftarrow 0$;

repeat

  $\nu^{(l+1)} = \nu^{(l)} + \frac{2\psi(\nu^{(l)})}{\psi'(\nu^{(l)})}\left(1 - \sqrt{\psi(\nu^{(l)})}\right)$;

  $l \leftarrow l + 1$;

until $\psi(\nu^{(l)}) \leq 1 + \eta$;

return $\nu^{(l)}$

Then the unique solution of $F(\nu;\alpha,\beta) = 1$ in $(0,+\infty)$ is chosen as the next iterate $\nu^{(l+1)}$. Omitting intermediate calculations, we can express the update rule at the lth iteration as

$$\nu^{(l+1)} = \nu^{(l)} + \frac{2\psi(\nu^{(l)})}{\psi'(\nu^{(l)})}\left(1 - \sqrt{\psi(\nu^{(l)})}\right). \tag{5.28}$$

Define $\delta_i = -\sigma_i^2$, $i = 1,\ldots,r$, which are the poles of ψ with $\delta_1 \leq \ldots \leq \delta_r < 0$. Ignoring the trivial case where all poles δ_i are identical, we derive the following bounding property.

Theorem 5.1. *$F(\nu;\alpha,\beta) < \psi(\nu)$ for all $\nu > \delta_r$ and $\nu \neq \nu^{(l)}$.*

Proof: See Appendix A.2.

Thus, if $\psi(\nu^{(l)}) > 1$, i.e., $\nu^{(l)}$ is below the solution $\widetilde{\nu}$ of equation $\psi(\nu) = 1$, then the solution of $F(\nu;\alpha,\beta) = 1$ falls between $\nu^{(l)}$ and $\widetilde{\nu}$, i.e., $\nu^{(l)} < \nu^{(l+1)} < \widetilde{\nu}$. Hence, using the proposed rational approximation, we monotonically approach $\widetilde{\nu}$ from an initial point $\nu^{(0)} < \widetilde{\nu}$. Moreover, as we solve the rational equation in the case where $\psi(0) > 1$, ν can be simply initialized as $\nu^{(0)} = 0$.

Like Newton's method, Algorithm 10 can be shown to have an asymptotically quadratic convergence. However, whereas Newton's method successively interpolates ψ by its tangent, Algorithm 10 interpolates ψ by a rational function, which, as demonstrated in Figure 5.1, leads to faster convergence due to the convexity of the rational functions in the considered interval. In the simulations, Algorithm 10 usually attains an accuracy of 10^{-9} within 4 iterations.

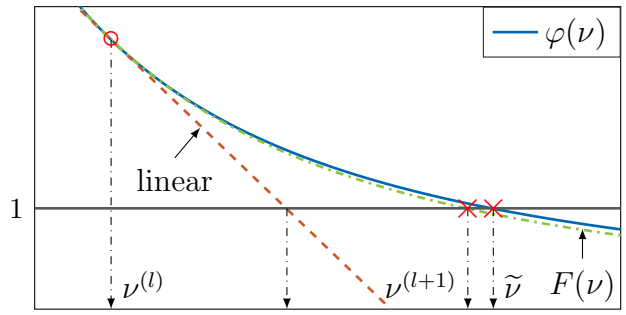


Figure 5.1. Linear approximation (Newton's method) vs. rational approximation

5.3.5 Stopping Criterion

As demonstrated in Section (4.2.2), the version of the smoothing SCA for a regularized problem (4.39) asymptotically converges to a C-stationary point of an equivalent refor-

mulation (4.41) of the regularized problem where the convex nonsmooth regularization is moved to the constraints, and the stationarity condition for the convergent point is mathematically expressed in (4.47). Customizing the stationarity condition (4.47) for the cPRDL problem in (5.5), we claim that the compact-SCAphase Algorithm 9 converges to a stationary point $\boldsymbol{S}^\star = (\boldsymbol{D}^\star, \boldsymbol{Z}^\star)$ that satisfies

$$\mathbf{0} \in \partial^C_{(\boldsymbol{D},\boldsymbol{Z})} f(\boldsymbol{S}^\star) + \underbrace{\left(\{\mathbf{0}\} \times \partial^C_{\boldsymbol{Z}} g(\boldsymbol{Z}^\star)\right)}_{\partial^C_{(\boldsymbol{D},\boldsymbol{Z})} g(\boldsymbol{Z}^\star)} + \underbrace{\left(\mathcal{N}_{\mathcal{D}}(\boldsymbol{D}^\star) \times \{\mathbf{0}\}\right)}_{\partial^C_{(\boldsymbol{D},\boldsymbol{Z})} \mathbb{I}_{\mathcal{D}}(\boldsymbol{D}^\star)}. \tag{5.29}$$

Ideally, Algorithm 9 can be terminated when $\boldsymbol{S}^{(t)}$ fulfills the above stationarity condition, which, however, cannot be easily verified due to the nonconvexity and nonsmoothness of f. On the other hand, the subgradient consistency (5.10) at $\boldsymbol{S}^{(t)}$ implies that, if $\boldsymbol{S}^{(t)}$ is stationary for the majorizing function according to the condition (5.29) with f replaced by $\widehat{f}$, i.e.,

$$\mathbf{0} \in \left\{\nabla \widehat{f}\left(\boldsymbol{S}; \boldsymbol{S}^{(t)}\right)\right\} + \left(\{\mathbf{0}\} \times \partial^C_{\boldsymbol{Z}} g(\boldsymbol{Z})\right) + \left(\mathcal{N}_{\mathcal{D}}(\boldsymbol{D}) \times \{\mathbf{0}\}\right), \tag{5.30}$$

then it also satisfies the stationary condition (5.29) for the original problem (5.5). The condition (5.30) can be easily evaluated. In particular, omitting the trivial intermediate calculations of the Clarke subdifferential $\partial^C_{\boldsymbol{Z}} g$ and the normal cone $\mathcal{N}_{\mathcal{D}}$, we can further express the condition (5.30) as follows: for all $p = 1, \ldots, P$ and $i = 1, \ldots, I$,

$$\nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \begin{cases} \mathbf{0} & \text{if } \|\boldsymbol{d}_p\|_2 < 1, \\ -\left\|\nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})\right\|_2 \boldsymbol{d}_p & \text{if } \|\boldsymbol{d}_p\|_2 = 1, \end{cases} \tag{5.31a}$$

$$\text{and} \quad \begin{cases} \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = -\lambda \mathrm{e}^{\mathsf{j} \arg(z_{p,i})} & \text{if } z_{p,i} \neq 0, \\ \left|\nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})\right| \leq \lambda & \text{if } z_{p,i} = 0. \end{cases} \tag{5.31b}$$

Thus, we define the minimum-norm subgradient $\nabla^{\mathsf{S}} \widehat{h}$ as the element of the sum of the Clarke subdifferentials on the right-hand side in the condition (5.30) with the minimum Frobenius norm [LTYP19]. Its components at the point $\boldsymbol{S}$ are given as follows: for all $p = 1, \ldots, P$ and $i = 1, \ldots, I$,

$$\nabla^{\mathsf{S}}_{\boldsymbol{d}_p} \widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \begin{cases} \nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) & \text{if } \|\boldsymbol{d}_p\|_2 < 1, \\ \nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) - \dfrac{\min\left\{0, \Re\left(\boldsymbol{d}_p^{\mathsf{H}} \nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})\right)\right\}}{\left\|\nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})\right\|_2} \boldsymbol{d}_p & \text{if } \|\boldsymbol{d}_p\|_2 = 1, \end{cases}$$

$$\nabla^{\mathsf{S}}_{z_{p,i}} \widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \begin{cases} \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) + \lambda \mathrm{e}^{\mathsf{j} \arg(z_{p,i})} & \text{if } z_{p,i} \neq 0, \\ \max\left\{0, \left|\nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)})\right| - \lambda\right\} & \text{if } z_{p,i} = 0. \end{cases}$$

The minimum-norm subgradient $\nabla^{\mathsf{S}} \widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)})$ vanishes at $\boldsymbol{S}^{(t)}$ if and only if $\boldsymbol{S}^{(t)}$ fulfills the stationarity condition (5.31). This leads to a termination criterion that the

minimum-norm subgradient must be small, i.e., given a tolerance $\varepsilon > 0$,

$$\begin{cases} \left\| \nabla^{\mathsf{S}}_{\boldsymbol{D}} \widehat{h}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)}) \right\|_{\mathsf{F}} \leq M_1 M_2 \cdot \sqrt{NP} \cdot \varepsilon, \\ \left\| \nabla^{\mathsf{S}}_{\boldsymbol{Z}} \widehat{h}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)}) \right\|_{\mathsf{F}} \leq M_1 M_2 \cdot \sqrt{PI} \cdot \varepsilon, \end{cases} \tag{5.32}$$

where the sizes of measurements and variables are considered.

Recall that the stationary points of the problem (5.5) satisfying (5.29) include not only local minima but also local maxima and saddle points of (5.5). As mentioned in Section 4.2, a subset of stationary points that are not local minima can be easily identified by Proposition 4.3. Moreover, this motivates the additional stopping criterion in the smoothing SCA framework described in Algorithm 4 for a general regularized problem so as to exclude the subset of stationary points that are not local minima. This stopping criterion is also customized for the problem (5.5) and included in Algorithm 9.

5.3.6 Debiasing

Having obtained a stationary point $(\boldsymbol{D}^\star, \boldsymbol{Z}^\star)$ of problem (5.5) using Algorithm 9, one may optionally perform a debiasing step, which minimizes only the data fidelity term f with the restriction that the components $z_{p,i}$ having zero values in $\boldsymbol{Z}^\star$ are fixed at zero [FNW07]. It can be solved by Algorithm 9 with a slight modification. That is, each subproblem that involves an entry $z_{p,i}$ that can take nonzero values reduces to an unconstrained univariate convex quadratic program, which has the closed-form solution

$$\widetilde{z}^{(t)}_{pi} = z^{(t)}_{p,i} - \frac{\nabla_{z_{p,i}} \widehat{f}\left(\boldsymbol{S}; \boldsymbol{S}^{(t)}\right)}{\left\| \boldsymbol{F}_i \boldsymbol{d}^{(t)}_p \right\|^2_2}.$$

Also, the point $(\boldsymbol{D}^\star, \boldsymbol{Z}^\star)$ can be chosen as the initial point.

Essentially, the regularized problem (5.5) is used to select the relevant elements of matrix $\boldsymbol{Z}$, whereas the debiasing step chooses the optimal values for those entries according to only the data fitting term f. However, we remark that debiasing is not always beneficial because it may also undo the desirable noise suppression introduced by the shrinkage [Don95].

5.4 Proposed Algorithm for Formulation PRDL

With the increase of the diversity of the linear measurement operator $\mathcal{F}$, the per-iteration complexity of compact-SCAphase dramatically grows due to the computation

of partial Hessians and SVDs of $\boldsymbol{H}_p$ for $p = 1, \ldots, P$. Therefore, in this section, we propose the SCAphase algorithm for the conventional formulation (5.6) similarly based on the smoothing SCA framework in Section 4.2.

Let $h(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{Z}) = f(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{Z}) + g(\boldsymbol{Z})$ denote the objective function in (5.6) with

$$f(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{Z}) = \frac{1}{2}\|\boldsymbol{Y} - |\mathcal{F}(\boldsymbol{X})|\|_{\mathsf{F}}^2 + \frac{\mu}{2}\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{Z}\|_{\mathsf{F}}^2 \quad \text{and} \quad g(\boldsymbol{Z}) = \rho\|\boldsymbol{Z}\|_{1,1}. \tag{5.33}$$

The first component f is nonconvex and nonsmooth, and the sparsity regularization g is convex but nonsmooth. In each iteration, we first find a descent direction by solving a separable convex approximate problem that is constructed based on a smoothing majorization for f. Then all variables are jointly updated along the descent direction by exact line search, which ensures a decrease of the original function h. An optional debiasing step, similar to that in Section 5.3, can be applied to the PRDL problem after a stationary point is obtained, to further improve the accuracy.

5.4.1 Smooth Majorization and Separable Convex Approximation

Similarly, let $\boldsymbol{S} = (\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{Z})$ be the collection of all variables. At iteration t, by the same majorization technique as in Section 5.3.1, we can derive a smooth majorizing function

$$\widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \frac{1}{2}\big\|\boldsymbol{Y}^{(t)} - \mathcal{F}(\boldsymbol{X})\big\|_{\mathsf{F}}^2 + \frac{\mu}{2}\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{Z}\|_{\mathsf{F}}^2 \tag{5.34}$$

for f at the current point $\boldsymbol{S}^{(t)} = (\boldsymbol{X}^{(t)}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$ that satisfies Assumption 4.1, where $\boldsymbol{Y}^{(t)} = \boldsymbol{Y} \odot \mathrm{e}^{\mathrm{j}\arg(\mathcal{F}(\boldsymbol{X}^{(t)}))}$. It has the partial gradients

$$\begin{aligned}
\nabla_{\boldsymbol{X}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) &= \mathcal{F}^*(\mathcal{F}(\boldsymbol{X}) - \boldsymbol{Y}^{(t)}) + \mu(\boldsymbol{X} - \boldsymbol{D}\boldsymbol{Z}), \\
\nabla_{\boldsymbol{D}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) &= \mu(\boldsymbol{D}\boldsymbol{Z} - \boldsymbol{X})\boldsymbol{Z}^{\mathsf{H}}, \\
\nabla_{\boldsymbol{Z}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) &= \mu\boldsymbol{D}^{\mathsf{H}}(\boldsymbol{D}\boldsymbol{Z} - \boldsymbol{X}).
\end{aligned} \tag{5.35}$$

Note that $\widehat{f}$ is nonconvex due to the bilinear map $\boldsymbol{D}\boldsymbol{Z}$. Then $\widehat{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) + g(\boldsymbol{Z})$ is a majorization of h at $\boldsymbol{S}^{(t)}$.

We then minimize a separable convex approximation of the majorizing function $\widehat{h}$ as the exact minimization of $\widehat{h}$ requires an iterative algorithm and, therefore, is computationally expensive. The following Jacobi-type approximation for $\widehat{h}$ is employed:

$$\widetilde{h}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \widetilde{f}_X(\boldsymbol{X}; \boldsymbol{S}^{(t)}) + \widetilde{f}_D(\boldsymbol{D}; \boldsymbol{S}^{(t)}) + \widetilde{f}_Z(\boldsymbol{Z}; \boldsymbol{S}^{(t)}) + \rho\|\boldsymbol{Z}\|_{1,1},$$

where $\widetilde{f}_X(\boldsymbol{X};\boldsymbol{S}^{(t)})$, $\widetilde{f}_D(\boldsymbol{D};\boldsymbol{S}^{(t)})$ and $\widetilde{f}_Z(\boldsymbol{Z};\boldsymbol{S}^{(t)})$ denote the approximate functions of $\widehat{f}$ over three block variables, respectively. The approximate functions $\widetilde{f}_D$ and $\widetilde{f}_Z$ are constructed in the same way as (5.12) in Section 5.3. To limit the complexity of minimizing $\widetilde{h}$, we perform the Jacobi-type approximation on each entry of $\boldsymbol{X}$, which leads to the approximation

$$\widetilde{f}_X(\boldsymbol{X};\boldsymbol{S}^{(t)}) = \sum_{i=1}^{I}\sum_{n=1}^{N} \widehat{f}\left(x_{n,i}, \boldsymbol{X}^{(t)}_{-(n,i)}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)}; \boldsymbol{S}^{(t)}\right),$$

where $\boldsymbol{X}_{-(n,i)}$ is the collection of all entries of $\boldsymbol{X}$ except $x_{n,i}$. The approximate problem at the tth iteration then reads

$$\left(\widetilde{\boldsymbol{X}}^{(t)}, \widetilde{\boldsymbol{D}}^{(t)}, \widetilde{\boldsymbol{Z}}^{(t)}\right) = \underset{\boldsymbol{X},\boldsymbol{D}\in\mathcal{D},\boldsymbol{Z}}{\operatorname{argmin}} \ \widetilde{h}(\boldsymbol{S};\boldsymbol{S}^{(t)}). \tag{5.36}$$

Likewise, problem (5.36) can be decomposed into independent subproblems, each of which exclusively depends on a column $\boldsymbol{d}_p$ or a single variable $x_{n,i}$ or $z_{p,i}$ and can be solved in parallel.

Define $\Delta\boldsymbol{X} = \widetilde{\boldsymbol{X}}^{(t)} - \boldsymbol{D}^{(t)}$, $\Delta\boldsymbol{D} = \widetilde{\boldsymbol{D}}^{(t)} - \boldsymbol{D}^{(t)}$, and $\Delta\boldsymbol{Z} = \widetilde{\boldsymbol{Z}}^{(t)} - \boldsymbol{Z}^{(t)}$. Then the following simultaneous update rule along the descent direction $(\Delta\boldsymbol{X}, \Delta\boldsymbol{D}, \Delta\boldsymbol{Z})$ of $\widehat{h}(\boldsymbol{S};\boldsymbol{S}^{(t)})$ is applied:

$$\begin{aligned} \boldsymbol{X}^{(t+1)} &= \boldsymbol{X}^{(t)} + \gamma^{(t)}\Delta\boldsymbol{X}, \\ \boldsymbol{D}^{(t+1)} &= \boldsymbol{D}^{(t)} + \gamma^{(t)}\Delta\boldsymbol{D}, \\ \boldsymbol{Z}^{(t+1)} &= \boldsymbol{Z}^{(t)} + \gamma^{(t)}\Delta\boldsymbol{Z}, \end{aligned} \tag{5.37}$$

with $\gamma^{(t)} \in (0, 1]$ being the step size. When $(\widetilde{\boldsymbol{X}}^{(t)}, \widetilde{\boldsymbol{D}}^{(t)}, \widetilde{\boldsymbol{Z}}^{(t)}) = (\boldsymbol{X}^{(t)}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$, the algorithm has converged to a stationary point of the convex approximation $\widetilde{h}(\boldsymbol{S};\boldsymbol{S}^{(t)})$, which is also stationary for the majorization and the original problem (5.5) in the Clarke sense according to Theorem 4.2.

In the following, the closed-form solutions for the subproblems decomposed from (5.36) are derived.

Descent Direction for $\boldsymbol{X}$

First, since $\widehat{f}$ is quadratic with respect to $\boldsymbol{X}$, each subproblem involving an entry $x_{n,i}$ is a univariate quadratic program and has a closed-form solution

$$\widetilde{x}^{(t)}_{n,i} = x^{(t)}_{n,i} - \frac{\nabla_{x_{n,i}}\widehat{f}(\boldsymbol{S}^{(t)};\boldsymbol{S}^{(t)})}{\|\boldsymbol{f}_{n+(i-1)N}\|_2^2 + \mu} \tag{5.38}$$

with $\boldsymbol{f}_{n+(i-1)N}$ being the $(n+(i-1)N)$th column of $\boldsymbol{F}$ in (5.4).

Descent Direction for $\boldsymbol{D}$

Next, the P independent subproblems decomposed from (5.36) that involve $\boldsymbol{D}$ are

$$\widetilde{\boldsymbol{d}}_p^{(t)} = \underset{\boldsymbol{d}_p}{\operatorname{argmin}} \frac{1}{2}\left\| \boldsymbol{X}^{(t)} - \boldsymbol{D}_{-p}^{(t)}\boldsymbol{Z}_{-p}^{(t)} - \boldsymbol{d}_p \boldsymbol{z}_{p:}^{(t)\mathsf{T}} \right\|_{\mathsf{F}}^2 \quad \text{s.t. } \|\boldsymbol{d}_p\|_2 \leq 1, \tag{5.39}$$

which can again be solved via the KKT optimality system. Unlike (5.15), problem (5.39) has a simple closed-form solution

$$\widetilde{\boldsymbol{d}}_p^{(t)} = \frac{\widehat{\boldsymbol{d}}_p}{\max\left\{1, \left\|\widehat{\boldsymbol{d}}_p\right\|_2\right\}} \quad \text{with} \quad \widehat{\boldsymbol{d}}_p = \boldsymbol{d}_p^{(t)} - \frac{\nabla_{\boldsymbol{d}_p}\widehat{f}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)})}{\mu\left\|\boldsymbol{z}_{p:}^{(t)}\right\|_2^2}. \tag{5.40}$$

Descent Direction for $\boldsymbol{Z}$

Then each subproblem involving an entry $z_{p,i}$ is the Lagrangian form of a univariate LASSO problem and has a closed-form solution [Don95]

$$\widetilde{z}_{p,i}^{(t)} = \frac{1}{\left\|\boldsymbol{d}_p^{(t)}\right\|_2^2} \mathcal{S}_{\frac{\rho}{\mu}}\left(\left\|\boldsymbol{d}_p^{(t)}\right\|_2^2 z_{p,i}^{(t)} - \frac{1}{\mu}\nabla_{z_{p,i}}\widehat{f}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)}) \right) \tag{5.41}$$

with the soft thresholding operator $\mathcal{S}_{\frac{\rho}{\mu}}$ defined in (5.1).

5.4.2 Step Size Computation

Similarly to Section 5.3.3, to efficiently find a step size $\gamma^{(t)}$ for the update in (5.37) that ensures a decrease of the original function in (5.6), we perform the modified exact line search method (4.65) on the majorizing function, which is customized for the majorizing function $\widehat{f}$ in (5.34) as

$$\gamma^{(t)} = \underset{\gamma\in[0,1]}{\operatorname{argmin}}\Big\{ \widehat{f}\big(\boldsymbol{X}^{(t)} + \gamma\Delta\boldsymbol{X}, \boldsymbol{D}^{(t)} + \gamma\Delta\boldsymbol{D}, \boldsymbol{Z}^{(t)} + \gamma\Delta\boldsymbol{Z}; \boldsymbol{S}^{(t)}\big) + \gamma\big(g(\widetilde{\boldsymbol{Z}}^{(t)}) - g(\boldsymbol{Z}^{(t)})\big)\Big\}. \tag{5.42}$$

Problem (5.42) is also a minimization of fourth-order polynomial and can be solved analytically by following the same procedure as in Section 5.3.3, i.e., rooting its derivative, which is a cubic polynomial; we omit the straightforward details.

Finally, the proposed SCAphase algorithm for solving the PRDL problem in (5.6) is outlined in Algorithm 11.

Algorithm 11: SCAphase

Input: $\boldsymbol{Y} \in \mathbb{R}_{+}^{M_1 \times M_2}$, $\mu \geq 0$, $\lambda \geq 0$, tolerance $\varepsilon > 0$

Initialize $\boldsymbol{X}^{(0)}$ and $\boldsymbol{D}^{(0)} \in \mathcal{D}$ randomly, $\boldsymbol{Z}^{(0)} \leftarrow (\boldsymbol{D}^{(0)})^{\dagger}\boldsymbol{X}^{(0)}$, $t \leftarrow 0$;
repeat
  for $n = 1, \ldots, N,\ i = 1, \ldots, I$ **do** in parallel
    Compute $\widetilde{x}_{n,i}^{(t)}$ according to (5.38);
  end
  for $p = 1, \ldots, P$ **do** in parallel
    Compute $\widetilde{\boldsymbol{d}}_{p}^{(t)}$ according to (5.40);
  end
  for $p = 1, \ldots, P,\ i = 1, \ldots, I$ **do** in parallel
    Compute $\widetilde{z}_{p,i}^{(t)}$ according to (5.41);
  end
  Compute step size $\gamma^{(t)}$ by exact line search (5.42);
  Update the variables using (5.37) and $t \leftarrow t + 1$;
until *stopping criterion* (5.46) *achieved* **and** *(* $\boldsymbol{D}^{(t)} \notin \operatorname{int}(\mathcal{D})$ **or** $\nabla f(\boldsymbol{X}^{(t)}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)})$ *exists)*;
return $\boldsymbol{X}^{(t)}, \boldsymbol{D}^{(t)}, \boldsymbol{Z}^{(t)}$

5.4.3 Stopping Criterion

Customizing the stationarity condition (4.47) for the PRDL problem in (5.6), we claim that the SCAphase Algorithm 11 asymptotically converges to a stationary point $\boldsymbol{S}^{\star} = (\boldsymbol{X}^{\star}, \boldsymbol{D}^{\star}, \boldsymbol{Z}^{\star})$ of (5.6) that satisfies

$$\mathbf{0} \in \partial_{\boldsymbol{S}}^{C} f(\boldsymbol{S}^{\star}) + \partial_{\boldsymbol{S}}^{C} g(\boldsymbol{Z}^{\star}) + \partial_{\boldsymbol{S}}^{C} \mathbb{I}_{\mathcal{D}}(\boldsymbol{D}^{\star}), \tag{5.43}$$

which, however, cannot be easily verified. Similar to the compact-SCAphase algorithm, we can evaluate the optimality of the current solution $\boldsymbol{S}^{(t)}$ by the stationarity of the majorizing function, i.e., the condition (5.43) with f replaced by $\widehat{f}$:

$$\mathbf{0} \in \left\{ \nabla_{\boldsymbol{S}} \widehat{f}\left(\boldsymbol{S}; \boldsymbol{S}^{(t)}\right) \right\} + \partial_{\boldsymbol{S}}^{C} g(\boldsymbol{Z}) + \partial_{\boldsymbol{S}}^{C} \mathbb{I}_{\mathcal{D}}(\boldsymbol{D}), \tag{5.44}$$

since is a necessary condition for (5.43) because of the subgradient consistency between the majorizing and original functions, i.e., $\nabla \widehat{f}\left(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)}\right) \in \partial^{C} f\left(\boldsymbol{S}^{(t)}\right)$. Thus, analogously to Section 5.3.5, the minimum-norm subgradient $\nabla^{\mathrm{S}} \widehat{h}$ is introduced based on the stationarity condition (5.44) of the majorizing function to evaluate the quality of the current solution. Omitting the trivial calculations of the involved Clarke

subdifferentials, we can further express the condition (5.44) as follows:

$$\nabla_{\boldsymbol{X}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \boldsymbol{0}, \tag{5.45a}$$

$$\nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = \begin{cases} \boldsymbol{0} & \text{if } \|\boldsymbol{d}_p\|_2 < 1, \\ -\left\| \nabla_{\boldsymbol{d}_p} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) \right\|_2 \boldsymbol{d}_p & \text{if } \|\boldsymbol{d}_p\|_2 = 1, \end{cases} \tag{5.45b}$$

$$\text{and} \quad \begin{cases} \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) = -\rho \mathrm{e}^{\mathrm{j} \arg(z_{p,i})} & \text{if } z_{p,i} \neq 0, \\ \left| \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{S}; \boldsymbol{S}^{(t)}) \right| \leq \rho & \text{if } z_{p,i} = 0. \end{cases} \tag{5.45c}$$

for all $p = 1, \ldots, P$ and $i = 1, \ldots, I$. Then the components of the minimum-norm subgradient with respect to matrices $\boldsymbol{Z}$ and $\boldsymbol{D}$ are defined in the same way as in Section 5.3.5, whereas the component $\nabla^{\mathsf{S}}_{\boldsymbol{X}} \widehat{h}$ of the minimum-norm subgradient with respect to $\boldsymbol{X}$ is simply defined as the gradient $\nabla_{\boldsymbol{X}} \widehat{f}$. The algorithm is terminated when the minimum-norm subgradient is sufficiently small, i.e., with a given tolerance $\varepsilon > 0$,

$$\begin{cases} \|\nabla^{\mathsf{S}}_{\boldsymbol{D}} \widehat{h}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)})\|_{\mathsf{F}} \leq M_1 M_2 \cdot \sqrt{NP} \cdot \varepsilon, \\ \|\nabla^{\mathsf{S}}_{\boldsymbol{Z}} \widehat{h}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)})\|_{\mathsf{F}} \leq M_1 M_2 \cdot \sqrt{PI} \cdot \varepsilon, \\ \|\nabla^{\mathsf{S}}_{\boldsymbol{X}} \widehat{h}(\boldsymbol{S}^{(t)}; \boldsymbol{S}^{(t)})\|_{\mathsf{F}} \leq M_1 M_2 \cdot \sqrt{NI} \cdot \varepsilon. \end{cases} \tag{5.46}$$

Likewise, the additional stopping criterion in the smoothing SCA framework described in Algorithm 4 is customized for the problem (5.6) and included in Algorithm 11, in order to exclude a subset of stationary points of (5.6) that are not local minima.

5.4.4 Comparison with SC-PRIME

The proposed SCAphase algorithm and the state-of-the-art SC-PRIME [QP17] adopt the same formulation, i.e., the PRDL problem in (5.6), and the same successive majorization technique (5.34) that preserves only a subgradient of the original function f, which differs from the stricter derivative consistency in Assumption 3.4 required by the classic BSUM at nonsmooth points of f. In [QP17], the authors address the convergence of SC-PRIME under the BSUM framework. This convergence analysis is incomplete since the aforementioned fact that the derivative consistency required by BSUM is not always satisfied is ignored. Nevertheless, SC-PRIME can be viewed as an instance of the generalized BSUM framework with smoothing majorization that we proposed in Section 4.3.1. Hence, the convergence there justifies that SC-PRIME converges to a coordinatewise stationary point of (5.6) corresponding to the same generalized concept of stationarity as SCAphase, i.e., C-stationarity, instead of the stricter d-stationarity in the classic BSUM framework. Then there are two main differences between SC-PRIME and the proposed SCAphase algorithm.

First, SC-PRIME updates the variables in a BCD manner, i.e., minimizes the majorizing function $\widehat{h}$ alternatively with respect to each block variable $\boldsymbol{X}$, $\boldsymbol{Z}$, and each column of $\boldsymbol{D}$, instead of using parallel updates. Since the problem (5.6) is not guaranteed to be coordinatewise regular everywhere, SC-PRIME may converge only to a coordinatewise stationary point of (5.6), but in the same generalized sense as in (5.43).

Next, to avoid the expensive exact minimization of $\widehat{h}$, SC-PRIME minimizes a different separable convex approximation for each block variable from SCAphase. Instead of using the Jacobi-type approximation, SC-PRIME further majorizes the LS objective $\widehat{f}$ by replacing the partial Hessian with respect to a block variable by the identity matrix scaled by an upper bound of its eigenvalues. The separability of this majorization leads to a minimizer in closed form and a decrease of the original objective function h is ensured without a step size search. Nevertheless, since the Hessian is typically ill-conditioned, this majorization tends to be conservative, which may result in slow convergence. In contrast, the Jacobi-type approximation $\widetilde{f}$ equivalently preserves all diagonal entries of the Hessian but is not necessarily a majorant of the original function f. Thus, discarding the global upper bound constraint provides more flexibility in designing an approximation that yields faster convergence to a good stationary point. This advantage is demonstrated numerically in Section 5.6.

5.5 Complexity Analysis

Table 5.1. Computational complexity of dominant operations in each iteration

	computation of gradient	computation of partial Hessians	computation of polynomial coefficients in line search function
compact-SCAphase	$c(\mathcal{F}) + 4NPI$	general case: $4M_1M_2NPI + \mathcal{O}(M_1M_2N^2P)$; special case with $\mathcal{F}$ in (5.47): $2M_1NP + 2M_2PI$	$2c(\mathcal{F})+6NPI$
SCAphase	$c(\mathcal{F}) + 4NPI$	$2NP + 2PI$	$c(\mathcal{F}) + 6NPI$
SC-PRIME	$2c(\mathcal{F})+6NPI$	–	–

In this section, we present a theoretic comparison of the complexity of the proposed algorithms, compact-SCAphase and SCAphase, and the state-of-the-art SC-PRIME [QP17].

As presented in Table 5.1, for each algorithm, we count the number of flops [GVL13] required by the dominant operations, such as matrix-matrix multiplication, in each iteration, which reflects the per-iteration complexity in the worst case where the flops are executed in sequence. The per-iteration complexity of the proposed algorithms is dominated by three components: the computation of the gradient and the partial Hessians of the smooth majorization $\widehat{f}$, which are required for solving the convex subproblems, and the computation of the polynomial coefficients of the line search function in the step size computation. In the simulations, the rational approximation algorithm employed by compact-SCAphase for solving the subproblems requires 3 or 4 iterations to achieve a precision of 10^{-9}. Therefore, the complexity of the rational approximation is comparable to that of computing a closed-form solution as in SCAphase and SC-PRIME, which is negligible compared to the other operations. In Table 5.1, $c(\mathcal{F})$ stands for the complexity of the linear operator $\mathcal{F}$ or, equivalently, that of its adjoint $\mathcal{F}^*$, which depends on the structure of $\mathcal{F}$ and the specific implementation. In principle, $c(\mathcal{F})$ admits the bounds $2NI \cdot \max\{M_1, M_2\} \leq c(\mathcal{F}) \leq 2M_1M_2NI$.

Compared to SCAphase, in the general case, compact-SCAphase has a per-iteration complexity of higher order due to the computation of partial Hessians and SVD of matrix $\boldsymbol{H}_p$ in (5.16). However, in the special case with the linear operator $\mathcal{F}$ in (5.47), such as Cases 1 and 2 in the simulations, the complexity of computation of partial Hessians in compact-SCAphase dramatically decreases and the SVD of $\boldsymbol{H}_p$ can be analytically calculated given the SVD of $\boldsymbol{A}$. Then compact-SCAphase and SCAphase have comparable per-iteration complexity. On the other hand, as shown in Figure 5.3, compared to SCAphase, compact-SCAphase typically uses half the number of iterations to achieve a stationary point due to the reduction of variables, which makes compact-SCAphase more competitive than SCAphase in the case with $\mathcal{F}$ in (5.47).

Next, we compare the complexity of SCAphase and SC-PRIME. The line search is not required in SC-PRIME as it employs the BCD update. In the specific implementation of SC-PRIME used in the simulations, constant rough upper bounds for the eigenvalues of the partial Hessians are used to construct the surrogate subproblems and, hence, only the gradient of $\widehat{f}$ is needed. However, compared to SC-PRIME, the additional line search in SCAphase does not cause a significant increase in the overall per-iteration complexity as several intermediate variables in the computation of the gradient can be updated recursively. For example, as $\mathcal{F}$ is a linear operator, $\mathcal{F}(\boldsymbol{X}^{(t)})$ required in (5.35)

is updated recursively by

$$\mathcal{F}(\boldsymbol{X}^{(t+1)}) = \mathcal{F}(\boldsymbol{X}^{(t)}) + \gamma^{(t)}\mathcal{F}(\Delta\boldsymbol{X}),$$

where $\mathcal{F}(\Delta\boldsymbol{X})$ was previously calculated in the computation of the coefficients of the line search function. Thus, SCAphase and SC-PRIME also have similar per-iteration complexity, especially in the case with a highly diverse linear operator $\mathcal{F}$, where the per-iteration complexity is dominated by the complexity of $\mathcal{F}$. On the other hand, with the additional line search, SCAphase exhibits faster convergence in terms of the number of iterations.

Finally, we remark that, in contrast to the BCD update in SC-PRIME, the computation of solutions of subproblems in compact-SCAphase and SCAphase can be fully parallelized with suitable hardware architectures.

5.6 Simulation Results

In this section, we compare the performance of the two proposed algorithms and the state-of-the-art SC-PRIME [QP17] on synthetic data in the context of blind channel estimation in a multi-antenna random access network. All experiments were conducted on a Linux machine assigned with two 2.3 GHz cores and 7 GB RAM running MATLAB R2021b. Although, theoretically, all the subproblems in each iteration in the proposed algorithms can be solved in parallel, for simplicity, the subproblems involving different block variables (i.e., $\boldsymbol{X}$, $\boldsymbol{D}$, or $\boldsymbol{Z}$) are solved sequentially, whereas the computation of solutions for subproblems involving the same block variable is parallelized by using vectorization in MATLAB.

5.6.1 Simulation Setup

We consider a multi-antenna random access network with magnitude-only measurements in Figure 5.2. The base station is equipped with N antennas and P single-antenna users with unknown spatial signatures $\{\mathbf{d}_p \in \mathbb{C}^N\}_{p=1}^P$ sporadically access the channel in I time-slots. In time-slot i user p transmits an unknown information symbol $z_{p,i} \neq 0$ with probability L/P and $z_{p,i} = 0$ with probability $(P-L)/P$, where L defines the expected sparsity level of the transmitted symbol vectors $\boldsymbol{z}_i = [z_{1,i}, \ldots, z_{P,i}]^\mathsf{T}$, $i = 1, \ldots, I$. With $\boldsymbol{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_P]$, the received symbol vector $\boldsymbol{x}_i = [x_{1,i}, \ldots, x_{N,i}]^\mathsf{T}$ at the antennas is given by $\boldsymbol{x}_i = \boldsymbol{D}\boldsymbol{z}_i$, which cannot be directly observed due to heavy

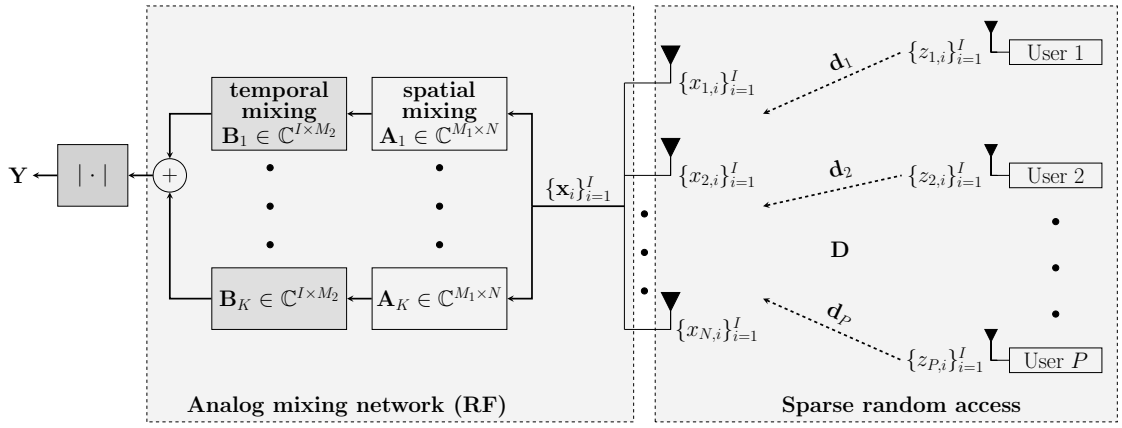


Figure 5.2. Multi-antenna Random Access Network.

phase errors caused by the imperfect phase synchronization and the phase noise of the local oscillators in the down-converters and analog-to-digital converters [ALJ19]. Hence, before down-converted and sampled, the received signals are first processed by an analog mixing network at radio frequency composed of analog phase shifters and analog filters. Then the objective is to jointly learn the spatial signature matrix $\boldsymbol{D}$ and the sparse transmitted symbol vectors $\boldsymbol{z}_i$ from spatially and temporally filtered subband magnitude measurements, which can be expressed by the model in (5.2), whereas the heavily corrupted phase measurements are discarded. Furthermore, the subband measurements can be acquired at a reduced sampling rate according to the bandwidth of the respective subband filters. Compared to communication schemes where the channel estimation and symbol detection are performed sequentially, the presented blind scheme avoids the signaling overhead, i.e., the transmission of pilots for the periodic channel estimation, for time-varying channels. In this application, as shown in Figure 5.2, the linear operator $\mathcal{F}$ in (5.3) is interpreted as K independent receiver chains of linear spatial mixing networks $\{\boldsymbol{A}_k \in \mathbb{C}^{M_1\times N}\}_{k=1}^K$ and temporal mixing networks $\{\boldsymbol{B}_k \in \mathbb{C}^{I\times M_2}\}_{k=1}^K$. Note that the order of the spatial and temporal mixing is interchangeable for each chain. In our simulations, the spatial mixing networks $\{\boldsymbol{A}_k\}_{k=1}^K$ are generated from a standard complex Gaussian distribution, and the following three particular cases of the linear operator $\mathcal{F}$ of different levels of diversity are investigated:

- **Case 1: Time-invariant spatial mixing and no temporal mixing.** In this case, $\mathcal{F}$ is interpreted by a single chain of mixing networks, i.e., $K = 1$. For simplicity, we omit the subscript on the mixing networks, and $\mathcal{F}$ reduces to

$$\mathcal{F}(\boldsymbol{X}) = \boldsymbol{A}\boldsymbol{X}\boldsymbol{B}. \tag{5.47}$$

  Moreover, the temporal mixing is set to be $\boldsymbol{B} = \boldsymbol{I}$.

- **Case 2: Time-invariant spatial mixing and STFT temporal mixing.** In this case, $\mathcal{F}$ can also be expressed by the model (5.47), whereas the temporal mixing $\boldsymbol{B}$ is designed to be the short-time Fourier transform (STFT) [JEH16], which can be implemented by analog subband filters.

- **Case 3: Time-variant spatial mixing and no temporal mixing.** In this case, $\mathcal{F}$ is expressed by the model (5.3) with $K = I$, and the kth temporal mixing is set to be $\boldsymbol{B}_k = [\boldsymbol{0}, \ldots, \boldsymbol{0}, \boldsymbol{e}_k, \boldsymbol{0}, \ldots, \boldsymbol{0}]$ with $\boldsymbol{e}_k$ being a standard basis vector, which simply selects the kth snapshot. Also, $\boldsymbol{A}_k$ is the spatial mixing network designed for the kth snapshot.

The basic simulation setup is as follows. In each time-slot i, L randomly selected elements of the true transmitted sparse signal $\boldsymbol{z}_i^{\text{true}}$ are set to be nonzero. The nonzero elements of matrix $\boldsymbol{Z}^{\text{true}}$, all elements of spatial mixing matrices $\{\boldsymbol{A}_k\}_{k=1}^K$ and the true spatial signature $\boldsymbol{D}^{\text{true}}$ are drawn from an i.i.d. standard complex Gaussian distribution. In particular, the i.i.d. zero-mean Gaussian distribution of $\boldsymbol{D}^{\text{true}}$ corresponds to a rich scattering environment. The magnitude measurements $\boldsymbol{Y}$ are generated according to (5.2) with additive white Gaussian noise. The number of Monte-Carlo runs is 50.

From the solutions $\boldsymbol{D}$ and $\boldsymbol{Z}$ obtained by compact-SCAphase, the variable $\boldsymbol{X}$ is constructed as $\boldsymbol{X} = \boldsymbol{D}\boldsymbol{Z}$ for the performance evaluation. Note that the analog mixing network architecture in Figure 5.2 is also applicable in other applications of the phase retrieval with dictionary learning problem such as diffraction imaging, where various optical masks and filters can be used to increase the diversity of the intensity measurements with the objective to improve the signal recovery. In this application, the signal $\boldsymbol{X}$ is the parameter of interest, and the dictionary $\boldsymbol{D}$ and the sparse code matrix $\boldsymbol{Z}$ are considered as nuisance parameters. In contrast, in the considered application of multi-antenna network, our main target is to estimate the spatial signature matrix $\boldsymbol{D}$ and transmitted signals $\boldsymbol{Z}$. Hence, only the estimation qualities of $\boldsymbol{D}$ and $\boldsymbol{Z}$ are presented in the following simulations. However, the solution $\boldsymbol{X}$ is still required in the disambiguation step, which is described below.

In both formulations (5.5) and (5.6), the variables can only be recovered up to three trivial ambiguities. Specifically, any combination of the following three trivial operations conserves the magnitude measurements and the sparsity pattern of $\boldsymbol{Z}$: 1) global phase shift: $(\boldsymbol{X}, \boldsymbol{Z}) \to (\boldsymbol{X}\mathrm{e}^{\mathrm{j}\phi}, \boldsymbol{Z}\mathrm{e}^{\mathrm{j}\phi})$, 2) scaling: $(\boldsymbol{d}_p, \boldsymbol{z}_{p:}) \to (\alpha_p \boldsymbol{d}_p, \alpha_p^{-1}\boldsymbol{z}_{p:})$ with any constant scalar $\alpha_p \in \mathbb{C}$ and $\alpha_p \neq 0$, 3) permutation: $(\boldsymbol{D}, \boldsymbol{Z}) \to (\boldsymbol{D}\boldsymbol{P}^{\mathsf{T}}, \boldsymbol{P}\boldsymbol{Z})$ with any permutation matrix $\boldsymbol{P} \in \mathbb{R}^{P\times P}$. Also, if no temporal mixing is applied, the signal in each time slot is measured independently and, hence, the global phase ambiguity holds

columnwise, i.e., $(\boldsymbol{x}_i, \boldsymbol{z}_i) \to (\boldsymbol{x}_i \mathrm{e}^{\mathrm{j}\phi_i}, \boldsymbol{z}_i \mathrm{e}^{\mathrm{j}\phi_i})$. In the practical application of communication in the network described in Figure 5.2, the influence of the global phase shift and scaling ambiguities can be removed by using differential encoding. The permutation ambiguity is only an issue in the evaluation of the estimation quality since the users always need to identified on higher network layers.

A disambiguation step is required to measure the estimation quality of the solutions. Let $\boldsymbol{X}^{\text{true}} = \boldsymbol{D}^{\text{true}}\boldsymbol{Z}^{\text{true}}$ be the true received signals. To resolve the global phase ambiguity, the solution $\boldsymbol{X}$ is corrected by the global phase shift

$$\phi^\star = \underset{\phi\in[0,2\pi)}{\operatorname{argmin}} \quad \left\|\boldsymbol{X}\mathrm{e}^{\mathrm{j}\phi} - \boldsymbol{X}^{\text{true}}\right\|_{\mathsf{F}}^2 = \arg\left(\operatorname{tr}\left(\boldsymbol{X}^{\mathsf{H}}\boldsymbol{X}^{\text{true}}\right)\right) \tag{5.48}$$

in the case of temporal mixing, and the phase correction is applied columnwise with

$$\phi_i^\star = \underset{\phi_i\in[0,2\pi)}{\operatorname{argmin}} \quad \left\|\boldsymbol{x}_i\mathrm{e}^{\mathrm{j}\phi_i} - \boldsymbol{x}_i^{\text{true}}\right\|_2^2 = \arg\left(\boldsymbol{x}_i^{\mathsf{H}}\boldsymbol{x}_i^{\text{true}}\right) \tag{5.49}$$

for $i = 1, \ldots, I$, in the case without temporal mixing. For the permutation ambiguity on $\boldsymbol{D}$ and $\boldsymbol{Z}$, a heuristic method is used to find the permutation that best matches the ground-truth with respect to the normalized cross correlation between columns in $\boldsymbol{D}$ and $\boldsymbol{D}^{\text{true}}$. After permutation, the estimation quality of $\boldsymbol{D}$ is evaluated by the minimum normalized squared error (MNSE) defined as

$$\text{MNSE}(\boldsymbol{D}) = \min_{\{\alpha_p\in\mathbb{C}\}_{p=1}^P} \frac{\sum_{p=1}^P \|\alpha_p\boldsymbol{d}_p - \boldsymbol{d}_p^{\text{true}}\|_2^2}{\|\boldsymbol{D}^{\text{true}}\|_{\mathsf{F}}^2}, \tag{5.50}$$

where the optimal solutions are $\alpha_p^\star = \boldsymbol{d}_p^\dagger\boldsymbol{d}_p^{\text{true}} = \boldsymbol{d}_p^{\mathsf{H}}\boldsymbol{d}_p^{\text{true}}$, as $\|\boldsymbol{d}_p\|_2 = 1$, for $p = 1, \ldots, P$. As for $\boldsymbol{Z}$, after permutation, we first perform the same global phase shift $\mathrm{e}^{\mathrm{j}\phi^\star}$ on $\boldsymbol{Z}$ or $\mathrm{e}^{\mathrm{j}\phi_i^\star}$ on each column $\boldsymbol{z}_i$ and then the MNSE of $\boldsymbol{Z}$ is analogously calculated as

$$\text{MNSE}(\boldsymbol{Z}) = \min_{\{\beta_p\in\mathbb{C}\}_{p=1}^P} \frac{\sum_{p=1}^P \|\beta_p\boldsymbol{z}_{p:} - \boldsymbol{z}_{p:}^{\text{true}}\|_2^2}{\|\boldsymbol{Z}^{\text{true}}\|_{\mathsf{F}}^2}, \tag{5.51}$$

where the optimal solutions are $\beta_p^\star = \boldsymbol{z}_{p:}^\dagger\boldsymbol{z}_{p:}^{\text{true}} = \frac{\boldsymbol{z}_{p:}^{\mathsf{H}}\boldsymbol{z}_{p:}^{\text{true}}}{\|\boldsymbol{z}_{p:}\|_2^2}$ for $p = 1, \ldots, P$. Moreover, the accuracy of the support of the estimated $\boldsymbol{Z}$ is evaluated by the metric *F-measure*. In particular, three classic metrics, namely, *Precision*, *Recall*, and *F-measure*, are typically used to evaluate the estimation quality in set retrieval [MRS08, Sec. 8.3]. Specifically, *Precision* is the fraction of retrieved nonzeros that are relevant, i.e., included in the ground-truth, and *Recall* is the fraction of relevant nonzeros that are retrieved. They are calculated as

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \end{aligned} \tag{5.52}$$

with the notions in the following contingency table:

	Relevant	Nonrelevant
Retrieved	true positives (TP)	false positives (FP)
Not retrieved	false negatives (FN)	true negatives (TN)

Then the *F-measure* is the harmonic mean of Precision and Recall, which measures the overall accuracy of the estimated sparsity pattern. Only to avoid the influence of numerical errors, the support of the estimated $\boldsymbol{Z}$ is determined by a threshold of 10^{-6}.

5.6.2 Hyperparameter Choices

Sparsity parameter of the cPRDL problem in (5.5)

The solution for $\boldsymbol{Z}$ in problem (5.5) tends to $\mathbf{0}$ as $\lambda \to \infty$ and there exists an upper bound $\lambda_{\max}$ such that, for $\lambda \geq \lambda_{\max}$, any point with $\boldsymbol{Z} = \mathbf{0}$ is stationary for problem (5.5) [KKL+07]. With knowledge of $\lambda_{\max}$, the problem of searching for a suitable sparsity regularization parameter λ for an instance is significantly reduced, since any $\lambda \geq \lambda_{\max}$ is ineffective.

From the stationarity conditions (5.31), an upper bound

$$\lambda_{\max} = \|\boldsymbol{Y}\|_{\mathsf{F}} \cdot \max_{i=1,\ldots,I} \{\sigma_{\max}(\boldsymbol{F}_i)\} \tag{5.53}$$

can be derived, where $\sigma_{\max}(\cdot)$ denotes the largest singular value. For $\lambda \geq \lambda_{\max}$, any point $(\boldsymbol{D}, \mathbf{0})$ with $\boldsymbol{D} \in \mathcal{D}$ is stationary for the original problem (5.5). Moreover, it is easy to verify that all points $(\boldsymbol{D}, \mathbf{0})$ with $\boldsymbol{D} \in \mathcal{D}$ are equally optimal for problem (5.5).

For the three investigated cases of the linear operator $\mathcal{F}$, $\lambda_{\max}$ can be further decreased. In Cases 1 and 2, where the spatial mixing is time-invariant, $\lambda_{\max}$ can be decreased to

$$\lambda_{\max} = \sigma_{\max}(\boldsymbol{A}) \cdot \max_{i=1,\ldots,I} \left\{ \sum_{m=1}^{M_2} |b_{i,m}| \cdot \|\boldsymbol{y}_m\|_2 \right\}. \tag{5.54}$$

In Case 3, $\lambda_{\max}$ can be decreased to

$$\lambda_{\max} = \max_{i=1,\ldots,I} \left\{ \sigma_{\max}(\boldsymbol{A}_i) \cdot \|\boldsymbol{y}_i\|_2 \right\}. \tag{5.55}$$

The intermediate derivations of the upper bounds $\lambda_{\max}$ in (5.53)-(5.55) can be found in Appendix A.3.

Regularization parameters of the PRDL problem in (5.6)

Problem (5.6) has two regularization parameters μ and ρ. Similar to λ in (5.5), ρ adjusts the sparsity level of matrix $\boldsymbol{Z}$, whereas μ controls the trade-off between the data fidelity and the approximation quality of the sparse representation.

Similarly, for the sparsity parameter ρ in (5.6), there exists an upper bound $\rho_{\max}$ such that, for any $\rho \geq \rho_{\max}$, problem (5.6) always admits a stationary point with $\boldsymbol{Z} = \boldsymbol{0}$. From the stationarity conditions (5.45)-(5.45a), we obtain an upper bound

$$\rho_{\max} = \frac{\mu \cdot \sigma_{\max}(\boldsymbol{F}) \cdot \|\boldsymbol{Y}\|_{\mathsf{F}}}{\sigma_{\min}^2(\boldsymbol{F}) + \mu}. \tag{5.56}$$

$\sigma_{\min}(\cdot)$ denotes the smallest singular value, which may be zero. Furthermore, in Cases 1 and 3, where no temporal mixing is applied, each snapshot $\boldsymbol{x}_i$ is observed independently and, hence, the upper bound $\rho_{\max}$ can be decreased to

$$\rho_{\max} = \max_{i=1,\ldots,I} \left\{ \frac{\mu \cdot \sigma_{\max}(\boldsymbol{A}_i) \cdot \|\boldsymbol{y}_i\|_2}{\sigma_{\min}^2(\boldsymbol{A}_i) + \mu} \right\}. \tag{5.57}$$

Note that Case 1 can be viewed as a special case of Case 3 where $\boldsymbol{A}_i = \boldsymbol{A}$ for all snapshots. The derivations of the upper bounds $\rho_{\max}$ in (5.56)-(5.57) are provided in Appendix A.4.

Next, to analyze the effect of parameter μ, we write the gradient $\nabla_{\boldsymbol{X}} \widehat{f}$ as

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{X})} \widehat{f}(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{Z}; \boldsymbol{S}^{(t)}) = \left(\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \mu \boldsymbol{I}_{NI}\right) \mathrm{vec}(\boldsymbol{X}) \\ - \left(\boldsymbol{F}^{\mathsf{H}} \mathrm{vec}(\boldsymbol{Y}^{(t)}) + \mu \, \mathrm{vec}(\boldsymbol{D}\boldsymbol{Z})\right)\end{aligned} \tag{5.58}$$

with the vectorized form in (5.4). Then the stationarity condition (5.45a) can be rewritten as

$$\mathrm{vec}(\boldsymbol{X}) = \left(\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \mu \boldsymbol{I}_{NI}\right)^{-1} \boldsymbol{F}^{\mathsf{H}} \mathrm{vec}(\boldsymbol{Y}^{(t)}) + \left(\frac{1}{\mu}\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \boldsymbol{I}_{NI}\right)^{-1} \mathrm{vec}(\boldsymbol{D}\boldsymbol{Z}). \tag{5.59}$$

As shown in (5.59), μ offers some control over how much the value of $\mathrm{vec}(\boldsymbol{X})$ at a stationary point of $\widehat{h}$ is influenced by the data fitting solution $\boldsymbol{F}^{\dagger} \mathrm{vec}(\boldsymbol{Y}^{(t)})$ and the sparse representation $\mathrm{vec}(\boldsymbol{D}\boldsymbol{Z})$. Also, the trade-off depends on both μ and $\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F}$. Thus, we propose to set μ to be proportional to $\sigma_{\min,\mathrm{nz}}^2(\boldsymbol{F})$, where $\sigma_{\min,\mathrm{nz}}(\cdot)$ denotes the smallest nonzero singular value. However, a suitable ratio has to be found by experiments.

Given training data, for which the ground-truth is available, one can quickly learn the suitable values of the regularization parameters by grid search with the upper bound

$\lambda_{\max}$ ($\rho_{\max}$) derived above, which is how we choose the regularization parameters in our simulations. Alternatively, the algorithm unrolling technique [MLE21] can be used to unrolls the iterative algorithm procedure into a deep network and then suitable values of the regularization parameters can be more efficiently learned by state-of-the-art deep learning methods from the given training data. Also, advanced approaches such as Expectation-Maximization-based methods [TDB21] may be applied for simultaneous estimation of hyperparameters, which is the subject of future research.

5.6.3 Computational Experiments

In the following, we evaluate the complexity and estimation accuracy of the proposed algorithms under various parameter setups, in comparison to SC-PRIME. The number of receive antennas is set to $N = 64$. The algorithms are terminated when the minimum-norm subgradient has achieved the tolerance $\varepsilon = 10^{-5}$ or after a maximum number of 2000 iterations. A following debiasing step is performed with the same termination condition. By default, the SNR is 15 dB, the spatial over-sampling rate is $M_1/N = 4$, and $I = 16N$ time slots are taken.

5.6.3.1 Case 1 – Time-invariant spatial mixing and no temporal mixing

We first consider the case without temporal mixing. The regularization parameters are set as follows: $\mu = \sigma^2_{\min,\mathrm{nz}}(\boldsymbol{F}) = \sigma^2_{\min,\mathrm{nz}}(\boldsymbol{A})$ for both SCAphase and SC-PRIME, $\lambda = 0.75^{16}\lambda_{\max}$ with $\lambda_{\max}$ in (5.54) for compact-SCAphase, and $\rho = 0.75^{16}\rho_{\max}$ with $\rho_{\max}$ in (5.57) for SCAphase. Although SC-PRIME adopts the same formulation, i.e., problem (5.6), as SCAphase, it typically requires a larger sparsity parameter ρ for achieving a good solution, due to the loose majorization on the data fitting term employed in the surrogate subproblems. Thus, for SC-PRIME, ρ is set to be $0.75^{15}\rho_{\max}$ and $0.75^{14}\rho_{\max}$ in the cases with $P = N/2$ and $P = N$, respectively.

Varying sparsity level: In the first simulation, as depicted in Figures 5.3 and 5.4, the performance of the algorithms is evaluated for various choices of $\{P, L/P\}$. The number of users P is varied in $\{N/2, N\}$, and the density of active users in each time-slot, i.e., L/P, is limited to be $\{0.025, 0.05, 0.1, 0.2, 0.4\}$. As both problems (5.5) and (5.6) are nonconvex, multiple random initializations are used to increase the chance of finding the global optimal solution. Specifically, for each Monte-Carlo trial, 10 initializations are performed, and the best reconstructed signal, determined by the lowest objective function value, is retained and further improved by a debiasing step. The estimation

error performance of the algorithms is compared in Figure 5.3, and the complexity in Figure 5.4. In particular, both the total number of iterations and computational time, including that of the debiasing step, are reported. The robustness of the algorithms to initialization is investigated afterward in Figure 5.5.

From Figure 5.3, it can be observed that sparse channel access, i.e., a small value of L/P, is required for all algorithms to achieve good recovery performance. However, in the extremely sparse case, the received signals $\boldsymbol{X}^{\text{true}} = \boldsymbol{D}^{\text{true}}\boldsymbol{Z}^{\text{true}}$ contain only few linear combinations of columns of spatial signature $\boldsymbol{D}^{\text{true}}$, which results in a degradation of estimation qualities. Furthermore, as shown in Figure 5.4, for all choices of $\{P, L/P\}$, SC-PRIME does not converge within 2000 iterations. The solution obtained by SC-PRIME within 2000 iterations can be improved by using a larger sparsity parameter ρ than that in SCAphase, as in the parameter setup of this simulation. However, in Figure 5.3, SC-PRIME still exhibits the poorest accuracy performance for most choices of $\{P, L/P\}$, compared to the other algorithms.

When $P = N/2$, all algorithms show good recovery performance, whereas compact-SCAphase and SCAphase exhibit faster convergence. Moreover, compared to SCAphase, compact-SCAphase uses half the number of iterations to attain a stationary point. However, the reduction of computation time achieved by compact-SCAphase is not as significant as the reduced number of iterations because, as discussed in Section 5.5, compact-SCAphase has the highest per-iteration complexity. In contrast, when the number of users is comparable to that of the antennas, i.e., $P = N$, only compact-SCAphase achieves the given tolerance within 2000 iterations. This is intuitive as in the regime of $P \geq N$, and with sparse channel access, the information of the users' channels contained in the measurements is insufficient. To resolve this challenge, a higher spatial oversampling rate is required. Nevertheless, compared to SC-PRIME, compact-SCAphase and SCAphase show a significant improvement in estimation accuracy. Then, compared to SCAphase, compact-SCAphase further improves the estimation quality of $\boldsymbol{Z}$ due to fast convergence.

Varying number of initializations: In the second simulation, we investigate the robustness of the algorithms to initialization. The performance behavior of the algorithms with the number of random initializations varied from 1 to 50 is presented in Figure 5.5. The number of users P and density are set to be $\{N/2, N\}$ and $\{0.05, 0.1\}$, respectively. In Figure 5.5, for most choices of $\{P, L/P\}$, all algorithms show similar robustness to initialization as the estimation quality achieved by each algorithm remains constant for the number of random initializations exceeding 10, and compact-SCAphase possesses the lowest estimation errors. In the cases with $P = N$, SC-PRIME shows a significant degradation in the estimation quality compared to the proposed algorithms, which, as

demonstrated in Figure 5.4, results from the fact that SC-PRIME generally does not converge within the limit of 2000 iterations. Additionally, if only the spatial signature $\boldsymbol{D}$ needs to be recovered, then 5 initializations are sufficient for all algorithms to attain a good estimation accuracy. Particularly, when $P = N/2$, compact-SCAphase achieves a good stationary point for $\boldsymbol{D}$ even with a single initialization.

5.6.3.2 Case 2 – Time-invariant spatial mixing and STFT temporal mixing

Next, a temporal mixing network that performs the same STFT independently on each output channel of the spatial mixing network is introduced (see [JEH16] for more details of the STFT measurement model). For the STFT, we use an I-point DFT, a rectangular window of length $I/2$, and a hop size of $I/4$. The above parameter setup results in a temporal oversampling rate of 5. Similar to the previous simulation, in Figure 5.6, the estimation accuracy of the algorithms is evaluated as a function of the number of initializations. We set $P = N/2$ and density $L/P = \{0.05, 0.1\}$. The regularization parameters are chosen to be $\lambda = 0.75^{25}\lambda_{\max}$, $\mu = \sigma^2_{\min,\mathrm{nz}}(\boldsymbol{F}) = \sigma^2_{\min,\mathrm{nz}}(\boldsymbol{A})\sigma^2_{\min,\mathrm{nz}}(\boldsymbol{B})$, and $\rho = 0.75^{28}\rho_{\max}$ and $\rho = 0.75^{23}\rho_{\max}$ for SCAphase and SC-PRIME, respectively, with $\rho_{\max}$ in (5.56).

Comparing the results in Figure 5.5 and 5.6, we observe that, given a sufficient number of initializations, the estimation qualities are significantly improved in the case with STFT temporal mixing due to the increase of the overall sampling rate. However, all algorithms become less robust to initialization as compared to Case 1. In particular, compact-SCAphase and SCAphase require 20 initializations to attain a good stationary point, whereas SC-PRIME cannot achieve the same estimation accuracy as the other algorithms even with 50 initializations since, as we discussed, SC-PRIME does not converge within 2000 iterations.

5.6.3.3 Case 3 – Time-variant spatial mixing and no temporal mixing

As discussed in Section 5.5, compared to the other two algorithms, compact-SCAphase has a per-iteration complexity of a higher order in the general case with a linear measurement operator $\mathcal{F}$ in (5.3) with multiple chains of mixing networks. Therefore, in the case of time-variant spatial mixing, we only compare SCAphase with SC-PRIME, as the running time of compact-SCAphase is unaffordable. As depicted in Figures 5.7 and 5.8, respectively, the accuracy and complexity of the algorithms are evaluated for

various choices of $\{P, L/P\}$. All parameters are the same as in Figure 5.3, except that the spatial mixing $\boldsymbol{A}_i$ for each snapshot is generated independently.

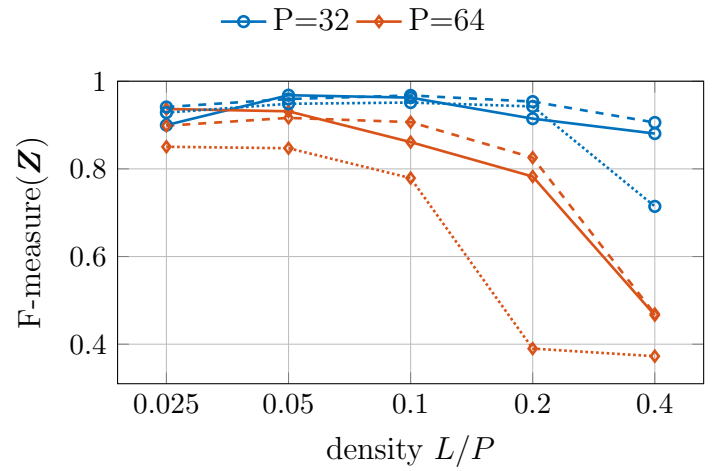
What stands out in Figure 5.7 is that the use of time-variant spatial mixing overcomes the challenge of lack of diversity in the extremely sparse case observed in Figure 5.3. On the other hand, the convergence rates of the two algorithms measured by the number of iterations in Figure 5.8 are similar to that in Figure 5.4. However, due to the increased complexity of the linear operator $\mathcal{F}$, the two algorithms possess similar per-iteration complexity. Hence, compared to SC-PRIME, SCAphase exhibits a significantly improved convergence rate in terms of both the number of iterations and computational time, when $P = N/2$.

Finally, we summarize the performance of the three considered cases. Comparing the two cases without temporal mixing, i.e., Cases 1 and 3, we observe that the use of time-variant spatial mixing in Case 3 overcomes the challenge of lack of diversity observed in Case 1 in the extremely sparse case and results in a better estimation quality. On the other hand, in the case without temporal mixing, the signal in each time slot is measured independently and, hence, from the magnitude-only measurements, the signals can only be recovered up to a global phase ambiguity for each time slot. Thus, the temporal mixing, which is applied in Case 2, is introduced to further recover the relative phase between the signals in different time slots.

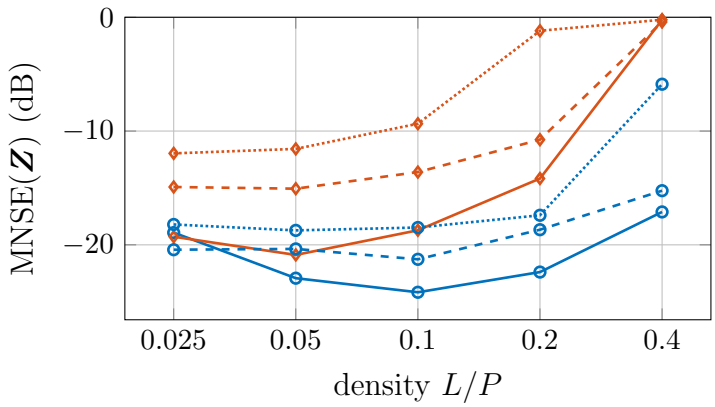
5.7 Summary

In this chapter, we employ the smoothing SCA framework in Section 4.2 to address the phase retrieval with dictionary learning problem. Two efficient parallel algorithms are proposed by applying the smoothing SCA to two complementary formulations, respectively. The first algorithm, termed *compact-SCAphase*, employs a compact ℓ_1-regularized nonconvex LS formulation, which avoids the auxiliary variables required in state-of-the-art methods such as SC-PRIME and DOLPHIn. The second algorithm, denoted by *SCAphase*, solves the conventional formulation as in SC-PRIME. An efficient procedure based on rational approximation is devised for solving the ℓ_2-norm constrained LS subproblems under the SCA framework. For both algorithms, we refined the search range for suitable values of the sparsity parameter. Simulation results on synthetic data in the context of blind channel estimation in a multi-antenna random access network demonstrate the fast convergence of SCAphase compared to SC-PRIME. Moreover, compact-SCAphase is more competitive than SCAphase in terms of both computational complexity and parameter tuning cost in the case with less diverse

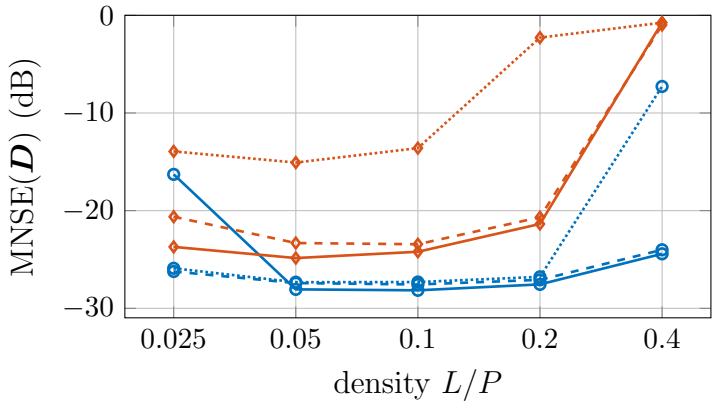
linear measurement operators. Nevertheless, SCAphase also has several advantages over compact-SCAphase. Compared to SCAphase, the computational complexity of compact-SCAphase dramatically grows with the increase of diversity of the designed linear measurement operator. Also, SCAphase can easily include potential side constraints on the signal of interest that are expected to restrict the ambiguities in the estimation. For instance, in X-ray crystallography, the unknown signal represents the electron density in the crystal, which is certainly nonnegative and bounded [Mil90, Har93].



(a)

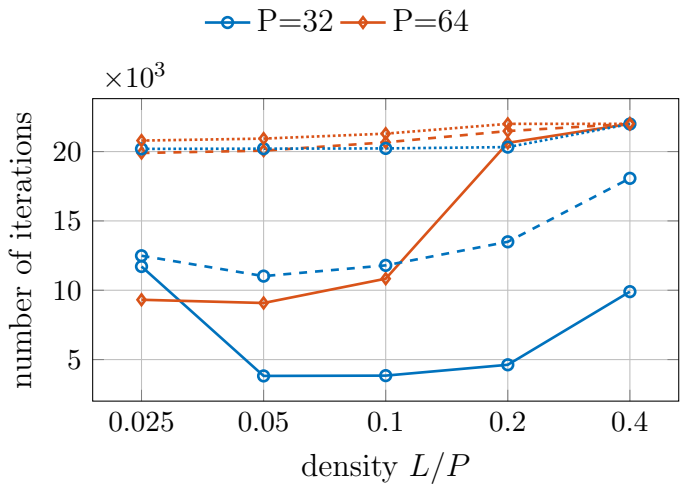


(b)

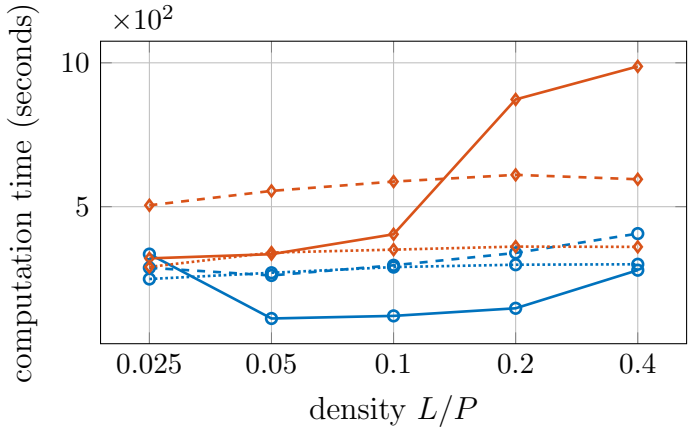


(c)

Figure 5.3. Estimation quality vs. density L/P using compact-SCAphase (solid), SCAphase (dashed), and SC-PRIME (dotted) in Case 1 with $N = 64, M_1 = 4N, I = 16N$.



(a)



(b)

Figure 5.4. Computational complexity vs. density L/P using compact-SCAphase (solid), SCAphase (dashed), and SC-PRIME (dotted) in Case 1 with $N = 64, M_1 = 4N, I = 16N$.

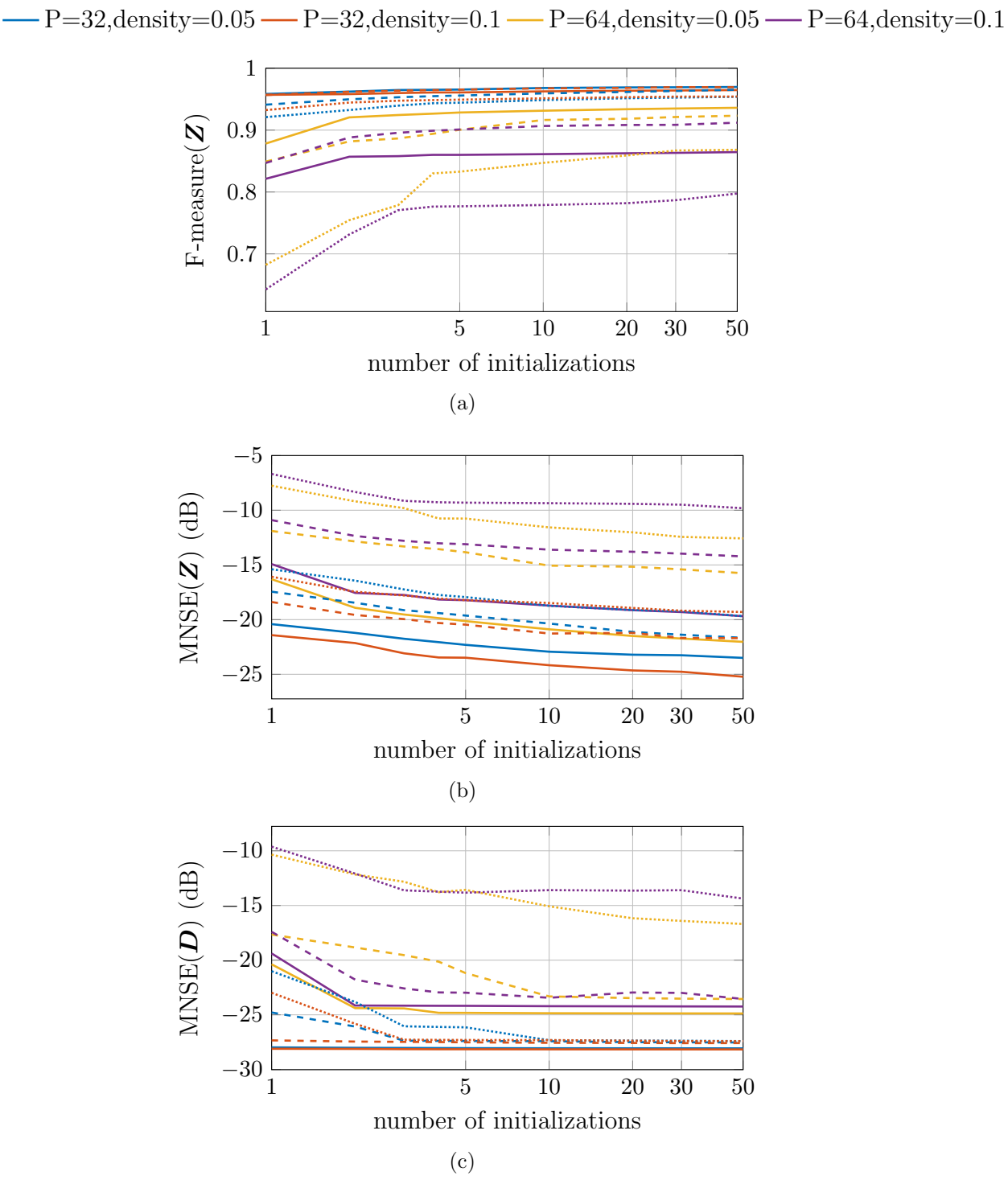


Figure 5.5. Estimation quality vs. number of initializations using compact-SCAphase (solid), SCAphase (dashed), and SC-PRIME (dotted) in Case 1 with $N = 64$, $M_1 = 4N$, $I = 16N$.

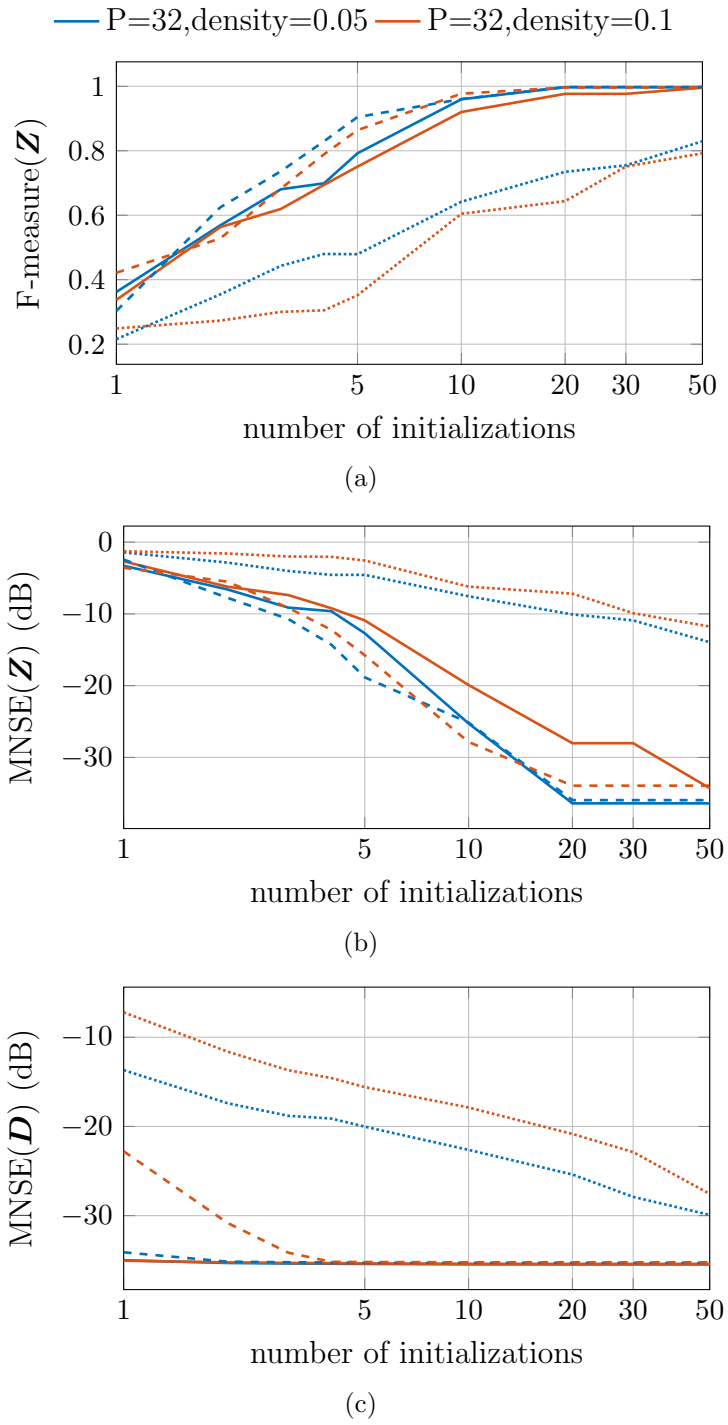


Figure 5.6. Estimation quality vs. number of initializations using compact-SCAphase (solid), SCAphase (dashed), and SC-PRIME (dotted) in Case 2 with $N = 64$, $M_1 = 4N$, $I = 16N$.

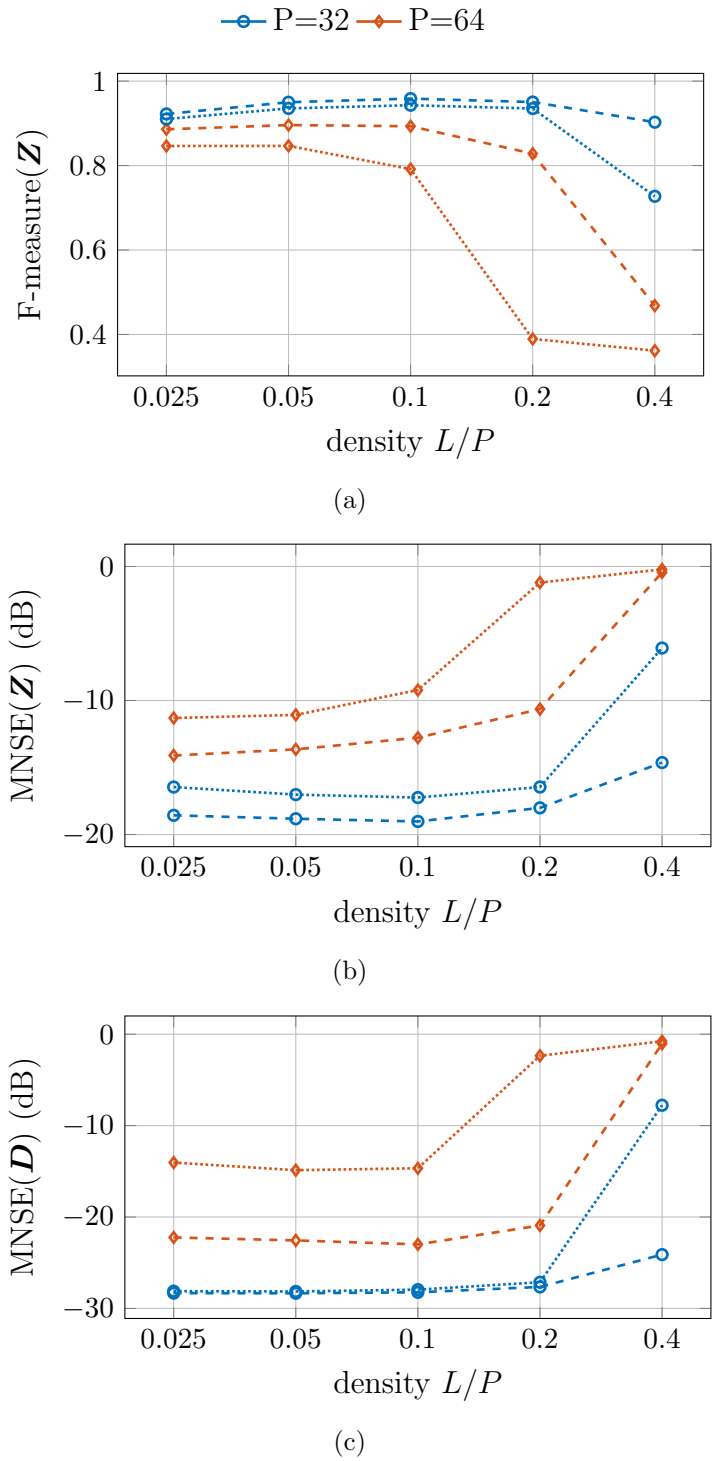
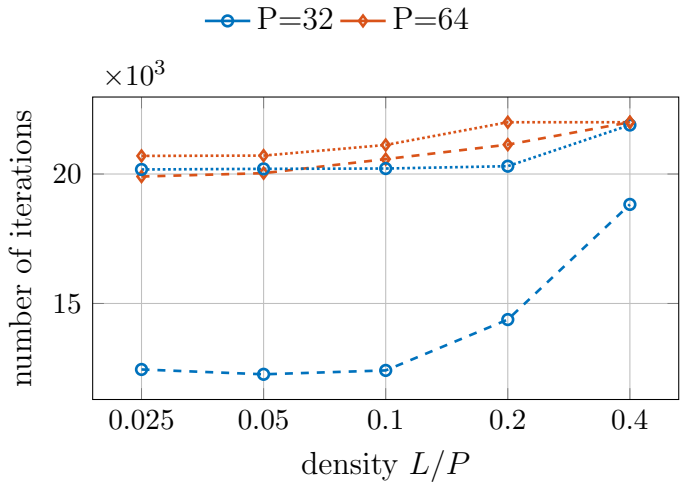
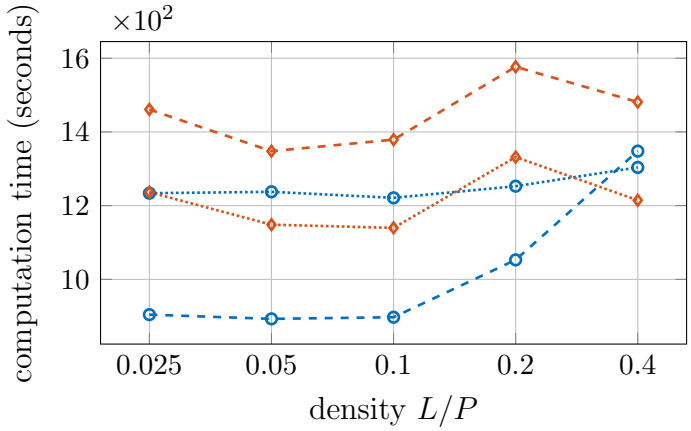


Figure 5.7. Estimation quality vs. density L/P using SCAphase (dashed) and SC-PRIME (dotted) in Case 3 with $N = 64$, $M_1 = 4N$, $I = 16N$.



(a)



(b)

Figure 5.8. Computational complexity vs. density L/P using SCAphase (dashed) and SC-PRIME (dotted) in Case 3 with $N = 64,\ M_1 = 4N,\ I = 16N$.

Chapter 6
Conclusions and Outlook

This dissertation concentrates on the design and analysis of approximation-based methods for nonconvex nonsmooth optimization problems. The main idea behind those methods is to solve a difficult optimization problem by converting it into a sequence of simpler surrogate/approximate problems. The existing approximation-based methods differ mainly in the construction of the approximate problems. Specifically, in the two related existing optimization frameworks, namely, the majorization-minimization (MM) framework and the successive convex approximation (SCA) framework, the approximate function is designed to be a global upper bound, called majorizer, of the original objective function and a convex function, respectively. Generally speaking, there are two desiderata of the approximate function, namely, the tightness to the original objective function and the low computational complexity of minimizing the approximate function. In particular, we focus on constructing an approximate problem that can be solved in a parallel or distributed manner so as to take advantage of modern multicore computing platforms.

In the first part of this thesis, we develop an efficient parallelizable approximation-based algorithmic framework for a broad class of nonconvex nonsmooth optimization problems. The classic MM framework requires the directional differentiability of the objective function and the consistency of directional derivatives in all directions between the original objective function and its majorizer at the point where the majorizer is constructed. This condition restricts the majorizer constructed at a nondifferentiable point of the original function to be also nonsmooth, which hinders its capability of simplifying nonsmooth problems since the minimization of the majorizing function, if restricted to be nonsmooth, may still be difficult. Therefore, in Chapter 4, we relax the derivative consistency in the majorization step so that a smooth majorizer that can be easily minimized is permitted for a wide class of nonsmooth problems. Specifically, as a generalization of the majorization technique that we employed in [LTY+22], we consider the situation that the majorizing function preserves only a subgradient of the original objective function. The MM framework with such smoothing majorization, abbreviated as *smoothing MM*, is detailed in Section 4.1, together with its convergence analysis. As a result of this relaxation of derivative consistency, the smoothing MM converges to a stationary point in a more relaxed sense than the classic MM. In other words, compared to the classic MM, the smoothing MM sacrifices the tightness of the convergence set with respect to the local minima in order to construct an approximate problem that can be easily addressed. In some scenarios, the exact minimization

of the smooth majorizing function may still be difficult, especially if it is nonconvex. Meanwhile, the smoothness of the majorizing function allows us to employ the idea of SCA, along with the available separable convex approximation techniques, to obtain an approximate minimizer of the majorizing function efficiently. This motivated our idea in [LTY+22] of combining the smoothing majorization and the separable convex approximation techniques to address the phase retrieval with dictionary learning problem. In Section 4.2, we generalize the algorithms in [LTY+22] to the aforementioned class of smooth majorization techniques to develop an inexact MM framework, named *smoothing SCA*, and provide a unified convergence analysis. Finally, similar to the classic MM and SCA frameworks, the smoothing MM and SCA can also be implemented in a BCD manner to exploit potential separable structures of the constraints in the optimization problem. The block-coordinatewise versions of the smoothing MM and SCA, as well as their convergence analyses, are presented in Section 4.3.

In the second part of this thesis, i.e., Chapter 5, as our mainly promoted framework, the smoothing SCA framework is employed to address the phase retrieval with dictionary learning problem. Whereas phase retrieval aims at recovering unknown signals from magnitude measurements of linear mixtures, the phase retrieval with dictionary learning problem includes other prior information that the signal admits a sparse representation over an unknown dictionary. The task is to jointly estimate the dictionary and the sparse representation from magnitude-only measurements. Two efficient parallel algorithms are developed by applying the smoothing SCA to two complementary nonconvex nonsmooth formulations, respectively, which are both based on a least-squares (LS) criterion. The first algorithm, termed *compact-SCAphase*, employs a compact ℓ_1-regularized nonconvex least-squares (LS) formulation, which avoids the auxiliary variables required in state-of-the-art methods such as SC-PRIME [QP17] and DOLPHIn [TEM16]. The second algorithm, denoted by *SCAphase*, solves the conventional formulation as in SC-PRIME. An efficient procedure based on rational approximation is devised for solving the ℓ_2-norm constrained LS subproblems under the SCA framework. For both algorithms, we refined the search range for suitable values of the sparsity parameter. Simulation results on synthetic data in the context of blind channel estimation in a multi-antenna random access network demonstrate the fast convergence of SCAphase compared to SC-PRIME. Moreover, compact-SCAphase is more competitive than SCAphase in terms of both computational complexity and parameter tuning cost in the case with less diverse linear measurement operators. Nevertheless, SCAphase also has several advantages over compact-SCAphase. Compared to SCAphase, the computational complexity of compact-SCAphase dramatically grows with the increase of diversity of the designed linear measurement operator. Also, SCAphase can easily include potential side constraints on the signal of interest.

Possible extensions of this work and open problems are listed below.

On the optimization framework aspect:

- In Section 4.3, the convergences of the block-coordinatewise versions of our developed frameworks, i.e., the smoothing BSUM and the smoothing BSCA, are only demonstrated in the case where the block variables are updated in a fixed-order. In practice, using other types of block selection rules, such as a greedy rule with an appropriate selection criterion, can lead to significantly faster convergence for a BCD-type method. However, the convergences of the proposed block-coordinatewise frameworks with such block selection rules require further investigation.
- The parallelizability of the proposed smoothing SCA framework, as well as its block-coordinatewise version, relies on separable structures of the constraints. It is still unclear how we can generalize the smoothing SCA framework to problems where the variables are coupled in the constraints. A common approach for problems with coupling constraints is the primal-dual method that alternately minimizes the Lagrangian in the primal space and updates the Lagrangian multipliers in the dual space. As can be easily seen, this approach results in double-loop algorithms that are typically slow.

On the application aspect:

- Several questions that have been answered for the classic phase retrieval remain open for phase retrieval with dictionary learning. First, further work needs to be done to establish the theoretical conditions for a guaranteed unique recovery (up to trivial ambiguities) of the dictionary and/or the sparse codes.
- Performance bounds, such as the Cramér-Rao bound, have been derived for the classic phase retrieval in diverse scenarios to assess the achievable estimation performance. However, further investigations are required to derive the corresponding Cramér-Rao bound for the phase retrieval with dictionary learning in the case where the parameters, i.e., the dictionary and the sparse codes, are identifiable.
- The simulation results in Section 5.6 show that multiple random initializations are required for attaining (near-)global minima of our nonconvex formulations. Hence, it is of great interest to develop a more sophisticated initialization strategy that can help avoid poor stationary points.

- As regularization-based methods, the choice of the regularization parameters has a strong influence on the performance of the proposed methods. The SCA framework has been studied in the literature as a basis for deep unrolling [LVS23]. Thus, as mentioned in Section 5.6, in the case where training data are available, it is of interest to unroll the proposed algorithms into deep networks so that suitable values of the regularization parameters can be more efficiently learned by state-of-the-art deep learning methods.

Chapter A
Appendix

A.1 Solution Approach for Subproblem (5.15)

A.1.1 Solution Approach for a General Linear Operator $\mathcal{F}$

In the following, we derive the solution approach for the subproblem (5.15) that involves vector $\boldsymbol{d}_p$ by solving the KKT optimality system, since the strong duality holds for (5.15). We first consider a general linear operator $\mathcal{F}$ and then the particular structure of $\mathcal{F}$ in (5.47) is exploited to further reduce the complexity of computing the solution for $\boldsymbol{d}_p$ in this special case.

The gradient of $L(\boldsymbol{d}_p, \nu_p)$ in (5.17) with respect to $\boldsymbol{d}_p$ is

$$\nabla_{\boldsymbol{d}_p} L(\boldsymbol{d}_p, \nu_p) = \left(\boldsymbol{H}_p^{\mathsf{H}} \boldsymbol{H}_p + \nu_p \boldsymbol{I}_N\right) \boldsymbol{d}_p - \boldsymbol{H}_p^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}_p^{(t)}).$$

Then the primal and dual optimal solutions must satisfy the following KKT system:

$$\nabla_{\boldsymbol{d}_p} L(\boldsymbol{d}_p, \nu_p) = \boldsymbol{0}, \qquad \text{(stationarity)} \tag{A.1a}$$

$$\|\boldsymbol{d}_p\|_2^2 \leq 1,\ \nu_p \geq 0, \qquad \text{(primal and dual feasibility)} \tag{A.1b}$$

$$\nu_p(\|\boldsymbol{d}_p\|_2^2 - 1) = 0. \qquad \text{(complementary slackness)} \tag{A.1c}$$

For notational simplicity, we omit the iteration index t and let $\widetilde{\boldsymbol{d}}_p$ and $\widetilde{\nu}_p$ denote any pair of primal and dual optimal solutions in the derivations below. Two mutually exclusive possibilities arise due to (A.1b): a) $\widetilde{\nu}_p = 0$, or b) $\widetilde{\nu}_p > 0$.

Given dual variable $\nu_p = 0$, the solution $\boldsymbol{d}_p^{(\nu_p=0)}$ of (A.1a) corresponds to a solution of problem (5.15) when the ℓ_2-norm constraint is ignored. It is expressed as follows:

$$\boldsymbol{d}_p^{(\nu_p=0)} = (\boldsymbol{H}_p^{\mathsf{H}} \boldsymbol{H}_p)^{\dagger} \boldsymbol{H}_p^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}_p^{(t)}). \tag{A.2}$$

If $\boldsymbol{H}_p$ has full column rank, i.e., $\operatorname{rank}(\boldsymbol{H}_p) = N$, $\boldsymbol{d}_p^{(\nu_p=0)}$ is the unique solution of (A.1a). Otherwise, $\boldsymbol{d}_p^{(\nu_p=0)}$ is the solution with minimum ℓ_2-norm. In both cases above, if $\|\boldsymbol{d}^{(\nu_p=0)}\|_2 \leq 1$, then all conditions in (A.1) are fulfilled and, consequently, $\widetilde{\nu}_p = 0$ and $\widetilde{\boldsymbol{d}}_p = \boldsymbol{d}_p^{(\nu_p=0)}$ is a pair of optimal solutions.

However, if $\boldsymbol{d}^{(\nu_p=0)}$ does not satisfy the primal constraint, then $\widetilde{\nu}_p > 0$ must hold. Let $\boldsymbol{H}_p = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$ be the compact SVD of $\boldsymbol{H}_p$ and $\sigma_1 \geq \ldots \geq \sigma_r > 0$ be the nonzero

singular values with $r = \operatorname{rank}(\boldsymbol{H}_p)$, $\boldsymbol{U} \in \mathbb{C}^{M_1 M_2 \times r}$, $\boldsymbol{\Sigma} \in \mathbb{D}_+^r$, $\boldsymbol{V} \in \mathbb{C}^{N \times r}$. Then, for $\nu_p > 0$, the stationarity condition (A.1a) can be rewritten as

$$\boldsymbol{d}_p = \boldsymbol{V}(\boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{\Sigma} + \nu_p \boldsymbol{I}_N)^{-1}\boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{U}^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}_p^{(t)}). \tag{A.3}$$

Thus, the squared ℓ_2-norm of $\boldsymbol{d}_p$ that satisfies condition (A.3) is a function of ν_p, which we define as $\psi_p(\nu_p) = \|\boldsymbol{d}_p\|_2^2$. Defining

$$\boldsymbol{c}_p = \boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{U}^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}_p^{(t)}), \tag{A.4}$$

function $\psi_p(\nu_p)$ can be explicitly written as a rational function:

$$\psi_p(\nu_p) = \sum_{i=1}^{r} \frac{|c_{i,p}|^2}{(\sigma_i^2 + \nu_p)^2}, \quad \text{for } \nu_p > 0, \tag{A.5}$$

where $c_{i,p}$ is the ith element of vector $\boldsymbol{c}_p$. Although (A.5) is derived for $\nu_p > 0$, the expression also holds for $\nu_p = 0$, i.e., $\psi_p(0) = \|\boldsymbol{d}_p^{(\widetilde{\nu}_p=0)}\|_2^2$, which is assumed to be above one in the present case b). Then, due to (A.1c), $\widetilde{\nu}_p$ coincides with the unique solution of $\psi_p(\nu_p) = 1$ in $(0, +\infty)$. Consequently, the optimal solution $\widetilde{\boldsymbol{d}}_p$ is obtained by substituting $\widetilde{\nu}_p$ into (A.3).

In conclusion, for $p = 1, \ldots, P$, the dual optimal points $\widetilde{\nu}_p$ are determined independently by the following rule:

$$\begin{cases} \widetilde{\nu}_p = 0, & \text{if } \psi_p(0) \leq 1, \\ \widetilde{\nu}_p \in \{\nu_p > 0 \mid \psi_p(\nu_p) = 1\}, & \text{otherwise.} \end{cases}$$

Then, combining (A.2) and (A.3) and exploiting the SVD of matrix $\boldsymbol{H}_p$ to further reduce the complexity, we express the optimal solution $\widetilde{\boldsymbol{d}}_p$ to problem (5.15) as

$$\widetilde{\boldsymbol{d}}_p = \boldsymbol{V}(\boldsymbol{\Sigma}^{\mathsf{H}}\boldsymbol{\Sigma} + \widetilde{\nu}_p \boldsymbol{I}_r)^{\dagger}\boldsymbol{c}_p.$$

A.1.2 Simplified Solution Approach for $\mathcal{F}$ in (5.47)

For the linear operator $\mathcal{F}$ in (5.47), the matrix $\boldsymbol{F}$ in the vectorized form is

$$\boldsymbol{F} = \boldsymbol{B}^{\mathsf{T}} \otimes \boldsymbol{A}, \tag{A.6}$$

and then, we have

$$\boldsymbol{H}_p = \left(\boldsymbol{B}^{\mathsf{T}} \otimes \boldsymbol{A}\right) \cdot \left(\boldsymbol{z}_{p:}^{(t)} \otimes \boldsymbol{I}_N\right) = \left(\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)}\right) \otimes \boldsymbol{A}.$$

Let $\boldsymbol{A} = \boldsymbol{U}_A \boldsymbol{\Sigma}_A \boldsymbol{V}_A^{\mathsf{H}}$ and $\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)} = \boldsymbol{U}_B \boldsymbol{\Sigma}_B \boldsymbol{V}_B^{\mathsf{H}}$ be the compact SVDs of $\boldsymbol{A}$ and $\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)}$, respectively. The compact SVD of $\boldsymbol{H}_p$ can be analytically calculated as [GVL13]:

$$\boldsymbol{H}_p = \underbrace{(\boldsymbol{U}_B \otimes \boldsymbol{U}_A)}_{\boldsymbol{U}} \underbrace{(\boldsymbol{\Sigma}_B \otimes \boldsymbol{\Sigma}_A)}_{\boldsymbol{\Sigma}} \underbrace{(\boldsymbol{V}_B \otimes \boldsymbol{V}_A)^{\mathsf{H}}}_{\boldsymbol{V}^{\mathsf{H}}}.$$

As a column vector, $\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)}$ has $\boldsymbol{V}_B = 1$ and only one nonzero singular value $\left\| \boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)} \right\|_2$. Thus, we have

$$\boldsymbol{U}\boldsymbol{\Sigma} = (\boldsymbol{U}_B \boldsymbol{\Sigma}_B) \otimes (\boldsymbol{U}_A \boldsymbol{\Sigma}_A) = \left(\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)}\right) \otimes (\boldsymbol{U}_A \boldsymbol{\Sigma}_A),$$

and the nonzero singular values of $\boldsymbol{H}_p$ are given by

$$\left\| \boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)} \right\|_2 \cdot \sigma_i^A \quad \text{for } i = 1, \ldots, r,$$

where $\{\sigma_i^A\}_{i=1}^r$ are the nonzero singular values of $\boldsymbol{A}$ and $r = \operatorname{rank}(\boldsymbol{H}_p) = \operatorname{rank}(\boldsymbol{A})$. Consequently, vector $\boldsymbol{c}_p$ in (5.19) can be written as

$$\boldsymbol{c}_p = \left(\boldsymbol{B}^{\mathsf{T}} \boldsymbol{z}_{p:}^{(t)}\right)^{\mathsf{H}} \otimes (\boldsymbol{U}_A \boldsymbol{\Sigma}_A)^{\mathsf{H}} \cdot \operatorname{vec}\left(\boldsymbol{Y}_p^{(t)}\right) = \boldsymbol{\Sigma}_A^{\mathsf{H}} \boldsymbol{U}_A^{\mathsf{H}} \boldsymbol{Y}_p^{(t)} \boldsymbol{B}^{\mathsf{H}} \bar{\boldsymbol{z}}_{p:}^{(t)}.$$

Finally, after having obtained the dual optimal solution $\widetilde{\nu}_p$ by the same procedure as described in the general case, we can also compute the optimal solution $\widetilde{\boldsymbol{d}}_p$ using simply the SVD of $\boldsymbol{A}$:

$$\widetilde{\boldsymbol{d}}_p = \boldsymbol{V}_A \left(\boldsymbol{\Sigma}_A^{\mathsf{H}} \boldsymbol{\Sigma}_A + \widetilde{\nu}_p \boldsymbol{I}_r\right)^{\dagger} \boldsymbol{c}_p,$$

and avoids the evaluation of the SVD of $\boldsymbol{H}_p$ that is required in the general case.

A.2 Proof of Theorem 5.1

The original rational function $\psi(\nu)$ in (5.19) and its derivative $\psi'(\nu)$ can be rewritten as

$$\psi(\nu) = \sum_{i=1}^{r} \frac{|c_i|^2}{(\delta_i - \nu)^2} \quad \text{and} \quad \psi'(\nu) = \sum_{i=1}^{r} \frac{2|c_i|^2}{(\delta_i - \nu)^3}$$

with the poles $\delta_1 \leq \cdots \leq \delta_r < 0$. We ignore the trivial case where all poles δ_i are identical. Define $\zeta(\nu) = F(\nu; \alpha, \beta) - \psi(\nu)$ with the approximate function F defined in (5.26). It is sufficient to show that $\zeta(\nu) < 0$ for all $\nu > \delta_r$ and $\nu \neq \nu^{(l)}$. To this end, define

$$\xi(\nu) = \zeta(\nu)(\beta - \nu)^2 \prod_{i=1}^{r} (\delta_i - \nu)^2.$$

Then ξ is a polynomial of degree $2r$ with real coefficients:

$$\xi(\nu) = \alpha \prod_{i=1}^{r} (\delta_i - \nu)^2 - (\beta - \nu)^2 \sum_{i=1}^{r} |c_i|^2 \prod_{j=1,\, j \neq i}^{r} (\delta_j - \nu)^2. \tag{A.7}$$

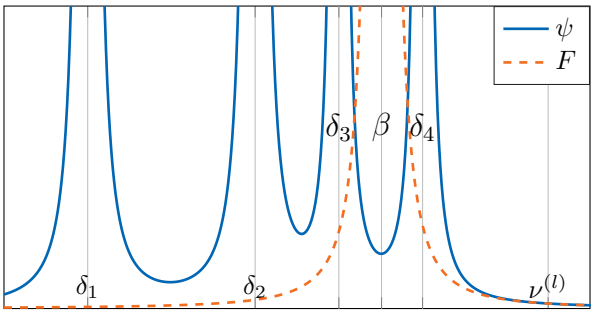


Figure A.1. Original and approximate rational functions, $r = 4$.

The product rule for differentiation determines that $\xi(\nu^{(l)}) = 0$ and its derivative $\xi'(\nu^{(l)}) = 0$ since $\zeta(\nu^{(l)}) = 0$ and its derivative $\zeta'(\nu^{(l)}) = 0$. Hence, $\nu^{(l)}$ is a double root of ξ, and we can extract the factor $(\nu - \nu^{(l)})^2$ and rewrite (A.7) as

$$\xi(\nu) = \left(\alpha - \sum_{i=1}^{r} |c_i|^2\right) \left(\nu - \nu^{(l)}\right)^2 \prod_{i=1}^{r-1} (\nu^2 - 2a_i\nu + b_i)$$

with appropriately chosen coefficients $a_i, b_i \in \mathbb{R}$.

We claim that $\nu^{(l)}$ is the only real double root of ξ in $(\delta_r, +\infty)$. To see this, observe from (5.27) that the pole of F

$$\beta = \frac{2}{\psi'(\nu^{(l)})} \sum_{i=1}^{r} \frac{\delta_i |c_i|^2}{(\delta_i - \nu^{(l)})^3} \in (\delta_1, \delta_r).$$

The roots of ζ are also the roots of ξ. The following result can be intuitively observed from Fig. A.1. Each interval (δ_i, δ_{i+1}) with $\delta_i \neq \delta_{i+1}$ contains either two real roots of ζ or the real part of a pair of complex conjugate roots. In contrast, if $\delta_i = \delta_{i+1}$ for some $i = 1, \ldots, r-1$, it can be trivially identified from (A.7) that δ_i is a double root of ξ. Hence, the real parts of the remaining $2r - 2$ roots of ξ fall in the interval $[\delta_1, \delta_r]$. The claim is established; it can be proved more formally by factorizing (A.7).

This argument shows that $\operatorname{sign}(\xi(\nu))$ remains constant in $[\delta_r, \nu^{(l)}) \cup (\nu^{(l)}, +\infty)$. Therefore, it follows from (A.7) that, for all $\nu > \delta_r$ and $\nu \neq \nu^{(l)}$,

$$\operatorname{sign}(\zeta(\nu)) = \operatorname{sign}(\xi(\nu)) = \operatorname{sign}(\xi(\delta_r)) = \operatorname{sign}\left(-(\beta - \delta_r)^2 \prod_{i=1}^{r-1} (\delta_i - \delta_r)^2\right) = -1.$$

This implies that $F(\nu; \alpha, \beta) < \psi(\nu)$ for all $\nu > \delta_r$ and $\nu \neq \nu^{(l)}$. □

A.3 Derivation of Upper Bound $\lambda_{\max}$

We derive the upper bound $\lambda_{\max}$ for the sparsity parameter λ in (5.5) using the stationarity conditions (5.31) with the gradients in (5.11). Condition (5.31a) is trivial for $\boldsymbol{Z} = \boldsymbol{0}$ as $\nabla_{\boldsymbol{D}} \widehat{f}(\boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) = \boldsymbol{0}$ for any $\boldsymbol{D}$. Then, adopting the vectorized form in (5.4) for $\mathcal{F}$ and the partition in (5.23), we can write the gradient $\nabla_{z_{p,i}} \widehat{f}$ at $\boldsymbol{Z} = \boldsymbol{0}$ as $\nabla_{z_{p,i}} \widehat{f}(\boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) = -\boldsymbol{d}_p^{\mathsf{H}} \boldsymbol{F}_i^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)})$. It follows that

$$\begin{aligned}
\left| \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) \right| &= \left| \boldsymbol{d}_p^{\mathsf{H}} \boldsymbol{F}_i^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)}) \right| \\
&\leq \|\boldsymbol{d}_p\|_2 \left\| \boldsymbol{F}_i^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)}) \right\|_2 && \text{(A.8a)} \\
&\leq \left\| \boldsymbol{F}_i^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)}) \right\|_2 && \text{(A.8b)} \\
&\leq \|\boldsymbol{F}_i\|_2 \|\boldsymbol{Y}\|_{\mathsf{F}}, && \text{(A.8c)}
\end{aligned}$$

where (A.8a) comes from the Cauchy–Schwartz inequality, (A.8b) from the constraint of problem (5.5), and (A.8c) from the definition of the operator norm. The matrix ℓ_2-norm $\|\boldsymbol{F}_i\|_2$ is equal to the largest singular value of $\boldsymbol{F}_i$, denoted by $\sigma_{\max}(\boldsymbol{F}_i)$. Inequality (A.8) holds for any solution with $\boldsymbol{Z} = \boldsymbol{0}$. Consequently, comparing (A.8) with (5.31b) yields the following result. Define

$$\lambda_{\max} = \|\boldsymbol{Y}\|_{\mathsf{F}} \cdot \max_{i=1,\dots,I} \{\sigma_{\max}(\boldsymbol{F}_i)\}.$$

For $\lambda \geq \lambda_{\max}$, any point $(\boldsymbol{D}, \boldsymbol{0})$ with $\boldsymbol{D} \in \mathcal{D}$ satisfies the conditions (5.31) and, therefore, is stationary for $\widehat{h}$ in the domain of problem (5.5). Note that $\lambda_{\max}$ above does not depend on the point $\boldsymbol{S}^{(t)}$ where the majorization is made. Hence, $(\boldsymbol{D}, \boldsymbol{0})$ is stationary for $\widehat{h}$ taken at any point, including $(\boldsymbol{D}, \boldsymbol{0})$. This implies that, for $\lambda \geq \lambda_{\max}$, any point $(\boldsymbol{D}, \boldsymbol{0})$ is stationary for the original problem (5.5). Also, it is easy to verify that all points $(\boldsymbol{D}, \boldsymbol{0})$ with $\boldsymbol{D} \in \mathcal{D}$ are equally optimal for both $\widehat{h}$ and h.

In addition, $\lambda_{\max}$ can be further decreased in the investigated cases 1 and 2 in Section 5.6, where the linear operator $\mathcal{F}$ is given by (5.47). In this case, we have, for $i = 1, \dots, I$,

$$\boldsymbol{F}_i = \boldsymbol{b}_{i:} \otimes \boldsymbol{A} \quad \text{and} \quad \boldsymbol{F}_i^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)}) = \boldsymbol{A}^{\mathsf{H}} \boldsymbol{Y}^{(t)} \bar{\boldsymbol{b}}_{i:}. \tag{A.9}$$

Then, directly substituting $\boldsymbol{F}_i$ in (A.9) into (A.8c), we obtain

$$\left| \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) \right| \leq \|\boldsymbol{A}\|_2 \|\boldsymbol{b}_{i:}\|_2 \|\boldsymbol{Y}\|_{\mathsf{F}}. \tag{A.10}$$

On the other hand, exploiting the structure of $\boldsymbol{F}_i$ in (A.9), we can further derive the following inequality from (A.8b):

$$\begin{aligned}
\left| \nabla_{z_{p,i}} \widehat{f}(\boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) \right| &\leq \left\| \boldsymbol{A}^{\mathsf{H}} \boldsymbol{Y}^{(t)} \bar{\boldsymbol{b}}_{i:} \right\|_2 \leq \|\boldsymbol{A}\|_2 \|\boldsymbol{Y}^{(t)} \bar{\boldsymbol{b}}_{i:}\|_2 \\
&= \|\boldsymbol{A}\|_2 \cdot \left\| \sum_{m=1}^{M_2} \bar{b}_{i,m} \boldsymbol{y}_m^{(t)} \right\|_2 \leq \|\boldsymbol{A}\|_2 \cdot \sum_{m=1}^{M_2} |b_{i,m}| \cdot \|\boldsymbol{y}_m\|_2. && \text{(A.11)}
\end{aligned}$$

It is shown by Cauchy–Schwartz inequality that (A.11) is a tighter bound for $\nabla_{z_{p,i}}\widehat{f}$ than (A.10). Consequently, in the case with $\mathcal{F}(\boldsymbol{X}) = \boldsymbol{A}\boldsymbol{X}\boldsymbol{B}$, the upper bound $\lambda_{\max}$ can be decreased to

$$\lambda_{\max} = \sigma_{\max}(\boldsymbol{A}) \cdot \max_{i=1,\dots,I} \left\{ \sum_{m=1}^{M_2} |b_{i,m}| \cdot \|\boldsymbol{y}_m\|_2 \right\}.$$

Furthermore, in Case 3 in Section 5.6, where spatial mixing is time-variant and temporal mixing is not applied, we have

$$\boldsymbol{F}_i = \boldsymbol{e}_i \otimes \boldsymbol{A}_i$$

from the vectorized form in (A.16). Therefore, following the same procedure as in (A.11), we obtain the following bound for $\nabla_{z_{p,i}}\widehat{f}$ tighter than (A.8c):

$$|\nabla_{z_{p,i}}\widehat{f}(\boldsymbol{D}, \boldsymbol{0})| \leq \|\boldsymbol{A}_i\|_2 \|\boldsymbol{y}_i\|_2.$$

Consequently, in Case 3, $\lambda_{\max}$ can be refined to

$$\lambda_{\max} = \max_{i=1,\dots,I} \left\{ \sigma_{\max}(\boldsymbol{A}_i) \cdot \|\boldsymbol{y}_i\|_2 \right\}.$$

A.4 Derivation of Upper Bound $\rho_{\max}$

We derive the upper bound $\rho_{\max}$ for the sparsity parameter ρ in (5.6) using the stationarity conditions (5.45) with the gradients in (5.35). Condition (5.45b) is trivial for $\boldsymbol{Z} = \boldsymbol{0}$ as, for any $\boldsymbol{X}$ and $\boldsymbol{D}$, $\nabla_{\boldsymbol{D}}\widehat{f}(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)}) = \boldsymbol{0}$. As for (5.45c), we have

$$|\nabla_{z_{p,i}}\widehat{f}(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{0}; \boldsymbol{S}^{(t)})| = \mu |\boldsymbol{d}_p^{\mathsf{H}} \boldsymbol{x}_i| \leq \mu \|\boldsymbol{d}_p\|_2 \|\boldsymbol{x}_i\|_2 \leq \mu \|\boldsymbol{x}_i\|_2. \tag{A.12}$$

Meanwhile, an upper bound for $\|\boldsymbol{x}_i\|_2$ can be derived from the vectorized form (5.59) of condition (5.45a), which reduces to

$$\operatorname{vec}(\boldsymbol{X}) = \left(\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \mu \boldsymbol{I}_{NI}\right)^{-1} \boldsymbol{F}^{\mathsf{H}} \operatorname{vec}(\boldsymbol{Y}^{(t)}), \tag{A.13}$$

for $\boldsymbol{Z} = \boldsymbol{0}$. It leads to the following upper bound for $\|\boldsymbol{x}_i\|_2$:

$$\|\boldsymbol{x}_i\|_2 \leq \|\boldsymbol{X}\|_{\mathsf{F}} \leq \left\|(\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \mu \boldsymbol{I}_{NI})^{-1}\right\|_2 \|\boldsymbol{F}\|_2 \|\boldsymbol{Y}\|_{\mathsf{F}}. \tag{A.14}$$

As an oversampling operator $\mathcal{F}$ is considered, i.e., $M_1 M_2 \geq NI$, we have

$$\left\|(\boldsymbol{F}^{\mathsf{H}}\boldsymbol{F} + \mu \boldsymbol{I}_{NI})^{-1}\right\|_2 = \left(\sigma_{\min}^2(\boldsymbol{F}) + \mu\right)^{-1}.$$

Consequently, combining (A.12) and (A.14) yields the following result. Define

$$\rho_{\max} = \frac{\mu \cdot \sigma_{\max}(\boldsymbol{F}) \cdot \|\boldsymbol{Y}\|_{\mathsf{F}}}{\sigma^2_{\min}(\boldsymbol{F}) + \mu}. \tag{A.15}$$

For $\rho \geq \rho_{\max}$, there always exists a feasible point $(\boldsymbol{X}, \boldsymbol{D}, \boldsymbol{0})$ that satisfies the stationarity conditions (5.45), and is, therefore, stationary for the majorizing function $\widehat{h}$. As $\rho_{\max}$ does not depend on the point $\boldsymbol{S}^{(t)}$ where the majorization is made, following the same line of arguments as in Appendix A.3, we further conclude that, for any $\rho \geq \rho_{\max}$, the original problem (5.6) admits a stationary point with $\boldsymbol{Z} = \boldsymbol{0}$.

In the investigated cases 1 and 3 in Section 5.6, where temporal mixing is not applied, the linear operator $\mathcal{F}$ and the corresponding matrix $\boldsymbol{F}$ can be expressed as

$$\mathcal{F}(\boldsymbol{X}) = \sum_{i=1}^{I} \boldsymbol{A}_i \boldsymbol{X} \boldsymbol{B}_i \quad \text{and} \quad \boldsymbol{F} = \sum_{i=1}^{I} \boldsymbol{B}_i^{\mathsf{T}} \otimes \boldsymbol{A}_i, \tag{A.16}$$

where $\boldsymbol{B}_i = [\boldsymbol{0}, \ldots, \boldsymbol{0}, \boldsymbol{e}_i, \boldsymbol{0}, \ldots, \boldsymbol{0}]$ selects the i-th snapshot $\boldsymbol{x}_i$ and $\boldsymbol{A}_i$ is the spatial mixing designed for $\boldsymbol{x}_i$. In Case 1, where the spatial mixing is time-invariant, all matrices $\boldsymbol{A}_i$ are set to be the same value $\boldsymbol{A}$. Substituting $\boldsymbol{F}$ in (A.16) into (A.15), we obtain

$$\rho_{\max} = \frac{\mu \cdot \max_{i=1,\ldots,I}\{\sigma_{\max}(\boldsymbol{A}_i)\}}{\min_{i=1,\ldots,I}\{\sigma^2_{\min}(\boldsymbol{A}_i)\} + \mu} \cdot \|\boldsymbol{Y}\|_{\mathsf{F}}.$$

However, the upper bound $\rho_{\max}$ can be further decreased considering that each snapshot $\boldsymbol{x}_i$ is observed independently when temporal mixing is not applied. Using the matrix $\boldsymbol{F}$ in (A.16), we can reformulate the stationary condition (A.13) as

$$\boldsymbol{x}_i = (\boldsymbol{A}_i^{\mathsf{H}} \boldsymbol{A}_i + \mu \boldsymbol{I}_N)^{-1} \boldsymbol{A}_i^{\mathsf{H}} \boldsymbol{y}_i^{(t)} \quad \text{for } i = 1, \ldots, I.$$

This results in a tighter bound of $\|\boldsymbol{x}_i\|_2$ than (A.14):

$$\|\boldsymbol{x}_i\|_2 \leq \|(\boldsymbol{A}_i^{\mathsf{H}} \boldsymbol{A}_i + \mu \boldsymbol{I}_N)^{-1}\|_2 \|\boldsymbol{A}_i\|_2 \|\boldsymbol{y}_i^{(t)}\|_2.$$

Thus, in Cases 1 and 3, the upper bound $\rho_{\max}$ is refined to

$$\rho_{\max} = \max_{i=1,\ldots,I} \left\{ \frac{\mu \cdot \sigma_{\max}(\boldsymbol{A}_i) \cdot \|\boldsymbol{y}_i\|_2}{\sigma^2_{\min}(\boldsymbol{A}_i) + \mu} \right\}.$$

List of Acronyms

BCD	Block Coordinate Descent
BSCA	Block Successive Convex Approximation
BSUM	Block Successive Upper-bound Minimization
CCCP	ConCave-Convex Procedures
DC	Difference of Convex
i.i.d	independent and identically distributed
LS	Least-Squares
MM	Majorization-Minimization
MNSE	Minimum Normalized Squared Error
MSE	Mean Squared Error
RMSE	Root-Mean-Square Error
resp.	respectively
SCA	Successive Convex Approximation
SCAphase	smoothing Successive Convex Approximation for phase retrieval with dictionary learning
SNR	Signal-to-Noise Ratio
SUM	Successive Upper-bound Minimization
SVD	Singular Value Decomposition
w.l.o.g	without loss of generality
w.r.t.	with respect to

List of Symbols

General symbols

a, A	Scalar
$\boldsymbol{a}$	Vector
$\boldsymbol{A}$	Matrix
$a_{i,j}$	The (i,j)th entry of matrix $\boldsymbol{A}$; the ith entry of vector $\boldsymbol{a}_j$
$\boldsymbol{a}_j$	The jth column of matrix $\boldsymbol{A}$
$\boldsymbol{a}_{i:}^{\mathsf{T}}$	The ith row of matrix $\boldsymbol{A}$
e	Euler's number
j	Imaginary unit
max	Maximum, maximize
min	Minimum, minimize
s.t.	Subject to
lim	limit
sup	supremum
lim sup	limit superior
$\arg(\cdot)$	Argument of a complex number
$\mathcal{O}(\cdot)$	Big O notation
$\mathcal{F}^*$	Adjoint of the linear operator $\mathcal{F}$
$\mathbb{I}_{\mathcal{X}}$	Indicator function of the set $\mathcal{X}$
$\wedge$	Logical conjunction

Special sets

$\mathbb{N}$	$\{0, 1, 2, \ldots\}$
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^m, \mathbb{R}^{m\times n}$	Set of real $m \times 1$ vectors ($m \times n$ matrices)
$\mathbb{R}_+^M$	Nonnegative orthant of $\mathbb{R}^m$
$\mathbb{C}$	Set of complex numbers
$\mathbb{C}^m, \mathbb{C}^{m\times n}$	Set of complex $m \times 1$ vectors ($M \times n$ matrices)
$\mathbb{D}^m$	Set of $m \times m$ diagonal matrices
$\mathbb{D}_+^m$	Set of $m \times m$ nonnegative diagonal matrices
$\mathcal{B}(\boldsymbol{c}, r)$	Open ball of radius r centered at $\boldsymbol{c}$, i.e., $\{\boldsymbol{x} \in \mathbb{R}^n \mid \|\boldsymbol{x} - \boldsymbol{c}\|_2 \leq r\}$

Special matrices and vectors

$\mathbf{0}$	All-zero vector or matrix of appropriate dimension
$\mathbf{1}$	All-one vector or matrix of appropriate dimension
$\boldsymbol{e}_m$	The m-th column of an identity matrix
$\boldsymbol{I}$	Identity matrix of of appropriate dimension
$\boldsymbol{I}_M$	Identity matrix of dimension $M \times M$

Operations on sets

co	Convex hull

Operations on matrices and vectors

$\Re(\cdot)$	Real part
$\Im(\cdot)$	Imaginary part
$\mathrm{Diag}\,(\cdot)$	Diagonal matrix whose elements are specified by the arguments
$\mathcal{P}_{\mathcal{X}}(\cdot)$	Orthogonal projection onto the set $\mathcal{X}$
$\mathrm{rank}\,(\cdot)$	Rank of the matrix argument
$\mathrm{tr}(\cdot)$	Trace of the matrix argument
$\mathrm{vec}\,(\cdot)$	Vectorization operator
$\boldsymbol{X} \succ 0$	$\boldsymbol{X}$ is positive definite
$\boldsymbol{X} \succeq 0$	$\boldsymbol{X}$ is positive semidefinite
$\sigma_{\max}(\cdot)$	The largest singular value
$\sigma_{\min}(\cdot)$	The smallest singular value
$\sigma_{\min,\mathrm{nz}}(\cdot)$	The smallest nonzero singular value
$(\cdot)^{\mathrm{T}}$	Transpose operator
$\overline{(\cdot)}$	Conjugate operator
$(\cdot)^{\dagger}$	Moore-Penrose pseudo-inverse operator
$(\cdot)^{-1}$	Inverse operator
$(\cdot)^{\mathrm{H}}$	Hermitian operator (conjugate transpose)
$\otimes$	Kronecker product
$\odot$	Hadamard product, i.e., elementwise product
$[\cdot]_{i,j}$	The (i,j)th entry of the matrix argument
$\lvert\cdot\rvert$	Elementwise absolute value
$\lVert\cdot\rVert_p$	ℓ_p-norm of the vector or matrix argument
$\lVert\cdot\rVert_{p,q}$	$\ell_{p,q}$-norm of the matrix argument
$\lVert\cdot\rVert_{\mathrm{F}}$	Frobenius norm of the matrix argument

Differentiation

$\dfrac{\partial f}{\partial x}$	First-order partial derivative of the function f w.r.t. $x \in \mathbb{R}$; Wirtinger derivative of the function f w.r.t. $x \in \mathbb{C}$
$\dfrac{\partial f}{\partial \overline{x}}$	Wirtinger derivative of the function f w.r.t. $\overline{x}$
$\dfrac{\partial^2 f}{\partial x^2}$	Second-order partial derivative of the function f w.r.t. x
∇f	Gradient of the function f

$\nabla_{\boldsymbol{x}} f$	Partial gradient of the function f w.r.t. $\boldsymbol{x}$
$\nabla^2 f$	Hessian matrix of the function f
$\nabla^2_{\boldsymbol{x}} f$	Partial Hessian matrix of the function f w.r.t. $\boldsymbol{x}$, i.e., the principal submatrix of ∇f corresponding to $\boldsymbol{x}$
$f'_{\boldsymbol{d}}(\cdot)$	The directional derivative of f in the direction $\boldsymbol{d}$
$f^{\circ}_{\boldsymbol{d}}(\cdot)$	The Clarke directional derivative of f in the direction $\boldsymbol{d}$
∂f	Convex subdifferential of the function f
$\underline{\partial} f$	Generalized subdifferential of f defined by directional derivatives
$\partial_C f$	Clarke generalized subdifferential of the function f
$\mathcal{N}_{\mathcal{X}}(\cdot)$	The convex normal cone of to the set $\mathcal{X}$
$\mathcal{N}^C_{\mathcal{X}}(\cdot)$	The Clarke normal cone of to the set $\mathcal{X}$

References

[ALJ19] M. U. Aminu, J. Lehtomäki, and M. Juntti, "Beamforming and transceiver optimization with phase noise for mmWave and THz bands," in *International Symposium on Wireless Communication Systems*, Aug. 2019, pp. 692–696.

[Ber97] C. Berge, *Topological spaces: including a treatment of multi-valued functions, vector spaces, and convexity.* Mineola, N.Y: Dover Publications, 1997.

[Ber16] D. P. Bertsekas, *Nonlinear programming*, 3rd ed. Belmont, Mass: Athena scientific, 2016.

[BNS78] J. R. Bunch, C. P. Nielsen, and D. C. Sorensen, "Rank-one modification of the symmetric eigenproblem," *Numer. Math*, vol. 31, pp. 31–48, 1978.

[BP18] A. Beck and D. Pan, "Convergence of an inexact majorization-minimization method for solving a class of composite optimization problems," in *Large-Scale and Distributed Optimization*, ser. Lecture Notes in Mathematics, P. Giselsson and A. Rantzer, Eds. Cham: Springer International Publishing, 2018, pp. 375–410.

[BT15] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods.* Nashua, NH: Athena Scientific, 2015.

[BV04] S. Boyd and L. Vandenberghe, *Convex optimization.* Cambridge University Press, 2004.

[CCFM19] Y. Chen, Y. Chi, J. Fan, and C. Ma, "Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval," *Mathematical Programming*, vol. 176, no. 1, pp. 5–37, Jul. 2019.

[CESV15] E. J. Candès, Y. C. Eldar, T. Strohmer, and V. Voroninski, "Phase retrieval via matrix completion," *SIAM review*, vol. 57, no. 2, pp. 225–251, 2015.

[Cla75] F. H. Clarke, "Generalized gradients and applications," *Transactions of the American Mathematical Society*, vol. 205, pp. 247–247, 1975.

[Cla90] ——, *Optimization and nonsmooth analysis*, ser. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, Jan. 1990.

[CLS15a] E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval from coded diffraction patterns," *Applied and Computational Harmonic Analysis*, vol. 39, no. 2, pp. 277–299, 2015.

[CLS15b] ——, "Phase retrieval via Wirtinger flow: theory and algorithms," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1985–2007, Apr. 2015.

[CSV13] E. J. Candès, T. Strohmer, and V. Voroninski, "PhaseLift: exact and stable signal recovery from magnitude measurements via convex programming," *Communications on Pure and Applied Mathematics*, vol. 66, no. 8, pp. 1241–1274, 2013.

[CZ97] Y. Censor and S. A. Zenios, *Parallel optimization: theory, algorithms, and applications*. Oxford University Press, USA, 1997.

[DFKS15] A. Daneshmand, F. Facchinei, V. Kungurtsev, and G. Scutari, "Hybrid random/deterministic parallel algorithms for convex and nonconvex big data optimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 15, pp. 3914–3929, Aug. 2015.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 39, no. 1, pp. 1–22, Sep. 1977.

[Don95] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[EHM16] Y. C. Eldar, N. Hammen, and D. G. Mixon, "Recent advances in phase retrieval [lecture notes]," *IEEE Signal Processing Magazine*, vol. 33, no. 5, pp. 158–162, Sep. 2016.

[ESM+14] Y. C. Eldar, P. Sidorenko, D. G. Mixon, S. Barel, and O. Cohen, "Sparse phase retrieval from short-time Fourier measurements," *IEEE Signal Processing Letters*, vol. 22, no. 5, pp. 638–642, 2014.

[FD87] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy," in *Image recovery: theory and application*, 1987, vol. 231, p. 275.

[Fie82] J. R. Fienup, "Phase retrieval algorithms: a comparison," *Applied Optics*, vol. 21, no. 15, pp. 2758–2769, Aug. 1982.

[FNW07] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec. 2007.

[FS20] A. Fannjiang and T. Strohmer, "The numerics of phase retrieval," *Acta Numerica*, vol. 29, pp. 125–228, May 2020.

[GS72] R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik*, vol. 35, pp. 237–246, 1972.

[GS18] T. Goldstein and C. Studer, "PhaseMax: convex phase retrieval via basis pursuit," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 2675–2689, Apr. 2018.

[GVL13] G. H. Golub and C. F. Van Loan, *Matrix computations*, 4th ed. The Johns Hopkins University Press, 2013.

[Har93] R. W. Harrison, "Phase problem in crystallography," *JOSA A*, vol. 10, no. 5, pp. 1046–1055, May 1993.

[HL01] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of convex analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001.

[HRLP16] M. Hong, M. Razaviyayn, Z.-Q. Luo, and J.-S. Pang, "A unified algorithmic framework for block-structured optimization involving big data: with applications in machine learning and signal processing," *IEEE Signal Processing Magazine*, vol. 33, no. 1, pp. 57–77, Jan. 2016.

[HS75] H. R. Howson and N. G. F. Sancho, "A new algorithm for the solution of multi-state dynamic programming problems," *Mathematical Programming*, vol. 8, no. 1, pp. 104–116, Dec. 1975.

[HSDR14] C.-J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. Ravikumar, "QUIC: quadratic approximation for sparse inverse covariance estimation," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2911–2947, 2014.

[JEH16] K. Jaganathan, Y. C. Eldar, and B. Hassibi, "STFT phase retrieval: uniqueness guarantees and recovery algorithms," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 770–781, Jun. 2016.

[KHE14] H. Kim, A. M. Haimovich, and Y. C. Eldar, "Non-coherent direction of arrival estimation from magnitude-only measurements," *IEEE Signal Processing Letters*, vol. 22, no. 7, pp. 925–929, 2014.

[KKL+07] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 606–617, Dec. 2007.

[Kru03] A. Ya. Kruger, "On Fréchet subdifferentials," *Journal of Mathematical Sciences*, vol. 116, no. 3, pp. 3325–3358, 2003.

[Lan16] K. Lange, *MM optimization algorithms*, ser. Other Titles in Applied Mathematics. Society for Industrial and Applied Mathematics, 2016.

[LB16] T. Lipp and S. Boyd, "Variations and extension of the convex–concave procedure," *Optimization and Engineering*, vol. 17, no. 2, pp. 263–287, Jun. 2016.

[Li93] R.-C. Li, "Solving secular equations stably and efficiently," EECS Department, University of California, Berkeley, Tech. Rep. UCB/CSD-94-851, 1993.

[LK70] R. Larson and A. Korsak, "A dynamic programming successive approximations technique with convergence proofs," *Automatica*, vol. 6, no. 2, pp. 245–252, Mar. 1970.

[LLN22] B. Lafferriere, G. Lafferriere, and M. Nguyen, *Introduction to mathematical analysis i - 3rd edition*, 3rd ed. Portland State University Library, Sep. 2022.

[LSJL17] J. Liang, P. Stoica, Y. Jing, and J. Li, "Phase retrieval via the alternating direction method of multipliers," *IEEE Signal Processing Letters*, vol. 25, no. 1, pp. 5–9, 2017.

[LSM20] J. Li, A. M.-C. So, and W.-K. Ma, "Understanding notions of stationarity in nonsmooth optimization: a guided tour of various constructions of subdifferential for nonsmooth functions," *IEEE Signal Processing Magazine*, vol. 37, no. 5, pp. 18–31, Sep. 2020.

[LTPD18] H. A. Le Thi and T. Pham Dinh, "DC programming and DCA: thirty years of developments," *Mathematical Programming*, vol. 169, no. 1, pp. 5–68, May 2018.

[LTPD23] ——, "Open issues and recent advances in DC programming and DCA," *Journal of Global Optimization*, Feb. 2023.

[LTY+22] T. Liu, A. M. Tillmann, Y. Yang, Y. C. Eldar, and M. Pesavento, "Extended successive convex approximation for phase retrieval with dictionary learning," *IEEE Transactions on Signal Processing*, vol. 70, pp. 6300–6315, 2022.

[LTYP19] T. Liu, M. Trinh-Hoang, Y. Yang, and M. Pesavento, "A block coordinate descent algorithm for sparse Gaussian graphical model inference with laplacian constraints," in *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Dec. 2019, pp. 236–240.

[LVS23] B. Li, G. Verma, and S. Segarra, "Graph-based algorithm unfolding for energy-aware power allocation in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 22, no. 2, pp. 1359–1373, Feb. 2023.

[LWLZ21] K. Lange, J.-H. Won, A. Landeros, and H. Zhou, "Nonconvex optimization via MM algorithms: convergence theory," in *Wiley StatsRef: Statistics Reference Online*. John Wiley & Sons, Ltd, Mar. 2021, pp. 1–22.

[Man94] O. L. Mangasarian, *Nonlinear programming*, ser. Classics in applied mathematics. Philadelphia, Pa: Society for Industrial and Applied Mathematics, 1994, no. 10.

[Mil90] R. P. Millane, "Phase retrieval in crystallography and optics," *JOSA A*, vol. 7, no. 3, pp. 394–411, Mar. 1990.

[MK07] G. J. McLachlan and T. Krishnan, *The EM algorithm and extensions*. John Wiley & Sons, 2007.

[MLE21] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: interpretable, efficient deep learning for signal and image processing," *IEEE Signal Processing Magazine*, vol. 38, no. 2, pp. 18–44, Mar. 2021.

[MN07] J. R. Magnus and H. Neudecker, *Matrix differential calculus with applications in statistics and econometrics*, 3rd ed. Wiley, 2007.

[MNY06] B. S. Mordukhovich, N. M. Nam, and N. D. Yen, "Fréchet subdifferential calculus and optimality conditions in nondifferentiable programming," *Optimization*, vol. 55, no. 5-6, pp. 685–708, Oct. 2006.

[Mor06] B. S. Mordukhovich, *Variational analysis and generalized differentiation i*, ser. Grundlehren der mathematischen Wissenschaften. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, vol. 330.

[MRS08] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. New York: Cambridge University Press, 2008.

[Ngu17] H. D. Nguyen, "An introduction to majorization-minimization algorithms for machine learning and statistical estimation," *WIREs Data Mining and Knowledge Discovery*, vol. 7, no. 2, p. e1198, 2017.

[NLS22] J. Nutini, I. Laradji, and M. Schmidt, "Let's make block coordinate descent converge faster: faster greedy rules, message-passing, active-set complexity, and superlinear convergence," *Journal of Machine Learning Research*, vol. 23, no. 131, pp. 1–74, 2022.

[NSL+15] J. Nutini, M. Schmidt, I. Laradji, M. Friedlander, and H. Koepke, "Coordinate descent converges faster with the Gauss-Southwell rule than random selection," in *Proceedings of the 32nd International Conference on Machine Learning*. PMLR, Jun. 2015, pp. 1632–1641.

[OR70] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*. SIAM, 1970.

[PB14] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[PBES18] E. J. R. Pauwels, A. Beck, Y. C. Eldar, and S. Sabach, "On Fienup methods for sparse phase retrieval," *IEEE Transactions on Signal Processing*, vol. 66, no. 4, pp. 982–991, Feb. 2018.

[Pow73] M. J. D. Powell, "On search directions for minimization algorithms," *Mathematical Programming*, vol. 4, no. 1, pp. 193–201, Dec. 1973.

[PRA17] J.-S. Pang, M. Razaviyayn, and A. Alvarado, "Computing B-stationary points of nonsmooth DC programs," *Mathematics of Operations Research*, vol. 42, no. 1, pp. 95–118, 2017.

[QD11] T. D. Quoc and M. Diehl, "Sequential convex programming methods for solving nonlinear optimization problems with DC constraints," 2011.

[QP17] T. Qiu and D. P. Palomar, "Undersampled sparse phase retrieval via majorization–minimization," *IEEE Transactions on Signal Processing*, vol. 65, no. 22, pp. 5957–5969, Nov. 2017.

[RHL13] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, Jan. 2013.

[Roc70] R. T. Rockafellar, *Convex analysis*. Princeton University Press, 1970.

[SAH18] F. Salehi, E. Abbasi, and B. Hassibi, "Learning without the phase: regularized PhaseMax achieves optimal sample complexity," in *Advances in neural information processing systems*, vol. 31. Curran Associates, Inc., 2018.

[Say14] A. H. Sayed, *Adaptation, learning, and optimization over networks*, ser. Foundations and trends in machine learning. Boston: Now, 2014, no. 7:4-5.

[SBE14] Y. Shechtman, A. Beck, and Y. C. Eldar, "GESPAR: efficient phase retrieval of sparse signals," *IEEE Transactions on Signal Processing*, vol. 62, no. 4, pp. 928–938, Feb. 2014.

[SBP17] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2017.

[SEC+15] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging: a contemporary overview," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 87–109, May 2015.

[SFL17a] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization—part i: theory," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1929–1944, Apr. 2017.

[SFL+17b] G. Scutari, F. Facchinei, L. Lampariello, S. Sardellitti, and P. Song, "Parallel and distributed methods for constrained nonconvex optimization-part ii: applications in communications and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1945–1960, Apr. 2017.

[SFS+14] G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J.-S. Pang, "Decomposition by partial linearization: parallel optimization of multi-agent systems," *IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 641–656, Feb. 2014.

[SRLH11] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4331–4340, Sep. 2011.

[SS18] G. Scutari and Y. Sun, "Parallel and distributed successive convex approximation methods for big-data optimization," in *Multi-agent Optimization: Cetraro, Italy 2014*, A. Nedić, J.-S. Pang, G. Scutari, Y. Sun, F. Facchinei, and J.-S. Pang, Eds. Cham: Springer International Publishing, 2018, pp. 141–308.

[TDB21] B. Tolooshams, S. Dey, and D. Ba, "Deep residual autoencoders for expectation maximization-inspired dictionary learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 6, pp. 2415–2429, Jun. 2021.

[TEM16] A. M. Tillmann, Y. C. Eldar, and J. Mairal, "DOLPHIn—dictionary learning for phase retrieval," *IEEE Transactions on Signal Processing*, vol. 64, no. 24, pp. 6485–6500, Dec. 2016.

[Tib96] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Royal Statistical Society*, vol. 58, no. 1, pp. 267–288, 1996.

[Tse01] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of Optimization Theory and Applications*, vol. 109, no. 3, pp. 475–494, Jun. 2001.

[TY09] P. Tseng and S. Yun, "A coordinate gradient descent method for nonsmooth separable minimization," *Mathematical Programming*, vol. 117, no. 1, pp. 387–423, Mar. 2009.

[WM15] I. Waldspurger, A. d'Aspremont, and S. Mallat, "Phase recovery, MaxCut and complex semidefinite programming," *Mathematical Programming*, vol. 149, no. 1, pp. 47–81, Feb. 2015.

[WFDL20] B. Wang, J. Fang, H. Duan, and H. Li, "PhaseEqual: convex phase retrieval via alternating direction method of multipliers," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1274–1285, 2020.

[WGE18] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 773–794, Feb. 2018.

[Wir27] W. Wirtinger, "Zur formalen Theorie der Funktionen von mehr komplexen Veränderlichen," *Mathematische Annalen*, vol. 97, no. 1, pp. 357–375, Dec. 1927.

[WYLM12] Z. Wen, C. Yang, X. Liu, and S. Marchesini, "Alternating direction methods for classical and ptychographic phase retrieval," *Inverse Problems*, vol. 28, no. 11, p. 115010, Oct. 2012.

[WZG+18] G. Wang, L. Zhang, G. B. Giannakis, M. Akçakaya, and J. Chen, "Sparse phase retrieval via truncated amplitude flow," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 479–491, Jan. 2018.

[YP17] Y. Yang and M. Pesavento, "A unified successive pseudoconvex approximation framework," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3313–3328, 2017.

[YPCO18] Y. Yang, M. Pesavento, S. Chatzinotas, and B. Ottersten, "Successive convex approximation algorithms for sparse signal estimation with nonconvex regularizations," *IEEE Journal on Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1286–1302, 2018.

[YPEO19] Y. Yang, M. Pesavento, Y. C. Eldar, and B. Ottersten, "Parallel coordinate descent algorithms for sparse phase retrieval," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2019, pp. 7670–7674.

[YPLO20] Y. Yang, M. Pesavento, Z.-Q. Luo, and B. Ottersten, "Inexact block coordinate descent algorithms for nonsmooth nonconvex optimization," *IEEE Transactions on Signal Processing*, vol. 68, pp. 947–961, 2020.

[YR03] A. L. Yuille and A. Rangarajan, "The concave-convex procedure," *Neural Computation*, vol. 15, no. 4, pp. 915–936, Apr. 2003.

[YT11] S. Yun and K.-C. Toh, "A coordinate gradient descent method for ℓ_1-regularized convex minimization," *Computational Optimization and Applications*, vol. 48, no. 2, pp. 273–307, Mar. 2011.

[ZW89] Z. Q. Zuo and C. P. Wu, "Successive approximation technique for a class of large-scale NLP problems and its application to dynamic programming," *Journal of Optimization Theory and Applications*, vol. 62, no. 3, pp. 515–527, Sep. 1989.