National University of Science and Technology Politehnica Bucharest
Faculty of Automatic Control and Computers
Department of Automatic Control and Systems Engineering

PHD THESIS

Higher-order methods for composite optimization and applications

Student
Yassine Nabou

Advisor
Professor Ion Necoară

Bucharest, 2024

Acknowledgement

I would like to express my deepest gratitude to my supervisor, Ion Necoară, for providing me with the opportunity to take on this role. I am grateful for the guidance, corrections, and invaluable insights that greatly contributed to the enhancement and enrichment of this work.

I would also like to thank François Glineur, Silvia Villa, Lucian Toma, Mehdi Madani and Pierre Artoisenet for their collaboration during my PhD. Their contributions significantly enriched this research.

I wish to thank my colleagues from the Automatic Control and System Engineering Department, Lahcen El Bourkhisi, Nitesh Kumar Singh, Daniela Lupu and Flavia Chorobura, whose insightful discussions and collaborations have enriched my research. Our fruitful exchanges of ideas and experiences have contributed significantly to my academic journey.

I wish also to thank my colleagues, mentors, and the whole TraDE-OPT team for their invaluable support that made this achievement possible.

Throughout my life, I've met many wonderful people who have taught and encouraged me in my research activities. I am grateful to Mohamed Alaoui, Ahmed Roubi, Pierre Maréchal, and Masoud Ahookhosh for influencing the path of my research.

Finally, I want to extend my heartfelt gratitude to my entire family for their unwavering emotional and financial support throughout this challenging journey. Their encouragement, love, and understanding have served as the bedrock of my strength. Completing this thesis would not have been possible without their steadfast support.

This project has received funding from the ITN-ETN project TraDE-OPT, the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skolodowska-Curie grant agreement No. 861137.

Contents

1 Introduction

In the mid-twentieth century, the fundamental concept of optimization emerged as a cornerstone in computational mathematics and computer science, significantly touching various domains such as engineering, management, physics, and machine learning. Researchers from all disciplines have embraced optimization as an essential tool for addressing complex problems in their respective fields. For instance, engineers employ optimization techniques to generate the best control commands for complex dynamical systems, design efficient transportation systems, and improve mechanical systems' performance [1]. In finance, analysts apply optimization approaches to boost earnings and reduce risks [2]. In machine learning, solving optimization problems is essential for training models, fine-tuning parameters, and optimizing neural network designs [3]. As a result, the search for successful optimization solutions goes beyond basic convenience; it is a strategic priority for businesses, academics, and governments alike. These strategies not only speed up decision-making processes, but they also promote innovation across other areas. Thus, prioritizing the progress of such approaches becomes critical to fulfilling the rapidly evolving demands within our global society. Addressing a specific optimization problem with the most efficient algorithms remains a significant challenge in optimization research. Attempting to devise a universal method for all optimization problems is overly ambitious given their extensive range and generality. Embracing this perspective allows researchers to tailor specialized methodologies to the complexities inherent in some specific class of optimization problems, enhancing effectiveness and expanding practical applications across various fields.

Many optimization problems can be mathematically described as the minimization of a given function, often referred to as an objective or cost function, under a set of constraints. The complicated nature of these problems, as described by their complex structures and constraints, highlights the pressing need for creative and efficient problem-solving approaches. For example, in game theory, one needs to identify a strategy that yields the best outcome under worst-case conditions. This is characterized as a *min-max* problem, because the goal is to reduce the largest possible loss.

Structured optimization problems can include scenarios like the sum of a convex function and a composition of a convex merit function with a smooth map, such as in min-max optimization problems, which pose significant challenges, particularly when the merit function is only convex. In this case, the optimization process becomes intricate due to the complexity of the problem. However, the difficulty diminishes when the merit exhibits specific structural characteristics or has additional properties beyond convexity, which can enable more efficient and effective optimization algorithms. These additional properties can provide valuable insights into the problem structure, guide the optimization algorithms towards better solutions, and potentially overcome the complexities inherent in the optimization process. Thus, while optimization problems may provide formidable challenges, exploiting the properties of the objective function can significantly facilitate their resolution.

First-order (i.e., gradient descent-based) approaches (GD) have become known as an important tool for solving nonlinear problems in optimization. Gradient descent method was introduced by Cauchy in 1847 [4] and later many variants have been proposed (conjugate gradient, accelerated gradient, stochastic gradient, etc). The extensive research on the convergence of first-order methods started in the 1950s; see, e.g., Polyak's paper from 1963 on the gradient method [5] and the subgradient method as discovered by Shor [6] in 1962. Each iteration of the GD scheme computes only a gradient of an objective function and returns a solution with a specified accuracy. First-order optimisation has advanced tremendously, and despite many great achievements, its convergence rate is slow due to the fundamental theoretical limitations (e.g., the rates depend on the condition number of the objective function) that are represented by the lower complexity bounds. This issue motivates the scientists to propose versions of this method by modifying it to fit specific classes of problems (e.g., projected gradient, stochastic gradient methods, etc) or by improving the convergence speed (e.g., conjugate gradient, accelerate gradient, Gauss-Newton, etc). As such, a significant portion of research has been devoted to examining this component [7, 8]. In convex optimization, first-order methods generally exhibit a convergence rate of order $\mathcal{O}(k^{-1})$ when measured by function values, where k is the iteration count. However, for nonconvex problems, the convergence rate in the norm of the gradient is of order $\mathcal{O}(k^{-1/2})$, [8].

Second-order methods are known for being powerful algorithms due to their ability to solve ill-conditioned problems. For example, in the classical Newton method, one needs to approximate the objective by its second-order Taylor approximation. Compared to the gradient method, Newton method has local quadratic convergence in a neighbourhood of the solution. However, the global behaviour of the Newton method has remained an active area of research for several decades. It is known that the classical Newton method with a unit stepsize may not converge globally, even if the problem is strongly convex (see Example 1.4.3 in [9]). A significant advancement in second-order optimization theory occurred following the paper [10] by Nesterov and Polyak in 2006, where the authors introduced a cubic regularization of Newton's method along with comprehensive global complexity guarantees. The fundamental concept proposed in [10] revolves around employing a global upper approximation model using the second-order Taylor polynomial and a cubic regularization term. The authors demonstrated that the cubic Newton algorithm offers global convergence rates compared to conventional Newton method. In convex settings, the convergence rate of the cubic Newton algorithm in function values is of order $\mathcal{O}(k^{-2})$, see [10], while in nonconvex settings, the convergence rate in the norm of the gradient is of order $\mathcal{O}\left(k^{-\frac{2}{3}}\right)$ [10], which are both faster than for the first-order methods.

A natural way to ensure faster convergence rates, extending beyond both gradient and second-order methods, is to use higher-order information (derivatives) to build sophisticated higher-order Taylor models. The unpublished preprint [11] stands as the first paper deriving theoretical results of the higher-order schemes for convex problems. However the extensive complexity associated with minimising nonconvex multivariate polynomials has posed significant challenges, rendering this initial effort unsuccessful. Despite these obstacles, a ray of hope emerged through the groundbreaking research of Nesterov in [12]. Specifically, Nesterov demonstrated that by appropriately regularizing the Taylor approximation, the auxiliary subproblem remains convex and can be solved efficiently, thereby offering a promising avenue for tackling convex unconstrained smooth problems. The convergence rate for convex problems in function values is of order $\mathcal{O}\left(k^{-p}\right)$. This significant result not only underscores the potential of higher-order methods but also serves as a catalyst for further exploration and refinement within the optimization community. As researchers delve deeper into the intricacies of optimization methods, particularly within the nonconvex setting [13, 14], a notable focus has been placed on analyzing the complexity of high-order approaches. These approaches aim to generate solutions with small gradient norms, a crucial aspect for navigating nonconvex optimization landscapes effectively. In addressing this challenge, it is essential for such methods to maintain a satisfactory adherence to first-order optimality conditions and ensure local reductions in the objective function. However, achieving this balance proves to be inherently challenging when dealing with nonconvex functions. Convergence guarantees, particularly in terms of the norm of the gradient, have been established to be of order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$. Increasing the order of the oracle, denoted as p, does offer some advantages, albeit not as pronounced as in convex settings. Despite the complexities involved, ongoing research efforts continue to explore the performance of high-order optimization methods in nonconvex scenarios, striving towards more robust and efficient techniques.

Optimization methods based on gradient information are widely used in applications where high accuracy is not desired, such as machine learning, data analysis, signal processing and statistics [15, 16, 17, 18]. The standard convergence analysis of gradient-based methods requires the availability of the exact first-order information. Namely, the oracle must provide at each given point the exact values of the function and of its gradient. However, in many optimization problems, one doesn't have access to exact gradients, e.g., the gradient is obtained by solving another optimization subproblem. In practice, we are often only able to solve these subproblems approximately. Hence, in that context, numerical methods solving the outer problem are provided with inexact first-order information. This led us to also investigate the behavior of first-order methods working with an inexact information.

1.1 Contributions

The main objective of this thesis is to build and analyze efficient high-order optimization methods designed to effectively address the challenges posed by increasingly complex and structured problem landscapes. Specifically, we focus on composite optimization problems that involve a sum of two terms, where the first term is a composition between a convex merit function and smooth maps. These types of problems frequently arise in various applications, including complex systems, game theory and control. We are interested in designing implementable algorithms and deriving explicit convergence rates, having the goal of gaining both theoretical and practical justification for our approaches. Hereafter, we outlines the main theoretical and numerical contributions of this thesis, primarily focusing on the results from Chapters 3 through 7. More specifically, the specific contributions of this thesis are as follows:

We provide an algorithmic framework in **Chapter 3** based on the notion of higher-order upper bound approximations for solving composite problems, where the first term is a composition between a convex merit function and maps. We consider general properties for our objects, e.g., the maps, can be smooth or nonsmooth, convex or nonconvex and the merit function is convex, subhomogeneous, nondecreasing and has full domain. Our framework consists of replacing the maps by a higher-order surrogate (majorizer), leading to a *General Composite Higher-Order* minimization algorithm, which we call GCHO. This majorization-minimization approach is relevant as it yields an array of algorithms, each of which is associated with the specific properties of the maps and the corresponding surrogate, and it provides a unified convergence analysis. Note that most of our variants of GCHO for $p > 1$ were not explicitly considered in the literature before (at least in the nonconvex settings).

We derive convergence guarantees for the GCHO algorithm when the upper bound approximate the maps from the objective function up to an error that is $p \geq 1$ times differentiable and has a Lipschitz continuous p derivative; we call such upper bounds composite higher-order surrogate functions. More precisely, on composite (possibly nonsmooth) nonconvex problems we prove for GCHO, with the help of a new auxiliary sequence, convergence rates $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$ in terms of first-order optimality conditions. We also characterize the convergence rate of GCHO algorithm locally, in terms of function values, under the Kurdyka-Lojasiewicz (KL) property. Our result show that the convergence behavior of GCHO ranges from sublinear to linear depending on the parameter of the underlying KL geometry. Moreover, on general (possibly nonsmooth) composite convex problems (i.e., f is convex function) our algorithm achieves global sublinear convergence rate of order $\mathcal{O}\left(k^{-p}\right)$ in function values. We summaries our convergence results in Table 1.1. Finally, for $p = 2$ and the merit function is the maximum function, we show that the subproblem, even in the nonconvex case, is equivalent to minimizing an explicitly written convex function over a convex set that can be solve using efficient convex optimization tools.

Besides providing a general framework for the design and analysis of composite higher-order methods, in special cases, where complexity bounds are known for some particular algorithms, our convergence results recover the existing bounds. For example, from our convergence analysis one can easily recover the convergence bounds of higher-order algorithms from [12] for unconstrained minimization and from [12, 19, 20] for simple composite minimization. Furthermore, in the composite *convex* case we recover the convergence bounds from [21] for $p = 1$ and particular choices of g and from [22] for $p \geq 1$. To the best of our knowledge, this is the first complete work dealing with composite problems in the nonconvex and nonsmooth settings, and explicitly deriving convergence bounds for higher-order majorization-minimization algorithms (including convergence under KL). The content of this chapter is based on paper [23].

	Assumptions	convergence rates	Theorem
nonconvex case	3.2.1 and 3.3.2	$\exists (y_k)_{k\geq 0}$ close to $(x_k)_{k\geq 0}$ s.t. $\min_{j=0:k} \mathrm{dist}(0, \partial f(y_j)) \leq \mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$	3.3.6
	3.2.1, 3.3.2, 3.3.9 and KL	$f(x_k) \to f_*$ sublinear or linear	3.3.10
convex case	3.2.1 and f convex	$f(x_k) - f^* \leq \mathcal{O}\left(k^{-p}\right)$	3.3.11

Table 1.1: Convergence results for the algorithm presented in Chapter 3.

Chapter 4 introduces a new moving Taylor approximations (MTA) method designed to tackle optimization problems with functional constraints. Our framework is flexible in the sense that we can approximate the objective function and the constraints with higher-order Taylor approximations of different degrees (i.e., we can approximate the smooth part of the objective function with a Taylor approximation of degree p and the constraints with a Taylor approximation of degree q). We derive convergence guarantees for MTA algorithm for (non)convex problems with smooth (non)convex functional constraints. More precisely, when the data is nonconvex, we show that the iterates generated by MTA converge to a KKT point and the convergence rate is of order $\mathcal{O}\left(k^{-\min\left(\frac{p}{p+1}, \frac{q}{q+1}\right)}\right)$, where k is the iteration counter and p and q are the degrees of the Taylor approximations for objective and constraints, respectively. When the data of the problem are semialgebraic, we derive linear/sublinear convergence rates in the iterates (depending on the parameter of the KL property). Moreover, for convex problems, we derive a global sublinear convergence rate of order $\mathcal{O}\left(k^{-\min(p,q)}\right)$ in function values, and additionally, if the objective function is uniformly convex, we derive a superlinear/linear convergence rate (depending on the degree of the uniform convexity). The convergence rates obtained are summarized in Table 1.2. Note that the subproblem we need to solve at each iteration of MTA is usually nonconvex and it can have local minima. However, we show for $p, q \leq 2$ that our approach is implementable, since this subproblem is equivalent to minimizing an explicitly written convex function over a convex set that can be solve using efficient convex optimization tools. We believe that this is an additional step towards practical implementation of higher-order (tensor) methods in smooth nonconvex optimization problems with smooth nonconvex functional constraints.

It is important to note that in special cases, where complexity bounds are known for some particular algorithms, our convergence results recover the existing bounds. For example, for $p = q = 1$, we recover the convergence results obtained in [24, 25] in the nonconvex setting. We also recover the sublinear convergence rate in the convex case derived in [21] for $p = q = 1$, as well as the linear convergence rate in function values obtained in [22], but we only assume uniform convexity on the objective and not on the constraints. The content of this chapter is based on paper [26].

	Assumptions	convergence rates	Theorem
nonconvex	4.2.1, 4.2.2 and 4.2.3	Measure of optimality (see section 4.3.1): $\min_{i=1:k} \mathcal{M}(x_i) \leq \mathcal{O}\left(k^{-\min\left(\frac{p}{p+1},\frac{q}{q+1}\right)}\right)$	4.3.4
	4.2.1, 4.2.2, 4.2.3 and KL	$x_k \to x^*$ sublinearly or linearly	4.3.9
convex	4.2.1, 4.2.2 and 4.2.3	$F(x_k) - F^* \leq \mathcal{O}\left(k^{-\min(p,q)}\right)$	4.4.1
uniformly convex	4.2.1, 4.2.2 and 4.2.3	$F(x_k) \to F^*$ linearly or superlinearly	4.4.2

Table 1.2: Convergence results for the algorithm presented in Chapter 4.

Chapter 5 presents a regularized higher-order Taylor approximation method (referred to as RHOTA) for solving composite problems, where the first term involves a convex Lipschitz function composed with a smooth map. The most representative class of problems that fit into this framework is the nonlinear least-squares. At each iteration of RHOTA, a higher-order composite model is constructed, which is then minimized to compute the next iteration. An adaptive variant of RHOTA is also introduced. We establish worst-case complexity bounds for the RHOTA algorithm. Specifically, we demonstrate that the iterates generated by RHOTA converge to a near-stationary point, and the convergence rate is of the order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$. When the data of the problem are semialgebraic or, more generally, satisfy Kurdyka-Lojasiewicz (KL) property, we derive linear/sublinear convergence rates in function values, depending on the parameter of the KL condition. When the merit function is the norm, i.e., $\|\cdot\|$, we present an efficient implementation of RHOTA algorithm when the Taylor approximation is of order $p = 2$. In particular, we show that the resulting nonconvex subproblem is equivalent to minimizing an explicitly written convex function over a convex set and, thus, can be solved using standard convex tools. This represents a significant advancement towards the practical implementation of higher-order methods for solving nonlinear least-squares problems.

We also analyze the convergence behavior of RHOTA algorithm when employed to address systems of nonlinear equations and optimization problems featuring nonlinear equality constraints. For these problems, we introduce non-conservative constraints qualification conditions that guarantee convergence of RHOTA. More precisely, for nonlinear system of equations, we show that our scheme solves this problem in a finite number of iterations under a non-degenerate Jacobian. Additionally, for optimization problems with nonlinear equality constraints, we show that the proposed algorithms converges to a KKT point with a rate of order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$. The convergence rates obtained are summarized in Table 1.3. The content of this chapter is based on paper [27].

Problem	Assumptions	convergence rates	Theorem
(5.4)	5.2.1	$\exists (y_k)_{k\geq 0}$ close to $(x_k)_{k\geq 0}$ s.t. $\min_{j=0:k} \mathrm{dist}(0, \partial f(y_j)) \leq \mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$	5.3.2
(5.19)	5.2.1 and 5.20	Exists finite $k < \infty$: $F(x_k) = 0$ or $F(y_k) = 0$	5.5.1
(5.22)	5.2.1 and 5.24	KKT point of order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$	5.5.2

Table 1.3: Convergence results for the algorithm presented in Chapter 5.

Chapter 6 explores the minimization of simple composite problems in which exact first-order information for the smooth component is unavailable. We propose a suitable definition of inexactness for a first-order oracle applied to the smooth component, incorporating a degree of uncertainty represented by a parameter $0 \leq q < 2$. Our definition is less conservative than those found in the existing literature, and it can be viewed as an interpolation between fully exact and the existing inexact first-order oracle definitions. We provide several examples that fit within this framework of an inexact first-order oracle, such as approximate gradients or weak level of smoothness, and show that, under this new definition of inexactness, we can remove the boundedness assumption of the domain of the composite problem, which is usually employed in the literature. We then consider an inexact proximal gradient algorithm based on the inexact first-order oracle and provide convergence rates for nonconvex and convex composite problems. The rates for nonconvex composite problems are $\mathcal{O}\left(k^{-1}+\delta^{\frac{2}{2-q}}\right)$ for $q \in [0,1)$ and $\mathcal{O}\left(k^{-1}+\delta k^{-q/2}+\delta^2 k^{-(q-1)}\right)$ for $q \in [1,2)$. For convex composite problems, the convergence rate is of order $\mathcal{O}\left(k^{-1}+\delta k^{-q/2}\right)$ for $q \in [0,2)$. Additionally, we derive convergence rates for a fast inexact proximal gradient algorithm for solving convex composite problems of order $\mathcal{O}(k^{-2}+\delta k^{-(3q-2)/2})$.

One can observe that as q increases, the convergence rates improve. For the inexact proximal gradient algorithm, the power of δ in the convergence estimate is higher for $q \in (0,1]$ compared to $q=0$, while for $q \geq 1$, the coefficients of δ decrease with increasing iterations. In the case of the fast inexact proximal gradient algorithm, there's no error accumulation for $q \geq 2/3$. This suggests that choosing an inexact first-order oracle with a degree of $q > 0$ is advantageous, allowing the use of less accurate approximations for the (sub)gradient of F as q increases.

The convergence rates obtained are summarized in Table 1.4. Numerical simulations on nonconvex optimization problems, such as those in image restoration, underscore the efficacy of the inexact proximal gradient scheme, particularly as the parameter q is increased. These results support our theoretical findings, indicating that significant performance improvements can be achieved with larger values of q. The content of this chapter is based on paper [28].

	convergence rates	Theorem

nonconvex case	Gradient mapping:
	if $0 \leq q < 2$:
	$\min_{j=0:k} \|g_j+p_{j+1}\|^2 \leq \mathcal{O}\left(\frac{1}{k+1}\right)+(q+1)(2-q)L^{\frac{2-2q}{2-q}}\delta^{\frac{2}{2-q}}.$
	if $1 \leq q < 2$:
	$\min_{j=0:k} \|g_j+p_{j+1}\|^2 \leq \mathcal{O}\left(\frac{1}{k+1}\right)+\mathcal{O}\left(\frac{1}{(k+1)^{q/2}}\right)+\frac{q(2-q)\delta^2 L^{1-q}(2\Delta_0)^{q-1}}{(k+1)^{q-1}}.$
		6.3.3

convex case	$f(\hat{x}_k)-f^* \leq \mathcal{O}\left(\frac{1}{k+1}\right)+\mathcal{O}\left(\frac{1}{k^{q/2}}\right).$ Accelerated scheme: $f(y_k)-f^* \leq \mathcal{O}\left(\frac{LR^2}{k^2}\right)+\mathcal{O}\left(\frac{R^q}{k^{\frac{3q}{2}-1}}\delta\right).$	6.3.6 6.3.7

Table 1.4: Convergence results for the algorithm presented in Chapter 6.

In **Chapter 7**, we explore several applications, including power flow analysis, phase retrieval, and output feedback control problems. We demonstrate how these problems, such as steady-state power flow equations [29], phase retrieval [30, 31, 32], and output feedback control [33, 34], can be effectively framed within the context of composite optimization problems. Thus, our algorithms from previous chapters (especially the one, called RHOTA, from Chapter 5) can be used to solve such applications.

Regarding phase retrieval, RHOTA algorithm for $p=2$ corresponds to a higher-order proximal point algorithm (HOPP), and our analysis indicates that HOPP converges quickly for this application. Furthermore, RHOTA aligns with the regularized Gauss-Newton algorithm presented in [35] for $p=1$, when addressing the power flow analysis problem. Finally, we present numerical simulations demonstrating the effectiveness of RHOTA algorithm in solving output feedback control problems.

Our numerical results on various IEEE bus cases [36], image recovery from handwritten digits in the MNIST library [37], and output feedback control for linear systems from the COMPl$_e$ib library [34] indicate the superior performance of our proposed algorithms compared to some state-of-the-art optimization approaches [31, 38] and specialized software [33] designed for these applications.

1.2 Collaborations

This PhD project is part of the Marie Skłodowska-Curie Action (MSCA) Innovative Training Network (ITN) TraDE-OPT (ID 861137). Following the mobility requirements of ITNs, this project included a three-month secondment from September to November 2021, during which I collaborated with Professor Silvia Villa from Università degli Studi di Genova, Italy, to develop higher-order methods for structured nonconvex optimization problems. Subsequently, from April to July 2022, I completed a secondment at Université catholique de Louvain, Belgium, under the supervision of Professor François Glineur. During this time, we introduced and analyzed an inexact first-order oracle for solving composite minimization problems. The results of this collaboration are presented in Chapter 6. Finally, between November and December 2023, I undertook two months industrial training at the company N-SIDE, Belgium. During this internship and in collaboration with Mehdi Madani and Pierre Artoisenet, our main objective was to develop efficient optimization techniques for solving steady-state power flow equations, with a specific focus on the European high-voltage transmission network. Specifically, we aim to adapt and extend the algorithm introduced in [35, 27] to handle large-scale problems effectively (this is work in progress).

1.3 Publications

The findings presented in this thesis are contained in the following papers:

- Journal papers:

    1. **Y. Nabou** and I. Necoara, *Efficiency of higher-order algorithms for minimizing composite functions*, Computational Optimization and Applications, 87: 441–473, 2023 (DOI: 10.1007/s10589-023-00533-9).

    2. **Y. Nabou**, F. Glineur and I. Necoara, *Proximal gradient methods with inexact oracle of degree q for composite optimization*, Optimization Letters, 2024 (DOI: 10.1007/s11590-024-02118-9).

    3. **Y. Nabou** and I. Necoara, *Moving higher-order Taylor approximations method for smooth constrained minimization problems*, under review in SIAM Journal on Optimization, 2023.

    4. **Y. Nabou** and I. Necoara, *Regularized higher-order Taylor approximation methods for composite nonlinear least-squares*, to be submitted.

- Conference papers:

    1. **Y. Nabou**, L. Toma and I. Necoara, Modified projected Gauss-Newton method for constrained nonlinear least-squares: application to power flow analysis, IEEE European Control Conference, Bucharest 2023 (DOI: 10.23919/ECC57647.2023.10178179).

1.4 Outline of the thesis

Chapter 2 provides first some preliminary linear algebra material followed by an in-depth overview of the theoretical aspects concerning (non)smooth and (non)convex optimization, highlighting the main relationships that play a key role in deriving our convergence rates.

In **Chapter 3**, we consider solving composite minimization problems, that involves a collection of functions which are aggregated in a nonsmooth manner through a merit function that is convex, nondecreasing, subhomogeneous and has full domain. It covers, as a particular case, smooth approximation of minimax games, minimization of max-type functions, and simple composite minimization problems, where the objective function has a nonsmooth component. We design a higher-order majorization-minimization algorithmic framework for such composite problems (possibly nonconvex). Our framework replaces each component with a higher-order surrogate such that the corresponding error function has a higher-order Lipschitz continuous derivative. We present convergence guarantees for our method for composite optimization problems with (non)convex and (non)smooth objective function. In particular, we prove stationary point convergence guarantees for general nonconvex (possibly nonsmooth) problems and under Kurdyka-Lojasiewicz (KL) property of the objective function we derive improved rates depending on the KL parameter. For convex (possibly nonsmooth) problems we also provide sublinear rates.

In **Chapter 4**, we change the settings of our problem, the goal being to minimize simple composite problems subject to nonlinear inequality constraints. We present a higher-order method for solving simple composite (non)convex minimization problems with smooth (non)convex functional constraints. At each iteration our method approximates the smooth part of the objective function and of the constraints by higher-order Taylor approximations, leading to a moving Taylor approximation method (MTA). We present convergence guarantees for MTA algorithm for both, nonconvex and convex problems. In particular, when the objective and the constraints are nonconvex functions, we prove that the sequence generated by MTA algorithm converges globally to a KKT point. Moreover, we derive convergence rates in the iterates when the problem's data satisfy the KL property. Further, when the objective function is (uniformly) convex and the constraints are also convex, we provide linear/superlinear/sublinear convergence rates for our algorithm, depending on the uniform convexity constant, respectively. Finally, we present an efficient implementation of the proposed algorithm and compare it with existing methods from the literature.

Chapter 5 develops a regularized higher-order Taylor based method for solving composite minimization problems with the merit function being convex and Lipschitz continuous (e.g., 2-norm). At each iteration, we replace each smooth component of the objective function by a higher-order Taylor approximation with an appropriate regularization, leading to a regularized higher-order Taylor approximation (RHOTA) algorithm. We derive global convergence guarantees for the RHOTA algorithm. In particular, we prove stationary point convergence guarantees for general composite problems, and leveraging the Kurdyka-Lojasiewicz (KL) property of the objective function, we derive improved rates depending on the KL parameter. When the Taylor approximation is of order 2, we present an efficient implementation of RHOTA algorithm, demonstrating that the resulting nonconvex subproblem can be effectively solved utilizing standard convex programming tools. Furthermore, we extend the scope of our investigation to include the behavior and efficacy of RHOTA algorithm in handling systems of nonlinear equations and optimization problems with nonlinear equality constraints and derive convergence rates specific for each class of problems.

In **Chapter 6**, we introduce the concept of inexact first-order oracle of degree q for nonconvex and nonsmooth functions, which naturally appears in the context of approximate gradient, weak level of smoothness and other situations. Our definition is less conservative than those found in the existing literature, and it can be viewed as an interpolation between fully exact and the existing inexact first-order oracle definitions. We analyze the convergence behavior of a (fast) inexact proximal gradient method using such an oracle for solving (non)convex composite minimization problems. We derive complexity estimates and study the dependence between the accuracy of the oracle and the desired accuracy of the gradient or of the objective function. Our results show that better rates can be obtained both theoretically and in numerical simulations when q is large.

In **Chapter 7**, we explore a diverse range of applications to demonstrate the effectiveness and efficiency of the optimization algorithms developed in the previous chapters. Our first case study involves power systems, showcasing how our algorithms can be used to solve the power flow analysis problem. Next, we turn to phase retrieval, a crucial problem in fields like optics and signal processing, where our algorithms excel at reconstructing signals with minimal error and computational overhead. Finally, in the realm of control systems, we illustrate how our numerical algorithms can be applied for solving the output feedback control problem, which is known to be a numerically challenging problem in control theory. These applications underscore the practical impact of our optimization techniques across multiple domains.

Finally, in **Chapter** 8, we present some conclusions and outline some potential new research directions.

2 Notations and preliminaries

This chapter provides the fundamental definitions and essential mathematical tools needed for developing the results presented in this thesis. In Section 2.1, we introduce some basic notations from linear algebra. Section 2.2 contains definitions and results from mathematical analysis that are vital for the optimization theory explored in this thesis. In Section 2.3, we address the basic definitions and results related to the generalization of derivatives. Finally, Section 2.4 presents the fundamental definitions and results relevant to nonlinear programming problems.

2.1 Fundamental components of linear algebra

We use the following notions:

- $\mathbb{R}$ - the set of real numbers and $\bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$
- $\mathbb{C}$ - the set of complex numbers
- x^T - the transpose of a vector $x \in \mathbb{R}^n$
- x^H - the Hermitian transpose of a vector $x \in \mathbb{C}^n$, i.e., $x^H = \bar{x}^T$.
- I_n - $n \times n$ identity matrix
- $\|\mathrm{A}\|_F = \sqrt{\mathrm{tr}(AA^T)} = \sqrt{\sum_{i=1}^n \sum_{j=1}^m |a_{ij}|^2}$ - the Frobenius norm of $A \in \mathbb{R}^{n\times m}$, where a_{ij} is the (i,j) entry of matrix A.

We denote a finite-dimensional real vector space with $\mathbb{E}$ and $\mathbb{E}^*$ its dual space composed of linear functions on $\mathbb{E}$. For any linear function $s \in \mathbb{E}$, the value of s at a point x is denoted by $\langle s, x\rangle$. Using a self-adjoint positive-definite operator $B : \mathbb{E} \to \mathbb{E}^*$ (notation $B = B^* \succ 0$), we can endow these spaces with the following *conjugate Euclidean norms*:

$$\|x\| = \langle Bx, x\rangle^{1/2} \quad \forall x \in \mathbb{E}, \quad \|s\|_* = \left\langle s, B^{-1}s\right\rangle^{1/2} \quad \forall s \in \mathbb{E}^*.$$

In the case when $\mathbb{E} = \mathbb{R}^n$ and $B = I_n$ we recover the usual scalar product and the Euclidean norm, i.e., $\langle x, y\rangle = x^T y$ and $\|x\| = \sqrt{\langle x, x\rangle}$ for all $x, y \in \mathbb{R}^n$. A well-known inequality that follows from the previous definition is the Cauchy-Schwarz:

$$|\langle s, x\rangle| \leq \|s\|_*\|x\| \quad \forall s \in \mathbb{E}^*, x \in \mathbb{E}.$$

2.2 Some tools for optimization

Consider a feasible set $X \subseteq \mathbb{E}$ and an objective function $f : \mathbb{E} \to \bar{\mathbb{R}}$. The goal is to find a point within X where the function f reaches its minimum value. Mathematically, this optimization problem can be formulated as follows:

$$\min_{x\in X} f(x). \tag{2.1}$$

For this optimization problem, we have:

- The problem dimension is defined by the dimension of $\mathbb{E}$, and denoted by $\dim(\mathbb{E})$.
- The domain of f is defined by $\operatorname{dom} f := \{x \in \mathbb{E} : \ f(x) < \infty\}$.
- The function f is proper if $\forall x \in \operatorname{dom} f$, we have $f(x) > -\infty$.
- The function f is lower semicontinuous at $x_0 \in \operatorname{dom} f$ if

$$\liminf_{x \to x_0} f(x) \geq f(x_0).$$

- The epigraph of a function f is defined as

$$\operatorname{epi} f := \{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} : f(x) \leq \alpha\}.$$

- A feasible point is any point that belongs to X.
- A global optimal solution is a feasible point $x^* \in X$ such that:

$$f(x^*) \leq f(x) \ \ \forall x \in X.$$

- A local optimal solution is a feasible point x^* for which there exists $r > 0$ satisfying:

$$f(x^*) \leq f(x), \ \ \forall x \in \mathbf{B}(x^*, r) \cap X, \ \text{where } \mathbf{B}(x^*, r) = \{x \in \mathbb{E} : \|x - x^*\| \leq r\}.$$

- The level set of f at a given x_0 is denoted by:

$$\mathcal{L}_f(x_0) = \{x \in \mathbb{E} : \ f(x) \leq f(x_0)\}.$$

- The indicator function of a set $X \subseteq \mathbb{E}$ is defined as:

$$1_X(x) := \begin{cases} 0, & \text{if } x \in X \\ \infty, & \text{if } x \notin X. \end{cases}$$

- The proximal mapping of a proper convex function f is defined by

$$\operatorname{prox}_{\gamma f}(x) := \arg\min_{y} f(y) + \frac{1}{2\gamma}\|y - x\|^2, \ \gamma > 0.$$

- The projection of a point x onto a given closed convex set X is defined as:

$$\operatorname{proj}_X(x) := \arg\min_{y \in X} \|x - y\|^2.$$

In this thesis, our primary focus lies within the realm of optimization problems as delineated in equation (2.1), where the function f is characterized by both nonsmooth and nonconvex properties, more precisely, f takes the following nonsmooth composite form $f(\cdot) = g(F(\cdot)) + h(\cdot)$, for appropriate functions F and h. Prior to delving into a more exhaustive exposition, it is imperative to revisit some fundamental concepts pertinent to (non)convex and (non)smooth optimization.

Definition 2.2.1 (Definition 2.1.1 in [38]). A set $X \subseteq \mathbb{E}$ is called convex if for any $x, y \in X$ and $\alpha \in [0,1]$ we have

$$\alpha x + (1-\alpha)y \in X.$$

This convexity notion can be extended to functions. For example, f is convex if $\operatorname{dom} f$ is nonempty, convex and additionally:

$$f\left(\alpha x + (1-\alpha)y\right) \leq \alpha f(x) + (1-\alpha)f(x) \quad \forall x, y \in \operatorname{dom} f \text{ and } \alpha \in [0,\, 1]\,.$$

Several equivalent characterisations of convex functions can be given. Below we provide such characterizations.

Proposition 2.2.2. *(Theorem 3.1.2 in [38]) A proper lower semicontinuous function $f : \mathbb{E} \to \bar{\mathbb{R}}$ is convex on the convex set* $\operatorname{dom} f$ *if and only if its epigraph is a convex set.*

Figure 2.1 shows an example of a convex and a nonconvex function using its epigraph.

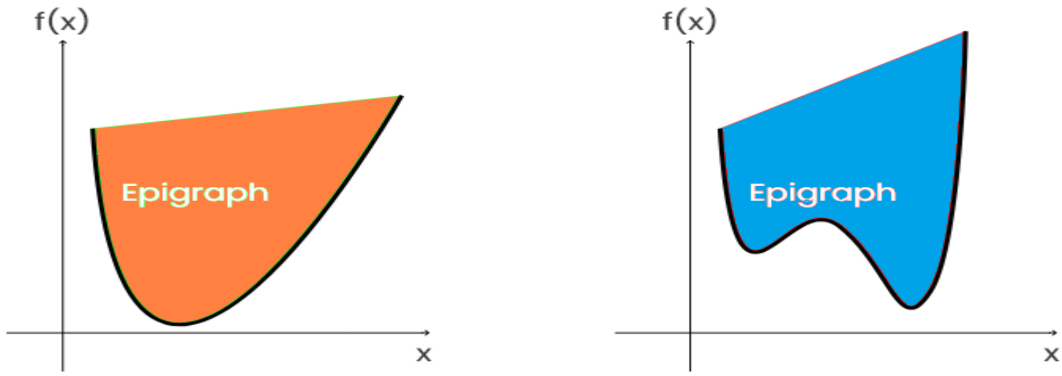

Figure 2.1: left - convex function; right - nonconvex function.

The notion of Fréchet derivative is defined as follows:

Definition 2.2.3. ([39]) Let $f : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper function with an open domain $\operatorname{dom} f$. Then, f is differentiable at $x \in \operatorname{dom} f$ if there exist a vector $\nabla f(x)$ called (Fréchet) derivative or gradient of f at x, such that:

$$\lim_{y \to 0, y \neq 0} \frac{|f(x+y) - f(x) - \langle \nabla f(x), y \rangle|}{\|y\|} = 0.$$

Similarly, one can define the Hessian (second derivative). For a twice differentiable function $f : \mathbb{E} \to \bar{\mathbb{R}}$ on a convex and open domain $\operatorname{dom} f \subseteq E$, we denote by $\nabla f(x)$ its gradient and $\nabla^2 f(x)$ its Hessian at the point $x \in \operatorname{dom} f$. When f is differentiable on its domain, the convexity is also equivalent to the fact that all its linear approximations (i.e. its first-order Taylor expansions) are below the graph of f.

Proposition 2.2.4. *(Definition 2.1.2 in [38]) Let $f : \mathbb{E} \to \bar{\mathbb{R}}$ such that* $\operatorname{dom} f$ *is convex and f is differentiable, i.e., the gradient ∇f exists at each point in* $\operatorname{dom} f$*. Then, f is convex if and only if for all $x, y \in \operatorname{dom} f$, we have:*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle.$$

Proposition 2.2.5. *(Theorem 2.1.4 in [38]) Let $f : \mathbb{E} \to \bar{\mathbb{R}}$ such that* $\mathrm{dom} f$ *is convex and f is twice differentiable, i.e., the Hessian $\nabla^2 f$ exists at each point in* $\mathrm{dom} f$. *Then, f is convex if and only if, for all $x \in \mathrm{dom} f$, we have:*

$$\nabla^2 f(x) \succeq 0.$$

We have $\nabla f(x) \in \mathbb{E}^*$ and $\nabla^2 f(x) d \in \mathbb{E}^*$ for all $d \in \mathbb{E}$. In what follows, we often work with directional derivatives of a function f at x along directions $d_i \in \mathbb{E}$ of order p:

$$D^p f(x) \left[d_1, \ldots, d_p\right].$$

For example, for a twice differentiable function f one has for any $x \in \mathrm{dom}\, f$ and $d, \bar{d} \in \mathbb{E}$ that

$$Df(x)[d] = \langle \nabla f(x), d \rangle \quad \text{and} \quad D^2 f(x)[d, \bar{d}] = \langle \nabla^2 f(x) d, \bar{d} \rangle.$$

Note that $D^p f(x)[\cdot]$ is a symmetric p multilinear form on $\mathbb{E}$. The notation $D^p f(x)[d]^p$ is used when all directions are the same, i.e., $d_1 = \cdots = d_p = d$ for some $d \in \mathbb{E}$. The norm of $D^p f(x)$ is defined in the standard way (see [12]):

$$\|D^p f(x)\| := \max_{\|d_1\|, \cdots, \|d_p\| \le 1} \left| D^p f(x) \left[d_1, \ldots, d_p\right] \right| = \max_{\|d\| \le 1} \left| D^p f(x) \left[d\right]^p \right|.$$

Note that for any fixed $x, y \in \mathrm{dom}\, f$ the form $D^p f(x) [\cdot] - D^p f(y)[\cdot]$ is also p multilinear and symmetric. Then, we define the following class of smooth functions:

Definition 2.2.6. Let $f : \mathbb{E} \to \mathbb{R}$ be $p \ge 1$ times continuously differentiable. Then, the p derivative of f is Lipschitz continuous if there exists $L_p^f > 0$ for which the following holds:

$$\|D^p f(x) - D^p f(y)\| \le L_p^f \|x - y\| \quad \forall x, y \in \mathrm{dom}\, f. \tag{2.2}$$

Next, we present several examples of functions satisfying this definition.

Example 2.2.7. Given $x_0 \in \mathbb{R}^n$ and a positive definite matrix B, which defines the norm $\|x\| = \langle Bx, x \rangle^{1/2}$ for all $x \in \mathbb{E}^n$, the function $f(x) = \|x - x_0\|^{p+1}$ with $p \ge 1$ satisfies the Lipschitz continuity condition in equation (2.2) with a Lipschitz constant $L_p^f = (p+1)!$.

Proof. This result is stated in Theorem 7.1 of [40]. For the reader's convenience, we also include the proof. Note that the following is a polynomial expression for the qth derivative of f:

$$\nabla^q f_{p+1}(x)[d]^q = \|x - x_0\|^{p+1-q} g_{q,p+1}(\tau_d(x)), \tag{2.3}$$

where $d \in \mathbb{E}$ is an arbitrary unit vector,

$$\tau_d(x) := \begin{cases} \frac{\langle B(x - x_0), d \rangle}{\|x - x_0\|}, & \text{if } x \ne x_0 \\ 0, & \text{if } x = x_0, \end{cases}$$

and the polynomial $g_{q,p+1}$ is a combination of the previous polynomial $g_{q-1,p+1}$ and its derivative $g_{q-1,p+1}^{'}$ (when $q = 0$, $g_{q,p+1}(\tau)$ is set to 1):

$$g_{q,p+1}(\tau) := \left(1 - \tau^2\right) g_{q-1,p+1}'(\tau) + (p - q + 2)\tau g_{q-1,p+1}(\tau) \quad \forall q \ge 1.$$

For (2.2) to hold it is sufficient to show that:

$$\left|\nabla^{p+1} f_{p+1}(x)[d]^{p+1}\right| \le (p+1)! \quad \forall\, x, d \in \mathbb{E}.$$

Considering (2.3), we have:

$$\nabla^{p+1} f_{p+1}(x)[d]^{p+1} = g_{p+1,p+1}(\tau_d(x)).$$

From Cauchy-Schwartz inequality we obtain that $|\tau_d(x)| \leq 1$ and therefore:

$$\left|\nabla^{p+1} f_{p+1}(x)[d]^{p+1}\right| = |g_{p+1,p+1}(\tau_d(x))| \leq \max_{\tau\in[-1,1]} |g_{p+1,p+1}(\tau)|.$$

However, by induction we can easily prove that (see also Proposition 4.5 in [40]):

$$\max_{[-1,1]} |g_{p+1,p+1}| = \prod_{i=0}^{p}(p+1-i) = (p+1)!,$$

which concludes the statement of Example 2.2.7. ■

Example 2.2.8. For given $(a_i)_{i=1}^m \in \mathbb{E}^*$, consider the log-sum-exp function:

$$f(x) = \log\left(\sum_{i=1}^m e^{\langle a_i,x\rangle}\right) \qquad \forall x \in \mathbb{E}.$$

Note that for $m = 2$ and $a_1 = 0$, we recover the logistic regression function, widely used in machine learning [3]. For $B := \sum_{i=1}^m a_i a_i^*$ (assuming $B \succ 0$, otherwise we can reduce dimensionality of the problem) we define the norm $\|x\| = \langle Bx, x\rangle^{1/2}$ for all $x \in \mathbb{E}$ and then the Lipschitz continuous condition (2.2) holds for $p = 1, 2$ and 3 with $L_1^f = 1, L_2^f = 2$ and $L_3^f = 4$, respectively.

Proof. This example has been analyzed in [41]. For completeness, we also provide the proof. Let us denote for simplicity $\kappa(x) = \sum_{i=1}^m e^{\langle a_i,x\rangle}$. Then, for all $x \in \mathbb{E}$ and $d \in \mathbb{E}$, we have:

$$\begin{aligned}
\langle \nabla f(x), d\rangle &= \frac{1}{\kappa(x)}\sum_{i=1}^m e^{\langle a_i,x\rangle}\langle a_i, d\rangle,\\
\left\langle \nabla^2 f(x)d, d\right\rangle &= \frac{1}{\kappa(x)}\sum_{i=1}^m e^{\langle a_i,x\rangle}\left(\langle a_i, d\rangle - \langle \nabla f(x), d\rangle\right)^2 \leq \sum_{i=1}^m \langle a_i, d\rangle^2 = \|d\|^2.
\end{aligned}$$

Taking maximum over $\|d\| = 1$ in the previous expression we get that $\|\nabla^2 f(x)\| \leq 1$, hence $L_1^f = 1$. Similarly, for $p = 2$ we have:

$$\begin{aligned}
\nabla^3 f(x)[d]^3 &= \frac{1}{\kappa(x)}\sum_{i=1}^m e^{\langle a_i,x\rangle}\left(\langle a_i, d\rangle - \langle \nabla f(x), d\rangle\right)^3\\
&\leq \left\langle \nabla^2 f(x)d, d\right\rangle \max_{1\leq i,j\leq m}\langle a_i - a_j, d\rangle \leq 2\|d\|^3.
\end{aligned}$$

Taking again maximum over $\|d\| = 1$ in the previous expression we obtain that $\|\nabla^3 f(x)\| \leq 2$, hence $L_2^f = 2$. Finally, for $p = 3$ we have:

$$\begin{aligned}
\nabla^4 f(x)[d]^4 &= \frac{1}{\kappa(x)}\sum_{i=1}^m e^{\langle a_i,x\rangle}\left(\langle a_i, d\rangle - \langle \nabla f(x), d\rangle\right)^4 - 3\left\langle \nabla^2 f(x)d, d\right\rangle^2\\
&\leq \nabla^3 f(x)[d]^3 \max_{1\leq i,j\leq m}\langle a_i - a_j, d\rangle \leq 4\|d\|^4.
\end{aligned}$$

Proceeding as before, i.e., taking maximum over $\|d\| = 1$ in the previous expression, we get that $\|\nabla^4 f(x)\| \leq 4$, hence $L_3^f = 4$. These prove the statements of Example 2.2.8. ■

Example 2.2.9. If the $p+1$ derivative of a function f is bounded, then the p derivative of f is Lipschitz continuous. Moreover, any polynomial of degree p (e.g., the p Taylor approximation of f, denoted T_p^f), has the p derivative Lipschitz with the Lipschitz constant zero.

Proof. Indeed, since $\nabla^p T_p^f(y;x) = \nabla^p f(x)$ (i.e., the p derivative is constant for all y), we have:

$$\|\nabla^p T_p^f(y;x) - \nabla^p T_p^f(z;x)\| = \|\nabla^p f(x) - \nabla^p f(x)\| = 0 \le L_p^{T_p^f}\|y-z\| \quad \forall y, z.$$

for any $L_p^{T_p^f} \ge 0$. Moreover, the p Taylor approximation of f has also the $p-1$ derivative Lipschitz with constant $L_{p-1}^{T_p^f} = \|\nabla^p f(x)\|$. These prove the statements of Example 3.1.1. ■

We denote the Taylor approximation of f at $x \in \operatorname{dom} f$ of order p by:

$$T_p^f(y;x) = f(x) + \sum_{i=1}^{p} \frac{1}{i!} D^i f(x)[y-x]^i \quad \forall y \in \mathbb{E}.$$

It is established that when (2.2) is satisfied, employing standard integration techniques allows for bounding the residual between the function value and its Taylor approximation [12]:

Lemma 2.2.10. *If the function $f : \mathbb{E} \to \mathbb{R}$ is p times differentiable with the p derivative is L_p^f, then we have:*

$$|f(y) - T_p^f(y;x)| \le \frac{L_p^f}{(p+1)!}\|y-x\|^{p+1} \quad \forall x, y \in \operatorname{dom} f. \tag{2.4}$$

Proof. Indeed, let us prove this relation for $p=1$ and $p=2$. For $p=1$, employing the mean value theorem in its integral form, we obtain:

$$\begin{aligned} f(y) - T_1^f(y;x) &= f(y) - f(x) - \langle \nabla f(x), y-x\rangle \\ &= \int_0^1 \langle \nabla f\left(x + t(y-x)\right), y-x\rangle \,\mathrm{dt} - \langle \nabla f(x), y-x\rangle \\ &= \int_0^1 \langle \nabla f\left(x + t(y-x)\right) - \nabla f(x), y-x\rangle \,\mathrm{dt}. \end{aligned}$$

Further, taking absolute value we obtain:

$$\begin{aligned} &|f(y) - T_1^f(y;x)| = \left|\int_0^1 \langle \nabla f\left(x+t(y-x)\right) - \nabla f(x), y-x\rangle \, dt\right| \\ &\le \int_0^1 |\langle \nabla f\left(x+t(y-x)\right) - \nabla f(x), y-x\rangle| \, dt \\ &\le \int_0^1 \|\nabla f(x+t(y-x)) - \nabla f(x)\|_* \|y-x\| dt \le \int_0^1 L_1^f t\|y-x\|^2 dt = \frac{L_1^f}{2}\|y-x\|^2, \end{aligned}$$

where the last inequality is a consequence of the Lipschitz continuity of the gradient of f. Next, let's demonstrate that the relation in (2.4) holds for $p=2$. Indeed:

$$\begin{aligned} f(y) - T_2^f(y;x) &= f(y) - f(x) - \langle \nabla f(x), y-x\rangle - \frac{1}{2}\langle \nabla^2 f(x)[y-x], y-x\rangle \\ &= \int_0^1 \langle \nabla f\left(x+t(y-x)\right) - \nabla f(x), y-x\rangle \,\mathrm{dt} - \frac{1}{2}\langle \nabla^2 f(x)[y-x], y-x\rangle \\ &= \int_0^1 \int_0^1 \langle \nabla^2 f\left(x+ts(y-x)\right) t[y-x], y-x\rangle \,\mathrm{ds}\,\mathrm{dt} - \frac{1}{2}\langle \nabla^2 f(x)[y-x], y-x\rangle \end{aligned}$$

$$= \int_0^1 t \int_0^1 \langle \nabla^2 f\left(x + ts(y-x)\right)[y-x] - \nabla^2 f(x)[y-x], y-x \rangle \, \mathrm{ds}\, \mathrm{dt},$$

where we have utilized the mean value theorem in its integral form twice. Similarly, taking the absolute value on both sides, we get:

$$\begin{aligned}
|f(y) - T_2^f(y;x)| &= \left| f(y) - f(x) - \langle \nabla f(x), y-x \rangle - \frac{1}{2}\langle \nabla^2 f(x)[y-x], y-x \rangle \right| \\
&\leq \int_0^1 t \int_0^1 \left| \langle \nabla^2 f\left(x + ts(y-x)\right)[y-x] - \nabla^2 f(x)[y-x], y-x \rangle \right| \, \mathrm{ds}\, \mathrm{dt} \\
&= \int_0^1 t \int_0^1 \left| D^2 f\left(x + ts(y-x)\right)[y-x]^2 - D^2 f(x)[y-x]^2 \right| \, \mathrm{ds}\, \mathrm{dt} \\
&= \int_0^1 t\|y-x\|^2 \int_0^1 \left| D^2 f\left(x + ts(y-x)\right)\frac{[y-x]^2}{\|y-x\|^2} - D^2 f(x)\frac{[y-x]^2}{\|y-x\|^2} \right| \, \mathrm{ds}\, \mathrm{dt} \\
&\leq \int_0^1 t\|y-x\|^2 \int_0^1 \left\| D^2 f\left(x + ts(y-x)\right) - D^2 f(x) \right\| \, \mathrm{ds}\, \mathrm{dt}.
\end{aligned}$$

Using the fact that f has the second derivative Lipschitz, we get:

$$\begin{aligned}
\left| f(y) - T_2^f(y;x) \right| &= \int_0^1 t\|y-x\|^2 \int_0^1 L_2 ts\, \|y-x\| \, \mathrm{ds}\, \mathrm{dt} \\
&\leq L_2 \|y-x\|^3 \int_0^1 t^2 \mathrm{dt} \int_0^1 s \, \mathrm{ds} = \frac{L_2}{6}\|y-x\|^3.
\end{aligned}$$

Following the same strategy, we can derive (2.4) for any $p \geq 1$. ■

By employing analogous reasoning for the functions $\langle \nabla f(x), h \rangle$ and $\langle \nabla^2 f(x) h, h \rangle$, with a fixed direction $h \in E$, we arrive at the following inequalities, applicable to all $x, y \in \operatorname{dom} f$ and $p \geq 2$, as detailed in [12]:

$$\|\nabla f(y) - \nabla T_p^f(y;x)\|_* \leq \frac{L_p^f}{p!}\|y-x\|^p, \tag{2.5}$$

$$\|\nabla^2 f(y) - \nabla^2 T_p^f(y;x)\| \leq \frac{L_p^f}{(p-1)!}\|y-x\|^{p-1}. \tag{2.6}$$

For the Hessian, the norm defined in (2.6) corresponds to the spectral norm of the self-adjoint linear operator (i.e., maximal module of all eigenvalues computed with respect to operator B. In the context of convexity, Nesterov [12] established a significant result demonstrating that a properly regularized Taylor approximation of a convex function yields a convex multivariate polynomial.

Lemma 2.2.11 (Theorem 1 in [12]). *Suppose f is a convex function with a p-th derivative that is Lipschitz with constant L_p^f, where $p \geq 2$. Let $T_p^f(y;x)$ represent the Taylor approximation of f of order p around x. If $M_p \geq pL_p^f$, then the function:*

$$s(y;x) = T_p^f(y;x) + \frac{M_p}{(p+1)!}\|y-x\|^{p+1}$$

is convex in y.

Proof. For the reader's convenience, we include the proof, adapted from [12]. Note that

$$\nabla^2 \left(\frac{1}{p}\|x\|^p \right) = (p-2)\|x\|^{p-4} B x x^* B + \|x\|^{p-2} B \succcurlyeq \|x\|^{p-2} B. \tag{2.7}$$

For an arbitrary x and y from $\operatorname{dom} f$, and for any direction $d \in \mathbb{E}$, the following holds:

$$\langle (\nabla^2 f(y) - \nabla^2 T_p^f(y;x))[d], d\rangle \leq \|\nabla^2 f(y) - \nabla^2 T_p^f(y;x)\| \|d\|^2 \overset{(2.6)}{\leq} \frac{L_p^f}{(p-1)!}\|y-x\|^{p-1}\|d\|^2.$$

This implies that:

$$\nabla^2 f(y) - \nabla^2 T_p^f(y;x) \preccurlyeq \frac{L_p^f}{(p-1)!}\|y-x\|^{p-1} \cdot B. \tag{2.8}$$

Additionally, based on the convexity of f, we can infer:

$$\begin{aligned} 0 \preccurlyeq \nabla^2 f(y) &\overset{(2.8)}{\preccurlyeq} \nabla^2 T_p^f(y;x) + \frac{L_p^f}{(p-1)!}\|y-x\|^{p-1} B \\ &\overset{(2.7)}{\preccurlyeq} \nabla^2 T_p^f(y;x) + \frac{pL_p^f}{(p+1)!}\nabla^2(\|y-x\|^{p+1}) \\ &\preccurlyeq \nabla^2 T_p^f(y;x) + \frac{M_p}{(p+1)!}\nabla^2(\|y-x\|^{p+1}) = \nabla^2 s(y;x). \end{aligned}$$

Thus, $s(y;x)$ is a convex function in the first component y. ■

Definition 2.2.12. A function $g : \mathbb{R}^m \to \mathbb{R}$ is said to be nondecreasing if for all $i = 1 : m$, g is nondecreasing in its ith argument, i.e., the univariate function:

$$x \mapsto g(x_1, \cdots, x_{i-1}, x, x_{i+1}, \cdots, x_m),$$

in nondecreasing. We say that g is homogeneous if:

$$g(\alpha x) \leq \alpha g(x) \quad \forall x \in \mathbb{R}^m \quad \forall \alpha \geq 1. \tag{2.9}$$

In what follows, if x and y are in $\mathbb{R}^m$, then $x \geq y$ means that $x_i \geq y_i$ for all $i = 1 : m$. Similarly, we define $x > y$. Since g is nondecreasing, then for all $x, y \in \mathbb{R}^m$ such that $x \leq y$ we have $g(x) \leq g(y)$.

Next, we recall the following classical weighted arithmetic-geometric mean inequality: if a, b are positive constants and $0 \leq \alpha_1, \alpha_2 \leq 1$, such that $\alpha_1 + \alpha_2 = 1$, then $a^{\alpha_1} b^{\alpha_2} \leq \alpha_1 a + \alpha_2 b$. For $\rho > 0$, $a = \rho\|x-y\|^2$, $b = \frac{\delta_q^{\frac{2}{2-q}}}{\rho^{\frac{q}{2-q}}}$, $\alpha_1 = \frac{q}{2}$ and $\alpha_2 = \frac{2-q}{2}$ we have:

$$\delta_q \|x-y\|^q \leq \frac{q\rho\|x-y\|^2}{2} + \frac{(2-q)\delta_q^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}. \tag{2.10}$$

2.3 Generalization of the derivative

A key notion in variational analysis is that of subdifferential, which extends the one of derivative for functions which are not differentiable. Hence, let us recall some definitions concerning subdifferential calculus.

Definition 2.3.1. (Definition 3.1.5 in [38]) Let $f : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper lower semicontinuous convex function. Then, a vector $g_x \in \mathbb{E}^*$ is called a subgradient of f at the point $x \in \operatorname{dom} f$ if for any $y \in \operatorname{dom} f$ one has:

$$f(y) \geq f(x) + \langle g_x, y - x\rangle.$$

The set of all subgradients of f at x is denoted by $\partial f(x)$.

Example 2.3.2. Consider the univariate convex function $f(x) = |x|$. Then, we have: $\partial f(0) = [-1, 1]$. Refer to Figure 2.2 for an illustration of this example.

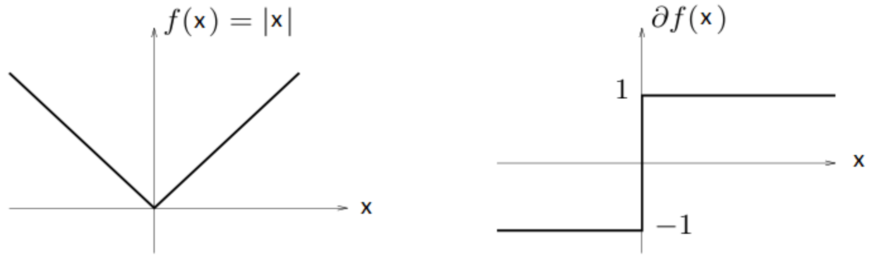

Figure 2.2: left - the absolute value function; right - its subdifferential $\partial f(x)$ as a function of x.

Below, we present the extension of the notion of subgradient for nonconvex functions.

Definition 2.3.3. (Definition 8.3 in [39]) Let $f : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper lower semicontinuous function. For a given $x \in \operatorname{dom} f$, the regular subdifferential of f at x, written $\hat{\partial} f(x)$, is the set of all vectors $g_x \in \mathbb{E}^*$ satisfying:

$$\liminf_{x \neq y \, y \to x} \frac{f(y) - f(x) - \langle g_x, y - x \rangle}{\|x - y\|} \geq 0.$$

When $x \notin \operatorname{dom} f$, we set $\hat{\partial} f(x) = \emptyset$. The limiting subdifferential, or simply the subdifferential, of f at $x \in \operatorname{dom} f$, written $\partial f(x)$, is defined as:

$$\partial f(x) := \left\{ g_x \in \mathbb{E}^* : \exists x^k \to x, f(x^k) \to f(x) \text{ and } \exists g_x^k \in \hat{\partial} f(x^k) \ \text{ such that } \ g_x^k \to g_x \right\}.$$

The horizon subdifferential of f at $x \in \operatorname{dom} f$, written $\partial^\infty f(x)$, is defined as:

$$\partial^\infty f(x) := \left\{ g_x \in \mathbb{E}^* : \exists x^k \to x, f(x^k) \to f(x) \text{ and } \exists g_x^k \in \hat{\partial} f(x^k) \ \text{ s.t.: } \ \lambda^k g_x^k \to g_x, \ \lambda^k \searrow 0 \right\}.$$

Example 2.3.4. (Example 4.7.(ii) in [42]) Let $f : \mathbb{R} \to \bar{\mathbb{R}}$ defined by $f(x) = \log(1 + |x|)$. Then, we have $\partial^\infty f(x) = \{0\}$ for all x. Additionally, we have:

$$\partial f(x) = \hat{\partial} f(x) = \begin{cases} \left\{ \dfrac{1}{1+x} \right\}, \text{ if } x > 0, \\ \left\{ \dfrac{-1}{1-x} \right\}, \text{ if } x < 0, \\ [-1, 1], \quad \text{ if } x = 0, \end{cases}$$

Example 2.3.5. (Page 304 in [39]) Let f be defined by:

$$f(x) = \begin{cases} x^2 \sin\left(\dfrac{1}{x}\right), \text{ if } x \neq 0, \\ 0, \text{ if } x = 0. \end{cases}$$

Then, we have:

$$\hat{\partial} f(0) = \partial^\infty f(0) = \{0\}, \ \partial f(0) = [-1, 1].$$

Note that $\hat{\partial} f(x) \subseteq \partial f(x)$, both sets $\hat{\partial} f(x)$ and $\partial f(x)$ are closed, and additionally, $\hat{\partial} f(x)$ is convex. If f is differentiable at x, then $\hat{\partial} f(x) = \{\nabla f(x)\} \subseteq \partial f(x)$. However, if f is continuously differentiable on a neighborhood of x, then we have equality, i.e., $\partial f(x) = \{\nabla f(x)\}$. Note also that when f is proper lower semicontinuous and convex function, then the regular subdifferential, the limiting subdifferential and the set of the subgradients of f at any point $x \in \operatorname{dom} f$ coincides [39, 43].

2.3.1 Normal cones

Let set $C \subseteq \mathbb{E}$. The regular normal cone to C at $\bar{x} \in C$ is defined by [39, 43]:

$$\hat{\mathcal{N}}_C(\bar{x}) := \left\{ w \in \mathbb{E}^* : \limsup_{x \underset{C}{\rightarrow} \bar{x}} \frac{\langle w, x - \bar{x} \rangle}{\|x - \bar{x}\|} \leq 0 \right\},$$

where the symbol $x \underset{C}{\rightarrow} \bar{x}$ indicates that $x \rightarrow \bar{x}$ with $x \in C$. The limiting normal cone to C at $\bar{x}$ is defined as [39, 43]:

$$\mathcal{N}_C(\bar{x}) = \left\{ w \in \mathbb{E}^* : \exists\, \bar{x}_k \underset{C}{\rightarrow} \bar{x}, w_k \rightarrow w \text{ as } k \rightarrow \infty \text{ with } w_k \in \hat{\mathcal{N}}_C(x_k) \right\}.$$

Figure 2.3 below provides examples that illustrate when normal cones coincide with regular normal cones and when they do not.

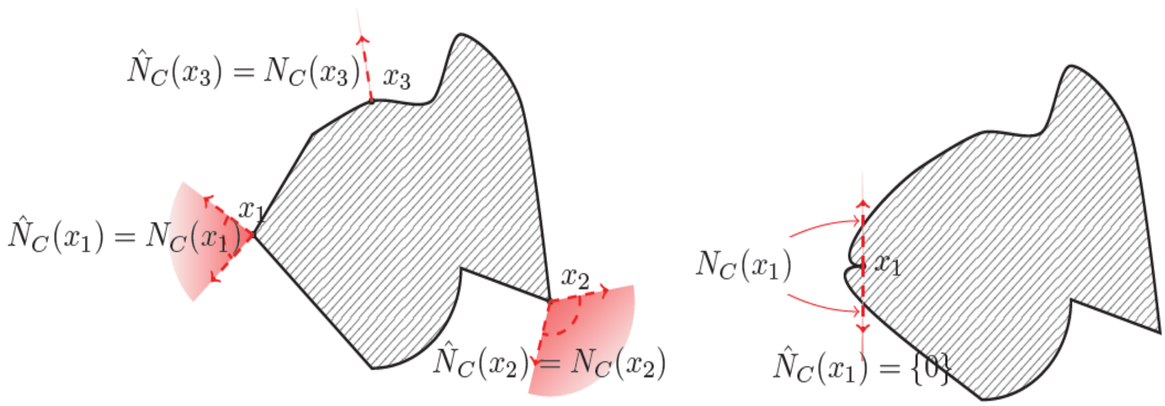

Figure 2.3: left - normal and regular normal cone coincide; right - normal and regular normal cone different (figure taken from [42]). In the left figure and at points x_1 and x_2, the regular and limiting normal cones are both represented by the space highlighted in red. At point x_3, both the regular and limiting normal cones are equal to the half-line indicated by the red vector. However, in the right figure, the regular normal cone is a singleton (i.e., set containing only 0), while the limiting normal cone is the line along the red vector.

When C is a nonempty closed convex set, the regular and the normal cones coincide [39, 43] (see Theorem 6.9 in [39]).

2.3.2 Subdifferential and normal cones

Subdifferential has a direct connection to the normal cone due to the variational geometry of epigraphs. For any $x \in \operatorname{dom} f$, the subdifferential and the horizon subdifferential of lsc function

f at x can be defined via the limiting normal cone [39, 43] (see Theorem 8.9 in [39] as well as Definition 1.78 in [43]):

$$\hat{\partial} f(x) = \left\{ g_x \in \mathbb{E}^* \colon (g_x, -1) \in \hat{\mathcal{N}}_{\text{epi}f}(x, f(x)) \right\}, \tag{2.11}$$

$$\partial f(x) = \left\{ g_x \in \mathbb{E}^* \colon (g_x, -1) \in \mathcal{N}_{\text{epi}f}(x, f(x)) \right\}, \tag{2.12}$$

$$\partial^\infty f(x) = \left\{ g_x \in \mathbb{E}^* \colon (g_x, 0) \in \mathcal{N}_{\text{epi}f}(x, f(x)) \right\}. \tag{2.13}$$

Obviously, $\partial^\infty f(x)$ is a cone. Below in Figure 2.4, we present an example showcasing the relation between the subdifferential and normal cone.

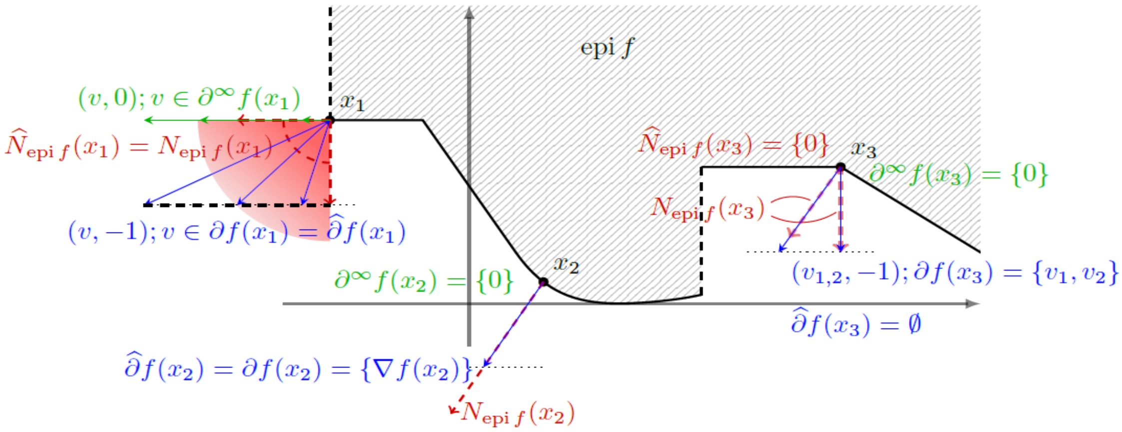

Figure 2.4: Relation between subdifferentials and normal cones (figure taken from [42]) which can be understood as follows: at the point x_1, the epigraph of the function f curves outward, resulting in the coincidence of the normal cone and the regular normal cone. Consequently, the regular subdifferential and the limiting subdifferential are also equivalent. As the function f is continuously differentiable at x_2, both the regular subdifferential and the limiting subdifferential are singleton sets, indicating that they contain only one element. At point x_3, the regular normal cone is trivial, containing only the zero vector. As a result, the regular subdifferential is also trivial, with an empty set. The general normal cone at point x_3 contains vectors originating from two distinct directions. The limiting subdifferential consists of precisely two vectors. This divergence between the regular and limiting subdifferentials at this point emphasizes the significance of the concept of normal cones in understanding the geometry of functions and their epigraphs.

Since this thesis focuses on composite problems, it's necessary to review certain chain rules for the composite functions of the form $g(F(\cdot))$, where g and F can be nonsmooth. Let's first discuss a few definitions related to regularity:

Definition 2.3.6. (Definition 8 in [44]) We say that f is regular at x_0 if for every d in $\mathbb{E}$ the usual one-sided directional derivative:

$$f^{'}(x_0; d) := \lim_{\lambda \to 0^+} \frac{f(x_0 + \lambda d) - f(x_0)}{\lambda},$$

exists and it is equal to:

$$f^*(x_0; d) := \lim \sup_{x \to x_0, \lambda \to 0^+} \frac{f(x + \lambda d) - f(x)}{\lambda}.$$

Then, we have the following chain rule:

Theorem 2.3.7. *(Theorem 6 in [44]) For the composite function $g(F)$, where $F = (F_1, \cdots, F_m)$ and g are locally Lipschitz, F_i's are regular at x, g is regular at $F(x)$ and $\partial g(F(x)) \subseteq \mathbb{R}^m_+$. Then, we have the following rule:*

$$\partial(g \circ F)(x) = co\left\{\sum_{i=1}^{m} u_i v_i \mid u \in \partial g\big(F(x)\big),\ v_i \in \partial F_i(x),\ i = 1 : m\right\}. \tag{2.14}$$

As a consequence, if g is the identity function and $m = 2$, then:

$$\partial(F_1 + F_2)(x) = \partial F_1(x) + \partial F_2(x).$$

Finally, if F is continuously differentiable at $x \in \operatorname{dom} F$, then [39][Proposition 8.12]:

$$\partial(F + f)(x) = \nabla F(x) + \partial f(x).$$

For any $x \in \operatorname{dom} f$ let us define:

$$S_f(x) = \operatorname{dist}\big(0, \partial f(x)\big) := \inf_{g_x \in \partial f(x)} \|g_x\|.$$

If $\partial f(x) = \emptyset$, we set $S_f(x) = \infty$. Further, if F is convex function and g is convex increasing, then $g(F)$ is a convex function. Indeed, let $x, y \in \operatorname{dom} g(F)$ and $\alpha \in [0, 1]$, then we get:

$$g(F(\alpha x + (1-\alpha)y)) \leq g(\alpha F(x) + (1-\alpha)F(y)) \leq \alpha g(F(x)) + (1-\alpha)g(F(y)),$$

where the first inequality follows by combining the convexity of F and that g is increasing, and the last inequality follows from convexity of g.

In certain situations, the assumption that f is convex is inadequate, necessitating the consideration of a more robust concept. Thus, let's introduce the concept of uniformly convex functions [19, 10], which will be central to the local convergence analysis of the algorithms discussed in this thesis, specifically in the context of convex optimization:

Definition 2.3.8. A function $f : \mathbb{E} \to \bar{\mathbb{R}}$ is *uniformly convex* of degree $\theta \geq 2$ if there exists a positive constant $\sigma_\theta > 0$ such that:

$$f(y) \geq f(x) + \langle g_x, y - x\rangle + \frac{\sigma_\theta}{\theta}\|x - y\|^\theta \quad \forall x, y \in \operatorname{dom} f, \text{and } g_x \in \partial f(x). \tag{2.15}$$

It's important to note that when $\theta = 2$ in equation (2.15), it aligns with the standard definition of a strongly convex function. A significant example of uniformly convex functions is presented below (refer to [19] for further details).

Example 2.3.9. For $\theta \geq 2$, consider the function $f(x) = \frac{1}{\theta}\|x - \bar{x}\|^\theta$, where $\bar{x}$ is a given point. Under these conditions, f is uniformly convex of degree θ, with a convexity parameter $\sigma_\theta = 2^{2-\theta}$.

2.3.3 Introduction to the Kurdyka-Łojasiewicz property

For nonconvex functions, there's a more encompassing concept than uniform convexity, referred to as the Kurdyka-Łojasiewicz (KL) property. This property accounts for various types of local geometries that nonconvex functions might display [45]. To explore this concept, let's start by revisiting the definition of semi-algebraic functions.

Definition 2.3.10. (Definition 5.1 in [46]) A set $X \subseteq \mathbb{R}^n$ is called semi-algebraic if there exists a finite number of real polynomial functions $p_{i,j}, q_{i,j} : \mathbb{R}^n \to \mathbb{R}$ such that:

$$X = \bigcup_{i=1}^{m} \bigcap_{j=1}^{n} \{x : \ p_{i,j}(x) = 0 \text{ and } q_{i,j}(x) \leq 0\}.$$

A function $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ is called semi-algebraic if its graph:

$$\{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} : \ f(x) = \alpha\},$$

is a semi-algebraic set.

Definition 2.3.11. A proper lower semicontinuous function $f : \mathbb{E} \to \bar{\mathbb{R}}$ satisfies *Kurdyka-Lojasiewicz (KL)* property on the compact set $\Omega \subseteq \operatorname{dom} f$ on which f takes a constant value f_* if there exist $\delta, \epsilon > 0$ such that one has:

$$\kappa'(f(x) - f_*) \cdot S_f(x) \geq 1 \quad \forall x\colon \operatorname{dist}(x, \Omega) \leq \delta, \ f_* < f(x) < f_* + \epsilon,$$

where $\kappa : [0, \epsilon] \to \mathbb{R}$ is concave differentiable function satisfying $\kappa(0) = 0$ and $\kappa' > 0$.

The KL property holds for a large class of functions including semi-algebraic functions (e.g., real polynomial functions), vector or matrix (semi)norms (e.g., $\|\cdot\|_p$ with $p \geq 0$ rational number), logarithm functions, exponential functions, and uniformly convex functions, see [45] for a comprehensive list. In particular, the max (sup) of semi-algebraic functions is a semi-algebraic function, see [46] (Example 2). For example, if f is semi-algebraic function, then we have $\kappa(t) = \sigma_q^{\frac{1}{q}} \frac{q}{q-1} t^{\frac{q-1}{q}}$, with $q > 1$ and $\sigma_q > 0$ [47]. Then, the KL property establishes the following local geometry of the nonconvex function f around a compact set Ω:

$$f(x) - f_* \leq \sigma_q S_f(x)^q \quad \forall x\colon \ \operatorname{dist}(x, \Omega) \leq \delta, \ f_* < f(x) < f_* + \epsilon. \tag{2.16}$$

Example 2.3.12. (Example 4.15 in [48]) The functions $f(x) = -\ln(1 - \|x\|^p)$ and $f(x) = \tan(\|x\|^p)$ satisfies the KL property with $\kappa(t) = pt^{\frac{1}{p}}$.

Note that the relevant aspect of the KL property is when Ω is a subset of stationary points for f, i.e. $\Omega \subseteq \{x : 0 \in \partial f(x)\}$, since it is easy to establish the KL property when Ω is not related to stationary points.

2.4 Nonlinear programming problems

Nonlinear programming (NLP) involves the process of solving optimization problems where the objective function or the constraints, or both, are nonlinear. It encompasses a wide range of problems in mathematics, engineering, economics, and other fields, where linear models do not suffice to capture the complexity of real-world scenarios. A general nonlinear programming problem takes the following form [49]:

$$\begin{aligned} &\min_x \ f(x) && (2.17)\\ &\text{s.t.: } F_i(x) \leq 0, \ i = 1:m, \qquad G_i(x) = 0, \ i = 1:p, \end{aligned}$$

where $f(x)$ is the objective function, $F_i(x)$, with $i = 1 : m$, represents the inequality constraints, and $G_j(x)$, with $j = 1 : p$, represents the equality constraints and all the functions are assumed differentiable. Depending on the context, the goal is to find a point x^* that minimizes $f(x)$ while satisfying all constraints.

2.4.1 KKT Conditions

The Karush-Kuhn-Tucker (KKT) conditions provide necessary conditions for a solution to be optimal in constrained optimization problems. For convex problems, they are also sufficient. The KKT conditions involve both the primal variables (the optimization variable x) and dual variables (Lagrange multipliers). Given the optimization problem with inequality and equality constraints (2.17), the KKT conditions consist of the following:

- Stationarity condition

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla F_i(x^*) + \sum_{j=1}^{p} \mu_j^* \nabla G_j(x^*) = 0,$$

  where λ_i^* are the Lagrange multipliers for the inequality constraints, and μ_j^* are for the equality constraints.

- Primal Feasibility:

$$F_i(x^*) \leq 0, \quad G_j(x^*) = 0 \quad \forall i = 1:m,\ j = 1:p.$$

- Dual Feasibility:

$$\lambda_i^* \geq 0 \quad \forall i = 1:m.$$

- Complementary Slackness:

$$\lambda_i^* \cdot G_i(x^*) = 0 \quad \forall i = 1:m.$$

If there exist the multipliers λ^* and μ^* bounded such that these conditions hold, then x^* is a Karush-Kuhn-Tucker (KKT) point for problem (2.17).

2.4.2 Constraint qualifications conditions

Constraint qualifications are conditions that ensure the validity of the KKT conditions in nonlinear programming. Without certain constraint qualifications, the KKT conditions might not hold, even if a solution seems to meet the optimality criteria. Let's discuss the key constraint qualifications, their role in ensuring that KKT conditions lead to correct optimality conclusions, and how they relate to convex and nonconvex problems.

Constraint qualifications provide the conditions under which the KKT conditions become valid necessary conditions for optimality. They help avoid pathological cases where the KKT conditions might not be applicable due to complex or irregular constraint configurations.

Here are some common constraint qualifications:

- Linear Independence Constraint Qualification (LICQ): this condition states that the gradients of the active constraints at a solution must be linearly independent. Formally, at a point x^*, the set of gradients $\{\nabla F_i(x^*) \mid i = 1:m \text{ s.t. } F_i(x^*) = 0\} \cup \{\nabla G_j(x^*) \mid j = 1:p\}$ should be linearly independent.

- Mangasarian-Fromovitz Constraint Qualification (MFCQ): requires that the gradients of the equality constraints be linearly independent, and there exists a nonzero feasible direction d such that for all active inequality constraints their gradients $\nabla Fi(x^*)$ are negative along that direction and additionally the gradients of all equality constraints are linearly independent and zero along the same direction. Formally, at a point x^*, there exists d such that $\nabla F_i(x^*)^T d < 0 \ \forall i = 1 : m$ s.t. $F_i(x^*) = 0$, and $\nabla G_j(x^*)^T d = 0 \ \forall j = 1 : p$, with $\nabla G_j(x^*) \ \forall j = 1 : p$ linearly independent. Note that MFCQ is more general than LICQ (since LICQ implies MFCQ).

- Slater's Condition: this condition is specific to convex problems and requires that there exists a strictly feasible point, meaning a point where all nonlinear inequality constraints are strictly satisfied and all equality constraints hold. It ensures the existence of interior points, aiding in the proof of convexity-related optimality results.

Constraint qualifications are critical because if they are not satisfied, then the KKT conditions might not be valid. This could lead to situations where optimization algorithms fail or yield incorrect results. Let us recall that a local minimum satisfying the Linear Independence Constraint Qualification (LICQ) or the Mangasarian-Fromovitz Constraint Qualification (MFCQ) is also a Karush-Kuhn-Tucker (KKT) point.

Theorem 2.4.1. *(Theorem 12.1 in [49]) Let x^* be a local minimum of* (2.17) *and assume that LICQ (or MFCQ) holds at x^*. Then, x^* is a KKT point for the problem* (2.17).

In convex optimization, satisfying a constraint qualification like Slater's condition ensures that KKT points are also global optima. Hence for convex problems, the KKT conditions become both necessary and sufficient for optimality, provided that the Slater's condition holds.

2.4.3 Approximate KKT conditions

Typically, theorems that underlie an optimality condition have the following structure: if a local minimizer x^* satisfies some certain constraint qualifications, then x^* is a KKT point. In another way, standard first-order necessary conditions for optimality are generally expressed as either the presence of KKT conditions or the absence of constraint qualifications. The implementations of most practical algorithms designed to solve large-scale nonlinear programming problems incorporate stopping criteria that signal when the current iterate is sufficiently close to an optimum. In most cases, computer codes check for approximate Karush-Kuhn-Tucker (AKKT) conditions. AKKT conditions refer to a situation where the KKT conditions are approximately satisfied. This arises in practice when exact KKT points are hard to compute due to numerical issues, high computational costs, or iterative methods that stop before achieving perfect accuracy. In an AKKT point, x^*_ϵ, the stationarity, primal feasibility, dual feasibility, and complementary slackness conditions are met within some tolerance $\epsilon > 0$. Specifically:

- Approximate Stationarity: The norm of the stationarity condition is less than ϵ:

$$\|\nabla f(x^*_\epsilon) + \sum_{i=1}^{m} \lambda_i^* \nabla F_i(x^*_\epsilon) + \sum_{j=1}^{p} \mu_j^* \nabla G_j(x^*_\epsilon)\| \leq \epsilon,$$

- Approximate Primal Feasibility: All inequality constraints are violated by no more than ϵ, and all equality constraints are met within ϵ:

$$\|F(x^*_\epsilon)\| \leq \epsilon, \quad \|\max(G(x^*_\epsilon), 0)\| \leq \epsilon.$$

- Approximate Complementary Slackness: The product of Lagrange multipliers and the constraints is within ϵ:
$$\|\lambda_i^* \cdot G_i(x_\epsilon^*)\| \leq \epsilon.$$

Note that any local minimum of the problem (2.17) satisfies these AKKT conditions for some $\epsilon > 0$ without requiring any constraint qualifications conditions, see [50, 51]. Note that if a point satisfies AKKT then either it converges to a KKT point as $\epsilon \to 0$ or no constraint qualification holds at that limit point [51]. Let's illustrate this situation with an example [51]:

$$\begin{aligned}
\min_x f(x) &= \frac{(x_2-2)^2}{2}, \\
\text{s.t. : } F_1(x) &= x_1 = 0, \\
F_2(x) &= x_1 x_2 = 0,
\end{aligned}$$

which has a solution at $(0, 2)$. Now, let's consider a small ϵ and examine the point $x_\epsilon^* = (\epsilon, 1)^T$. The first-order optimality condition at x_ϵ^* is calculated as:

$$\begin{aligned}
&\left\|\nabla f(x_\epsilon^*) - \frac{1}{\epsilon}\nabla F_1(x_\epsilon^*) + \frac{1}{\epsilon}\nabla F_2(x_\epsilon^*)\right\| = \left\|(0,-1)^T - \frac{1}{\epsilon}(1,0)^T + \frac{1}{\epsilon}(1,\epsilon)^T\right\| = 0, \\
&|F_1(x_\epsilon^*)| = \epsilon \text{ and } |F_2(x_\epsilon^*)| = \epsilon.
\end{aligned}$$

Hence x_ϵ^* is a AKKT point. However, the limit point of x_ϵ^* is not a KKT point because the multipliers $-\frac{1}{\epsilon}$ and $\frac{1}{\epsilon}$ are unbounded as $\epsilon \to 0$. Moreover, no constraint qualification holds at such limit point since $\nabla F_1(x_\epsilon^*)$ and $\nabla F_2(x_\epsilon^*)$ are linearly dependent when ϵ goes to zero.

In summary, AKKT points are valuable in practice because they offer a sense of when an iterative optimization process yields an approximate KKT point. This allows practitioners to make decisions based on nearly optimal solutions, while effectively managing computational resources.

Finally, let us also recall the following lemma that will be used when analysing convergence of optimization algorithms developed in this thesis, whose proof is similar to the one in [52](Theorem 2). For completeness, we give the proof below.

Lemma 2.4.2. *Let $\theta > 0$, $C_1, C_2 \geq 0$ and $(\lambda_k)_{k\geq 0}$ be a nonnegative, nonincreasing sequence, satisfying the following recurrence:*

$$\lambda_{k+1} \leq C_1 \left(\lambda_k - \lambda_{k+1}\right)^{\frac{1}{\theta}} + C_2 \left(\lambda_k - \lambda_{k+1}\right). \tag{2.18}$$

If $\theta \leq 1$, then there exists an integer k_0 such that:

$$\lambda_k \leq \left(\frac{C_1 + C_2}{1 + C_1 + C_2}\right)^{k-k_0} \lambda_0 \qquad \forall k \geq k_0.$$

If $\theta > 1$, then there exists $\alpha > 0$ and integer k_0 such that:

$$\lambda_k \leq \frac{\alpha}{(k-k_0)^{\frac{1}{\theta-1}}} \qquad \forall k \geq k_0.$$

Proof. Note that the sequence λ_k is nonincreasing and nonnegative, thus it is convergent. Let us consider first $\theta \leq 1$. Since $\lambda_k - \lambda_{k+1}$ converges to 0, then there exists k_0 such that $\lambda_k - \lambda_{k+1} \leq 1$ and $\lambda_{k+1} \leq (C_1 + C_2)(\lambda_k - \lambda_{k+1})$ for all $k \geq k_0$. It follows that:

$$\lambda_{k+1} \leq \frac{C_1 + C_2}{1 + C_1 + C_2}\lambda_k,$$

which proves the first statement. If $1 < \theta \leq 2$, then there exists also an integer k_0 such that $\lambda_k - \lambda_{k+1} \leq 1$ for all $k \geq k_0$. Then, we have:

$$\lambda_{k+1}^{\theta} \leq (C_1 + C_2)^{\theta} (\lambda_k - \lambda_{k+1}).$$

Since $1 < \theta \leq 2$, then taking $0 < \beta = \theta - 1 \leq 1$, we have:

$$\left(\frac{1}{C_1 + C_2}\right)^{\theta} \lambda_{k+1}^{1+\beta} \leq \lambda_k - \lambda_{k+1},$$

for all $k \geq k_0$. From Lemma 11 in [19], we further have:

$$\lambda_k \leq \frac{\lambda_{k_0}}{(1 + \sigma(k - k_0))^{\frac{1}{\beta}}}$$

for all $k \geq k_0$ and for some $\sigma > 0$. Finally, if $\theta > 2$, then define $h(s)=s^{-\theta}$ and let $R > 1$ be fixed. Since $1/\theta < 1$, then there exists a k_0 such that $\lambda_k - \lambda_{k+1} \leq 1$ for all $k \geq k_0$. Then, we have $\lambda_{k+1} \leq (C_1 + C_2)(\lambda_k - \lambda_{k+1})^{\frac{1}{\theta}}$, or equivalently:

$$1 \leq (C_1 + C_2)^{\theta}(\lambda_k - \lambda_{k+1})h(\lambda_{k+1}).$$

If we assume that $h(\lambda_{k+1}) \leq Rh(\lambda_k)$, then:

$$1 \leq R(C_1 + C_2)^{\theta}(\lambda_k - \lambda_{k+1})h(\lambda_k) \leq \frac{R(C_1 + C_2)^{\theta}}{-\theta + 1}\left(\lambda_k^{-\theta+1} - \lambda_{k+1}^{-\theta+1}\right).$$

Denote $\mu = \frac{-R(C_1+C_2)^{\theta}}{-\theta+1}$. Then:

$$0 < \mu^{-1} \leq \lambda_{k+1}^{1-\theta} - \lambda_k^{1-\theta}. \tag{2.19}$$

If we assume that $h(\lambda_{k+1}) > Rh(\lambda_k)$ and set $\gamma = R^{-\frac{1}{\theta}}$, then it follows immediately that $\lambda_{k+1} \leq \gamma\lambda_k$. Since $1 - \theta$ is negative, we get:

$$\lambda_{k+1}^{1-\theta} \geq \gamma^{1-\theta}\lambda_k^{1-\theta} \quad \iff \quad \lambda_{k+1}^{1-\theta} - \lambda_k^{1-\theta} \geq (\gamma^{1-\theta} - 1)\lambda_k^{1-\theta}.$$

Since $1 - \theta < 0$, $\gamma^{1-\theta} > 1$ and λ_k has a nonnegative limit, then there exists $\bar{\mu} > 0$ such that $(\gamma^{1-\theta} - 1)\lambda_k^{1-\theta} > \bar{\mu}$ for all $k \geq k_0$. Therefore, in this case, we also obtain:

$$0 < \bar{\mu} \leq \lambda_{k+1}^{1-\theta} - \lambda_k^{1-\theta}. \tag{2.20}$$

If we set $\hat{\mu} = \min(\mu^{-1}, \bar{\mu})$ and combine (2.19) and (2.20), we obtain:

$$0 < \hat{\mu} \leq \lambda_{k+1}^{1-\theta} - \lambda_k^{1-\theta}.$$

Summing the last inequality from k_0 to k, we obtain $\lambda_k^{1-\theta} - \lambda_{k_0}^{1-\theta} \geq \hat{\mu}(k - k_0)$, i.e.:

$$\lambda_k \leq \frac{\hat{\mu}^{-\frac{1}{\theta-1}}}{(k-k_0)^{\frac{1}{\theta-1}}}$$

for all $k \geq k_0$. This concludes our proof. ∎

3 General composite higher-order algorithms

In this chapter, we introduce a higher-order algorithmic framework based on the majorization-minimization approach designed to solve composite optimization problems that involves a collection of functions which are aggregated in a nonsmooth manner through a merit function that is convex, nondecreasing, subhomogeneous and with full domain. We derive global convergence guarantees for this method when applied to composite problems with (non)convex and nonsmooth objective functions, with improved convergence rates under the Kurdyka-Łojasiewicz (KL) property. We also present an efficient implementation of the proposed method and provide numerical simulations to demonstrate its effectiveness.

The chapter is structured as follows: Section 3.1 provides a comprehensive literature review of composite higher-order methods. In Section 3.2, we introduce our general composite higher-order framework and the associated algorithm. In Section 3.3 we derive global and local convergence results for this algorithm in both convex and nonconvex scenarios. Additionally, we present an adaptive scheme that does not require prior knowledge of the Lipschitz constants. The chapter concludes with Section 3.4, where we present numerical simulations demonstrating the efficiency of our proposed scheme. The content covered in this chapter is based on the findings reported in published paper [23].

3.1 State of the art

Efficient decision-making is a innate human desire. One fundamental approach to optimizing a given function involves a step-by-step minimization of basic *models* that serve as upper bounds for the function [53, 54, 55]. Advanced modeling frameworks demand tackling a broader spectrum of nonsmooth and nonconvex optimization challenges compared to simpler problems. In this chapter, our focus is on solving the following composite minimization problem:

$$\min_x f(x) := g\big(F(x)\big) + h(x), \tag{3.1}$$

where $h : \mathbb{E} \to \bar{\mathbb{R}}$ and $F : \mathbb{E} \to \bar{\mathbb{R}}^m$ are general proper lower semicontinuous functions on their closed domains and $g : \mathbb{R}^m \to \mathbb{R}$ is a proper closed convex increasing, subhomogeneous (see Definition (2.2.12)) function defined everywhere, and $F = (F_1, \cdots, F_m)$. This formulation unifies many particular cases, such as smooth approximation of minimax games, max-type minimization problems, while recent instances include robust phase retrieval and matrix factorization problems [21, 22, 56, 57].

Example 3.1.1. (Minimax strategies for nonlinear games) Let us consider the problem:

$$\min_{x \in \triangle_n} \left\{ f(x) := \max_{u \in \triangle_m} \langle F(x), u \rangle \right\},$$

where $\triangle_n$, $\triangle_m$ are the standard simplexes in $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. For example, in matrix games that is a particular problem in game theory, the function F takes a linear form, more precisely, we have $\langle F(x), u\rangle = \langle Ax, u\rangle + \langle b, x\rangle + \langle b, u\rangle$ [58]. The smooth approximation for this problem using the entropy distance is as follows [58]:

$$\min_{x\in\triangle_n}\left\{ f_\mu(x) := \max_{u\in\triangle_m}\left\{\langle F(x), u\rangle - \mu\sum_{j=1}^{m} u_j \ln(u_j) - \mu\ln(m)\right\}\right\},$$

for some $\mu > 0$. Using Lemma 4 in [58], we get:

$$f_\mu(x) = \mu \ln\left(\sum_{j=1}^{m} e^{\frac{F_i(x)}{\mu}}\right).$$

Hence, considering $g(y) = \mu\ln\left(\sum_{j=1}^{m} e^{\frac{y_i}{\mu}}\right)$, then the original minimax problem can be approximated, for sufficiently small μ, with the composite problem of the form (3.1):

$$\min_{x\in\Delta_n} f_\mu(x) := g(F(x)).$$

Note that g is indeed convex, increasing and subhomogeneous function, hence, this smooth re-formulation fits into the composite problem (3.1).

Example 3.1.2. (Min-max problems) Let us consider the following min-max problem:

$$\min_{x\in Q}\max_{i=1:m} F_i(x).$$

This type of problem is classical in optimization and arises in many fields, from operations research to statistics and from numerical analysis to finance. Note that if we define $g(y) = \max_{i=1:m} y_i$ and $h = 1_Q$, then, the previous min-max problem can be written as problem (3.1).

Example 3.1.3. (Simple composite problems) Let us consider the following simple composite minimization problem:

$$\min_{x\in\mathbb{R}^n} F_0(x) + h(x).$$

Taking $F(x) = F_0(x)$ and g the identity function, the previous problem can be written as problem (3.1).

3.1.1 Simple composite problems

Let as analyse the last example in more details, i.e., when F is a single function and g is the identity function, problem (3.1) reduces to an optimization problems expressed as:

$$\min_x f(x) := F_0(x) + h(x), \tag{3.2}$$

with F representing the smooth component and h the nonsmooth component. Significant attention has been devoted to such problems in the domain of large-scale optimization [52, 59, 8]. This formulation, often referred to as *simple* composite optimization, demands a more sophisticated approach. This is where the proximal gradient method comes into play, offering a structured combination of gradient-based optimization and proximal operators [52, 59, 8]. When F_0 has the gradient L_1^F-Lipschitz continuous, for a given current iterate x_k, the proximal gradient method generates the next iterate to be the minimizer of the following model for a given positive constant $M > 0$:

$$x_{k+1} = \arg\min_x F_0(x_k) + \langle \nabla F_0(x_k), x - x_k \rangle + \frac{M}{2}\|x - x_k\|^2 + h(x)$$
$$\iff x_{k+1} = \operatorname{prox}_{\frac{1}{M}h}\left(x_k - \frac{1}{M}\nabla F_0(x_k)\right).$$

The proximal gradient method is a fundamental technique for traversing through additive composite minimization (i.e., sum of a smooth and nonsmooth functions) with well-defined convergence rates. The convergence rate for convex f is of order $\mathcal{O}(k^{-1})$ in function values, where k denotes the iteration counter. This means that in order to obtain an ϵ solution (i.e., $f(x_k) - f^* \leq \epsilon$) one needs to perform $\mathcal{O}(\epsilon^{-1})$ number of iterations. However, when f is nonconvex, the convergence rate towards a stationary point is of order $\mathcal{O}(k^{-\frac{1}{2}})$, [52, 59, 8]. Although proximal point strategies have demonstrated empirical efficacy in handling intricate and difficult optimization tasks, it's acknowledged that their convergence rates are slow. A natural way to speed up these convergence rates is to leverage higher-order information, specifically derivatives, i.e., to use higher-order information (derivatives) to build a higher-order (Taylor) models. For example, the classical Newton's method approximate F_0 by its second-order Taylor evaluated in a given current iteration. Namely, Newton's method for solving (3.2) is:

$$x_{k+1} = \arg\min_x F_0(x_k) + \langle \nabla F_0(x_k), x - x_k \rangle + \frac{1}{2}\langle \nabla^2 F_0(x_k)(x - x_k), x - x_k \rangle + h(x).$$

If $h(\cdot) = 0$ and the Hessian $\nabla^2 F(x_k)$ is invertible, this step has the following closed form:

$$x_{k+1} = x_k - (\nabla^2 F_0(x_k))^{-1}\nabla F_0(x_k).$$

When compared to the proximal gradient method, this scheme clearly has higher computational cost at every step, given the necessity to invert the Hessian $\nabla^2 F_0(x_k)$. Nonetheless, there is reason to believe that exploiting second-order information can significantly accelerate the convergence rate. Newton's Method is generally recognized for its local quadratic convergence, as described in standard literature [38]. More precisely, if the initial point x_0 is close to a local minimum x^*, then Newton's method has the following quadratic convergence:

$$\|x_{k+1} - x^*\| \leq c\|x_k - x^*\|^2,$$

for a given constant $c > 0$ and for all $k \geq 0$. This aspect was further extended to cover composite optimization problems [60]. This method has some potential drawbacks, for example, the Hessian matrix could be degenerated at the current iteration, leaving the approach ill-defined. Additionally, it is difficult to establish global convergence for this scheme. Several proposals have been suggested to improve this iterative process, including the Levenberg-Marquardt regularization, the damped Newton method, and the trust-region approach. Refer to [61] for a thorough analysis of various combinations and implementations of aforementioned ideas. In some recent publications [62, 63], the authors have established a significant advancement in the realm of numerical optimization by demonstrating that the incorporation of a quadratic regularization, yields a pathway to achieve global convergence within Newton's method in the convex setting. This finding unveils a promising avenue for enhancing the efficiency and reliability of iterative optimization algorithms. More precisely, the scheme presented in [62, 63] is of the form:

$$x_{k+1} = \arg\min_x F_0(x_k) + \langle \nabla F_0(x_k), x - x_k \rangle + \frac{1}{2}\langle \nabla^2 F_0(x_k)(x - x_k), x - x_k \rangle + \frac{\beta_k}{2}\|x - x_k\|^2$$
$$\iff x_{k+1} = x_k - (\nabla^2 F_0(x_k) + \beta_k I)^{-1}\nabla F_0(x_k),$$

where $\beta_k = c\sqrt{\|\nabla F_0(x_k)\|}$ for some appropriate $c > 0$. While these approaches have made significant strides in addressing the crucial issue of worst-case guarantees for the global behavior of the Newton's method within the convex setting, a notable gap remains: the absence of a comprehensive treatment of global convergence for Newton's method in the nonconvex scenario. This represents a significant challenge, as the nonconvex landscape poses unique complexities and requires different tactics to assure convergence across diverse optimization landscapes. In [10], the cubic regularization of the Newton's method is present for nonconvex problems. Paper [10] derives the first global convergence rate of cubic regularization of Newton method for unconstrained smooth nonconvex minimization problems with the hessian Lipschitz continuous (i.e., using a second-order oracle). Namely, when F_0 has the hessian, $\nabla^2 F_0$, L_2^F-Lipschitz, the scheme presented in [10] aims to solve the following cubic model:

$$\begin{aligned} x_{k+1} &= \arg\min_x T_2^{F_0}(x; x_k) + \frac{M}{6}\|x - x_k\|^3 + h(x) \\ &= \arg\min_x F_0(x_k) + \langle \nabla F_0(x_k), x - x_k\rangle + \frac{1}{2}\langle \nabla^2 F_0(x_k)(x - x_k), x - x_k\rangle + \frac{M}{6}\|x - x_k\|^3 + h(x), \end{aligned}$$

where M is positive constant. When F_0 is convex with Hessian Lipschitz continuous, paper [10] establishes a global convergence in function value of order $\mathcal{O}(k^{-2})$, while in the nonconvex setting the convergence to an approximate first-order critical point is of order $\mathcal{O}(k^{-\frac{2}{3}})$. Higher-order methods (or Tensor methods) is a natural generalization of the gradient and the Newton methods to arbitrary order. Assume that F_0 has the p derivative L_p^F-Lipschitz (with $p \geq 1$ positive integer), then the basic iteration of a higher-order method is as follows:

$$x_{k+1} = \arg\min_x T_p^{F_0}(x; x_k) + \frac{M}{(p+1)!}\|x - x_k\|^{p+1} + h(x),$$

where x_k is the current iteration and $M > 0$ is a given positive constant and recall that $T_p^{F_0}$ is the Taylor approximation of order p of the function F_0 at x_k. This idea was first proposed in the unpublished preprint [11]. As researchers delve deeper into the intricacies of optimization methods, particularly within the nonconvex setting [13, 14], a notable focus has been placed on analyzing the complexity of high-order approaches with convergence guarantees, particularly in terms of the norm of the gradient, and it has been established rate of order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$. However, the extensive complexity associated with minimising nonconvex multivariate polynomials has posed significant challenges, rendering this initial effort unsuccessful. Despite these obstacles, a ray of hope emerged through the groundbreaking research of Nesterov in [12]. Specifically, Nesterov demonstrated that by appropriately regularizing the Taylor approximation, the auxiliary subproblem remains convex and can be solved efficiently, thereby offering a promising avenue for tackling convex unconstrained smooth problems (see Lemma 2.2.11 in Chapter 2). The convergence rate of such a scheme for convex problems in function values is of order $\mathcal{O}(k^{-p})$. Recently, [20] provided a unified framework for the convergence analysis of higher-order optimization algorithms for solving simple composite optimization problem (3.2) using the *majorization-minimization* approach. This is a technique that approximates an objective function by a majorization function, which can be minimized in closed form or yielding a new point of some acceptable improvement. A full convergence analysis is performed in [20] for such an algorithmic framework and both global and local convergence rates are derived.

3.1.2 Composite problems

When optimization problems become more complex, for example, the objective function given in (3.1) is nonsmooth, see Example 3.1.2, i.e., $g(\cdot) = \max(\cdot)$, classic gradient approaches become limited. In response, subgradient approaches emerge as a versatile alternative that accommodates nonsmooth functions using generalized gradients. These methods broaden the scope of optimization to domains where classical gradients fail, providing robustness in managing functions with discontinuities or non-differentiable points. Subgradients methods generate the next iterates for a given current iterates x_k as follows:

$$x_{k+1} = x_k - \alpha_k \lambda_k,$$

where $\lambda_k \in \partial f(x_k)$ and α_k is a stepsize. When f is convex function and Lipschitz continuous, subgradient methods with $\alpha_k = \mathcal{O}(k^{-\frac{1}{2}})$ exhibit a sublinear convergence rate and the convergence rate is of order $\mathcal{O}(k^{-\frac{1}{2}})$, which is worse than the proximal gradient based methods (see Theorem 3.2 in [64]). In response to this issue, a direct extension of the prox-gradient algorithm to the entire problem class (3.1) has been proposed, which consists in linearizing the smooth part, leaving the nonsmooth term unchanged and adding an appropriate quadratic regularization term. This is the approach considered, e.g., in [56, 65], leading to a proximal Gauss-Newton method, i.e., given the current point x_k and a regularization parameter $M > 0$, solve at each iteration the subproblem:

$$x_{k+1} = \arg\min_x g\Big(F(x_k) + \nabla F(x_k)(x - x_k)\Big) + \frac{M}{2}\|x - x_k\|^2 + h(x).$$

For such a method it was proved in [56] that it converges to a near stationary point at a sublinear rate of order $\mathcal{O}(k^{-\frac{1}{2}})$, while convergence of the iterates under KL inequality was recently shown in [65]. In [21] a flexible method is proposed, where the smooth part F is replaced by its quadratic approximation, i.e., given x_k, solve:

$$x_{k+1} = \arg\min_x g\left(F(x_k) + \nabla F(x_k)(x - x_k) + \frac{M}{2}\|x - x_k\|^2\right) + h(x),$$

where $M = (M_1, \cdots, M_m)^T$, with L_i being the Lipschitz constant of the gradient of F_i, for $i = 1 : m$. Assuming F, g and h are convex functions, and g additionally is component wise nondecreasing and Lipschitz on its domain, [21] derives sublinear convergence rate of order $\mathcal{O}(k^{-1})$ in function values. One can notice that all the previously mentioned methods belong to the category of *first-order methods*. Despite their demonstrated efficacy in tackling challenging optimization problems empirically, it is well-established that their convergence speed tends to be slow. In the recent paper [22], problem (3.1) is considered, where $F = (F_1, \cdots, F_m)$, with F_i's being convex and p-smooth functions on $\mathbb{E}$ and having the p-derivative Lipschitz, with $p \geq 1$. Under these settings, [22] replaces the smooth part by its Taylor approximation of order p plus a proper regularization term, i.e., given x_k, solve the following subproblem:

$$x_{k+1} = \arg\min_x g\left(T_p^F(x; x_k) + \frac{L}{(p+1)!}\|x - x_k\|^{p+1}\right),$$

where $L = (L_1, \cdots, L_m)^T$, with L_i being related to the Lipchitz constant of the p-derivative of F_i and $T_p^F(x; x_k)$ is the p-Taylor approximation of F around the current point x_k. For such a higher-order method, in the convex settings and assuming that g has full domain in the second argument, [22] derives a sublinear convergence rate in function values of order $\mathcal{O}(k^{-p})$.

However, global complexity bounds for higher-order methods based on the majorization-minimization principle for solving composite problem (3.1) *(possibly nonconvex) are not yet given.* This is the goal of this chapter.

3.2 General composite higher-order algorithm

In this section, we propose a higher-order algorithm for solving the composite problem (3.1) and derive convergence rates.

Assumption 3.2.1. We consider the following assumptions for optimization problem (3.1):

1. The functions F_i, with $i = 1\!:\!m$, g and h are proper lower semicontinuous on their domains, satisfy the chain rule (2.14) and $\operatorname{dom} h \subseteq g(\operatorname{dom} F)$.

2. Additionally, g is convex, nondecreasing, with full domain, and subhomogeneous (see Definition 2.2.12).

3. Problem (3.1) has a solution and thus $f^* := \inf_{x \in \operatorname{dom} f} f(x) > -\infty$.

From Assumption 3.2.1(1), it follows that $\operatorname{dom} f = \operatorname{dom} h$. Moreover, if Assumption 3.2.1(2) holds, then from [22](Theorem 4) it follows that:

$$g(x + ty) \leq g(x) + tg(y) \quad \forall t \geq 0. \tag{3.3}$$

Further, let us introduce the notion of a higher-order surrogate, see also [20].

Definition 3.2.2. Let $\phi : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper lower semicontinuous function and $x \in \operatorname{dom} \phi$ (assumed closed). We call the function $s(\cdot\,; x) : \mathbb{E} \to \bar{\mathbb{R}}$, with $\operatorname{dom} s(\cdot\,; x) = \operatorname{dom} \phi$, a p higher-order surrogate of ϕ at x if it has the following properties:

(i) the error function

$$e(y; x) = s(y; x) - \phi(y), \text{ with } y \in \operatorname{dom} \phi, \tag{3.4}$$

is p differentiable on $\operatorname{dom} \phi$ and the p derivative is smooth with Lipschitz constant L_p^e.

(ii) the derivatives of the error function e satisfy

$$\nabla^i e(x; x) = 0 \quad \forall i = 0 : p, \; x \in \operatorname{dom} \phi, \tag{3.5}$$

and there exist a positive constant $R_p^e > 0$ such that

$$e(y; x) \geq \frac{R_p^e}{(p+1)!} \|y - x\|^{p+1} \quad \forall x, y \in \operatorname{dom} \phi. \tag{3.6}$$

Note that $\operatorname{dom} e$ (assumed open) usually includes strictly $\operatorname{dom} \phi$ (see examples below). Moreover, from (3.6) we have $s(y; x) \geq \phi(y)$ for all $x, y \in \operatorname{dom} \phi$. Next, we give two nontrivial examples of higher-order surrogate functions, see [20] for more examples.

Example 3.2.3. (Composite functions) Let $F_1 : \mathbb{E} \to \mathbb{R}$ be p times differentiable and the p derivative be Lipschitz with constant $L_p^{F_1}$ and let $F_2 : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper closed function. Then, for the composite function $F = F_1 + F_2$, where $\operatorname{dom} F = \operatorname{dom} F_2$, one can consider the following p higher-order surrogate function:

$$s(y; x) = T_p^{F_1}(y; x) + \frac{M_p}{(p+1)!} \|x - y\|^{p+1} + F_2(y) \;\; \forall\, x, y \in \operatorname{dom} F,$$

where $M_p > L_p^{F_1}$. Indeed, from the definition of the error function, we get:

$$e(y; x) = T_p^{F_1}(y; x) - F_1(y) + \frac{M_p}{(p+1)!} \|x - y\|^{p+1}. \tag{3.7}$$

Thus $e(\cdot;x)$, with $\operatorname{dom} e = \mathbb{E}$ and $\operatorname{dom} F \subseteq \operatorname{dom} e$, has the p derivative Lipschitz with constant $L_p^{F_1} + M_p$. Further, from the definition of the error function e, we have:

$$\nabla^i e(x;x) = \nabla T_p^{F_1}(x;x) - \nabla^i F_1(x) = \nabla^i F_1(x) - \nabla^i F(x) = 0 \ \ \forall i = 1:p.$$

Moreover, since F_1 has the p derivative Lipschitz, it follows from (2.4) that:

$$T_p^{F_1}(y;x) - F_1(y) \geq \frac{-L_p^{F_1}}{(p+1)!}\|x-y\|^{p+1}.$$

Combining this inequality with (3.7), we get:

$$e(y;x) \geq \frac{M_p - L_p^{F_1}}{(p+1)!}\|x-y\|^{p+1}. \tag{3.8}$$

Hence, the error function e has $L_p^e = M_p + L_p^{F_1}$ and $R_p^e = M_p - L_p^{F_1}$.

Example 3.2.4. (proximal higher-order) Let $F : \mathbb{E} \to \bar{\mathbb{R}}$ be a proper lower semicontinuous function. Then, we can consider the following higher-order surrogate function:

$$s(y;x) = F(y) + \frac{M_r}{(r+1)!}\|y-x\|^{r+1},$$

where r is a positive integer. Indeed, the error function is:

$$e(y;x) = s(y;x) - F(x) = \frac{M_r}{(r+1)!}\|y-x\|^{r+1},$$

where $\operatorname{dom} F \subseteq \operatorname{dom} e = \mathbb{E}$. In this case, the error function e has the r derivative Lipschitz with $L_r^e = M_r$ and $R_r = M_r$.

In the following, we assume for problem (3.1) that each function F_i, with $i = 1:m$, admits a p higher-order surrogate as in Definition 3.2.2. Then, we propose the following General Composite Higher-Order algorithm, called GCHO, which is based on the majorization-minimization principle (i.e., minimizes at each iteration a majorizer of the objective).

Algorithm 1 Algorithm GCHO

Given $x_0 \in \operatorname{dom} f$. For $k \geq 1$ do.
Compute surrogate $s(x;x_k) := \big(s_1(x;x_k), \cdots, s_m(x;x_k)\big)$ of F near x_k.
Compute x_{k+1} satisfying the following descent:

$$g\big(s(x_{k+1};x_k)\big) + h(x_{k+1}) \leq f(x_k). \tag{3.9}$$

Although our algorithm requires that the next iterate x_{k+1} only to satisfy the descent (3.9), we usually generate x_{k+1} by solving the following subproblem:

$$\min_x g\big(s(x;x_k)\big) + h(x). \tag{3.10}$$

If F and h are convex functions, then the subproblem (3.10) can be also convex. Indeed, for Example 3.2.3, if $M_p \geq pL_p^{F_1}$ and F_2 is convex, then the surrogate function s is convex and hence the problem (3.10) is convex (see Theorem 1 [12]), while for Example (3.2.4), the surrogate is convex if $M_p \geq 0$. Hence, in the convex case we assume that x_{k+1} is the global optimum of the subproblem (3.10). However, in the nonconvex case, we cannot guarantee the convexity of the subproblem. In this case, we either assume that we can compute a stationary point of the subproblem (3.10) if g is the identity function or we can compute an inexact solution as defined in (3.21) if g is a general function. Note that our algorithmic framework is quite general and yields an array of algorithms, each of which is associated with the specific properties of F and the corresponding surrogate. For example, if F is a sum between a smooth term and a nonsmooth one we can use a surrogate as in Example 3.2.3; if F is fully nonsmooth we can use a surrogate as in Example 3.2.4. This is the first time such an analysis is performed, and most of our variants of GCHO were not explicitly considered in the literature before (especially in the nonconvex settings). Note that in both Examples 3.2.3 and 3.2.4, x_{k+1} can be computed *inexactly*, as detailed in the next sections.

3.3 Nonconvex convergence analysis

In this section we consider that each F_i, with $i = 1 : m$, and h are nonconvex functions (possible nonsmooth). Then, problem (3.1) becomes a pure nonconvex optimization problem. Now we are ready to analyze the convergence behavior of GCHO algorithm under these general settings. In the sequel, we assume that $g(-R_p^e) < 0$. Note that since the vector $R_p^e > 0$, then for all the optimization problems considered in Examples 3.1.1, 3.1.2 and 3.1.3 this assumption holds provided that M_p is large enough.

Theorem 3.3.1. *Let F, g and h satisfy Assumption 3.2.1 and additionally each F_i admits a p higher-order surrogate s_i as in Definition 3.2.2 with the constants $L_p^e(i)$ and $R_p^e(i)$, for $i = 1 : m$. Let $(x_k)_{k\geq 0}$ be the sequence generated by Algorithm GCHO, $R_p^e = \big(R_p^e(1), \cdots, R_p^e(m)\big)$ and $L_p^e = \big(L_p^e(1), \cdots, L_p^e(m)\big)$. Then, the sequence $(f(x_k))_{k\geq 0}$ is nonincreasing and satisfies the following descent relation:*

$$f(x_{k+1}) \leq f(x_k) + \frac{g(-R_p^e)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \qquad \forall k \geq 0. \tag{3.11}$$

Proof. Denote $e(x_{k+1}; x_k) = \big(e_1(x_{k+1}; x_k), \cdots, e_m(x_{k+1}; x_k)\big)$. Then, from the definition of the error function e and (3.6), we have:

$$\frac{R_p^e}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \leq e(x_{k+1}; x_k) = s(x_{k+1}; x_k) - F(x_{k+1}).$$

This implies that:

$$F(x_{k+1}) \leq s(x_{k+1}; x_k) - \frac{R_p^e}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}.$$

Since g is nondecreasing, we get:

$$\begin{aligned} g(F(x_{k+1})) &\leq g\left(s(x_{k+1}; x_k) - \frac{R_p^e}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}\right) \\ &\overset{(3.3)}{\leq} g\big((s(x_{k+1}; x_k)\big) + \frac{g(-R_p^e)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}. \end{aligned}$$

Finally, we obtain that:

$$\begin{aligned} f(x_{k+1}) &\leq g\big(s(x_{k+1}; x_k)\big) + h(x_{k+1}) + \frac{g(-R_p^e)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\ &\overset{(3.9)}{\leq} f(x_k) + \frac{g(-R_p^e)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}, \end{aligned}$$

which yields our statement. ■

Summing (3.11) from $j = 0$ to k, we get:

$$\sum_{j=0}^{k} -\frac{g(-R_p^e)}{(p+1)!}\|x_{j+1} - x_j\|^{p+1} \leq \sum_{j=0}^{k} f(x_j) - f(x_{j+1})$$
$$= f(x_0) - f(x_{k+1}) \leq f(x_0) - f^*.$$

Taking the limit as $k \to +\infty$, we obtain:

$$\sum_{k=0}^{+\infty}\|x_k - x_{k+1}\|^{p+1} < +\infty. \tag{3.12}$$

Hence $\lim_{k\to+\infty}\|x_k - x_{k+1}\| = 0$. In our convergence analysis, we also consider the following additional assumption which requires the existence of some auxiliary sequence that must be closed to the sequence generated by GCHO algorithm and some first-order relation holds:

Assumption 3.3.2. Given the sequence $(x_k)_{k\geq 0}$ generated by GCHO algorithm, there exist two constants $L_p^1, L_p^2 > 0$ and a sequence $(y_k)_{k\geq 0}$ such that:

$$\|y_{k+1} - x_k\| \leq L_p^1\|x_{k+1} - x_k\| \text{ and } \; S_f(y_{k+1}) \leq L_p^2\|y_{k+1} - x_k\|^p \quad \forall k \geq 0. \tag{3.13}$$

In the next section, we provide concrete examples for the sequence $(y_k)_{k\geq 0}$ satisfying Assumption 3.3.2, and the corresponding expressions for L_p^1 and L_p^2.

3.3.1 Approaching the set of stationary points

Before continuing with the convergence analysis of GCHO algorithm, let us analyze the relation between $\|x_{k+1} - x_k\|^p$ and $S_f(x_{k+1})$ and also give examples when Assumption 3.3.2 is satisfied. For simplicity, consider the following simple composite minimization problem:

$$\min_x f(x) := F(x) + h(x),$$

where F is p times differentiable, having the p derivative L_p^F-Lipschitz and h is proper lower semicontinuous function. In this case g is the identity function and we can take as a surrogate $s(y;x) = T_p^F(y;x) + \frac{M_p}{(p+1)!}\|x - y\|^{p+1} + h(y)$, with the positive constant M_p satisfying $M_p > L_p^F$ and $g(-R_p^e) < 0$. The following lemma gives an example when Assumption 3.3.2 holds.

Lemma 3.3.3. *Assume g is the identity function, F has the p derivative Lipschitz and x_{k+1} is a stationary point of the following subproblem:*

$$x_{k+1} \in \arg\min_x T_p^F(x; x_k) + \frac{M_p}{(p+1)!}\|x - x_k\|^{p+1} + h(x). \tag{3.14}$$

Then, Assumption 3.3.2 holds with $y_{k+1} = x_{k+1}$, $L_p^1 = 1$ and $L_p^2 = \frac{M_p + L_p^F}{p!}$.

Proof. Since x_{k+1} is a stationary point of subproblem (3.14), using (2.14), we get:

$$\frac{M_p}{p!}\|x_{k+1} - x_k\|^{p-1}B(x_k - x_{k+1}) - \nabla T_p^F(x_{k+1}; x_k) \in \partial h(x_{k+1}),$$

or equivalently

$$\frac{M_p}{p!}\|x_{k+1}-x_k\|^{p-1}B(x_k-x_{k+1}) + \Big(\nabla F(x_{k+1}) - \nabla T_p^F(x_{k+1};x_k)\Big)$$
$$\in \nabla F(x_{k+1}) + \partial h(x_{k+1}) = \partial f(x_{k+1}).$$

Taking into account that F is p-smooth, we further get:

$$S_f(x_{k+1}) \leq \frac{M_p}{p!}\|x_{k+1}-x_k\|^p + \|\nabla F(x_{k+1}) - \nabla T_p^F(x_{k+1},x_k)\|_* \tag{3.15}$$
$$\stackrel{(2.6)}{\leq} \frac{M_p+L_p^F}{p!}\|x_{k+1}-x_k\|^p. \tag{3.16}$$

Hence, Assumption 3.3.2 holds with $y_{k+1}=x_{k+1}$, $L_p^1=1$ and $L_p^2=\frac{M_p+L_p^F}{p!}$. ■

The algorithm GCHO which generates a sequence $(x_k)_{k\geq 0}$ satisfying the descent (3.9) and the stationary condition (3.14) has been also considered, e.g., in the recent papers [20, 19], with h assumed to be a convex function. Here we remove this assumption on h.

Combining (3.15) and (3.11), we further obtain:

$$S_f(x_{k+1})^{\frac{p+1}{p}} \leq \left(\frac{M_p+L_p^F}{p!}\right)^{\frac{p+1}{p}} \frac{(p+1)!}{M_p-L_p^F}\Big(f(x_k)-f(x_{k+1})\Big) = C_{M_p,L_p^F}\Big(f(x_k)-f(x_{k+1})\Big),$$

where $C_{M_p,L_p^F} = \left(\frac{M_p+L_p^F}{p!}\right)^{\frac{p+1}{p}} \frac{(p+1)!}{M_p-L_p^F}$. Summing the last inequality from $j=0:k-1$, and using that f is bounded from below by f^*, we get:

$$\sum_{j=0}^{k-1} S_f(x_j)^{\frac{p+1}{p}} \leq C_{M_p,L_p^F}\Big(f(x_0)-f(x_k)\Big) \leq C_{M_p,L_p^F}\Big(f(x_0)-f^*\Big).$$

Hence:

$$\min_{j=0:k-1} S_f(x_j) \leq \frac{\Big(C_{M_p,L_p^F}(f(x_0)-f^*)\Big)^{\frac{p}{p+1}}}{k^{\frac{p}{p+1}}}.$$

Thus, we have proved convergence for the simple composite problem under slightly more general assumptions than in [20, 19], i.e., F and h are possibly nonconvex functions. Finally, if we have $\|x_{k+1}-x_k\|^p \leq \frac{p!}{L_p^F+M_p}\epsilon$, then from (3.15) it follows that $S_f(x_{k+1}) \leq \epsilon$, i.e., x_{k+1} is nearly stationary for f. Note that in the previous Lemma 3.3.3, we assume x_{k+1} to be a stationary point of the following subproblem (see (3.14)):

$$x_{k+1} \in \arg\min_x s(x;x_k). \tag{3.17}$$

However, our stationary condition for x_{k+1} can be relaxed to the *following inexact* optimality criterion (see also [13]):

$$\|g_{x_{k+1}}\| \leq \theta\|x_{k+1}-x_k\|^p, \tag{3.18}$$

where $g_{x_{k+1}} \in \partial s(x_{k+1};x_k)$ and $\theta > 0$. For simplicity of the exposition, in our convergence analysis below for this particular case (i.e., g identity function) we assume however that x_{k+1} satisfies the exact stationary condition (3.17), although our results can be extended to the *inexact* stationary condition from above. The situation is dramatically different for the general composite problem (3.1). When g is nonsmooth, the distance $\mathrm{dist}\big(0, \partial f(x_{k+1})\big)$ will typically not even tend to zero in the limit, although we have seen that $\|x_{k+1} - x_k\|^p$ converges to zero. Indeed, consider the minimization of the following function:

$$f(x) = \max\left(x^2 - 1, 1 - x^2\right).$$

For $p = 1$, we have $L_1^F(1) = L_1^F(2) = 2$. Taking $x_0 > 1$ and $M_1 = M_2 = 4$, then the iterates of GCHO algorithm are of the form:

$$x_{k+1} = \arg\min_x Q(x, x_k) \left(:= \max\left(Q_1(x, x_k), Q_1(x, x_k) - 4xx_k + 2x_k^2 + 2\right)\right),$$

where $Q_1(x, x_k) = 2x^2 - 2xx_k + x_k^2 - 1$. Let us prove by induction that $x_k > 1$ for all $k \geq 0$. Assume that $x_k > 1$ for some $k \geq 0$. We notice that the polynomials $Q_2(x, x_k) := Q_1(x, x_k) - 4xx_k + 2x_k^2 + 2$ and $Q_1(x, x_k)$ are 2-strongly convex functions and they intersect in a unique point $\bar{x} = \frac{x_k^2+1}{2x_k}$. Also, the minimum of Q_2 is $\bar{x}_2 = \frac{3}{2}x_k$ and the minimum of Q_1 is $\bar{x}_1 := \frac{1}{2}x_k$, satisfying $\bar{x}_1 \leq \bar{x} \leq \bar{x}_2$. Let us prove that $x_{k+1} = \bar{x}$. Indeed, if $x \leq \bar{x}$, then $Q(x, x_k) = Q_2(x, x_k)$ and it is nonincreasing on $(-\infty, \bar{x}]$. Hence, $Q(x, x_k) \geq Q(\bar{x}, x_k)$ for all $x \leq \bar{x}$. Further, if $x \geq \bar{x}$, then $Q(x, x_k) = Q_1(x, x_k)$ and it is nondecreasing on $[\bar{x}, +\infty)$. In conclusion, $Q(x, x_k) \geq Q(\bar{x}, x_k)$ for all $x \leq \bar{x}$. Finally, we have that: $Q(x, x_k) \geq Q(\bar{x}, x_k)$ for all $x \in \mathbb{R}$. Since $x_k > 1$, we also get that $x_{k+1} = \frac{x_k^2+1}{2x_k} > 1$. Since $x_k > 1$, then $\partial f(x_k) = 2x_k > 2$ and $S_f(x_k) \geq 2 > 0$. Moreover, $x_{k+1} < x_k$ and bounded below by 1, thus $(x_k)_{k\geq 0}$ is convergent and its limit is 1. Indeed, assume that $x_k \to \hat{x}$ as $k \to \infty$. Then, we get $\hat{x} = \frac{\hat{x}^2+1}{2\hat{x}}$ and thus $\hat{x} = 1$ (recall that $\hat{x} \geq 1$). Consequently, $\|x_{k+1} - x_k\|$ also converges to 0. Therefore, we must look elsewhere for a connection between $S_f(\cdot)$ and $\|x_{k+1} - x_k\|^p$.

Let us now consider the following subproblem:

$$\mathcal{P}(x_k) = \arg\min_y \mathcal{M}_p(y, x) := f(y) + \frac{\mu_p}{(p+1)!}\|y - x_k\|^{p+1}, \tag{3.19}$$

where $\mu_p > g(L_p^e)$. Since f is assumed to be bounded from bellow, then for any fixed x, the function $y \mapsto \mathcal{M}_p(y, x)$ is coercive, and hence the optimal value $\mathcal{M}_p^* = \inf_y \mathcal{M}_p(y, x)$ is finite. Then, the subproblem (3.19) is equivalent to:

$$\inf_{y\in\mathcal{B}_k} f(y) + \frac{\mu_p}{(p+1)!}\|y - x_k\|^{p+1},$$

for some compact set $\mathcal{B}_k$. Since $\mathcal{M}_p$ is proper lower semicontinuous function in the first argument and $\mathcal{B}_k$ is compact set, then from Weierstrass theorem we have that the infimum $\mathcal{M}_p^*$ is attained, i.e., there exists $\bar{y}_{k+1} \in \mathcal{P}(x_k)$ such that $\mathcal{M}_p(\bar{y}_{k+1}, x_k) = \mathcal{M}_p^*$. Since the level sets of $y \mapsto \mathcal{M}_p(x, y)$ are compact, then $\mathcal{P}(x_k)$ is nonempty and compact and one can consider the point:

$$y_{k+1} = \arg\min_{y\in\mathcal{P}(x_k)} \|y - x_k\|. \tag{3.20}$$

Let us assume that F_i admits a higher-order surrogate as in Definition 3.2.2, where the error functions e_i are p smooth with Lipschitz constants $L_p^e(i)$ for all $i = 1 : m$. Denote $L_p^e = \big(L_p^e(1), \cdots, L_p^e(m)\big)$ and define the following positive constant $C_{L_p^e}^{\mu_p} = \dfrac{\mu_p}{\mu_p - g(L_p^e)}$ (recall that μ_p is chosen such that $\mu_p > g(L_p^e)$). Next lemma shows that Assumption 3.3.2 holds provided that we compute x_{k+1} as an approximate local solution of subproblem (3.10) (hence, x_{k+1} doesn't need to be global optimum) and y_{k+1} as in (3.20).

Lemma 3.3.4. *Let the assumptions of Theorem 3.3.1 hold, and additionally, there exists $\delta > 0$*

such that x_{k+1} satisfies the following inexact optimality condition:

$$g\big(s(x_{k+1};x_k)\big) + h(x_{k+1}) - \min_{x:\, \|x-x_k\|\leq D_k} \big(g\big(s(x;x_k)\big) + h(x)\big) \leq \delta\|x_{k+1}-x_k\|^{p+1}, \tag{3.21}$$

where $D_k := \left(\frac{(p+1)!}{\mu_p}(f(x_k)-f^*)\right)^{\frac{1}{p+1}}$. Then, Assumption 3.3.2 holds with y_{k+1} given in (3.20), $L_p^1 = \left(C^{\mu_p}_{L_p^e} + \frac{\delta(p+1)!}{\mu_p - g(L_p^e)}\right)^{1/(p+1)}$ and $L_p^2 = \frac{\mu_p}{p!}$.

Proof. From the definition of y_{k+1} in (3.20), we have:

$$\begin{aligned} f(y_{k+1}) + \frac{\mu_p}{(p+1)!}\|y_{k+1}-x_k\|^{p+1} &= \min_y f(y) + \frac{\mu_p}{(p+1)!}\|y-x_k\|^{p+1} \\ &\leq f(x_{k+1}) + \frac{\mu_p}{(p+1)!}\|x_{k+1}-x_k\|^{p+1}. \end{aligned} \tag{3.22}$$

Further, taking $y = x_k$ in (3.22) we also have:

$$f(y_{k+1}) + \frac{\mu_p}{(p+1)!}\|y_{k+1}-x_k\|^{p+1} \leq f(x_k),$$

which implies that:

$$\|y_{k+1}-x_k\| \leq \left(\frac{(p+1)!}{\mu_p}(f(x_k)-f^*)\right)^{\frac{1}{p+1}} = D_k. \tag{3.23}$$

Note that since the error functions e_i's have the p derivative Lipschitz with constants $L_p^e(i)$'s, then using (2.4), we get:

$$|e_i(y;x_k) - T_p^{e_i}(y;x_k)| \leq \frac{L_p^e(i)}{(p+1)!}\|y-x_k\|^{p+1} \quad \forall i = 1:m, \quad \forall y \in \operatorname{dom} e_i.$$

From (3.5), the Taylor approximations of e_i's of order p at x_k, $T_p^e(y;x_k)$, are zero. Hence:

$$|s_i(y;x_k) - F_i(y)| = |e_i(y;x_k)| \leq \frac{L_p^e(i)}{(p+1)!}\|y-x_k\|^{p+1} \quad \forall i = 1:m. \tag{3.24}$$

Further, since $F(x_{k+1}) \leq s(x_{k+1};x_k)$ (see (3.6)) and g is a nondecreasing function, we have:

$$\begin{aligned} f(x_{k+1}) &\leq g\Big(s(x_{k+1};x_k)\Big) + h(x_{k+1}) \\ &\overset{(3.21)}{\leq} \min_{y:\, \|y-x_k\|\leq D_k} g\Big(s(y;x_k)\Big) + h(y) + \delta\|x_{k+1}-x_k\|^{p+1} \\ &\overset{(3.24)}{\leq} \min_{y:\, \|y-x_k\|\leq D_k} g\left(F(y) + \frac{L_p^e}{p+1!}\|y-x_k\|^{p+1}\right) + h(y) + \delta\|x_{k+1}-x_k\|^{p+1} \\ &\overset{(3.3)}{\leq} \min_{y:\, \|y-x_k\|\leq D_k} f(y) + \frac{g(L_p^e)}{(p+1)!}\|y-x_k\|^{p+1} + \delta\|x_{k+1}-x_k\|^{p+1} \\ &\overset{(3.23)}{\leq} f(y_{k+1}) + \frac{g(L_p^e)}{(p+1)!}\|y_{k+1}-x_k\|^{p+1} + \delta\|x_{k+1}-x_k\|^{p+1}. \end{aligned}$$

Then, combining the last inequality with (3.22), we get:

$$\|y_{k+1}-x_k\|^{p+1} \leq \frac{\mu_p + \delta(p+1)!}{\mu_p - g(L_p^e)}\|x_{k+1}-x_k\|^{p+1},$$

which is the first statement of Assumption 3.3.2. Further, using (2.14) and optimality conditions for y_{k+1}, we obtain:

$$0 \in \partial f(y_{k+1}) + \frac{\mu_p}{p!}\|y_{k+1} - x_k\|^{p-1} B(y_{k+1} - x_k).$$

It follows that:

$$S_f(y_{k+1}) \leq \frac{\mu_p}{p!}\|y_{k+1} - x_k\|^p.$$

Hence, Assumption 3.3.2 holds with y_{k+1} given in (3.20), $L_p^1 = \left(C_{L_p^e}^{\mu_p} + \frac{\delta(p+1)!}{\mu_p - g(L_p^e)}\right)^{1/(p+1)}$ and $L_p^2 = \frac{\mu_p}{p!}$. ■

Finally, we provide a third (practical) example satisfying Assumption 3.3.2 when $p = 2$, $h(\cdot) = 0$ and $g(\cdot) = \max(\cdot)$ function.

Lemma 3.3.5. *Let the assumptions of Theorem 3.3.1 hold and additionally assume that $p = 2$, $g(\cdot) = \max(\cdot)$ and the surrogate function $s(\cdot;\cdot)$ is given in Example 3.2.3 with $F_2 = 0$. Then, the global solution of the subproblem (3.10) with $h = 0$, denoted x_{k+1}, can be computed efficiently and consequently Assumption 3.3.2 holds with y_{k+1} given in (3.20), $L_p^1 = \left(C_{L_p^e}^{\mu_p}\right)^{1/3}$ and $L_p^2 = \frac{\mu_p}{2}$.*

Proof. Let us first prove that for $p = 2$, $g(\cdot) = \max(\cdot)$ and $h(\cdot) = 0$, one can compute efficiently the global solution x_{k+1} of the subproblem (3.10). Indeed, in this particular case (3.10) is equivalent to the following subproblem:

$$\min_{x\in\mathbb{R}^n} \max_{i=1:m} \left\{ F_i(x_k) + \langle \nabla F_i(x_k), x - x_k\rangle + \frac{1}{2}\left\langle \nabla^2 F_i(x_k)(x - x_k), x - x_k\right\rangle \right. \\ \left. + \frac{M_i}{6}\|x - x_k\|^3 \right\}. \tag{3.25}$$

Further, this is equivalent to:

$$\min_{x\in\mathbb{R}^n} \max_{u\in\Delta_m} \sum_{i=1}^m u_i F_i(x_k) + \left\langle \sum_{i=1}^m u_i \nabla F_i(x_k), x - x_k \right\rangle \\ + \frac{1}{2}\left\langle \sum_{i=1}^m u_i \nabla^2 F_i(x_k)(x - x_k), x - x_k \right\rangle + \frac{\sum_{i=1}^m u_i M_i}{6}\|x - x_k\|^3,$$

where $u = (u_1, \cdots, u_m)$ and $\Delta_m := \{u \geq 0 : \sum_{i=1}^m u_i = 1\}$ is the standard simplex in $\mathbb{R}^m$. Further, this min − max problem can be written as follows:

$$\min_{x\in\mathbb{R}^n} \max_{u\in\Delta_M} \sum_{i=1}^m u_i F_i(x_k) + \left\langle \sum_{i=1}^m u_i \nabla F_i(x_k), x - x_k \right\rangle \\ + \frac{1}{2}\left\langle \sum_{i=1}^m u_i \nabla^2 F_i(x_k)(x - x_k), (x - x_k) \right\rangle + \max_{w\geq 0}\left(\frac{w}{4}\|x - x_k\|^2 - \frac{1}{12(\sum_{i=1}^m u_i M_i)^2} w^3 \right).$$

Denote for simplicity $H_k(u,w) = \sum_{i=1}^m u_i \nabla^2 F_i(x_k) + \frac{w}{2} I$, $g_k(u) = \sum_{i=1}^m u_i \nabla F_i(x_k)$, $l_k(u) = \sum_{i=1}^m u_i F_i(x_k)$ and $\tilde{M}(u) = \sum_{i=1}^m u_i M_i$. Then, the dual formulation of this problem takes the form:

$$\min_{x\in\mathbb{R}^n} \max_{\substack{u\in\Delta_m \\ w\in\mathbb{R}_+}} \; l_k(u) + \langle g_k(u), x-x_k\rangle + \frac{1}{2}\left\langle H_k(u,w)(x-x_k),(x-x_k)\right\rangle - \frac{w^3}{12\tilde{M}(u)^2}.$$

Consider the following notations:

$$\begin{aligned}
\theta(x,u) &= l_k(u) + \langle g_k(u), x - x_k\rangle + \frac{1}{2}\left\langle\left(\sum_{i=1}^{m} u_i \nabla^2 F_i(x_k)\right)(x-x_k), x-x_k\right\rangle + \frac{\tilde{M}(u)}{6}\|x-x_k\|^3,\\
\beta(u,w) &= l_k(u) - \frac{1}{2}\left\langle H_k(u,w)^{-1} g(u), g(u)\right\rangle - \frac{1}{12\tilde{M}(u)^2} w^3,\\
D &= \left\{(u,w)\in\Delta_m\times\mathbb{R}_+ : \;\text{ s.t. } \sum_{i=1}^{m} u_i\nabla^2 F_i(x_k) + \frac{w}{2} I \succ 0\right\}.
\end{aligned}$$

Below, we prove that if there exists an $M_i > 0$, for some $i = 1:m$, then we have the following relation:

$$\theta^* := \min_{x\in\mathbb{R}^n}\max_{u\in\Delta_m} \theta(x,u) = \max_{(u,w)\in D} \beta(u,w) = \beta^*.$$

Additionally, for any $(u,w)\in D$ the direction $x_{k+1} = x_k - H_k(u,w)^{-1} g_k(u)$ satisfies:

$$0 \le \theta(x_{k+1},u) - \beta(u,w) = \frac{\tilde{M}(u)}{12}\left(\frac{w}{\tilde{M}(u)} + 2r_k\right)\left(r_k - \frac{w}{\tilde{M}(u)}\right)^2, \tag{3.26}$$

where $r_k := \|x_{k+1} - x_k\|$. Indeed, let us first show that $\theta^* \ge \beta^*$. Using a similar reasoning as in [10], we have:

$$\begin{aligned}
\theta^* &= \min_{x\in\mathbb{R}^n} \max_{\substack{u\in\Delta_m \\ w\in\mathbb{R}_+}} \; l_k(u) + \langle g_k(u), x-x_k\rangle + \frac{1}{2}\left\langle H_k(u,w)(x-x_k), x-x_k\right\rangle - \frac{w^3}{12\tilde{M}(u)^2}\\
&\ge \max_{\substack{u\in\Delta_m \\ w\in\mathbb{R}_+}} \min_{x\in\mathbb{R}^n} \; l_k(u) + \langle g_k(u), x-x_k\rangle + \frac{1}{2}\left\langle H_k(u,w)(x-x_k), x-x_k\right\rangle - \frac{w^3}{12\tilde{M}(u)^2}\\
&\ge \max_{(u,w)\in D} \min_{x\in\mathbb{R}^n} \; l_k(u) + \langle g_k(u), x-x_k\rangle + \frac{1}{2}\left\langle H_k(u,w)(x-x_k), x-x_k\right\rangle - \frac{w^3}{12\tilde{M}(u)^2}\\
&= \max_{(u,w)\in D} l_k(u) - \frac{1}{2}\left\langle H_k(u,w)^{-1} g_k(u), g_k(u)\right\rangle - \frac{1}{12\tilde{M}(u)^2} w^3 = \beta^*.
\end{aligned}$$

Let $(u,w)\in D$. Then, we have $g_k(u) = -H_k(u,w)(x_{k+1} - x_k)$ and thus:

$$\begin{aligned}
\theta(x_{k+1},u) &= l_k(u) + \langle g_k(u), x_{k+1} - x_k\rangle\\
&\quad + \frac{1}{2}\left\langle\left(\sum_{i=1}^{m} u_i\nabla^2 F_i(x_k)\right)(x_{k+1}-x_k), x_{k+1}-x_k\right\rangle + \frac{\tilde{M}(u)}{6} r_k^3\\
&= l_k(u) - \langle H_k(u,w)(x_{k+1}-x_k), x_{k+1}-x_k\rangle\\
&\quad + \frac{1}{2}\left\langle\left(\sum_{i=1}^{m} u_i\nabla^2 F_i(x_k)\right)(x_{k+1}-x_k), x_{k+1}-x_k\right\rangle + \frac{\tilde{M}(u)}{6} r_k^3\\
&= l_k(u) - \frac{1}{2}\left\langle\left(\sum_{i=1}^{m} u_i\nabla^2 F_i(x_k) + \frac{w}{2} I\right)(x_{k+1}-x_k), x_{k+1}-x_k\right\rangle - \frac{w}{4} r_k^2 + \frac{\tilde{M}(u)}{6} r_k^3\\
&= \beta(u,w) + \frac{1}{12\tilde{M}(u)^2} w^3 - \frac{w}{4} r_k^2 + \frac{\tilde{M}(u)}{6} r_k^3
\end{aligned}$$

$$= \beta(u,w) + \frac{\tilde{M}(u)}{12}\left(\frac{w}{\tilde{M}(u)}\right)^3 - \frac{\tilde{M}(u)}{4}\left(\frac{w}{\tilde{M}(u)}\right)r_k^2 + \frac{\tilde{M}(u)}{6}r_k^3$$
$$= \beta(u,w) + \frac{\tilde{M}(u)}{12}\left(\frac{w}{\tilde{M}(u)} + 2r_k\right)\left(r_k - \frac{w}{\tilde{M}(u)}\right)^2,$$

which proves (3.26). Note that we have [10]:

$$\nabla_w \beta(u,w) = \frac{1}{4}\|x_{k+1} - x_k\|^2 - \frac{1}{4\tilde{M}(u)^2}w^2 = \frac{1}{4}\left(r_k + \frac{w}{\tilde{M}(u)}\right)\left(r_k - \frac{w}{\tilde{M}(u)}\right).$$

Therefore, if β^* is attained at some $(u^*, w^*) \in D$, then we have $\nabla\beta(u^*, w^*) = 0$. This implies $\frac{w^*}{\tilde{M}(u^*)} = r_k$ and by (3.26) we conclude that $\theta^* = \beta^*$.
Finally, if x_{k+1} is a global solution of the subproblem (3.10) (or equivalently (3.25)), then it satisfies the inexact condition (3.21) with $\delta = 0$. Hence, using the proof of Lemma 3.3.4 with $\delta = 0$ we can conclude that Assumption 3.3.2 holds with y_{k+1} given in (3.20), $L_p^1 = \left(C_{L_p^e}^{\mu_p}\right)^{1/3}$ and $L_p^2 = \frac{\mu_p}{2}$. ■

From the proof of Lemma 3.3.5 one can see that *the global minimum* of subproblem (3.10) can be computed as:

$$x_{k+1} = x_k - H_k(u,w)^{-1}g_k(u),$$

where $H_k(u,w) = \sum_{i=1}^m u_i \nabla^2 F_i(x_k) + \frac{w}{2}I$, $g_k(u) = \sum_{i=1}^m u_i \nabla F_i(x_k)$ and $l_k(u) = \sum_{i=1}^m u_i F_i(x_k)$, with (u,w) the solution of the following dual problem:

$$\max_{(u,w)\in D} l_k(u) - \frac{1}{2}\left\langle H_k(u,w)^{-1}g_k(u), g_k(u)\right\rangle - \frac{1}{12(\sum_{i=1}^m u_i M_i)^2}w^3, \tag{3.27}$$

with $D = \{(u,w) \in \Delta_m \times \mathbb{R}_+ :\ \text{s.t. } H_k(u,w) \succ 0\}$, i.e., a maximization of a concave function over a convex set D. Hence, if m is not too large, this convex dual problem can be solved efficiently by interior point methods [66]. In conclusion, GCHO algorithm can be implementable for $p = 2$ even for nonconvex problems since we can effectively compute the global minimum x_{k+1} of subproblem (3.10) using the powerful tools from convex optimization.

Define the following constant: $D_{R_p^e, L_p^{1,2}} = \frac{\left(L_p^1 (L_p^2)^p\right)^{\frac{p+1}{p}}(p+1)!}{-g(-R_p^e)}$. Then, we derive the following convergence result for GCHO algorithm in the nonconvex case.

Theorem 3.3.6. *Let the assumptions of Theorem 3.3.1 hold. Additionally, Assumption 3.3.2 holds. Then, for the sequence $(x_k)_{k\geq 0}$ generated by Algorithm GCHO we have the following sublinear convergence rate:*

$$\min_{j=0:k-1} S_f(y_j) \leq \frac{\left(D_{R_p^e, L_p^{1,2}}(f(x_0) - f^*)\right)^{\frac{p}{p+1}}}{k^{\frac{p}{p+1}}}.$$

Proof. From Assumption 3.3.2, we have:

$$S_f(y_{k+1}) \leq L_p^2 \|y_{k+1} - x_k\|^p \leq L_p^2 \left(L_p^1\right)^p \|x_{k+1} - x_k\|^p.$$

Using the descent (3.11), we get:

$$S_f(y_{k+1})^{\frac{p+1}{p}} \leq \frac{\left(L_p^2 \left(L_p^1\right)^p\right)^{\frac{p+1}{p}}(p+1)!}{-g(-R_p^e)}\left(f(x_k) - f(x_{k+1})\right).$$

Summing the last inequality from $j = 0 : k-1$ and taking the minimum, we get:

$$\min_{j=0:k-1} S_f(y_j) \leq \frac{\left(D_{R_p^e, L_p^{1,2}}(f(x_0) - f^*)\right)^{\frac{p}{p+1}}}{k^{\frac{p}{p+1}}},$$

which proves the statement of the theorem. ■

Theorem 3.3.6 requires that x_{k+1} satisfies the descent (3.9) and Assumption 3.3.2. However Assumption 3.3.2, according to Lemmas 3.3.3 and 3.3.4, holds if x_{k+1} is an (inexact) stationary point or an inexact solution of the subproblem (3.10), respectively.

Remark 1. To this end, Assumption 3.3.2 requires an auxiliary sequence y_{k+1} satisfying:

$$\begin{cases} \|y_{k+1} - x_k\| \leq L_p^1 \|x_{k+1} - x_k\| \\ S_f(y_{k+1}) \leq L_p^2 \|x_{k+1} - x_k\|^p. \end{cases} \tag{3.28}$$

If $\|x_{k+1} - x_k\|$ is small, the point x_k is near y_{k+1}, which is nearly stationary for f (recall that $\|x_{k+1} - x_k\|$ converges to 0). Hence, we do not have approximate stationarity for the original sequence x_k but for the auxiliary sequence y_k, which is close to the original sequence. Note that in practice, y_{k+1} does not need to be computed. The purpose of y_{k+1} is to certify that x_k is approximately stationary in the sense of (3.28). For $p = 1$ a similar conclusion was derived in [56]. For a better understanding of the behavior of the sequence y_{k+1}, let us come back to our example $f(x) = \max\left(x^2 - 1, 1 - x^2\right)$ and $p = 1$. Recall that we have proved $x_k > 1$ and choosing $\mu_p = 4$, then y_{k+1} is the solution of the following subproblem:

$$y_{k+1} = \arg\min_y \max\left(y^2 - 1, 1 - y^2\right) + 2(y - x_k)^2.$$

Then, it follows immediately that:

$$y_{k+1} = \begin{cases} \frac{2}{3} x_k, & \text{if } x_k > \frac{3}{2} \\ 1, & \text{if } 1 \leq x_k \leq \frac{3}{2}. \end{cases}$$

Since we have already proved that $x_k \to 1$, we conclude that $|y_{k+1} - x_k| \to 0$ and consequently $\operatorname{dist}(0, \partial f(y_{k+1})) \to 0$ for $k \to \infty$, as predicted by our theory.

3.3.2 Better rates for GCHO under KL

In this section, we show that improved rates can be derived for GCHO algorithm if the objective function satisfies the KL property. This is the first time when such convergence analysis is derived for the GCHO algorithm on the composite problem (3.1). We believe that this lack of analysis comes from the fact that, when g is nonsmooth and different from the identity function, one can't bound directly the distance $S_f(x_{k+1})$ by $\|x_{k+1} - x_k\|$. However, using the newly introduced (artificial) point y_{k+1}, we can now overcome this difficulty.

Lemma 3.3.7. *Let $(x_k)_{k\geq 0}$ generated by Algorithm GCHO be bounded and $(y_k)_{k\geq 0}$ satisfies Assumption 3.3.2. Then, $(y_k)_{k\geq 0}$ is bounded and the set of limit points of the sequence $(y_k)_{k\geq 0}$ coincides with the set of limit points of $(x_k)_{k\geq 0}$.*

Proof. Indeed, since $(x_k)_{k\geq 0}$ is bounded, then it has limit points. Let x_* be a limit point of the sequence $(x_k)_{k\geq 0}$. Then, there exists a subsequence $(x_{k_t})_{t\geq 0}$ such that $x_{k_t} \to x_*$ for $t \to \infty$. Furthermore we have:

$$\begin{aligned} \|y_{k_t} - x_{k_t}\| &\leq \|y_{k_t} - x_{k_t - 1}\| + \|x_{k_t} - x_{k_t-1}\| \\ &\overset{(3.13)}{\leq} \left(L_p^1 + 1\right) \|x_{k_t} - x_{k_t-1}\| \qquad \forall k \geq 0, \end{aligned} \tag{3.29}$$

Since $(x_k)_{k\geq 0}$ is bounded and $\|x_{k+1} - x_k\| \to 0$, then $(y_k)_{k\geq 0}$ is also bounded. This implies that $y_{k_t} \to x_*$. Hence, x_* is also a limit point of the sequence $(y_k)_{k\geq 0}$. Further, let y_* be a limit point of the bounded sequence $(y_k)_{k\geq 0}$. Then, there exists a subsequence $(y_{\bar{k}_t})_{t\geq 0}$ such that $y_{\bar{k}_t} \to y_*$ for $t \to \infty$. Taking $t \to \infty$ in an inequality similar to (3.29) and using $\lim_{t\to\infty} \|x_{\bar{k}_t} - x_{\bar{k}_t - 1}\| = 0$ and boundedness of $(x_k)_{k\geq 0}$, we get that $x_{\bar{k}_t} \to y_*$, i.e., y_* is also a limit point of $(x_k)_{k\geq 0}$. ∎

Note that usually for deriving convergence rates under KL condition, we need to assume that the sequence generated by the algorithm is bounded (see e.g., Theorem 1 in [46]). Let us denote the set of limit points of $(x_k)_{k\geq 0}$ by:

$$\Omega(x_0) = \{\bar{x} \in \mathbb{E} : \exists \text{ an increasing sequence of integers } (k_t)_{t\geq 0}, \text{ such that } x_{k_t} \to \bar{x} \text{ as } t \to \infty\},$$

and the set of stationary points of problem (3.1) by stat$f$.

Lemma 3.3.8. *Let the assumptions of Theorem 3.3.1 hold. Additionally, assume that $(x_k)_{k\geq 0}$ is bounded, $(y_k)_{k\geq 0}$ satisfies Assumption 3.3.2 and f is continuous. Then, we have: $\emptyset \neq \Omega(x_0) \subseteq$ stat$f$, $\Omega(x_0)$ is compact and connected set, and f is constant on $\Omega(x_0)$, i.e., $f(\Omega(x_0)) = f_*$.*

Proof. First let us show that $f(\Omega(x_0))$ is constant. From (3.11) we have that $(f(x_k))_{k\geq 0}$ is monotonically decreasing and since f is assumed bounded from below, it converges, let us say to $f_* > -\infty$, i.e. $f(x_k) \to f_*$ as $k \to \infty$. On the other hand let x_* be a limit point of the sequence $(x_k)_{k\geq 0}$. This means that there exist a subsequence $(x_{k_t})_{t\geq 0}$ such that $x_{k_t} \to x_*$. Since f is continuous, then $f(x_{k_t}) \to f(x_*) = f_*$. In conclusion, we have $f(\Omega(x_0)) = f_*$. The closeness property of ∂f implies that $S_f(x_*) = 0$, and thus $0 \in \partial f(x_*)$. This proves that x_* is a stationary point of f and thus $\Omega(x_0)$ is nonempty. By observing that $\Omega(x_0)$ can be viewed as an intersection of compact sets:

$$\Omega(x_0) = \cap_{q\geq 0} \overline{\cup_{k\geq q}\{x_k\}},$$

so it is also compact. This completes our proof. ∎

Note that f_* from Lemma 3.3.8 is usually different from $f^* = \inf_{x\in \mathrm{dom}\, f} f(x)$ defined in Assumption 3.2.1. In addition, let us consider the following assumption:

Assumption 3.3.9. For the sequence $(x_k)_{k\geq 0}$ generated by GCHO algorithm, there exist positive constants $\theta_{1,p}, \theta_{2,p} > 0$ such that:

$$f(x_{k+1}) \leq f(y_{k+1}) + \theta_{1,p}\|y_{k+1} - x_k\|^{p+1} + \theta_{2,p}\|x_{k+1} - x_k\|^{p+1} \quad \forall k \geq 0. \tag{3.30}$$

Remark 2. Note that Assumption 3.3.9 holds when e.g., g is the identity function or when $(y_k)_{k\geq 0}$ is given in (3.20) and x_{k+1} satisfies (3.21) (see Lemmas 3.3.4 and 3.3.5). Indeed, if g is the identity function, then taking $y_{k+1} = x_{k+1}$ one can see that Assumption 3.3.9 holds for any $\theta_{1,p}$ and $\theta_{2,p}$ nonnegative constants. If g is a general function, then Assumption 3.3.9 holds, provided that x_{k+1} satisfies the inexact optimality condition (3.21). Indeed, in this case, we have:

$$
\begin{aligned}
f(x_{k+1}) &\leq g\Big(s(x_{k+1};x_k)\Big) + h(x_{k+1}) \\
&\overset{(3.21)}{\leq} \min_{y\colon \|y-x_k\|\leq D_k} g\Big(s(y;x_k)\Big) + h(y) + \delta\|x_{k+1}-x_k\|^{p+1} \\
&\overset{(3.24),(3.3)}{\leq} \min_{y\colon \|y-x_k\|\leq D_k} g\big(F(y)\big) + h(y) + \frac{g(L_p^e)}{(p+1)!}\|y-x_k\|^{p+1} + \delta\|x_{k+1}-x_k\|^{p+1} \\
&\leq f(y_{k+1}) + \frac{g(L_p^e)}{(p+1)!}\|y_{k+1}-x_k\|^{p+1} + \delta\|x_{k+1}-x_k\|^{p+1},
\end{aligned}
$$

where the last inequality follows taking $y = y_{k+1}$. Hence, Assumption 3.3.9 holds in this case for $\theta_{1,p} = \dfrac{g(L_p^e)}{(p+1)!}$ and $\theta_{2,p} = \delta$. Finally, if $p = 2$ and $g(\cdot) = \max(\cdot)$, then x_{k+1} is the global solution of the subproblem (3.10) and hence, using similar arguments as above, we can prove that Assumption 3.3.9 also holds in this case.

From previous lemmas, all the conditions of the KL property from Definition 2.3.11 are satisfied. Then, we can derive the following convergence rates depending on the KL parameter.

Theorem 3.3.10. *Let the assumptions of Lemma 3.3.8 hold. Additionally, assume that f satisfies the KL property (2.16) on $\Omega(x_0)$ and Assumption 3.3.9 is valid. Then, the following convergence rates hold for the sequence $(x_k)_{k\geq 0}$ generated by GCHO algorithm:*

- *If $q \geq \frac{p+1}{p}$, then $f(x_k)$ converges to f_* linearly for k sufficiently large.*
- *If $q < \frac{p+1}{p}$, then $f(x_k)$ converges to f_* at sublinear rate of order $\mathcal{O}\left(\frac{1}{k^{\frac{pq}{p+1-pq}}}\right)$ for k sufficiently large.*

Proof. Since $(x_k)_{k\geq 0}$ and $(y_k)_{k\geq 0}$ have the same limit points, we get:

$$
\begin{aligned}
f(x_{k+1}) - f_* &\overset{(3.30)}{\leq} f(y_{k+1}) - f_* + \theta_{1,p}\|y_{k+1}-x_k\|^{p+1} + \theta_{2,p}\|x_{k+1}-x_k\|^{p+1} \\
&\overset{(2.16)+(3.13)}{\leq} \sigma_q S_f(y_{k+1})^q + \big(\theta_{1,p}(L_p^1)^{p+1} + \theta_{2,p}\big)\|x_{k+1}-x_k\|^{p+1} \\
&\overset{(3.13)}{\leq} \sigma_q\left(L_p^2(L_p^1)^p\right)^q\|x_{k+1}-x_k\|^{qp} + \big(\theta_{1,p}(L_p^1)^{p+1} + \theta_{2,p}\big)\|x_{k+1}-x_k\|^{p+1}.
\end{aligned}
$$

If we define $\Delta_k = f(x_k) - f_*$, then combining the last inequality with (3.11), we get the following recurrence:

$$
\Delta_{k+1} \leq C_1\left(\Delta_k - \Delta_{k+1}\right)^{\frac{qp}{p+1}} + C_2\left(\Delta_k - \Delta_{k+1}\right),
$$

where $C_1 = \sigma_q(L_p^2(L_p^1)^p)^q\left(\frac{(p+1)!}{-g(-R_p^e)}\right)^{\frac{pq}{p+1}}$ and $C_2 = \big(\theta_{1,p}(L_p^1)^{p+1} + \theta_{2,p}\big)\frac{(p+1)!}{-g(-R_p^e)}$. Using Lemma 2.4.2, with $\theta = \frac{p+1}{pq}$ we get our statements. ∎

Remark 3. Contrary to Theorem 3.3.6, under KL we prove in Theorem 3.3.10 that the original sequence $(x_k)_{k\geq 0}$ converge in function values. When the objective function f is uniformly convex of order $p+1$ and g not necessarily with full domain, [22] proves linear convergence for their algorithm in function values. Our results are different, i.e., we provide convergence rates for GCHO algorithm for possibly nonconvex objective f.

3.3.3 Convex convergence analysis

In this section, we assume that the objective function f in (3.1) is convex. Since the problem (3.1) is convex, we assume that x_{k+1} is a global minimum of the subproblem (3.10), which is convex provided that M_p is sufficiently large (see Theorem 1 in [12]). Below, we also assume that the level sets of f are bounded. Since GCHO algorithm is a descent method, this implies that there exist a constant $R_0 > 0$ such that $\|x_k - x^*\| \leq R_0$ for all $k \geq 0$, where x^* is an optimal solution of (3.1). Then, we get the following sublinear rate for GCHO algorithm.

Theorem 3.3.11. *Let F, g and h satisfy Assumption 3.2.1 and additionally each F_i admits a p higher-order surrogate s_i as in Definition 3.2.2 with the constants $L_p^e(i)$ and $R_p^e(i)$, for $i = 1:m$. Additionally, f is a convex function and has bounded level sets. Let $(x_k)_{k\geq 0}$ be the sequence generated by Algorithm GCHO, $R_p^e = \big(R_p^e(1), \cdots, R_p^e(m)\big)$ and $L_p^e = \big(L_p^e(1), \cdots, L_p^e(m)\big)$. Then, we have the following convergence rate:*

$$f(x_k) - f(x^*) \leq \frac{g(L_p^e) R_0^{p+1} (p+1)^p}{p! k^p}.$$

Proof. Since $F(x_{k+1}) \leq S(x_{k+1}; x_k)$ (see (3.6)) and g is nondecreasing, we have:

$$\begin{aligned} f(x_{k+1}) &\leq g\big(s(x_{k+1};x_k)\big) + h(x_{k+1}) \\ &\overset{(3.17)}{=} \min_x g\big(s(x;x_k)\big) + h(x) \\ &\overset{(3.24)}{\leq} \min_x g\left(F(x) + \frac{L_p^e}{(p+1)!}\|x - x_k\|^{p+1}\right) + h(x). \end{aligned}$$

Hence we get:

$$\begin{aligned} f(x_{k+1}) &\overset{(3.3)}{\leq} \min_x g\big(F(x)\big) + \frac{g(L_p^e)}{(p+1)!}\|x - x_k\|^{p+1} + h(x) \\ &= \min_x f(x) + \frac{g(L_p^e)}{(p+1)!}\|x - x_k\|^{p+1} \\ &\leq \min_{\alpha\in[0,1]} f(x_k) + \alpha\big[(f(x^*) - f(x_k)\big] + \alpha^{p+1}\frac{R_0^{p+1}}{(p+1)!} g\left(L_p^e\right), \end{aligned}$$

where the last inequality follows from the convexity of f and the boundness of the level sets of f. The minimum in $\alpha \geq 0$ is achieved at:

$$\alpha^* = \left(\frac{f(x_k) - f(x^*)p!}{g(L_p^e) R_0^{p+1}}\right)^{\frac{1}{p}}.$$

We have $0 \leq \alpha^* < 1$. Indeed, since $\big(f(x_k)\big)_{k\geq 0}$ is decreasing, we have:

$$\begin{aligned} f(x_k) \leq f(x_1) \leq g\big(s(x_1;x_0)\big) + h(x_1) &= \min_x g\big(s(x;x_0)\big) + h(x) \\ &\overset{(3.24)}{\leq} \min_x g\left(F(x) + \frac{L_p^e}{(p+1)!}\|x - x_0\|^{p+1}\right) + h(x) \\ &\leq g\left(F(x^*) + \frac{L_p^e}{(p+1)!}\|x^* - x_0\|^{p+1}\right) + h(x^*) \\ &\leq f(x^*) + \frac{g(L_p^e) R_0^{p+1}}{(p+1)!}. \end{aligned}$$

Hence:

$$
\begin{aligned}
0 \leq \alpha^* &\leq \left(\frac{\big(f(x_1) - f(x^*)\big)p!}{g(L_p^e)R_0^{p+1}} \right)^{\frac{1}{p}} \leq \left(\frac{g(L_p^e)R_0^{p+1}p!}{g(L_p^e)R_0^{p+1}(p+1)!} \right)^{\frac{1}{p}} \\
&= \left(\frac{p!}{(p+1)!} \right)^{\frac{1}{p}} = \left(\frac{1}{p+1} \right)^{\frac{1}{p}} < 1.
\end{aligned}
$$

Thus, we conclude:

$$
\begin{aligned}
f(x_{k+1}) &\leq f(x_k) - \alpha^* \left(f(x_k) - f(x^*) - \frac{g(L_p^e)R_0^{p+1}}{(p+1)!}(\alpha^*)^p \right) \\
&= f(x_k) - \frac{p\alpha^*}{p+1}\big[f(x_k) - f(x^*)\big].
\end{aligned}
$$

Denoting $\delta_k = f(x_k) - f(x^*)$, we get the following estimate:

$$
\delta_k - \delta_{k+1} \geq C\delta_k^{\frac{p+1}{p}},
$$

where $C = \frac{p}{p+1}\left(\frac{p!}{g(L_p^e)R_0^{p+1}} \right)^{\frac{1}{p}}$. Thus, for $\mu_k = C^p\delta_k$ we get the following recurrence:

$$
\mu_k - \mu_{k+1} \geq \mu_k^{\frac{p+1}{p}}.
$$

Following the same proof as in [12](Theorem 4), we get:

$$
\frac{1}{\mu_k} \geq \left(\frac{1}{\mu_1^{\frac{1}{p}}} + \frac{k-1}{p} \right)^p.
$$

Since:

$$
\frac{1}{\mu_1^{\frac{1}{p}}} = \frac{1}{C\delta_1{}^{\frac{1}{p}}} = \frac{p+1}{p}\left(\frac{g(L_p^e)R_0^{p+1}}{p!(f(x_1) - f^*)} \right)^{\frac{1}{p}} \geq \frac{1}{p}(p+1)^{\frac{p+1}{p}},
$$

then:

$$
\begin{aligned}
\delta_k = C^{-p}\mu_k &= \left(\frac{p+1}{p} \right)^p \frac{g(L_p^e)R_0^{p+1}}{p!}\mu_k \\
&\leq \left(\frac{p+1}{p} \right)^p \frac{g(L_p^e)R_0^{p+1}}{p!} \left(\frac{1}{p}(p+1)^{\frac{p+1}{p}} + \frac{k-1}{p} \right)^{-p} \\
&= \frac{g(L_p^e)R_0^{p+1}}{p!} \left((p+1)^{\frac{1}{p}} + \frac{k-1}{p+1} \right)^{-p} \leq \frac{(p+1)^p g(L_p^e)R_0^{p+1}}{p!k^p}.
\end{aligned}
$$

This proves the statement of the theorem. ∎

Note that in the convex case the convergence results from [56, 21, 22] assume Lipschitz continuity of the p derivative of the object function F, which may be too restrictive. However, Theorem 3.3.11 assumes Lipschitz continuity of the p derivative of the error function $e(\cdot)$ (note that we may have the error function $e(\cdot)$ p times differentiable and with the p derivative Lipschitz, while the objective function F may not be even differentiable, see Examples 3.2.3 and 3.2.4). Hence, our proof is different and more general than [56, 21, 22]. Moreover, our convergence rate from the previous theorem covers the usual convergence rates $\mathcal{O}(\frac{1}{k^p})$ of higher-order Taylor-based methods in the convex unconstrained case [12], simple composite case [12, 19] and composite case for $p \geq 1$ [22, 21]. Therefore, Theorem 3.3.11 provides a unified convergence analysis for general composite higher-order algorithms, that covers in particular, minimax strategies for nonlinear games, min-max problems and simple composite problems, under possibly more general assumptions.

3.3.4 Adaptive GCHO algorithm

In this section, we propose an adaptive variant of GCHO algorithm. Since the surrogate functions in all the examples given in this chapter depend on a given constant M (see Examples 3.2.3 and 3.2.4, where $M = M_p$), below we consider the following notation $s(\cdot;\cdot) := s_M(\cdot;\cdot)$. Note that the convergence results from Theorems 3.3.1, 3.3.6 and 3.3.10 are derived provided that Assumption 3.3.2 and 3.3.9 and the following properties of the sequence $(x_k)_{k\geq 0}$ generated by GCHO algorithm hold:

$$g\big(s_M(x_{k+1};x_k)\big) + h(x_{k+1}) \leq f(x_k), \tag{3.31}$$

$$g(s_M(x_{k+1};x_k)) - g(F(x_{k+1})) \geq \frac{C_p^e}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}, \tag{3.32}$$

where $C_p^e := -g(-R_p^e)$ is a given constant depending on the choice of the surrogate $s_M(x_{k+1};x_k)$, which may be difficult to find in practice. Hence, in the following we propose an adaptive general composite higher-order algorithm, called (A-GCHO):

Algorithm 2 Algorithm A-GCHO

Given x_0 and $M_0, R_0 > 0$ and $i, k = 0$.
while some criterion is not satisfied **do**
 1. Compute a p higher-order surrogate $s_{2^i M_k}(\cdot;x_k)$ of F near x_k.
 2. Compute x_{k+1} satisfying the descent (3.31) with $M = 2^i M_k$.
 if (3.32) holds with $C_p^e = -g(-R_p)$ and $M = 2^i M_k$, then go to step 3.
  else set $i = i + 1$ and go to step 1.
 end if
 3. set $k = k + 1$, $M_{k+1} = 2^{i-1} M_k$ and $i = 0$.
end while

For a better understanding of this process, let us consider Example 3.2.3, where $F = F_1 + F_2$, having the p derivative of F_1 $L_p^{F_1}$-Lipschitz and F_2 proper closed convex function. Then, in this case, the surrogate is $s_M(y;x) = T_p^{F^1}(y;x) + \frac{M}{(p+1)!}\|y - x\|^{p+1} + F_2(y)$. Let $R_p, M_0 > 0$ be fixed. Then, step 1 in A-GCHO algorithm can be seen as a line search procedure (see for example [67]): that is, at each step $k \geq 0$ we choose $M_k \geq M_0$, then build $s_{M_k}(y;x_k) = T_p^{F^1}(y;x_k) + \frac{M_k}{(p+1)!}\|y - x_k\|^{p+1} + F_2(y)$ and compute x_{k+1} satisfying (3.31). If (3.32) doesn't hold, then we increase $M_k \leftarrow 2 \cdot M_k$, recompute $s_{M_k}(y;x_k)$ using the new M_k and go to step 2. We repeat this process until condition (3.32) is satisfied. Note that this line search procedure finishes in a finite number of steps. Indeed, if $M_k \geq R_p + L_p^{F^1}$, then from inequality (3.8), we get $s_{M_k}(y;x_k) - F(y) \geq \frac{R_p}{(p+1)!}\|y - x_k\|^{p+1}$ for all y and thus for $y = x_{k+1}$ and g increasing function (3.32) holds. Note also that in this case, the error function e satisfies Definition 3.2.2 (i) with $L_p^e = 2(R_p + L_p^{F^1})$. Hence, using the same convergence analysis as in the previous sections, we can derive similar convergence rates as in Theorems 3.3.1, 3.3.6 and 3.3.10 for A-GCHO algorithm under Assumption 3.3.2 and 3.3.9, since the sequence $(x_k)_{k\geq 0}$ generated by A-GCHO automatically satisfies (3.31) and (3.32). For the convex case, as in Section 3.4, in A-GCHO algorithm, we require that x_{k+1} is the global solution of the corresponding subproblem, and consequently, similar convergence results as in Theorem 3.3.11 can be derived for this adaptive general composite higher-order algorithm.

3.4 Numerical simulations

In this section, we present some preliminary numerical results for GCHO algorithm. For simulations, we consider the tests set from [68]. In [68], one can find systems of nonlinear equations, where one searches for x^* such that $F_i(x^*) = 0$ for all $i = 1, \cdots, m$. For solving these problems, we implement our GCHO algorithm for $p = 1, 2$. We consider two formulations: min-max and least-squares problems, respectively. The min-max formulation has the form:

$$\min_{x\in\mathbb{R}^n} f(x) := \max(F_1^2(x), \cdots, F_m^2(x)). \tag{3.33}$$

Similarly, the least-squares problem formulation can be also written as a simple composite minimization problem:

$$\min_{x\in\mathbb{R}^n} f(x) := \sum_{i=0}^{m} F_i^2(x). \tag{3.34}$$

Note that both formulations fit into our general problem (3.1). We consider the following 2 implementations: First, for problem (3.33), we compare GCHO algorithm for $p = 1, 2$ with IPOPT (the results are given in Table 3.1). Secondly, for problem (3.34), we compare GCHO algorithm for $p = 1, 2$ with IPOPT and the method proposed in [69] (the results are given in Table 3.2). At each iteration of GCHO algorithm, we replace each function F_i by its Taylor approximation of order p, with $p = 1, 2$, and a quadratic/cubic regularization and solve the corresponding subproblem (3.10) using IPOPT [70]. In the numerical simulations, we have noticed that for $p = 2$ IPOPT was able to detect a global minimizer of the subproblem at each iteration, i.e., the solution of IPOPT coincided with the solution obtained by solving the dual problem, as described in the proof of Lemma 3.3.5 given in the appendix. Since it is difficult to compute the corresponding Lipschitz constants for the gradient/hessian, we use the line search procedure described in Section 3.3.4. Note that since in practice it is difficult to compute the sequence $(y_k)_{k\geq 1}$, we cannot consider $\mathrm{dist}(0, \partial f(y_k)) \leq \epsilon$ as a stooping criterion for the proposed algorithm. Thus, the stopping criterion considered in our simulations is the same as in [71]:

$$\frac{f(x_k) - f_{\text{best}}}{\max(1, f_{\text{best}})} \leq 10^{-4},$$

where $f_{\text{best}} = f^* \approx 0$, but positive, and the starting point x_0 are taken from [68]. In Tables 3.1 and 3.2, we summarize our numerical results in terms of cpu time and number of iterations for GCHO algorithm $p = 1, 2$, IPOPT and [69]. Note that the test functions we consider in the two tables are nonconvex and most of them satisfy the KL condition (as semi-algrabraic functions). From the tables, we observe that GCHO algorithm ($p = 1$ or $p = 2$) applied to the min-max formulation performs better than the GCHO algorithm ($p = 1$ or $p = 2$) applied to the the least-squares problem, both in cpu time and number of iterations. This is due to the fact that the regularization constants for the min-max problem (3.33), $M_p^{\max} = \big(M_p^{\max}(1), \cdots, M_p^{\max}(m)\big)$, are much smaller than the one for the least-squares formulation (3.34), M_p^{ls}, i.e., $M_p^{\text{ls}} \approx \sum_{i=1}^{m} M_p^{\max}(i)$. Moreover, from the tables we observe that increasing the order of the Taylor approximation is beneficial for the GCHO algorithm: e.g., in the min-max formulation, GCHO with $p = 2$ is at least twice faster than GCHO with $p = 1$. We also observe from Table 3.2 that GCHO algorithm applied to min-max formulation for $p = 2$ has a better behavior (in both cpu time and number of iterations) than the method proposed in [69] for the least-squares formulation. Finally, GCHO algorithm ($p = 1, 2$) for both formulations is able to identify the global optimal points/values given in [68], while IPOPT directly applied to the formulations (3.34) or (3.33) may fail to identify the global optimal points/values (see Tables 2 and 3).

min-max formulation	GCHO(p=1)		GCHO(p=2)		IPOPT for (3.33)	
test functions	iter	cpu	iter	cpu	iter	cpu
(2) Fre	32	0.78	5	0.18	85	0.01
(7) Hel	33	0.74	11	0.4	29	0.02
(8) Bar	19	0.69	8	0.42	27	0.06*
(9) Gau	9	0.33	2	0.15	16	0.04*
(12) Box	23	0.9	9	0.5	36	0.08*
(15) Kow ($m = 11, n = 4$)	48	0.7	7	0.35	5000	7.8*
(17) Osb-1($m = 33, n = 5$)	57	3.8	9	1.7	40	0.9
(18) Big ($m = 13, n = 6$)	149	7.73	14	0.6	593	1.5*
(19) Osb-2 (m=65, n=11)	67	18.75	20	12.1	55	3.7*
(20) Wat (m=31,n=9)	23	2.56	7	2.63	5000	50.5*
(21) E-Ros ($n = m = 6$)	21	0.63	3	0.21	379	0.72
(21) E-Ros ($n = m = 20$)	26	1.7	3	0.53	124	3.8
(21) E-Ros ($n = m$=100)	25	102.5	5	40.1	119	133.9
(24) Pen II ($n = 10$)	61	6.4	3	0.32	64	0.9*
(26) Tri ($n = 10$)	20	0.53	3	0.22	45	0.2*
(30) Bro ($n = 10$)	44	0.88	3	0.25	118	0.3*

Table 3.1: Behaviour of GCHO for $p = 1, 2$ and IPOPT for the min-max formulation (3.33). Here "*" means that IPOPT didn't find x^*/f^* reported in [68].

L.S formulation	GCHO(p=1)		GCHO(p=2)		[69]		IPOPT for (3.34)	
test functions	iter	cpu	iter	cpu	iter	cpu	iter	cpu
(2) Fre	562	7.2	23	0.48	7	0.19	85	0.06
(7) Hel	59	1.2	25	0.95	15	0.55	12	0.02
(8) Bar	88	1.3	13	0.5	12	0.48	26	0.04
(9) Gau	71	1.25	13	0.65	5	0.17	8	0.03*
(12) Box	719	12.1	51	2.05	13	0.68	34	0.05
(15) Kow	534	13.1	14	0.67	10	0.49	825	1.98*
(17) Osb-1	815	45.8	101	9.6	18	3.6	103	1.9
(18) Big	968	18.5	44	2.19	17	0.79	44	0.15*
(19) Osb-2	365	45.9	82	35.6	29	15.3	329	11.5*
(20) Wat	161	50.6	21	7.6	10	3.66	794	8.16*
(21) E-Ros	2563	38.7	12	0.93	4	0.28	83	0.33
(21) E-Ros	3040	82.3	28	9.4	5	1.53	233	1.8
(21) E-Ros	530	253	33	288.2	7	71.5	223	162.4
(24) Pen II	147	10.2	7	0.8	3	0.35	22	0.08*
(26) Tri	28	0.55	5	0.3	3	0.22	26	0.05*
(30) Bro	56	0.9	12	0.59	4	0.35	36	0.07*

Table 3.2: Behaviour of GCHO algorithm for $p = 1, 2$, algorithm [69] and IPOPT for the least-squares problem (3.34). Here "*" means that IPOPT didn't find x^*/f^* reported in [68].

3.5 Conclusions

This chapter considers higher-order algorithms for solving composite problems, where the first term involves a composition between a convex, increasing, subhomogeneous merit function and a (non)smooth map. We derive global convergence guarantees for the proposed algorithm in terms of first-order stationarity and characterize a local convergence rate in function value under the Kurdyka-Łojasiewicz property. We present an efficient implementation of the proposed methods and provide a numerical comparison with existing methods from the literature.

4 Moving higher-order Taylor approximations algorithm for functional constraints minimization

Following our previous chapter, now we consider solving simple composite oprimization problems with functional constraints and develop a moving higher-order Taylor approximations algorithm for solving these problems, called MTA. We derive global convergence guarantees in both nonconvex and convex cases. We also show that the proposed algorithm, MTA, is implementable and efficient in numerical simulations.

The chapter is structured as follows: Section 4.1 provides a comprehensive literature review of first and higher-order methods for optimization problems with functional constraints. In Section 4.2, we introduce our composite higher-order framework and the associated algorithm. In Section 4.3 we derive global and local convergence results for this approach in nonconvex scenarios. Further, in Section 4.4 we derive global convergence results for this approach in (uniformly)convex case. In Section 4.5, we present an efficient implementation of the method. The chapter concludes with a summary of the numerical simulations and their results. The content presented in this chapter is derived from the paper [26].

4.1 State of the art

In this chapter, we delve into a specific class of composite optimization problems. Specifically, we consider the following simple composite minimization problem that incorporates nonlinear functional constraints:

$$\begin{aligned} \min_{x\in\mathbb{E}} F(x) &:= F_0(x) + h(x) \\ \text{s.t.}:\ & F_i(x) \leq 0 \quad \forall i = 1:m, \end{aligned} \tag{4.1}$$

where $F_i : \mathbb{E} \mapsto \mathbb{R}$, for $i = 0 : m$, are continuous differentiable functions (possibly nonconvex) and $h : \mathbb{E} \mapsto \bar{\mathbb{R}}$ is proper and convex function. We have m nonlinear inequality constraints. Problem (4.1) is now called a nonlinear programming problem. This problem is equivalent to:

$$\min_x g(F(x)) + h(x),$$

with g being equal to $g(y_0, \cdots, y_m) := y_0 + 1_{\mathbb{R}^m_-}(y_1, \cdots, y_m)$. Nonlinear programming problems have a long and rich history (see, for example, the monograph [72]), because they model many practical applications. Some of the most common are power systems, engineering design, control, signal and image processing, machine learning, and statistics, see e.g., [73, 74, 75, 76, 77, 78].

4.1.1 First-order methods

By now, the benefits of modeling problems as convex optimization problems should be quite clear for the reader: convex optimization problems are generally solvable at a global level, and the solutions can typically be obtained quickly. Sequential Convex Programming (SCP) is a local optimization method for nonconvex problems, employing convex optimization as a key component [49, 79]. The fundamental concept is straightforward, the convex elements of the problem are solved precisely and efficiently, while the nonconvex elements are approximated with convex functions that are at least locally accurate. For solving problem (4.1), and for a given x_k, SCP generate the next iteration x_{k+1} by solving the following *convex* subproblem:

$$\begin{aligned} x_{k+1} = \arg\min_{x} \; & s_0(x;x_k) + h(x) \\ \text{s.t.:} \; & s_i(x;x_k) \leq 0, \; i = 1:m, \\ & x \in \Delta_k, \end{aligned}$$

where Δ_k is a trust region. The models $s_i(x;x_k)$'s may be either an affine (first-order) Taylor approximation:

$$s_i(x;x_k) := F_i(x_k) + \langle \nabla F_i(x_k), x - x_k \rangle,$$

or the convex part of the second order Taylor expansion:

$$s_i(x;x_k) := F_i(x_k) + \langle \nabla F_i(x_k), x - x_k \rangle + \frac{1}{2}(x - x_k)^T P_i^k (x - x_k),$$

where P_i^k's are the positive semidefinite part of the Hessian, i.e., for $\nabla^2 F_i(x_k) = U_i^k \Sigma_i^k U_i^{T,k}$, then $P_i^k = U_i^k \left[\Sigma_i^k\right]_+ U_i^{T,k}$, which simply zeroes out all negative eigenvalues of $\nabla^2 F_i(x_k)$. SCP is a heuristic approach that may not always yield the optimal solution. Furthermore, its effectiveness is influenced by the choice of the initial starting point x_0.

4.1.2 Interior point methods

The interior point method (IPM) is an optimization technique designed to solve constrained optimization problems, particularly those that involve inequalities, as in (4.1). Unlike the Simplex algorithm, which navigates along the boundary of the feasible region, the interior point method finds optimal solutions by starting from within the feasible region and progressively moving towards the boundary, where the optimal solution is often located.

Inequality constraints impose limits on the solution space, defining where solutions are permitted. The interior point method incorporates these constraints by introducing a *barrier* term in the objective function. This barrier penalizes the approach toward the boundary, preventing the solution from crossing into infeasible territory [38, 80]. By strategically increasing the barrier parameter as the algorithm progresses, the method guides the optimization process towards the boundary while ensuring compliance with the constraints. More precisely, IPM transform the constrained problem into an unconstrained problem using a barrier function that penalizes the violation of constraints:

$$\phi(x, \mu) = F_0(x) + h(x) - \mu \sum_{i=1}^{m} \log(-F_i(x)),$$

where $\mu > 0$ is the barrier parameter. Then, it uses an optimization method (like Newton's method or gradient-based approaches) to solve the unconstrained problem derived from the barrier function.

Interior point methods are widely used in large-scale optimization, particularly for linear programming, nonlinear programming, and convex optimization (for more details, refer to the books by Nesterov [38] and Boyd and Vandenberghe [80]). Their ability to handle complex constraint structures and the fact that they enjoy fast convergence make them a popular choice for solving a range of optimization problems. While IPMs are efficient for linear and convex optimization, they can become significantly more complex for nonlinear problems, particularly when dealing with non-convex constraints. The convergence behavior might be less predictable, and ensuring global optimality becomes more challenging.

4.1.3 Primal-dual algorithms

In the context of optimization, the primal problem is the original optimization problem that one seeks to solve, while the dual problem is derived from the primal by associating Lagrange multipliers (also known as dual variables) with the constraints [49]. Primal-dual algorithms work with both problems at once, using the insights from one to guide the progress in the other.

This primal-dual approach is particularly useful when constraints are involved, as it allows the algorithm to balance the primal objective with the dual representation, helping to ensure that the constraints are satisfied. The Lagrangian function associated to the nonlinear programming problem (4.1) takes the following form:

$$\mathcal{L}(x, \lambda) := F_0(x) + \sum_{i=1}^{m} \lambda_i F_i(x).$$

Below, we present two of the most fundamental primal-dual algorithms, that are Sequential Quadratic Programming (SQP) and Lagrangian methods.

Sequential Quadratic Programming methods

Sequential Quadratic Programming (SQP) is a type of primal-dual algorithms used to solve nonlinear optimization problems with both equality and inequality constraints. SQP operates by approximating the problem at each iteration with a quadratic programming subproblem, derived from a second-order Taylor expansion of the primal objective function and linearizations of the constraints. More precisely, SQP generate and solve the following Quadratic Programming (QP) subproblem for a given current iteration x_k and a dual multipliers λ_k [49]:

$$\begin{aligned} x_{k+1} = \arg\min_x \; & T_1^{F_0}(x; x_k) + \frac{1}{2}(x - x_k)^T \nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)(x - x_k) \\ \text{s.t.:} \; & T_1^{F_i}(x; x_k) \leq 0, \; i = 1 : m, \end{aligned}$$

where we recall $T_1^{F_i}(x, x_k) = F_i(x_k) + \langle \nabla F_i(x_k), x - x_k \rangle$. In SQP, the dual problem is integrated into the algorithm through Lagrange multipliers, which represent the influence of the constraints on the primal objective. By solving these quadratic programming subproblems, SQP iteratively refines both the primal and dual solutions, leading to a convergence towards optimality while satisfying the problem's constraints.

The performance of SQP can be influenced by the choice of initial estimates for the solution and Lagrange multipliers. Poor initial estimates might lead to slower convergence or even failure of converge, additionally, it may struggle with non-convex optimization problems. Non-convexity can lead to convergence issues, local minima, and difficulties in ensuring global optimality.

Augmented Lagrangian methods

Augmented Lagrangian methods are a powerful class of optimization techniques used to solve constrained optimization problems, especially when other methods might struggle with the presence of constraints. These methods combine Lagrange multipliers with penalty terms to create an augmented Lagrangian function that incorporates both the original objective and constraints. The motivation behind these methods is to address the limitations of pure penalty methods while maintaining the ability to handle complex constraint structures. By integrating penalty terms with the traditional Lagrange multipliers, these methods can effectively handle constraint satisfaction while avoiding the need for excessively large penalty parameters [49, 81, 82]. A typical augmented Lagrangian algorithm for solving this problem involves the following steps:

- Define the augmented Lagrangian with a penalty parameter $\rho > 0$ [83]:

$$\mathcal{L}(x, \lambda, \rho) = F_0(x) + \frac{\rho}{2}\sum_{i=1}^{m} \max\left(F_i(x) + \frac{\lambda^i}{\rho}, 0\right)^2,$$

  where the Lagrange multipliers λ^i represent the dual variables corresponding to the inequality constraints.

- For a given set of Lagrange multipliers λ_k and penalty parameter ρ_k, find the minimizer x_{k+1} of $\mathcal{L}(x, \lambda_k, \rho_k)$.

- Update the Lagrange multipliers based on constraint violations:

$$\lambda^i_{k+1} = \max\left(0, \lambda^i_k + \rho_k F_i(x_{k+1})\right),\ \forall i = 1 : m.$$

- Increase the penalty parameter to ensure stronger enforcement of constraints. A common strategy multiplies the penalty parameter by a constant factor: $\rho_{k+1} = \alpha\rho_k$, where $\alpha > 1$.

Despite its strengths, the augmented Lagrangian methods for inequality constraints has some limitations:

- The choice of the penalty parameter ρ_k and its update rate affect convergence and performance. Tuning these parameters requires expertise and might involve trial and error.

- Each iteration involves solving an optimization problem, which can be computationally expensive, especially for large-scale or nonconvex problems.

- Convergence can be slow or unstable if the penalty parameter is not properly adjusted or if the Lagrange multipliers are initialized poorly.

Various methods have been proposed in the literature to address these challenges. For smooth nonconvex problems, the approach presented in [84] achieves a complexity of order $\mathcal{O}(\epsilon^{-3})$. For smooth convex problems, a sublinear convergence rate of order $\mathcal{O}(\epsilon^{-1})$ has been established in [85, 86]. Additionally, some techniques transform inequality constraints into equalities by introducing slack variables, as discussed in [87]. In situations where only equality constraints are present, augmented Lagrangian-based methods can be particularly effective, as demonstrated in [88] and related references. Typically, they have a computational complexity of order$\mathcal{O}(\epsilon^{-2})$, indicating that the computational effort required is inversely proportional to the square of the desired solution precision.

4.1.4 Moving balls approximation type methods

For the particular class of constrained optimization problems with smooth data, [24] introduces a moving ball approximation method (MBA) that approximates the objective function with a quadratic and the feasible set by a sequence of appropriately defined balls. Namely, the algorithm presented in [24] is of the form:

$$\begin{aligned} x_{k+1} = &\arg\min_{x} \; F_0(x_k) + \langle F_0(x_k), x - x_k\rangle + \frac{M}{2}\|x - x_k\|^2 \\ &\text{s.t.: } F_i(x_k) + \langle \nabla F_i(x_k), x - x_k\rangle + \frac{M_i}{2}\|x - x_k\|^2 \leq 0, \; i = 1 : m \end{aligned}$$

where M and M_i's are positive constants and x_k is the current iteration. The authors provide asymptotic convergence guarantees for the sequence generated by MBA when the data is (non)convex, and linear convergence if the objective function is strongly convex. Later, several papers considered variants of the MBA algorithm; see [21, 89, 25]. For example, in [25] the authors present a line search MBA algorithm for difference-of-convex minimization problems and derive asymptotic convergence in the nonconvex settings and local convergence in the iterates when a special potential function related to the objective satisfies the Kurdyka-Lojasiewicz (KL) property. In [21], the authors consider convex composite minimization problems that cover, in particular, problems of the form (4.1), and use a similar MBA type scheme for solving such problems, deriving a sublinear convergence rate for it. Stochastic variants of moving balls approximation framework has been recently proposed in [90] and sublinear convergence rates have been derived in the strongly convex and convex cases. Note that all these previous methods are *first-order methods*, and despite their empirical success in solving difficult optimization problems, their convergence speed is known to be slow.

4.1.5 Higher-order methods for functional constraints minimization

Several papers have already proposed higher-order methods for solving composite optimization problems of the form (4.1) with complexity guarantees, see, e.g., [91, 92, 22, 93]. For example, in a recent paper [22], the authors consider fully composite problems (which cover, as a particular case, (4.1)) and assume the data are p times continuously differentiable with the pth derivative Lipschitz. At each iterations, it builds a p higher-order model and solves the following subproblem:

$$\begin{aligned} x_{k+1} = &\arg\min_{x} \; T_p^{F_0}(x; x_k) + \frac{M_0}{(p+1)!}\|x - x_k\|^{p+1} \\ &\text{s.t.: } T_p^{F_i}(x; x_k) + \frac{M_i}{(p+1)!}\|x - x_k\|^{p+1} \leq 0, \; i = 1 : m. \end{aligned}$$

When F_i's are uniformly convex, they derive linear convergence in function values, but there is no convergence analysis for general convex case. For nonconvex problems, [91] derives worst-case complexity bounds for computing approximate first-order critical points using higher-order derivatives for problem (4.1) with nonlinear equality constraints. Paper [92] also considers problem (4.1) with nonlinear equality constraints and employs an approach wherein the objective is approximated with a model of arbitrarily high-order while the constraints remain unchanged. Each iteration requires the computation of an approximate KKT point for the subproblem, devoiding of any constraint qualification condition. The authors show that their scheme converges to an approximate KKT point within $\mathcal{O}\left(\epsilon^{-\frac{p+\beta}{p+1-\beta}}\right)$ iterations, where $\beta \in [0, 1]$. We refer to the recent book [93] for a more detailed exposition on higher-order methods.

4.2 Moving higher-order Taylor approximations method

In this section, we introduce a new higher-order algorithm, which we call *Moving higher-order Taylor Approximations* (MTA) algorithm, for solving the constrained optimization problem (4.1) (possibly nonconvex). We denote the feasible set of (4.1) by $\mathcal{F} = \{x \in \mathbb{E} : F_i(x) \leq 0 \ \ \forall i = 1 : m\}$.We consider the following assumptions for the objective and the constraints:

Assumption 4.2.1. We have the following assumptions for problem (4.1):

1. The (possibly nonconvex) function F_0 is p-times continuously differentiable (with $p \geq 1$) and its pth derivative satisfy the Lipschitz condition:

$$\|D^p F_0(x) - D^p F_0(y)\| \leq L_p \|x - y\| \quad \forall x, y \in \mathbb{E}.$$

2. The (possibly nonconvex) constraints F_i are q-times continuously differentiable (with $q \geq 1$) and their qth derivatives satisfy the Lipschitz condition:

$$\|DF_i^q(x) - DF_i^q(y)\| \leq L_q^i \|x - y\| \quad \forall x, y \in \mathbb{E}, \ i = 1 : m.$$

3. h is a simple convex and locally Lipschitz continuous function.

Next, we assume that our problem is feasible and has bounded level sets:

Assumption 4.2.2. Problem (4.1) is feasible, i.e., $\mathcal{F} \neq \emptyset$ and the set:

$$\mathcal{A}(x_0) := \{x \in \mathbb{E} : x \in \mathcal{F} \text{ and } F(x) \leq F(x_0)\},$$

is bounded for any fixed $x_0 \in \mathcal{F}$.

Finally, we assume that the MFCQ holds for the problem (4.1):

Assumption 4.2.3. The MFCQ holds for the optimization problem (4.1):

$$\forall x \in \mathcal{F} \ \ \exists\, d \in \mathbb{E} \ \text{ s.t. } \ \langle \nabla F_i(x), d \rangle < 0 \ \ \forall i \in I(x),$$

where $I(x) := \{i \in [m], \ F_i(x) = 0\}$.

Note that Assumptions 4.2.2 and 4.2.3 are standard in the context of nonlinear programming. In particular, the MFCQ guarantees the existence of bounded Lagrange multipliers satisfying the KKT optimality conditions at any x. Note that in general, for an optimization algorithm, if one wants to prove only local convergence rates around a local minimum x^*, then is is sufficient to require MFCQ to hold only at x^* (see Section 2.4). However, if we want to prove global convergence for an algorithm, one needs to require MFCQ to hold on a set where the iterates lie [94, 95, 96]. From Assumption 4.2.1, we have for all $x, y \in \mathbb{E}$ [12]:

$$|F_0(y) - T_p^{F_0}(y; x)| \leq \frac{L_p}{(p+1)!} \|y - x\|^{p+1}, \tag{4.2}$$

$$|F_i(y) - T_q^{F_i}(y; x)| \leq \frac{L_q^i}{(q+1)!} \|y - x\|^{q+1}, \ i = 1 : m, \tag{4.3}$$

At each iteration our algorithm constructs Taylor approximations for the objective function and the functional constraints using the inequalities given in (4.2) and (4.3). To this end, for simplicity, we consider the following notations:

$$s_{M_p}^{M}(y;x) \overset{\text{def}}{=} T_p^{F_0}(y;x) + \frac{M_p}{(p+1)!}\|y-x\|^{p+1} + \frac{M}{(q+1)!}\|y-x\|^{q+1},$$

$$s_{M_q^i}(y;x) \overset{\text{def}}{=} T_q^{F_i}(y;x) + \frac{M_q^i}{(q+1)!}\|y-x\|^{q+1},$$

where M_p, M and M_q^i, for $i = 1 : m$, are positive constants. The MTA algorithm is defined below. Note that if F_i's, for $i = 0 : m$, are convex functions, then the subproblem (4.4) is also

Algorithm 3 MTA: Moving Taylor approximation

Given $x_0 \in \mathcal{F}$ and $M_p, M, M_q^i > 0$, for $i = 1 : m$, and $k = 0$.
while stopping criteria **do**
compute x_{k+1} a stationary point of the subproblem:

$$\begin{aligned} &\min_{x\in\mathbb{E}} s_{M_p}^{M}(x;x_k) + h(x) \\ &\text{s.t.}: \; s_{M_q^i}(x;x_k) \le 0, \;\; i = 1:m, \end{aligned} \tag{4.4}$$

satisfying the following descent:

$$s_{M_p}^{M}(x_{k+1};x_k) + h(x_{k+1}) \le s_{M_p}^{M}(x_k;x_k) + h(x_k) \; (:= F(x_k)). \tag{4.5}$$

update $k = k+1$.
end while

convex. Indeed, if $M_p \ge pL_p$ and $M_q^i \ge qL_q^i$ for $i = 1 : m$, then the Taylor approximations $s_{M_p}^{M}(\cdot;x_k)$ and $s_{M_q^i}(\cdot;x_k)$ for $i = 1 : m$ are (uniformly) convex functions (see Theorem 2 in [12]). Hence, in the convex case, we assume that x_{k+1} is the global optimum of the subproblem (4.4). However, in the nonconvex case, we cannot always guarantee the convexity of the subproblem (4.4). In this case, we just assume that x_{k+1} is a stationary (KKT) point of the subproblem (4.4) satisfying the descent (4.5). In Section 4.5 we show that one can still use the powerful tools from convex optimization for solving subproblem (4.4) even in the nonconvex case. Note that our novelty comes from using two regularization terms in the objective function of the subproblem (4.4), i.e., $\frac{M_p}{(p+1)!}\|x-x_k\|^{p+1}$ is to insure the convexity of the subproblem in the convex case (provided that $M_p \ge pL_p$), while $\frac{M}{(q+1)!}\|x-x_k\|^{q+1}$ is to ensure a descent for an appropriate Lyapunov function (see Lemma 4.3.7) and a better convergence rate (see Remark 4). We denote the feasible set of subproblem (4.4) by $\mathcal{F}(x_k) := \{y \in \mathbb{E} : \; s_{M_q^i}(y;x_k) \le 0 \;\; \forall i = 1 : m\}$.

4.3 Nonconvex convergence analysis

In this section, we assume that the problem (4.1) is nonconvex, i.e., F_i's, for $i = 0 : m$, are nonconvex functions. Then, the subproblem (4.4) is also nonconvex. Consequently, we only assume that x_{k+1} is a stationary (KKT) point of the subproblem (4.4) satisfying the descent (4.5). Next, we show that the sequence $(F(x_k))_{k\ge0}$ is strictly noninecreasing.

Lemma 4.3.1. *Let Assumptions 4.2.1, 4.2.2, and 4.2.3 hold and $(x_k)_{k\ge0}$ be generated by MTA algorithm with $M_p > L_p$, $x_0 \in \mathcal{F}$ and $M_q^i \ge L_q^i$ for all $i = 1 : m$. Then, we have:*

(i) The sequence $(F(x_k))_{k\ge0}$ is nonincreasing and satisfies the descent:

$$F(x_{k+1}) \le F(x_k) - \left(\frac{M_p - L_p}{(p+1)!}\|x_{k+1}-x_k\|^{p+1} + \frac{M}{(q+1)!}\|x_{k+1}-x_k\|^{q+1}\right).$$

(ii) The set $\mathcal{F}(x_k)$ is nonempty, $\mathcal{F}(x_k) \subseteq \mathcal{F}$ for all $k \geq 0$, and additionally the sequence $(x_k)_{k\geq 0}$ is feasible for the original problem (4.1), bounded, and has a finite length:

$$\sum_{k=0}^{\infty} \left(\|x_{k+1} - x_k\|^{q+1} + \|x_{k+1} - x_k\|^{p+1} \right) < \infty.$$

Proof. (i) Writing inequality (4.5) explicitly, we have:

$$T_p^{F_0}(x_{k+1}; x_k) + \frac{M_p}{(p+1)!}\|x_{k+1}{-}x_k\|^{p+1} + \frac{M}{(q+1)!}\|x_{k+1}{-}x_k\|^{q+1} + h(x_{k+1}) \leq F(x_k).$$

On the other hand, from (4.2) we have:

$$-\frac{L_p}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + F_0(x_{k+1}) \leq T_p^{F_0}(x_{k+1}; x_k),$$

which, combined with the previous inequality, yields:

$$\frac{M_p - L_p}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + \frac{M}{(q+1)!}\|x_{k+1} - x_k\|^{q+1} + F(x_{k+1}) \leq F(x_k),$$

proving the first statement of the lemma.

(ii) Further, if $M_q^i \geq L_q^i$ and $x_0 \in \mathcal{F}$, then the subproblem (4.4) is well-defined, i.e., the feasible set $\mathcal{F}(x_k) \neq \emptyset \, \forall \, k \geq 0$. Additionally, $\mathcal{F}(x_k) \subseteq \mathcal{F}$ and thus x_k is feasible for the original problem (4.1) for all $k \geq 0$. Indeed, note that $\mathcal{F}(x_0) \neq \emptyset$ since $x_0 \in \mathcal{F}(x_0)$ (recall that $x_0 \in \mathcal{F}$, hence we have $s_{M_q^i}(x_0; x_0) = F_i(x_0) \leq 0$ for all $i = 1 : m$). Now let us prove that x_1 is feasible for problem (4.1). Since $M_q^i \geq L_q^i$, for all $i = 1 : m$, then from Assumption 4.2.1.2 and relation (4.3), we get:

$$\begin{aligned} F_i(y) &\leq T_q^{F_i}(y; x_0) + \frac{L_q^i}{(q+1)!}\|y - x_0\|^{q+1} \\ &= s_{L_q^i}(y; x_0) \leq s_{M_q^i}(y; x_0) \leq 0 \quad \forall i = 1 : m, \ \forall y \in \mathbb{E}. \end{aligned}$$

Consequently, $\mathcal{F}(x_0) \subseteq \mathcal{F}$. Additionally, given that x_1 is feasible for the subproblem (4.4), i.e., $x_1 \in \mathcal{F}(x_0)$ or equivalently $s_{M_q^i}(x_1; x_0) \leq 0 \;\; \forall i = 1 : m$, we further get:

$$F_i(x_1) \leq s_{M_q^i}(x_1; x_0) \leq 0 \;\; \forall i = 1 : m.$$

Therefore, the iterate x_1 is also feasible for the original problem (4.1). By induction, using the same arguments as before, we can easily prove that $\mathcal{F}(x_k) \neq \emptyset$, $\mathcal{F}(x_k) \subseteq \mathcal{F}$ and thus x_k is feasible for the original problem (4.1) for all $k \geq 0$. Further, since $(F(x_k))_{k\geq 0}$ is nonincreasing, then $x_k \in \mathcal{A}(x_0)$ and hence from Assumption 4.2.2 the sequence $(x_k)_{k\geq 0}$ is bounded. Finally, the last statement follows by summing up the descent inequality in function values from (i). ■

From the previous lemma, it follows that there exists $D > 0$ such that $\|x_k\| \leq D$ for all $k \geq 0$. Therefore, the sequence $(x_k)_{k\geq 0}$ has limit points. In the sequel, we impose the following assumption, which states that subproblem (4.4) admits KKT points.

Assumption 4.3.2. There exist multipliers $u^{k+1} = (u_1^{k+1}, \cdots, u_m^{k+1}) \geq 0$ and $\Lambda_{k+1} \in \partial h(x_{k+1})$ such that the following KKT conditions hold for (4.4):

$$\begin{aligned} &\nabla s_{M_p}^{M}(x_{k+1}; x_k) + \Lambda_{k+1} + \sum_{i=1}^{m} u_i^{k+1} \nabla s_{M_q^i}(x_{k+1}; x_k) = 0, \qquad (4.6) \\ &u_i^{k+1} s_{M_q^i}(x_{k+1}; x_k) = 0, \;\; s_{M_q^i}(x_{k+1}; x_k) \leq 0 \quad \forall i = 1 : m. \end{aligned}$$

Note that, in general, if the original problem (4.1) satisfies some constraint qualifications (e.g., MFCQ) on $\mathcal{F}$, then the corresponding subproblem (4.4) may satisfy some constraint qualifications as well, which however are not necessarily of the same type as that of the original problem. For example, if the original problem satisfies MFCQ and the subproblem (4.4) is convex, then the Slater condition holds for the subproblem and consequently, Assumption 4.3.2 is valid at x_{k+1}. Indeed, let us prove that the Slater condition holds under MFCQ (Assumption 4.2.3), i.e., for any $x \in \mathcal{F}$ fixed, there exists a $\zeta \in \mathbb{E}$ such that the inequality constraints in (4.4) hold strictly for all $i = 1 : m$, i.e.:

$$F_i(x) + \langle \nabla F_i(x), \zeta - x \rangle + \sum_{j=2}^{q} \frac{1}{j!} \nabla^j F_i(x)[\zeta - x]^j + \frac{M_q^i}{(q+1)!} \|\zeta - x\|^{q+1} < 0.$$

Using a similar argument as in [24], we show that a point of the form $\zeta = x + td$, with $t \in \mathbb{R}_+$ and $d \in \mathbb{E}$ such that $\|d\| = 1$, satisfies strictly these inequalities provided that t is sufficiently small. Indeed, for all $i \notin I(x)$ we have $F_i(x) < 0$ and thus:

$$\begin{aligned}
&F_i(x) + \langle \nabla F_i(x), \zeta - x \rangle + \sum_{j=2}^{q} \frac{1}{j!} \nabla^j F_i(x)[\zeta - x]^j + \frac{M_q^i}{(q+1)!} \|\zeta - x\|^{q+1} \\
&= F_i(x) + t\langle \nabla F_i(x), d \rangle + \sum_{j=2}^{q} t^j \frac{1}{j!} \nabla^j F_i(x)[d]^j + t^{q+1} \frac{M_q^i}{(q+1)!} \|d\|^{q+1} \\
&\leq F_i(x) + t\|\nabla F_i(x)\| + \sum_{j=2}^{q} t^j \frac{1}{j!} \|\nabla^j F_i(x)\| + t^{q+1} \frac{M_q^i}{(q+1)!} < 0,
\end{aligned}$$

where the first inequality follows from Cauchy-Schwartz and the last inequality from $F_i(x) < 0$ and t is sufficiently small. If $i \in I(x)$, then from Assumption 4.2.3 we have $\langle \nabla F_i(x), d \rangle < 0$ for some d and $F_i(x) = 0$. Hence, using a similar argument, we have:

$$\begin{aligned}
&F_i(x) + \langle \nabla F_i(x), \zeta - x \rangle + \sum_{j=2}^{q} \frac{1}{j!} \nabla^j F_i(x)[\zeta - x]^j + \frac{M_q^i}{(q+1)!} \|\zeta - x\|^{q+1} \\
&= t\langle \nabla F_i(x), d \rangle + \sum_{j=2}^{q} t^j \frac{1}{j!} \|\nabla^j F_i(x)\| + t^{q+1} \frac{M_q^i}{(q+1)!} < 0,
\end{aligned}$$

provided that t is sufficiently small. This shows that the Taylor approximation inequality constraints in (4.4) have a nonempty interior, and thus the Slater condition holds for the subproblem (4.4). However, if the subproblem (4.4) is nonconvex, we are not aware of any result guaranteeing the existence of Lagrange multipliers that together with x_{k+1} satisfy Assumption 4.3.2, even if the original problem satisfies certain constraint qualifications. However, in our convergence analysis below we have the flexibility to relax Assumption 4.3.2 by considering that x_{k+1} satisfies together with some $(u_i^{k+1})_{i=1}^m \geq 0$ the following Complementary Approximate KKT (CA-KKT) conditions (see also [50, 51, 92]):

$$\begin{aligned}
&\left\| \nabla s_{M_p}^M(x_{k+1}; x_k) + \Lambda_{k+1} + \sum_{i=1}^{m} u_i^{k+1} \nabla s_{M_q^i}(x_{k+1}; x_k) \right\| \leq \eta_1 \|x_{k+1} - x_k\|^{\min(p,q)}, \\
&|u_i^{k+1} s_{M_q^i}(x_{k+1}; x_k)| \leq \eta_2 \|x_{k+1} - x_k\|^{q+1}, \; \Big(s_{M_q^i}(x_{k+1}; x_k)\Big)_+ \leq \frac{\eta_3}{(q+1)!} \|x_{k+1} - x_k\|^{q+1},
\end{aligned} \tag{4.7}$$

for all $i = 1 : m$ and for some $\eta_1, \eta_2, \eta_3 > 0$, where $(a)_+ = \max(0, a)$. Note that in [50] (page 3) it has been shown that at every local minimizer of subproblem (4.4) there exist x_{k+1} and $(u_i^{k+1})_{i=1}^m \geq 0$ such that CA-KKT conditions (4.7) hold. Similar approximate optimality conditions have been considered in [51, 92]. For simplicity of the exposition we assume below that x_{k+1} satisfies KKT conditions from Assumption 4.3.2, although our convergence results can also be derived using the complementary approximate optimality KKT conditions, CA-KKT, from above (in all the proofs, we sketch how our derivations remain valid when replacing Assumption 4.3.2 with CA-KKT (4.7)). For example, if x_{k+1} satisfies (4.7), then x_{k+1} is feasible for the original problem (4.1) provided that $M_q^i \geq \eta_3 + L_q^i$ for all $i = 1 : m$. Indeed, from (2.4):

$$\frac{M_q^i - L_q^i}{(q+1)!}\|x_{k+1} - x_k\|^{q+1} + F_i(x_{k+1}) \leq s_{M_q^i}(x_{k+1}; x_k) \leq \left(s_{M_q^i}(x_{k+1}; x_k)\right)_+,$$

for all $i = 1 : m$, this implies that:

$$F_i(x_{k+1}) \leq \frac{\eta_3 + L_q^i - M_q^i}{(q+1)!}\|x_{k+1} - x_k\|^{q+1} \leq 0 \quad \forall i = 1 : m.$$

Next, we show that the sequence of the multipliers $(u^k)_{k\geq 1}$ given in (4.6) is bounded.

Lemma 4.3.3. *Let Assumptions 4.2.1, 4.2.2, 4.2.3 and 4.3.2 hold. Then, the multipliers $(u^k)_{k\geq 0}$ defined in (4.6) are bounded, i.e., there exists $C_u > 0$ such that:*

$$\|u^k\| \leq C_u \ \ \forall k \geq 0.$$

Proof. We employ a similar line of reasoning to that used in [24, 25]. Assume, for the sake of contradiction, that there exist $u_i \geq 0$ for $i = 1 : m$, $x_\infty \in \mathcal{F}$, a subsequence $(u^k)_{k\in K}$, and $(x_k)_{k\in K}$, with $K \subseteq \mathbb{N}$, as well as $\Lambda_{k+1} \in \partial h(x_{k+1})$, such that the following holds:

$$\lim_{k\to\infty, k\in K} \sum_{i=1}^m u_i^{k+1} = +\infty, \quad \lim_{k\to\infty, k\in K} \frac{u_i^{k+1}}{\sum_{i=1}^m u_i^{k+1}} = u_i, \ \text{with} \sum_{i=1}^m u_i = 1,$$

and additionally $\lim_{k\to\infty, k\in K} x_{k+1} = x_\infty$ and $\lim_{k\to\infty, k\in K} \Lambda_{k+1} = \Lambda \in \partial h(x_\infty)$ (note that $\partial h(x)$ is closed and bounded for all $x \in \operatorname{dom} h$ since h is assumed to be locally Lipschitz, see Theorem 9.13 in [39]). Further, dividing the first and second equalities in (4.6) (i.e., Assumption (4.3.2)) with the quantity $\sum_{i=1}^m u_i^{k+1}$, we get:

$$\frac{1}{\sum_{i=1}^m u_i^{k+1}}\left(\nabla s_{M_p}^M(x_{k+1}; x_k) + \Lambda_{k+1} + \sum_{i=1}^m u_i^{k+1}\nabla s_{M_q^i}(x_{k+1}; x_k)\right) = 0,$$
$$\frac{u_i^{k+1}}{\sum_{i=1}^m u_i^{k+1}}\left(s_{M_q^i}(x_{k+1}; x_k)\right) = 0, \ \ s_{M_q^i}(x_{k+1}; x_k) \leq 0, \ \ \text{for } i = 1 : m.$$

Since the Taylor functions $s_{M_p}^M$ and $s_{M_q^i}$'s, for $i = 1 : m$, are continuous, then passing to the limit as $k \to \infty$, $k \in K$, we obtain:

$$\sum_{i=1}^m u_i \nabla s_{M_q^i}(x_\infty; x_\infty) = 0, \qquad u_i s_{M_q^i}(x_\infty; x_\infty) = 0, \ s_{M_q^i}(x_\infty; x_\infty) \leq 0 \ \text{ for } i = 1 : m.$$

Notice that similar relation can be obtained by substituting Assumption 4.3.2 with CA-KKT (4.7), while also using the fact that $\|x_{k+1} - x_k\| \to 0$ as k approaches infinity. According to the definition of $s_{M_q^i}$, for $i = 1 : m$, it can be deduced that:

$$\sum_{i=1}^m u_i \nabla F_i(x_\infty) = 0, \ \ u_i F_i(x_\infty) = 0, \ \ F_i(x_\infty) \leq 0 \ \text{ for } i = 1 : m.$$

If $I(x_\infty) = \emptyset$ (see Assumption 4.2.3), then for all $i = 1 : m$, $F_i(x_\infty) < 0$ and hence $u_i = 0$, for $i = 1 : m$. This is a contradiction with $\sum_{i=1}^m u_i = 1$. Further, assume that $I(x_\infty) \neq \emptyset$. Since we have $\sum_{i=1}^m u_i = 1$, then there exists $\mathcal{I} \subseteq I(x_\infty)$, $\mathcal{I} \neq \emptyset$, such that $u_i > 0$ for all $i \in \mathcal{I}$. From Assumption 4.2.3, there exists $d \in \mathbb{E}$ such that:

$$0 = \left\langle \sum_{i=1}^m u_i \nabla F_i(x_\infty), d \right\rangle = \sum_{i\in\mathcal{I}} u_i \langle \nabla F_i(x_\infty), d\rangle < 0,$$

which is a contradiction with MFCQ assumption. Hence our statement follows. ∎

4.3.1 Convergence rate to KKT points

In the general nonconvex case we want to see how fast we can satisfy (approximately) the KKT optimality conditions for the problem (4.1). We consider points satisfying the first order local necessary optimality conditions for problem (4.1), i.e., points which belong to $\mathcal{S}$:

$$\begin{aligned} \mathcal{S} = \{ & x \in \mathcal{F} : \exists\, u_i \geq 0,\ \Lambda \in \partial h(x) \text{ s.t. :} \\ & \left. \nabla F_0(x) + \Lambda + \sum_{i=1}^m u_i \nabla F_i(x) = 0, u_i F_i(x) = 0,\ i = 1:m \right\}. \end{aligned} \tag{4.8}$$

Hence, an appropriate measure of optimality is optimality and complementary violations of KKT conditions. Therefore, for $\Lambda_{k+1} \in \partial h(x_{k+1})$ we define the map:

$$\begin{aligned} \mathcal{M}(x_{k+1}) = \max \Bigg\{ & \left\| \nabla F_0(x_{k+1}) + \Lambda_{k+1} + \sum_{i=1}^m u_i^{k+1} \nabla F_i(x_{k+1}) \right\|, \\ & \Big(-u_i^{k+1} F_i(x_{k+1}) \Big)^{\frac{q}{q+1}},\ i = 1:m \Bigg\}. \end{aligned}$$

Assume $M_p > L_p$ and, for simplicity, let us introduce the following constants $C_1 = \frac{L_p+M_p}{p!}$, $C_2 = \left(\frac{C_u \sum_{i=1}^m (M_q^i + L_q^i) + M}{q!} + \left(C_u \max\limits_{i=1:m} \frac{M_q^i + L_q^i}{(q+1)!} \right)^{\frac{q}{q+1}} \right)$ and

$$\begin{aligned} C = \max \Bigg(& \frac{((q+1)!)^{\frac{q}{q+1}} \left(C_1 (2D)^{p-q} + C_2 \right)^{\frac{q+1}{q}} (q+1)!}{M^{\frac{q}{q+1}}}, \\ & \frac{((p+1)!)^{\frac{p}{p+1}} \left(C_1 + C_2 (2D)^{q-p} \right)^{\frac{p+1}{p}} (p+1)!}{(M_p - L_p)^{\frac{p}{p+1}}} \Bigg). \end{aligned}$$

Then, we have the following convergence rate for the measure of optimality $\mathcal{M}(x_k)$:

Theorem 4.3.4. *Let the assumptions of Lemma 4.3.1 and, additionally, Assumption 4.3.2 hold. Let $(x_k)_{k\geq 0}$ be generated by MTA algorithm. Then, there exists $\Lambda_{k+1} \in \partial h(x_{k+1})$ such that:*

(i) The following bound hold:

$$\mathcal{M}(x_{k+1}) \leq C_1 \|x_{k+1} - x_k\|^p + C_2 \|x_{k+1} - x_k\|^q.$$

(ii) The sequence $(\mathcal{M}(x_k))_{k\geq 0}$ converges to 0 with the following sublinear rate:

$$\min_{j=1:k} \mathcal{M}(x_j) \leq \frac{C(F(x_0) - F_\infty)}{k^{\min\left(\frac{q}{q+1}, \frac{p}{p+1}\right)}}.$$

Proof. Since $u_i^{k+1} \geq 0$, then from (4.3) we get for all $i = 1 : m$:

$$
\begin{aligned}
-u_i^{k+1}\frac{L_q^i}{(q+1)!}\|x_{k+1}-x_k\|^{q+1} &\leq u_i^{k+1}\Big(F_i(x_{k+1}) - T_q^{F_i}(x_{k+1};x_k)\Big)\\
&= u_i^{k+1}F_i(x_{k+1}) - u_i^{k+1}\left(T_q^{F_i}(x_{k+1};x_k) + \frac{M_q^i}{(q+1)!}\|x_{k+1}-x_k\|^{q+1}\right)\\
&\quad + u_i^{k+1}\frac{M_q^i}{(q+1)!}\|x_{k+1}-x_k\|^{q+1}\\
&= u_i^{k+1}F_i(x_{k+1}) + u_i^{k+1}\frac{M_q^i}{(q+1)!}\|x_{k+1}-x_k\|^{q+1},
\end{aligned}
$$

where the last equality follows from (4.6) (i.e., Assumption (4.3.2)). Note that a similar relation can be derived when replacing Assumption 4.3.2 with CA-KKT (4.7). Since the multipliers are bounded, taking the maximum, we get:

$$
\begin{aligned}
\max_{i=1:m}\left\{\Big(-u_i^{k+1}F_i(x_{k+1})\Big)^{\frac{q}{q+1}}\right\} &\leq \max_{i=1:m}\left\{\left(u_i^{k+1}\frac{M_q^i+L_q^i}{(q+1)!}\right)^{\frac{q}{q+1}}\|x_{k+1}-x_k\|^q\right\}\\
&\leq \left(C_u\max_{i=1:m}\frac{M_q^i+L_q^i}{(q+1)!}\right)^{\frac{q}{q+1}}\|x_{k+1}-x_k\|^q.
\end{aligned} \tag{4.9}
$$

Further, let $\Lambda_{k+1} \in \partial h(x_{k+1})$, then we have:

$$
\begin{aligned}
&\left\|\nabla F_0(x_{k+1}) + \Lambda_{k+1} + \sum_{i=1}^m u_i^{k+1}\nabla F_i(x_{k+1})\right\|\\
&\leq \|\nabla F_0(x_{k+1}) - \nabla T_p^{F_0}(x_{k+1};x_k)\| + \Big\|\nabla T_p^{F_0}(x_{k+1};x_k) + \Lambda_{k+1}\\
&\qquad + \frac{M_p}{p!}\|x_{k+1}-x_k\|^{p-1}(x_{k+1}-x_k) + \frac{M}{q!}\|x_{k+1}-x_k\|^{q-1}(x_{k+1}-x_k)\\
&\qquad + \sum_{i=1}^m u_i^{k+1}\left(\nabla T_q^{F_i}(x_{k+1};x_k) + \frac{M_q^i}{q!}\|x_{k+1}-x_k\|^{q-1}(x_{k+1}-x_k)\right)\Big\|\\
&\quad + \left\|-\frac{M_p}{p!}\|x_{k+1}-x_k\|^{p-1}(x_{k+1}-x_k) - \frac{\sum_{i=1}^m u_i^{k+1}M_q^i + M}{q!}\|x_{k+1}-x_k\|^{q-1}(x_{k+1}-x_k)\right\|\\
&\quad + \left\|\sum_{i=1}^m u_i^{k+1}\left(\nabla F_i(x_{k+1}) - \nabla T_q^{F_i}(x_{k+1};x_k)\right)\right\|\\
&\leq \frac{L_p+M_p}{p!}\|x_{k+1}-x_k\|^p + 0 + \frac{\sum_{i=1}^m u_i^{k+1}(M_q^i+L_q^i) + M}{q!}\|x_{k+1}-x_k\|^q\\
&\leq \frac{L_p+M_p}{p!}\|x_{k+1}-x_k\|^p + \frac{C_u\sum_{i=1}^m (M_q^i+L_q^i) + M}{q!}\|x_{k+1}-x_k\|^q,
\end{aligned}
$$

where the second inequality follows from (4.2), (4.3), and Assumption 4.3.2. Note that one can replace zero from above, due to Assumption 4.3.2, with the right hand side in the approximate optimality condition from CA-KKT (4.7) and the subsequent derivations still follow. The last inequality follows from Lemma 4.3.3. Combining this inequality with (4.9), we get:

$$\mathcal{M}(x_{k+1}) \leq \frac{L_p + M_p}{p!}\|x_{k+1} - x_k\|^p + \left(\frac{C_u \sum_{i=1}^m (M_q^i + L_q^i) + M}{q!}\right.$$
$$\left. + \left(C_u \max_{i=1:m} \frac{M_q^i + L_q^i}{(q+1)!}\right)^{\frac{q}{q+1}}\right) \|x_{k+1} - x_k\|^q = C_1\|x_{k+1} - x_k\|^p + C_2\|x_{k+1} - x_k\|^q.$$

Hence, the first assertion follows. Further, if $q \leq p$, then it follows that:

$$\mathcal{M}(x_{k+1}) \leq \left(C_1\|x_{k+1} - x_k\|^{p-q} + C_2\right) \|x_{k+1} - x_k\|^q.$$

Since we have $\|x_{k+1} - x_k\| \leq 2D$ (see Lemma 4.3.1), then:

$$\mathcal{M}(x_{k+1}) \leq \left(C_1(2D)^{p-q} + C_2\right) \|x_{k+1} - x_k\|^q.$$

Combining this inequality with the descent (4.5), we get:

$$\mathcal{M}(x_{k+1})^{\frac{q+1}{q}} \leq \frac{\left(C_1(2D)^{p-q} + C_2\right)^{\frac{q+1}{q}} (q+1)!}{M}\big(F(x_k) - F(x_{k+1})\big).$$

Summing up this inequality and taking the minimum, we obtain:

$$\min_{j=1:k} \mathcal{M}(x_j) \leq \frac{((q+1)!)^{\frac{q}{q+1}}(C_1(2D)^{p-q} + C_2)(F(x_0) - F_\infty)}{M^{\frac{q}{q+1}} k^{\frac{q}{q+1}}}. \tag{4.10}$$

On the other hand, if $p \leq q$, then we also have:

$$\begin{aligned}\mathcal{M}(x_k) &\leq \left(C_1 + C_2\|x_{k+1} - x_k\|^{q-p}\right) \|x_{k+1} - x_k\|^p \\ &\leq \left(C_1 + C_2(2D)^{q-p}\right) \|x_{k+1} - x_k\|^p.\end{aligned}$$

Combining this inequality with the descent (4.5), we get:

$$\mathcal{M}(x_{k+1})^{\frac{p+1}{p}} \leq \frac{\left(C_1 + C_2(2D)^{q-p}\right)^{\frac{p+1}{p}} (p+1)!}{M_p - L_p} \left(F(x_k) - F(x_{k+1})\right).$$

Summing up this inequality and taking the minimum, we obtain:

$$\min_{j=1:k} \mathcal{M}(x_j) \leq \frac{((p+1)!)^{\frac{p}{p+1}}(C_1 + C_2(2D)^{q-p})(F(x_0) - F_\infty)}{(M_p - L_p)^{\frac{p}{p+1}} k^{\frac{p}{p+1}}}. \tag{4.11}$$

Hence, combining inequalities (4.10) and (4.11), our assertion follows. ■

Remark 4. Theorem 4.3.4, shows that there exist a subsequence of the sequence $(x_k)_{k\geq 0}$, generated by MTA algorithm, which converges to a KKT point of the original problem (4.1). If $p = q$, then the convergence rate is of order $\mathcal{O}(k^{-\frac{p}{p+1}})$, which is the usual convergence rate for higher-order algorithms for (unconstrained) nonconvex problems [92, 20, 97, 93]. If $M = 0$ (in this case (4.10) is replaced with an inequality similar to the one in (4.11)), then the convergence rate in the minimum of the optimality map $\mathcal{M}(x_k)$ is of order $\mathcal{O}\left(k^{-\min\left(\frac{q}{p+1}, \frac{p}{p+1}\right)}\right)$. Thus, if $q \leq p$ we have $\frac{q}{p+1} \leq \frac{p}{p+1}$, and hence the convergence rate becomes $\mathcal{O}\left(k^{-\frac{q}{p+1}}\right)$, which is worse than the rate $\mathcal{O}\left(k^{-\frac{q}{q+1}}\right)$. For a better understanding of this situation, consider a particular case: $p = 2$ and $q = 1$. Then, for $M = 0$, the rate is $\mathcal{O}(k^{-\frac{1}{3}})$, while for $M > 0$ the rate is

$\mathcal{O}(k^{-\frac{1}{2}})$. In conclusion, it is beneficial to have additionally the regularization of order $q+1$ in the objective function since it leads to faster convergence rates.

4.3.2 Better convergence under KL

In this section, we derive convergence rates for our algorithm under the KL property. To this end, consider the following Lagrangian function for the problem (4.1) and for the subproblem given in (4.4):

$$\mathcal{L}_p(x;u) = F(x) + \sum_{i=1}^m u_i F_i(x), \quad \mathcal{L}_{sp}(y;x;u) = s_{M_p}^M(y;x) + h(x) + \sum_{i=1}^m u_i s_{M_q^i}(y;x).$$

Next, we establish the following results, known as descent-ascent [98]:

Lemma 4.3.5. *Let the assumptions of Theorem 4.3.4 hold and assume that $M_q^i > L_q^i$ for $i = 1:m$. Then, we have:*

$$\mathcal{L}_p(x_{k+1};u^{k+1}) - \mathcal{L}_p(x_k;u^k) \leq -\frac{(M_p - L_p)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \tag{4.12}$$
$$- \left(\frac{M + \sum_{i=1}^m u_i^{k+1}(M_q^i - L_q^i)}{(q+1)!}\right)\|x_{k+1} - x_k\|^{q+1} + \frac{C_u\|M_q + L_q\|}{(q+1)!}\|x_k - x_{k-1}\|^{q+1}.$$

Proof. We have:

$$\begin{aligned}
&\frac{M_p - L_p}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + \frac{M}{(q+1)!}\|x_{k+1} - x_k\|^{q+1} \leq F(x_k) - F(x_{k+1}) \\
&= F(x_k) - F(x_{k+1}) - \sum_{i=1}^m u_i^{k+1} s_{M_q^i}(x_{k+1};x_k) \\
&\leq F(x_k) - F(x_{k+1}) - \sum_{i=1}^m u_i^{k+1}\left(F_i(x_{k+1}) + \frac{M_q^i - L_q^i}{(q+1)!}\|x_{k+1} - x_k\|^{q+1}\right) \\
&= \mathcal{L}_p(x_k;u^{k+1}) - \mathcal{L}_p(x_{k+1};u^{k+1}) - \sum_{i=1}^m u_i^{k+1}F_i(x_k) - \sum_{i=1}^m u_i^{k+1}\left(\frac{M_q^i - L_q^i}{(q+1)!}\|x_{k+1} - x_k\|^{q+1}\right),
\end{aligned}$$

where the first inequality follows from Lemma 4.3.1 (i), the first equality follows from the KKT conditions (4.6) (i.e., Assumption 4.3.2), the second inequality follows from (2.4) and $u_i^k \geq 0$. The last equality follows from the definition of $\mathcal{L}_p$. Note that a similar relation can be derived if we replace Assumption 4.3.2 with the approximate complementary condition from (4.7), provided that $M \geq \eta_2$. Furthermore, we have:

$$\begin{aligned}
&\mathcal{L}_p(x_k;u^{k+1}) - \mathcal{L}_p(x_{k+1};u^{k+1}) - \sum_{i=1}^m u_i^{k+1}F_i(x_k) \\
&= \mathcal{L}_p(x_k;u^k) - \mathcal{L}_p(x_{k+1};u^{k+1}) + \mathcal{L}_p(x_k;u^{k+1}) - \mathcal{L}_p(x_k;u^k) - \sum_{i=1}^m u_i^{k+1}F_i(x_k) \\
&= \mathcal{L}_p(x_k;u^k) - \mathcal{L}_p(x_{k+1};u^{k+1}) + \sum_{i=1}^m u_i^{k+1}F_i(x_k) - \sum_{i=1}^m u_i^k F_i(x_k) - \sum_{i=1}^m u_i^{k+1}F_i(x_k) \\
&= \mathcal{L}_p(x_k;u^k) - \mathcal{L}_p(x_{k+1};u^{k+1}) - \sum_{i=1}^m u_i^k F_i(x_k),
\end{aligned}$$

where the second equality follows from the definition of the Lagrangian function $\mathcal{L}_p$. On the other hand, from (2.4) and $u_i^k \geq 0,\ i = 1:m$, we have:

$$-\sum_{i=1}^{m} u_i^k F_i(x_k) \leq -\sum_{i=1}^{m} u_i^k s_{M_q^i}(x_k; x_{k-1}) + \frac{\sum_{i=1}^m u_i^k (M_q^i + L_q^i)}{(q+1)!}\|x_k - x_{k-1}\|^{q+1}$$
$$= 0 + \frac{\sum_{i=1}^m u_i^k (M_q^i + L_q^i)}{(q+1)!}\|x_k - x_{k-1}\|^{q+1} \leq \frac{C_u\|M_q + L_q\|}{(q+1)!}\|x_k - x_{k-1}\|^{q+1}.$$

Hence, our statement follows by combining these three inequalities. ■

Consider the following Lyapunov function:

$$\xi_p(x; u; z) := \mathcal{L}_p(x; u) + \frac{\theta_1}{(p+1)!}\|x - z\|^{p+1} + \frac{\theta_2}{(q+1)!}\|x - z\|^{q+1},$$

where θ_1, θ_2 are positive constants that will be defined later. The following lemma derives a relation between the critical points of the functions ξ_p and $\mathcal{L}_p$.

Lemma 4.3.6. *For any $x, y \in \mathbb{E}$ and $u \in \mathbb{R}^m$, it holds that:*

$$(x, u, z) \in \text{crit } \xi_p \Rightarrow (x, u) \in \text{crit } \mathcal{L}_p \ \text{ and } \ \xi_p(x; u; z) = \mathcal{L}_p(x; u).$$

Proof. If $0 \in \partial\xi_p(x; u; z) = \big(\partial_x \xi_p(x; u; z), \nabla_u \xi_p(x; u; z), \nabla_z \xi_p(x; u; z)\big)$, then:

$$0 \in \partial_x \xi_p(x; u; z) = \partial_x \mathcal{L}_p(x; u) + \left(\frac{\theta_1}{p!}\|x - z\|^{p-1} + \frac{\theta_2}{q!}\|x - z\|^{q-1}\right)(x - z),$$
$$0 = \nabla_u \xi_p(x; u; z) = \nabla_u \mathcal{L}_p(x; u),$$
$$0 = \nabla_z \xi_p(x; u; z) = \left(\frac{\theta_1}{p!}\|x - z\|^{p-1} + \frac{\theta_2}{q!}\|x - z\|^{q-1}\right)(z - x).$$

Hence, from the last equality we get that $z = x$. This implies that $0 \in \partial\mathcal{L}_p(x; u)$ and $\xi_p(x; u; z) = \mathcal{L}_p(x; u)$, which proves our assertion. ■

Up to this stage, we have not considered any assumption on the constant M given in subproblem (4.4). Hence, by restricting the choice of this constant, we can derive the following descent in the Lyapunov function ξ_p.

Lemma 4.3.7. *Let the assumptions of Theorem 4.3.4 hold and assume that $\theta_1 = \frac{M_p - L_p}{2}$, $\theta_2 = 2C_u\|M_q + L_q\|$ and $M = 3C_u\|M_q + L_q\|$. Then, we have:*

$$\xi_p(x_{k+1}; u^{k+1}; x_k) - \xi_p(x_k; u^k; x_{k-1}) \leq -\frac{(M_p - L_p)}{2(p+1)!}\left(\|x_{k+1} - x_k\|^{p+1} + \|x_k - x_{k-1}\|^{p+1}\right)$$
$$- \left(\frac{C_u\|M_q + L_q\|}{(q+1)!}\right)\left(\|x_{k+1} - x_k\|^{q+1} + \|x_k - x_{k-1}\|^{q+1}\right).$$

Proof. We have:

$$\begin{aligned}
&\xi_p(x_{k+1};u^{k+1};x_k) - \xi_p(x_k;u^k;x_{k-1}) \\
&= \mathcal{L}_p(x_{k+1};u^{k+1}) + \frac{\theta_1}{(p+1)!}\|x_{k+1}-x_k\|^{p+1} + \frac{\theta_2}{(q+1)!}\|x_{k+1}-x_k\|^{q+1} \\
&\quad - \mathcal{L}_p(x_k;u^k) - \frac{\theta_1}{(p+1)!}\|x_k - x_{k-1}\|^{p+1} - \frac{\theta_2}{(q+1)!}\|x_k - x_{k-1}\|^{q+1} \\
&\overset{(4.12)}{\leq} -\frac{(M_p-L_p)-\theta_1}{(p+1)!}\|x_{k+1}-x_k\|^{p+1} - \frac{\theta_1}{(p+1)!}\|x_k - x_{k-1}\|^{p+1} \\
&- \left(\frac{M + \sum_{i=1}^m u_i^{k+1}(M_q^i - L_q^i) - \theta_2}{(q+1)!}\right)\|x_{k+1}-x_k\|^{q+1} - \left(\frac{\theta_2 - C_u\|M_q + L_q\|}{(q+1)!}\right)\|x_k - x_{k-1}\|^{q+1}.
\end{aligned}$$

Then, it follows that:

$$\begin{aligned}
\xi_p(x_{k+1};u^{k+1};x_k) - \xi_p(x_k;u^k;x_{k-1}) &\leq -\frac{(M_p-L_p)}{2(p+1)!}\|x_{k+1}-x_k\|^{p+1} \\
&- \left(\frac{C_u\|M_q+L_q\| + \sum_{i=1}^m u_i^{k+1}(M_q^i - L_q^i)}{(q+1)!}\right)\|x_{k+1}-x_k\|^{q+1} \\
&- \left(\frac{C_u\|M_q+L_q\|}{(q+1)!}\right)\|x_k - x_{k-1}\|^{q+1} - \frac{(M_p-L_p)}{2(p+1)!}\|x_k - x_{k-1}\|^{p+1}.
\end{aligned}$$

Hence, our statement follows. ∎

Remark 5. The main difficulty in getting the descent in the Lagrangian function $\mathcal{L}_p(x;u)$ is the extra positive term that depends on the multipliers (see Theorem 4.3.5). We overcome this challenge by introducing a new Lyapunov function ξ_p for which we can establish the strict descent.

Define the following constants $\beta_1 = \left(C_1 + \frac{M_p - L_p}{2p!}\right)$ and $\beta_2 = \left(C_2 + \frac{2D\|M_q+L_q\|}{(q+1)!}\right)$. Next, we establish a bound on the (sub)gradient of the Lyapunov function ξ_p:

Lemma 4.3.8. *Let the assumptions of Lemma 4.3.7 hold. Then, there exists $G_{k+1} \in \partial\xi_p(x_{k+1};u^{k+1};x_k)$ such that we have the following bound:*

$$\|G_{k+1}\| \leq \beta_1\|x_{k+1}-x_k\|^p + \beta_2\|x_{k+1}-x_k\|^q.$$

Proof. We have:

$$\begin{aligned}
\nabla_z \xi_p(x_{k+1};u^{k+1};z)_{z=x_k} &= \frac{M_p - L_p}{2p!}\|x_{k+1}-x_k\|^{p-1}(x_{k+1}-x_k) \\
&\quad + \frac{2C_u\|M_q+L_q\|}{q!}\|x_{k+1}-x_k\|^{q-1}(x_{k+1}-x_k).
\end{aligned}$$

Then, it follows that:

$$\|\nabla_z \xi_p(x_{k+1};u^{k+1};z)_{z=x_k}\| \leq \frac{M_p-L_p}{2p!}\|x_{k+1}-x_k\|^p + \frac{C_u\|M_q+L_q\|}{q!}\|x_{k+1}-x_k\|^q.$$

Further, for $\Lambda_{k+1} \in \partial h(x_{k+1})$ given in Theorem 4.3.4, we have:

$$\nabla F(x_{k+1}) + \Lambda_{k+1} + \sum_{i=1}^{m} u_i^{k+1} \nabla F_i(x_{k+1}) \in \partial_x \xi_p(x; u^{k+1}; x_k)_{x=x_{k+1}}$$
$$\left\| \nabla F(x_{k+1}) + \Lambda_{k+1} + \sum_{i=1}^{m} u_i^{k+1} \nabla F_i(x_{k+1}) \right\| \leq \mathcal{M}(x_{k+1}),$$

where the last inequality follows from definition of $\mathcal{M}(x_{k+1})$. From (4.3) we have:

$$F_i(x_{k+1}) \leq s_{M_q^i}(x_{k+1}; x_k) + \frac{M_q^i + L_q^i}{(q+1)!} \|x_{k+1} - x_k\|^{q+1} \leq \frac{M_q^i + L_q^i}{(q+1)!} \|x_{k+1} - x_k\|^{q+1},$$

where the last inequality follows from $s_{M_q^i}(x_{k+1}; x_k) \leq 0$ given in Assumption 4.3.2 (a similar result will follow if we replace this feasibility condition with the approximate feasibility from CA-KKT (4.7)). Finally, we get:

$$\begin{aligned} \|\nabla_u \xi_p(x_{k+1}; u; x_k)_{u=u^{k+1}}\| &= \| \left(F_1(x_{k+1}), \cdots, F_m(x_{k+1})\right) \| \\ &\leq \frac{\|L_q + M_q\|}{(q+1)!} \|x_{k+1} - x_k\|^{q+1} \leq \frac{2D\|L_q + M_q\|}{(q+1)!} \|x_{k+1} - x_k\|^{q}. \end{aligned}$$

Denote $G_{k+1} = \Big(\nabla F(x_{k+1}) + \Lambda_{k+1} + \sum_{i=1}^{m} u_i^{k+1} F_i(x_{k+1}); \nabla_u \xi_p(x_{k+1}; u^{k+1}; x_k)\ ;$ $\nabla_z \xi_p(x_{k+1}; u^{k+1}; z)_{z=x_k}\Big)$. Then, combining the last three inequalities, we get:

$$\begin{aligned} \|G_{k+1}\| \leq & \frac{M_p - L_p}{2p!} \|x_{k+1} - x_k\|^p + \frac{C_u \|M_q + L_q\|}{q!} \|x_{k+1} - x_k\|^q \\ & + \mathcal{M}(x_{k+1}) + \frac{2D\|M_q + L_q\|}{(q+1)!} \|x_{k+1} - x_k\|^q \\ \leq & \left(C_1 + \frac{M_p - L_p}{2p!}\right) \|x_{k+1} - x_k\|^p + \left(C_2 + \frac{2D\|M_q + L_q\|}{(q+1)!}\right) \|x_{k+1} - x_k\|^q, \end{aligned}$$

where the last inequality follows from Theorem 4.3.4. This proves our assertions. ■

From Lemma 4.3.7, we have that $\left(\xi_p(x_k, u^k, x_{k-1})\right)_{k\geq 1}$ is monotonically nonincreasing. Since ξ_p is continuous, then it is bounded from below, and hence $\left(\xi_p(x_k, u^k, x_{k-1})\right)_{k\geq 1}$ convergences, let us say to ξ_p^*. For simplicity, we assume that $p \leq q$ and denote $S_k := \xi_p(x_k; u^k; x_{k-1}) - \xi_p^*$. Next, we establish global convergence.

Theorem 4.3.9. *Let the assumptions of Lemma 4.3.7 hold, and let $(x_k)_{k\geq 0}$ be generated by MTA algorithm. Then, the following holds:*

1. *If ξ_p satisfies the KL property at (x^*, u^*, x^*), where u^* is a limit point of the bounded sequence $(u^k)_{k\geq 1}$, and x^* is a limit point of $(x_k)_{k\geq 0}$, then the hull sequence $(x_k)_{k\geq 0}$ converges to x^* and there exists $k_1 \geq 1$ such that:*

$$\|x_k - x^*\| \leq \rho \max\left(\kappa(S_k), S_k^{\frac{1}{p+1}}\right) \quad \forall k \geq k_1.$$

2. *Moreover, if ξ_p satisfies KL with $\kappa(s) = s^{1-\nu}$, where $\nu \in [0, 1)$, then the following convergence rates hold:*

   i. *If $\nu = 0$, then x_k converges to x^* in a finite number of iterations.*

ii. If $\nu \in \left(0, \frac{p}{p+1}\right]$, then we have the following linear rate:

$$\|x_k - x^*\| \leq \rho \left(\frac{\Gamma^{\frac{p}{\nu(p+1)^2}}}{(1+\Gamma^{\frac{p}{\nu(p+1)}})^{\frac{1}{p+1}}} \right)^{k-(1+k_1)} S_0^{\frac{1}{p+1}} \quad \forall k > k_1.$$

iii. If $\nu \in \left(\frac{p}{p+1}, 1\right)$, then there exists $\alpha > 0$ such that the following sublinear rate holds:

$$\|x_k - x^*\| \leq \frac{\rho\alpha^{1-\nu}}{(k-k_1)^{\frac{p(1-\nu)}{\nu(p+1)-p}}} \quad \forall k > k_1.$$

Proof. For simplicity, denote $\xi_p^k = \xi_p(x_k; u^k; x_{k-1})$ and consider $p \leq q$ (the case where $q \leq p$ is similar). From Lemma 4.3.7, we have:

$$\|x_{k+1} - x_k\|^{p+1} \leq \frac{2(p+1)!}{M_p - L_p}(\xi_p^k - \xi_p^{k+1}) = \frac{2(p+1)!}{M_p - L_p}(S_k - S_{k+1}). \tag{4.13}$$

Further, since ξ_p satisfies the inequality (2.16), then there exists an integer k_1 and $G_k \in \partial\xi_p(x_k; u^k; x_{k-1})$ such that for all $k \geq k_1$ we have:

$$\begin{aligned}
&\|x_{k+1} - x_k\|^{p+1} \leq \|x_{k+1} - x_k\|^{p+1}\kappa^{'}(S_k)\|G_k\| \\
&\leq \frac{2(p+1)!}{M_p - L_p}\kappa^{'}(S_k)(S_k - S_{k+1})\|G_k\| \leq \frac{2(p+1)!}{M_p - L_p}(\kappa(S_k) - \kappa(S_{k+1}))\|G_k\| \\
&\leq \frac{2(p+1)!}{M_p - L_p}\left(\kappa\left(S_k\right) - \kappa\left(S_{k+1}\right)\right)(\beta_1 + \beta_2 D^{q-p})\|x_k - x_{k-1}\|^p \\
&= \frac{2(\beta_1 + \beta_2 D^{q-p})(p+1)!}{M_p - L_p}(\kappa(S_k) - \kappa(S_{k+1}))\|x_k - x_{k-1}\|^p,
\end{aligned}$$

where the second inequality follows from (4.13), the third inequality follows from κ is concave, and the last inequality follows from Lemma 4.3.8. For simplicity, let's define $T = \frac{2(\beta_1+\beta_2 D^{q-p})\cdot p!}{M_p - L_p}$. We can then derive the following:

$$\begin{aligned}
\|x_{k+1} - x_k\| &\leq \Big(T(p+1)\big(\kappa(S_k) - \kappa(S_{k+1})\big)\Big)^{\frac{1}{p+1}}\|x_k - x_{k-1}\|^{\frac{p}{p+1}} \\
&\leq \frac{T(p+1)}{p+1}\big(\kappa(S_k) - \kappa(S_{k+1})\big) + \frac{p}{p+1}\|x_k - x_{k-1}\| \\
&= T\big(\kappa(S_k) - \kappa(S_{k+1})\big) + \frac{p}{p+1}\|x_k - x_{k-1}\|,
\end{aligned}$$

where in the second inequality we use the following classical result: if a, b are positive constants and $0 \leq \alpha_1, \alpha_2 \leq 1$, such that $\alpha_1 + \alpha_2 = 1$, then $a^{\alpha_1}b^{\alpha_2} \leq \alpha_1 a + \alpha_2 b$. Summing up the above inequality over $k \geq k_1$, we get:

$$\sum_{k\geq k_1} \|x_{k+1} - x_k\| \leq (p+1)T\kappa(S_{k_1}) + p\|x_{k_1} - x_{k_1-1}\|.$$

Hence, it follows that $(x_k)_{k\geq 0}$ is a Cauchy sequence and thus converges to x^*. Further:

$$\begin{aligned}
&\|x_k - x^*\| \leq \sum_{t\geq k}\|x_{t+1} - x_t\| \leq (p+1)T\kappa(S_k) + p\|x_k - x_{k-1}\| \\
&\leq (p+1)T\kappa(S_k) + p\left(\frac{2(p+1)!}{M_p - L_p}\right)^{\frac{1}{p+1}} S_{k-1}^{\frac{1}{p+1}} \leq \rho\max\left(\kappa(S_k), S_{k-1}^{\frac{1}{p+1}}\right),
\end{aligned}$$

where the third inequality follows from (4.13) and $S_k \geq 0$. The last inequality is straightforward by introducing $\rho = 2\max\left((p+1)T, p\left(\frac{2(p+1)!}{M_p-L_p}\right)^{\frac{1}{p+1}}\right)$. Thus, we have:

$$\|x_k - x^*\| \leq \rho \max\left(\kappa(S_k), S_{k-1}^{\frac{1}{p+1}}\right). \tag{4.14}$$

Let us assume now that $\kappa(s) = s^{1-\nu}$, where $\nu \in [0,1)$. Then, it follows that:

$$\|x_k - x^*\| \leq \rho \max\left(S_k^{1-\nu}, S_{k-1}^{\frac{1}{p+1}}\right). \tag{4.15}$$

Further, from the KL property (2.16) and Lemma 4.3.8, we have:

$$S_k^{\nu} \leq (1-\nu)\|G_k\| \leq (\beta_1 + \beta_2 D^{q-p})\|x_k - x_{k-1}\|^p.$$

Hence, combining this inequality with inequality (4.13), we further get:

$$S_k^{\nu} \leq (\beta_1 + \beta_2 D^{q-p})\left(\frac{2(p+1)!}{M_p - L_p}(S_{k-1} - S_k)\right)^{\frac{p}{p+1}}.$$

Denote $\Gamma = (\beta_1 + B_2 D^{q-p})^{\frac{p+1}{p}} \frac{2(p+1)!}{M_p - L_p}$, then we have the following recurrence:

$$S_k^{\frac{\nu(p+1)}{p}} \leq \Gamma(S_{k-1} - S_k). \tag{4.16}$$

1. Let $\nu = 0$. If $S_k > 0$ for all $k \geq k_1$, then $\frac{1}{\Gamma} \leq S_{k-1} - S_k$. Letting $k \to \infty$ we get $0 < \frac{1}{\Gamma} \leq 0$ which is a contradiction. Hence, there exist $k > k_1$ such that $S_k = 0$ and finally $S_k \to 0$ in a finite number of steps and from (4.15), $x_k \to x^*$ in a finite number of iterations.

2. Let $\nu \in (0, \frac{p}{p+1}]$, then $\frac{\nu(p+1)}{p} \leq 1$ and $1 - \nu \geq \frac{1}{p+1}$. Using Lemma 2.4.2, for $\theta = \frac{\nu(p+1)}{p}$, we further obtain:

$$S_k \leq \left(\frac{\Gamma^{\frac{p}{\nu(p+1)}}}{1 + \Gamma^{\frac{p}{\nu(p+1)}}}\right)^{k-k_1} S_0.$$

   Since $S_k < 1$ for all $k > k_1$ and S_k is nonincreasing, then we have $\max\left(S_k^{1-\nu}, S_{k-1}^{\frac{1}{p+1}}\right) = S_{k-1}^{\frac{1}{p+1}}$ and thus:

$$\|x_k - x^*\| \leq \rho \left(\frac{\Gamma^{\frac{p}{\nu(p+1)^2}}}{(1 + \Gamma^{\frac{p}{\nu(p+1)}})^{\frac{1}{p+1}}}\right)^{k-(1+k_1)} S_0^{\frac{1}{p+1}}.$$

3. Let $\frac{p}{p+1} < \nu < 1$, then $\frac{\nu(p+1)}{p} > 1$ and thus using Lemma 2.4.2 for $\theta = \frac{\nu(p+1)}{p}$, there exists $\alpha > 0$ such that:

$$S_k \leq \frac{\alpha}{(k-k_1)^{\frac{p}{\nu(p+1)-p}}}.$$

   In this case, we have $\max\left(S_k^{1-\nu}, S_{k-1}^{\frac{1}{p+1}}\right) = S_{k-1}^{1-\nu}$ and thus we have:

$$\|x_k - x^*\| \leq \frac{\rho\alpha^{1-\nu}}{(k-k_1)^{\frac{p(1-\nu)}{\nu(p+1)-p}}}.$$

Hence, our assertions follow. ■

Remark 6. In this section, we derived global convergence rates using higher-order information for solving nonconvex problems with functional constraints provided that the Lyapunov ξ_p satisfies the KL property. Note that if F and F_i's, for $i = 1 : m$, satisfy the KL property, then ξ_p also satisfies the KL property. For $p = q = 1$, we recover the convergence results from [25, 89].

4.4 Convex convergence analysis

In this section we assume that the functions F_i's, for $i = 0 : m$, are convex functions. Then, the subproblem (4.4) is also convex and x_{k+1} is a corresponding minimum point for sufficiently large M_p, M_q^i for $i = 1 : m$. Note that since the Lagrangian function (see section 4.3.2):

$$x \mapsto \mathcal{L}_{sp}(y; x; u),$$

is convex (provided that $M_p \geq pL_p$ and $M_q^i \geq qL_q^i$ for $i = 1 : m$, see [12]), then from the optimality condition of x_{k+1} it follows that x_{k+1} is the global minimizer of the function $\mathcal{L}_{sp}(y; x_k; u^{k+1})$. Let introduce the following constants $D_1 = \frac{M_p+L_p}{(p+1)!}$ and $D_2 = \left(\frac{M+\sum_{i=1}^m C_u(M_q^i+L_q^i)}{(q+1)!}\right)$, then we have the following sublinear convergence rate:

Theorem 4.4.1. *Let the assumptions of Lemma 4.3.1 hold, and, additionally, F_i's, for $i = 0 : m$, be convex functions. Let also $(x_k)_{k \geq k}$ be generated by MTA algorithm with $M_p \geq pL_p$ and $M_q^i \geq qL_q^i$ for $i = 1 : m$. Then, we have the following sublinear convergence rate:*

$$F(x_k) - F^* \leq \frac{2\max\Big((p+1)^{p+1}, (q+1)^{q+1}\Big)\Big(D_1(2D)^{p+1} + D_2(2D)^{q+1}\Big)}{k^{\min(p,q)}} \quad \forall k \geq 1.$$

Proof. If the subproblem (4.4) is convex, then we proved on page 8 that the Slater's condition holds for this subproblem and consequently Assumption 4.3.2 is valid. Then, we have:

$$\begin{aligned}
F(x_{k+1}) &\overset{(4.2),(4.6)}{\leq} T_p^{F_0}(x_{k+1}; x_k) + \frac{M_p}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + \frac{M}{(q+1)!}\|x_{k+1} - x_k\|^{q+1} \\
&\quad + h(x) + \sum_{i=1}^m u_i^{k+1}\left(T_q^{F_i}(x_{k+1}; x_k) + \frac{M_q^i}{(q+1)!}\|x_{k+1} - x_k\|^{q+1}\right) \\
&= \min_x T_p^{F_0}(x; x_k) + \frac{M_p}{(p+1)!}\|x - x_k\|^{p+1} + \frac{M}{(q+1)!}\|x - x_k\|^{q+1} \\
&\quad + h(x) + \sum_{i=1}^m u_i^{k+1}\left(T_q^{F_i}(x; x_k) + \frac{M_q^i}{(q+1)!}\|x - x_k\|^{q+1}\right) \\
&\overset{(4.2)}{\leq} \min_x F(x) + \frac{M_p + L_p}{(p+1)!}\|x - x_k\|^{p+1} + \frac{M}{(q+1)!}\|x - x_k\|^{q+1} \\
&\quad + \sum_{i=1}^m u_i^{k+1}\left(F_i(x) + \frac{(M_q^i + L_q^i)}{(q+1)!}\|x - x_k\|^{q+1}\right) \\
&\leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}\frac{M_p + L_p}{(p+1)!}\|x_k - x^*\|^{p+1} \\
&\quad + \frac{\alpha^{q+1}M}{(q+1)!}\|x_k - x^*\|^{q+1} + \sum_{i=1}^m u_i^{k+1}\Big(\alpha F_i(x^*) + (1-\alpha)F_i(x_k)\Big)
\end{aligned}$$

$$
\begin{aligned}
&+ \alpha^{q+1}\frac{u_i^{k+1}(M_q^i+L_q^i)}{(q+1)!}\|x_k-x^*\|^{q+1}\\
&\leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}\frac{M_p+L_p}{(p+1)!}\|x_k-x^*\|^{p+1}\\
&+ \frac{\alpha^{q+1}M}{(q+1)!}\|x_k-x^*\|^{q+1} + \sum_{i=1}^{m}\alpha^{q+1}\frac{u_i^{k+1}(M_q^i+L_q^i)}{(q+1)!}\|x_k-x^*\|^{q+1},
\end{aligned}
$$

where the first equality follows from x_{k+1} being the global minimum of the Lagrangian function $\mathcal{L}_{sp}(y;x_k;u^{k+1})$, the last inequality follows from the fact that x_k and x^* are feasible for the problem (4.1). Since the multipliers are bounded, we get:

$$
\begin{aligned}
F(x_{k+1}) \leq &\min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}\frac{M_p+L_p}{(p+1)!}\|x_k-x^*\|^{p+1}\\
&+ \alpha^{q+1}\left(\frac{M+\sum_{i=1}^m C_u(M_q^i+L_q^i)}{(q+1)!}\right)\|x_k-x^*\|^{q+1}.
\end{aligned}
$$

Hence, we get the following relation:

$$
F(x_{k+1}) \leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}D_1\|x_k-x^*\|^{p+1} + \alpha^{q+1}D_2\|x_k-x^*\|^{q+1}, \quad (4.17)
$$

which implies that:

$$
F(x_{k+1}) \leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}D_1(2D)^{p+1} + \alpha^{q+1}D_2(2D)^{q+1}. \quad (4.18)
$$

If $q \leq p$, then from inequality (4.18), we get:

$$
F(x_{k+1}) \leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right).
$$

Since the previous inequality holds for all $\alpha \in [0,1]$, then we consider:

$$
A_k := k(k+1)\cdots(k+p), \quad a_{k+1} := A_{k+1} - A_k, \quad \alpha_k := \frac{a_{k+1}}{A_{k+1}}.
$$

Therefor, we obtain:

$$
\begin{aligned}
F(x_{k+1}) &\leq \frac{a_{k+1}}{A_{k+1}}F^* + \frac{A_k}{A_{k+1}}F(x_k) + \left(\frac{a_{k+1}}{A_{k+1}}\right)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right)\\
&\leq \frac{a_{k+1}}{A_{k+1}}F^* + \frac{A_k}{A_{k+1}}F(x_k) + \left(\frac{p+1}{k+p+1}\right)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right).
\end{aligned}
$$

Multiplying both sides with A_{k+1}, we get:

$$
\begin{aligned}
A_{k+1}(F(x_{k+1}) - F^*) &\leq A_k(F(x_k)-F^*) + A_{k+1}\left(\frac{p+1}{k+p+1}\right)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right)\\
&\leq A_k(F(x_k)-F^*) + (q+1)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right).
\end{aligned}
$$

Summing up this inequality, we get for all $k \geq 1$:

$$
\begin{aligned}
F(x_k) - F^* &\leq \frac{1}{A_k}k(q+1)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right)\\
&\leq \frac{(q+1)^{q+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right)}{k^q}.
\end{aligned}
$$

Further, if $p \leq q$, then from inequality (4.18), we get:

$$F(x_{k+1}) \leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{p+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right).$$

Following the same procedure as before, we get:

$$F(x_k) - F^* \leq \frac{(p+1)^{p+1}\left(D_1(2D)^{p+1} + D_2(2D)^{q+1}\right)}{k^p}.$$

Hence, our assertion follows. ■

Remark 7. To the best of our knowledge, this is the first convergence rate using higher-order information for a convex problem with smooth functional constraints. Note that for $p = 1$ and $q = 1$, we recover the convergence rate in [21]. If $m = 0$, we recover the convergence rate in [12].

4.4.1 Uniform convex convergence analysis

In this section, we derive (super)linear convergence in function value for the sequence $(x_k)_{k\geq 0}$ generated by MTA algorithm, provided that the objective function F is uniformly convex of degree θ with constant σ and the constraints F_i's are only convex functions. For simplicity, let us introduce the following constants $U = \frac{(D_1(2D)^{p-q}+D_2)(q+1)!}{\sigma}$ and $\bar{U} = \frac{(D_1+D_2(2D)^{q-p})(p+1)!}{\sigma}$, where D_1 and D_2 are defined in Section 4.4. Then, we can establish the following convergence rate in function values:

Theorem 4.4.2. *Let the assumptions of Theorem 4.4.1 hold. Additionally, assume that F is uniformly convex of order θ with constant σ. Then, the following hold:*
(i) If $\theta = \min(p+1, q+1)$, then:

$$F(x_{k+1}) - F^* \leq \max\left(\left(1 - \frac{q}{U^{\frac{1}{q}}(q+1)^{1+\frac{1}{q}}}\right), \left(1 - \frac{p}{\bar{U}^{\frac{1}{p}}(p+1)^{1+\frac{1}{p}}}\right)\right)(F(x_k) - F^*).$$

(ii) If $\theta < \min(p+1, q+1)$, then:

$$F(x_{k+1}) - F^* \leq \max\left(\frac{(D_1 D^{p-q} + D_2)\theta^{\frac{q+1}{\theta}}}{\sigma^{\frac{q+1}{\theta}}}, \frac{(D_1 + D_2 D^{q-p})\theta^{\frac{p+1}{\theta}}}{\sigma^{\frac{p+1}{\theta}}}\right)(F(x_k) - F^*)^{\frac{\min(p+1,q+1)}{\theta}}.$$

Proof. If $q \leq p$, then, from inequality (4.17), we get:

$$F(x_{k+1}) \leq \alpha F^* + (1-\alpha)F(x_k) + \alpha^{q+1}(D_1(2D)^{p-q} + D_2)\|x_k - x^*\|^{q+1}.$$

Let $\Lambda^* \in h(x^*)$. Since F is uniformly convex, we get:

$$\begin{aligned} F(x_k) &\geq F^* + \langle \nabla F_0(x^*) + \Lambda^*, x_k - x^* \rangle + \frac{\sigma}{(q+1)!}\|x_k - x^*\|^{q+1} \\ &\geq F^* + \frac{\sigma}{(q+1)!}\|x_k - x^*\|^{q+1}, \end{aligned}$$

where the last inequality follows from the optimality condition of x^*: $\langle \nabla F(x^*) + \Lambda^*, x - x^* \rangle \geq 0$, $\forall x \in \mathcal{F}$ and that the sequence $(x_k)_{k\geq 0}$ is feasible, i.e., $x_k \in \mathcal{F}$ for all $k \geq 0$. Combining the last two inequalities, we get:

$$F(x_{k+1}) \leq \min_{\alpha\in[0,1]} \alpha F^* + (1-\alpha)F(x_k) + \alpha^{q+1}\frac{(D_1(2D)^{p-q}+D_2)(q+1)!}{\sigma}(F(x_k) - F^*).$$

Hence, it follows that:

$$F(x_{k+1}) - F^* \leq \min_{\alpha\in[0,1]} \left(1-\alpha+\alpha^{q+1}\frac{(D_1(2D)^{p-q}+D_2)(q+1)!}{\sigma}\right)(F(x_k)-F^*).$$

By minimizing the right-hand side over α, the optimal choice is:

$$0 \leq \alpha = \frac{1}{(q+1)^{\frac{1}{q}}U^{\frac{1}{q}}} \leq 1.$$

Replacing this choice in the last inequality, we get:

$$\begin{aligned} F(x_{k+1}) - F^* &\leq \left(1 - \frac{1}{(q+1)^{\frac{1}{q}}U^{\frac{1}{q}}} + \frac{U}{(q+1)^{\frac{q+1}{q}}U^{\frac{q+1}{q}}}\right)(F(x_k) - F^*) \\ &\leq \left(1 - \frac{q}{U^{\frac{1}{q}}(q+1)^{1+\frac{1}{q}}}\right)(F(x_k) - F^*). \end{aligned}$$

Further, if $p \leq q$, then we have:

$$F(x_{k+1}) \leq \alpha_k F^* + (1-\alpha_k)F(x_k) + \alpha^{p+1}(D_1 + D_2(2D)^{q-p})\|x_k - x^*\|^{p+1}.$$

Since F is uniformly convex of degree $p+1$, we get:

$$F(x_k) \geq F^* + \frac{\sigma}{(p+1)!}\|x_k - x^*\|^{p+1}.$$

Combining the last two inequalities and following the same procedure as in the first case, we get the following statement:

$$F(x_{k+1}) - F^* \leq \left(1 - \frac{p}{\bar{U}^{\frac{1}{p}}(p+1)^{1+\frac{1}{p}}}\right)(F(x_k) - F^*).$$

Hence, our first assertion holds. Further, we have:

$$F(x_k) \geq F^* + \frac{\sigma}{\theta}\|x_k - x^*\|^{\theta}. \tag{4.19}$$

Taking $\alpha = 1$ in inequality (4.17) we get:

$$F(x_{k+1}) - F^* \leq D_1\|x_k - x^*\|^{p+1} + D_2\|x_k - x^*\|^{q+1}.$$

Assume $q \leq p$. Since the sequence $(x_k)_{k\geq 1}$ is bounded, then we further get:

$$F(x_{k+1}) - F^* \leq (D_1 D^{p-q} + D_2)\|x_k - x^*\|^{q+1}.$$

Combining this inequality with (4.19), we get:

$$F(x_{k+1}) - F^* \leq \frac{(D_1 D^{p-q} + D_2)\theta^{\frac{q+1}{\theta}}}{\sigma^{\frac{q+1}{\theta}}}(F(x_k) - f^*)^{\frac{q+1}{\theta}}.$$

If $p \leq q$, then we also get:

$$F(x_{k+1}) - F^* \leq \frac{(D_1 + D_2 D^{q-p})\theta^{\frac{p+1}{\theta}}}{\sigma^{\frac{p+1}{\theta}}}(F(x_k) - f^*)^{\frac{p+1}{\theta}},$$

which proves the second statement of the theorem. ∎

Remark 8. In [22] the authors propose an algorithm, different from MTA, for solving problem (4.1) based on lower approximations of the functions F_i, for $i = 0 : m$. Moreover, in [22] both the objective and the constraints functions F_i's are assumed uniformly convex. Under these settings, [22] derives linear convergence in function values for their algorithm. Our result is less conservative since we require only the objective function to be uniformly convex and the functional constraints are assumed convex. Note also that if $p = q = 1$ and $h = 0$ we recover the convergence rate in [24].

4.5 Efficient solution of subproblem for (non)convex problems

For the case where $p = q = 1$, it has been demonstrated that subproblem (4.4) can be efficiently solved, as indicated in [24]. In this section, we will show that subproblem (4.4) can also be solved efficiently when $p = q = 2$ or when $p = 2$ and $q = 1$, by utilizing efficient convex optimization techniques. To achieve this, consider the scenario where $p = q = 2$ and $h = 0$. To compute x_{k+1} in subproblem (4.4), one needs to solve the following problem (here $M_0 = M_p + M$):

$$\begin{aligned}
&\min_{x\in\mathbb{R}^n} F_0(x_k) + \langle \nabla F_0(x_k), x - x_k\rangle + \frac{1}{2}\left\langle \nabla^2 F_0(x_k)(x - x_k), (x - x_k)\right\rangle + \frac{M_0}{6}\|x - x_k\|^3 \qquad (4.20)\\
&\text{s.t.}: F_i(x_k)+\langle \nabla F_i(x_k), x - x_k\rangle+\frac{1}{2}\left\langle \nabla^2 F_i(x_k)(x - x_k), (x - x_k)\right\rangle + \frac{M_i}{6}\|x - x_k\|^3 \leq 0 \;\; i = 1\!:\!m.
\end{aligned}$$

Denote $u = (u_0, u_1, \cdots, u_m)$. Then, this problem is equivalent to:

$$\begin{aligned}
\min_{x\in\mathbb{R}^n} \max_{\substack{u\in\mathbb{R}^{m+1}_+\\ u_0=1}} & \sum_{i=0}^m u_i F_i(x_k) + \left\langle \sum_{i=0}^m u_i \nabla F_i(x_k), x - x_k \right\rangle \\
& + \frac{1}{2}\left\langle \sum_{i=0}^m u_i \nabla^2 F_i(x_k)(x - x_k), (x - x_k)\right\rangle + \frac{\sum_{i=0}^m u_i M_i}{6}\|x - x_k\|^3.
\end{aligned}$$

Further, we get:

$$\begin{aligned}
\min_{x\in\mathbb{R}^n} \max_{\substack{u\in\mathbb{R}^{m+1}_+\\ u_0=1}} & \sum_{i=0}^m u_i F_i(x_k) + \left\langle \sum_{i=0}^m u_i \nabla F_i(x_k), x - x_k \right\rangle \\
+ \frac{1}{2}&\left\langle \sum_{i=0}^m u_i \nabla^2 F_i(x_k)(x - x_k), (x - x_k)\right\rangle + \max_{w\geq 0}\left(\frac{w}{4}\|x - x_k\|^2 - \frac{1}{12(\sum_{i=0}^m u_i M_i)^2} w^3\right).
\end{aligned}$$

For simplicity, we denote $H(u,w) = \sum_{i=0}^m u_i \nabla^2 F_i(x_k) + \frac{w}{2} I$, $g(u) = \sum_{i=0}^m u_i \nabla F_i(x_k)$, $l(u) = \sum_{i=0}^m u_i F_i(x_k)$ and $\tilde{M}(u) = \sum_{i=0}^m u_i M_i$. Then, the dual formulation of this problem:

$$\min_{x\in\mathbb{R}^n} \max_{\substack{(u,w)\in\mathbb{R}^{m+2}_+\\ u_0=1}} l(u) + \langle g(u), x - x_k\rangle + \frac{1}{2}\langle H(u,w)(x - x_k), (x - x_k)\rangle - \frac{w^3}{12(\sum_{i=0}^m u_i M_i)^2}.$$

Consider the following notations:

$$\begin{aligned}
&\theta(x,u) = l(u) + \langle g(u), x - x_k\rangle + \frac{1}{2}\left\langle\left(\sum_{i=0}^m u_i \nabla^2 F_i(x_k)\right)(x - x_k), x - x_k\right\rangle + \frac{\tilde{M}(u)}{6}\|x - x_k\|^3,\\
&\beta(u,w) = l(u) - \frac{1}{2}\left\langle H(u,w)^{-1} g(u), g(u)\right\rangle - \frac{1}{12(\tilde{M}(u))^2} w^3,
\end{aligned}$$

$$\mathcal{F}_k = \left\{ (u_0, u_1, \cdots, u_m, w) \in \mathbb{R}^{m+2}_+ : \; u_0 = 1 \text{ and } \sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) + \frac{w}{2} I \succ 0 \right\}.$$

Then, we have the following theorem:

Theorem 4.5.1. *If there exists an $M_i > 0$, for some $i = 0 : m$, then we have the following relation:*

$$\theta^* := \min_{x \in \mathbb{R}^n} \max_{u \geq 0} \theta(x, u) = \max_{(u,w) \in \mathcal{F}_k} \beta(u, w) = \beta^*.$$

Moreover, for any $(u, w) \in \mathcal{F}_k$ the direction $x(u, w) = x_k - H(u, w)^{-1} g(u)$ satisfies:

$$0 \leq \theta(x(u, w), u) - \beta(u, w) = \frac{\tilde{M}(u)}{12} \left(\frac{w}{\tilde{M}(u)} + 2r_k \right) \left(r_k - \frac{w}{\tilde{M}(u)} \right)^2, \tag{4.21}$$

where $r_k = \|x(u, w) - x_k\|$.

Proof. First, we show that $\theta^* \geq \beta^*$. Indeed, using a similar reasoning as in [10], we have:

$$\begin{aligned}
\theta^* &= \min_{x \in \mathbb{R}^n} \max_{\substack{(u,w) \in \mathbb{R}^{m+2}_+ \\ u_0 = 1}} l(u) + \langle g(u), x - x_k \rangle + \frac{1}{2} \langle H(u, w)(x - x_k), (x - x_k) \rangle - \frac{w^3}{12 \tilde{M}(u)^2} \\
&\geq \max_{\substack{(u,w) \in \mathbb{R}^{m+2}_+ \\ u_0 = 1}} \min_{x \in \mathbb{R}^n} l(u) + \langle g(u), x - x_k \rangle + \frac{1}{2} \langle H(u, w)(x - x_k), (x - x_k) \rangle - \frac{w^3}{12 \tilde{M}(u)^2} \\
&\geq \max_{(u,w) \in \mathcal{F}_k} \min_{x \in \mathbb{R}^n} l(u) + \langle g(u), x - x_k \rangle + \frac{1}{2} \langle H(u, w)(x - x_k), (x - x_k) \rangle - \frac{w^3}{12 \tilde{M}(u)^2} \\
&= \max_{(u,w) \in \mathcal{F}_k} l(u) - \frac{1}{2} \langle H(u, w)^{-1} g(u), g(u) \rangle - \frac{1}{12 (\sum_{i=0}^{m} u_i M_i)^2} w^3 = \beta^*.
\end{aligned}$$

Let $(u, w) \in \mathcal{F}_k$. Then, we have $g(u) = -H(u, w)(x(u, w) - x_k)$ and thus:

$$\begin{aligned}
\theta(x(u, w), u) &= l(u) + \langle g(u), x(u, w) - x_k \rangle + \frac{\tilde{M}(u)}{6} r_k^3 \\
&\quad + \frac{1}{2} \left\langle \left(\sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) \right) (x(u, w) - x_k), x(u, w) - x_k \right\rangle \\
&= l(u) - \langle H(u, w)(x(u, w) - x), x(u, w) - x \rangle + \frac{\tilde{M}(u)}{6} r_k^3 \\
&\quad + \frac{1}{2} \left\langle \left(\sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) \right) (x(u, w) - x_k), x(u, w) - x_k \right\rangle \\
&= l(u) - \frac{1}{2} \left\langle \left(\sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) + \frac{w}{2} I \right) (x(u, w) - x_k), x(u, w) - x_k \right\rangle - \frac{w}{4} r_k^2 + \frac{\tilde{M}(u)}{6} r_k^3 \\
&= \beta(u, w) + \frac{1}{12 \tilde{M}(u)^2} w^3 - \frac{w}{4} r_k^2 + \frac{\tilde{M}(u)}{6} r_k^3 \\
&= \beta(u, w) + \frac{\tilde{M}(u)}{12} \left(\frac{w}{\tilde{M}(u)} \right)^3 - \frac{\tilde{M}(u)}{4} \left(\frac{w}{\tilde{M}(u)} \right) r_k^2 + \frac{\tilde{M}(u)}{6} r_k^3 \\
&= \beta(u, w) + \frac{\tilde{M}(u)}{12} \left(\frac{w}{\tilde{M}(u)} + 2r_k \right) \left(r_k - \frac{w}{\tilde{M}(u)} \right)^2,
\end{aligned}$$

which proves (4.21). Note that we have:

$$\nabla_w \beta(u,w) = \frac{1}{4}\|x(u,w) - x_k\|^2 - \frac{1}{4\tilde{M}(u)^2}w^2 = \frac{1}{4}\left(r_k + \frac{w}{\tilde{M}(u)}\right)\left(r_k - \frac{w}{\tilde{M}(u)}\right).$$

Therefore if β^* is attained at some $(u^*, w^*) > 0$ from $\mathcal{F}_k$, then we have $\nabla\beta(u^*, w^*) = 0$. This implies $\frac{w^*}{\tilde{M}(u^*)} = r_k(u^*, w^*)$ and by (4.21) we conclude that $\theta^* = \beta^*$. ■

Remark 9. Note that in nondegenerate situations *the global minimum* of nonconvex cubic problem over nonconvex cubic constraints (4.20) can be computed by:

$$x_{k+1} = x_k - H(u,w)^{-1}g(u),$$

where recall that $H(u,w) = \sum_{i=0}^m u_i \nabla^2 F_i(x_k) + \frac{w}{2}I$, $g(u) = \sum_{i=0}^m u_i \nabla F_i(x_k)$ and $l(u) = \sum_{i=0}^m u_i F_i(x_k)$, with (u,w) the solution of the following dual problem:

$$\max_{(u,w)\in\mathcal{F}_k} \; l(u) - \frac{1}{2}\left\langle H(u,w)^{-1}g(u), g(u)\right\rangle - \frac{1}{12(\sum_{i=0}^m u_i M_i)^2}w^3, \tag{4.22}$$

i.e., a maximization of a concave function over a convex set $\mathcal{F}_k$. Hence, if m is not too large, this dual problem can be solved very efficiently by interior point methods [66].

Corollary 4.5.2. *If there exist $M_i > 0$, then the set $\mathcal{F}_k$ is nonempty and convex. If the problem (4.22) has solution, then strong duality holds for the subproblem (4.20).*

In conclusion, MTA algorithm can be implementable for $p = q = 2$ even for nonconvex problems, since we can effectively compute the global minimum x_{k+1} of subproblem (4.4) using the powerful tools from convex optimization. Note that a similar analysis can be derived for $p = 2$ and $q = 1$. Next, we show the efficiency of the MTA algorithm numerically and compare it with existing methods from literature.

4.6 Experimental results

In this section we present numerical experiments illustrating the performance of MTA algorithm and compare it with existing algorithms from the literature. We consider an optimization problem that is based on the convex function $x \mapsto \log\left(1 + \exp(a_0^T x)\right)$ having gradient Lipschitz with constant $\|a_0\|^2$ and hessian Lipschitz with constant $2\|a_0\|^3$, and on the nonconvex function $x \mapsto \log\left(\frac{(c^T x + e)^2}{2} + 1\right)$ having also gradient Lipschitz with constant $2\|c\|^2$ and hessian Lipschitz with constant $4\|c\|^3$. These functions appear frequently in machine learning applications [99]. In order to make our problem nontrivial and highly nonconvex, we add quadratic regularizers, i.e., we consider the problem:

$$\begin{aligned} &\min_x F(x) = \log\left(1 + \exp(a_0^T x)\right) + \frac{1}{2}x^T Q_0 x + c_0^T x + d_0 \\ &\text{s.t.}: \; F_i(x) = \log\left(\frac{(a_i^T x + b_i)^2}{2} + 1\right) + \frac{1}{2}x^T Q_i x + c_i^T x + d_i \leq 0, \; i = 1:m. \end{aligned} \tag{4.23}$$

We generate the data a_i, b_i, Q_i, c_i, d_i, for $i = 0:m$, randomly, where Q_i's are symmetric indefinite matrices such that the problem is strictly feasible (this is ensured by the choice $d_i < \log\left(\frac{b_i^2}{2}\right)$ for $i = 1:m$). Hence, the problem (4.23) is nonconvex, i.e., both the objective and the constraints are nonconvex functions. Our numerical simulation are performed as follows: for given problem data and an initial feasible point x_0 we compute an approximate F^* and x^* solution of (4.23) using IPOPT [70]; then, we implement our algorithm MTA(2,1) (i.e., $p = 2, q = 1$), MTA(2,2) (i.e., $p = q = 2$), with the regularization parameters chosen to satisfy $M_p > L_p$ and $M_q^i > L_q^i$ for all $i = 1 : m$, and compare with SCP [100] and MBA algorithm proposed in [24]. Note that MBA coincides with MTA for $p = q = 1$. The stopping criterion are: $F(x_k) - F^* \leq 10^{-3}$ and $\max_{i=1:m}(0, F_i(x_k)) \leq 10^{-3}$ and each subproblem is solved using IPOPT (i.e., for MTA(2,1) and MTA(2,2) we use IPOPT to solve the corresponding dual subproblem (4.22) at each iteration). The results are given in Table 4.1 for different choices of the problem dimension (n) and number of constraints (m). In the table we report the cpu time and number of iterations for each method. From our numerical simulations, one can observe that our MTA(2,1) performs better than MBA and SCP methods, although the theoretical convergence rates are the same for all three methods (see [24, 25]). Moreover, increasing p and q, e.g., algorithm MTA(2,2), leads to even much better performance for our algorithm, i.e., MTA(2,2) is superior to MTA(2,1), MBA and SCP, which is expected from our convergence theory. Figure 1 also shows that increasing the approximation orders p and q is beneficial in our MTA algorithm, leading to better performance than first order methods (e.g. MBA and SCP) or than MTA(2,1).

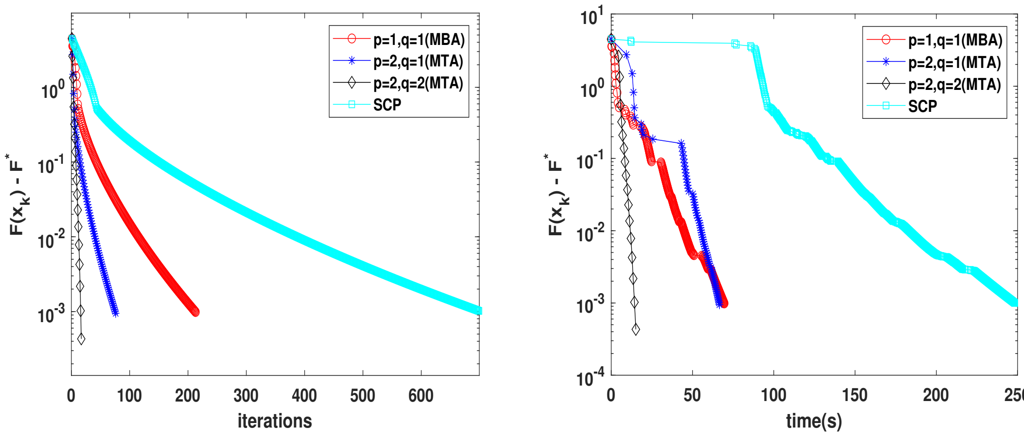

Figure 4.1: Behaviour of residual function for MTA with $q = 2, p = 1$ and with $q = p = 2$, MBA and SCP along iterations (left) and time in sec (right): $n = 100, m = 10$.

		SCP		MBA[24] (MTA(1,1))		MTA(2,1)		MTA(2,2)	
n	m	cpu	iter	cpu	iter	cpu	iter	cpu	iter
10	10	16.3	565	9.2	230	5.5	131	1.3	20
	20	24.5	458	9.8	149	4.2	64	2.5	23
	50	81.5	401	52.5	184	30.9	118	8.5	18
	100	428.7	818	145.6	246	55.2	90	25	19
	500	1119	146	534.7	50	252.8	27	135.9	9
	10^3	$1.1{\cdot}10^4$	394	$6.7{\cdot}10^3$	188	$4.2{\cdot}10^3$	103	499.3	10
20	10	149.8	2477	40.7	700	25	243	13.5	84
	20	396.5	4134	166.9	1253	69.1	494	25.3	122
	50	544.5	1288	266.7	580	151.6	331	37.5	37
	100	862.4	551	441	243	264.2	148	72.3	23
	500	$1.9{\cdot}10^4$	1078	10^4	454	5052	245	767.2	26
100	10	247	696	69.3	211	66	76	15.5	17
	20	6159	1179	713.3	306	350	79	252	28
	50	$2.3{\cdot}10^4$	1460	5974	235	1026	25	711.3	18
	100	$5.6{\cdot}10^4$	5611	$1.1 \cdot 10^4$	1138	2055	89	1252	40
	500	$1.2 \cdot 10^5$	1384	$3{\cdot}10^4$	338	6325	67	2155	20

Table 4.1: Comparison between MTA with $q = 2, p = 1$ and with $q = p = 2$, MBA and SCP in terms of iterations and cpu time (sec) for different values m and n.

4.7 Conclusions

In this chapter, we have proposed a higher-order algorithm for solving composite problems with smooth functional constraints, called MTA. Our method uses higher-order derivatives to build a model that approximates the objective and the functional constraints. We have proven global convergence guarantees in both nonconvex and convex cases. We have also shown that our algorithm MTA is implementable and efficient in numerical simulations.

5 Regularized higher-order Taylor approximation methods for nonlinear least-squares

In this chapter, we introduce a higher-order algorithm specifically designed to minimize composite nonlinear least-squares problems. We establish global convergence guarantees for this method when applied to composite problems with nonconvex and nonsmooth objective functions, with improved convergence rates under the Kurdyka-Łojasiewicz (KL) property. Furthermore, we extend the scope of our investigation to include the behavior and efficacy of our algorithm in handling systems of nonlinear equations and optimization problems with nonlinear equality constraints and derive convergence rates specific for each class of problems. We also provide an efficient implementation of the proposed method.

The chapter is structured as follows: Section 5.1 provides a comprehensive literature review on methods for nonlinear least-squares problems. In Section 5.2, we introduce our general composite higher-order framework and the associated algorithm. We derive global and local convergence results for this approach in both convex and nonconvex scenarios. Additionally, we present an adaptive scheme that does not require prior knowledge of the Lipschitz constants. Section 5.4 presents an efficient implementation of the proposed algorithm. The chapter concludes with applications of our algorithm to particular classes of nonlinear programming in Section 5.5. The content presented in this chapter is derived from the paper [27].

5.1 State of the art

In the field of numerical analysis and optimization, solving nonlinear systems has long been a fundamental aspect of various engineering and scientific applications. The task of finding a solution $x \in \mathbb{R}^n$ to a nonlinear system of equations, typically represented as:

$$F(x) = 0, \tag{5.1}$$

where $F(x) = (F_1(x), \ldots, F_m(x))$ with F_i's being a differentiable functions, holds significant importance for a wide range of real-world problems, from engineering design to data analysis. One powerful method for tackling this problem when $n = m$ is the Newton's method. In this method, the next iteration, x_{k+1}, is calculated by solving the following linear system:

$$\nabla F(x_k) \cdot (x_{k+1} - x_k) = -F(x_k),$$

where x_k is the current iteration. When the iteration x_k is close to a non-degenerate root x^*, i.e., $F(x^*) = 0$ and $\nabla F(x^*)$ has full rank, Newton's method converges superlinearly, as noted in Theorem 11.2 of [49]. However, if the initial guess is far from the root, this method may not converge. Moreover, if the Jacobian $\nabla F(x_k)$ is singular, the Newton step may not be well-defined, potentially leading to undefined or unstable behavior. Further, the previous system of equations can be formulated into the following optimization problem:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|F(x)\|^2, \tag{5.2}$$

which we refers to as the quadratic least-squares problem. Among the established methods for tackling this problem, the most well-known method is the Gauss-Newton algorithm [49, 101], a popular scheme that leverages an iterative process to find solutions efficiently. Namely, the iterations are of the form:

$$\begin{aligned} x_{k+1} &= \arg\min_x \frac{1}{2}\|F(x_k) + \nabla F(x_k)(x - x_k)\|^2, \\ &\iff \nabla F(x_k)^T \nabla F(x_k)(x_{k+1} - x_k) = -\nabla F(x_k)^T F(x_k). \end{aligned}$$

While Newton's method is designed to find zeros of a function, Gauss-Newton naturally suits problems where we are minimizing the squared norm of the residuals. This structure makes it inherently stable for these types of problems. However, like Newton's method, Gauss-Newton method can become unstable or fail to converge when dealing with ill-conditioned Jacobians. The Levenberg-Marquardt method extends the Gauss-Newton method by introducing a regularization term, providing greater stability and flexibility when faced with ill-conditioned Jacobians or challenging convergence scenarios [102, 103, 104]. The iterations in the Levenberg-Marquardt method are defined as follows:

$$\begin{aligned} x_{k+1} &= \arg\min_x \frac{1}{2}\|F(x_k) + \nabla F(x_k)(x - x_k)\|^2 + \frac{\gamma}{2}\|x - x_k\|^2, \\ &\iff \left(\nabla F(x_k)^T \nabla F(x_k) + \gamma I_n\right)(x_{k+1} - x_k) = -\nabla F(x_k)^T F(x_k), \end{aligned}$$

where x_k represents the current iteration, I_n is the identity matrix, and γ is a regularization parameter. When $\gamma = 0$, the Levenberg-Marquardt method coincides with the Gauss-Newton method. Unlike the Gauss-Newton method, the Levenberg-Marquardt method has a global convergence rate to a stationary point of the objective $\|F(x)\|^2$, for more details, see Theorem 10.3 in [49]. Additionally, the Levenberg-Marquardt method has a similar local convergence behavior to the Gauss-Newton method when the initial point is near the solution.

In [94], Nesterov proposed a novel approach to solve the systems of non-linear equations (5.1), which bears a resemblance to the traditional Gauss-Newton method. The core concept is to convert the original problem (5.1) into an optimization problem with a non-smooth merit function. One common choice for this transformation is to use the 2-norm, leading to the following non-smooth nonlinear least-squares problem:

$$\min_{x \in \mathbb{R}^n} \|F(x)\|. \tag{5.3}$$

This approach differs from the smooth problem formulation in (5.2), as the optimization problem in (5.3) is inherently non-smooth. However, if the system of equations is linear, the transformation defined in (5.2) effectively squares the condition number of the problem, which may affect the convergence speed. Due to this drawback, in this chapter, we focus on a composite problem with a non-smooth merit function. More precisely, we consider the following composite optimization problem:

$$\min_{x \in \mathbb{R}^n} f(x) := g(F(x)) + h(x), \tag{5.4}$$

where F represents a real-vector function, defined as $F = (F_1, \ldots, F_m)$. We assume that each function $F_i : \mathbb{R}^n \to \mathbb{R}$ is $p \geq 1$ times differentiable and has the pth derivative Lipschitz continuous, the function $g : \mathbb{R}^m \to \mathbb{R}$ is nonsmooth, convex, and Lipschitz continuous (e.g., g is the 2-norm), and the function $h : \mathbb{R}^n \to \bar{\mathbb{R}}$ is proper, lower semicontinuous, and convex. Note that we have $\operatorname{dom} f = \operatorname{dom} h$. This formulation covers many problems from the nonlinear programming literature and appears in many real-world applications such as control, statistical estimation, grey-box minimization, machine learning, and phase retrieval [56, 33, 30, 49, 31].

5.1.1 First-order methods

A natural approach for solving problem (5.4), is one that resembles the Gauss-Newton approach, consists in linearizing the smooth part, F, and adding an appropriate quadratic regularization. More precisely, to obtain the next iteration, one needs to solve the following subproblem for a given current iteration x_k:

$$x_{k+1} = \arg\min_x g\Big(F(x_k) + \nabla F(x_k)(x - x_k)\Big) + \frac{L}{2}\|x - x_k\|^2 + h(x).$$

This scheme, known as (proximal) Gauss-Newton method, has been well-studied in the literature [105, 106, 94, 56, 35]. When analyzing convergence for nonconvex problems, which can also be nonsmooth, a common informal proof strategy involves the following two steps [52, 107, 98]:

- (i) *Sufficient decrease*: Find a positive constant $M_1 > 0$ such that

  $$M_1\|x_{k+1} - x_k\|^2 \leq f(x_k) - f(x_{k+1}).$$

  This condition ensures that the objective function decreases sufficiently as the iterations progress.

- (ii) *Subgradient lower bound*: Find a positive constant $M_2 > 0$ such that

  $$\mathrm{dist}(0, \partial f(x_{k+1})) \leq M_2\|x_{k+1} - x_k\|.$$

  This step provides a lower bound on the distance between the zero vector and the subgradient of f at x_{k+1}, indicating the gap between successive iterations.

Thus, if $\|x_{k+1} - x_k\| \to 0$, then $\mathrm{dist}(0, \partial f(x_{k+1})) \to 0$, indicating that any limit point of the sequence $(x_k)_{k\geq 0}$ is a critical point. However, if f is in the form of equation (5.4), condition (ii) might be impossible to meet, as demonstrated in the following example taken from [56], i.e., if g is nonsmooth, the quantity $\mathrm{dist}(0, \partial f(x_{k+1}))$ will typically not even tend to zero in the limit, in spite of $\|x_{k+1} - x_k\|$ tending to zero. More specifically, let us consider the (proximal) Gauss-Newton applied to

$$\min_x |x^2 - 1|,$$

with an initialization satisfying $x_0 > 1$. This will generate a decreasing sequence that converges to 1 and thus $\|x_{k+1} - x_k\| \to 0$. However, for any $x_k > 1$ we have $f^{'}(x_k) = 2x_k \to 2$. In order to overcome this obstacle, the authors in [56] introduce an artificial sequence, $(y_k)_{k\geq 0}$, that is close to the original sequence $(x_k)_{k\geq 0}$. This proximity is defined as follows:

$$\begin{aligned}&\|y_{k+1} - x_k\| \leq M_1\|x_{k+1} - x_k\|,\\ &\mathrm{dist}(0, \partial f(y_{k+1})) \leq M_2\|y_{k+1} - x_k\|,\end{aligned}$$

where M_1 and M_2 are positive constants. This construction aims to circumvent the limitations associated with condition (ii) by ensuring that the artificial sequence $(y_k)_{k\geq 0}$ retains sufficient proximity to the original sequence $(x_k)_{k\geq 0}$, while still allowing a measurable relationship with the subgradient of f. Thus, under the Lipschitz continuity of the Jacobian, ∇F, the iterates of (proximal) Gauss-Newton method converges to a near stationary point at a sublinear rate of order $\mathcal{O}(k^{-\frac{1}{2}})$ [56], while convergence rates under the Kurdyka-Lojasiewicz (KL) property were recently derived in [65, 35]. Trust region based Gauss-Newton methods have been also considered in [108] for solving problems of the form (5.4). The authors in [108] show that their proposed algorithms take at most $\mathcal{O}(\epsilon^{-2})$ function evaluations to reduce the size of a first-order criticality measure below a given accuracy ϵ.

5.1.2 Higher-order methods

To our knowledge, there are very few studies considering the utilization of higher-order derivatives to address problems of the form (5.4) where the function g is both convex and Lipschitz continuous. For example, in [109], the authors explore the scenario where $g(\cdot) = \frac{1}{2}\|\cdot\|^2$, $h(\cdot) = 0$, and the next iterate, given the current pointy x_k, is the minimizer of the following subproblem:

$$x_{k+1} = \arg\min_x \, (x - x_k)^T J_F(x_k)^T F(x_k) + \frac{1}{2}(x - x_k)^T B(x_k)(x - x_k) + \frac{\bar{\sigma}}{3}\|x - x_k\|^3,$$

where $\bar{\sigma} > 0$, $J_F(x_k)$ is the Jacobian of F at x_k and $B(x_k)$ is an approximation of the true hessian of the function $\frac{1}{2}\|F(x)\|^2$ at x_k. When the residuals F_i, the Jacobian ∇F and the Hessian $\nabla^2 F_i$ for each $i \in \{1, \cdots, m\}$ are simultaneuosly Lipschitz continuous on a neighborhood of $\bar{x}$, [109] shows that this scheme takes at most $\mathcal{O}\left(\epsilon^{-\frac{3}{2}}\right)$ residual and Jacobian evaluations to drive either the Euclidean norm of the residual or its gradient below ϵ. Further, in [69] a similar approach is adopted, with $g(\cdot) = \frac{1}{2}\|\cdot\|^2$ and $h(\cdot) = 0$, by constructing a quadratic approximation of F alongside an appropriate regularization, i.e., given the current point x_k, the next iterate is the minimizer of the following r-regularized subproblem:

$$x_{k+1} = \arg\min_x \frac{1}{2}\left\|F(x_k) + \nabla F(x_k)(x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 F(x_k)(x - x_k)\right\|^2 + \frac{1}{r}\|x - x_k\|^r,$$

where $r \geq 2$ is a given constant. Paper [69] establishes convergence to an ϵ first-order stationary point of the objective within $\mathcal{O}\left(\epsilon^{-\min\left(\frac{r}{r-1}, \frac{3}{2}\right)}\right)$ iterations, provided that the residuals F_i, the Jacobian ∇F and the Hessian $\nabla^2 F_i$ for each $i \in \{1, \cdots, m\}$ are Lipschitz continuous on a neighborhood of a stationary point. It's important to note that the aforementioned subproblem is at least quartic and thus hard to solve. Therefore, using the norm $\|\cdot\|$ as the merit function instead of $\|\cdot\|^2$ is more beneficial, since in the later case the condition number usually doubles, although the objective function for the first choice is nondifferentiable [94]. This strategy has been explored in [93], wherein the authors introduce an adaptive higher-order trust-region algorithm for solving problem (5.4) with h smooth, where F and h are approximated with higher-order Taylor expansions. Paper [93] establishes convergence of order $\mathcal{O}\left(\epsilon^{-\frac{p+1}{p}}\right)$ to achieve a reduction in a given criticality measure below a prescribed accuracy ϵ.

Note that, the optimization problem, the algorithm, and consequently the convergence analysis in [93] are different from the present work. Moreover, it remains open whether one can solve the corresponding subproblem in [93] efficiently for $p \geq 2$, along with establishing convergence rates under the Kurdyka-Lojasiewic (KL) property.

5.2 Regularized higher-order Taylor approximation (RHOTA) method

In this section, we present a regularized higher-order Taylor approximation algorithm for solving composite problem (5.4). We consider the following assumptions:

Assumption 5.2.1. The following statements hold for optimization problem (5.4):

1. For $F = (F_1, \cdots, F_m)$, each component F_i is p times differentiable function with the pth derivative Lipschitz continuous with constant L_p^i.

2. Function g is convex, Lipschitz continuous with constant L_g and h is proper lower semi-continuous and simple convex function.

3. Problem (5.4) has a solution and hence $\inf_{x \in \mathrm{dom}\, f} f(x) \geq f^*$.

From Assumption 5.2.1 and the inequality (2.4), we get for all $i = 1 : m$:

$$\left|F_i(x) - T_p^{F_i}(x;y)\right| \leq \frac{L_p^i}{(p+1)!}\|y-x\|^{p+1} \quad \forall x, y \in \mathbb{R}^n. \tag{5.5}$$

Further, using that the function g is Lipschitz continuous, we get the following inequality valid for all $x, y \in \mathbb{R}^n$:

$$\left|g(F(x)) - g\left(T_p^F(x;y)\right)\right| \leq L_g \left\|F(x) - T_p^F(x;y)\right\| \leq \frac{L_g\|L_p\|}{(p+1)!}\|x-y\|^{p+1}, \tag{5.6}$$

where $T_p^F(x;y) = \left(T_p^{F_1}(x;y), \cdots, T_p^{F_m}(x;y)\right)$ and $L_p = \left(L_p^1, \cdots, L_p^m\right)$. Then, based on this upper bound approximation of the objective function, one can consider an iterative process, where given the current iterate, $\bar{x}$, and a proper regularization parameter $M > 0$, the next point is computed from the following subproblem:

$$x \leftarrow \arg\min_{y \in \mathbb{R}^n} s_M(y;\bar{x}) := g\left(T_p^F(y;\bar{x})\right) + \frac{M}{(p+1)!}\|y-\bar{x}\|^{p+1} + h(y). \tag{5.7}$$

Note that if $x = \bar{x}$ in the previous subproblem, then x is a stationary point of the original problem (5.4). Note also that for $p = 1$, this algorithm reduces to the regularized Gauss-Newton method analyzed in [94, 56, 35]. Now we are ready to present our regularized higher-order Taylor approximation method, called RHOTA (see Algorithm 4). Note that usually the subproblem

Algorithm 4 RHOTA

Given x_0 and $M > 0$. For $k \geq 0$ do:
compute x_{k+1} *inexact* solution of subproblem (5.7) satisfying the following *descent*:

$$s_M(x_{k+1};x_k) \leq s_M(x_k;x_k). \tag{5.8}$$

(5.7) is nonconvex for any $p \geq 2$. In order to get descent for the sequence $(f(x_k))_{k \geq 0}$, it is enough to assume that x_{k+1} satisfies the descent (5.8). However, to derive convergence rates to a stationary point or in function values (under the KL property), we need to require additionally properties for x_{k+1}, e.g., x_{k+1} generated by algorithm must satisfy an inexact (local) optimality condition, see (5.13) in Theorem 5.3.2, i.e., computing a minimizer of the Taylor based model $s_M(\cdot;x_k)$ within an Euclidean ball. We show in Section 5.4 that one can still use the powerful tools from convex optimization to solve the *nonconvex* subproblem (5.7) globally for some particular choices of $p > 1$. More precisely, when the outer function g is the norm and the Taylor approximation is of order $p = 2$, we show that the corresponding subproblem can be solved globally by efficient convex algorithms.

5.3 Convergence analysis of RHOTA

In this section, we analyze the convergence behavior of RHOTA algorithm under different assumptions for problem (5.4), i.e., when Assumption 5.2.1 holds and when, additionally, the objective function satisfies the KL. First, let us prove that the sequence $(f(x_k))_{k\geq 0}$ is a nonincreasing sequence.

Theorem 5.3.1. *Let Assumption 5.2.1 hold and let $(x_k)_{k\geq 0}$ be generated by RHOTA with $M - L_g\|L_p\| > 0$. Then, we have:*

1. *The sequence $(f(x_k))_{k\geq 0}$ is nonincreasing and satisfies:*

$$f(x_{k+1}) \leq f(x_k) - \frac{M - L_g\|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}. \tag{5.9}$$

2. *The sequence $(x_k)_{k\geq 0}$ satisfies:*

$$\sum_{i=1}^{\infty} \|x_{k+1} - x_k\|^{p+1} < \infty, \quad \lim_{k\to\infty} \|x_{k+1} - x_k\| = 0 \;\text{ and }\; \min_{j=0:k} \|x_{j+1} - x_j\|^{p+1} \leq \mathcal{O}\left(\frac{1}{k}\right).$$

Proof. From inequality (5.6), we get:

$$-\frac{L_g\|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + g(F(x_{k+1})) \leq g\left(T_p^F(x_{k+1}; x_k)\right).$$

Further, using the descent (5.8), we also get:

$$\begin{aligned}
&\frac{M - L_g\|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + f(x_{k+1}) \\
&\qquad\leq g\left(T_p^F(x_{k+1}; x_k)\right) + h(x_{k+1}) + \frac{M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\qquad= s_M(x_{k+1}; x_k) \leq s_M(x_k; x_k) = f(x_k).
\end{aligned}$$

Hence, the sequence $(f(x_k))_{k\geq 0}$ is monotonically nonincreasing. Further, summing up the last inequality and using that f is bounded from below by f^*, we get:

$$\sum_{j=0}^{k} \frac{M - L_g\|L_p\|}{(p+1)!}\|x_{j+1} - x_j\|^{p+1} \leq f(x_0) - f(x_k) \leq f(x_0) - f^*.$$

Hence, there exists $\bar{k} \in \{0, \cdots, k\}$ such that:

$$\|x_{\bar{k}+1} - x_{\bar{k}}\|^{p+1} = \min_{j=0:k} \|x_{j+1} - x_j\|^{p+1} \leq \frac{(f(x_0) - f^*)(p+1)!}{(M - L_g\|L_p\|)(k+1)}, \tag{5.10}$$

and then our assertions follow. ■

Remark 10. Theorem 5.3.1 requires $M - L_g\|L_p\| > 0$, where $\|L_p\| = \|(L_p^1, \cdots, L_p^m)\|$. If L_g and $(L_p)_{i=1}^m$ are known, then one can choose $M = L_g\|L_p\| + R_0$ for some $R_0 > 0$. In Section 5.3.3, we propose an adaptive variant of RHOTA that does not require the knowledge of the Lipschitz constants L_g and $(L_p)_{i=1}^m$.

5.3.1 First order convergence

In [56, 23], the authors prove that for a composite problem of the form (5.4), the quantity $\mathrm{dist}(0, \partial f(x_{k+1}))$ doesn't invariably approach zero as $\|x_{k+1} - x_k\|$ tends to zero. Hence, in alignment with the framework outlined in [23], for a given $\mu > 0$ we introduce the (artificial) sequence:

$$y_{k+1} = \arg\min_{y \in \mathbb{R}^n} f(y) + \frac{\mu}{(p+1)!}\|y - x_k\|^{p+1}. \tag{5.11}$$

From the optimality conditions of the iteration y_{k+1}, we get:

$$-\frac{\mu}{p!}\|y_{k+1} - x_k\|^{p-1}(y_{k+1} - x_k) \in \partial f(y_{k+1}).$$

This implies that

$$\mathcal{S}_f(y_{k+1}) = \mathrm{dist}\left(0, \partial f(y_{k+1})\right) \leq \frac{\mu}{p!}\|y_{k+1} - x_k\|^p. \tag{5.12}$$

In the next theorem we establish that the sequence $(y_k)_{k\geq 0}$ is close to the sequence $(x_k)_{k\geq 0}$, both sequences have the same set of limit points, and $(y_k)_{k\geq 0}$ converges towards a stationary point of the original problem with a rate $\mathcal{O}(k^{-\frac{p}{p+1}})$. These results are valid under the condition that the sequence $(x_k)_{k\geq 0}$ generated by RHOTA algorithm satisfies an inexact optimality criterion.

Theorem 5.3.2. *Let the assumptions of Theorem 5.3.1 hold. Let $(x_k)_{k\geq 0}$ be generated by RHOTA algorithm. Let $\mu > M + L_g\|L_p\|$ and y_{k+1} be given in (5.11) and assume x_{k+1} satisfies the following inexact optimality condition for subproblem (5.7):*

$$s_M(x_{k+1}; x_k) - \min_{y:\, \|y - x_k\| \leq D_k} s_M(y; x_k) \leq \frac{\delta}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}, \tag{5.13}$$

where $\delta \geq 0$, $D_k := \left(\frac{(p+1)!}{\mu}(f(x_k) - f^*)\right)^{\frac{1}{p+1}}$ and $s_M(\cdot; x_k)$ is given in (5.7). If we denote $L_\mu = \left(\frac{\mu+\delta+L_g\|L_p\|-M}{\mu-(M+L_g\|L_p\|)}\right)$, then we have:

1. *The sequences $(y_k)_{k\geq 0}$ satisfies $\|y_{k+1} - x_k\|^{p+1} \leq L_\mu\|x_{k+1} - x_k\|^{p+1} \quad \forall k \geq 0$.*
2. *The following convergence rate holds:*

$$\min_{j=0:k} S_f(y_{j+1})^{\frac{p+1}{p}} \leq \frac{L_\mu(f(x_0) - f^*)}{k+1}\left(\frac{\mu}{p!}\right)^{\frac{p+1}{p}} \frac{(p+1)!}{M - L_g\|L_p\|}.$$

Proof. From the definition of y_{k+1}, we have:

$$\begin{aligned}
f(y_{k+1}) + \frac{\mu}{(p+1)!}\|y_{k+1} - x_k\|^{p+1} &\overset{(5.11)}{\leq} f(x_{k+1}) + \frac{\mu}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\overset{(5.6)}{\leq} s_M(x_{k+1}; x_k) + \frac{\mu + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\overset{(5.13)}{\leq} \min_{y:\, \|y - x_k\| \leq D_k} s_M(y; x_k) + \frac{\delta}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} + \frac{\mu + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\overset{(5.6)}{\leq} \min_{y:\, \|y - x_k\| \leq D_k} f(y) + \frac{M + L_g\|L_p\|}{(p+1)!}\|y - x_k\|^{p+1} + \frac{\mu + \delta + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\leq f(y_{k+1}) + \frac{M + L_g\|L_p\|}{(p+1)!}\|y_{k+1} - x_k\|^{p+1} + \frac{\mu + \delta + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1},
\end{aligned}$$

where the last inequality is derived from the observation that $\|y_{k+1} - x_k\| \leq D_k$. Indeed, from the definition of y_{k+1} from (5.11) we have:

$$f(y_{k+1}) + \frac{\mu}{(p+1)!}\|y_{k+1} - x_k\|^{p+1} \leq f(x_k),$$

which implies that:

$$\frac{\mu}{(p+1)!}\|y_{k+1} - x_k\|^{p+1} \leq f(x_k) - f^*.$$

and thus $\|y_{k+1} - x_k\| \leq D_k$. Further, we have:

$$\|y_{k+1} - x_k\|^{p+1} \leq \left(\frac{\mu + \delta + L_g\|L_p\| - M}{\mu - (M + L_g\|L_p\|)}\right)\|x_{k+1} - x_k\|^{p+1} \quad \forall k \geq 0, \tag{5.14}$$

which is the first assertion. It follows immediately from the last inequality that $(y_k)_{k\geq 0}$ and $(x_k)_{k\geq 0}$ have the same set of limit points. Additionally, from (5.10), we have that there exists $\bar{k} \in \{0, \cdots, k\}$ such that:

$$\begin{aligned} \min_{j=0:k} S_f(y_{j+1})^{\frac{p+1}{p}} &\leq S_f(y_{\bar{k}+1})^{\frac{p+1}{p}} \overset{(5.12)}{\leq} \left(\frac{\mu}{p!}\right)^{\frac{p+1}{p}} \|y_{\bar{k}+1} - x_{\bar{k}}\|^{p+1} \\ &\overset{(5.14)}{\leq} L_\mu \left(\frac{\mu}{p!}\right)^{\frac{p+1}{p}} \|x_{\bar{k}+1} - x_{\bar{k}}\|^{p+1} \overset{(5.10)}{\leq} L_\mu \left(\frac{\mu}{p!}\right)^{\frac{p+1}{p}} \frac{(f(x_0) - f^*)(p+1)!}{(M - L_g\|L_p\|)(k+1)}. \end{aligned} \tag{5.15}$$

Hence, our second statement follows. ■

Remark 11. In Theorem 5.3.2, we establish convergence rate guarantees to a near stationary point of order $\mathcal{O}\left(k^{-\frac{p}{p+1}}\right)$, which is the usual convergence rate for higher-order algorithms for (unconstrained) nonconvex p-smooth problems [110, 93, 23]. In our convergence analysis, we additionally assume that the sequence generated by RHOTA algorithm satisfy an inexact optimality condition (5.13), which requires computing a minima over an Euclidean ball. Nevertheless, in Section 5.4 we present an efficient implementation of RHOTA algorithm for the particular case where $g(\cdot) = \|\cdot\|$ and $p = 2$. More precisely, we show that one can still use powerful tools from convex optimization to compute the *global solution of the nonconvex* subproblem (5.7), which automatically satisfies the inexact optimality condition (5.13).

5.3.2 Better convergence under KL

In this section, we establish improved convergence rates for RHOTA algorithm under the KL property, i.e., we prove linear/sublinear convergence in function values for the original sequence $(x_k)_{k\geq 0}$ generated by RHOTA. We denote the set of limit points of $(x_k)_{k\geq 0}$ by $\Omega(x_0)$:

$$\Omega(x_0) = \{\bar{x} \in \mathbb{R}^n : \exists (k_t)_{t\geq 0} \nearrow, \text{ such that } x_{k_t} \to \bar{x} \text{ as } t \to \infty\}.$$

Next lemma derives some properties for $\Omega(x_0)$.

Lemma 5.3.3. *Let the assumptions of Theorem 5.3.2 hold. Additionally, assume that $(x_k)_{k\geq 0}$ is bounded and f is continuous. Then, we have: $\emptyset \neq \Omega(x_0) \subseteq \texttt{stat}\, f$, $\Omega(x_0)$ is compact and connected set, and f is constant on $\Omega(x_0)$, i.e., $f(\Omega(x_0)) = f_*$.*

Proof. Let us prove that $f(\Omega(x_0))$ is constant. From the descent (5.9) we have that $(f(x_k))_{k\geq 0}$ is monotonically decreasing, and since f is assumed to be bounded from below, it converges.

Let us say to $f_* > -\infty$, i.e., $f(x_k) \to f_*$ as $k \to \infty$. On the other hand, let x_* be a limit point of the sequence $(x_k)_{k\geq 0}$. This means that there exists a subsequence $(x_{k_t})_{t\geq 0}$ such that $x_{k_t} \to x_*$. Since f is continuous, we get $f(x_{k_t}) \to f(x_*) = f_*$ and hence, we have $f(\Omega(x_0)) = f_*$. The closeness property of ∂f implies that $S_f(x_*) = 0$, and thus $0 \in \partial f(x_*)$. This proves that x_* is a stationary point of f and thus $\Omega(x_0)$ is nonempty. By observing that $\Omega(x_0)$ can be viewed as an intersection of compact sets, $\Omega(x_0) = \cap_{q\geq 0}\overline{\cup_{k\geq q}\{x_k\}}$ so it is also compact. The connectedness follows from [46]. This completes the proof. ■

Next, we derive improved convergence rates in function values for the sequence $(x_k)_{k\geq 0}$ generated by RHOTA, not for the artificial sequence $(y_k)_{k\geq 0}$ as in Theorem 5.3.2.

Theorem 5.3.4. *Let the assumptions of Lemma 5.3.3 hold. Additionally, assume that f satisfy the KL property (2.16) on $\Omega(x_0)$. Then, the following convergence rates hold for $(x_k)_{k\geq 0}$ generated by RHOTA algorithm for k sufficiently large:*

1. *If $q \geq \frac{p+1}{p}$, then $f(x_k)$ converges to f_* linearly.*
2. *If $q < \frac{p+1}{p}$, then $f(x_k)$ converges to f_* at sublinear rate of order $\mathcal{O}\left(\frac{1}{k^{\frac{pq}{p+1-pq}}}\right)$.*

Proof. We have:

$$
\begin{aligned}
f(x_{k+1}) - f_* &\overset{(5.6)}{\leq} s_M(x_{k+1}; x_k) + \frac{L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} - f_* \\
&\overset{(5.13)}{\leq} \min_{y\colon \|y-x_k\|\leq D_k} s_M(y; x_k) + \frac{\delta + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} - f_* \\
&\overset{(5.6)}{\leq} \min_{y\colon \|y-x_k\|\leq D_k} f(y) - f_* + \frac{M + L_g\|L_p\|}{(p+1)!}\|y - x_k\|^{p+1} + \frac{\delta + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \\
&\leq f(y_{k+1}) - f_* + \left(\frac{M + L_g\|L_p\|}{(p+1)!}\|y_{k+1} - x_k\|^{p+1} + \frac{\delta + L_g\|L_p\| - M}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}\right) \\
&\leq \sigma_q \mathrm{dist}(0, \partial f(y_{k+1}))^q + \left(\frac{M + L_g\|L_p\| + L_\mu(\delta + L_g\|L_p\| - M)}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}\right) \\
&\leq \left(\frac{\sigma_q (L_\mu)^{\frac{pq}{p+1}}\mu^q}{(p!)^q}\|x_{k+1} - x_k\|^{pq} \frac{(1-L_\mu)M + (1+L_\mu)L_g\|L_p\| + L_\mu\delta}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}\right) \\
&\overset{(5.9)}{\leq} C_1\big(f(x_k) - f(x_{k+1})\big)^{\frac{pq}{p+1}} + C_2\big(f(x_k) - f(x_{k+1})\big),
\end{aligned}
$$

where the fourth inequality follows from $\|y_{k+1} - x_k\| \leq D_k$, the fifth inequality is deduced from (2.16) combined with the first assertion of Theorem 5.3.2, (i.e., the sequences $(x_k)_{k\geq 0}$ and $(y_k)_{k\geq 0}$ share the same limit point) and the sixth inequality follows from (5.12) combined with the first assertion of Theorem 5.3.2. Here $C_1 = \frac{\sigma_q (L_\mu)^{\frac{pq}{p+1}}\mu^q}{(p!)^q}\left(\frac{(p+1)!}{M - L_g\|L_p\|}\right)^{\frac{pq}{p+1}}$ and $C_2 = \frac{M + L_g\|L_p\| + L_\mu(\delta + L_g\|L_p\| - M)}{M - L_g\|L_p\|}$. Let us denote $\Delta_k = f(x_k) - f_*$. Subsequently, we derive the following recurrence:

$$
\Delta_{k+1} \leq C_1\left(\Delta_k - \Delta_{k+1}\right)^{\frac{qp}{p+1}} + C_2\left(\Delta_k - \Delta_{k+1}\right).
$$

Using Lemma 2.4.2 with $\theta = \frac{p+1}{pq}$, our assertions follow. ■

Remark 12. In this section, we have derived improved convergence rates in terms of function values for sequence $(x_k)_{k\geq 0}$ generated by RHOTA, not for $(y_k)_{k\geq 0}$, by leveraging higher-order information to solve problem (5.4), and *to our knowledge, these rates represent novel findings for such problems when employing higher-order information*. Notably, for $p = 1$, our results align with the convergence rates in [35].

5.3.3 Adaptive regularized higher-order Taylor approximation method

In RHOTA algorithm, we need to compute a regularization parameter $M > L_g\|L_p\|$. However, in practice, determining Lipschitz constants L_g and L_p may be challenging. Consequently, in this section, we introduce an adaptive regularized higher-order Taylor algorithm (A-RHOTA), which does not require prior knowledge of these constants.

Algorithm 5 A-RHOTA algorithm

Given x_0 and $M_0, R_0 > 0$ and $i, k = 0$.
while some criterion is not satisfied **do**
  1. define $s_{M_k}(y; x_k) := g\left(T_p^F(y; x_k)\right) + \frac{2^i M_k}{(p+1)!}\|y - x_k\|^{p+1} + h(y)$.
  2. compute x_{k+1} inexact solution of $\min_y s_{M_k}(y; x_k)$.
  if descent (5.9) holds, then go to step 3.
    else set $i = i + 1$ and go to step 1.
  end if
  3. set $k = k + 1$, $M_{k+1} = 2^{i-1}M_k$ and $i = 0$.
end while

This line search procedure ensures the decrease (5.9) and finishes in a finite number of steps. Indeed, if $M_k \geq R_0 + L_g\|L_p\|$, then from inequality (5.6), we get:

$$g\left(T_p^F(x_{k+1}; x_k)\right) - f(x_{k+1}) \geq \frac{-L_g\|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}.$$

This implies that:

$$\begin{aligned} f(x_k) - f(x_{k+1}) &= s_{M_k}(x_k; x_k) - f(x_{k+1}) \geq s_{M_k}(x_{k+1}; x_k) - f(x_{k+1}) \\ &\geq \frac{M_k - L_g\|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \geq \frac{R_0}{(p+1)!}\|x_{k+1} - x_k\|^{p+1}. \end{aligned}$$

Note also that we have $M_k \leq 2(R_0 + L_g\|L_p\|)$ for all $k \geq 0$. Consequently, using similar arguments as before allows us to derive convergence rates similar to Theorems 5.3.2 and 5.3.4 for the adaptive regularized higher-order Taylor algorithm (A-RHOTA).

5.4 Efficient solution of the nonconvex subproblem

In this section, we present an efficient implementation of RHOTA algorithm for the case $g(\cdot) = \|\cdot\|$, $p = 2$ and quadratic $h(x) = (1/2)x^T Bx + ax$. Within this context, x_{k+1} is the solution of the following nonconvex subproblem (see the subproblem (5.7)):

$$\begin{aligned} \mathcal{P}^* = \min_{x \in \mathbb{R}^n} &\left\| F(x_k) + \langle \nabla F(x_k), x - x_k \rangle + \frac{1}{2}\nabla^2 F(x_k)[x - x_k]^2 \right\| + \frac{M}{6}\|x - x_k\|^3 \\ &+ \frac{1}{2}(x - x_k)^T B(x - x_k) + \langle a + Bx_k, x - x_k \rangle. \end{aligned} \tag{5.16}$$

with $\langle \nabla F(x_k), x - x_k \rangle = [\langle \nabla F_1(x_k), x - x_k \rangle, \cdots, \langle \nabla F_m(x_k), x - x_k \rangle]$ and $\nabla^2 F(x_k)[x - x_k]^2 = \left[\nabla^2 F_1(x_k)[x - x_k]^2, \cdots, \nabla^2 F_m(x_k)[x - x_k]^2\right]$. Denote $u = (u_1, \cdots, u_m)$. Then, this subproblem is equivalent to:

$$\min_{x\in\mathbb{R}^n}\max_{\|u\|\leq 1}\sum_{i=1}^m u_iF_i(x_k)+\left\langle\sum_{i=1}^m u_i\nabla F_i(x_k)+a+Bx_k, x-x_k\right\rangle$$
$$+\frac{1}{2}\left\langle\left(\sum_{i=1}^m u_i\nabla^2F_i(x_k)+B\right)(x-x_k),(x-x_k)\right\rangle+\frac{M}{6}\|x-x_k\|^3.$$

Further, the last term can be written equivalently as:

$$\frac{M}{6}\|x-x_k\|^3=\max_{w\geq 0}\left(\frac{w}{4}\|x-x_k\|^2-\frac{1}{12M^2}w^3\right).$$

Denote $H_k(u,w)=\sum_{i=1}^m u_i\nabla^2F_i(x_k)+B+\frac{w}{2}I_n$, $G_k(u)=\sum_{i=1}^m u_i\nabla F_i(x_k)+a+Bx_k$ and $l_k(u)=\sum_{i=1}^m u_iF_i(x_k)$. Then, we have:

$$\mathcal{P}^*=\min_{x\in\mathbb{R}^n}\max_{\|u\|\leq 1 w\geq 0} l_k(u)+\langle G_k(u),x-x_k\rangle+\frac{1}{2}\langle H_k(u,w)(x-x_k),x-x_k\rangle-\frac{w^3}{12M^2}.$$

Consider the following notations:

$$\theta_k(x,u)=l_k(u)+\langle G_k(u),x-x_k\rangle+\frac{1}{2}\left\langle\left(\sum_{i=1}^m u_i\nabla^2F_i(x_k)+B\right)(x-x_k),x-x_k\right\rangle+\frac{M}{6}\|x-x_k\|^3$$

$$\beta_k(u,w)=l_k(u)-\frac{1}{2}\left\langle H_k(u,w)^{-1}G_k(u),G_k(u)\right\rangle-\frac{1}{12M^2}w^3,\quad r_k=\|x_{k+1}-x_k\|\ \text{ and}$$

$$\mathcal{F}_k=\left\{(u,w)\in\mathbb{R}^m\times\mathbb{R}_+:\|u\|\leq 1 \text{ and } \sum_{i=1}^m u_i\nabla^2F_i(x_k)+B+\frac{w}{2}I\succ 0\right\}.$$

Then, we have the following theorem:

Theorem 5.4.1. *If $M>0$, then we have the following relation:*

$$\theta^*:=\min_{x\in\mathbb{R}^n}\max_{\|u\|\leq 1}\theta_k(x,u)=\max_{(u,w)\in\mathcal{F}_k}\beta_k(u,w)=\beta^*.$$

For any $(u,w)\in\mathcal{F}_k$ the direction $x_{k+1}=x_k-H_k(u,w)^{-1}G_k(u)$ satisfies:

$$0\leq\theta_k(x_{k+1},u)-\beta_k(u,w)=\frac{M}{12}\left(\frac{w}{M}+2r_k\right)\left(r_k-\frac{w}{M}\right)^2. \tag{5.17}$$

Proof. First, we show $\theta^*\geq\beta^*$. Indeed, using a similar reasoning as [10], we have:

$$\begin{aligned}\theta^*&=\min_{x\in\mathbb{R}^n}\max_{\substack{(u,w)\in\mathbb{R}^m\times\mathbb{R}_+\\ \|u\|\leq 1}} l_k(u)+\langle G_k(u),x-x_k\rangle+\frac{1}{2}\langle H_k(u,w)(x-x_k),x-x_k\rangle-\frac{w^3}{12M^2}\\
&\geq\max_{\substack{(u,w)\in\mathbb{R}^m\times\mathbb{R}_+\\ \|u\|\leq 1}}\min_{x\in\mathbb{R}^n} l_k(u)+\langle G_k(u),x-x_k\rangle+\frac{1}{2}\langle H_k(u,w)(x-x_k),x-x_k\rangle-\frac{w^3}{12M^2}\\
&\geq\max_{(u,w)\in\mathcal{F}_k}\min_{x\in\mathbb{R}^n} l_k(u)+\langle G_k(u),x-x_k\rangle+\frac{1}{2}\langle H_k(u,w)(x-x_k),x-x_k\rangle-\frac{w^3}{12M^2}\\
&=\max_{(u,w)\in\mathcal{F}_k} l_k(u)-\frac{1}{2}\left\langle H_k(u,w)^{-1}G_k(u),G_k(u)\right\rangle-\frac{1}{12M^2}w^3=\beta^*.\end{aligned}$$

Further, let $(u, w) \in \mathcal{F}_k$. Then, we have $G_k(u) = -H_k(u, w)(x_{k+1} - x_k)$ and thus:

$$\begin{aligned}
\theta(x_{k+1}, u) &= l_k(u) - \langle H_k(u,w)(x_{k+1} - x_k), x_{k+1} - x_k\rangle \\
&\quad + \frac{1}{2}\left\langle\left(\sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) + B\right)(x_{k+1} - x_k), x_{k+1} - x_k\right\rangle + \frac{M}{6} r_k^3 \\
&= l_k(u) - \frac{1}{2}\left\langle\left(\sum_{i=0}^{m} u_i \nabla^2 F_i(x_k) + B + \frac{w}{2} I_n\right)(x_{k+1} - x_k), x_{k+1} - x_k\right\rangle - \frac{w}{4} r_k^2 + \frac{M}{6} r_k^3 \\
&= \beta_k(u, w) + \frac{1}{12M^2} w^3 - \frac{w}{4} r_k^2 + \frac{M}{6} r_k^3 = \beta_k(u, w) + \frac{M}{12}\left(\frac{w}{M}\right)^3 - \frac{M}{4}\left(\frac{w}{M}\right) r_k^2 + \frac{M}{6} r_k^3 \\
&= \beta_k(u, w) + \frac{M}{12}\left(\frac{w}{M} + 2r_k\right)\left(r_k - \frac{w}{M}\right)^2,
\end{aligned}$$

which proves (5.17). Note that we have:

$$\nabla_w \beta_k(u, w) = \frac{1}{4}\|x_{k+1} - x_k\|^2 - \frac{1}{4M^2} w^2 = \frac{1}{4}\left(r_k + \frac{w}{M}\right)\left(r_k - \frac{w}{M}\right).$$

Therefore if β^* is attained at some $(u^*, w^*) > 0$ from $\mathcal{F}_k$, then $\nabla \beta_k(u^*, w^*) = 0$. This implies $\frac{w^*}{M} = r_k(u^*, w^*)$ and by (5.17) we conclude that $\theta^* = \beta^*$. ■

Remark 13. *The global minimum* of the nonsmooth nonconvex problem (5.16) is:

$$x_{k+1} = x_k - H_k(u_k, w_k)^{-1} G_k(u_k),$$

with (u_k, w_k) solution of the following convex dual problem:

$$\max_{(u,w)\in\mathcal{F}_k} \; l_k(u) - \frac{1}{2}\left\langle H_k(u, w)^{-1} G_k(u), G_k(u)\right\rangle - \frac{1}{12M^2} w^3, \tag{5.18}$$

i.e., a maximization of a concave function over a convex set $\mathcal{F}_k$. Hence, if m is of moderate size, this *convex dual problem* (5.18) can be solved very efficiently by interior point methods [66]. Thus, RHOTA algorithm is implementable for $p = 2$, since we can effectively compute in fact the *global minimum* x_{k+1} of the subproblem (5.7) for $g(\cdot) = \|\cdot\|$ using the powerful tools from convex optimization.

5.5 Applications of RHOTA to nonlinear programming

In this section, we investigate the behavior of RHOTA algorithm when applied to specific problems of the form (5.4). First, we consider solving systems of nonlinear equations and then we consider optimization problems with nonlinear equality constraints. In both cases, we derive new convergence rates that have not been yet considered in the literature.

5.5.1 Nonlinear least-squares

In this section we focus on the task of solving a system of nonlinear equations (we assume that this system admits solutions):

$$\text{Find } x \in \mathbb{R}^n \text{ such that}: \; F_i(x) = 0 \quad \forall i = 1:m \quad (\text{with } m \le n).$$

This problem can be reformulated as the following nonlinear least-squares problem:

$$\min_{x\in\mathbb{R}^n} \|F(x)\|, \tag{5.19}$$

which, essentially, represents a particular form of problem (5.4) with $h = 0$ and $g(\cdot) = \|\cdot\|$ (hence, $L_g = 1$). For this particular problem, we assume that the statements from Assumption 5.2.1 related to F_i's hold and additionally, there exists $\sigma > 0$ such that:

$$\sigma_{\min}(\nabla F(x)) \geq \sigma > 0, \quad \forall x \in \mathcal{L}(x_0). \tag{5.20}$$

This nondegeneracy condition has been considered frequently in the literature; see, e.g., [94, 111]. It holds, e.g., when the Mangasarian-Fromovitz constraint qualification is satisfied (i.e., the gradients of $F_i(x)$'s, $i = 1 : m$, are linearly independent for all $x \in \mathcal{L}(x_0)$) and $\mathcal{L}(x_0)$ is bounded, see [111]. Note that to ensure a global convergence guarantee, the nondegenerate assumption must be satisfied across the entire level set. However, for local convergence, this assumption only needs to hold in a (local) solution, denoted as x^*. For simplicity, denote $C_\sigma = \frac{M-\|L_p\|}{(p+1)!L_\mu}\left(\frac{p!\sigma}{\mu}\right)^{\frac{p+1}{p}}$. Under this additional nondegeneracy condition, we can establish finite convergence for RHOTA.

Theorem 5.5.1. *Let the assumptions of Theorem 5.3.2 hold for problem (5.19). Let also $(x_k)_{k\geq 0}$ be generated by RHOTA algorithm and $(y_k)_{k\geq 0}$ as defined in (5.11), and, additionally, assume that $\sigma_{\min}(\nabla F(x)) \geq \sigma > 0$ for all $x \in \mathcal{L}(x_0)$. Then, there exists finite $k \in \{0, 1, \cdots, \bar{k}\}$, with $\bar{k} = \left\lceil \frac{f(x_0)}{C_\sigma} \right\rceil$, such that either $F(x_k) = 0$ or $F(y_k) = 0$.*

Proof. Combining (5.9) with the first statement of Theorem 5.3.2, we obtain:

$$f(x_{k+1}) \leq f(x_k) - \frac{M - \|L_p\|}{(p+1)!}\|x_{k+1} - x_k\|^{p+1} \leq f(x_k) - \frac{M - \|L_p\|}{(p+1)!L_\mu}\|y_{k+1} - x_k\|^{p+1}.$$

From the optimality condition of y_{k+1} (see (5.11)), we get:

$$-\frac{\mu}{p!}\|y_{k+1} - x_k\|^{p-1}(y_{k+1} - x_k) \in \partial f(y_{k+1}) = \nabla F(y_{k+1})d_{k+1},$$

where

$$d_{k+1} \in \partial\|F(y_{k+1})\| = \begin{cases} \frac{F(y_{k+1})}{\|F(y_{k+1})\|} & \text{if } \ F(y_{k+1}) \neq 0 \\ \{d \in \mathbb{R}^m : \|d\| \leq 1\} & \text{if } \ F(y_{k+1}) = 0. \end{cases} \tag{5.21}$$

We distinguish two cases. First, given $\bar{k} \geq 1$, consider that for all $k \in \{0, 1, \cdots, \bar{k} - 1\}$, $F(y_{k+1}) \neq 0$. Since $(x_k)_{k\geq 0}, (y_k)_{k\geq 0} \subset \mathcal{L}(x_0)$, from the nondegeneracy condition of the Jacobians (i.e., $\|\nabla F(x)d\| \geq \sigma\|d\|$ for any $d \in \mathbb{R}^m$ and $x \in \mathcal{L}(x_0)$), we have:

$$\frac{\mu}{p!}\|y_{k+1} - x_k\|^p = \frac{\|\nabla F(y_{k+1})F(y_{k+1})\|}{\|F(y_{k+1})\|} \geq \sigma \quad \forall k = 0 : \bar{k} - 1.$$

Hence, we obtain constant decrease in function values for the RHOTA iterates:

$$f(x_{k+1}) \leq f(x_k) - \frac{M - \|L_p\|}{(p+1)!L_\mu}\left(\frac{p!\sigma}{\mu}\right)^{\frac{p+1}{p}} = f(x_k) - C_\sigma \quad \forall k = 0 : \bar{k} - 1.$$

Summing up the last inequality from 0 to k, we get:

$$0 \leq f(x_k) \leq f(x_0) - kC_\sigma \quad \forall k = 0 : \bar{k}.$$

Thus, if $\bar{k} = \left\lceil \frac{f(x_0)}{C_\sigma} \right\rceil$, we deduce that $0 \leq f(x_{\bar{k}}) \leq f(x_0) - \bar{k}C_\sigma \leq 0$, or equivalently $F(x_{\bar{k}}) = 0$. In the second case there exists some $k \in \{0, 1, \cdots, \bar{k} - 1\}$ such that $F(y_{k+1}) = 0$. Together, both cases prove our statement. ∎

5.5.2 Equality constrained nonlinear problems

Let us now consider an optimization problem with nonlinear equality constraints:

$$\min_{x\in\mathbb{R}^n} \; h(x) \quad \text{s.t.:} \quad F(x) = 0, \tag{5.22}$$

where $F(x) = (F_1(x), \cdots, F_m(x))$, with $m \leq n$, and h is proper lsc function. For a given positive constant ρ, the exact penalty reformulation of (5.22) is [49]:

$$\min_{x\in\mathbb{R}^n} f(x) := h(x) + \rho\|F(x)\|, \tag{5.23}$$

which fits into the formulation (5.4) with $g(\cdot) = \rho\|\cdot\|$. It is known that, under proper constraint qualification conditions and for sufficiently large ρ, any stationary point x^* of the exact penalty problem (5.23) corresponds to a Karush-Kuhn-Tucker (KKT) point of the constrained problem (5.22) (i.e., $\exists\lambda^*$ s.t. $0 \in \partial h(x^*) + \nabla F(x^*)^T\lambda^*$ and $F(x^*) = 0$), see e.g., [108]. In the realm of nonlinear programming, when the objective function h exhibits smoothness, constraint qualifications are naturally related to the constraints themselves, such as LICQ or MFCQ (see Section 2.4 for more details). However, when the objective function, h, takes on a non-smooth character, a shift occurs, necessitating the introduction of new constraint qualifications. This adjustment becomes imperative because the non-smoothness of the objective function has the potential to significantly impact the behavior and satisfaction of the constraints. Hence, in such scenarios, a nuanced understanding of these new constraint qualifications becomes essential to navigate the complexities inherent in optimizing non-smooth objectives within nonlinear programming. In this section, for problem (5.22), we assume that the statements from Assumption 5.2.1 related to F_i's and f hold, and additionally, there exists $\sigma > 0$ such that the following constraint qualification holds:

$$\sigma\|\lambda\| \leq \operatorname{dist}\left(-\nabla F(x)^T\lambda, \partial^\infty h(x)\right) \quad \forall x \in \mathcal{L}(x_0) \;\text{ and }\; \lambda \in \partial\|F(x)\|. \tag{5.24}$$

Note that if $h = 0$ or h is locally Lipschitz continuous, then $\partial^\infty h(x) = \{0\}$ (see Theorem 9.13 in [39]) and thus (5.24) reduces to the nondegeneracy condition from Section 5.5.1: $\sigma_{\min}(\nabla F(x)) \geq \sigma$ for all $x \in \mathcal{L}(x_0)$. A constraint qualification condition of the form $\sigma\|\lambda\| \leq \operatorname{dist}\left(-\nabla F(x)^T\lambda, \partial h(x)\right)$ for all $x \in \mathcal{L}(x_0)$ and $\lambda \in \partial\|F(x)\|$ has been adopted when analyzing the convergence of iterative algorithms for solving optimization problems with nonconvex functional constraints; see, e.g., [95, 96]. However, in [95, 96], the proposed constraint qualification condition loses coherence in cases where, e.g., the nonsmooth component exhibits (local) Lipschitz continuity, such as $h(\cdot) = \|\cdot\|_1$ or $\|\cdot\|_2$, while our (5.24) imposes only a condition on the Jacobian ∇F for this particular case.

Theorem 5.5.2. *Let the assumptions of Theorem 5.3.2 hold, and, additionally, the constraint qualification condition (5.24) holds. Let $\bar{\rho} > 0$ be fixed sufficiently large and the sequence $(x_k)_{k\geq 0}$ be generated by RHOTA applied to penalty problem (5.23), with $\rho \geq \bar{\rho}$, $M \geq 2\rho\|L_p\|$, and $(y_k)_{k\geq 0}$ be given in (5.11), with $\mu = 2(M + \rho\|L_p\|)$. Then, any limit point of the sequence $(x_k)_{k\geq 0}$ is a KKT point of (5.22). Moreover, the convergence rate to a KKT point is of order $\mathcal{O}\left(\rho k^{-\frac{p}{p+1}}\right)$.*

Proof. From the optimality conditions of y_{k+1} applied to f given in (5.23) (see (5.11)), there exists $\lambda_{k+1} \in \partial\|F(y_{k+1})\|$ such that:

$$\frac{\mu}{p!}\|y_{k+1} - x_k\|^{p-1}(x_k - y_{k+1}) \in \rho\nabla F(y_{k+1})^T\lambda_{k+1} + \partial h(y_{k+1}) \quad \forall k \geq 0. \tag{5.25}$$

This implies that (for simplicity, we denote $\mathcal{N}^{k+1}_{\text{epi}h} = \mathcal{N}_{\text{epi}h}\big(y_{k+1}, h(y_{k+1})\big)$):

$$\begin{aligned}
&\text{dist}\Big(\left(-\rho\nabla F(y_{k+1})^T\lambda_{k+1}, 0\right), \mathcal{N}^{k+1}_{\text{epi}h}\Big) - \left\| \left(\frac{\mu}{p!}\|y_{k+1} - x_k\|^{p-1}(y_{k+1} - x_k), 1\right) \right\| \\
&\leq \text{dist}\left(\Big(-\rho\nabla F(y_{k+1})^T\lambda_{k+1} - \frac{\mu}{p!}\|y_{k+1}-x_k\|^{p-1}(y_{k+1}-x_k), -1\Big), \mathcal{N}^{k+1}_{\text{epi}h}\right) = 0.
\end{aligned}$$

On the other hand, from the definition of the horizon subdifferential (2.13), we get:

$$\text{dist}\left(-\rho\nabla F(y_{k+1})^T\lambda_{k+1}, \partial^\infty h(y_{k+1})\right) = \text{dist}\Big(\big(-\rho\nabla F(y_{k+1})^T\lambda_{k+1}, 0\big), \mathcal{N}^{k+1}_{\text{epi}h}\Big).$$

Therefore, combining the last two inequalities with the constraint qualification condition and using that $\partial^\infty h(y_{k+1})$ is a cone, we obtain for any $\rho > 0$:

$$\sigma\rho\|\lambda_{k+1}\| \leq \text{dist}\left(-\rho\nabla F(y_{k+1})^T\lambda_{k+1}, \partial^\infty h(y_{k+1})\right) \leq \frac{\mu}{p!}\|y_{k+1} - x_k\|^p + 1. \tag{5.26}$$

Or, equivalently, using the definition of M and $\rho \geq \bar{\rho}$, we have:

$$\|\lambda_{k+1}\| \leq \left(\frac{2M}{\sigma\bar{\rho}p!} + \frac{2\|L_p\|}{\sigma p!}\right)\|y_{k+1} - x_k\|^p + \frac{1}{\sigma\bar{\rho}}.$$

Since $\|y_{k+1} - x_k\| \to 0$ as $k \to \infty$ (see Theorems 5.3.1 and 5.3.2), then the previous relation implies that for fixed $\bar{\rho} > 0$ sufficiently large (e.g., $\sigma\bar{\rho} > 1$) there exists integer $\bar{k} \geq 0$ such that:

$$\|\lambda_{k+1}\| < 1 \Longrightarrow F(y_{k+1}) \overset{(5.21)}{=} 0 \quad \forall k \geq \bar{k},\ \rho \geq \bar{\rho}.$$

Hence, feasibility is achieved after a finite number of iterations. Additionally, it also follows from (5.25) that for any $k \geq 0$ there exists $h_{y_{k+1}} \in \partial h(y_{k+1})$ such that:

$$\|\nabla F(y_{k+1})^T(\rho\lambda_{k+1}) + h_{y_{k+1}}\| = \frac{\mu}{p!}\|y_{k+1} - x_k\|^p \to 0 \ \text{ as } \ k \to \infty.$$

Using the closedness of the graph of ∂h and basic limit rules, we deduce that any limit point of the sequence $(y_k)_{k\geq 0}$ is a KKT point of (5.22). Since the set of limit points of $(x_k)_{k\geq 0}$ coincides with the set of limit points of $(y_k)_{k\geq 0}$ (see Theorem 5.3.2), the first statement follows. Further, from Theorem 5.3.2, there exists $\bar{k} \in \{0, \cdots, k\}$ such that:

$$S_f(y_{\bar{k}+1}) \leq \left(\frac{\left(\frac{\mu+\delta+\rho\|L_p\|-M}{\mu-(M+\rho\|L_p\|)}\right)\mu^{\frac{p+1}{p}}(p+1)!}{(M-\rho\|L_p\|)(p!)^{\frac{p+1}{p}}(k+1)}(f(x_0) - f^*)\right)^{\frac{p}{p+1}}.$$

Since Assumption 5.2.1.3 holds, then $f(x) = h(x) + \rho\|F(x)\| \geq f^*$ and, consequently, we have $f(x_0) - f^* = \mathcal{O}(\rho)$. In addition, since $\delta \ll \rho$, we deduce the following bound:

$$S_f(y_{\bar{k}+1}) \leq \mathcal{O}\left(\frac{\rho}{k^{\frac{p}{p+1}}}\right). \tag{5.27}$$

Further, combining (5.26) with first assertion of Th. 5.3.2 and with eq. (5.10), we get:

$$\|\lambda_{\bar{k}+1}\| \leq \frac{\mu(L_\mu)^{\frac{p}{p+1}}}{p!\sigma\rho}\frac{((f(x_0) - f^*)(p+1)!)^{\frac{p}{p+1}}}{((M-\rho\|L_p\|)(k+1))^{\frac{p}{p+1}}} + \frac{1}{\sigma\rho} = \mathcal{O}\left(\frac{1}{k^{\frac{p}{p+1}}}\right) + \frac{1}{\sigma\rho},$$

where $\mathcal{O}(\cdot)$ does not depend on ρ. Hence, for any given ϵ, with $0 < \epsilon < \frac{1}{2}$, and for any $\rho > \frac{2}{\sigma}$, if

$k \geq \mathcal{O}\left(\rho^{\frac{p+1}{p}} \epsilon^{-\frac{p+1}{p}}\right)$, then there exists $h_{y_{\bar{k}+1}} \in \partial h(y_{\bar{k}+1})$ such that:

$$S_f(y_{\bar{k}+1}) = \|\nabla F(y_{\bar{k}+1})^T(\rho\lambda_{\bar{k}+1}) + h_{y_{\bar{k}+1}}\| \leq \epsilon \text{ and } \|\lambda_{\bar{k}+1}\| \leq \epsilon + \frac{1}{\sigma\rho} < 1 \Rightarrow F(y_{\bar{k}+1}) \overset{(5.21)}{=} 0,$$

i.e., $y_{\bar{k}+1}$ satisfies ϵ-KKT conditions (but exact feasibility), i.e., second statement. ■

Remark 14. From previous proof, one notices that in order to guarantee feasibility, ρ needs to be sufficiently large, e.g., $\rho > \frac{1}{\sigma}$. On the other hand, it is known that in exact penalty methods, one needs to choose ρ larger than the norm of Lagrange multiplier associated to a KKT point of (5.22) [49]. Let (x^*, λ^*) be a KKT point of (5.22), i.e.:

$$(-\nabla F(x^*)^T \lambda^*, -1) \in \mathcal{N}_{\text{epi}h}(x^*, h(x^*)).$$

This implies that:

$$\sigma\|\lambda^*\| \overset{(5.24)}{\leq} \text{dist}(-\nabla F(x^*)^T \lambda^*, \partial^\infty h(x^*)) \leq 1 \quad \Rightarrow \quad \rho > \frac{1}{\sigma} \geq \|\lambda^*\|,$$

i.e., we have established the connection between our lower bound $1/\sigma$ and the known lower bound from literature $\|\lambda^*\|$ on the exact penalty parameter ρ. *To the best of our knowledge, Theorem 5.5.2 provides the first convergence results for a higher-order exact penalty method for solving the equality constrained optimization problem* (5.22)*, i.e., finding a KKT point.* Specifically, for p very large, we get a rate of order $\mathcal{O}(\epsilon^{-1})$, while for $p = 1$ our rate $\mathcal{O}(\epsilon^{-2})$ aligns with that previously obtained in e.g., [108].

In Chapter 7, we will implement RHOTA algorithm on concrete applications to evaluate its performance and versatility. Through a series of case studies, we will test the algorithm across various domains, including power systems, phase retrieval, and output feedback control problems, analyzing how it handles complex data structures and different operational conditions. The outcomes from these applications will offer insights into the algorithm's practical utility and robustness.

5.6 Conclusions

In this chapter, we introduced a regularized higher-order Taylor approximation (called RHOTA) method designed to solve composite problems, such as nonlinear least-squares, which take the form given by equation (5.4). Our approach utilizes higher-order derivatives to construct a model that closely approximates the objective function. We derived global convergence guarantees for general cases, and we demonstrated faster convergence rates under the Kurdyka-Łojasiewicz property. Our discussion includes an efficient implementation of the proposed method. We also explored extensions of our theoretical results to nonlinear systems of equations and constrained optimization problems.

6 Inexact first-order oracle of degree q

This chapter addresses a simple composite optimization problem where direct access to the first order information of the smooth component is unavailable. We introduce an inexact first order oracle of degree q, which arises naturally in various contexts, such as in approximate gradient methods. Subsequently, we examine the behavior of an inexact proximal gradient method for both convex and non-convex problems. We establish global convergence rates that depend on the degree q and demonstrate that our convergence rates are better as q increases. Additionally, we provide numerical simulations to illustrate the effectiveness of our proposed approach.

The chapter is structured as follows: Section 6.1 provides a comprehensive literature review on inexact first order oracles and methods. In Section 6.2, we introduce our inexact first order oracle of degree q, along with several examples that satisfy our definition. Following this, in Section 6.3, we present gradient based algorithms based on the proposed inexact oracle and derive global convergence results in both convex and nonconvex scenarios. Section 6.4 then delves into numerical simulations and their corresponding results. The content presented in this chapter is derived from the published paper [28].

6.1 State of the art

In this chapter, we consider the following simple composite optimization problem:

$$\min_{x\in\mathbb{E}} f(x) := F(x) + h(x), \tag{6.1}$$

where $h : \mathbb{E} \to \bar{\mathbb{R}}$ is a simple (i.e., proximal easy) closed convex function, $F : \mathbb{E} \to \mathbb{R}$ is a general lower semicontinuous function (possibly nonconvex) and there exist f_∞ such that $f(x) \geq f_\infty > -\infty$ for all $x \in \operatorname{dom} f = \operatorname{dom} h$. We assume that we can compute exactly the proximal operator of h and that we cannot have access to the (sub)differential of F but can compute an approximation of it at any given point.

Gradient-based optimization methods are commonly applied in scenarios where high accuracy is not crucial, such as in machine learning, data analysis, signal processing, and statistics [15, 16, 17, 18]. Traditional convergence analyses of these methods typically assume exact gradient information for the objective function. Consider the following optimization problem (particular case of problem (6.1) by considering h the indicator function of the set Q):

$$\min_{x\in Q} F(x),$$

where Q is a simple convex and compact set in $\mathbb{R}^n$, and f is a convex function with L- Lipschitz continuous gradients, i.e.:

$$\|\nabla F(x) - \nabla F(y)\| \leq L\|x - y\|, \quad \forall x, y \in Q.$$

It is well established that this problem can be solved by a gradient type method with a complexity of $\mathcal{O}(\epsilon^{-1/2})$, where ϵ is the desired precision accuracy [112]. However, in many real-world applications, exact gradients are often unavailable or impractical to compute. This is especially true when gradients are derived by solving another optimization problem or involve large datasets. In such cases, using inexact (approximate) gradient information becomes necessary. Optimization algorithms that rely on inexact first-order oracles are widely studied in the literature; see, for example, [113, 114, 115, 116, 117, 118, 86, 119]. For example, D'Aspremont [120] demonstrated that even when substituting the exact gradient with an approximate one, significant computational and storage savings can be achieved without compromising the optimal convergence rate. To maintain the desired complexity, the approximate gradient $\tilde{\nabla} f(x)$ at any $x \in Q$ must meet the following condition [120]:

$$\langle \tilde{\nabla} F(x) - \nabla F(x), y - z \rangle \leq \delta, \quad \forall y, z \in Q, \tag{6.2}$$

where $\delta > 0$ represent a positive tolerance indicating the precision of an approximate gradient. By meeting this condition, gradient-based methods can achieve substantial reductions in computational cost, thereby facilitating their application across a wider range of use cases without compromising efficiency or scalability. For instance, when this approach is applied to semidefinite programs, it often involves calculating only the first few leading eigenvalues of the current iterate instead of performing a complete matrix exponential [120]. This strategy significantly reduces computational overhead, making these methods more practical and scalable. However, condition (6.2) requires that the set Q be bounded, and it also necessitates the existence of the gradient at all points, as it must be compared with the approximate gradient. This creates a need to address these limitations, encouraging researchers to propose a new, more suitable definition of inexactness.

Further, paper [115] considers the case where h is the indicator function of a convex set Q and F is a convex function, and introduces the so-called inexact first-order (δ, L)-oracle for F, i.e., for any $y \in Q$ one can compute an inexact oracle consisting of a pair $(F_{\delta,L}(y), g_{\delta,L}(y))$ such that:

$$0 \leq F_{\delta,L}(x) - \Big(F_{\delta,L}(y) + \langle g_{\delta,L}(y), x - y \rangle\Big) \leq \frac{L}{2}\|x - y\|^2 + \delta \quad \forall x \in Q. \tag{6.3}$$

It has been shown that when f is smooth, condition (6.2) implies to condition (6.3) (refer to Section 7 in [115]). Then, [115] introduces (fast) inexact first-order methods based on $g_{\delta,L}(y)$ information and derives asymptotic convergence in function values of order $\mathcal{O}\left(\frac{1}{k} + \delta\right)$ or $\mathcal{O}\left(\frac{1}{k^2} + k\delta\right)$, respectively. One can notice that in the nonaccelerated scheme, the objective function accuracy decreases with k and asymptotically tends to δ, while in the accelerated scheme, there is error accumulation. Further, [117] considers problem (6.1) with domain of h bounded, and introduces the following inexact first-order oracle:

$$|F(x) - F_{\delta,L}(x)| \leq \delta, \;\; F(x) - F_{\delta,L}(y) - \langle g_{\delta,L}(y), x - y \rangle \leq \frac{L}{2}\|x - y\|^2 + \delta.$$

Under the assumptions that F is nonconvex and h is convex but with bounded domain, [117] derives a sublinear rate in the squared norm of the generalized gradient mapping of order $\mathcal{O}\left(\frac{1}{k} + \delta\right)$ for an inexact proximal gradient method based on $g_{\delta,L}(y)$ information. For a clearer grasp of this inexact framework, let us illustrate with an example where such an occurrence arises.

Example 6.1.1. Let f be a convex differentiable function and the derivative is L-Lipschitz continuous. Suppose that at each point x, obtaining the gradient $\nabla f(x)$ either isn't available or is challenging to compute directly. However, we can compute an approximation of it, denoted as $g_\Delta(x)$, satisfying the following conditions:

$$\|\nabla f(x) - g_\Delta(x)\| \leq \Delta,$$

where Δ is a positive constant. Let's demonstrate how this scenario fits into the inexact definitions provided in (6.2) and (6.3). Indeed, by employing the Cauchy-Schwarz inequality, we obtain the following for all $x, y, z \in Q$:

$$\langle g_\Delta(x) - \nabla f(x), y - z\rangle \leq \|g_\Delta(x) - \nabla f(x)\| \|y - z\| \leq \Delta \operatorname{diam}(Q).$$

This implies that inequality (6.2) holds with $\delta = \Delta \operatorname{diam} Q$ and $\tilde{\nabla} F(x) = g_\Delta(x)$, provided that Q is bounded. Further, using the smoothness property of f, we get:

$$f(y) - f(x) - \langle \nabla f(x), y - x\rangle \leq \frac{L}{2}\|y - x\|^2.$$

This implies that:

$$f(y) - f(x) - \langle g_\Delta(x), y - x\rangle + \langle g_\Delta(x) - \nabla f(x), y - x\rangle \leq \frac{L}{2}\|y - x\|^2.$$

Thus:

$$\begin{aligned} f(y) - f(x) - \langle g_\Delta(x), y - x\rangle &\leq \frac{L}{2}\|y - x\|^2 + \|\nabla f(x) - g_\Delta(x)\| \, \|y - x\| \\ &\leq \frac{L}{2}\|y - x\|^2 + \Delta\|y - x\| \\ &\leq \frac{L}{2}\|y - x\|^2 + \Delta \operatorname{diam}(Q). \end{aligned}$$

Further, from the convexity of f, we get:

$$\begin{aligned} f(y) &\geq f(x) + \langle \nabla f(x), y - x\rangle \\ &\geq f(x) + \langle g_\Delta(x), y - x\rangle - \Delta\|y - x\| \\ &\geq f(x) - \Delta \operatorname{diam}(Q) + \langle g_\Delta(x), y - x\rangle. \end{aligned}$$

Hence, inequality (6.3) holds with $L = L_f$, $g_{L,\delta}(x) = g_\Delta(x)$, $f_{\delta,L}(x) = f(x) - \Delta \operatorname{diam}(Q)$ and $\delta = \Delta \operatorname{diam}(Q)$, provided that the set Q is bounded.

Hence, drawing from this example, the preceding results offer convergence rates under the assumption of the boundedness of the domain of f (or equivalently, of the set Q). An open question arises: *Can we modify the previous definitions of inexact first-order oracles to encompass both convex and nonconvex settings, thereby improving the convergence results? Can we do so without assuming the boundedness of the domain, as exemplified in Example 6.1.1? In essence, is it possible whether a general inexact oracle can be defined to bridge the gap between an exact oracle (providing exact gradient information) and the existing inexact first-order oracle definitions found in the literature [115, 117]?.*

In this chapter we answer to these questions positively for both convex and nonconvex problems, introducing a suitable definition of inexactness for a first-order oracle for F involving some degree $0 \leq q < 2$, which consist in multiplying the constant δ in (6.3) with quantity $\|x - y\|^q$ (see Definition 6.2.1).

6.2 Inexact first-order oracle of degree q

In this section, we introduce our new inexact first-order oracle of degree $0 \leq q < 2$ and provide some nontrivial examples that fit into our framework. Our oracle can deal with general functions (possibly with unbounded domain), unlike the previous results in [115, 117], but requires exact zero-order information.

Definition 6.2.1. The function F is equipped with an inexact first-order (δ, L)-oracle of degree $q \in [0, 2)$ if for any $y \in \operatorname{dom} f$ one can compute $g_{\delta,L,q}(y) \in \mathbb{E}^*$ such that:

$$F(x) - (F(y) + \langle g_{\delta,L,q}(y), x - y\rangle) \leq \frac{L}{2}\|x - y\|^2 + \delta\|x - y\|^q \quad \forall x \in \operatorname{dom} f. \tag{6.4}$$

To the best of our knowledge this definition of a first-order inexact oracle is new. The motivation behind this definition is to introduce a versatile inexact first-order oracle framework that bridges the gap between exact oracle (exact gradient information, i.e., $q = 2$) and the existing inexact first-order oracle definitions found in the literature (i.e., $q = 0$). More specifically, when $q = 2$, Definition 6.2.1 aligns with established results for smooth functions under exact gradient information, while when $q = 0$, our definition has been previously explored in the literature, see [115, 117]. Next, we provide several examples that satisfy Definition 6.2.1 naturally, and then we provide theoretical results showing the advantages of this new inexact oracle over the existing ones from the literature.

Example 6.2.2. (Smooth function with inexact first-order oracle). Let F be differentiable and its gradient be Lipschitz continuous with constant L_F over $\operatorname{dom} f$. Assume that for any $x \in \operatorname{dom} f$, one can compute $g_{\Delta,L_F}(x)$, an approximation of the gradient $\nabla F(x)$ satisfying:

$$\|\nabla F(x) - g_{\Delta,L_F}(x)\| \leq \Delta. \tag{6.5}$$

Then, F is equipped with an (δ, L)-oracle of degree $q = 1$ as in Definition 6.2.1, with $\delta = \Delta$, $L = L_F$, and $g_{\delta,L,1}(x) = g_{\Delta,L_F}(x)$.

Proof. Indeed, since F is L_F-smooth, we get:

$$F(y) - F(x) - \langle\nabla F(x), y - x\rangle \leq \frac{L_F}{2}\|y - x\|^2.$$

Implies that:

$$F(x) - F(y) - \langle g_{\Delta,L_F,q}(y), x - y\rangle + \langle g_{\Delta,L_F,q}(y) - \nabla f(y), x - y\rangle \leq \frac{L_F}{2}\|x - y\|^2.$$

Thus:

$$\begin{aligned} F(y) - F(x) - \langle g_{\Delta,L_F}(x), y - x\rangle &\leq \frac{L_F}{2}\|y - x\|^2 + \|\nabla F(x) - g_{\Delta,L_F}(x)\|\,\|y - x\| \\ &\leq \frac{L_F}{2}\|y - x\|^2 + \Delta\|y - x\|. \end{aligned}$$

which completes our statement. ■

Finite sum optimization problems appear widely in machine learning [17] and deal with an objective $F(x) := \sum_{i=1}^N F_i(x)$, where N is possibly large. In the stochastic setting, we sample stochastic derivatives at each iteration in order to form a mini-batch approximation for the gradient of F. If we define:

$$g_S(x) = \frac{1}{|S|}\sum_{j\in S}\nabla F_i(x), \tag{6.6}$$

where S is a subset of $\{1, \ldots, N\}$, then condition (6.5) holds with probability at least $1 - \Delta$ if the batch size S satisfies $|S| = \mathcal{O}\left(\frac{\Delta^2}{L_F^2} + \frac{1}{N}\right)^{-1}$ (see Lemma 11 in [121]).

Remark 15. This example has been also considered in [115, 117]. However, in these papers, δ depends on the diameter of the domain of f, assumed to be bounded. Our inexact oracle is more general and doesn't require boundedness of the domain of f, i.e., in our case $\delta = \Delta$, while in [115, 117], $\delta = 2\Delta D$, where D is the diameter of the domain of f. Hence, our definition is more natural in this setting.

Example 6.2.3. (Computations at shifted points) Let F be differentiable with Lipschitz continuous gradient with constant L_F over $\operatorname{dom} f$. For any $x \in \operatorname{dom} f$ we assume we can compute the exact value of the gradient, albeit evaluated at a shifted point $\bar{x}$, different from x and satisfying $\|x - \bar{x}\| \leq \Delta$. Then, F is equipped with a (δ, L)-oracle of degree $q = 1$ as in Definition 6.2.1, with $g_{\delta,L,1}(x) = \nabla F(\bar{x})$, $L = L_F$ and $\delta = L_F \Delta$. Indeed, since F is L_F smooth, we have:

$$\begin{aligned}
F(y) &\leq F(x) + \langle \nabla F(x), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&= F(x) + \langle \nabla F(\bar{x}), y - x\rangle + \langle \nabla F(x) - \nabla F(\bar{x}), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&\leq F(x) + \langle \nabla F(\bar{x}), y - x\rangle + \frac{L_F}{2}\|y - x\|^2 + L_F\|x - \bar{x}\|\|y - x\|,
\end{aligned}$$

where the second inequality follows from the Cauchy-Schwartz inequality. This proves our statement.

Remark 16. This example was also considered in [115, 117], with the corresponding (δ, L)-oracle having $\delta = L_F\Delta^2$, $L = 2L_F$ and $q = 0$. Note that our L in Definition 6.2.1 is twice smaller than the corresponding L in [115, 117].

Example 6.2.4. (Accuracy measures for approximate solutions) Let us consider a F that is L_F smooth, given by:

$$F(x) = \max_{u \in U} \psi(x, u) := \max_{u \in U} G(u) + \langle Au, x\rangle,$$

where $A : \mathbb{E} \to \mathbb{E}^*$ is a linear operator and $G(\cdot)$ is a differentiable strongly concave function with concavity parameter $\kappa > 0$. Under these assumptions, the maximization problem $\max_{u \in U} \psi(x, u)$ has only one optimal solution $u^*(x)$ for a given x. Moreover, F is convex and smooth with Lipschitz continuous gradient $\nabla F(x) = \nabla_x \psi(x, u^*(x)) = Au^*(x)$ having Lipschitz constant $L_F = \frac{1}{\kappa}\|A\|^2$ [115]. Suppose that for any $x \in \operatorname{dom} f$, one can compute u_x an approximate minimizer of $\psi(x, u)$ such that $\|u^*(x) - u_x\| \leq \Delta$. Then, F is equipped with (δ, L)-oracle of degree $q = 1$ with $\delta = \Delta\|A\|$, $L = L_F$ and $g_{\delta,L,1}(x) = Au_x$. Indeed, since F has Lipschitz-continuous gradient, we have:

$$\begin{aligned}
F(y) &\leq F(x) + \langle \nabla F(x), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&= F(x) + \langle \nabla_x \psi(x, u^*(x)), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&= F(x) + \langle Au^*(x), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&= F(x) + \langle Au_x, y - x\rangle + \langle A(u^*(x) - u_x), y - x\rangle + \frac{L_F}{2}\|y - x\|^2, \\
&\leq F(x) + \langle Au_x, y - x\rangle + \|A\|\|u^*(x) - u_x\|\|y - x\| + \frac{L_F}{2}\|y - x\|^2, \\
&\leq F(x) + \langle Au_x, y - x\rangle + \Delta\|A\|\|y - x\| + \frac{L_F}{2}\|y - x\|^2.
\end{aligned}$$

Hence, our statement follows.

Remark 17. This example was also considered in [115] with the corresponding $(\delta, L)-$ oracle having $\delta = \Delta$, $L = 2L_F$ and $q = 0$, while in our case, we have $\delta = \Delta\|A\|$, $L = L_F$ and $q = 1$.

Example 6.2.5. (Weak level of smoothness) Let F be a proper lower semicontinuous function with the subdifferential $\partial F(x)$ nonempty for all $x \in \operatorname{dom} f$. Assume that F satisfies the following Hölder condition with $H_\nu < \infty$:

$$\|g(x) - g(y)\| \leq H_\nu \|y - x\|^\nu, \tag{6.7}$$

for all $g(x) \in \partial F(x)$, $g(y) \in \partial F(y)$, where $x, y \in \operatorname{dom} f$ and $\nu \in [0, 1]$. Then, F is equipped with (δ, L)-oracle of degree q as in Definition 6.2.1, with $g_{\delta,L,q}(x) \in \partial F(x)$, for any arbitrary degree $0 \leq q < 1 + \nu$ and any accuracy $\delta > 0$, and a constant L depending on δ given by:

$$L(\delta) = \frac{1+\nu-q}{2-q}\left(\frac{H_\nu}{1+\nu}\right)^{\frac{2-q}{1+\nu-q}}\left(\frac{1-\nu}{\delta(2-q)}\right)^{\frac{1-\nu}{1+\nu-q}}.$$

Indeed, we have from Hölder condition [38]:

$$F(x) - F(y) - \langle g(y), x - y\rangle \leq \frac{H_\nu}{1+\nu}\|x - y\|^{1+\nu}.$$

For any given $\delta > 0$, we compute $L(\delta)$ such that the following inequality holds:

$$\frac{H_\nu}{1+\nu}\|x - y\|^{1+\nu} \leq \frac{L(\delta)}{2}\|x - y\|^2 + \delta\|x - y\|^q.$$

Denote $r = \|x - y\|$ and let $\lambda \in (0, 1)$. Using the weighted arithmetic-geometric mean inequality with $\alpha_1 = \lambda$ and $\alpha_2 = 1 - \lambda$, we have:

$$\begin{aligned}\frac{L(\delta)r^2}{2} + \delta r^q &= \lambda\frac{L(\delta)}{2\lambda}r^2 + (1-\lambda)\frac{\delta}{1-\lambda}r^q\\ &\geq \left(\frac{L(\delta)}{2\lambda}r^2\right)^\lambda\left(\frac{\delta}{1-\lambda}r^q\right)^{1-\lambda} = \left(\frac{L(\delta)}{2\lambda}\right)^\lambda\left(\frac{\delta}{1-\lambda}\right)^{1-\lambda} r^{2\lambda+q(1-\lambda)}.\end{aligned}$$

Thus $\frac{H_\nu}{1+\nu} = \left(\frac{L(\delta)}{2\lambda}\right)^\lambda\left(\frac{\delta}{1-\lambda}\right)^{1-\lambda}$ and $1+\nu = 2\lambda + q(1-\lambda)$. It follows that $\lambda = \frac{1+\nu-q}{2-q}$, $1 - \lambda = \frac{1-\nu}{2-q}$ and $\frac{1}{\lambda} - 1 = \frac{1-\nu}{1+\nu-q}$. Hence, for a given positive δ one may choose:

$$L(\delta) = 2\lambda\left(\frac{H_\nu}{1+\nu}\right)^{\frac{1}{\lambda}}\left(\frac{1-\lambda}{\delta}\right)^{\frac{1}{\lambda}-1} = \frac{1+\nu-q}{2-q}\left(\frac{H_\nu}{1+\nu}\right)^{\frac{2-q}{1+\nu-q}}\left(\frac{1-\nu}{\delta(2-q)}\right)^{\frac{1-\nu}{1+\nu-q}},$$

and this is our statement. Note that if $\nu > 0$, then we have $\partial F(x) = \{\nabla F(x)\}$ for all x and thus F is differentiable. Indeed, letting $y = x$ in (6.7) we get: $g(x) = \bar{g}(x)$. This implies that the set $\partial F(x)$ has a single element, thus F is differentiable. This example covers large classes of functions. Indeed, when $\nu = 1$, we get functions with Lipschitz-continuous subgradient. For $\nu < 1$, we get a weaker level of smoothness. In particular, when $\nu = 0$, we obtain functions whose subgradients have bounded variation. Clearly, the latter class includes functions whose subgradients are uniformly bounded by M (just take $H_0 = 2M$). It also covers functions smoothed by local averaging and Moreau–Yosida regularization (see [115] for more details). We believe that the readers may find other examples that satisfy our Definition 6.2.1 of an inexact first-order oracle of degree q.

6.3 Inexact proximal gradient methods

In this section, we introduce an inexact proximal gradient method based on the previous inexact oracle definition for solving (non)convex composite minimization problems (6.1). We derive complexity estimates for this algorithm and study the dependence between the accuracy of the oracle and the desired accuracy of the gradient or of the objective function. Hence, we consider the following Inexact Proximal Gradient Method (I-PGM). Note that Algorithm 1 is an inexact

Algorithm 6 Inexact proximal gradient method (I-PGM)

1. Given $x_0 \in \operatorname{dom} h$ and $0 \leq q < 2$.

   For $k \geq 0$ do:

2. Choose δ_k, L_k and α_k. Obtain $g_{\delta_k,L_k,q}(x_k)$.

3. Compute $x_{k+1} = \operatorname{prox}_{\alpha_k h}\left(x_k - \alpha_k g_{\delta_k,L_k,q}(x_k)\right)$.

proximal gradient method, where the inexactness comes from the approximate computation of the (sub)gradient of F, denoted $g_{\delta_k,L_k,q}(x_k)$. In the next sections we analyze the convergence behavior of this algorithm when $g_{\delta_k,L_k,q}(x_k)$ satisfies Definition 6.2.1.

6.3.1 Nonconvex convergence analysis

In this section we consider a nonconvex function F that admits an inexact first-order (δ, L)-oracle of degree q as in Definition 6.2.1. Using this definition and inequality (2.10), for all $\rho > 0$ we get the following upper bound:

$$F(x) - \Big(F(y) + \langle g_{\delta,L,q}(y), x - y\rangle\Big) \leq \frac{L + q\rho}{2}\|x - y\|^2 + \frac{(2-q)\delta^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}. \tag{6.8}$$

This inequality will play a key role in our convergence analysis. We define the *gradient mapping* at iteration k as $g_{\delta_k,L_k,q}(x_k)+p_{k+1}$, where $p_{k+1} \in \partial h(x_{k+1})$ such that $g_k+p_{k+1} = -\frac{1}{\alpha_k}(x_{k+1}-x_k)$ (i.e., p_{k+1} is the subgradient of h at x_{k+1} coming from the optimality condition of the prox at x_k). Next we analyze the global convergence of I-PGM in the norm of the gradient mapping. We have the following theorem:

Theorem 6.3.1. *Let F be a nonconvex function admitting a (δ_k, L_k)-oracle of degree $q \in [0, 2)$ at each iteration k, with $\delta_k \geq 0$ and $L_k > 0$ for all $k \geq 0$. Let $(x_k)_{k\geq 0}$ be generated by I-PGM and assume that $\alpha_k \leq \frac{1}{L_k+q\rho}$, for some arbitrary parameter $\rho > 0$. Then, there exists $p_{k+1} \in \partial h(x_{k+1})$ such that:*

$$\sum_{j=0}^{k} \alpha_j \|g_{\delta_j,L_j,q}(x_j) + p_{j+1}\|^2 \leq f(x_0) - f_\infty + \frac{\sum_{j=0}^{k}(2-q)\delta_j^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}. \tag{6.9}$$

Proof. Denote $g_{\delta_k,L_k,q}(x_k) = g_k$. From the optimality conditions of the proximal operator defining x_{k+1}, we have:

$$g_k + p_{k+1} = -\frac{1}{\alpha_k}(x_{k+1} - x_k).$$

Further, from inequality (6.8), we get:

$$
\begin{aligned}
F(x_{k+1}) &\leq F(x_k) + \langle g_k, x_{k+1} - x_k \rangle + \frac{L_k + q\rho}{2}\|x_{k+1} - x_k\|^2 + \frac{(2-q)\delta_k^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}} \\
&= F(x_k) + \langle g_k + p_{k+1}, x_{k+1} - x_k \rangle - \langle p_{k+1}, x_{k+1} - x_k \rangle + \frac{L_k + q\rho}{2}\|x_{k+1} - x_k\|^2 + \frac{(2-q)\delta_k^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}} \\
&\leq F(x_k) - \alpha_k\left(1 - \frac{(L_k + q\rho)\alpha_k}{2}\right)\|g_k + p_{k+1}\|^2 + h(x_k) - h(x_{k+1}) + \frac{(2-q)\delta_k^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}} \\
&\leq F(x_k) - \frac{\alpha_k}{2}\|g_k + p_{k+1}\|^2 + h(x_k) - h(x_{k+1}) + \frac{(2-q)\delta_k^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}},
\end{aligned}
$$

where the second inequality follows from the convexity of h and $p_{k+1} \in \partial h(x_{k+1})$, and the last inequality follows from the definition of α_k. Hence, we get that:

$$
f(x_{k+1}) \leq f(x_k) - \frac{\alpha_k}{2}\|g_k + p_{k+1}\|^2 + \frac{(2-q)\delta_k^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}.
$$

Summing up this inequality from $j = 0$ to $j = k$ and using the fact that $f(x_{k+1}) \geq f_\infty$, where recall that f_∞ denotes a finite lower bound for the objective function, we get:

$$
\sum_{j=0}^{k} \frac{\alpha_j}{2}\|g_j + p_{j+1}\|^2 \leq f(x_0) - f(x_{k+1}) + \frac{\sum_{j=0}^{k}(2-q)\delta_j^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}} \leq f(x_0) - f_\infty + \frac{\sum_{j=0}^{k}(2-q)\delta_j^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}.
$$

Hence, our statement follows. ■

For a particular choice of the algorithm parameters, we can get simpler convergence estimates.

Theorem 6.3.2. *Let the assumptions of Theorem 4.3.4 hold and consider for all $k \geq 0$:*

$$
L_k = L,\ \delta_k = \frac{\delta}{(k+1)^{\frac{\beta(2-q)}{2}}},\ \alpha_k = \frac{1}{(L + q\rho)(k+1)^\zeta},\ \text{where } \beta, \zeta \in [0, 1).
$$

Then, we have:

$$
\min_{j=0:k}\|g_j + p_{j+1}\|^2 \leq \frac{2(L + q\rho)(f(x_0) - f_\infty)}{(1-\zeta)(k+1)^{1-\zeta}} + \frac{(2-q)(L + q\rho)\delta^{\frac{2}{2-q}}}{(1-\zeta)(1-\beta)\rho^{\frac{q}{2-q}}(k+1)^{\beta-\zeta}}. \tag{6.10}
$$

Proof. Taking the minimum in the inequality (6.9), we get:

$$
\min_{j=0:k}\|g_j + p_{j+1}\|^2 \leq \frac{2(f(x_0) - f_\infty)}{\sum_{j=0}^{k}\alpha_j} + \frac{\sum_{j=0}^{k}(2-q)\delta_j^{\frac{2}{2-q}}}{\rho^{\frac{q}{2-q}}\sum_{j=0}^{k-1}\alpha_j}.
$$

Further, since we have:

$$
\sum_{j=0}^{k}\frac{1}{(L + q\rho)(j+1)^\zeta} = \sum_{j=1}^{k+1}\frac{1}{(L + q\rho)j^\zeta},
$$

and similarly for δ_j, we get:

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{2(L+q\rho)(f(x_0)-f_\infty)}{\sum_{j=1}^{k+1}\frac{1}{j^\zeta}} + \frac{(2-q)(L+q\rho)\delta^{\frac{2}{2-q}}\sum_{j=1}^{k+1}\frac{1}{j^\beta}}{\rho^{\frac{q}{2-q}}\sum_{j=1}^{k+1}\frac{1}{j^\zeta}}.$$

Since $0 \leq \zeta < 1$, then we have for all $k \geq 0$:

$$\begin{aligned}(1-\zeta)(k+1)^{1-\zeta} \leq \frac{(k+2)^{1-\zeta}-1}{1-\zeta} &= \int_1^{k+2}\frac{1}{u^\zeta}du \\ &\leq \sum_{j=1}^{k+1}\frac{1}{j^\zeta} \leq \int_1^{k+1}\left(\frac{1}{u^\zeta}\right)du + 1 \leq \frac{(k+1)^{1-\zeta}}{1-\zeta}.\end{aligned}$$

It follows that for all $k \geq 0$:

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{2(L+q\rho)(f(x_0)-f_\infty)}{(1-\zeta)(k+1)^{1-\zeta}} + \frac{(2-q)(L+q\rho)\delta^{\frac{2}{2-q}}}{(1-\zeta)(1-\beta)\rho^{\frac{q}{2-q}}(k+1)^{\beta-\zeta}}.$$

Hence, our statement follows. ∎

Let us analyze in more details the bound from Theorem 6.3.2. For simplicity, consider the case $q = 1$ (see Example 6.2.2). Then, we have:

$$\begin{aligned}\min_{j=0:k} \|g_j + p_{j+1}\|^2 &\leq \frac{2(L+\rho)(f(x_0)-f_\infty)}{(1-\zeta)(k+1)^{1-\zeta}} + \frac{(L+\rho)\delta^2}{\rho(1-\zeta)(1-\beta)(k+1)^{\beta-\zeta}} \\ &= \frac{2L(f(x_0)-f_\infty)}{(1-\zeta)(k+1)^{1-\zeta}} + \frac{2\rho(f(x_0)-f_\infty)}{(1-\zeta)(k+1)^{1-\zeta}} \\ &\quad + \frac{L\delta^2}{\rho(1-\zeta)(1-\beta)(k+1)^{\beta-\zeta}} + \frac{\delta^2}{(1-\zeta)(1-\beta)(k+1)^{\beta-\zeta}}.\end{aligned}$$

Denote $\Delta_0 := f(x_0) - f_\infty$. Since parameter $\rho > 0$ is a degree of freedom, minimizing the right hand side of the previous relation w.r.t. ρ we get an optimal choice $\rho = \frac{\delta\sqrt{L}}{\sqrt{2\Delta_0(1-\beta)}}(k+1)^{\frac{1-\beta}{2}}$. Hence, replacing this expression for ρ in the last inequality, we get:

$$\begin{aligned}\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq &\frac{2L\Delta_0}{(1-\zeta)(k+1)^{1-\zeta}} + \frac{2\delta\sqrt{2L\Delta_0}}{((1-\zeta)\sqrt{1-\beta})(k+1)^{\frac{1+\beta}{2}-\zeta}} \\ &+ \frac{\delta^2}{(1-\zeta)(1-\beta)(k+1)^{\beta-\zeta}}.\end{aligned}$$

This bound is of order $\mathcal{O}\left(\frac{1}{k^{1-\zeta}} + \frac{\delta}{k^{\frac{1+\beta}{2}-\zeta}} + \frac{\delta^2}{k^{\beta-\zeta}}\right)$. Note that, if $\beta > \zeta$, the gradient mapping $\min_{j=0:k}\|g_j + p_{j+1}\|^2$ converges regardless of the accuracy of the oracle δ and the convergence rate is of order $\mathcal{O}(k^{-\min(1-\zeta,\beta-\zeta)})$ (since we always have $\frac{1+\beta}{2} - \zeta \geq \beta - \zeta$). Note that this is not the case for $q = 0$, where the convergence rate is of order $\mathcal{O}\left(\frac{1}{k} + \delta\right)$, see also [117]. The following corollary provides a convergence rate for general q, but for a particular choice of the parameters ζ and β.

Corollary 6.3.3. *Let the assumptions of Theorem 6.3.2 hold and let assume that $\zeta = \beta = 0$. Then, we have the following convergence rates:*

1. *If $0 \leq q < 2$ and $\rho = L$, then $\delta_k = \delta$, $\alpha_k = \frac{1}{L+qL}$ and*

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{2(q+1)L\Delta_0}{k+1} + (q+1)(2-q)L^{\frac{2-2q}{2-q}}\delta^{\frac{2}{2-q}} \quad \forall k \geq 0.$$

2. *If $1 \leq q < 2$, fixing the number of iterations k and taking $\rho = \frac{L^{\frac{2-q}{2}}\delta}{(2\Delta_0)^{\frac{2-q}{2}}}(k+1)^{\frac{2-q}{2}}$, then $\delta_j = \delta$, $\alpha_j = \frac{1}{L+q\rho}$ for all $j = 0 : k$ and*

$$\begin{aligned}&\min_{j=0:k} \|g_j + p_{j+1}\|^2 \\ &\leq \frac{2L\Delta_0}{k+1} + \frac{L^{\frac{2-q}{2}}(2\Delta_0)^{\frac{q}{2}}\delta + (2-q)\delta L^{1-\frac{q}{2}}(2\Delta_0)^{\frac{q}{2}}}{(k+1)^{\frac{q}{2}}} + \frac{q(2-q)\delta^2 L^{1-q}(2\Delta_0)^{q-1}}{(k+1)^{q-1}}.\end{aligned}$$

Proof. Replacing $\zeta = \beta = 0$ in inequality (6.10), we get:

$$\begin{aligned}\min_{j=0:k} \|g_j + p_{j+1}\|^2 &\leq \frac{2(L+q\rho)\Delta_0}{k+1} + \frac{(2-q)(L+q\rho)\delta^{\frac{2}{2-q}}}{\rho^{\frac{q}{2-q}}} \\ &= \frac{2L\Delta_0}{k+1} + \frac{2q\rho\Delta_0}{k+1} + \frac{(2-q)L\delta^{\frac{2}{2-q}}}{\rho^{\frac{q}{2-q}}} + \frac{q(2-q)\delta^{\frac{2}{2-q}}}{\rho^{\frac{2q-2}{2-q}}}.\end{aligned}$$

If $0 \leq q < 2$, then taking $\rho = L$ in the last inequality we get the first statement. Further, if $1 \leq q < 2$, minimizing over ρ the second and the third terms of the right side of the last inequality yields the optimal choice $\rho = \frac{L^{\frac{2-q}{2}}\delta}{(2\Delta_0)^{\frac{2-q}{2}}}(k+1)^{\frac{2-q}{2}}$. Replacing this expression for ρ in the last inequality, we get:

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{2L\Delta_0}{k+1} + \frac{L^{\frac{2-q}{2}}(2\Delta_0)^{\frac{q}{2}}\delta + (2-q)\delta L^{1-\frac{q}{2}}(2\Delta_0)^{\frac{q}{2}}}{(k+1)^{\frac{q}{2}}} + \frac{q(2-q)\delta^2 L^{1-q}(2\Delta_0)^{q-1}}{(k+1)^{q-1}},$$

and this is the second statement. ■

Remark 18. Let us analyse in more details this convergence rate for Example 6.2.2. For $q = 0$, we have that $\delta = 2D\Delta$ and $L = L_F$, where D is the diameter of $\operatorname{dom} f$. Hence, the convergence rate in this case becomes:

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{4L_F\Delta_0}{k+1} + 4DL_F\Delta.$$

On the other hand, for $q = 1$, we have $\delta = \Delta$ and $L = L_F$. Thus, we get the following convergence rate:

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{4L_F\Delta_0}{k+1} + 2\Delta^2.$$

Hence, if we want to achieve $\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \epsilon$, for $q = 0$ we impose $4DL_F\Delta \leq \epsilon/2$, which implies that one needs to compute an approximate gradient with accuracy $\Delta = \mathcal{O}(\epsilon)$, while for $q = 1$ we impose $2\Delta^2 \leq \epsilon/2$, meaning that one only needs to compute an approximate gradient with accuracy $\Delta = \mathcal{O}(\epsilon^{1/2})$. Hence, for this example, it is more natural to use our inexact first-order oracle definition for $q = 1$ than for $q = 0$, since it requires less accuracy for approximating the true gradient.

Note that in the second result of Corollary 6.3.3, the parameter ρ depends on the difference $\Delta_0 = f(x_0) - f_\infty$, and, usually, f_∞ is unknown. In practice, we can approximate Δ_0 by using an estimate for f_∞ in place of its exact value. For example, one can consider $\Delta_0^k = f(x_0) - f_{\text{best}}^k$, where $f_{\text{best}}^k = \min_{j=0:k} f(x_j) - \varepsilon_k$ for some $\varepsilon_k \geq 0$, see [122]. Under this setting, the sequence ε_k and the iterates of I-PGM corresponding to the case of the second result of Corollary 6.3.3 are updated as follows:

Algorithm 7 Adaptive I-PGM algorithm when f_∞ is unknown

1. Given $\varepsilon_0 > 0$ and $f_{\text{best}}^0 = f(x_0) - \varepsilon_0$.
   For $k \geq 0$ do:
2. Compute x_{k+1} by I-PGM with $\Delta_0^k = f(x_0) - f_{\text{best}}^k$.
   While $f(x_{k+1}) < f_{\text{best}}^k$
   a) Set $\varepsilon_k = 2 \times \varepsilon_k$ and update $f_{\text{best}}^k = \min\limits_{j=0:k} f(x_j) - \varepsilon_k$.
   b) Re-compute x_{k+1} by I-PGM method.
   End While
3. Set $\varepsilon_{k+1} = \frac{\varepsilon_k}{2}$.

This process is well defined, i.e., the "while" step finishes in a finite number of iterations. Indeed, one can observe that if $\varepsilon_k \geq \min_{j=0:k} f(x_j) - f_\infty$ then $\varepsilon_k \geq \min_{j=0:k} f(x_j) - f(x_{k+1})$, which implies that $f(x_{k+1}) \geq f_{\text{best}}^k$. Additionally, we have $\varepsilon_k \leq 2(\min_{j=0:k} f(x_j) - f_\infty)$ for all $k \geq 0$. Hence, we can still derive a convergence rate for the second result of Corollary 6.3.3 using this adaptive process since one can observe that:

$$f(x_0) - f(x_{k+1}) \leq f(x_0) - f_{\text{best}}^k = \Delta_0^k.$$

Additionally, we have the following bound on Δ_0^k:

$$\begin{aligned}
\Delta_0^k &\leq f(x_0) - \min_{j=0:k} f(x_j) + 2\left(\min_{j=0:k} f(x_j) - f_\infty\right) \\
&= (f(x_0) - f_\infty) + \left(\min_{j=0:k} f(x_j) - f_\infty\right).
\end{aligned}$$

Hence, we can replace in (6.9) the difference $\Delta_0 = f(x_0) - f_\infty$ with Δ_0^k and then the second statement of Corollary 6.3.3 remains valid with Δ_0^k instead of Δ_0.

Remark 19. We observe that for $q = 0$ we recover the same convergence rate as in [117]. However, our result does not require the boundedness of the domain of f, while in [117] the rate depends explicitly of the diameter of the domain of f. Moreover, for $q > 0$ our convergence bounds are better than in [117], i.e., the coefficients of the terms in δ are either smaller or even tend to zero, while in [117] they are always constant.

Further, let us consider the case of Example 6.2.5, where F satisfies the Hölder condition with constant $\nu \in (0, 1]$ and $\beta = \zeta = 0$. We have shown that for any $\delta > 0$ this class of functions can be equipped with a (δ, L)-oracle of degree $q < 1 + \nu$ with $L = C(H_\nu, q)\left(\frac{1}{\delta}\right)^{\frac{1-\nu}{1+\nu-q}}$ (see Example 6.2.5 for the expression of the constant $C(H_\nu, q)$). In view of the first result of Corollary 6.3.3, after k iterations, we have:

$$\begin{aligned}\min_{j=0:k}\|g_j+p_{j+1}\|^2 &\leq \frac{2(q+1)\Delta_0 L}{k+1}+(q+1)(2-q)L^{\frac{2-2q}{2-q}}\delta^{\frac{2}{2-q}}\\
&=\frac{C_1}{k+1}\left(\frac{1}{\delta}\right)^{\frac{1-\nu}{1+\nu-q}}+C_2\left(\frac{1}{\delta}\right)^{\frac{(1-\nu)(2-2q)}{(1+\nu-q)(2-q)}}\delta^{\frac{2}{2-q}}\\
&=\frac{C_1}{k+1}\delta^{-\frac{1-\nu}{1+\nu-q}}+C_2\delta^{-\frac{(1-\nu)(2-2q)}{(1+\nu-q)(2-q)}+\frac{2}{2-q}}\\
&=\frac{C_1}{k+1}\delta^{-\frac{1-\nu}{1+\nu-q}}+C_2\delta^{\frac{2\nu}{1+\nu-q}},\end{aligned}$$

where $C_1 := 2(q+1)\Delta_0 C(H_\nu,q)$ and $C_2=(q+1)(2-q)C(H_\nu,q)^{\frac{2-2q}{2-q}}$. Since in this example we can choose δ, its optimal value can be computed from the following equation:

$$-\frac{C_1(1-\nu)}{(1+\nu-q)}\frac{1}{(k+1)}\delta^{\frac{q-2}{1+\nu-q}}+\frac{2\nu C_2}{1+\nu-q}\delta^{\frac{-1+\nu+q}{1+\nu-q}}=0.$$

Hence, we get:

$$\delta=C_3(k+1)^{-\frac{1+\nu-q}{1+\nu}},$$

where $C_3=\left(\frac{2\nu C_2}{(1-\nu)C_1}\right)^{-\frac{1+\nu-q}{1+\nu}}$. Replacing this optimal choice of δ in the last inequality, we get:

$$\min_{j=0:k}\|g_j+p_{j+1}\|^2\leq C_1C_3\left((k+1)^{-(1-\frac{1-\nu}{1+\nu})}\right)+C_2C_3\left((k+1)^{-\frac{2\nu}{1+\nu}}\right)=\frac{C_3(C_1+C_2)}{(k+1)^{\frac{2\nu}{1+\nu}}}.$$

Remark 20. Note that our convergence rate of order $\mathcal{O}(k^{-\frac{2\nu}{1+\nu}})$ for Algorithm 1 (I-PGM) for nonconvex problems having the first term F with a Hölder continuous gradient (Example 6.2.5) recovers the rate obtained in [117] under the same settings.

Finally, let us now show that when the gradient mapping is small enough, i.e., $\|g_k+p_{k+1}\|$ is small, x_{k+1} is a good approximation for a stationary point of problem (6.1). Note that any choice $\alpha_k\leq\frac{1}{L+q\rho_k}$ yields:

$$\|x_{k+1}-x_k\|\leq\frac{1}{L}\left\|\frac{1}{\alpha_k}(x_{k+1}-x_k)\right\|=\frac{1}{L}\|g_k+p_{k+1}\|.$$

Hence, if the gradient mapping is small, then the norm $\|x_{k+1}-x_k\|$ is also small.

Theorem 6.3.4. *Let $(x_k)_{k\geq 0}$ be generated by I-PGM and let $p_{k+1}\in\partial h(x_{k+1})$. Assume that we are in the case of Example 6.2.2. Then, we have:*

$$\text{dist}(0,\partial f(x_{k+1}))\leq\|g_{\Delta,L_F,q}(x_k)+p_{k+1}\|+L_F\|x_{k+1}-x_k\|+\Delta.$$

Further, if we are in the case of Example 6.2.5, then we have:

$$\text{dist}(0,\partial f(x_{k+1}))\leq\|g(x_k)+p_{k+1}\|+H_\nu\|x_{k+1}-x_k\|^\nu,\ \ g(x_k)\in\partial F(x_k).$$

Proof. Let us consider Example 6.2.2, where F is L_F smooth and h is convex. Since $\nabla F(x_{k+1})+$

$p_{k+1} \in \partial f(x_{k+1})$, then we have:

$$\begin{aligned}
&\|\nabla F(x_{k+1}) + p_{k+1}\| \\
&\leq \|g_{\Delta,L_F,q}(x_k) + p_{k+1}\| + \|\nabla F(x_k) - g_{\Delta,L_F,q}(x_k)\| + \|\nabla F(x_{k+1}) - \nabla F(x_k)\| \\
&\leq \|g_{\Delta,L_F,q}(x_k) + p_{k+1}\| + \Delta + L_F\|x_{k+1} - x_k\|.
\end{aligned}$$

Further, let us assume that we are in the case of Example 6.2.5. Then, we have $g(x_k) \in \partial F(x_k)$. Further, let $g(x_{k+1}) \in \partial F(x_{k+1})$, then we get:

$$\begin{aligned}
\|g(x_{k+1}) + p_{k+1}\| &\leq \|g(x_k) + p_{k+1}\| + \|g(x_{k+1}) - g(x_k)\| \\
&\leq \|g(x_k) + p_{k+1}\| + H_\nu\|x_{k+1} - x_k\|^\nu.
\end{aligned}$$

This proves our statements. ∎

Thus, for $\|\frac{1}{\alpha_k}(x_{k+1} - x_k)\| = \|g_k + p_{k+1}\|$ small, x_{k+1} is an approximate stationary point of problem (6.1). Note that our convergence rates from this section are better as q increases, i.e., the terms depending on δ are smaller for $q > 0$ than for $q = 0$. In particular, the power of δ in the convergence estimate is higher for $q \in (0, 1)$ than for $q = 0$, while for $q \geq 1$ the coefficients of δ even diminish with k. Hence, it is beneficial to have an inexact first-order oracle of degree $q > 0$, as this allows us to work with less accurate approximation of the (sub)gradient of the nonconvex function F than for $q = 0$.

6.3.2 Convex convergence analysis

In this section, we analyze the convergence rate of I-PGM for problem (6.1), where F is now assumed to be a convex function. By adding extra information to the oracle (6.4), we consider the following modification of Definition 6.2.1:

Definition 6.3.5. Let F be convex. Then it is equipped with an inexact first-order (δ, L)-oracle of degree $0 \leq q < 2$ if for any $y \in \mathrm{dom}\, f$ we can compute a vector $g_{\delta,L,q}(y)$ such that:

$$0 \leq F(x) - (F(y) + \langle g_{\delta,L,q}(y), x - y\rangle) \leq \frac{L}{2}\|x - y\|^2 + \delta\|y - x\|^q \quad \forall x \in \mathrm{dom}\, f. \tag{6.11}$$

Note that Example 6.2.5 satisfies this definition. In (6.11), the zero-order information is considered to be exact. This is not the case in [115], which considers the particular choice $q = 0$. Further, the first-order information $g_{\delta,L,q}$ is a subgradient of f at y in (6.11), while in [115] it is a δ-subgradient. However, using this inexact first-order oracle of degree q, I-PGM provides better rates compared to [115]. From (6.11) and (2.10), we get:

$$0 \leq F(x) - (F(y) + \langle g_{\delta,L,q}(y), x - y\rangle) \leq \frac{L + q\rho}{2}\|x - y\|^2 + \frac{(2 - q)\delta_q^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}, \tag{6.12}$$

for all $\rho > 0$. Next, we analyze the convergence rate of I-PGM in the convex setting. We have the following convergence rate:

Corollary 6.3.6. *Let F be a convex function admitting a (δ, L)-oracle of degree $q \in [0,2)$ (see Definition 6.3.5). Let $(x_k)_{k\geq 0}$ be generated by I-PGM and assume that $\alpha_k = \frac{1}{L+q\rho}$, with $\rho > 0$. Define $\hat{x}_k = \frac{\sum_{i=0}^{k} x_{i+1}}{k+1}$ and $R = \|x_0 - x^*\|$. Then, we have:*

$$f(\hat{x}_k) - f^* \leq \frac{(L+q\rho)R^2}{2k} + \frac{(2-q)\delta^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}. \tag{6.13}$$

Proof. It follows from (6.12) and Theorem 2 in [115]. ■

Since we have the freedom of choosing ρ, let us minimize the right-hand side of (6.13) over ρ. Then, ρ must satisfy $\frac{qR^2}{2k} - \frac{q\delta^{\frac{2}{2-q}}}{2}\rho^{\frac{-2}{2-q}} = 0$. Thus, the optimal choice is $\rho = \frac{\delta}{R^{2-q}} k^{\frac{2-q}{2}}$. Finally, fixing the number of iterations k and replacing this expression in equation (6.13), we get:

$$f(\hat{x}_k) - f^* \leq \frac{LR^2}{2k} + \delta\frac{(2+q)R^q}{2k^{\frac{q}{2}}}.$$

One can notice that our rate in function values is of order $\mathcal{O}(k^{-1} + \delta k^{-\frac{q}{2}})$, while in [115] the rate is of order $\mathcal{O}(k^{-1} + \delta)$. Hence, when $q > 0$, regardless of the accuracy of the oracle, our second term diminishes, while in [115] it remains constant. Hence, our new definition of inexact oracle of degree q, Definition 6.3.5, is also beneficial in the convex case when analysing proximal gradient type methods, i.e., large q yields better rates.

We also explore an extension of the fast inexact projected gradient method from [115], where the projection step is replaced by a proximal step with respect to the function h, as described in [8]. This extended method is referred to as FI-PGM. It's important to note that the inexactness in FI-PGM arises from the approximate computation of the (sub)gradient of F, specifically $g_{\delta_k, L_k, q}(x_k)$, as defined in Definition 6.3.5. Let $(\theta_k)_{k\geq 0}$ represent a sequence that satisfies the following conditions:

$$\theta_0 \in (0,1], \qquad \frac{\theta_{k+1}^2}{L_{k+1}} \leq A_{k+1} := \sum_{i=0}^{k+1} \frac{\theta_i}{L_i} \quad \forall k \geq 0. \tag{6.14}$$

Then, the fast inexact proximal gradient method (FI-PGM) is as follows:

Algorithm 8 Fast inexact proximal gradient method (FI-PGM)

1. Given $x_0 \in \operatorname{dom} h$, $\theta_0 \in (0,1]$ and $0 \leq q < 2$.

   For $k \geq 0$ do:

2. Choose δ_k, L_k and α_k. Obtain $g_{\delta_k, L_k, q}(x_k)$.
3. Compute $y_k = \operatorname{prox}_{\alpha_k h}(x_k - \alpha_k g_{\delta_k, L_k, q}(x_k))$.
4. Compute $z_k = \arg\min \frac{1}{2}\|x - x_0\|^2 + \sum_{i=0}^{k} \frac{\theta_i}{L_i}\langle g_{\delta_k, L_k, q}(x_i), x - x_i\rangle + h(x)$.
5. Choose θ_{k+1} satisfying condition (6.14) and compute $A_{k+1} = \sum_{i=0}^{k+1} \frac{\theta_i}{L_i}$.
6. Compute $x_{k+1} = \tau_k y_k + (1-\tau_k) z_k$ using $\tau_k = \frac{\theta_{k+1}}{A_{k+1} L_{k+1}}$.

Using a similar proof as in [115], we get the following convergence rate for FI-PGM algorithm:

Corollary 6.3.7. *Let F satisfy the assumptions of Lemma 6.3.6 and $(y_k)_{k\geq 0}$ be generated by FI-PGM. Then, for all $\rho > 0$, we have the following rate:*

$$f(y_k) - f^* \leq \frac{4(L+q\rho)R^2}{(k+1)(k+2)} + \frac{(k+3)(2-q)\delta^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}}. \tag{6.15}$$

Proof. The proof follows from (6.12) and Theorem 4 in [115]. ∎

The optimal ρ in the right hand side of inequality (6.15) is

$$\rho^* = \frac{\big((k+1)(k+2)(k+3)\big)^{\frac{2-q}{2}}}{(8R^2)^{\frac{2-q}{2}}}\delta.$$

Further, replacing ρ with its optimal value in the inequality (6.15), we get

$$\begin{aligned} f(y_k) - f^* &\leq \frac{4LR^2}{(k+1)(k+2)} + \frac{q8^{\frac{q}{2}}R^q(k+3)}{2((k+1)(k+2)(k+3))^{\frac{q}{2}}}\delta + \frac{(2-q)8^{\frac{q}{2}}R^q(k+3)}{2((k+1)(k+2)(k+3))^{\frac{q}{2}}}\delta \\ &= \frac{4LR^2}{(k+1)(k+2)} + \frac{8^{\frac{q}{2}}R^q(k+3)}{((k+1)(k+2)(k+3))^{\frac{q}{2}}}\delta. \\ &= \mathcal{O}\left(\frac{LR^2}{k^2}\right) + \mathcal{O}\left(\frac{R^q}{k^{\frac{3q}{2}-1}}\delta\right). \end{aligned}$$

Hence, if $q > \frac{2}{3}$, then FI-PGM doesn't have error accumulation under our inexact oracle as the rate is of order $\mathcal{O}\left(k^{-2} + \delta k^{1-\frac{3q}{2}}\right)$, while in [115] the FI-PGM scheme always displays error accumulation, as the convergence rate is of order $\mathcal{O}(k^{-2} + \delta k)$. Therefore, the same conclusion holds as for I-PGM, i.e., for the FI-PGM scheme in the convex setting it is beneficial to have an inexact first-order oracle with large degree q.

Remark 21. In our Definition 6.2.1 we have considered exact zero-order information. However, it is possible to change this definition considering also inexact zero-order information for the nonconvex case. More precisely, we can change Definition 6.2.1 as follows

$$\begin{cases} F_{\delta_0}(x) - F(x) \leq \delta_0, \\ F(x) - (F_{\delta_0}(y) + \langle g_{\delta,L,q}(y), x-y\rangle) \leq \frac{L}{2}\|x-y\|^2 + \delta\|x-y\|^q. \end{cases}$$

With this new definition, the convergence result in Theorem 4.3.4 becomes:

$$\sum_{j=0}^{k} \alpha_j \|g_{\delta_j,L_j,q}(x_j) + p_{j+1}\|^2 \leq f(x_0) - f_\infty + \frac{\sum_{j=0}^{k}(2-q)\delta_j^{\frac{2}{2-q}}}{2\rho^{\frac{q}{2-q}}} + \sum_{j=0}^{k}\delta_0.$$

Hence the rate in this case is also influenced by the inexactness of the zero-order information (i.e., δ_0). Note that for the convex case, the previous extension is not possible in Definition 6.3.5 when $q > 0$, since we must have:

$$0 \leq F(x) - (F_{\delta_0}(y) + \langle g_{\delta,L,q}(y), x-y\rangle) \leq \frac{L}{2}\|x-y\|^2 + \delta\|x-y\|^q,$$

which implies for $x = y$ that $F(x) = F_{\delta_0}(x)$. Since we want to have consistency between Definitions 6.2.1 and 6.3.5, we have chosen to work with the exact zero-order information in our previous nonconvex convergence analysis.

6.4 Numerical simulations

In this section, we evaluate the performance of I-PGM for a composite problem arising in image restoration. Namely, we consider the following nonconvex optimization problem [123]:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} \sum_{i=1}^{N} \log\left(\left(a_i^T x - b_i\right)^2 + 1\right), \\ &\text{s.t. } \|x\|_1 \leq R, \end{aligned} \tag{6.16}$$

where $R > 0$, $b \in \mathbb{R}^N$ and $a_i \in \mathbb{R}^n$, for $i = 1 : N$. In image restoration, b represents the noisy blurred image and $A = (a_1, \cdots, a_N) \in \mathbb{R}^{n\times N}$ is a blur operator [123]. This problem fits into our general problem (6.1), with $F(x) = \sum_{i=1}^{N} \log\left(\left(a_i^T x - b_i\right)^2 + 1\right)$, which is a nonconvex function with Lipschitz continuous gradient of constant $L_F := \sum_{i=1}^{N} \|a_i\|^2$, and $h(x)$ is the indicator function of the bounded convex set $\{x : \|x\|_1 \leq R\}$. We generate the inexact oracle by adding normally distributed random noise δ to the true gradient, i.e., $g_{\delta,L,q}(x) := \nabla F(x) + \delta$. This is a particular case of Example 6.2.2. However, for all x and y satisfying $\|x\| \leq R$, $\|y\| \leq R$, we have the following:

$$\delta\|x-y\| = \delta\|x-y\|^{1-q}\|x-y\|^q \leq \delta(2R)^{1-q}\|x-y\|^q.$$

Thus, this example satisfies Definition 6.2.1 for all $q \in [0, 1]$. We apply I-PGM for this particular example where we consider three choices for the degree q: 0, 1/2 and 1. Recall that the convergence rate of I-PGM with constant step size is (see Corollary 6.3.3, first statement):

$$\min_{j=0:k} \|g_j + p_{j+1}\|^2 \leq \frac{2(q+1)L(f(x_0) - f^*)}{k+1} + (q+1)(2-q)L^{\frac{2-2q}{2-q}}\delta^{\frac{2}{2-q}}. \tag{6.17}$$

At each iteration of I-PGM we need to solve the following convex subproblem:

$$\min_{x\in\mathbb{R}^n} F(x_k) + \langle g_{\delta,q}(x_k), x - x_k\rangle + \frac{L+q\rho}{2}\|x - x_k\|^2, \text{ s.t. } \|x\|_1 \leq R.$$

This subproblem has a closed form solution (see e.g., [124]). We compare I-PGM with constants step size $\alpha_k = \frac{1}{2(L_F+q\rho)}$ and $\rho = L_F$ for three choices of $q = 0, 1/2, 1$ and three choices of noise norm $\|\delta\| \leq 0.1, 1, 3$, respectively. The results are given in Figure 6.4 (dotted lines), where we plot the evolution of the error $\min_{j=0:k} \|\frac{1}{\alpha_k}(x_{j+1} - x_j)\|^2$, which corresponds to the gradient mapping. In the same figure we also plot the theoretical bounds (6.17) for $q = 0, 1/2, 1$ (full lines). Our main figures are Figure 6.1, 6.3, and 6.4, while Figure 6.2 is a subfigure (zoom) of Figure 6.1, displaying only the first 300 iterations. Moreover, one can see in these main figures (i.e., Figure 6.1, 6.3, and 6.4 that the behaviour of our algorithm for $q = 1$ is better than for $q = 1/2$. Similarly, the behaviour of our algorithm for $q = 1/2$ is better than for $q = 0$. One can observe these better behaviours after 300 iterations when the error δ is small (see Figure 6.3 and 6.4). However, when the error δ is large, we need to perform a larger number of iterations before we can observe these behaviours, (see Figure 6.1 and 6.2). This is natural, since large errors on the gradient approximation must have impact on the convergence speed. Hence, as the degree q increases or the norm of the noise decreases, better accuracies for the norm of the gradient mapping can be achieved, which supports our theoretical findings.

Moreover, from the numerical simulations, one can observe that the gap between the theoretical and the practical bounds is large in Figure 6.3 and 6.4. We believe that this happens because, in the convergence analysis, the theoretical bounds are derived under worst-case scenarios (i.e., the convergence analysis must account for the worst case direction generated by the inexact first-order oracle, while in practical implementations, which often involve randomness, one usually doesn't encounter these worst-case directions). However, the simulations in Figure 6.1 show that the gap between the theoretical bounds and the practical behavior is not too large. More precisely, we have generated at each iteration 100 random directions and, in order to update the new point, we have chosen the worst direction with respect to the gradient mapping (i.e., the largest) $\|x_{k+1} - x_k\|$). The results are given in Figure 6.1, where one can see that the theoretical and practical bounds are getting closer for sufficiently large number of iterations.

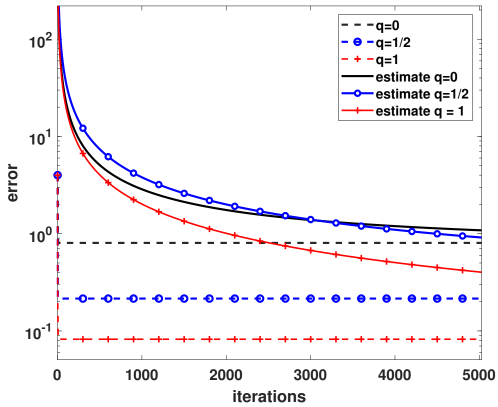

Figure 6.1: $\|\delta\| \leq 3$

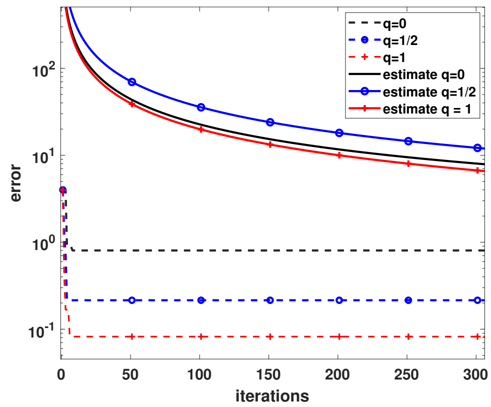

Figure 6.2: $\|\delta\| \leq 3$

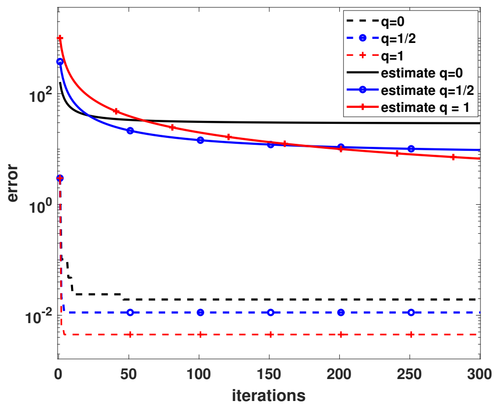

Figure 6.3: $\|\delta\| \leq 1$

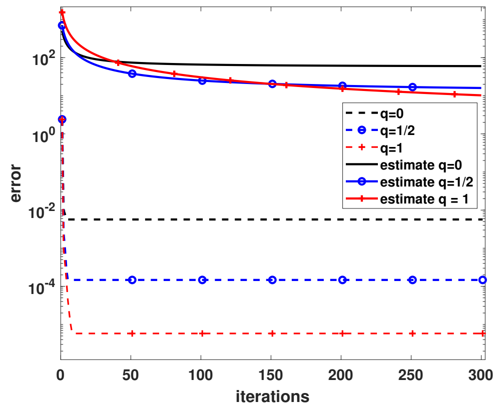

Figure 6.4: $\|\delta\| \leq 0.1$

Figure 6.5: Practical (dotted lines) and theoretical (full lines) performances of the I-PGM algorithm for different choices of q and δ, with $R = 4$. Figure (b) represents a zoom of the left corner from Figure (a).

6.5 Conclusions

In this chapter introduce the concept of inexact first-order oracle of degree q for a possibly nonconvex and nonsmooth function, which naturally appears in the context of approximate gradient, weak level of smoothness and other situations. Our definition is less conservative than those found in the existing literature, and it can be viewed as an interpolation between fully exact and the existing inexact first-order oracle definitions. We analyze the convergence behavior of an inexact proximal gradient method using such an oracle for solving (non)convex composite minimization problems. We derive complexity estimates and study the dependence between the accuracy of the oracle and the desired accuracy of the gradient or of the objective function. Our results show that better rates can be obtained both theoretically and in numerical simulations when q is large.

7 Applications

In this chapter, we test our algorithms from the previous chapters on a range of applications, including power flow analysis, phase retrieval and output feedback control problems, using real data. These applications underscore the versatility and practical impact of our optimization methods across multiple domains.

The chapter is structured as follows: Section 7.1 presents our first case study, involving power systems, showcasing how our algorithms can be used and applied to solve the static power flow problem. Next, in Section 7.2, we turn to the phase retrieval, a crucial problem in fields like optics and signal processing, where our algorithms excel at reconstructing signals with minimal error and computational overhead. Finally, Section 7.3 is devoted to the study of control systems and we illustrate how our numerical optimization algorithms can be applied in solving hard output feedback control problems.

7.1 Power flow analysis

The reliable delivery of electrical energy is a cornerstone of modern society, underpinning the operation of critical infrastructure, industries, and households. To maintain a stable and efficient power system, engineers must understand how electrical power flows through the network. This understanding is the essence of power flow analysis, a foundational technique in power systems engineering.

Power flow analysis, sometimes referred to as load flow analysis, is used to evaluate the distribution of electrical power within a network, ensuring it meets the required demand and operational standards. It involves calculating the voltages at various points in the system and the corresponding active (real) and reactive power flows. These calculations provide insights into the network's performance, allowing for effective management, planning, and optimization of the power system.

In a typical power system, buses represent nodes where components like generators, loads, and transformers are connected, while lines represent the electrical connections between these nodes. The power flow problem involves determining the voltage magnitudes and angles at each bus, given a specific set of known values for power generation, load, and the system's topology. This information is used to ensure that the system operates within safe limits, to identify power losses, and to guide decisions related to system expansion and reconfiguration.

A power system with N buses requires solving a set of nonlinear equations derived from Kirchhoff's laws and network admittances. The solution provides a complete picture of the system's behavior, including voltage levels, active and reactive power generation and consumption, and power flows across transmission lines.

Consider a steady-state power system with N buses (see e.g., Figure 7.1 for the IEEE 14 bus system). This implies that the voltages, active and reactive powers are assumed to be constant and not time-dependent. We denote v_i, p_i and q_i the complex voltage, active power and reactive power for the i bus, respectively. Let $Y := G + jB$ be the admittance matrix and denote $p =$

$(p_1, \cdots, p_N)$, $q = (q_1, \cdots, q_N)$ and $v = (v_1, \cdots, v_N)$. Given a complex load vector $s := s_R + js_I$, then the power flow analysis problem is to find $v = (v_1, \cdots, v_N)$ such that [29]:

$$F(v) = s, \quad F(v) = p + jq = \text{diag}(vv^H Y^H), \tag{7.1}$$

where recall that $(\cdot)^H$ is the Hermitian transpose and v is described by its magnitude u and its phase θ. Finding the complex voltage v is a critical parameter in bus system analysis and operation, influencing load allocation, equipment performance, system stability, and network planning. Proper voltage management is essential for maintaining a reliable and efficient electrical distribution system. This problem is equivalent to the following optimization problem:

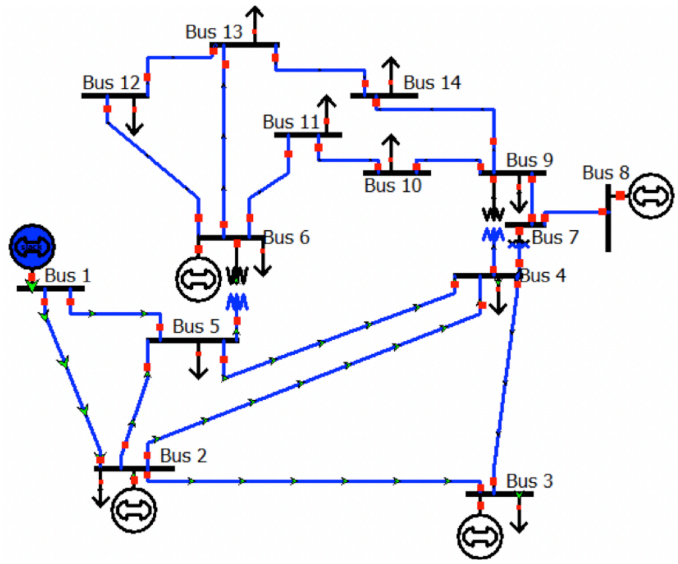

Figure 7.1: Representation of the IEEE 14-bus system [36].

$$\begin{aligned} &\min_{v=(u,\theta)} \|F(v) - s\| \\ &s.t. \quad u \in [u_{\min}, u_{\max}], \quad \theta \in [-\pi, \pi]. \end{aligned}$$

In [29], the authors provide a similar formulation for the power flow analysis problem, but using $\|\cdot\|^2$ as the merit function to measure the distance between the objective function $F(\cdot)$ and the desired complex load s. As we have mentioned earlier (see Section 5.1), it is beneficial to use only $\|\cdot\|$ as the merit function. Further, since we have (see e.g., [125]):

$$\begin{aligned} p_i(u,\theta) &= \sum_{k=1}^{N} u_i u_k \left(G(i,k)\cos(\theta_i - \theta_k) + B(i,k)\sin(\theta_i - \theta_k)\right), \\ q_i(u,\theta) &= -\sum_{k=1}^{N} u_i u_k (B(i,k)\cos(\theta_i - \theta_k) + G(i,k)\sin(\theta_i - \theta_k)), \end{aligned}$$

and using the notation

$$\mathbf{C} = \{(u,\theta) : u \in [u_{\min}, u_{\max}], \quad \theta \in [-\pi, \pi]\},$$

then, the previous optimization problem is equivalent to the following optimization problem:

$$\min_{x=(u;\theta)\in\mathbf{C}} f(x) = \left\| \begin{matrix} p(x) - s_R \\ q(x) - s_I \end{matrix} \right\|. \tag{7.2}$$

The most efficient algorithm for solving the (unconstrained) power flow analysis problem is the Newton method [126]. However, it may lead to poor performance when the initialization point is far from the optimum or the system is stressed (i.e., the problem is ill-conditioned). In a recent paper, [29], the authors proposed a hybrid method that combines stochastic gradient descent (SGD) and the Newton methods to overcome the numerical challenges in this problem. The iterative process starts with the Newton algorithm, and if the method detect a divergence (e.g., when the condition number of the Jacobian deteriorates), then switch to the SGD algorithm. After running a few SGD steps, switch again to the Newton iterations and repeat the process until an (approximate) optimal solution is found. Since this hybrid algorithm cannot deal with (simple) constraints as in (7.2), we propose to use our new method, RHOTA for $p = 1$ from Chapter 5, and compare its performance with the projected gradient descent (PGD) method applied to the problem:

$$\min_{x \in \mathbb{R}^n} \|F(x)\|^2 + I_{\mathbf{C}}(x), \tag{7.3}$$

More precisely, the iterates of the PGD method are of the form [94]:

$$x_{k+1} = \text{proj}_C \left(x_k - \frac{1}{L} \nabla F(x_k)^T F(x_k) \right).$$

In order to apply both methods, one needs to evaluate the gradient of the functions $p(x)$ and $q(x)$. We have the following expressions for the derivatives of p_i's and q_i's:

$$\begin{aligned}
\frac{\partial p_i}{\partial u_i} &= 2G(i,i) + \sum_{\substack{k=1 \\ k \neq i}}^{N} u_k \left(G(i,k)\cos(\theta_i - \theta_k) + B(i,k)\sin(\theta_i - \theta_k) \right), \\
\frac{\partial p_i}{\partial u_k} &= u_i \left(G(i,k)\cos(\theta_i - \theta_k) + B(i,k)\sin(\theta_i - \theta_k) \right), \forall k \neq i, \\
\frac{\partial p_i}{\partial \theta_i} &= \sum_{\substack{k=1 \\ k \neq i}}^{N} u_k u_i \left(-G(i,k)\sin(\theta_i - \theta_k) + B(i,k)\cos(\theta_i - \theta_k) \right) \\
\frac{\partial p_i}{\partial \theta_k} &= -u_i u_k \left(-B(i,k)\cos(\theta_i - \theta_k) - G(i,k)\sin(\theta_i - \theta_k) \right), \forall k \neq i, \\
\frac{\partial q_i}{\partial u_i} &= -2B(i,i) - \sum_{\substack{k=1 \\ k \neq i}}^{N} u_k \left(B(i,k)\cos(\theta_i - \theta_k) - G(i,k)\sin(\theta_i - \theta_k) \right), \\
\frac{\partial q_i}{\partial u_k} &= -u_i \left(B(i,k)\cos(\theta_i - \theta_k) - G(i,k)\sin(\theta_i - \theta_k) \right), \forall k \neq i, \\
\frac{\partial q_i}{\partial \theta_i} &= \sum_{\substack{k=1 \\ k \neq i}}^{N} u_k u_i \left(B(i,k)\sin(\theta_i - \theta_k) + G(i,k)\cos(\theta_i - \theta_k) \right), \\
\frac{\partial q_i}{\partial \theta_k} &= -u_k u_i \left(G(i,k)\cos(\theta_i - \theta_k) + B(i,k)\sin(\theta_i - \theta_k) \right), \forall k \neq i.
\end{aligned}$$

Hence, $\nabla F(x) \in \mathbb{R}^{2N}$ and we have:

$$\nabla F(x) = \sum_{i=1}^{N} \frac{\partial p_i(x)}{\partial x} (p_i(x) - s_R) + \frac{\partial q_i(x)}{\partial x} (q_i(x) - s_I),$$

where:

$$\frac{\partial p_i(x)}{\partial x} = \left(\frac{\partial p_i(x)}{\partial u_1}; \cdots; \frac{\partial p_i(x)}{\partial u_N}; \frac{\partial p_i(x)}{\partial \theta_1}; \cdots; \frac{\partial p_i(x)}{\partial \theta_N} \right)$$

$$\frac{\partial q_i(x)}{\partial x} = \left(\frac{\partial q_i(x)}{\partial u_1}; \cdots ; \frac{\partial q_i(x)}{\partial u_N}; \frac{\partial q_i(x)}{\partial \theta_1}; \cdots ; \frac{\partial q_i(x)}{\partial \theta_N} \right),$$

for $i = 1 : N$. Note that the Jacobian ∇F may be ill-conditioned.

Below, we illustrate the efficacy of RHOTA algorithm for $p = 1$, as detailed in Chapter 5, comparing with PGD algorithm, utilizing multiple IEEE bus test cases from [36] (specifically, IEEE 14 bus, IEEE 39 bus, IEEE 57 bus, and IEEE 118 bus). We select an optimal point $x^* \in \mathbf{C}$, from which we generate $s_R = p(x^*)$ and $s_I = q(x^*)$ (also refer to [29]). Subsequently, we apply RHOTA algorithm (for $p = 1$) to solve problem (7.2) and the PGD method to tackle problem (7.3), where F is defined in (7.1). We then evaluate whether these algorithms can converge to x^* from a feasible starting point. Both algorithms utilize a stopping criterion of $\|F(x_k)\| \leq 10^{-3}$. The results are depicted in Figure (7.2), illustrating the evolution of the function value $\|F(x_k)\|$ across iterations.

The plotted data indicates that initially, PGD method outperforms RHOTA method. However, RHOTA method exhibits a significantly quicker convergence, requiring fewer iterations (up to five times fewer) than the PGD method to attain the desired accuracy.

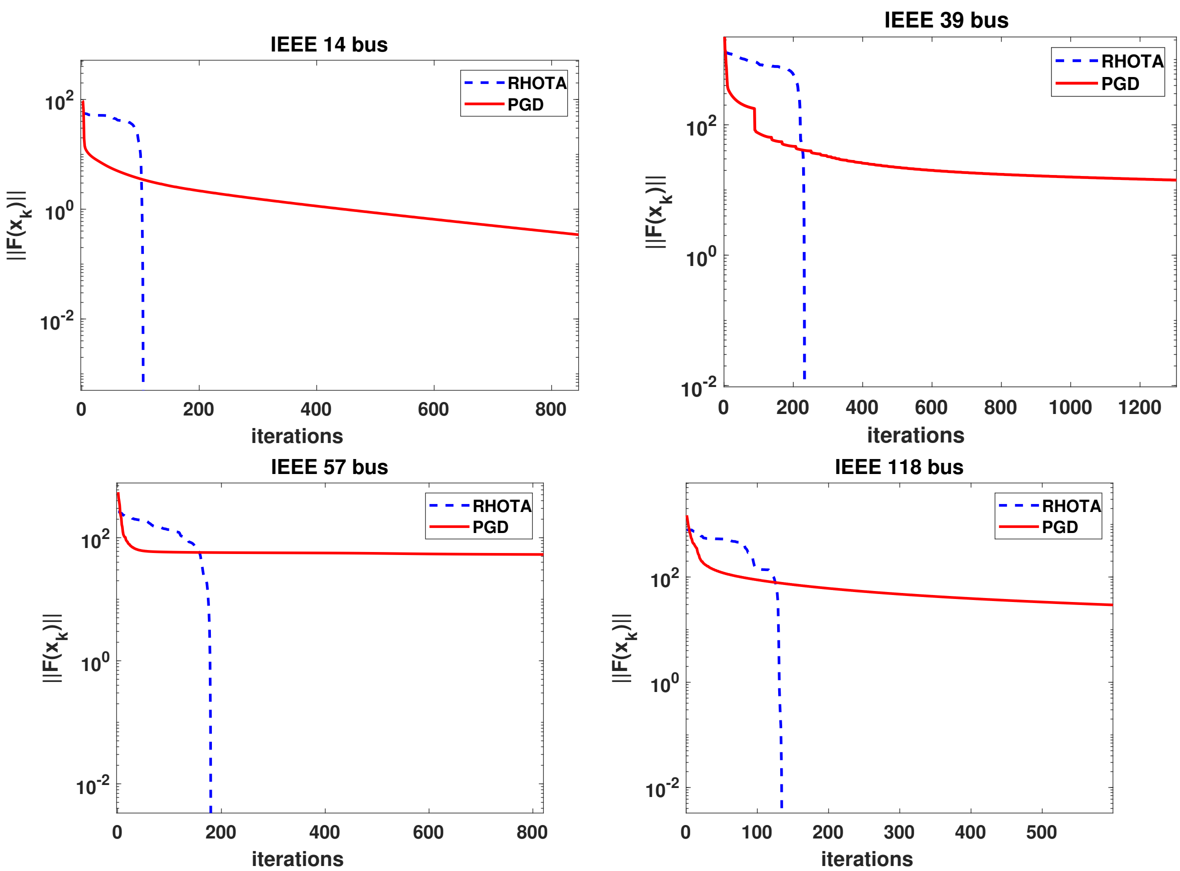

Figure 7.2: Comparison between RHOTA (for $p = 1$) and PGD methods in terms of $\|F(x)\|$ along iterations on several IEEE bus systems.

7.2 Phase retrieval

In various scientific and engineering fields, the ability to reconstruct a signal or image from incomplete or indirect measurements is a fundamental challenge. One key problem in this area is phase retrieval [127, 128], where the goal is to determine a complex signal from its magnitude measurements, often derived from its Fourier transform or other linear transformations [129].

This problem is common in optics [130], crystallography [131], astronomical imaging [132], speech processing [133], computational biology [134] among other domains.

In many applications, direct measurement of complex signals is challenging due to technical limitations, measurement noise, or physical constraints. For example, in X-ray crystallography [131], only the magnitudes of the diffracted X-rays can be measured, while the phase information is lost. Similarly, in optical systems [130], detectors often record the intensity of light without direct access to its phase. Phase retrieval provides a means to reconstruct the missing phase information, allowing the complete signal or image to be recovered.

The phase retrieval problem is inherently ill-posed, as multiple phase configurations can lead to the same magnitude measurements. This ambiguity, combined with the nonlinear relationship between phase and magnitude, makes phase retrieval a complex and challenging task. Researchers have developed various algorithms and techniques to address these difficulties, ranging from iterative optimization methods to deep learning approaches. Figure 7.3 illustrate the important of the Fourier phase.

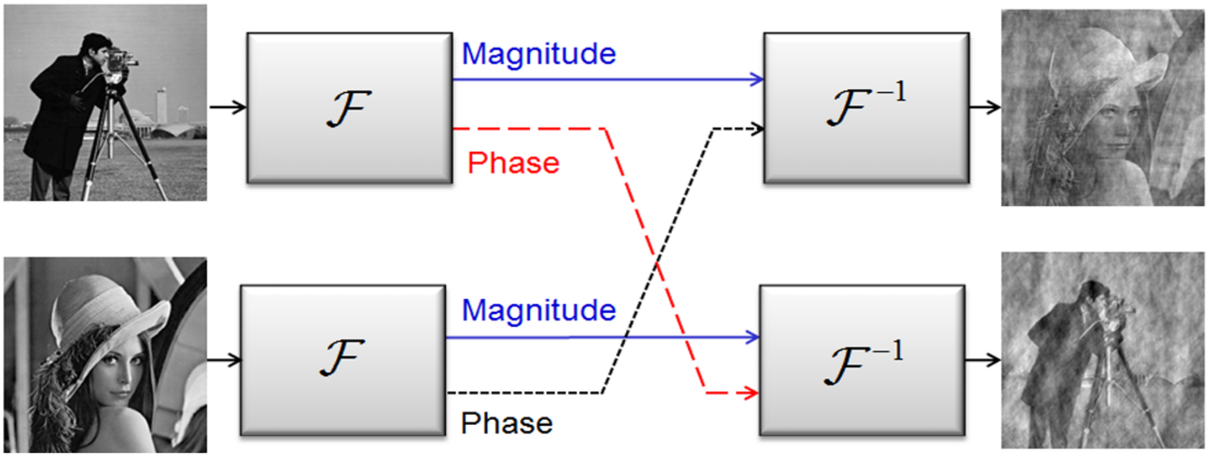

Figure 7.3: The Significance of Fourier Phase: Two images, cameraman and Lenna, are subjected to a Fourier transform. After swapping the phase information between these transformed images, they are then inverse Fourier transformed back to the spatial domain. The resulting images reveal that the swapped phase information leads to an unexpected recombination, highlighting the critical role that phase plays in the recovery and reconstruction of images. This image is taken from [129].

The objective of phase retrieval is to reconstruct the original signal from its magnitude measurements, which can be mathematically stated as [30]:

$$\text{Find } x \in \mathbb{R}^n \text{ (or } \mathbb{C}^n) \quad \text{s.t.}: \quad z_i = |\langle a_i, x\rangle|^2, \ i = 1:m, \tag{7.4}$$

where $a_i \in \mathbb{R}^n$ (or $\mathbb{C}^n$) represents a known measurement vector and z_i denotes a known magnitude, for $i = 1 : m$. When $a, x \in \mathbb{C}^n$, $\langle a, x\rangle := x^H a$, with x^H the conjugate transpose of x. Various approaches have been explored to tackle phase retrieval, with recent research focusing on non-convex methods. For instance, in [30], the authors introduce the Wirtinger flow, a gradient-based method that performs gradient descent on the objective function:

$$x \mapsto \frac{1}{2m}\sum_{i=1}^{m}\left(|\langle a_i, x\rangle|^2 - z_i\right)^2.$$

Similarly, [32] proposes a modified objective and applies a gradient descent method to:

$$x \mapsto \frac{1}{2m} \sum_{i=1}^{m} \left(|\langle a_i, x \rangle| - \sqrt{z_i} \right)^2.$$

Both approaches in [30, 32] rigorously demonstrate the exact retrieval of phase information from a nearly minimal number of random measurements, achieved through careful initialization using spectral methods. In a recent study [31], the authors address phase retrieval using a nonsmooth l_1 norm formulation:

$$x \mapsto \frac{1}{2m} \sum_{i=1}^{m} \left| |\langle a_i, x \rangle|^2 - z_i \right| = \frac{1}{2m} \left\| x^H Q x - z \right\|_1,$$

where $x^H Q x = (x^H Q_1 x, \ldots, x^H Q_m x)^T$ and $Q_i = a_i a_i^H$ for $i = 1 : m$. This problem formulation can be expressed as a composition $f(x) = g(F(x))$. To address this nonsmooth minimization problem, [31] proposes a prox-linear method (equivalent to the RHOTA algorithm presented in this paper for $p = 1$). The signal recovery in their procedure requires a stability condition that is typically satisfied with a high probability for suitable designs Q, a bound on the operator $\|Q\|^2$, and a well-initialized iterative process. The prox-linear method exhibits quadratic convergence and achieves exact signal recovery if the number of measurements satisfies $m = 2n$. In the following, we consider a_i, x in $\mathbb{R}^n$ (note that if these quantities are complex vectors, then by a proper change of variables the problem can be formulated over real vectors). In this paper, inspired by [31], we consider the following nonsmooth composite minimization problem:

$$\min_{x \in \mathbb{R}^n} f(x) := \|x^T Q x - z\|, \tag{7.5}$$

where $x^T Q x := \left(x^T Q_1 x, \cdots, x^T Q_m x\right)^T$ and $Q_i = a_i a_i^T$ for $i = 1 : m$. In the sequel, we present a higher-order proximal point algorithm (called HOPP) for solving this type of problems and then proceed to analyze its convergence rate and efficiency.

Algorithm 9 HOPP Algorithm

Given x_0, positive integer p and $M > 0$. For $k \geq 0$ do:

Compute x_{k+1} solution of the following subproblem:

$$x_{k+1} \in \arg\min_{x \in \mathbb{R}^n} \left\|x^T Q x - z\right\| + \frac{M}{p+1} \|x - x_k\|^{p+1}. \tag{7.6}$$

The "argmin" in (7.6) refers to the set of global minimizers. Higher-order proximal point algorithms have been considered recently in the literature. Indeed, the convergence rates have been extensively studied, with works such as [135] focusing on the convex case and [20] investigating the nonconvex scenarios. Notably, the objective in (7.5) is weakly convex with $L := 2\|(\|Q_1\|, \cdots, \|Q_m\|)\|$, as established in [56]. Therefore, for $p = 1$, the subproblem (7.6) becomes strongly convex when $M > L$. However, if the constant $M < L$, one cannot ensure the convexity of the subproblem (7.6) for $p = 1$. In the sequel, we show that when L is difficult to compute, one can still employ convex optimization tools to solve the subproblem (7.6) for any $M > 0$ and $p = 1$ or $p = 2$. Indeed, following the same reasoning as in Section 5.4, the *global solution* of the (non)convex proximal subproblem (7.6) for $p = 1$ is:

$$x_{k+1} = x_k - H_{k,1}(u)^{-1} g_k(u),$$

where we denote:

$$H_{k,1}(u) := \sum_{i=1}^{m} 2u_i Q_i + \frac{M}{2} I_n,\; g_k(u) := \sum_{i=1}^{m} 2u_i Q_i x_k,\; l_k(u) := \sum_{i=1}^{m} u_i (x_k^T Q_i x_k - z_i),$$

with u representing the solution of the convex dual problem:

$$\max_{u \in \mathcal{F}_1} l_k(u) - \frac{1}{2} \left\langle H_{k,1}(u)^{-1} g_k(u), g_k(u) \right\rangle,$$

where $\mathcal{F}_1 := \{u \in \mathbb{R}^m : \|u\| \leq 1 \text{ and } H_{k,1}(u) \succ 0\}$.

Similarly, for $p = 2$, we have:

$$x_{k+1} = x_k - H_{k,2}(u,w)^{-1} g_k(u),$$

where $H_{k,2}(u,w) := \sum_{i=1}^{m} 2u_i Q_i + \frac{w}{2} I_n$, with (u,w) is the solution of the following convex dual problem:

$$\max_{(u,w) \in \mathcal{F}_2} \; l_k(u) - \frac{1}{2} \left\langle H_{k,2}(u,w)^{-1} g_k(u), g_k(u) \right\rangle - \frac{1}{12M^2} w^3,$$

where $\mathcal{F}_2 := \{(u,w) \in \mathbb{R}^m \times \mathbb{R}_+ : \|u\| \leq 1 \text{ and } H_{k,2}(u,w) \succ 0\}$. Hence, our algorithm HOPP can be easily implemented for any $M > 0$ and $p = 1, 2$ using standard convex optimization tools (such as interior point methods [66]). Next, we establish the global convergence rate to a stationary point for this algorithm:

Theorem 7.2.1. *Let f be given as in (7.5) and let $(x_x)_{k \geq 0}$ be generated by HOPP algorithm. Then, we have:*

$$\min_{i=0:k} S_f(x_i) \leq \left(\frac{(M(p+1)^p)^{\frac{1}{p+1}} (f(x_0) - f^*)}{k^{\frac{p}{p+1}}} \right).$$

Proof. From the definition of x_{k+1} in (7.6), we get:

$$f(x_{k+1}) + \frac{M}{p+1} \|x_{k+1} - x_k\|^{p+1} \leq f(x_k) \;\text{ and }\; S_f(x_{k+1}) \leq M \|x_{k+1} - x_k\|^p.$$

Hence, combining the last two inequalities, we get:

$$S_f(x_{k+1})^{\frac{p+1}{p}} \leq M^{\frac{p+1}{p}} \frac{p+1}{M} \big(f(x_k) - f(x_{k+1}) \big).$$

Summing up and taking the minimum, we get our statement. ■

In order to establish rapid local convergence, we introduce an additional assumption that is related to sharpness or error bound condition for the objective function, as discussed in [31].

Assumption 7.2.2. There exists $\lambda > 0$ such that for all $x \in \mathbb{R}^n$ the objective function f defined in (7.5), having the set of global minima X^*, satisfies:

$$f(x) - f(x^*) \geq \sigma_0 \operatorname{dist}(0, X^*) \operatorname{dist}(x, X^*) \quad \forall x^* \in X^*, \;\text{ with } \sigma_0 > 0.$$

This condition has been proved to hold in the context of phase retrieval, see [31]. For instance, it holds when the matrices Q_i's satisfy the following stability condition [31]:

$$\|(Q_i x)^2 - (Q_i y)^2\| \geq \bar{\sigma}_0 \|x - y\| \, \|x + y\| \quad \forall x, y \in \mathbb{R}^n, \; i = 1 : m, \;\; \text{with } \bar{\sigma}_0 > 0.$$

Next, we derive a fast convergence rate for HOPP algorithm under sharpness.

Theorem 7.2.3. *Let f be defined as in (7.5) and satisfy Assumption 7.2.2. Moreover, let $(x_k)_{k\geq 0}$ be generated by HOPP algorithm. Then, we have:*

$$\frac{\text{dist}(x_k, X^*)}{\text{dist}(0, X^*)} \leq \left(\frac{\sigma_0(p+1)}{M \text{dist}(0, X^*)^{p-1}} \right)^{\frac{1}{p}} \left(\frac{M^{\frac{1}{p}} \text{ dist}(x_0, X^*)}{(\sigma_0(p+1)\text{dist}(0, X^*))^{\frac{1}{p}}} \right)^{(p+1)^k}.$$

Proof. Since x_{k+1} is the global minimum of (7.6), we have:

$$f(x_{k+1}) \leq \min_{x \in \mathbb{R}^n} f(x) + \frac{M}{p+1}\|x - x_k\|^{p+1} \leq f(x^*) + \frac{M}{p+1}\|x^* - x_k\|^{p+1}.$$

Taking the infimum over $x^* \in X^*$, we further obtain:

$$f(x_{k+1}) - f(x^*) \leq \frac{M}{p+1} \text{dist}(x_k, X^*)^{p+1}.$$

Combining this inequality with Assumption 7.2.2, we get:

$$\sigma_0(p+1) \text{ dist}(0, X^*) \text{dist}(x_{k+1}, X^*) \leq M \text{dist}(x_k, X^*)^{p+1}.$$

Dividing each side by $\text{dist}(0, X^*)^{p+1}$, we get:

$$\frac{\text{dist}(x_{k+1}, X^*)}{\text{dist}(0, X^*)} \leq \frac{M \text{dist}(0, X^*)^{p-1}}{\sigma_0(p+1)} \left(\frac{\text{dist}(x_k, X^*)}{\text{dist}(0, X^*)} \right)^{p+1}.$$

Unrolling the last recurrence, yields our statement. ∎

Note that if $M\text{dist}(x_0, X^*)^p < \sigma_0(p+1)\text{dist}(0, X^*)$, then faster convergence is guaranteed for HOPP iterates with the increasing value of p. Note also that for $p = 1$, we recover the quadratic convergence rate obtained in [31]. Furthermore, the flexibility in selecting the parameter M is significant: given an arbitrary initial point x_0 (not necessarily close to X^*), an appropriate choice of M (i.e., sufficiently small) guarantees very fast convergence of HOPP iterates to the global minima of (7.5).

Below, we present numerical simulations for solving the phase retrieval problem, using real images from the collection of handwritten digits, accessible at [37]. The primary objective is to evaluate the performance of HOPP method in image recovery and compare it with the prox-linear method introduced in [31]. Given that [31] demonstrates perfect image recovery under real-valued random Gaussian measurements, even when $m = 2 \times n$, we adopt similar settings. Specifically, we evaluate the performance of our method for $p = 1, 2$ and the prox-linear method in [31], aiming to recover a digit image using Gaussian measurement vectors $a_i \in \mathbb{R}^n$ and set $Q_i = a_i^T a_i$ for $i = 1 : m$ with $m = 2 \times n$. To initialize the process, we introduce some noise to the real-digit image to generate the starting point x_0. The stopping criterion for both methods is set as $f(x_k) \leq 10^{-4}$ or $k \geq 100$. Each subproblem is solved using CVX [136].

The results are presented in Figures 7.4 and 7.5. In Figure 7.4, we initialize the starting point x_0 (by adding some noise to the original image) to satisfy the constant relative error guarantee

$$\operatorname{dist}(x_0, x^*) < \frac{\sigma_0}{L}\|x^*\|,$$

with $L = \|\left(\|a_1\|^2, \cdots, \|a_m\|^2\right)\|$, as presented in [31]. From Figure 7.4, it's evident that both algorithms achieve good recovery of the original image, with HOPP ($p = 2$) given the best error. However, HOPP algorithm for $p = 1, 2$ is much faster than the prox-linear algorithm [31].

In Figure 7.5, we set the initial point x_0 randomly, so that it does not satisfy the condition $\operatorname{dist}(x_0, x^*) < \frac{\sigma_0}{L}\|x^*\|$. Notably, Figure 7.5 illustrates that the prox-linear algorithm [31] fails to recover the original image after 100 iterations. In contrast, HOPP algorithm (for both $p = 1$ and $p = 2$) is able to recover the original image for sufficiently small M. This highlights the efficiency and robustness of HOPP algorithm. In cases where the true image x^* is unknown, we posit that the HOPP method's flexibility, that follows from the free choice of the regularization parameter M, allows it to perform effectively even with an initial point that is not necessarily close to the true image. Such an initial point could be generated more affordably than the methods proposed in [31, 30]. Finally, one can notice from Figures 7.4 and 7.5 the considerable time taken by CVX to solve the convex subproblems. Thus, it would be interesting to explore more efficient convex solvers for solving these subproblems. This aspect remains open for further investigations.

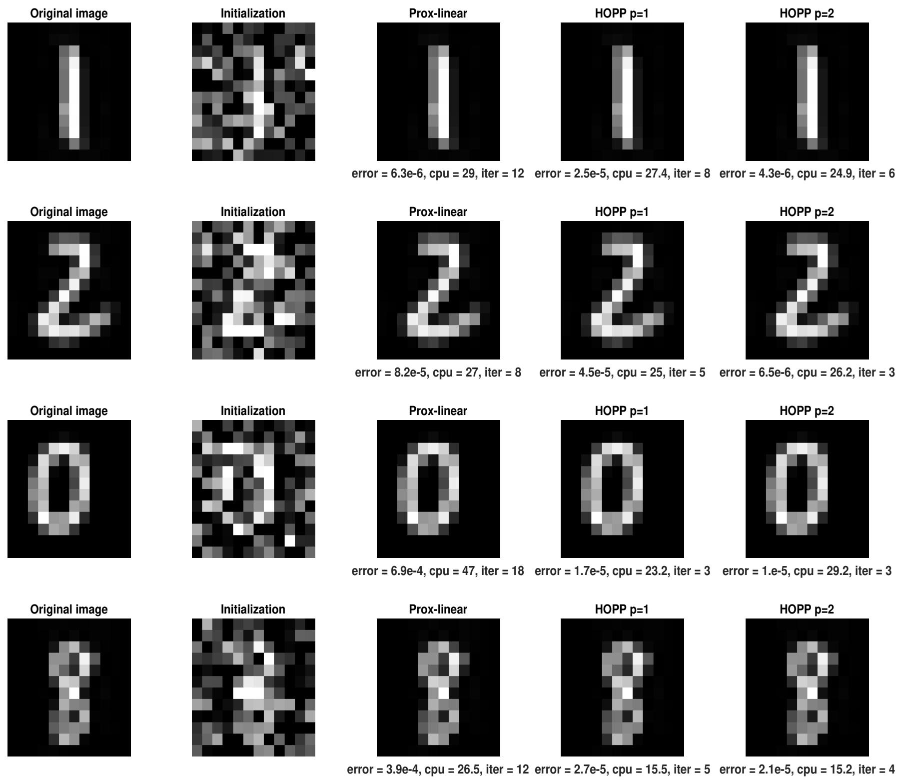

Figure 7.4: Performance of prox-linear method [31] and HOPP for $p = 1$ and $p = 2$ with $M = 0.1$ on 12×12 digit images: initialization satisfying $\operatorname{dist}(x_0, x^*) < \|x^*\|\sigma_0/L$.

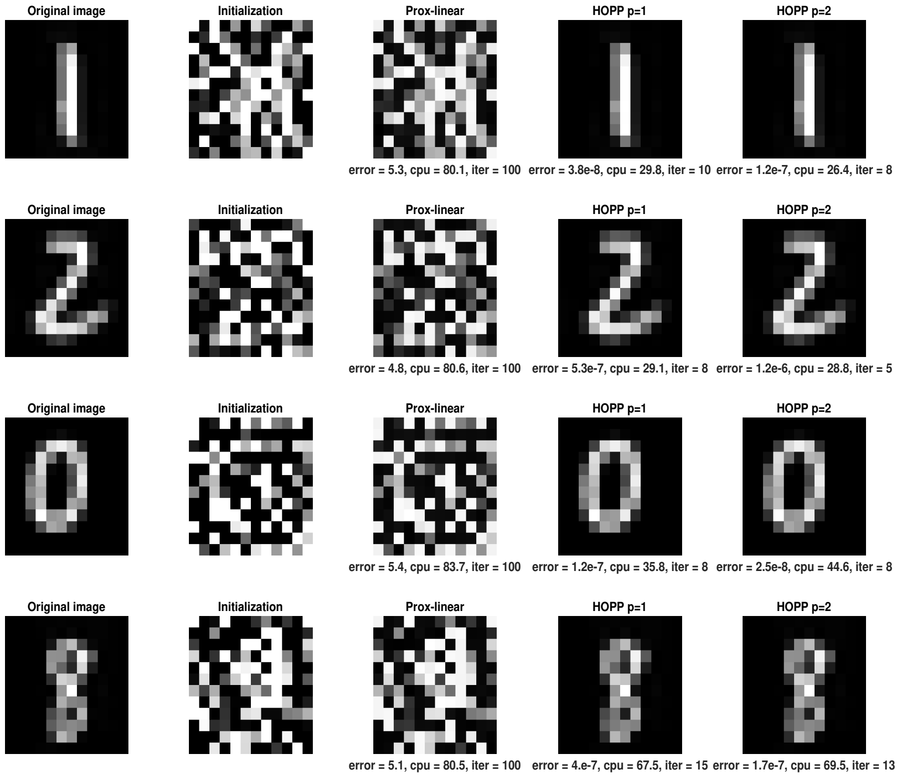

Figure 7.5: Performance of prox-linear method [31] and HOPP for $p = 1$ and $p = 2$, with $M = 0.01$ on 12×12 digit images: random initialization x_0.

7.3 Static output feedback control

In modern control systems, the ability to regulate the behavior of dynamic systems is crucial for a wide range of applications, from robotics and aerospace to industrial processes and automotive systems [137, 138, 139]. At the heart of control theory lies the concept of feedback, where system outputs are used to inform control inputs, guiding the system toward desired performance and stability. One of the fundamental challenges in this field is the output feedback control problem, where control actions are based solely on observable outputs rather than complete state information.

The static output feedback control problem is one of the most well-known problems is control theory and many other control problems can be posed into this framework, see [140, 34] and the surveys [141, 142, 140] (e.g., it has been observed that the design of a dynamic output feedback controller can be reduced to solving a static output feedback control problem [143, 144]). The problem can be simply stated as follows: find a static output feedback control strategy that ensures the closed-loop system is asymptotically stable. This problem is significant because static output feedback controllers are generally more cost-effective to implement and are often more reliable in practice.

A typical output feedback control problem involves designing a control strategy that uses observed outputs to generate control inputs. This is in contrast to state feedback control, where the full state vector is assumed to be known. Output feedback control is inherently more challenging due to the limited information available to the controller and the need to estimate unobserved states or system parameters. The problem becomes even more complex when the system dynamics are nonlinear or uncertain. Consider the continuous time linear system:

$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= Cx, \end{aligned} \tag{7.7}$$

where $x \in \mathbb{R}^{n_x}$ is the state vector, $u \in \mathbb{R}^{n_u}$ is the control input, and $y \in \mathbb{R}^{n_y}$ is the measured output, with A, B, and C being matrices with constant real entries of appropriate dimensions. We say that the system (7.7) is output feedback stabilizable if there exists a static output feedback control law:

$$u = Ky,$$

such that the resulting matrix $A+BKC$ has all its eigenvalues with negative real parts. Matrices that meet this condition are described as stable.

The pair (A, B) is considered stabilizable if the following condition holds:

$$\operatorname{rank}[\lambda I_{n_x} - A \ \ B] = n_x, \quad \forall \lambda \in \sigma(A) \cap \mathbb{C}^-,$$

where $\sigma(A)$ is the spectrum of A and $\mathbb{C}^-$ denotes the closed right half plane. This condition is weaker than the following controllability condition:

$$\operatorname{rank}[\lambda I_{n_x} - A \ \ B] = n_x, \quad \forall \lambda \in \sigma(A),$$

which is equivalent to the Kalman controllability condition:

$$\operatorname{rank}[B \ \ AB \ \ A^2B \ \cdots \ A^{n_x-1}B] = n_x.$$

It is well known that dynamical system (7.7) with $C = I_{n_x}$ (i.e., $y = x$) is stabilizable via a state feedback controller $u = Kx$ if and only if there exist matrices $X \succ 0$ and K, of compatible dimensions, such that:

$$X(A + BK) + (A + BK)^T X \prec 0.$$

Further, multiplying this inequality from both sides by $W = X^{-1}$ we get:

$$(A + BK)W + W(A + BK)^T \prec 0.$$

Now, defining $L = KW$, then we obtain:

$$AW + WA^T + BL + L^T B^T \prec 0.$$

In fact, it is a well-known result [140] that the previous linear matrix inequality (LMI) is feasible in the variables (W, L) if and only if the pair (A, B) is stabilizable, and in this case the state feedback controller $u = LW^{-1}x = Kx$ stabilizes system (7.7). To find a solution to this problem or to declare the problem unfeasible, if solutions do not exist within a given precision, is a simple task that can be easily carried out with efficient convex optimization algorithms [140].

However, when neither C nor B are different from identity matrix, the problem becomes in fact a very difficult one because the resulting matrix inequality is not convex in general as described below. In fact, to the best of our knowledge, conditions for exact stabilizability are unknown in this case. The static output feedback stabilization problem for the system described by equation (7.7) hinges on determining the feasibility of finding a static control law denoted by $u = Ky$, whereby the closed-loop system $\dot{x} = Ax + BKCx = (A + BKC)x$ achieves stability, indicating that the matrix $A + BKC$ is stable [145]. If an output feedback controller exists to satisfy this condition, the system (7.7) is said to be output stabilizable, with the matrix K providing a solution to the problem. The subsequent theorem establishes conditions for this scenario:

Theorem 7.3.1. *(Theorem 1 in [145]) The system (7.7) is static output feedback stabilizable if there exist $X \succ 0$ and K satisfying the following matrix inequality:*

$$(A + BKC)^T X + X(A + BKC) \prec 0. \tag{7.8}$$

We can reformulate this bilinear matrix inequality as an equality by introducing a matrix $Q \succ 0$ such that

$$(A + BKC)^T X + X(A + BKC) + Q = 0.$$

We can solve this bilinear matrix equality by minimizing the norm of

$$F(X, K, Q) := (A + BKC)^T X + X(A + BKC) + Q,$$

which is a second order polynomial in X, K and Q. Thus, the minimization problem to be solved becomes:

$$\min_{X,Q,K} \|F(X, K, Q)\|_F + h(X, Q), \tag{7.9}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix and

$$h(X, Q) = \mathbf{1}_{\mathbb{S}^n_+}(X) + \mathbf{1}_{\mathbb{S}^n_+}(Q),$$

with $\mathbb{S}^n_+$ the cone of positive definite matrices.

We evaluate the performance of RHOTA algorithm for solving static output feedback control problem (7.9) using data from the COMPl$_e$ib library available at [34]. Let us note that $\nabla^2 F$, where F is given in (7.9), is constant, and therefore ∇F is Lipschitz (i.e., $p = 1$). Hence, problem (7.9) is a particular case of the composite problem considered in Chapter 5 and consequently our algorithm RHOTA with $p = 1$ can be used for solving the output feedback control problem with mathematical guarantees for finding stationary points. We can effectively implement RHOTA algorithm by utilizing the Fréchet differentiable of the matrix function F [146]. Thus, at each iteration of RHOTA with $p = 1$, the following convex subproblem needs to be solved:

$$\begin{aligned}(X_{k+1}, K_{k+1}, Q_{k+1}) = \arg\min_{X \succ 0, Q \succ 0, K} &\Big\| F(X_k, K_k, Q_k) + \nabla F(X_k, K_k, Q_k)[X - X_k, K - K_k, Q - Q_k] \Big\|_F \\ &+ \frac{M}{2} \Big\| [X - X_k, K - K_k, Q - Q_k] \Big\|_F^2,\end{aligned}$$

where the directional derivative is:

$$\begin{aligned}&\nabla F(X_k, K_k, Q_k)[X - X_k, K - K_k, Q - Q_k] \\ =&\nabla_X F(X_k, K_k, Q_k)[X - X_k] + \nabla_K F(X_k, K_k, Q_k)[K - K_k] + \nabla_Q F(X_k, K_k, Q_k)[Q - Q_k],\end{aligned}$$

with the following expressions

$$\begin{aligned}\nabla_X F(X_k, K_k, Q_k)[X - X_k] &= (A + BK_kC)^T (X - X_k) + (X - X_k)(A + BK_kC) \\ \nabla_K F(X_k, K_k, Q_k)[K - K_k] &= (B(K - K_k)C)^T X_k + X_k(B(K - K_k)C) \\ \nabla_Q F(X_k, K_k, Q_k)[Q - Q_k] &= Q - Q_k.\end{aligned}$$

We compare RHOTA algorithm for $p = 1$ with BMIsolver [33]. BMIsolver is specifically designed to optimize the spectral abscissa of the closed-loop system $\dot{x} = (A + BKC)x$ (refer to [33] for comprehensive details). In Table 7.1, we report the number of iterations, CPU time, the obtained solution K, and the maximum eigenvalue of the real part of the matrix $A+BKC$ (called spectral abscissa). The stopping criterion utilized is

$$\|F(X_k, K_k, Q_k)\| \leq 10^{-3}$$

and we use CVX to solve the subproblem in RHOTA [136]. Both algorithms, BMIsolver and RHOTA, commence with identical initial values X_0 and K_0. Each test case from COMPl$_e$ib is initialized differently. From the data presented in Table 7.1, it is evident that RHOTA outperforms BMIsolver [33] in terms of both, CPU time and number of iterations. Moreover, RHOTA yields a slightly smaller value for the spectral abscissa, showcasing the efficiency of the proposed method. The superior performance of RHOTA algorithm (in time and iterations) can be attributed to two facts: first, by linearizing inside the norm, RHOTA leverages a portion of the Hessian of the objective, while BMIsolver solely utilizes first-order information; second, our formulation (7.9) satisfies the KL property (2.16) (as composition of semi-algebraic functions, i.e., the 2-norm and quadratic functions, see [46] and also [146]), which, according to Theorem 4.3.5, ensures fast convergence for RHOTA.

	RHOTA ($p=1$)				BMIsolver [33]			
	iter	time(s)	K	MEV	iter	time(s)	K	MEV
ac3 ($n_x=5$)	**4**	**1.08**	$\begin{pmatrix} 2.7633 & -0.4060 & -2.6203 & -0.0605 \\ -0.1880 & 1.5857 & -3.5001 & 1.8552 \end{pmatrix}$	**-0.89**	29	18	$\begin{pmatrix} 2.9051 & -0.4423 & -2.7215 & 0.0038 \\ -0.1084 & 1.7357 & -3.3988 & 1.9438 \end{pmatrix}$	-0.85
ac8 ($n_x=9$)	**6**	**1.8**	$\begin{pmatrix} 1.0279 & -0.4365 & -1.15850.0085 & 0.46237 \end{pmatrix}$	**-0.44**	43	5.6	$\begin{pmatrix} 1.0279 & -0.4365 & -1.15850.0085 & 0.46237 \end{pmatrix}$	**-0.44**
cm1_is ($n_x=20$)	**12**	**5.7**	$\begin{pmatrix} -19.98 \\ -10.97 \end{pmatrix}$	**-4.3e^{-3}**	22	10.6	$\begin{pmatrix} -17.85 \\ -22 \end{pmatrix}$	-4.2e^{-3}
cm2_is ($n_x=60$)	**19**	**533**	$\begin{pmatrix} -5 \\ -7.87 \end{pmatrix}$	**-1.07e^{-2}**	114	2691	$\begin{pmatrix} -5.6 \\ -7.18 \end{pmatrix}$	-1.02e^{-2}
dis1 ($n_x=8$)	**24**	**7.6**	$\begin{pmatrix} 3.125 & 2.817 & -7.584 & -5.446 \\ 2.817 & 3.71 & -4.244 & -4.256 \\ -7.584 & -4.244 & -0.435 & 1.352 \\ -5.446 & -4.256 & 1.352 & 2.877 \end{pmatrix}$	**-1.363**	100	13.4	$\begin{pmatrix} 3.125 & 2.817 & -7.584 & -5.446 \\ 2.817 & 3.71 & -4.244 & -4.256 \\ -7.584 & -4.244 & -0.435 & 1.352 \\ -5.446 & -4.256 & 1.352 & 2.877 \end{pmatrix}$	**-1.363**
dlr2 ($n_x=40$)	**7**	**21.6**	$\begin{pmatrix} 1.85 & 1.06 \\ -0.27 & -2.09 \end{pmatrix}$	**-5e^{-3}**	120	477	$\begin{pmatrix} -5.6 & 5.5 \\ 5.5 & -1.4 \end{pmatrix}$	**-5e^{-3}**
eb1 ($n_x=10$)	**8**	**3.5**	-0.551	**-0.066**	26	9.2	-47.377	-0.0212
he1 ($n_x=4$)	**5**	**1.4**	$\begin{pmatrix} 0.981 \\ 4.469 \end{pmatrix}$	-0.22	12	2.1	$\begin{pmatrix} 0.883 \\ 4.022 \end{pmatrix}$	-0.22
he4 ($n_x=8$)	**15**	**8.5**	$\begin{pmatrix} -2.12 & 3.87 & 1.47 & -0.26 & -0.04 & 0.65 \\ 3.75 & -14.55 & -1.48 & 1.14 & 5.35 & 2.03 \\ -0.67 & 2.20 & 2.23 & 0.07 & -2.79 & -0.14 \\ -7.98 & -3.28 & -12.94 & -0.12 & 5.37 & 0.23 \end{pmatrix}$	**-0.77**	89	40.5	$\begin{pmatrix} -2.12 & 3.87 & 1.47 & -0.26 & -0.04 & 0.65 \\ 3.75 & -14.55 & -1.48 & 1.14 & 5.35 & 2.03 \\ -0.67 & 2.20 & 2.23 & 0.07 & -2.79 & -0.14 \\ -7.98 & -3.28 & -12.94 & -0.12 & 5.37 & 0.23 \end{pmatrix}$	**-0.77**
hf2d_is5 ($n_x=5$)	**5**	**1.4**	$\begin{pmatrix} 5.8 & 2.7 & 0.08 & -0.28 \\ -1.18 & -1.07 & 1.41 & 2.04 \end{pmatrix}$	**-5.17**	14	3.5	$\begin{pmatrix} 5.8 & 2.7 & 0.08 & -0.28 \\ -1.18 & -1.07 & 1.41 & 2.04 \end{pmatrix}$	**-5.17**
hf2d_cd4 ($n_x=7$)	**6**	**1.8**	$\begin{pmatrix} -3.2 & -3.7 \\ -3.5 & -3.9 \end{pmatrix}$	**-2.5**	78	17	$\begin{pmatrix} -3.3 & -4.3 \\ -4.3 & -5.5 \end{pmatrix}$	-2.48
hf2d_cd5 ($n_x=7$)	**8**	**2.5**	$\begin{pmatrix} -0.23 & -0.21 \\ -1.38 & -0.43 \end{pmatrix}$	**-1.79**	257	19.6	$\begin{pmatrix} -0.57 & -1.54 \\ -1.54 & -3.65 \end{pmatrix}$	-1.37
je2 ($n_x=21$)	**16**	**11.6**	$\begin{pmatrix} 1.328 & 0.087 & -0.090 \\ -1.462 & 0.1918 & 1.927 \\ 1.893 & 0.4696 & 2.7049 \end{pmatrix}$	**-2.51**	47	56.5	$\begin{pmatrix} 1.328 & 0.087 & -0.090 \\ -1.462 & 0.1918 & 1.927 \\ 1.893 & 0.4696 & 2.7049 \end{pmatrix}$	**-2.51**
lah ($n_x=48$)	**5**	**51**	-6	**-2.69**	99	1037.3	-5.99	**-2.69**
real1 ($n_x=4$)	**5**	**1.4**	$\begin{pmatrix} -1.740 & 4.229 & -2.175 \\ 5.147 & -16.347 & 6.728 \end{pmatrix}$	**-3**	22	3.5	$\begin{pmatrix} -1.740 & 4.229 & -2.175 \\ 5.147 & -16.347 & 6.728 \end{pmatrix}$	**-3**
wec2 ($n_x=10$)	**14**	**8.5**	$\begin{pmatrix} -1.0733 & -0.34109 & 0.9588 & 0.0988 \\ 0.1757 & -0.1420 & -1.39116 & -0.10933 \\ 0.9581 & 0.80115 & 0.19483 & 0.66336 \end{pmatrix}$	**-1.3796**	40	28.9	$\begin{pmatrix} 0.2788 & 0.09640 & 34.9399 & 0.06837 \\ -0.3283 & -0.1234 & -0.07736 & -0.00402 \\ 0.0329 & 8.6410 & 1.3824 & 0.79048 \end{pmatrix}$	-0.6829

Table 7.1: Performance of RHOTA algorithm for $p=1$ and BMIsolver [33] using data from COMPl$_e$ib library $\big($MEV = maximum eigenvalue of the real part of $(A+BKC)\big)$.

7.4 Conclusions

We have applied the algorithms proposed in the previous chapters to solve various problems arising in complex systems, including power flow analysis, phase retrieval, and output feedback control, using real data. Our numerical simulations demonstrate the efficiency of these algorithms in real-world applications. Notably, our results show that these algorithms outperform some state-of-the-art methods and solvers from existing literature in terms of CPU time and/or the number of iterations.

8 Conclusions and future work

Composite optimization problems, particularly those where the merit function exhibits only convexity, pose significant hurdles in terms of finding solutions. The inherent complexity in such scenarios often leads to computational inefficiencies and limits the practical applicability of existing algorithms. However, by introducing additional assumptions on the merit function, we can alleviate some of these challenges and facilitate the resolution of these problems. By leveraging these assumptions, in this thesis we have developed novel methods that streamline the optimization process and enable more efficient and effective solutions to be obtained.

At the forefront of our contributions is the development of higher-order methods for solving the intricacies of composite optimization problems under additional assumptions on the merit function. Our algorithms represent a significant advancement in the field, offering enhanced precision and efficacy in solving complex optimization tasks. Through meticulous experimentation and theoretical analysis, we have demonstrated the effectiveness of our numerical approaches in achieving global convergence and improving computational efficiency. We have demonstrated the implementability of the proposed algorithms in some particular settings, which ensures their practical viability in real-world applications. By bridging the gap between theory and practice, our research strives to make substantial contributions to the field of optimization, facilitating the resolution of challenging composite optimization problems in diverse domains.

In summary, as potential extensions, delving deeper into the following points holds promise:

- In Chapter 3, an intriguing opportunity presents itself for further refinement: exploring avenues to enhance the GCHO method within convex settings. For example, the incorporation of a Nesterov momentum step holds promise in potentially accelerating the convergence rate of GCHO. Such enhancements could not only broaden the applicability of the algorithm but also contribute to advancing the field of optimization by addressing challenges specific to convex scenarios.

- In Chapters 3 and 5, an interesting avenue for exploration emerges: investigating whether convergence in the sequence can be derived for our algorithms under the KL property. By shedding light on this aspect, we can deepen our understanding of the algorithms' performance and potentially uncover new insights into their applicability across diverse optimization landscapes.

- In Chapter 4, an open question remains: can one derive a theoretical convergence rate for an accelerated MTA incorporating the Nesterov momentum step? This query delves into the realm of optimization theory, probing the potential for further enhancing the efficiency and efficacy of MTA method. Clarifying this aspect could shed light on the algorithm's behavior and pave the way for more faster optimization strategies for solving convex composite problems.

- Given that the convergence rates discussed in both Chapter 3 and Chapter 5 hinge on an inexact solution necessitating the computation of a global minimum of a nontrivial Taylor approximation within a ball, a compelling avenue for future exploration arises: is it feasible to relax this condition? This inquiry delves into the intricacies of optimization methodology, questioning the necessity of strict requirements for achieving convergence.

- In Chapters 3, 4, and 5, we present an efficient implementation of the proposed methods for the case $p = 2$. It would be interesting to explore the possibility of solving the subproblems efficiently for $p \geq 3$. By delving into this aspect, we can potentially extend the applicability of our algorithms using higher-order information, thus enriching the field of optimization with enhanced methodologies.

- Furthermore, an promising direction for future research lies in extending all the obtained results to a stochastic framework, as discussed for example in [147]. This endeavor holds promise for broadening the applicability of our findings, potentially unlocking new insights into optimization methodologies in stochastic setting with wide range of applications in machine learning.

Bibliography

[1] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.

[2] W. N. Goetzmann, S. J. Brown, M. J. Gruber, and E. J. Elton, "Modern portfolio theory and investment analysis", *John Wiley & Sons*, vol. 237, 2014.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[4] A. Cauchy *et al.*, "Méthode générale pour la résolution des systemes d'équations simultanées", *Comp. Rend. Sci. Paris*, vol. 25, no. 1847, pp. 536–538, 1847.

[5] B. T. Polyak., "Gradient methods for minimizing functionals", *[in Rus- sian]. Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, vol. 3, no. 4, pp. 643–653, 1963.

[6] N. Z. Shor, "Minimization methods for non-differentiable functions", *Springer Science & Business Media*, vol. 3, 2012.

[7] J. M. Altschuler and P. A. Parrilo, "Acceleration by stepsize hedging i: Multi-step descent and the silver stepsize schedule", *arXiv preprint: 2309.07879*, 2023.

[8] Y. Nesterov, "Gradient methods for minimizing composite functions", *Mathematical programming*, vol. 140, no. 1, pp. 125–161, 2013.

[9] N. Doikov, "New second-order and tensor methods in convex optimization", *PhD thesis, Universite catholique de Louvain*, 2021.

[10] Y. Nesterov and B. Polyak, "Cubic regularization of newton method and its global performance", *Mathematical Programming*, no. 108, pp. 177–205, 2006.

[11] M. Baes, "Estimate sequence methods: Extensions and approximations", *Institute for Operations Research, ETH, Zürich, Switzerland*, vol. 2, no. 1, 2009.

[12] Y. Nesterov, "Implementable tensor methods in unconstrained convex optimization", *Mathematical Programming*, vol. 186, pp. 157–183, 2021.

[13] E. G. Birgin, J. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint, "Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models", *Mathematical Programming*, vol. 163, pp. 359–368, 2017.

[14] C. Cartis, N. I. Gould, and P. L. Toint, "Universal regularization methods: Varying the power, the smoothness and the accuracy", *SIAM Journal on Optimization*, vol. 29, no. 1, pp. 595–615, 2019.

[15] B. Polyak, "Gradient methods for the minimisation of functionals", *Computational Mathematics and Mathematical Physics*, vol. 3, no. 4, pp. 864–878, 1963.

[16] A. Beck, "First-order methods in optimization", *SIAM*, vol. 25, pp. 49–58, 2017.

[17] L. Bottou, "Large-scale machine learning with stochastic gradient descent", in *IEEE International Conference on Computational Statistics*, IEEE, 2010.

[18] W. P.W. and L. C-J., "Iteration complexity of feasible descent methods for convex optimization", *Journal of Machine Learning Research*, vol. 15, no. 4, pp. 1523–1548, 2023.

[19] Y. Nesterov, "Inexact basic tensor methods for some classes of convex optimization problems", *Optimization Methods and Software*, vol. 37, no. 3, pp. 878–906, 2022.

[20] I. Necoara and D. Lupu, "General higher-order majorization-minimization algorithms for (non) convex optimization", *arXiv preprint arXiv:2010.13893*, 2020.

[21] Z. C. J. Bolte and E. Pauwels, "The multiproximal linearization method for convex composite problems", *Mathematical Programming*, vol. 182, pp. 1–36, 2020.

[22] N. Doikov and Y. Nesterov, "High-order optimization methods for fully composite problems", *SIAM Journal on Optimization*, vol. 32, no. 3, pp. 2402–2427, 2022.

[23] Y. Nabou and I. Necoara, "Efficiency of higher-order algorithms for minimizing composite functions", *Computational Optimization and Applications*, vol. 87, no. 2, pp. 441–473, 2024.

[24] R. S. A. Auslender and M. Teboulle, "A moving balls approximation method for a class of smooth constrained minimization problems", *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3232–3259, 2010.

[25] T. P. P. Yu and Z. Lu, "Convergence rate analysis of a sequential convex programming method with line search for a class of constrained difference-of-convex optimization problems", *SIAM Journal on Optimization*, vol. 31, no. 3, pp. 2024–2054, 2011.

[26] Y. Nabou and I. Necoara, "Moving higher-order taylor approximations method for smooth constrained minimization problems", *arXiv preprint arXiv:2402.15022*, 2024.

[27] Y. Nabou and I. Necoara, "Regularized higher-order taylor approximation methods for composite nonlinear least-squares", *To be submited*, 2024.

[28] Y. Nabou, F. Glineur, and I. Necoara, "Proximal gradient methods with inexact oracle of degree q for composite optimization", *Optimization letters*, 2024. DOI: https://doi.org/10.1007/s11590-024-02118-9.

[29] N. Costilla-Enriquez, Y. Weng, and B. Zhang, "Combining newton-raphson and stochastic gradient descent for power flow analysis", *IEEE Transactions on Power Systems*, vol. 36, no. 1, pp. 514–517, 2020.

[30] X. L. E. J. Candès and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms", *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.

[31] J. C. Duchi and F. Ruan, "Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval", *Information and Inference: A Journal of the IMA*, vol. 8, pp. 471–529, 2019.

[32] G. B. G. G. Wang 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, 2017.

[33] W. M. Q. T. Dinh S. Gumussoy and M. Diehl, "Combining convex–concave decompositions and linearization approaches for solving bmis, with application to static output feedback", *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1377–1390, 2011.

[34] F. Leibfritz, "Compleib, constraint matrix-optimization problem library-a collection of test examples for nonlinear semidefinite programs, control system design and related problems", *Dept. Math., Univ. Trier, Trier, Germany, Tech. Rep*, vol. 2004, 2004.

[35] Y. Nabou, L. Toma, and I. Necoara, "Modified projected gauss-newton method for constrained nonlinear least-squares: Application to power flow analysis", in *2023 European Control Conference (ECC)*, IEEE, 2023, pp. 1–6.

[36] *Illinois Center, Smarter Electric Grid*, http://publish.illinois.edu/smartergrid/.

[37] *MNIST dataset*, https://www.kaggle.com/datasets/scolianni/mnistasjpg.

[38] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2003, vol. 87.

[39] R. T. Rockafellar and R. J.-B. Wets, *Variational analysis*. Springer Science & Business Media, 2009, vol. 317.

[40] A. Rodomanov and Y. Nesterov, "Smoothness parameter of power of euclidean norm", *Journal of Optimization Theory and Applications*, vol. 185, pp. 303–326, 2020.

[41] N. Doikov and Y. E. Nesterov, "Inexact tensor methods with dynamic accuracies", in *37th Internationa Conferenceon Machine Learning*, 2020, pp. 2577–2586.

[42] P. Ochs, *Long term motion analysis for object level grouping and nonsmooth optimization methods*. 2015.

[43] B. Mordukhovich, *Variational analysis and generalized differentiation I. Basic theory*. Springer, 2006.

[44] J.-B. Hiriart-Urruty, "New concepts in nondifferentiable programming", *Memoires de la Societe Mathematique de France*, vol. 60, pp. 57–85, 1979.

[45] J. Bolte, A. Daniilidis, and A. Lewis, "The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems", *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 1205–1223, 2007.

[46] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems", *Mathematical Programming*, vol. 146, no. 1, pp. 459–494, 2014.

[47] J. Bolte, A. Daniilidis, and A. Lewis, "The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems", *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 1205–1223, 2007.

[48] Z. Wang, "The exact modulus of the generalized kurdyka-łojasiewicz property", Ph.D. dissertation, University of British Columbia, 2020. DOI: http://dx.doi.org/10.14288/1.0392646.

[49] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.

[50] R. Andreani, J. M. Martínez, L. T. Santos, and B. F. Svaiter, "On the behaviour of constrained optimization methods when lagrange multipliers do not exist", *Optimization Methods and Software*, vol. 29, no. 3, pp. 646–657, 2014.

[51] R. Andreani, J. M. Martínez, and B. F. Svaiter, "A new sequential optimality condition for constrained optimization and algorithmic consequences", *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3533–3554, 2010.

[52] H. Attouch and J. Bolte, "On the convergence of the proximal algorithm for nonsmooth functions involving analytic features", *Mathematical Programming*, vol. 116, pp. 5–16, 2009.

[53] K. Lange, D. R. Hunter, and I. Yang, "Optimization transfer using surrogate objective functions", *Journal of computational and Graphical Statistics*, vol. 9, no. 1, pp. 1–20, 2000.

[54] J. Mairal, "Optimization with first-order surrogate functions", in *International Conference on Machine Learning*, 2013, pp. 783–791.

[55] 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, 2013.

[56] D. Drusvyatskiy and C. Paquette, "Efficiency of minimizing compositions of convex functions and smooth maps", *Mathematical Programming*, vol. 178, pp. 503–558, 2019.

[57] C. Li and K. Ng, "Majorizing functions and convergence of the gauss-newton method for convex composite optimization", *SIAM Journal on Optimization*, vol. 18, no. 2, pp. 613–642, 2007.

[58] Y. Nesterov, "Smooth minimization of non-smooth functions", *Mathematical programming*, vol. 103, pp. 127–152, 2005.

[59] J. Mairal, "Incremental majorization-minimization optimization with application to large-scale machine learning", *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 829–855, 2015.

[60] Y. S. J. D. Lee and M. A. Saunders, "Proximal newtontype methods for minimizing composite functions", *SIAM Journal on Optimization*, vol. 24, no. 3, pp. 1420–1443, 2014.

[61] A. R. Conn, N. I. Gould, and P. L. Toint, *Trust region methods*. SIAM, 2000.

[62] K. Mishchenko, "Regularized newton method with global convergence", *SIAM Journal on Optimization*, vol. 33, no. 3, pp. 1440–1462, 2023.

[63] N. Doikov and Y. Nesterov, "Gradient regularization of newton method with bregman distances", *Mathematical programming*, pp. 1–25, 2023.

[64] S. Bubeck, "Convex optimization: Algorithms and complexity", *Foundations and Trends in Machine Learning*, vol. 8, no. 3-4, pp. 231–357, 2025.

[65] E. Pauwels, "The value function approach to convergence analysis in composite optimization", *Operations Research Letters*, vol. 44, no. 6, pp. 790–795, 2016.

[66] Y. Nesterov and A. Nemirovskii, "Interior-point polynomial algorithms in convex programming", *SIAM Studies in Applied Mathematics*, vol. 13, 1994.

[67] G. N. Grapiglia and Y. Nesterov, "Tensor methods for minimizing convex functions with hölder continuous higher-order derivatives", *SIAM Journal on Optimization*, vol. 30, no. 4, pp. 2750–2779, 2020.

[68] B. G. J. More and K. Hillstrom, "Testing unconstrained optimization software", *ACM Tra. on Math. Sof*, vol. 7, no. 1, pp. 17–41, 1981.

[69] T. R. N.I.M. Gould and J. Scott, "Convergence and evaluation-complexity analysis of a regularized tensor-newton method for solving nonlinear least-squares problems", *Computational Optimization and Applications*, vol. 73, no. 1, pp. 1–35, 2019.

[70] A. Wächter and L. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming", *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[71] J. M. E.G. Birgin J.L. Gardenghi and S. Santos, "On the use of third-order models with fourth-order regularization for unconstrained optimization", *Optimization Letters*, vol. 14, pp. 815–838, 2020.

[72] D. P. Bertsekas, "Nonlinear programming", *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.

[73] J. Bandler, "Optimization of design tolerances using nonlinear programming", *Journal of Optimization Theory and Applications*, vol. 14, pp. 99–114, 1974.

[74] M. Chiang, C. W. Tan, D. P. Palomar, D. O'neill, and D. Julian, "Power control by geometric programming", *IEEE transactions on wireless communications*, vol. 6, no. 7, pp. 2640–2651, 2007.

[75] D. P. Palomar and M. Chiang, "Alternative decompositions for distributed maximization of network utility: Framework and applications", in *INFOCOM*, 2006.

[76] P. Rigollet and X. Tong, "Neyman-pearson classification, convexity and stochastic constraints", *Journal of machine learning research*, vol. 12, pp. 2831–2855, 2011.

[77] A. J. Santos and G. Da Costa, "Optimal-power-flow solution by newton's method applied to an augmented lagrangian function", *IEE Proceedings-Generation, Transmission and Distribution*, vol. 142, no. 1, pp. 33–36, 1995.

[78] P. Shen and K. Zhang, "Global optimization of signomial geometric programming using linear relaxation", *Applied Mathematics and Computation*, vol. 150, no. 1, pp. 99–114, 2004.

[79] F. Messerer, K. Baumgärtner, and M. Diehl, "Survey of sequential convex programming and generalized gauss-newton methods", *ESAIM: Proceedings and Surveys*, vol. 71, pp. 64–88, 2021.

[80] S. Boyd and L. Vandenbergh, *Convex optimization.* 2020, vol. 35, pp. 243–256.

[81] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods.* Academic press, 2014.

[82] M. R. Hestenes, "Multiplier and gradient methods", *Journal of optimization theory and applications*, vol. 4, no. 5, pp. 303–320, 1969.

[83] E. G. Birgin and J. M. Martínez, *Practical augmented Lagrangian methods for constrained optimization.* SIAM, 2014.

[84] S. Lu, "A single-loop gradient descent and perturbed ascent algorithm for nonconvex functional constrained optimization", in *International Conference on Machine Learning*, PMLR, 2022, pp. 14 315–14 357.

[85] V. Nedelcu, I. Necoara, and Q. Tran-Dinh, "Computational complexity of inexact gradient augmented lagrangian methods: Application to constrained mpc", *SIAM Journal on Control and Optimization*, vol. 52, no. 5, pp. 3109–3134, 2014.

[86] I. Necoara and V. Nedelcu, "Rate analysis of inexact dual first-order methods application to dual decomposition", *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1232–1243, 2013.

[87] X.-W. Liu, Y.-H. Dai, Y.-K. Huang, and J. Sun, "A novel augmented lagrangian method of multipliers for optimization with general inequality constraints", *Mathematics of Computation*, vol. 92, no. 341, pp. 1301–1330, 2023.

[88] E. Cohen, N. Hallak, and M. Teboulle, "A dynamic alternating direction of multipliers for nonconvex minimization with nonlinear functional equality constraints", *Journal of Optimization Theory and Applications*, pp. 1–30, 2022.

[89] J. Bolte and E. Pauwels, "Majorization-minimization procedures and convergence of sqp methods for semi-algebraic and tame programs", *Mathematics of Operations Research*, vol. 41, no. 2, pp. 442–465, 2016.

[90] N. K. Singh and I. Necoara, "A stochastic moving ball approximation method for smooth convex constrained minimization", *arXiv preprint arXiv:2402.15016*, 2024.

[91] E. G. Birgin, J. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint, "Evaluation complexity for nonlinear constrained optimization using unscaled kkt conditions and high-order models", *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 951–967, 2016.

[92] J. M. Martínez, "On high-order model regularization for constrained optimization", *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2447–2458, 2017.

[93] C. Cartis, N. I. Gould, and P. L. Toint, *Evaluation Complexity of Algorithms for Nonconvex Optimization: Theory, Computation and Perspectives.* SIAM, 2022.

[94] Y. Nesterov, "Modified gauss–newton scheme with worst case guarantees for global performance", *Optimisation methods and software*, vol. 22, no. 3, pp. 469–483, 2007.

[95] S. Lu, "A single-loop gradient descent and perturbed ascent algorithm for nonconvex functional constrained optimization", in *International Conference on Machine Learning*, PMLR, 2022, pp. 14 315–14 357.

[96] M. F. Sahin, A. Alacaoglu, F. Latorre, V. Cevher, *et al.*, "An inexact augmented lagrangian framework for nonconvex optimization with nonlinear constraints", *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[97] C. Cartis, N. I. Gould, and P. L. Toint, "A concise second-order complexity analysis for unconstrained optimization using high-order regularized models", *Optimization Methods and Software*, vol. 35, no. 2, pp. 243–256, 2020.

[98] J. Bolte, S. Sabach, and M. Teboulle, "Nonconvex lagrangian-based optimization: Monitoring schemes and global convergence", *Mathematics of Operations Research*, vol. 43, no. 4, pp. 1210–1232, 2018.

[99] D. Zhou, P. Xu, and Q. Gu, "Stochastic variance-reduced cubic regularized newton methods", in *International Conference on Machine Learning*, PMLR, 2018, pp. 5990–5999.

[100] F. Messerer, K. Baumgartner, and M. Diehl, "Survey of sequential convex programming and generalized gauss-newton methods", *ESAIM: Proceedings and Surveys*, vol. 71, pp. 64–88, 2021.

[101] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables.* SIAM, 2000.

[102] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters", *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[103] D. D. Morrison, "Methods for nonlinear least squares problems and convergence proofs", in *JPL SEMINAR PROC. ON TRACKING PROGRAMS AND ORBIT DETERMINATION 26 FEB. 1960*, 1960.

[104] K. Levenberg, "A method for the solution of certain problems in least squares. quart. appl. math. 2", 1944.

[105] Y. Yuan, "Conditions for convergence of trust-region algorithms for nonsmooth optimization", *Mathematical Programming*, vol. 31, pp. 220–228, 1985.

[106] R. Fletcher, "A model algorithm for composite ndo problems", *Mathematical Programming Studies*, vol. 17, pp. 67–76, 1982.

[107] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: Proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods", *Mathematical Programming*, vol. 137, no. 1, pp. 91–129, 2013.

[108] C. Cartis, N. I. Gould, and P. L. Toint, "On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming", *SIAM Journal on Optimization*, vol. 21, no. 4, pp. 1721–1739, 2011.

[109] C. Cartis, N. I. Gould, and P. L. Toint, "On the evaluation complexity of cubic regularization methods for potentially rank-deficient nonlinear least-squares problems and its relevance to constrained nonlinear optimization", *SIAM Journal on Optimization*, vol. 23, no. 3, pp. 1553–1574, 2013.

[110] E. G. Birgin, J. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint, "Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models", *Mathematical Programming*, vol. 163, pp. 359–368, 2017.

[111] Y. Xie and S. J. Wright, "Complexity of proximal augmented lagrangian for nonconvex optimization with nonlinear equality constraints", *Journal of Scientific Computing*, vol. 86, pp. 1–30, 2021.

[112] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$", *Doklady Akademii Nauk SSSR*, vol. 269, no. 3, p. 543, 1983.

[113] L. Bogolubsky, P. Dvurechenskii, A. Gasnikov, G. Gusev, Y. Nesterov, A. M. Raigorodskii, A. Tikhonov, and M. Zhukovskii, "Learning supervised pagerank with gradient-based and gradient-free optimization methods", vol. 29, 2016.

[114] M. Cohen, J. Diakonikolas, and L. Orecchia, "On acceleration with noise-corrupted gradients", in *International Conference on Machine Learning*, PMLR, 2018, pp. 1019–1028.

[115] F. G. O. Devolder and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle", *Mathematical programming*, vol. 146, pp. 37–75, 2013.

[116] P. Dvurechensky and A. Gasnikov, "Stochastic intermediate gradient method for convex problems with stochastic inexact oracle", *Journal of Optimization Theory and Applications*, vol. 171, no. 1, pp. 121–145, 2016.

[117] P. Dvurechensky, "A gradient method with inexact oracle for composite nonconvex optimization", *Computer Research and Modeling*, vol. 14, no. 2, 2022.

[118] F. S. Stonyakin, D. Dvinskikh, P. Dvurechensky, A. Kroshnin, O. Kuznetsova, A. Agafonov, A. Gasnikov, A. Tyurin, C. A. Uribe, D. Pasechnyuk, *et al.*, "Gradient methods for problems with inexact model of the objective", in *Mathematical Optimization Theory and Operations Research: 18th International Conference, MOTOR 2019, Ekaterinburg, Russia, July 8-12, 2019, Proceedings 18*, Springer, 2019, pp. 97–114.

[119] Q. Tran-Dinh, I. Necoara, and M. Diehl, "Fast inexact decomposition algorithms for large-scale separable convex optimization", *Optimization*, vol. 65, no. 2, pp. 325–356, 2016.

[120] A. d'Aspremont, "Smooth optimization with approximate gradient", *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1171–1183, 2008.

[121] A. Agafonov, D. Kamzolov, P. Dvurechensky, A. Gasnikov, and M. Takac, "Inexact tensor methods and their application to stochastic convex optimization", *Optimization Methods and Software*, 2017.

[122] A. Geary and D. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization", 1999.

[123] A. T. L. Stella and P. Patrinos, "Forward-backward quasi-newton methods for nonsmooth optimization problems", *Computational Optimization and Applications*, vol. 67, pp. 443–487, 2017.

[124] R. Tibshirani, "Regression shrinkage and selection via the lasso", *Journal of the Royal Statistical Society*, vol. 58, no. 1, pp. 267–288, 1996.

[125] H. L. Nguyen, "Newton-raphson method in complex form", *IEEE Power Engineering Review*, vol. 17, no. 8, pp. 62–62, 1997.

[126] B. Stott, "Review of load-flow calculation methods", *Proceedings of the IEEE*, vol. 62, no. 7, pp. 916–929, 1974.

[127] A. L. Patterson, "A fourier series method for the determination of the components of interatomic distances in crystals", *Physical Review*, vol. 46, no. 5, p. 372, 1934.

[128] A. L. Patterson, "Ambiguities in the x-ray analysis of crystal structures", *Physical Review*, vol. 65, no. 5-6, p. 195, 1944.

[129] 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, 2015.

[130] A. Walther, "The question of phase retrieval in optics", *Optica Acta: International Journal of Optics*, vol. 10, no. 1, pp. 41–49, 1963.

[131] R. P. Millane, "Phase retrieval in crystallography and optics", *JOSA A*, vol. 7, no. 3, pp. 394–411, 1990.

[132] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy", *Image recovery: theory and application*, vol. 231, p. 275, 1987.

[133] L. R. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. Tsinghua University Press, 1999.

[134] M. Stefik, "Inferring dna structures from segmentation data", *Artificial Intelligence*, vol. 11, no. 1-2, pp. 85–114, 1978.

[135] Y. Nesterov, "Inexact accelerated high-order proximal-point methods", *Mathematical Programming*, pp. 1–26, 2023.

[136] M. Grant, S. Boyd, and Y. Ye, *Cvx: Matlab software for disciplined convex programming*, 2008.

[137] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic, *Biped locomotion: dynamics, stability, control and application*. Springer Science & Business Media, 2012, vol. 7.

[138] S. H. Lane and R. F. Stengel, “Flight control design using non-linear inverse dynamics”, *Automatica*, vol. 24, no. 4, pp. 471–483, 1988.

[139] M. Janakiram and J. B. Keats, “Combining spc and epc in a hybrid industry”, *Journal of Quality Technology*, vol. 30, no. 3, pp. 189–200, 1998.

[140] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[141] D. S. Bernstein, “Some open problems in matrix theory arising in linear systems and control”, *Linear Algebra and its Applications*, vol. 162, pp. 409–432, 1992.

[142] V. L. Syrmos, C. T. Abdallah, P. Dorato, and K. Grigoriadis, “Static output feedback—a survey”, *Automatica*, vol. 33, no. 2, pp. 125–137, 1997.

[143] C. Nett, D. Bernstein, and W. Haddad, “Minimal complexity control law synthesis, part 1: Problem formulation and reduction to optimal static output feedback”, in *1989 American Control Conference*, IEEE, 1989, pp. 2056–2064.

[144] B. Mårtensson, “The order of any stabilizing regulator is sufficient a priori information for adaptive stabilization”, *Systems & Control Letters*, vol. 6, no. 2, pp. 87–91, 1985.

[145] Y.-Y. Cao, J. Lam, and Y.-X. Sun, “Static output feedback stabilization: An ilmi approach”, *Automatica*, vol. 34, no. 12, pp. 1641–1645, 1998.

[146] I. Fatkhullin and B. Polyak, “Optimizing static linear feedback: Gradient method”, *SIAM Journal on Control and Optimization*, vol. 59, no. 5, pp. 3887–3911, 2021.

[147] D. Lupu and I. Necoara, “Convergence analysis of stochastic higher-order majorization–minimization algorithms”, *Optimization Methods and Software*, pp. 1–30, 2023.